

LS-DYNA Software Revision 13.0.0

Release Notes



Oasys

LS-DYNA ENVIRONMENT

Release Notes for R13.0.0

September 8, 2021

Contents

1 Release notes revision	3
2 License	3
3 Documentation	4
4 Notes	4
5 New Features	4
5.1 Acoustics, NVH, and Frequency Domain	4
5.2 ALE	5
5.3 Constraint Cards	5
5.4 Control Cards	6
5.5 Discrete Element Method	6
5.6 Dual CESE Compressible Fluid Solver	6
5.6.1 Dual CESE Compressible Multiphase CFD Solvers	8
5.7 Elements	10
5.8 EM (Electromagnetic Solver)	11
5.9 ICFD (Incompressible Fluid Solver)	12
5.10 Implicit (Mechanical) Solver	12
5.11 Isogeometric Analysis (IGA)	12
5.12 Load Cards	13
5.13 Materials and Equations-of-State	13
5.14 MPP	14
5.15 Multiscale	15
5.16 Output	15
5.17 Peridynamics	15

5.18	SPG (Smooth Particle Galerkin)	16
5.19	SPH	17
5.20	Thermal Solver	17
5.21	XFEM (eXtended Finite Element Method)	17
6	Improvements and Enhancements	18
6.1	Acoustics, NVH, and Frequency Domain	18
6.2	Adaptivity	19
6.3	Airbag Cards	20
6.4	ALE	20
6.5	*BATTERY (Electrochemistry Solver)	23
6.6	Blast	24
6.7	Boundary Cards	24
6.8	CESE Compressible Fluid Solver	25
6.9	*CHEMISTRY	25
6.10	Constraint Cards	25
6.11	Contact	26
6.12	Control Cards	28
6.13	Discrete Element Method	29
6.14	Dual CESE Compressible Fluid Solver	30
6.15	Elements	32
6.16	EFG (Element Free Galerkin)	33
6.17	EM (Electromagnetic Solver)	33
6.18	Fatigue Cards	33
6.19	Forming Analysis	34
6.20	ICFD (Incompressible Fluid Solver)	35
6.21	Implicit (Mechanical) Solver	36
6.22	Initial Cards	37
6.23	Isogeometric Analysis (IGA)	38
6.24	Load Cards	39
6.25	Materials and Equations-of-State	39
6.26	MPP	46
6.27	Multiscale	46
6.28	Output	47
6.29	Restarts	48
6.30	Sensors	48
6.31	SPG (Smooth Particle Galerkin)	48
6.32	SPH	49
6.33	Thermal Solver	49
6.34	Miscellaneous	50

7 Bug Fixes	52
7.1 Acoustics, NVH, and Frequency Domain	52
7.2 Adaptivity	53
7.3 Airbag Cards	54
7.4 ALE	54
7.5 Boundary Cards	55
7.6 CESE Compressible Fluid Solver	56
7.7 *CHEMISTRY	56
7.8 Constraint Cards	56
7.9 Contact	57
7.10 Control Cards	60
7.11 Discrete Element Method	62
7.12 Elements	62
7.13 Initial Cards	63
7.14 Implicit (Mechanical) Solver	63
7.15 Isogeometric Analysis (IGA)	65
7.16 Load Cards	65
7.17 Materials and Equations-of-State	65
7.18 MPP	70
7.19 Multiscale	71
7.20 Output	71
7.21 Restarts	73
7.22 Sensors	74
7.23 SPH	74
7.24 Thermal Solver	75
7.25 XFEM (eXtended Finite Element Method)	75
7.26 Miscellaneous	75

§1 Release notes revision

This file constitutes revision 0 of the release notes for LS-DYNA version R13.0.0.

§2 License

The string "REVISION 13" must appear in the LS-DYNA license file in order to run version R13. Please contact your LS-DYNA distributor or your Ansys sales representative if you have to have your license updated.

§3 Documentation

Until such time the R13.0 User's manual is posted at www.lstc.com/download/manuals, the DRAFT User's Manual may be used as a temporary substitute. Note that there may be features mentioned in the DRAFT User's manual which are not implemented for R13.0.0.

With this release, we have a new manual: Volume IV Multiscale for analyses that involve multiple scales. The solvers described in this manual are intended for including manufacturing results in structure analyses.

§4 Notes

The remainder of this file provides a brief description of new features, enhancements, and bug fixes in version R13.0.0. Some of the bug fixes may have also been included in other recent releases.

The items are arranged by category in the sections New Features, Improvements and Enhancements, and Bug Fixes. Understand that in many cases, a particular item may pertain to more than one category. In the interest of brevity, each item is listed only once, under a single category.

§5 New Features

§5.1 Acoustics, NVH, and Frequency Domain

1. Added keyword ***CONTROL_ACOUSTIC_SPECTRAL** to request an acoustic spectral element analysis.
2. Added keyword ***CONTROL_IMPLICIT_SSD_DIRECT** to request a direct complex solution to steady state vibration.
3. Added new boundary condition keywords for acoustics: ***BOUNDARY_ACOUSTIC_COUPLING_SPECTRAL**, ***BOUNDARY_ACOUSTIC_FREE_SURFACE**, ***BOUNDARY_ACOUSTIC_IMPEDANCE_COMPLEX**, ***BOUNDARY_ACOUSTIC_IMPEDANCE_MECHANICAL**, ***BOUNDARY_ACOUSTIC_INTERFACE**, ***BOUNDARY_ACOUSTIC_NON_REFLECTING**, ***BOUND-**

ARY_ACOUSTIC_PRESCRIBED_MOTION, and *BOUNDARY_ACOUSTIC_PRESCRIBED_SPECTRAL

4. Added keyword ***CONTROL_ACOUSTIC_COUPLING** for altering defaults parameters for ***BOUNDARY_ACOUSTIC_COUPLING_MISMATCH** and ***BOUNDARY_ACOUSTIC_COUPLING_SPECTRAL**.
5. Added keyword ***LOAD_ACOUSTIC_SOURCE** for specifying loading sources for acoustic spectral analyses (***CONTROL_ACOUSTIC_SPECTRAL**) and SSD analyses (***CONTROL_IMPLICIT_SSD_DIRECT**).
6. Added keyword ***INTERFACE_ACOUSTIC** to create an interface file of boundary motions that can be used by ***BOUNDARY_ACOUSTIC_INTERFACE** for weakly coupled acoustic fluid-structure interaction in **SSD_DIRECT** analyses.
7. Added ***DATABASE_ACEOUT** and ***DATABASE_HISTORY_ACOUSTIC** to control the output for acoustic analyses.

§5.2 ALE

1. Added support for S-ALE using multi-materials to work with the thermal solver.
2. Added option **COVER_LAG** to ***ALE_STRUCTURED_MESH_MOTION** to expand the S-ALE mesh to cover and follow a Lagrangian structure.
3. Added field **ISYM** for the **FOLLOW_GC** option of ***ALE_STRUCTURED_MESH_MOTION** to constrain the S-ALE mesh expansion / contraction along one or more symmetry planes.
4. 2D S-ALE is now fully supported with major bug fixes and algorithm enhancements. Prior to this release, 2D existed but was experimental and under development.

§5.3 Constraint Cards

1. Added feature invoked with ***CONSTRAINED_SOIL_PILE** that enables modeling of piled foundations in soil/structure interaction models. Applications include seismic engineering and staged construction modeling. See the R13 keyword manual for further details.

2. First experimental version to support solid element parts connected by ***CONSTRAINED_SPR2**.

§5.4 Control Cards

1. Implemented a feature for finding an approximate reference configuration given a measured (deformed) geometry and boundary conditions. It is invoked with ***CONTROL_REFERENCE_CONFIGURATION**.

§5.5 Discrete Element Method

1. Added keywords ***DEFINE_DE_MESH_BEAM** and ***DEFINE_DE_MESH_SURFACE** for coupling beam and shell structure elements to ICFD elements through DES shadow elements on the surface of the structure elements.
2. Added keyword ***DEFINE_DE_BOND_OVERRIDE** to support the modification of bond properties within a certain node set, part set or part.

§5.6 Dual CESE Compressible Fluid Solver

1. With R13, multiphase models are introduced in the DUALCESE framework for the first time. An expanded description of these solvers will be given in §5.6.1.
2. Dual CESE Boundary Conditions:
 - ***DUALCESE_BOUNDARY_CYCLIC**. Cyclic or periodic boundary condition for parallel or rotational sectors
 - ***DUALCESE_BOUNDARY_SOLID_WALL_ROTATE**. Rotating solid wall boundary conditions
 - ***DUALCESE_BOUNDARY_SLIDING**. Sliding boundary condition for special cases involving the moving mesh FSI solver where the nodes move along a surface that maintains its shape, like the surface swept out by the motion of the outside surface of a piston
3. Dual CESE equations-of-state:
 - REFPROP v10.0 is now used (replaces REFPROP v9.1 support), and makes more engineering-type real gas and real liquid

equations-of-state available to the user (***DUALCESE_EOS_REF-PROP**)

- New equations-of-state that are intended to be used with the new multiphase / multifluid dual CESE solvers described later:
 - ***DUALCESE_EOS_COCHRAN_CHAN**. Condensed-phase explosive EOS of type Cochran-Chan
 - ***DUALCESE_EOS_JWL**. Condensed-phase explosive EOS of JWL-type
 - ***DUALCESE_EOS_STIFFENED_GAS**. A stiffened gas type fluid EOS
 - ***DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED**. A generalized Van der Waals type of EOS

4. Dual CESE Reaction Rate:

- ***DUALCESE_REACTION_RATE_IG**. A reaction rate law for the Ignition and Growth model to describe the conversion of reactants to products in the modeling of a condensed-phase explosive
- ***DUALCESE_REACTION_RATE_IG_REDUCED**. A reduced form reaction rate law for the Ignition and Growth model to describe the conversion of reactants to products in the modeling of a condensed-phase explosive
- ***DUALCESE_REACTION_RATE_P_DEPEND**. An explicitly pressure-dependent reaction rate law for describing the conversion of reactants to products in the modeling of a condensed-phase explosive

5. New multiphase solver for multiphase flow problems with two immiscible materials. Keywords associated with this are:

***DUALCESE_EOS_SET**

***DUALCESE_PART_MULTIPHASE**

***DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE**

***DUALCESE_INITIAL_TWO-PHASE**

***DUALCESE_INITIAL_TWO-PHASE_SET**

6. A new hybrid multiphase model solver useful for modeling multiphase / multifluid systems that involve condensed phase explosives. Note that the reaction rate laws and condensed-phase explosive equations-of-state need to be employed for the case of explosion modeling. Keywords associated with this solver are:

***DUALCESE_EOS_SET**

***DUALCESE_PART_MULTIPHASE**

***DUALCESE_BOUNDARY_PRESCRIBED_HYBRID**

***DUALCESE_INITIAL_HYBRID**

***DUALCESE_INITIAL_HYBRID_SET**

7. Support the ***MESH** (ICFD) mesher with boundary layer meshing. ***MESH** keywords are now available for use in creating ***DUALCESE** volume meshes.
8. Immersed boundary method (IBM) FSI with material erosion is now supported for the 3D case, where the eroding elements can be from the following structural element types: solid, thick shell, and thin shell elements.

§5.6.1 Dual CESE Compressible Multiphase CFD Solvers

Our multiphase solver is based on a hybrid formulation, i.e., a mixture of multiphase formulation and augmented Euler formulation. It is designed for the numerical simulation of condensed-phase explosives. It can handle two miscible components, namely the reactant and the products of reaction, and one inert immiscible component, such as the confiner in the case of the rate-stick.

We also implemented a solver for one of the hybrid formulation reduced models, i.e., a two-phase model (a 5-equations model). In this model, only two immiscible materials can be included. Another reduced model, i.e., a fluid-mixture limiting case, will be implemented in the near future.

Notable features of this hybrid model include that it can provide the accurate recovery of the temperature field across all components, and also can sustain large density differences across material interfaces. It can accommodate realistic EOS and arbitrary (pressure or temperature based) reaction-rate laws.

The main applications of these multiphase solvers include the numerical simulation of condensed-phase commercial- and military-grade explosives. Of particular interest is the propagation of detonations in compliantly-confined charges (rate-stick-type problems) and the sensitization of commercial explosives by means of collapsing micro-balloons. For example,

- simulation of combustion and transition to detonation of condensed-phase explosives
- propagation of detonations in compliantly-confined charges (rate-stick-type problem)
- sensitization simulation of commercial explosives by means of collapsing micro-balloons
- shock-induced cavity collapse in liquid explosives

Currently available EOSes: (we will implement more EOSes based on user requirements in the future)

- Mie-Gruneisen type EOSes: Ideal-gase EOS, JWL EOS, and Cochran-Chan EOS
- generalized Van der Waals EOS, including standard Van der Waals EOS and a stiffened gas EOS

Currently available reaction rate laws: (we will implement more models based on user requirements in the future)

- Ignition and Growth (I&G) model
- Simplified Ignition and Growth (I&G) model
- Simple pressure dependent reaction rate law

Note: Multiphase flow and structure interaction (FSI) capabilities will be added in the near future.

Keywords involved with these new multiphase solvers:

- *DUALCESE_BOUNDARY_PRESCRIBED_HYBRID**
- *DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE**
- *DUALCESE_EOS_COCHRAN_CHAN**
- *DUALCESE_EOS_JWL**
- *DUALCESE_EOS_STIFFENED_GAS**
- *DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED**
- *DUALCESE_EOS_IDEAL_GAS**

***DUALCESE_EOS_SET**
***DUALCESE_INITIAL_HYBRID**
***DUALCESE_INITIAL_HYBRID_SET**
***DUALCESE_INITIAL_TWO-PHASE**
***DUALCESE_INITIAL_TWO-PHASE_SET**
***DUALCESE_PART_MULTIPHASE**
***DUALCESE_REACTION_RATE_IG**
***DUALCESE_REACTION_RATE_IG_REDUCED**
***DUALCESE_REACTION_RATE_P_DEPEND**

Solver Characteristics:

- Explicit
- Double precision
- Dynamic memory handling
- SMP and MPP
- 3D solver / special case 2D solver and 2D axisymmetric solver

Solver Main Features:

- dual CESE method (Conservation Element / Solution Element) used for solving the system of multiphase partial differential equations.
- Highly accurate shock wave capturing

§5.7 Elements

1. Added two solid element formulations ***SECTION_SOLID**:
 - Element formulation 62 is an 8 node fully integrated solid brick element that uses an assumed strain field of incompatible modes.
 - Element formulation -18 is an enhanced assumed strain hex/brick formulation for coarse mesh accuracy. It contains 13 incompatible modes to avoid shear and volumetric locking. It is primarily intended for implicit analysis of thin components.

§5.8 EM (Electromagnetic Solver)

1. Erosion of elements can be combined with the batmac model (***EM_RANDLES_BATMAC**) for nail penetration and localized short study
2. Added support for ***EM_RANDLES_EXOTHERMIC_REACTION** to batmac (***EM_RANDLES_BATMAC**)
3. Added contribution of positive current collector to Joule heat rate in batmac (***EM_RANDLES_BATMAC**)
4. Added support for pentahedral elements in ***EM_RANDLES_SOLID**.
5. Added ability to replace the ECM (Randles) model in batmac with a user defined model (***EM_RANDLES_BATMAC**)
6. Implemented NREL's four-equation model in the thermal solver for batter thermal runaway study. This is invoked with ***LOAD_HEAT_EXOTHERMIC_REACTION**.
7. Introduction of mortar contact to treat FEM part in Eddy current solver (***EM_CONTROL_CONTACT**)
8. Support ***EM_CIRCUIT_SOURCE** with inductive heating solver
9. Support the 2D axisymmetric solver with the inductive heating solver
10. Added the monolithic solver (***EM_SOLVER_FEMBEM_MONOLITHIC**) for simulations involving permanent magnets and ferromagnetic materials.
11. Added ***EM_PERMANENT_MAGNET** to define a permanent magnet.
12. Addition of several cell models as well as a user defined cell model. See keywords ***EM_EP_CELLMODEL_FITZHUGH-NAGUMO**, ***EM_EP_CELLMODEL_FENTONKARMA**, ***EM_EP_CELLMODEL_TENTUSSCHER**, and ***EM_EP_CELLMODEL_DEFINEFUNCTION**
13. The EM solver can be coupled with the explicit thermal solver.

§5.9 ICFD (Incompressible Fluid Solver)

1. Added new keyword ***ICFD_BOUNDARY_PRESCRIBED_LEVELSET** to prescribe the fluid height on a boundary.
2. Added keyword ***ICFD_BOUNDARY_NAVIERSLIP** to impose a Navier-slip boundary condition. It works like a regular free slip condition except we are calculating a shear term and adding it to the right-hand side (just like the turbulent law of the wall)
3. Added keyword ***ICFD_BOUNDARY_FSI_FIXED** to allow a solid surface to “slide” over a fluid surface without the fluid surface moving. This boundary condition is mostly useful for conjugate heat transfer applications.

§5.10 Implicit (Mechanical) Solver

1. Added implicit incompressible control volumes for simplified analysis of pressure vessels.
2. For ***CONTROL_IMPLICIT_SOLVER**:
 - LSOLVR = 30 (MUMPS) can now be used for nonsymmetric problems.
 - The fill-reducing method invoked with ORDER = 4 (LS-GPart) is now supported for eigenvalue problems.

§5.11 Isogeometric Analysis (IGA)

1. Added IGA shell element formulations 5 and 6 (***SECTION_IGA_SHELL**) that are analogous to shell element formulations 25 and 26 (***SECTION_SHELL**).
2. Added ***IGA** keyword family for isogeometric analysis. This set of keywords sets up and controls the isogeometric-related capabilities of LS-DYNA. It is fully compatible with standard CAD. Connections are automatic through topology. It supports unstructured splines. It includes extensions for NURBS volumes. It supports boundary conditions, contacts, constraints, and connections (such as welds with HAZ, RBes, bolts) on exact CAD geometry.
3. Extended mechanical coupling of shell patches to T-joints with ***IGA_TIED_EDGE_TO_EDGE**

4. Geometry can be input using elements in Bezier decomposed form with ***IGA_INCLUDE_BEZIER**.
5. Introduced a stabilization scheme for light-control points for trimmed shell NURBS patches in SMP and MPP (see ***IGA_SHELL** and ***ELEMENT_SHELL_NURBS_PATCH**). The basis functions in the trimmed geometry have reduced support which can result in adverse numerical effects in both implicit and explicit.
6. Full MPP support for tied-edge-to-edge contact, for tying multiple NURBS patches along their topological connected interfaces with ***IGA_TIED_EDGE_TO_EDGE**.
7. Allow the use of ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) for IGA through interpolation elements. (SMP/MPP)
8. Enable implicit contact in MPP via interpolation elements which applies to both ***ELEMENT_SHELL/SOLID_NURBS_PATCH** and ***IGA_SHELL/SOLID**.
9. Added support for using rigid walls with IGA.
10. Enable Dirichlet boundary conditions at a parametric edge using Nitsche's method with ***BOUNDARY_PRESCRIBED_MOTION**.
11. New meshing scheme for interpolation shells that allows for non-convex domains.
12. Enable nodal boundary conditions (fully MPP support) with ***IGA_POINT_UVW**.

§5.12 Load Cards

1. Added keyword ***LOAD_PYRO_ACTUATOR** to provide a simple way to simulate hood lift pyrotechnics.

§5.13 Materials and Equations-of-State

1. For the ACTIVE module of ***MAT_295** (***MAT_ANISOTROPIC_HYPERELASTIC**, included Martins et al. 2006 model for skeletal tissues for both explicit and implicit (ACTYPE = 5).

2. Implemented new material model ***MAT_COHESIVE_GASKET** (***MAT_326**) for analyzing gaskets. It is used with cohesive elements. User defined loading/unloading determine the out-of-plane behavior, while the in-plane behavior is linear elastic.
3. Added material model ***MAT_263** / ***MAT_LOU-YOON_-ANISOTROPIC_PLASTICITY**. This material is based on the anisotropic yield function proposed by Lou and Yoon, 2017. It includes an anisotropic form of the Drucker function. This material is more accurate in representing material anisotropy than the other general anisotropic models. It can be accurate when predicting earing during cup-drawing simulations. It is faster than ***MAT_133**. This material is supported for shell and solid elements.
4. New thermal material ***MAT_T17** / ***MAT_THERMAL_CHEMICAL_REACTION_ORTHOTROPIC**:
 - Material incorporates the chemical reaction kinetics from ***MAT_T06** and relies on the same input data.
 - Orthotropic thermal behavior. Three load curves for the thermal conductivity values in the different directions are expected in the input for the filler and for each species.
5. Added material ***MAT_249_CRASH**. It is an anisotropic material model for crashworthiness analyses of reinforced composite materials. The overall stress is found by adding the fiber and matrix stresses. The matrix is modeled with a relatively simple orthotropic elastic plastic formulation. The reinforcement is modeled with a hyperelastic constitutive model with up to three distinct fiber directions. It uses a phenomenological description of damage and failure based on tabular input data. Driving forces for damage and failure are fiber elongation and reorientation.
6. Added materials ***MAT_ACOUSTIC_COMPLEX** (***MAT_090_COMPLEX**), ***MAT_ACOUSTIC_DAMP** (***MAT_090_DAMP**), and ***MAT_ACOUSTIC_POROUS_DB** (***MAT_090_POROUS**) for modeling acoustic fluids. See the materials user's manual for details.

§5.14 MPP

1. Added keyword ***CONTROL_MPP_REBALANCE** to control features associated with dynamic load balancing.

§5.15 Multiscale

1. New keywords ***INCLUDE_MULTISCALE** and ***DEFINE_MULTISCALE** to generalize ***INCLUDE_MULTISCALE_SPOTWELD** and ***DEFINE_MULTISCALE_SPOTWELD** for multiscale modeling applications. For instance, these keywords may be used to couple meso-scale solder joint models with a macro-scale printed-circuit-board for IC reliability analysis.
2. Added Representative Volume Element (RVE)-based micromechanical analysis for composite materials invoked with ***RVE_ANALYSIS_FEM**. It is intended for predicting nonlinear macroscopic constitutive behaviors based on 2D or 3D material microstructure. It works in the double precision versions for both SMP and MPP.
3. Added ***DATABASE_RVE** to output RVE homogenization results to the rveout file
4. Added new keyword ***INCLUDE_COSIM** for two-scale co-simulation. The two-scale co-simulation method allows for running two separate LS-DYNA MPP jobs simultaneously: one for large-scale and the other for meso-scale. Two jobs run at different time step sizes and synchronize automatically at every large scale time step.

§5.16 Output

1. Output of ***ALE_STRUCTURED_FSI** now can be controlled with the following DATABASE keywords: ***DATABASE_FSI**, ***DATABASE_FSI_SENSOR**, ***DATABASE_BINARY_FSIFOR**, ***DATABASE_SLEOUT**
2. Implemented ***DATABASE_BINARY_D3MAX** to output maximum stresses and strains during a transient analysis. These results may be output to d3part if OUTPUT = 2. A Butterworth low-pass filter may also be applied to the results (IFILT = 1).

§5.17 Peridynamics

1. Added peridynamics laminate model to capture in-plane and delamination failure modes of Unidirectional Fiber Reinforced Polymer (UD-FRP) laminates in SMP and MPP. With this model you specify

each lamina as a separate part and the laminate as a part set. ***SET_PERI_LAMINATE** assembles the laminate and specifies the fiber angles for each lamina. Peridynamic parts require `ELFORM = 48` on ***SECTION_SOLID_PERI**. The laminate must be meshed as 4-node surface elements. It is discretized by layers of these elements along the thickness direction. To represent material failure, adjacent elements cannot share nodes, meaning the total number of nodes is 4 times the total number of peridynamic laminate elements. ***MAT_ELASTIC_SOLID_PERI** specifies the transverse isotropic material properties for one lamina and failure parameters for the laminate.

§5.18 SPG (Smooth Particle Galerkin)

1. Added MC-SPG invoked with `ITB = 3` on ***SECTION_SOLID_SPG**. MC-SPG uses a momentum consistent smoothing algorithm to stabilize SPG. Input field `NMC` for this keyword sets the damping coefficient for MC-SPG. Four spline functions specified with `ISPLINE` are available for MC-SPG: box-type cubic spline function (`ISPLINE = 4`), box-type WF3 function (`ISPLINE = 5`), elliptic-type spline function (`ISPLINE = 14`), and elliptic-type WF3 function (`ISPLINE = 15`).

We also added two damage models specifically for MC-SPG. Setting `IDAM = 11` specifies a pre-damage model for controlling the crack growth speed. `ALPHA_B` is used to control the crack growth speed. Alternatively, bond-based damage is available by setting `IDAM = 13`. In this case, shear strain and stretch are used to calculate the damage accumulation. The damage is then applied to the flow stress. This type of damage is useful for simulating shear bands in cutting simulations.

2. Two-stage SPG analysis, particularly for strength analysis of mechanical joints after joining process analysis, such as flow drill screw (FDS), self-piercing riveting (SPR), wood/concrete screwing, etc. This analysis requires two new keywords ***INTERFACE_SPG_1** and ***INTERFACE_SPG_2**.
3. Support small restart for both SMP and MPP SPG calculations.
4. Developed a fully implicit ISPG formulation invoked with ***SECTION_FPD**. ISPG is a new generation incompressible Navier-Stokes CFD solver for the accurate and efficient simulation of solder reflowing in the electronics packaging process. This method can simulate

complex solder reflow scenarios including solder pad, mask, copper trace, etc. It accurately and efficiently handles surface tension and wall adhesion. It can be coupled to the thermal and structure (implicit and explicit) solvers for both SMP and MPP. Applications of this coupling include large scale thermal-mechanical PCB and solder reflow analysis.

§5.19 SPH

1. Added optional calculation of drag force in incompressible SPH with ***DEFINE_SPH_AMBIENT_DRAG**.
2. Added keyword ***DEFINE_SPH_MASSFLOW_PLANE** to track mass of SPH particles going through a set of elements.
3. Added keyword ***DEFINE_SPH_VICINITY_SENSOR** to track mass of SPH particles coming in the vicinity of set of elements. Added capability to selectively trace them in LS-PrePost.
4. Added SPH Interface file for visualization of wetness results. This file is activated with `isph = "filename"` on the command line. The output interval is controlled with ***DATABASE_BINARY_ISPHFOR**.

§5.20 Thermal Solver

1. Added keyword ***BOUNDARY_TEMPERATURE_PERIODIC_SET** to specify rotational periodic, reflective periodic, and sliding periodic boundary conditions for the thermal solver.

§5.21 XFEM (eXtended Finite Element Method)

1. Implemented crack propagation in junction structures for both SMP and MPP solvers (***SECTION_SHELL_XFEM** with `ELFORM = 54`).
2. ***MAT_240** can now be used to model the XFEM cohesive zone for ductile fracture.
3. Added stress triaxiality based failure strain to ductile XFEM for shells (see `FAILCR = -4` on ***SECTION_SHELL_XFEM**).
4. Added directional center of effective plastic strain for crack propagation option (`PROPCR = 3` in ***SECTION_SHELL_XFEM**).

§6 Improvements and Enhancements

§6.1 Acoustics, NVH, and Frequency Domain

1. The following are improvements for ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**:
 - For METHOD = 2 (Variational Indirect BEM), matrix assembly is now multithreaded, leading to considerable performance improvements in SMP.
 - Enabled Kirchhoff method and Rayleigh method for panel contribution analysis.
 - The **PANEL_CONTRIBUTION** keyword option can be used with the MATV keyword option. Mismatching BEM/FEM mesh is supported.
2. Enabled ***BOUNDARY_ACOUSTIC_MAPPING** to map the user provided velocity history file (RESTRT = 3 on ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**) to the BEM mesh. The mesh in the history file may be different from the BEM mesh. This also works in MPP.
3. Implemented d3acc output (***DATABASE_FREQUENCY_BINARY_D3ACC**) for MPP.
4. Added input fields HALF1 and HALF2 to ***FREQUENCY_DOMAIN_ACOUSTIC_FRINGE_PLOT_SPHERE** to create complete, half or quarter sphere mesh.
5. Updated ***BOUNDARY_ACOUSTIC_MAPPING** so that when a BEM node overlaps with a structural node, no interpolation is needed and the nodal velocity is just copied for the BEM acoustic computation.
6. For ***FREQUENCY_DOMAIN_ACOUSTIC_FEM**,
 - Enabled ***FREQUENCY_DOMAIN_PATH** for modal acoustics.
 - Enabled mode selection by ***FREQUENCY_DOMAIN_MODE** for modal acoustics.
7. For ***FREQUENCY_DOMAIN_RESPONSE_SPECTRUM**,
 - Implemented an option to run response spectrum analysis with user defined mode combination (MCOMB = 99).

- Implemented ABS method (absolute value method) to response spectrum analysis (MCOMB = 8).
 - Implemented an option to use user defined coefficients in 100-40-40 rule for multidirectional load spectrum cases (MPRS = 2).
 - Implemented 100-40-40 rule for multidirectional response combinationi (MPRS = 1).
 - Implemented adding static response results to d3spcm (BINARY = 3 in ***DATABASE_FREQUENCY_BINARY_D3SPCM**).
8. For ***FREQUENCY_DOMAIN_SSD**,
- Implemented keyword option SUBCASE to include multiple load cases in one run. Also added keyword option SUBCASE to the various databases (D3SSD, NODOUT_SSD and ELOUT_SSD) so the output frequencies for each loading case can be different.
 - Implemented the output of nodout_ssd for direct SSD (***FREQUENCY_DOMAIN_SSD_DIRECT**)

§6.2 Adaptivity

1. Added support for the file INTFOR during in-core adaptivity
2. For ***USER_INTERFACE_FRICTION** support *h*-adaptivity (MPP), meaning if ***CONTROL_ADAPTIVE** with ADPTYP = 1 or 2 is used, then history variables uhnew of subroutine usrfrc are transferred correctly from the old to new mesh in each adaptive refinement step.
3. The following are improvements for 3D *r*-adaptivity (ADPTYP = 7 in ***CONTROL_ADAPTIVE**)
 - For local refinement, LS-DYNA smooths out the mesh size transition from the refined area to the coarse area.
 - 3D TET adaptivity (ADPOPT = 2 in ***PART**) now only accepts TET initial meshes
 - When the auto remesher fails, you can now perform manual remeshing.
4. 3D tetrahedral adaptivity now supports defining moving adaptive boxes (***DEFINE_BOX_ADAPTIVE**).
5. Load curve, LCADP, used by 3D *r*-adaptivity is now output to d3hsp if it is nonzero.

6. Some improvements to the curvature calculation for $ADPTYP = 2$ on ***CONTROL_ADAPTIVE**. The mesh refinement happens closer to the curved region.

Improvement to adaptivity algorithm associated with ***CONTROL_ADAPTIVE_CURVE**. More elements than necessary were being refined.

7. Six-node solid elements are allowed to be refined during adaptivity for sandwich parts with multiple layers.

§6.3 Airbag Cards

1. ***AIRBAG_PARTICLE**

- Avoid asynchronous particle to fabric contact collision sorting for parallel processing which may select collision pairs in a different order and produce various results.
 - Solid parts can be included in the CPM airbag part list
 - Added to the CPM interface forces file the ratio of the impact from the initial air inside the bag to the impact from the inflator particles, p_{air} . This output allows you to visualize the initial air effect for $IAIR = 2$ and 4.
 - Added keyword option JET to ***AIRBAG_PARTICLE**. This option calculates the thrust force for an external vent and gives a better reaction force on the structure
 - Store maximum temperature for nonlinear C_p curve to keep the value monotonic for the energy calculation.
2. Extended ***DEFINE_CPM_VENT** so that the part pressure of specified parts (PID1 and PID2) can be used to evaluate the flow rate through internal vents without requiring a chamber definition. It has also been enhanced to be used for external vents for both the with chamber and without chamber cases.

§6.4 ALE

1. Added support for ***ALE_STRUCTURED_MESH_TRIM** in 2D. It previously only worked in 3D.
2. For keywords ***ALE_STRUCTURED_MESH_TRIM** and ***ALE_STRUCTURED_MESH_VOLUME_FILLING**, created new algorithm

for determining if a point lies inside or outside of a structure. The new logic works well for handling structures with narrow spaces between them or with narrow openings (like car doors).

3. Added field ICASE to ***ALE_STRUCTURED_MESH_CONTROL_POINTS** to optionally select from two special methods of automatically setting up the mesh spacing.
4. Reimplemented ***CONTROL_EXPLOSIVE_SHADOW_SET** for S-ALE to achieve better accuracy and much shorter run time.
5. Added ***ALE_STRUCTURED_MULTI-MATERIAL_GROUP** for a more user-friendly way to set up ALE fluids in a S-ALE model. 3D and 2D are both supported.
6. Enhanced vacuum treatment in S-ALE for better stability. This will cause a small discrepancy in the results, but this version is more robust and has a more stable time step size.
7. Added support for S-ALE with MPP pre-decomposition.
8. For ***ALE_STRUCTURED_VOLUME_FILLING**, if specifying how to fill the mesh with GEOM = PART or PARTSET, the part/part set of the Lagrangian structure can now be meshed with thick shell or solid elements (shell in 2D). Previously only shell element parts were supported. All the volume occupied by these parts will be filled with the assigned fluid.
9. ***EOS_MURNAGHAN** supported for ***ALE_AMBIENT_HYDROSTATIC** and ***INITIAL_HYDROSTATIC_ALE**.
10. ***ALE_STRUCTURED_FSI** works in 2D. It previously only worked in 3D.
11. Algorithm enhancements for S-ALE Fluid-Structure Interaction (FSI) with ***ALE_STRUCTURED_FSI** to ensure a more stable FSI with less leakage, especially for explosion simulations. These improvements include
 - better leakage detection and removal;
 - constraining ALE nodes immersed inside LAG solid/thick shells;
 - enhancements in determining the spring stretch more accurately based on different cases (explosion/sloshing/airbags);

- better estimation of the fractional penalty stiffness; and
 - better choice of penalty spring direction.
12. For ***ALE_STRUCTURED_FSI**, added variable friction coefficient as a function of contact pressure and relative velocity (see field FRIC).
 13. ***DATABASE_FSI** supports structure segments from thick shell parts.
 14. For ***BOUNDARY_AMBIENT** AMBTYP = 5 (***LOAD_BLAST_ENHANCED** receptor), we've made the following enhancements:
 - Support multiple detonation sources
 - Obtain element density from ***LOAD_BLAST_ENHANCED** for better accuracy
 - Obtain nodal velocity from the area-weighted average of the segment velocity squared instead of segment velocity. This better conserves kinetic energy.
 15. Improved the stability of ***EOS_TABULATED_COMPACTION** when used with ALE/S-ALE.
 16. Enhanced ***BOUNDARY_NON_REFLECTING** when used with ALE/S-ALE to prevent artificial influx from outside.
 17. Extended the feature invoked with CNTTYP = 1 of ***INITIAL_VOLUME_FRACTION_GEOMETRY** to fill ALE elements with the material of a solid part in 3D (or shell part in 2D). The boundaries of the part specify the geometry of the container to be filled. The material in the part can optionally fill this container. This feature is useful for setting up an ALE model with a complex dummy structure that outlines a volume of ALE elements to fill.
 18. Added field TBEG to ***INITIAL_ALE_MAPPING** to shift the initial time of the current run from the map file. The current run usually starts at $t = 0$.
 19. LS-DYNA creates the mapping file read by ***INITIAL_ALE_MAPPING** during the last cycle. If a sensor switch defined by ***TERMINATOR_SENSOR** reaches a critical condition, the run is about to be terminated. Under these conditions, LS-DYNA will output the mapping file as well.

20. ***ALE_MESH_INTERFACE** now generates a mapping file that can be used in a 2nd run to initialize a Lagrangian (not ALE) tetrahedral model. ***INITIAL_LAG_MAPPING** reads this file and maps data from the last cycle of the ALE run to the Lagrangian tetrahedra.
21. Implemented a multiple ideal gases version of the flux corrected transport invoked with METH = 6 in ***CONTROL_ALE** for Structured ALE.
22. Reimplemented CTYPE = 3 in ***CONSTRAINED_LAGRANGE_IN_SOLID** to constrain ALE materials and structures to move together in the structure normal direction. This version, unlike the previous version, is supported in MPP.
23. ***DEFINE_CURVE_FUNCTION** can be used instead of ***DEFINE_CURVE** to compute the mesh velocities in ***ALE_REFERENCE_SYSTEM_CURVE**.
24. For ELFORM = -8 on ***SECTION_ALE1D**, single point constraints are automatically applied to the nodes on the beam mesh ends if EBC > 0 on ***CONTROL_ALE**. These constraints are the same as applying ***BOUNDARY_SPC_NODE** to the ends.

§6.5 ***BATTERY (Electrochemistry Solver)**

For the current updates, the 1D electrochemistry (EC) and its coupled thermal and mechanical (ECTM) solvers can be summarized as follows:

1. 1D EC can run up to almost 0
2. The solver can run without any limitation of storage (memory).
3. The convergence ability of the solver is improved by applying a higher-order method in the time marching scheme.
4. The material properties of the anode, cathode, and electrolyte are updated for practical applications.
5. The MPP capability for the ECTM solver is improved, and some instability issues are resolved.
6. Users can directly print out an input file for the source term of the LS-DYNA thermal solver by providing an output file name to the ***BATTERY_ECHEM_THERMAL** keyword card. The data then written by

the 1D EC solver to this ascii file has the format of time in the first column, and heat flux in the second column. The thermal solver can then simulate the battery thermal effects to test if the thermal solver is working correctly when the heat flux from the battery is input to it. Please refer to the thermal solver manual pages for a detailed description of the source term input.

The keywords influenced by the current updates and improvements are:

- ***BATTERY_ECHEM_CONTROL_SOLVER**
- ***BATTERY_ECHEM_MAT_ANODE**
- ***BATTERY_ECHEM_MAT_CATHODE**
- ***BATTERY_ECHEM_MAT_ELECTROLYTE**
- ***BATTERY_ECHEM_MAT_THERMAL**

§6.6 Blast

1. Output mole per particle for HE and resident air to `abstat_pbm` so the user can adjust the particle ratio

§6.7 Boundary Cards

1. Enabled ***BOUNDARY_CYCLIC** to set the electric voltage of node set NSID1 equal to the voltage of node set NSID2 for piezoelectric materials.
2. Enabled segments of thick shells to be subject to a non-reflecting boundary condition with ***BOUNDARY_NON_REFLECTING**.
3. Added support for thermal thick shells with ***BOUNDARY_TEMPERATURE_TRAJECTORY**.
4. Simplified the boundary condition redundancy check to improve the performance of ***BOUNDARY_TEMPERATURE_TRAJECTORY**.
5. Implemented the parameters AD and AS from ***BOUNDARY_NON_REFLECTING** in ***BOUNDARY_NON_REFLECTING_2D** to activate dilatational and shear waves, respectively.
6. Improvement for ***BOUNDARY_SPC_SYMMETRY_PLANE_SET**: The default for TOL is 0.2 for shell elements, so for these elements the tolerance does not need to be specified.

7. Extended ***BOUNDARY_SPC_SYMMETRY_PLANE** to support thick shells.

§6.8 CESE Compressible Fluid Solver

1. Removed the CFL < 1.0 limit. Note that there is no such limit in the dual CESE solvers.
2. Turned off any ***CESE_DATABASE_...** output when running CESE cavitation problems, and warn the user about this limitation.
3. Allow cyclic boundary condition segment sets to contain more than 9999 segments.
4. For the case where a problem attempts to use immersed boundary FSI with the non-inertial rotating reference frame solver, a fatal error is emitted informing the user that this is not supported.

§6.9 *CHEMISTRY

1. Add code to properly deal with missing transport data in chemistry models when using ***CHEMISTRY** with ***CESE**.

§6.10 Constraint Cards

1. Added the ID keyword option to ***CONSTRAINED_LOCAL**.
2. Support constraining the electric voltage of piezoelectric materials (DOF = 7) with ***CONSTRAINED_LINEAR_GLOBAL**.
3. Allow ***CONSTRAINED_NODE_SET** to constrain the electric potential of the piezoelectric material.
4. Rigid body stoppers invoked with ***CONSTRAINED_RIGID_BODY_STOPPERS** is supported in implicit by way of Lagrangian Multipliers and inequality constraints.
5. Added "state variable" of ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) to binout/swforc output.
6. Improved two rarely occurring issues with ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3):
 - More accurate search of associated shell element nodes, helpful in cases where shell elements are bent around corners.

- Better solution for rotation vector if both normals are colinear.
7. ***CONSTRAINED_TIED_NODES_FAILURE** now supports damage values from ***MAT_107** and ***MAT_110** to be used as the failure variable for EPPF.
 8. PENALTY option of ***CONSTRAINED_BEAM_IN_SOLID** now supports 3D *r*-adaptivity (ADPTYP = 7 on ***CONTROL_ADAPTIVE**).
 9. Added deactivating ***CONSTRAINED_BEAM/SHELL_IN_SOLID** during dynamic relaxation.
 10. Specify whether ***CONSTRAINED_BEAM/SHELL_IN_SOLID** are the PENALTY versions in the d3hsp file.

§6.11 Contact

1. Add option TETFAC to optional Card F on ***CONTACT_OPTION** to scale the computed volume of tet elements for the mass calculation in SOFT = 2 contact. Better agreement is achieved with TETFAC between 3 and 5 when contact forces are compared between hex and tet meshes. This input field is supported for 4-node and 10-node tets.
2. Enabled the SWRADF factor on ***CONTROL_CONTACT** to work with solid elements and solid element assembly spot welds.
3. Added a tapered segment option (SWTPR on optional Card F of ***CONTACT_OPTION**) for segment-to-segment (SOFT = 2) contact. This option uses tapered segments around spot welds when SPOTHIN on ***CONTROL_CONTACT** is used to thin segment at spot welds. The segment thickness tapers from full thickness to the reduced thickness at nodes shared with thinned segments.
4. Enabled the offset option with CODE = 4 of ***DEFORMABLE_TO_-RIGID_AUTOMATIC** to work with segment-to-segment (SOFT = 2) contact.
5. Enabled PSTIFF on ***CONTACT** to work with a contact birth time. Also, modified how friction parameters are used in segment-to-segment (SOFT = 2) contact when ***PART_CONTACT** is used. If a segment pair in contact has nonzero friction parameters from only one of the segments, the nonzero friction parameters will be used for the pair. If friction parameters are defined for both parts, then the

average is used. Previously, the average was used without checking values, so zero values could be used in the average causing the coulomb coefficient to be half of the expected value.

6. Added support to segment-to-segment (SOFT = 2) contact friction for the viscous friction parameters VC on ***DEFINE_FRICTION**.
7. Improved ***DEFINE_FRICTION_SCALING** in segment-to-segment (SOFT = 2) contact so that it averages the friction coefficients only if nonzero scale factors are defined for both segments of a pair. Previously, a scale factor of 1.0 was assumed for undefined scale factors, and an average was done including that 1.0. Now only the defined scale factor is used.
8. Enabled the ***DEFINE_FRICTION_SCALING** option to be used with orthotropic friction options when using segment-to-segment (SOFT = 2) contact.
9. Enabled FRCFRQ on Optional Card A of ***CONTACT_OPTION** to work with MPP contact when SOFT = 0 or 1.
10. Enabled the use of ***LOAD_SEGMENT_CONTACT_MASK** with segment-to-segment (SOFT = 2) contact. This works with any supported contact keyword in all versions, SMP, MPP and HYBRID.
11. Enabled ***ELEMENT_SHELL_SOURCE_SINK** when used with segment-to-segment (SOFT = 2) contact.
12. Added an option for orthotropic friction when using segment-to-segment (SOFT = 2) contact. If friction properties on either side are set to zero or left blank, then properties of the other side will be used, and the material direction will be calculated from those segments.
13. Added input parameter ENDTOL to ***CONTACT_GUIDED_CABLE** to dictate how far beyond the end of a cable element a sliding node must pass before it starts contacting the next cable element. The default, which is 0.002 times the cable element length, normally works fine. But if one or more of the cable elements is especially long, the transition can cause noise. In that case specifying the tolerance manually using ENDTOL may be desirable.
14. The following are now supported for 3D Mortar contact (***CONTACT_..._MORTAR**):

- Orthotropic friction, governed by four static friction coefficients that models sliding interaction between fiber/fiber, fiber/cross-fiber, crossfiber/fiber and crossfiber/crossfiber directions. Manual provides full details.
 - A linear relationship between contact pressure and penetration depth, as opposed to the parabolic relation which is default. The new option is activated by specifying $IMGAP < 0$, and can presumably be used to reduce penetrations for small to moderate contact pressures.
 - Redecomposition and full deck restarts.
 - User defined tiebreak and tied weld for modeling lamination/delamination.
 - LCFST for a friction coefficient that depends on the temperature.
15. The following are now supported for 2D Mortar contact (***CONTACT_2D_..._MORTAR**):
- MPP
 - User defined friction
 - Frictional energy calculations
 - Interference contact using TDPEN, which is similar to MPAR1 and IGNORE = 3 for 3D Mortar contact
16. Added ***CONTACT_DRAWBEAD_BENDING**. This keyword option invokes a new contact algorithm for drawbeads that includes a weakening effect on the blank as it travels through the beads. It also stretches the blank as it goes through the beads.

§6.12 Control Cards

1. For ***CONTROL_PZELECTRIC**:
 - Added the field NDTRFK to form a new dielectric matrix for every NDTRFK steps.
 - Allow optional output of averaged electric flux and electric field with input field EPZMSG.
2. Keyword: ***CONTROL_TIMESTEP** Allow time step factor TSSFAC on ***CONTROL_TIMESTEP** to be a time-dependent curve.

3. For ***CONTROL_STAGED_CONSTRUCTION** and ***DEFINE_STAGED_CONSTRUCTION_PART**, added staged construction for thick shell ELFORM = 3 and 5. Previously, the staged construction keywords had no effect on these elements. Also, the manual did not state that ***DEFINE_STAGED_CONSTRUCTION** was applicable to thick shells; this has been corrected.
4. Pore fluid capability ***CONTROL_PORE_FLUID** added for thick shell elements.
5. Add new option LISPSID on ***CONTROL_SHELL** to define a part set for which the stresses from ***INITIAL_STRESS_SHELL** are with respect to the local coordinate system.
6. Add new keyword ***CONTROL_CONSTRAINED** for setting constraint related properties. Currently it only sets the behavior of the initialization check for SPR2/SPR3.
7. Added UMCHK on ***CONTROL_MAT** to potentially issue an error if only unmodified user material subroutines are called.
8. Added 5 more materials supported for the ISTUPD = 4 option of ***CONTROL_SHELL: *MAT_120, *MAT_187, *MAT_224, *MAT_225,** and ***MAT_251**.
9. Added new detailed materials energy option MATEN on ***CONTROL_ENERGY**. Currently this feature is supported material models 3, 4, 15, 19, 24, 63, 81, 82, 98, 104, 105, 106, 107, 123, 124, 188, 224, 225, 240, and 251 for shell and solid elements. Setting MATEN = 2 causes the detailed material energies to be computed. For this calculation, the internal energy is split into the elastic, plastic, and damage portions. They are computed and reported as "mat_energy_elastic", "mat_energy_plastic", and "mat_energy_damage" in glstat and matsum.
10. Support ICOH = 1 of ***CONTROL_SOLID** for ***MAT_169**. This is automatic deletion if neighboring elements fail. It is already available for cohesive elements but is now also available for this material.

§6.13 Discrete Element Method

1. Allow user control over the number of neighboring DES to search for across processors by specifying MAXNEI on ***CONTROL_DISCRETE_ELEMENT** (default = 20)

2. Added new drag coefficient for external ballistic drag force calculation for shotgun pallet analysis (CD = -2 on ***DEFINE_DE_DRAG_FLOW**).
3. Added the keyword option TRANSDUCER to ***DEFINE_DE_TO_SURFACE_COUPLING** to collect coupling forces from different active coupling interfaces.
4. Support moving DES active region (see ***DEFINE_DE_ACTIVE_REGION**).
5. Automatically generate DES element between start and end NODEs specified with NID and NID2 on ***ELEMENT_DISCRETE_SPHERE**
6. Added injecting particles with multiple predefined radii by specifying IMULTI on ***DEFINE_DE_INJECTION**. The composition of the particles (radius and mass percentage) can be specified by setting IMULTI > 1.
7. Added optional input fields LCVX, LCVY, and LCVZ to allow the particle injection velocity to vary through time instead of being constant.
8. Added flag ISKIP to ***CONTROL_DISCRETE_ELEMENT** to skip the particle to particle interaction computation.
9. Added input field MAXGAP to ***DEFINE_DE_TO_SURFACE_TIED** to determine the maximum gap between the DES and master surface.
10. For ***DEFINE_DE_INJECT_BONDED**, optimized the DES bond injection model to increase the injection mass capacity with the same injection plane.
11. Added passing more wear history variables, including contact surface area and accumulated surface wear, to the user defined wear interface.

§6.14 Dual CESE Compressible Fluid Solver

1. Added an afterburn energy field to the ideal gas EOS that is used only in the new hybrid model multiphase solver ***DUALCESE_EOS_IDEAL_GAS**).

2. For dual CESE moving mesh FSI problems, added a second argument to the ***DUALCESE_BOUNDARY_FSI** card that is the reference pressure. This permits one-sided FSI calculations.
3. Also for dual CESE moving mesh FSI problems, using a displacement threshold, allow the expensive mesh morphing calculation to be avoided when the FSI displacements are all below the specified threshold (***DUALCESE_CONTROL_MESH_MOV**).
4. Added support to several initialization and prescribed boundary cards to allow them to use the ***DEFINE_FUNCTION** capability. For the initialization cards, the function used can only be a function of x , y , and z (position). For the boundary conditions, these functions can involve the following variables defined at the point of evaluation: x , y , z (position), v_x , v_y , v_z (fluid velocity), density, temp (temperature), pres (pressure), time, and dt (dual CESE time step). This functionality applies to:

***DUALCESE_INITIAL**

***DUALCESE_INITIAL_SET**

***DUALCESE_INITIAL_HYBRID**

***DUALCESE_INITIAL_HYBRID_SET**

***DUALCESE_INITIAL_TWO-PHASE**

***DUALCESE_INITIAL_TWO-PHASE_SET**

***DUALCESE_BOUNDARY_PRESCRIBED**

***DUALCESE_BOUNDARY_PRESCRIBED_HYBRID**

***DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE**

***DUALCESE_BOUNDARY_PRESCRIBED_VN**

5. For code speed purposes, improved the treatment of wedge/prism elements.
6. Many corrections to d3p1ot output have been made, along with corresponding enhancements to LSPP4.9. This applies mostly to the moving mesh FSI and multiphase solvers. Also, the dual CESE keyword input decks can now be read into LSPP4.9 and modified after that in LSPP4.9.
7. Instead of leading to a code failure, unused mesh nodes in the dual CESE mesh are now ignored. Also, the user is now informed about

elements defined in the dual CESE mesh that are invalid, and the simulation run is terminated if any are found.

8. Corrected the overallocation of some state arrays and otherwise reduced the memory footprint of the dual CESE solvers. This includes re-reading (from an internal file) a large array each time it is needed for `d3plot` output with the moving mesh solvers.

§6.15 Elements

1. Updated ***ELEMENT_BEAM_SOURCE** to allow resultant beam elements (***SECTION_BEAM**, `ELFORM=2`) with bending resistance to be pulled out of a beam source. This feature is supported for ***MAT_ELASTIC** and ***MAT_RESULTANT_PLASTICITY**.
2. Improve MPP decomposition (and therefore efficiency) for ***ELEMENT_BEAM_SOURCE**. All beam elements inside the source still go on one processor, but the others that are initially outside are equally distributed now.
3. Added optional pull-in force limiter (`LMTFRC`) to ***ELEMENT_SEATBELT_PRETENSIONER** for type 4, 6 and 7 pretensioners.
4. For ***ELEMENT_SEATBELT_SLIPRING** added the output of slipring friction coefficient, normal force, swap angle, skew angle and attached belt force to `SBTOUT`.
5. For `ELFORM = -29` and `29` on ***SECTION_SHELL**, increased mass scaling factor for stability.
6. Enabled the `ORIENTATION` option on ***ELEMENT_BEAM** to be used with form 9 spotweld beam. It was not available before.
7. Enabled thick shell form 3 to work with material models that use an equation of state.
8. For ***ELEMENT_DIRECT_MATRIX_INPUT**:
 - Enhanced the reading of superelement input files to support connections using nodes, scalar nodes, and internal dofs. Also accept dof labels of "0" for scalar nodes.
 - Modified MPP decomposition logic for collecting nodes when reading binary formatted files for superelements.

9. Add the possibility to define 3-Point Simpson integration rule through the thickness per layer of a composite shell defined via ***PART_COMPOSITE**. This can be defined via an irregular optional Card 2 (OPTCARD).
10. Keyword: COHOFF on ***SECTION_SOLID** can be used to place the relative location of cohesive interface between shells. It is only relevant for ELFORM = 20/22.
11. Added element quality check for solid cohesive elements (***SECTION_SOLID**, ELFORM=19/20) to detect inverted elements in the first cycle.
12. Added keyword option MISC for ***SECTION_SHELL** with first new parameter THKSCL. This allows the scaling of shell element thickness for all elements in the corresponding section at once.

§6.16 EFG (Element Free Galerkin)

1. Pressure smoothing (IPS = 1 on ***SECTION_SOLID_EFG**) now supports usermat when using EFG solid formulation 42.

§6.17 EM (Electromagnetic Solver)

1. Added more strain related options for ***DEFINE_FUNCTION** in ***EM_RANDLES_SHORT**.
2. Added option to ***EM_MAT_002** to allow EOS to be a function of temperature for the permeability
3. Added input field NPERIO to ***EM_CONTROL** so you can choose the period for calculating the average Joule heat rate for the induced heating solver.

§6.18 Fatigue Cards

1. For ***FATIGUE_{OPTION}**, extended strain based fatigue (STRSN = 1) to MPP (solid elements).
2. Implemented ***FATIGUE_D3PLOT** for MPP.

§6.19 Forming Analysis

1. Dynain file is output when using ***INTERFACE_SPRINGBACK_SEAMLESS** upon termination.
2. Added field DEPTH to ***DEFINE_CURVE_TRIM_2D**. If the distance between the element and the curve is greater than DEPTH - 1, than the element is not trimmed. In this way, we can prevent the trimming through to the opposite side of the part.
3. Changes to lancing (***ELEMENT_LANCING**):
 - Lancing can happen at different times.
 - Lancing can occur in an arbitrary direction. Previously lancing could only occur in the z -direction.
 - Lancing can have depth.
4. Added input field I2DYNAIN to ***CONTROL_FORMING_MAXID**. When I2DYNAIN = 1, ***CONTROL_FORMING_MAXID** is output to a dyanin file with the new maximum node and elements IDs. This feature simplifies post-processing for multi-step forming processes by ensuring that the new element and node IDs generated in subsequent steps are bigger than those in previous steps.
5. Added keyword ***CONTROL_FORMING_HOME_GAP** to calculate the minimum gap between the upper and lower tools. If the gap is smaller than the blank thickness, then problems can occur during forming simulations.
6. Output keyword ***INITIAL_STRAIN_SHELL/SOLID** after trimming. This will not affect the simulations after trimming.
7. Improved algorithm for searching in ***INCLUDE_STAMPED_PART** which significantly sped it up.
8. Extended ***DEFINE_BOX_NODES_ADAPTIVE** so that tube adaptivity can be applied to part sets, not just parts (set PID < 0). This extensions is usefule for simulating the forming of tailor welded blanks.
9. ***CONTROL_FORMING_REMOVE_ADAPTIVE_CONSTRAINTS** is used to remove the adaptive constraint by refining the master element with triangle elements. However, the newly generated elements do not have forming history information. Per customer request, we have added the history information to the newly generated elements.

10. ***MAT_036** is supported for the one-step method (***CONTROL_FORMING_ONESTEP**). Note that the stress-strain is only allowed to be specified with a curve (HR = 3).
11. Modified trimming of sandwich parts. We assumed sandwich parts were made up of three layers: a bottom shell layer, a solid layer, and a top shell layer. Some customers cover the solid elements not only from the top and bottom but also the sides. With this improvement, the shell elements on the side will also be trimmed.
12. Added NRBST = 2 to ***CONTROL_FORMING_PROJECTION** so that the blank nodes move in the tool's normal direction when there is initial penetration. This feature was developed for contact between a guide pin and blank.
13. Added keyword option POSITIVE to ***CONTROL_FORMING_AUTOPosition_PARAMETER** which ensures that the calculated distance to reposition will always be a positive value.
14. Improvement for ***CONTROL_ADAPTIVE_CURVE**: This keyword is used to refine the elements along the trimming curve. When nodal temperature is included, the newly added nodes do have an initial temperature which can cause problems for subsequent calculations. With this fix, the newly added nodes have the average temperature of their constraining nodes.

§6.20 ICFD (Incompressible Fluid Solver)

1. Added input field AVER to ***ICFD_CONTROL_MESH** to select the automatic volumer mesher version. Version 14 is the default, but version 16 is also available.
2. Added mesh movement method where the movement is controlled by solving a linear elasticity problem using a constant size which is invoked by setting MMSH = 20 on ***ICFD_CONTROL_MESH_MOV**. This method is useful for avoiding large distortion in rotating problems that involve large discrepancies in mesh sizes (typically in cases involving boundary layer mesh).
3. Added generating level set surface with a sphere (STYPE = 3) to ***ICFD_INITIAL_LEVELSET**. Also, multiple definition of ***ICFD_INITIAL_LEVELSET** is allowed as kibg as the different shapes generated do not intersect.

4. Added one and two parameter Pierson-Moskowitz spectrum for irregular wave generation invoked with WTYPE = 6 and 7, respectively, on ***ICFD_BOUNDARY_FSWAVE**.
5. Made boundary conditions at free surfaces more stable and accurate. Stability has particularly increased in simulations that combine FSI with free surfaces.
6. Added input field TPS to ***ICFD_CONTROL_EMBEDSHELL** to control the fluid escape through triple points.
7. Added option OWC = 3 to ***ICFD_CONTROL_FSI** to perform two-way coupling in which the forces weakly couple (no sub-stepping) with the implicit mechanical solver
8. Added backflow stabilization for pressure boundary conditions
9. Added fields KS and CS to ***ICFD_BOUNDARY_PRESCRIBED_TURBULENCE** to prescribe local roughness height and roughness constant
10. Added input field DTT to ***ICFD_CONTROL_TIME**, so you can set a thermal timestep that is larger than the one used for the velocity/-pressure solve.
11. Added input flag DTBL to ***ICFD_CONTROL_TIME** to specify whether boundary layer elements should be included in the automatic time step calculation.

§6.21 Implicit (Mechanical) Solver

1. For ***CONTROL_IMPLICIT_SOLVER**:
 - We upgraded MUMPS (LSOLVR = 30) from version 5.2.0 to 5.3.5.
 - For intermittent eigenvalue analysis the direct solvers used by the eigenvalue analysis were forced for the entire execution. We now save and restore the solver option selected by the user to use during the non-eigenvalue execution of the run. This enables the use of iterative solvers.
2. For ***CONTROL_IMPLICIT_EIGENVALUE**:
 - Performance improvements for Block Lanczos (EIGMTH = 2) in the matrix-vector multiplies in MPP.

- Performance improvements for LOBPCG (EIGMTH = 102) in the preconditioner calculation.
 - Enabled additional keyword controls (optional Card 3) for MPP execution. These controls were initially only available for SMP.
 - Rewrote how data is output to the d3plot, d3eigv, and similar files to vastly improve performance of the output. This saves 15% of the wall clock time for large eigenvalue problems where lots and lots of eigenmodes are computed and written to d3eigv.
3. Enhanced ***PART_MODES** to work in single precision.
 4. Some contact types were being turned off for the inertia relief computation ***CONTROL_IMPLICIT_INERTIA_RELIEF**. These contacts are now enabled.
 5. Allow shell element 18 to be used in modal dynamics ***CONTROL_IMPLICIT_MODAL_DYNAMICS** when reading the modes from d3eigv.
 6. Enhanced the implicit solver's processing of nodal mass matrices to include the local coordinate transformation.
 7. For ***CONTROL_IMPLICIT_SOLUTION**:
 - Prior to this revision implicit linear analysis (NSOLVR = 1) used the forces at the end of the linear step to compute resultant forces. This did not match expectations of users. We changed the computation of resultant forces, such as those to bndout, to use the force at the beginning of the linear step.
 - Shell types 4 and 16 are supported for the nonsymmetric solver (LCPACK = 3), for the few materials that need this feature (***MAT_SAMP_LIGHT** currently being the only one).

§6.22 Initial Cards

1. Enabled ***INITIAL_VELOCITY_GENERATION** to work for solid element forms 24, 25, 26, and 61. Previously, only the first 8 nodes had velocities initialized.
2. For ***INITIAL_STRESS_SECTION**:
 - IZSHEAR = 2 now supports all cross section types. Previously cross sections specified with ***DATABASE_CROSS_SECTION_SET** could not be used.

- Bolt preload can now start from a nonzero stress state.
3. Added support for initializing solids with ***INITIAL_HISTORY_NODE(_SET)**. So far ELFORMs -2, -1, 1, 2, 10, and 13 are supported.

§6.23 Isogeometric Analysis (IGA)

1. Print out the first 100 smallest time steps and the element connectivities for IGA shell and solid elements. In SMP, the time step information is output to the d3hsp file; in MPP the time step information is printed out to each message file on each processor.
2. Support consistency flag (CONST in ***CONTROL_PARALLEL**) in SMP for IGA thick shells specified with ***ELEMENT_SHELL_NURBS_PATCH**.
3. Add cross section force to new IGA shell/solid (see ***DATABASE_CROSS_SECTION**).
4. ***DEFINE_STOCHASTIC_VARIATION** is supported for isogeometric analysis with shell elements in combination with some material models that allow the STOCHASTIC option (***MAT_015**, ***MAT_024**, ***MAT_081** and ***MAT_123**).
5. ***MAT_ADD_DAMAGE_GISSMO_STOCHASTIC** is supported for isogeometric analysis with shell elements.
6. Material models ***MAT_122** (***MAT_HILL_3R**) and ***MAT_122_TABULATED** (***MAT_HILL_3R_TABULATED**) are supported for isogeometric analysis with shell elements.
7. Support ***MAT_030** with IGA shells and solids.
8. Support ***MAT_SAMP-1** and ***MAT_SAMP_LIGHT** (***MAT_187**/***MAT_187L**) with IGA shells
9. Enable restart possibility for IGA elements.
10. Improved multi-patch coupling for thin (rotation-free) IGA shells with initial kinks.
11. Expand modeling capabilities of HAZ for IGA shells

§6.24 Load Cards

1. Accept thick shells in the input for ***LOAD_SEGMENT_FILE**
2. Implemented ***LOAD_SEISMIC_SSI_DECONV** for deconvolved earthquake input in soil-structure interaction and site-response analysis.
3. Added having ***LOAD_SEISMIC_SSI_AUX** read gmbin files created by ***INTERFACE_SSI_AUX** or ***INTERFACE_SSI_AUX_EMBEDDED** in a previous run. This allows multiple gmbin files to be used in the same run. Previously this was done as part of ***INTERFACE_SSI**, but that approach is now deprecated.
4. Added keyword ***LOAD_EXPANSION_PRESSURE** to apply a uniform pressure to a section of chamber that may change in size due to a moving edge. It works in both SMP and MPP.

§6.25 Materials and Equations-of-State

1. ***MAT_ADD_COHESIVE** now supports hyperelastic foam materials ***MAT_057** and ***MAT_083**.
2. For ***MAT_ADD_DAMAGE_DIEM**:
 - The FLD/MSFLD criteria now applies to solid elements, with certain mesh restrictions (consult manual for details). This feature allows for modelling membrane and bending failure relatively independent.
 - Exponential damage function for accurate post-instability behavior is available.
 - Add new parameter P5 to refer to a curve/table ID that defines
 - regularization factor vs. element size (curve) or
 - regularization factor vs. element size vs. abscissa value of criterion (table)Plastic strain at onset of damage can be scaled by that factor. This is similar to LCREGD in ***MAT_ADD_DAMAGE_GISSMO**.
3. Enabled GISSMO to be used with ***DEFINE_HAZ_PROPERTIES**. It is necessary to use ***MAT_ADD_DAMAGE_GISSMO_STOCHASTIC** to make it work.

4. Support VOLFRAC in ***MAT_ADD_EROSION** so that it works correctly with failure criteria in ***MAT_ADD_DAMAGE_DIEM/GISSMO** and ***MAT_ADD_EROSION** itself. Before it only worked with failure in materials, such as FAIL in ***MAT_024**.
5. For ***MAT_ADD_GENERALIZED_DAMAGE**:
 - Add new option IFLG3 = 2. This is a special case called "Domain of Shell-to-Solid Equivalence" or DSSE, where two damage variables are used but the second one is not really affecting the stresses.
 - Improved robustness for when DMGEXP < 1.
6. New features for ***MAT_ADD_INELASTICITY** include
 - (i) orthotropic field overlay for thermal expansion and failure
 - (ii) temperature shifts in hypo-viscoelasticity,
 - (iii) new hyper-viscoelastic laws,
 - (iv) support of higher order tetrahedra and
 - (v) norton-bailey creep.
7. For ***MAT_ADD_PZELECTRIC**:
 - Added support for 20-node solid elements (ELFORM = 23 in ***SECTION_SOLID**) and 10-node tetrahedron elements (ELFORM = 16 in ***SECTION_SOLID**).
 - Implemented axisymmetric and plane-strain formulation.
 - Added iterative solver.
 - Added optional dielectric loss parameters.
8. Extended to anisotropic behaviour for ***MAT_ADD_SOC_EXPANSION** by specifying LCIDY, MULTY, LCIDZ and MULTZ similar to ***MAT_ADD_THERMAL_EXPANSION**.
9. Added a negative MACF option for solid element orthotropic material models which causes the axis swapping to be done prior to the element rotation. This allows the rotation to be about any axis and not just the *c*-axis.
10. Added new option RATEOP = 5 to ***MAT_015** with VP = 1: strain rate dependent term based on Couque (2014).
11. For ***MAT_024**:

- Properly support VP = 3 in implicit analysis for solid and shell elements.
 - LCSS with VP = 3 now supports *DEFINE_TABLE_XD/COMPACT up to a level of 9, meaning yield stress can be a function of plastic strain, strain rate, and up to seven history variables. If ***MAT_ADD_DAMAGE_DIEM** is used together with ***MAT_024** with VP = 3, then P1 of ***MAT_ADD_DAMAGE_DIEM** also supports a table up to level 9.
12. For ***MAT_030** / ***MAT_SHAPE_MEMORY**:
 - Updated beam, shell and solid element tangent stiffness to account for plasticity.
 - Added warning message for too low plastic yield stress with respect to phase transformation limits.
 - Added curve/table dependence on plastic strain/temperature for transition stress parameters.
 13. Output "maximum effective strain" as history variable number 29 for ***MAT_034** (FORM = 14, -14, and 24).
 14. Add ***MAT_036** as a potential weld partner material for PRUL=2/3 of *DEFINE_CONNECTION_PROPERTIES.
 15. Support thermal option (thermal structure coupling) for ***MAT_USER_DEFINED_MATERIAL_MODELS** with SPH particles.
 16. Added rate filtering option, RFLTF, to ***MAT_064** with VP = 0 for solid elements.
 17. For ***MAT_068**:
 - Added velocity dependent hardening. Hardening curve IDs LCPD{R,S,T} and LCPM{R,S,T} can now refer to tables, meaning velocity dependent yield forces/moments can be defined.
 - Added velocity dependent failure. Negative input of {F,M,U,T}FAIL{R,S,T} can now refer to curves, meaning velocity dependent failure forces/moments/displacements/rotations can be defined.
 18. Added a new option for solid element spot weld failure. Setting TTOPT = 2 on ***MAT_SPOTWELD** (***MAT_100**) causes the failure calculate to use average forces and moments on the opposite weld faces

for the failure calculation. This makes the welds invariant in terms of how the weld elements are numbered. This works for single brick welds and solid weld assemblies.

19. Added keyword option UNIAXIAL for ***MAT_100** and ***MAT_100_DA**. In addition to setting the transverse shear stresses to zero ($E < 0$), the transverse shear strain increments get zeroed as well. This can reduce occasional instabilities.
20. For ***MAT_100_DA**, improved the material tangent in the plastic regime to yield better convergence in implicit analysis.
21. Implemented erosion flag for ***MAT_JOHNSON_COOK**/***MAT_110** for shells.
22. For ***MAT_112** / ***MAT_FINITE_ELASTIC_STRAIN_PLASTICITY**, implemented material tangent stiffness for solid elements.
23. For ***MAT_122_3D** / ***MAT_HILL_3R_3D**, updated plasticity algorithm to increase implicit stability.
24. Added non-linear viscoelastic model to ***MAT_123** ($VP = 1$) for shell and solid elements. Nonzero LCEMOD activates this model. This viscoelastic approach is adopted from ***MAT_187**.
25. Add strain rate filtering parameter, SRFILT, to ***MAT_124**. This enables strain rate filtering using an exponential moving average. Works for LCSRC or LCSRT being nonzero and SRFLAG = 0 or 1.
26. Added support for ***MAT_MODIFIED_HONEYCOMB** / ***MAT_126** with solid element types 0 and 9 to work with ***CONSTRAINED_TIED_NODES_FAILURE**.
27. Added option to specify the energy release rates for modes I and II (GIC and GIIC) in ***MAT_138** as element size dependent curves.
28. For ***MAT_DAMAGE_3** / ***MAT_153**:
 - Extended to allow for temperature dependence, including for elastic moduli, hardening parameters, and back stress parameters.
 - Enhanced accuracy for implicit simulations.
29. Improved robustness of ***MAT_154** / ***MAT_DESHPANDE_FLECK_FOAM** when using DERFI = 0 for the iteration of the equivalent stress.

30. Implemented a viscoelastic option for ***MAT_SIMPLIFIED RUBBER/FOAM / *MAT_181** for solid formulation 13, tetrahedron with nodal pressure.
31. Resolved single precision issue with small BETA values in ***MAT_187** to increase accuracy of results if LCEMOD > 0.
32. Improved tangents for ***MAT_SAMP_1** and ***MAT_SAMP_LIGHT** for the implicit solver.
33. Added new parameter RATEOP to ***MAT_SAMP_LIGHT (*MAT_187L)** to choose type of effective strain rate calculation.
34. Improvements to ***MAT_SAMP_LIGHT** to better deal with highly nonlinear inputs, such as sharp curves and logarithmic strain rate interpolation.
35. For ***MAT_199 / *MAT_BARLAT_YLD2004**, added possibility to have Young's modulus and yield stress as functions of plastic strain and temperature.
36. For ***MAT_BOLT_BEAM**, added feature (see input parameter HOLSHR) to enable modeling of a shear deformation mode in which the bolt shank tears through the plates. After the clearance gap has been closed, further shear deformation would enlarge the hole in the plates rather than deform the bolt itself. The force-deformation relation of this mechanism is still governed by LCSHR, but the deformation (i.e. enlargement of the hole) is tracked separately in each of four principal directions. Thus, enlargement of the hole in the positive *Y* direction has no effect on the position of the edge of the hole in the negative *Y* direction.
37. Align element failure behavior in ***MAT_224_GYS** with the one in ***MAT_224**, e.g., NUMINT = -200 should prevent element erosion at all.
38. Added two options to ***MAT_224_GYS**:
 - Keyword option LOG_INTERPOLATION invokes logarithmic interpolation of strain rates in tables LCK1, LCCR, and LCSR.
 - Added BETA < 0 with abs(BETA) referring to a curve, table, table_3d, or table_4d as in ***MAT_224**.
39. Young's modulus for ***MAT_226** can be a function of plastic strain.

40. Added keyword option **FUNCTIONS** to ***MAT_240** to include connection partner properties. With **FUNCTIONS**, the following parameters can/must be defined as ***DEFINE_FUNCTION** IDs: **EMOD**, **GMOD**, **G1C_0**, **T0**, **FG1**, **G2C_0**, **S0**, and **FG2** (and **GMOD3**, **G3C_0**, **R0**, and **FG3** if keyword option **3MODES** is also used). The arguments of these functions include thicknesses and maximum yield stresses of both weld partners (from tied contact) as well as the strain rate and element area. A similar capability is already available for ***MAT_169**.
41. Added new option for yield/damage initiation criterion of ***MAT_240**. Parameter **INICRT** can now be given as negative value, invoking a flexible exponent in the yield and damage initiation criteria.
42. Added new history variable number 15 to ***MAT_240**: "maximum strain value".
43. The coefficients α_i and m in ***MAT_242** can now be functions of effective plastic strain.
44. Added parameter **IHIS** to ***MAT_249** to allow initialization of fiber directions element-wise with ***INITIAL_STRESS_SHELL**.
45. Support ***MAT_251** in implicit analysis for solid and shell elements.
46. Add ***MAT_251** as potential weld partner material for **PRUL=2/3** of ***DEFINE_CONNECTION_PROPERTIES**. Therefore use values from constant inputs **PHASE1** and **PHASE2** to find the associated curve in a ***DEFINE_TABLE_3D** or ***DEFINE_TABLE_4D**.
47. For ***MAT_254**:
 - Added evolution equations for internal variables (grain size, dislocation density, globularization,...)
 - Accept user define functions for yield stress computations of individual phases
48. Add new option to define rate dependent strength in ***MAT_GLASS** (***MAT_280**). It is invoked by **FT < 0**, with curve ID **|FT|** specifying tensile strength vs. strain rate.
49. Minor change for ***MAT_280** with **ENGCRIT** and **RADCRIT**. If used together with ***MAT_ADD_EROSION** (where these criteria are also available), ***MAT_280** only "wins" if those parameters are nonzero.

50. For ***MAT_295** / ***MAT_ANISOTROPIC_HYPERELASTIC**, improved initial effective moduli computation used for both stable time step estimates as well as hourglass control.
51. For the ACTIVE module of ***MAT_295** / ***MAT_ANISOTROPIC_HYPERELASTIC**:
- Made the default of the active stress scaling factor unity.
 - Enabled activation/deactivation for all active models used in conjunction with electrophysiology coupling.
 - Included an activation threshold on the calcium concentration for active models 1-3 for electrophysiology coupling.
 - Increased number of history variables.
52. For ***MAT_SEATBELT**, improved the robustness of the 2D belt for $FORM \geq 2$.
53. Consolidation and enhancements of all orthotropic thermal materials (***MAT_T02**, ***MAT_T04**, ***MAT_T08**, ***MAT_T11-15**, ***MAT_T17**):
- New material axes definition AOPT = 3 implemented, defining material direction based on element normal and a vector d. First material direction is orthogonal to both of them.
 - Materials now all accept all the available options (AOPT = 0 through 4).
54. For ***MAT_T08** and ***MAT_T10**, heat generation, specific heat, and thermal conductivity (TGRLC, LCC, LCK*i*) can be load curves or tables depending on mechanical history variables.
55. NOCOPY = 1 in ***CONTROL_SOLUTION** invokes a keep-in-memory treatment of material history variables, thus avoiding the overhead of copying to/from local buffers. This may result in a significant reduction in simulation time, depending on how much this gathering/scattering of data takes in relation to the rest of the computations. The best result is seen when "fast" materials, like ***MAT_024**, is used together with gissmo/diem. This option works only for a selection of materials and elements; consult manual for details. Open-MP threading (ncpu > 1 or ncpu < -1) is not supported.
56. Added new user interface to define number of history variables on per part basis. New subroutine "umat_setnhvpp" is added to dyn21umat.F and called during initialization.

57. The new implementation of ***DEFINE_CURVE_STRESS** allows you to change the number of output stress pairs based on the value from LCINT of ***CONTROL_SOLUTION**.

§6.26 MPP

1. Add MPP rebalance and in-core adaptive support for ***DEFINE_CURVE_FUNCTIONS** that reference nodal quantities
2. The following are enhancements for redecomposition (***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION**):
 - Added start time field (STIME) to MPP redecomposition.
 - Added keyword option ONCE to ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION** to allow for performing the redecomposition only once and then keep the job running with the adjusted decomposition
 - Support redecomposition problem with dynamic relaxation with polar symmetry
 - Support implicit joints in redecomposition.
 - Add new feature to ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION** (input field WEIGHT) to estimate element cost based on element in contact and stress state to achieve better load balancing.
 - Support new partition of the model based on the previous redecomposition step's element cost (SAMPT).
3. Allow a later case to run using a predecomposition file from an earlier case.
4. Restructure MPP beam weld scale factor calculation for HAZ to have better consistency with SMP and different MPP ranks

§6.27 Multiscale

1. Added support for Hughes-Liu beams (ELFORM = 1 on ***SECTION_BEAM**) in the macroscale model of the multiscale spot weld. Previously only spot weld beams (ELFORM = 9) were supported.
2. Material models that support Hughes-Liu beams are available in the macroscale model of the spot weld when that beam formulation is used.

§6.28 Output

1. Remove discrete and seat belt visualization beams from the e1out file: they are NULL, have no meaningful data, and were filtered out anyway.
2. For ***DATABASE_RECOVER_NODE**:
 - Added support for one-point nodal pressure tetrahedron element (formulation 13).
 - Added support for solid formulations -1 and -2 for the elemental extrapolation method (METHOD = 1).
3. Added echoes in d3hsp for all default values set by ***CONTROL_THERMAL_FORMING**.
4. Enabled output of moments to the section force file, secforc, for 2D plane strain beam element form 7.
5. Added FILTER keyword option to ***DATABASE_SECFORC** for Butterworth filtering to smooth the forces. This is similar to the filter for ***DATABASE_NCFORC_FILTER**.
6. Added an optional card to ***DATABASE_BINARY_D3PART** to output element data for certain elements to d3part. Fields 1-4 of the optional card specify the set IDs of the solid elements, beam elements, shell elements, and thick shell elements for which the data will be output. This option will work in addition to PSETID defined in Card 1.
7. Added forces from ***LOAD_ERODING_PART_SET** to nodfor output.
8. Output 1D spherical part masses to d3hsp when ELFORM = -8 is specified on ***SECTION_ALE1D**.
9. Added ***DATABASE_ALE_OPERATION** to output user-defined ALE values from existing ALE data that can be fringed in d3plot and/or output in a history .xy file.
10. For ***DEFINE_MATERIAL_HISTORIES**
 - Plastic Energy Density is available as an output quantity.
 - Added keyword option NAMES. This will allow you to specify user-defined names for history variables.

§6.29 Restarts

1. Support LCTM of ***CONTROL_TIMESTEP** in small restart.

§6.30 Sensors

1. For ***SENSOR_CONTROL**:
 - Added TYPE = EM to control the EM solver.
 - Add TYPE = CONSTRL to control ***CONSTRAINED_LOCAL**.
 - Added TYPE = PZBC to control the application of the piezoelectric boundary condition.
2. Added keyword option UPDATE to all versions of ***SENSOR_DEFINE_OPTION** so that you can define how often a sensor will be updated.
3. Added VID = M and MMOMENT to ***SENSOR_DEFINE_FORCE** for measuring the magnitude of force and moment, respectively.
4. Allow as many as sensors as needed for ***SENSOR_SWITCH_CALC_LOGIC**.

§6.31 SPG (Smooth Particle Galerkin)

1. MPP version is now available.
2. The maximum number of neighboring particles is increased to 2048 in SPG dynamic memory allocation.
3. To model fiber reinforced composites with the Immersed SPG method, added a special MPP decomposition checkup to make sure that the immersed beam node and the corresponding SPG solid are on the same CPU. By doing this, it will guarantee the coupling between SPG solids and FEM beams when using immersed SPG.
4. Add the input field ALPHA_S to ***SECTION_SOLID_SPG** to control the surface particles' quadrature fraction. This is used to overcome shear locking in thin structures.
5. Added support for ***MAT_005**, ***MAT_014**, ***MAT_059**, ***MAT_124**, and ***MAT_193**.
6. New input format for the data cards of ***CONSTRAINED_IMMERSED_IN_SPG** to make setup easier

§6.32 SPH

1. Reduced memory footprint and hid some of the MPI communication latency for very large SPH models.
2. Improved performance of SPH models with a large ratio of inactive particles (injection, moving box,...) by creating a sublist of active particles.
3. Added option to explicitly calculate pressure on structure particles (see ***CONTROL_SPH_INCOMPRESSIBLE**).
4. Added field ISHOW to ***DEFINE_BOX_SPH** to add a dummy shell for visualization in LSPP.
5. Added time step condition based on viscous forces for incompressible SPH.
6. Added support of the Thermal Speedup Factor (TSF) on ***CONTROL_THERMAL_SOLVER** to the explicit SPH thermal calculation.
7. Added support for multiple ***DEFINE_SPH_MESH_BOX** keyword definitions.
8. Adaptive SPH (FORM = 9 and 10 on ***CONTROL_SPH**) supported in MPP.
9. Support thermal solver for thermal-structure coupling with adaptive SPH.
10. Added SPH particle shifting stabilization algorithm. This algorithm is invoked with ISHIFT on Card 3 of ***CONTROL_SPH**. It shifts particles slightly across streamlines to reduce particle clustering in the maximum compression and stretching directions.

§6.33 Thermal Solver

1. For ***CONTROL_THERMAL_FORMING**:
 - Support setting the default thermal contact parameters for contact types ***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE**, ***CONTACT_ONE_WAY_SURFACE_TO_SURFACE**, ***CONTACT_SURFACE_TO_SURFACE** and ***CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE**. Previously only forming contacts were supported.

- Added ability to define a default ***USER_INTERFACE_CONDUCTIVITY** for contact pairs without the THERMAL_FRICTION extension with ***CONTROL_THERMAL_FORMING** by setting FORMULA < 0.
 - Added initial temperature condition check for thermal thick shells. Default initial temperatures will be assigned to layers without initial temperatures to avoid unwanted temperature gradient at the 1st time step.
2. For ***CONTROL_THERMAL_SOLVER**, performance improvement in SMP of the GMRES solver for conjugate heat transfer (SOLVER = 17).
 3. New temperature scaling option for ***BOUNDARY_TEMPERATURE_RSW** and ***LOAD_THERMAL_RSW**. If load curve ID in the input is negative, the scaling operation acts on the difference between default and prescribed values in order to ensure a smooth temperature evolution in the weld nugget and heat affected zone.
 4. For ***LOAD_THERMAL_BINOUT** enhancements for handling multiple LSDA temperature files:
 - If one lsdA file has reached the final point in time and the mechanical run continues, keep the respective temperatures constant
 - If all lsdA files have reached the end time while the mechanical run continues, write a message and keep all temperatures constant.

§6.34 Miscellaneous

1. Significantly sped up some load curve evaluations, particularly if there are many points in the curve definition, by changing the search from linear to binary.
2. Added two sense switches:
 - SWE terminates dynamic relaxation and continues LS-DYNA with the transient analysis.
 - endtime=t causes the termination time to be changed to time t .
3. Added ECHG and ECHGBC to ***DEFINE_CURVE_FUNCTION** to output the reactive charge of piezoelectric materials.

4. Allow character parameters specified with ***PARAMETER** to be used as part of a larger string.
5. For ***USER_NONLOCAL_SEARCH**, check same material model is used within the part set for averaged and surrounding elements.
6. For ***DEFINE_PRESSURE_TUBE**:
 - Added Discontinuous Galerkin (MTD = 1) and Discontinuous Galerkin for the Isothermal Euler Equations (MTD = 2) to the solution methods.
 - Cavities (CAV_i) can now be visualized on beam elements in `d3plot` when using automatically generated shell/solid tubes (TYPE \neq 0).
7. For ***DEFINE_QUASAR_COUPLING**:
 - Output initial state from LS-DYNA for QUASAR
 - Allow user control of the frequency of data exchange between the two solvers with `FRCFRQ`
 - Support multiple quasar coupling interfaces
 - Added new option (`IOPT`) to allow user select different data exchanged between solvers
 - Cleanup and speedup quasar data collection and message passing routine
 - New feature to exclude node set from original list
 - Support local coordinates system for Quasar ROM model mapping to LS-DYNA global coordinates system.
8. Implemented ***DEFINE_DRIFT_REMOVE** for MPP.
9. Added a new flag, `IMONFLAG`, to field 5 of ***DEFINE_HAZ_TAILOR_WELDED_BLANK**, to allow load curves, `ISW` and `IFW`, from Card 2 of ***DEFINE_HAZ_PROPERTIES** to either increase monotonically (`IMONFLAG` = 0) or arbitrarily (`IMONFLAG` = 1).
10. For ***RAIL_TRAIN**:
 - Output file containing wheel/rail forces is now csv-format by default, enabling easier post-processing (see input parameter `FMT`)
 - New option for wheel flanges to be on the outside of the rails rather than on the inside (see input parameter `LATDIR`)

- New option for friction resisting lateral sliding across the rail (see input parameter FRIC).
11. Add keyword ***DEFINE_TABLE_COMPACT**. It allows a more compact notation of 2D-, 3D-, 4D-, ... tables. Everything takes place in the keyword reader, where a ***DEFINE_TABLE_{X}D** is created internally.
 12. Extended the ***USER_LOADING** interface to support thick shell elements.
 13. The following are enhancements for specifying the FMI co-simulation settings (***COSIM_FMI_CONTROL**):
 - Allow different time units for the solvers
 - Allow the FMU to have a time step that is smaller than LS-DYNA's time step
 14. Added more helpful warning and error messages in `fmu_msg` for co-simulations

§7 Bug Fixes

§7.1 Acoustics, NVH, and Frequency Domain

1. For ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**:
 - Fixed Rayleigh method (METHOD = 0) for the HALF_SPACE keyword option when using triangular elements.
 - Added the missing incident wave in the Rayleigh method for pressure output in fringe plots.
 - Fixed bug in restarting BEM using `d3dump`.
2. Fixed a bug in running FRF ***FREQUENCY_DOMAIN_FRF** with pressure load (VAD1 = 4).
3. For ***FREQUENCY_DOMAIN_RANDOM_VIBRATION**
 - Fixed a bug in dumping `d3psd`, `d3rms` and `d3zcf` when `neips` and `neiph` in ***DATABASE_EXTENT_BINARY** are not zero.
 - Fixed a bug in random vibration analysis with random pressure load (VAFLAG = 2) when the normal direction of the surface elements is not in the x , y , or z axis direction.

4. For ***FREQUENCY_DOMAIN_RESPONSE_SPECTRUM**, fixed a bug in combining ground motion with structural relative motion, when the load is defined by time history (LCTYP = 10,11,12).
5. Fixed a bug in SSD analysis ***FREQUENCY_DOMAIN_SSD** with pressure load (VAD = 1) when the normal direction of the surface elements is not in x, y, or z axis direction.

§7.2 Adaptivity

1. Fix for MPP in-core adaptivity when initial refinement is active. In some cases it was not refining elements it should have, resulting in invalid adaptive constraint situations.
2. Fix for MPP in-core adaptivity which was generating inconsistent user IDs for shared nodes in some cases, resulting in invalid DYNAIN files (and probably other unknown issues).
3. Fix bug in in-core adaptivity for forming blanks that occurs when some of the elements in the blank failed during the simulation.
4. Fixed remeshing of parts with voids for 2D *r*-adaptivity ($|\text{ADPTYP}| = 8$ on ***CONTROL_ADAPTIVE**). Previously, the voids were remeshed with the elements. With this fix, the voids remain as voids and are not remeshed with the elements.
5. Fixed bug causing error message STR+389 (solid elements referencing a non-existing part) that occurs when using ***RIGIDWALL_GEOMETRIC_FLAT_MOTION_ID_DISPLAY** with adaptivity.
6. Fixed bug in 3D *r*-adaptivity invoked with ***CONTROL_ADAPTIVE** (ADPTYP=7) and ***CONTROL_REMESHING** when used with cases. The bug caused edges and corners to be lost during *r*-adaptivity after the first case.
7. Fixed bug that caused incorrect output to d3plot for 3D *r*-adaptivity (ADPTYP = 7 on ***CONTROL_ADAPTIVE**). It was incorrect when using interactive adaptivity IAT = 3 (***CONTROL_REMESHING**) in implicit analysis.
8. Fixed a bug to obtain correct resultant forces for prescribe motion constraints applied to a box region of space (***BOUNDARY_PRESCRIBED_MOTION_SET_BOX**) when 3D *r*-adaptivity (ADPTYP =

7 on ***CONTROL_ADAPTIVE**) is enabled. The results are incorrect when the region in the box is adapted.

9. Fixed bug in 3D r -adaptivity so the stresses in non-adapted parts are correct.

§7.3 Airbag Cards

1. Fix inconsistent results between runs which was caused by incorrect CV decomposition data on local processors.
2. Fix ***AIRBAG_HYBRID** structure input translation error under long = s
3. Fix mixing of reference geometry and reference shell geometry
4. The following are bug fixes for CPM airbags (***AIRBAG_PARTICLE**):
 - Fixed error in vent mass flow rate calculation after the switch from a corpuscular particle method airbag to a uniform pressure airbag. This error occurs with or without a chamber definition (CHM on Card 6).
 - Fixed issue with CPM/UP switch when multiple chambers are interacting.
 - Fixed issue associated with the initial air particle assignment for a multiple airbags model. The bug caused results to depend on the order of the airbags in the input.
 - Fixed issue in reading XM_i (Card 12) in the ***AIRBAG_PARTICLE** input that occurred when the ID associated with ***DEFINE_CPM_GAS_PROPERTIES** is at least 9 digits long.
 - Fixed bug for multiple CPM bags that occurred when NPRLX was the number of cycles for some of the bags and the end time for the other bags. This issue occurred only in MPP.
 - Fixed species porous leakage output in `abstat_cpm`

§7.4 ALE

1. Fixed a bug that occurs when ***BOUNDARY_NON_REFLECTING_2D** is used with S-ALE. The bug caused errors when the geometry was a ring. It also caused errors in MPP.

2. Bug fixes for ***ALE_STRUCTURED_MESH_TRIM**, ***ALE_STRUCTURED_MESH_VOLUME_FILLING**, and ***ALE_STRUCTURED_MESH_MOTION** in Windows for issues leading to crashes.
3. For ***ALE_STRUCTURED_FSI**:
 - Fixed bugs that were causing crashes or incorrect results when the structure contained triangular segments.
 - Fixed bugs that were causing crashes or instabilities (time step decreasing) due to memory issues.
4. Fixed the data output by ***ALE_STRUCTURED_MESH** to the mapping file. This file is read by ***INITIAL_ALE_MAPPING**.

§7.5 Boundary Cards

1. Fixed a bug associated with ***BOUNDARY_NON_REFLECTING** which might cause a failure in locating a segment belonging to a 10-node tetrahedron element.
2. Fixed behavior of ***BOUNDARY_NON_REFLECTING** in implicit.
3. For ***BOUNDARY_PRESCRIBED_MOTION_SET_BOX**, fixed an MPP bug that happens when the node set is part of a rigid body.
4. For ***BOUNDARY_TEMPERATURE_TRAJECTORY**:
 - Fixed bug causing incorrect error message "redundant temperature boundary conditions". It occurred when the model contains only solids and THSHEL = 1 on ***CONTROL_SHELL**.
 - Fixed the local coordinate system bug which results in an incorrect normal calculation when NSID2 < 0 and the surface normal jumps.
 - Fixed bug that caused an incorrect welding torch position when multiple welding paths exist.
5. Corrected a precision conversion problem associated with using super-elements in explicit single precision with velocity boundary conditions.

§7.6 CESE Compressible Fluid Solver

1. Corrected an MPP d3plot bug when outputting surface fluid or moving mesh or conjugate heat transfer boundary surface mechanical solver variables.
2. Fixed ***CESE_DATABASE_POINTOUT** for the 2D case where the user sets some z -coordinates to be nonzero. This practice is not recommended since it can lead to other problems, especially for FSI cases. Also, please note that ***CESE_DATABASE_POINTOUT** ignores the ***CESE_EOS** cards, making the assumption that the EOS is a perfect gas EOS with $\gamma = 1.4$.
3. For the ***CESE** moving mesh solvers, correct the nodal connectivity for the edge/prism element case.
4. Several bug fixes were made that make these solvers more stable.

§7.7 *CHEMISTRY

1. For TBX flow, fixed the problem of negative temperatures when using ***CHEMISTRY** with ***CESE**.

§7.8 Constraint Cards

1. For single precision fixed gradual underflow issue that occurred with ***CONSTRAINED_INTERPOLATION_GLOBAL**.
2. Fixed issue with ***CONSTRAINED_JOINT_STIFFNESS_TRANSLATIONAL**. When such a joint is activated with a sensor and $JNTF = 2$ in ***CONTROL_RIGID**, you can have a scenario where lower (called negative in the manual) stop displacements are positive, or upper (called positive in the manual) stop displacements are negative. The logic in the force calculation did not cover this scenario correctly.
3. Fixed bug associated with the penalty formulation of ***CONSTRAINED_SHELL/SOLID_IN_SOLID** in the MPP version that was causing LS-DYNA to crash.
4. Fixed incorrect initial velocity of ***CONSTRAINED_NODAL_RIGID_BODY** that occurs when using ***INITIAL_VELOCITY_GENERATION** with $NX = -999$. The rotation axis was incorrect.

5. Corrected error in the implementation of ***CONSTRAINED_INTERPOLATION** that was leading to a core dump.
6. Fix for ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) with huge number of associated nodes. A memory issue was likely to occur, if more than 200 nodes were found in the domain of influence of the upper or lower shell element parts.
7. Fixed bug for ***CONSTRAINED_BEAM_IN_SOLID** and ***CONSTRAINED_SHELL_IN_SOLID**: coupling between beams/shells and thick shell elements did not work correctly in the MPP version.
8. Defining multiple ***CONSTRAINED_BEAM_IN_SOLID** instantiations with 3D adaptivity did not work correctly.

§7.9 Contact

1. Fix for MPP dynamic load balancing with ***CONTACT_DRAWBEAD** which was mixing up the node order along the drawbead if the nodes as input were not in increasing number order.
2. Fix initial tying in MPP groupable tied contact which was not honoring the part specific thickness from ***PART_CONTACT**. This issue could result in some nodes not being tied that should be.
3. Fix for MPP tied contact birth/death times when given by load curves.
4. Include all slave nodes in the bounding box for ***CONTACT_SINGLE_SURFACE** in MPP when performing bucket sort – even those not from segments.
5. Fixes for load balancing of eroding contacts
6. Fix for MPP eroding contact to properly handle element deletion during small deck restarts
7. Fix for MPP groupable drawbead behavior when the contact is off the edge of a thin triangle. The (u, v) coordinates were not being truncated along the $u+v=1$ line. Thus, the computed mass was coming out negative due to extrapolation. This resulted in NaNs during the friction calculation.
8. Fix bug in input processing of ***CONTACT_BEAM_TO_SURFACE** in MPP

9. Fix for MPP groupable ***CONTACT_DRAWBEAD**, which had an overly restrictive bucket sort and so could miss some of the ramp up force a drawbead should create. Also fix incorrect handling of viscous damping for this contact.
10. Add some missing initialization code for groupable ***CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET** which could cause instabilities when these contacts have a nonzero birth time.
11. Fix for cohesive element failure issue with MPP groupable tied contact. Groupable tied contact does not initialize until the first cycle, so checking the "is this node tied" flags before then was causing premature element failure.
12. Changes for MPP SOFT = 0 and 1 contact with birth time, to check for any contact (shell) segments that may have failed before the contact became active. Previously, these checks only happened once the contact was active and so neglected to delete contact segments from shells that failed before the contact birth time. This fix covers both groupable and non-groupable ***CONTACT_AUTOMATIC_SINGLE_SURFACE** and ***CONTACT_AUTOMATIC_..._TO_SURFACE**.
13. Fix for ***CONTACT_ERODING** in MPP, which was not treating exposed segments correctly
14. Free hash table related to 2 sided force transducers in MPP, thereby fixing a resource exhaustion issue that caused them to stop working after 9 adaptive steps (or redecomposition steps).
15. Suppress warning saying force transducers aren't supported with full deck restarts. They have no history so re-initializing them from scratch works just fine.
16. Fix bug in MPP groupable contact that could cause incorrect behavior for contact nodes that should be deleted due to element failure.
17. Fixed an MPP bug for surface-to-surface contact with SOFT = 1 that happens the contact involves 2D seat belt elements that are initially inside a retractor.
18. Enabled segment-to-segment (SOFT = 2) eroding contact to work correctly when elements erode before the contact birth time.

19. Fixed force transducers when used with groupable MPP segment-to-segment (SOFT = 2) contact. A memory error termination was likely.
20. Fixed ***TERMINATION_CONTACT** to make it work when used with THRES > 0 or DOF = 2 or 3. Some data was scrambled causing wrong behavior.
21. Fixed force transducers when used with MPP contact with SOFT = 0 or 1. The fixed types are **AUTOMATIC_GENERAL**, **SURFACE_TO_SURFACE**, **NODES_TO_SURFACE**, and **ONE_WAY_SURFACE_TO_SURFACE**. Forces were sometimes being assigned to the wrong interfaces or to nonexistent interfaces, in which case a memory error could have caused a segmentation fault.
22. Fixed MPP solutions with 2D force transducers (***CONTACT_2D_FORCE_TRANSUDCER**) that have both a slave and master surface defined.
23. Fixed a flaw in the segment-to-segment (SOFT = 2) contact thickness calculation. It could cause shell segments near welds to reduce to zero thickness when SPOTHIN is active on ***CONTROL_CONTACT**, shell thickness updates are active, and either OPTT > 0 ***PART_CONTACT** or OPTT = 0 and SFT = 0 on ***PART_CONTACT** and either SST > 0 or MST > 0 on ***CONTACT**.
24. Previously ***CONTACT_AUTOMATIC_GENERAL** in SMP missed the contact between beams generated with ***ELEMENT_BEAM_SOURCE**.
25. Added field TIEDIED to optional card D on ***CONTACT_OPTION**. Setting this field to 1 causes an incremental update of the contact normal. Incremental updates were added to prevent spurious stresses that occur even with no loading when using ***CONTACT_TIEBREAK_NODES_TO_SURFACE**.
26. Fixed bug causing ***CONTACT_TIED_SURFACE_TO_SURFACE** to be ineffective after dynamic relaxation when death time is set to -9999.
27. Fixed bug causing an input error saying the element has zero nodal thickness (STR+230) when using triangular elements with shell type 23 (8-node quadratic quadrilateral shell) for contact.

28. Fix issue causing an immediate core dump during runtime or memory error upon termination when using `SOFT = 0` or `1` tied contacts with `SOFT = 2` surface-to-surface and single surface contacts and running with SMP.
29. Fixed incorrect duplication of higher order shell(23 and 24) parts for contact.
30. Fixed bug that was resulting in contact not being detected when the slave box (SBOXID) is defined with a ***DEFINE_BOX_LOCAL** and the master box (MBOXID) is defined with a ***DEFINE_BOX** as specified on Card 1 of ***CONTACT_OPTION**.
31. Fixed seg fault or incorrect frictional behavior when using the `THERMAL_FRICTION` keyword option for ***CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE** and ***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE** and running with SMP.
32. Fixed a bug that occurred during initialization for MPP when there were a large number of contacts in the model. The bug caused a segmentation fault.
33. Fix a bug that that entered the code in 2015 for ***CONTACT_SURFACE_TO_SURFACE** in SMP. LS-DYNA was performing one-way contact instead of two-way contact. As a result, the contact force was half of what it should be.
34. Extended the logic for collecting the resultant forces reported to the INTFOR file for ***CONTACT_TIED_SURFACE_TO_SURFACE** when using implicit in MPP. Before the resultant forces were being reported as zeroes for this contact type.
35. Added a warning that ***CONTACT_TIEBREAK_NODES_TO_SURFACE_ID** is not supported for implicit computations.

§7.10 Control Cards

1. Issue warning message when the number of iterations reaches the upper limit, `MAXITR`, in ***CONTROL_PZELECTRIC**.
2. Fixed incorrect mass scaling, `DTMS2<0` in ***CONTROL_TIMESTEP**, when solid type 48 (***SECTION_SOLID_PERI**) is used in the presence of the keyword ***INITIAL_FOAM_REFERENCE_GEOMETRY**.

3. For any kind of implicit eigenvalue computation, LS-DYNA before ignored the value of TET10S8 on ***CONTROL_OUTPUT** and only wrote the corner nodes of the elements to d3plot for the connectivity. Now it accepts the setting of TET10S8, so the full connectivity can be written to the d3eigv database.
4. For staged construction, fixed bug related to unexpected forces being present in dormant elements. The bug affected solids (hourglass type 6 only), thick shells (all hourglass types), and shells (stiffness-method hourglass types). When a part is dormant (meaning the construction stage when the part becomes active has not yet been reached), stresses and forces are calculated but are scaled down (see input parameter FACT on ***CONTROL_STAGED_CONSTRUCTION**). The scaling down had been erroneously omitted for the hourglass forces pertaining to the element types and hourglass settings mentioned above.
5. For ***CONTROL_PORE_FLUID**:
 - Fixed bug that existed in R12 but not previous versions such as R11. In some models containing ***CONTROL_PORE_FLUID**, the output to d3plot was wrong. The pore pressure variables could be written to the wrong position in the file, resulting in stresses or other extra history variables being overwritten with pore pressure, and the extra history variables that were expected to contain pore pressure data instead contained zero. The bug apparently occurred only with small simple models and did not seem to affect large complex models.
 - Fixed bug affecting ATYPE = 4 (steady state consolidation), which could occasionally occur when thin elements are sandwiched between thick elements. The calculation could become unstable in the first few time steps, with pore pressure oscillating between high positive and negative values. A work-around for the bug was to use the curve ACURVE on ***BOUNDARY_PORE_FLUID** to set the analysis type to time-dependent consolidation (ATYPE = 3) for at least 100 time steps before switching to ATYPE = 4.
 - Fixed two bugs affecting pore fluid analysis with flow through tied contact (FTIED on ***CONTROL_PORE_FLUID**):
 - If a node on the tied contact reached the suction limit (see SUCCLIM on ***BOUNDARY_PORE_FLUID**), the suction limit pressure then spread along the tied contact in a non-physical way.

- Pressures could become unstable at nodes shared across MPP processors if the node belonged to more than one tied contact.

§7.11 Discrete Element Method

1. Fix torque evaluation from rolling friction for DES surface coupling (***DEFINE_DE_TO_SURFACE_COUPLING**)
2. Fix bug of negative bond damage that occurred with ***DEFINE_DE_ACTIVE_REGION**.

§7.12 Elements

1. Bug fix for PARM3 for discrete beams defined with ***ELEMENT_BEAM_THICKNESS**. The local coordinate ID specified by PARM3 was ignored by the discrete beam material models ***MAT_069**, ***MAT_070**, ***MAT_071**, ***MAT_074**, ***MAT_094**, ***MAT_197**, ***MAT_205** and ***MAT_208**. The global coordinate system was used instead.
2. Fixed a 2D belt bug that happens when a belt has two retractors (***ELEMENT_SEATBELT_RETRACTOR**).
3. For ***USER_NONLOCAL_SEARCH**:
 - Fixed bug in memory allocation with respect to number of requested user element history variables.
 - Fixed bug related to reading of history variable lines of the user element keyword.
 - Fixed bug that may cause segmentation fault for user element history variables (NUELHSV > 0).
4. Fixed the elbow beam (***SECTION_BEAM**, ELFORM = 14) when the inside pressure is zero. The bug could cause a wrong result.
5. Reduced round-off error in the initial volume calculation for thick shell forms 3, 5, and 7 so that they can correctly initialize the pressure to zero when used with equation-of-state materials.
6. Fix incorrect mass/inertia when using rigid bodies affected by ***ELEMENT_MASS_PART** and ***CONSTRAINED_RIGID_BODIES**. The mass contributions from ***ELEMENT_MASS_PART** were not added to the merged rigid bodies from ***CONSTRAINED_RIGID_BODIES**.

7. Fixed spurious error message KEY+1567, saying 'ELEMENT ### is not found,' that occurs using ***ELEMENT_SHELL_COMPOSITE** if ***MAT_ADD_THERMAL_EXPANSION** is present in the input.
8. Corrected the setting of nodal inertias for solid element formulation 3 for eigenvalue computations. It was zero and now is the standard computed value.
9. Corrected the computation of the critical time step for explicit when using superelements (***ELEMENT_DIRECT_MATRIX_INPUT**).
10. Fix for ***PART_STACKED_ELEMENTS**: memory overwrite. Models could fail with "Error 10183 (KEY+183), part ... not defined", even if that part did not exist at all.
11. Fixed a bug writing out dynain and reading in ***INITIAL_STRESS_SHELL** for shell type 55.

§7.13 Initial Cards

1. For ***INITIAL_FOAM_REFERENCE_GEOMETRY**, fix an SMP bug that occurs when $NCPU > 1$.
2. Fixed SMP bug where $ISTIFF < 0$ in ***INITIAL_STRESS_SECTION** could cause a segmentation fault.
3. Fix for ***INITIAL_AXIAL_FORCE_BEAM** with beam element 1 (H-L beam). Available since R12.0, this feature could have lead to various side effects, such as severe beam deformations.

§7.14 Implicit (Mechanical) Solver

1. Fixed incorrect handling of multiple right-hand-sides during the forward / back solve for the Power Method. The Power Method extracts buckling modes (see ***CONTROL_IMPLICIT_BUCKLING**).
2. Corrected the processing of energy and other such statistics in file `glstat` for the multistep linear solution method ($NSOLVR = -1$ on ***CONTROL_IMPLICIT_SOLUTION**). Prior to the bug fix, the statistics were additive, meaning the statistics for the new time step were added to the ones from the previous time step. Now the statistics are reset at the beginning of the new time step, so they just reflect the current time step.

3. For ***CONTROL_IMPLICIT_EIGENVALUE**:
 - Removed implicit input/output files at the end of any eigencomputation. These were scratch files that were of no interest to the user.
 - Corrected an error associated with the the nonsymmetric eigensolver which impacted any problem with damping.
4. Fixed the generation of superelements using ***CONTROL_IMPLICIT_MODES** in MPP when nodes involved were shared across processors.
5. Fixed application of implicit constraint modes ***CONTROL_IMPLICIT_MODES** in MPP for the case where a constraint mode is also a shared node.
6. Correct how the load curve IDs for IAUTO and IMASS were being processed for implicit keywords ***CONTROL_IMPLICIT_DYNAMICS** and ***CONTROL_IMPLICIT_AUTO** with the options DYN or SPR enabled. Before if the curve IDs did not match the input order (such as an ID of 97125 for the curve input, then LS-DYNA would give an error. This is not how curves are normally processed. Now they are being processed like all other curves in LS-DYNA were the curve ID and order of input are independent.
7. For ***CONTROL_IMPLICIT_GENERAL**, adjust logic of when last intermittent eigenvalue computation is performed during an explicit simulation if the last one is performed near the termination time. For instance, if the termination time is 10.00 and the explicit time step is 0.01, then the explicit calculation terminates between 9.99 and 10.00. Therefore, if you wanted an eigenvalue computation performed at 10.00, it would not happen because the calculation would have already terminated. This change causes the intermittent eigenvalue computation to occur before the termination time.
8. During MPP decomposition, the nodes on an interface (see ***INTERFACE**) may be shared by multiple processes. We fixed a bug in implicit where we were double counting the interface forces for the shared nodes.
9. Correct a long standing problem associated with computing the Lagrange Multipliers for constraint processing in Implicit Mechanics.

The displacements on the independent dofs were inappropriately being zeroed during this computation leading to inaccurate reporting of resultant forces for Implicit Mechanics.

10. In the previous releases, we had reduced the size of `d3e1gv` database for models with no rotational dofs like solid element only models. Modal Dynamics ***CONTROL_IMPLICIT_MODAL_DYNAMICS** was still trying to read the nonexistent data for rotational dofs and became lost. This revision corrects that oversight.
11. Restored the ordering reuse logic to the MPP implementation of implicit dynamics. This is an important time saving feature for very large models with multiple time steps.

§7.15 Isogeometric Analysis (IGA)

1. Fixed a bug that causes the deletion of the wrong interpolation shell elements (IGA).

§7.16 Load Cards

1. Corrected effective seismic input calculation for tetrahedral and pentahedral elements.

§7.17 Materials and Equations-of-State

1. Fixed the use of equation of state materials with thick shells, when ***PART_COMPOSITE** or ***INTEGRATION_SHELL** was not used to define an integration rule.
2. Fix for ***MAT_ADD_DAMAGE_GISSMO** with `LCSDG < 0` to avoid incorrect element deletion (only SMP/Hybrid with `ncpu > 1`).
3. Correction for ***MAT_ADD_DAMAGE_GISSMO** and ***MAT_ADD_GENERALIZED_DAMAGE** in full deck restart. Element size for `LCREGD` and `FADEXP < 0` was re-computed after restart, but it should not.
4. Fix for ***MAT_ADD_GENERALIZED_DAMAGE** with `long = s`. The long input format was not working with this keyword before.
5. For ***MAT_ADD_PZELECTRIC**:

- Fixed a bug that happens when coupling coefficient for shear, P_{ijk} , is nonzero.
 - Fixed a bug for strain-based input that happens when a part and its material have different IDs.
6. Fixed bug to make 2D shell type 13 with NIP = 4 work correctly with ***MAT_ADD_THERMAL_EXPANSION** for implicit.
 7. Fixed ***MAT_NONLOCAL** when used with eroding elements. The smoothing calculation was using uninitialized data due to a failure to set initial values for the eroded elements prior to smoothing. This error could be subtle or cause an error termination.
 8. Fixed the use of material type 26, 126, or 201 with thick shell elements when AOPT = 2 or 3. The material directions were being miscalculated causing wrong stress.
 9. Fixed incorrect initial strains when using ***MAT_MOONEY-RIVLIN_-RUBBER / *MAT_027** with ***INITIAL_FOAM_REFERENCE_GEOMETRY** and running with single precision version.
 10. For ***MAT_030 / *MAT_SHAPE_MEMORY**
 - Fixed issue where shells failed to converge for plasticity.
 - Fixed issue where incorrect curve ID was used in LCSS for solids.
 - Fixed issue that could cause a segmentation fault when using this material.
 11. for ***MAT_053** extrapolate load curve for von Mises stress as a function of negative volumetric strain using the first 2 points of the load curve if the volumetric strain is less than the first point of the curve. This fix prevents an error termination.
 12. For ***MAT_LAMINATED_COMPOSITE_FABRIC (*MAT_058)**:
 - Fix contact stiffness in case EA, EB, or GAB are defined with load curves or tables:
 - Fixed bug for using ≥ 8 digit load-curve ID for GAB
 - Fixed bug for LCDFAIL check. Prior to this fix the LCDFAIL option did not work for shells!
 13. Fix for ***MAT_076** with temperature shift. Initial temperature in an uncoupled analysis is not always zero (such as coming from ***LOAD_-THERMAL**).

14. Fix for combination of ***MAT_077 / *MAT_181** with VFLAG = 1 and ***INCLUDE_TRANSFORM**. Normalized shear relaxation moduli G_i (unitless) should not be converted.
15. Fixed incorrect effect of BVFLAG in card 2 field 1 of ***MAT_FU_-CHANG_FOAM / *MAT_083** on initial time step size.
16. Fixed bug in ***MAT_WINFRITH_REINFORCEMENT / *MAT_084_-REINF**: Reinforcement when defined with specific groups of elements (Card 1a) did not work in MPP. It would error terminate.
17. Fixed bug in ***MAT_100 (*MAT_SPOTWELD)** for OPT = 0/-1 when using ***DEFINE_CURVES** for defining force/moment resultants as a function of the effective strain rate, by setting NRR, NRS, NRT, MRR, MSS, MTT as negative values.
18. Fixed bug in damage evolution of ***MAT_SPOTWELD_DAMAGE-FAILURE**. Prior to this fix an earlier and wrong damage evolution could have happened. This is for beam elements only.
19. Fix initial temperature increment in ***MAT_106** for shell elements. In a structural-only analysis, the "old" temperature should be set to the initial temperature, e.g. from ***LOAD_THERMAL_LOAD_CURVE**.
20. Fixed incorrect computation of phel for ***MAT_110 / *MAT_JOHNSON_HOLMQUIST_CERAMICS** that occurs when you input phel = 0 in the data cards.
21. For ***MAT_JOHNSON_HOLMQUIST_CONCRETE / *MAT_111**, bug fix in the erosion criteria.
22. For ***MAT_157_IHIS / *MAT_ANISOTROPIC_ELASTIC_PLASTIC**:
 - Fixed bug for IHIS = 2 where history variables were not read correctly.
 - Corrected the update of transversal (through the thickness) strain. Prior to this correction, the thickness strain was not computed correctly and did not fulfill the condition that was set with the Lankford parameters (R -values). This applies to shells only.
23. For ***MAT_172**:
 - Fixed bug affecting shear capacity check option TYPESC = 2. Note that the shear capacity calculated by ***MAT_172** affects only the

output history variables, not the deformations or stresses. The constants in the equations have been corrected to match ACI 318-05M equations 11-4 and 11-8. Before this change, any axial compressive stress would have caused the reported shear capacity to be greater than required by 11-4. Furthermore, minor corrections have been made to other terms to match the metric edition of ACI 318 - previously the constants had been converted from imperial units with a greater level of precision than in ACI 318-05M.

- Fixed bug in implicit solution that occurred when TYPEC = 3 or 6 and PR36 was left zero: a crash occurred on the first time step. A work-around was to set nonzero PR36.
24. Fix problem with long=s for ***MAT_196** to avoid error termination.
 25. Fixed bug in ***MAT_BOLT_BEAM** that existed in R12 but not earlier versions such as R11. The combination of LCSHR being a DEFINE_TABLE with AXSHFL = 0 did not work correctly, i.e. the shear force plastic limit would not be the one expected based on LCSHR and the current shear deformation.
 26. Make option E < 0 of ***MAT_224_GYS** work correctly, which was not the case before.
 27. For ***MAT_CAZACU_BARLAT** / ***MAT_233**, removed a redundant term in the hardening modulus calculation when the material hardening is temperature and strain rate sensitive.
 28. Fixes for ***MAT_PIECEWISE_LINEAR_PLASTIC_THERMAL (*MAT_255)**:
 - Avoid error termination if huge number of curves (> 9999) is used in a model.
 - Avoid error termination with large curve IDs for E < 0 and PR < 0.
 29. Fixes for ***MAT_258**: repair visco-plasticity, use correct thickness of shell type 16, and eliminate potential dependence on ncpu.
 30. For ***MAT_LAMINATED_FRACTURE_DAIMLER_PINHO (*MAT_261)**:
 - Check for fiber kinking, even if FMAT = 1.0. This allows for proper behavior if fiber compression and matrix tension is present at the same time.

- Define minimal values for fracture toughnesses if not defined to avoid division by zero.
 - Make sure that stress will not exceed the stress at damage initiation (i.e. $\sigma_{\text{kink}} > \sigma_{\text{kink}_0} / \sigma_t > \sigma_{t_0}$). The damage variable has to be at least high enough to limit the stress. There might be special scenarios where the stress keeps growing while the driving strain is not (which would lead to no damage accumulation).
31. Fixed issue with ***MAT_264** / ***MAT_TABULATED_JOHNSON_-COOK_ORTHO_PLASTICITY** which gives incorrect results when running with adaptivity.
 32. Fixed bug in ***MAT_274** where shell thickness strain was not updated properly.
 33. For ***MAT_295** / ***MAT_ANISOTROPIC_HYPERELASTIC**:
 - Fixed a bug which let the simulation run even if negative determinants were reported in the stress update routine in MPP.
 - For the ANISO module, fixed a bug for fiber models other than the Holzapfel-Gasser-Ogden; the compression tension switch is now uniformly based on the total fiber stretch.
 - For the ANISO module, fixed a bug when using fiber stress vs stretch relations defined by a curve.
 - Fixed a bug in the keyword reader of the ACTIVE module.
 34. For ***MAT_SEATBELT**
 - Fixed a bug for 2D seat belts that can result in seg faults when there is no retractor in the model.
 - Fixed a bug when initializing initial strain for seat belt elements initially inside a retractor.
 35. For ***MAT_SPH_VISCOUS**:
 - Fixed bugs to make this material model work properly with SPH particles.
 - Fixed bugs of the user subroutine inside ***MAT_SPH_VISCOUS** for the user defined viscosity to make this option work properly.
 36. Fixed bug causing an error termination with error message STR+1475 that occurred when mixing materials with resultant and non-resultant materials in the same ***INTEGRATION_SHELL** or ***PART_-COMPOSITE**.

§7.18 MPP

1. Fix failure during MPP tied contact initialization caused by ***AIRBAG_REFERENCE_GEOMETRY**.
2. Fix MPP to support general functions when performing pre-decomposition based runs.
3. Fix dynamic load balancing issue that could cause deadlock when processors were being removed from contact interfaces.
4. Fix for MPP pre-decomposition in models with airbags. When pre-decomposition is done for more processors than the calculation, the airbag ownership was not being properly handled leading to problems including possibly seg fault.
5. Fix bug in ***TERMINATION_DELETED_SOLIDS**, which wasn't terminating properly in MPP.
6. Fix some initialization checks related to ***PART_AVERAGED**, and prevent execution of related code on processors that have none of these beams (MPP only)
7. Add MPP support for ***TERMINATION_BODY**, which was never properly implemented in MPP.
8. Fix for possible divide by 0 during dynamic load balancing and in-core adaptivity on processors with no contact interfaces
9. Fix for MPP treatment of ***CONSTRAINED_SHELL_TO_SOLID** which in rare cases, depending on the specific decomposition, could have generated NaNs during initialization when scaling the node rotations for stability.
10. Fix for MPP handling of ***DEFINE_CURVE_FUNCTION**. Specifically, if the built-in function PIDCTL was used, along with MPP pre-decomposition, the behavior was unpredictable.
11. For ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION**, fix full deck restart with deleted rigidwall.

§7.19 Multiscale

1. Fixed bugs associated with including several different mesoscale spot weld models, each included with ***INCLUDE_MULTISCALE_SPOTWELD**.
2. Fixed the offset option for the contact between the beams in the macroscale spot weld and the master interface when the offset distance is larger than the beam length

§7.20 Output

1. Fixed issue in MPP that resulted in incorrect reporting of rotational masses to `massout`.
2. Fixed bug that caused shell data to be skipped in the `d3thdt` output routines ***DATABASE_BINARY_D3THDT**. This issue resulted in corrupted `d3thdt` files.
3. For thermal-only simulations, honor the setting of `BINARY` in ***DATABASE_CURVOUT**. It was previously only outputting an ASCII format even if an LSDA format was requested.
4. Removed very old diagnostic write statements that could have potentially corrupted the `deforc` file in any model using ***MAT_WOOD**.
5. For linear DK shell elements (***SECTION_SHELL**, `ELFORM = 18`) the moments are now correctly written to the cross section output generated by ***DATABASE_SECFORC**. Before, incorrect values were reported.
6. Fixed error in `e1out` for beam, solid, shell and `tshell` elements that occurred if a beam ID had 9 digits.
7. Fixed error in `nodout` that occurred if a node ID had 9 digits.
8. Restored how encrypted files are echoed to `d3hsp` back to how it was done in version R9.3.1.
9. Fix for `DCOMP = 5` and `6` on ***DATABASE_EXTENT_BINARY** option when `PSET` on ***DATABASE_BINARY_D3PLOT** option is also used. This issue was corrupting the `d3plot` file.
10. Fixed a bug pertaining to the gathering of nodal volumes in MPP for solid element formulation 13. This fixes issues with erratic stress distributions in `d3plot`.

11. Honor the setting of IEVERP in ***DATABASE_EXTENT_BINARY** for thermal only calculations.
12. For ***DATABASE_CROSS_SECTION**:
 - Fixed a bug that happens when a cross section contains 2D seat belts.
 - Fixed bug where cross-sectional area was not displayed properly in `secforc` for local coordinate systems.
 - Fixed bug causing NaN in `secforc` when using beam type 6 with zero area and length.
 - With this fix, forces/moments are being output to `secforc` for beam type 13.
13. For ***DATABASE_RECOVER_NODE**:
 - Fix an MPP bug that might happen when there are 10-node tetrahedron element in the model.
 - Issue warning message when no vertex is detected for Zienkiewicz-Zhu method (`METHOD = 0`).
14. Fixed a bug that happens when `NEWLEG` of ***CONTROL_OUTPUT** is > 0 and `SBTOUT` (seat belt output with ***DATABASE_SBTOUT**) is not requested.
15. Fixed segmentation fault when writing out dump file after dynamic relaxation when using a large number of layers in ***ELEMENT_SHELL_COMPOSITE**. This affects MPP only.
16. Fixed bug so that the strain tensor is output to `d3part` when `STRFLG > 0` in ***DATABASE_EXTENT_D3PART** and `STRFLG = 0` in ***DATABASE_EXTENT_BINARY**. Previously, the strains were not being output to `d3part`.
17. Fixed zero torsion output to `swforc` for beam spot welds.
18. Fixed incorrect ***INITIAL_INTERNAL_DOF_SOLID_TYPE3** output to the `dynain` file. It caused negative volume when used to initialize stresses in solid types 3 and 4.
19. Fixed pressure computation in `intfor` so that it matches the pressure computed using forces from `ncforc` and the segment area.

20. Fixed a bug in the moment computation for the moments output to `rcforc` (***DATABASE_RCFORC_MOMENT**). This bug only affects post-processing, not the analysis itself.
21. Fixed possible bug in output of stresses for non-isotropic materials when using `CMPFLG = 1` on ***DATABASE_EXTENT_BINARY** and ***DEFINE_MATERIAL_HISTORIES**. Prior to this fix the output of stresses might not have been properly transformed into the material frame. This only affects the post-processing and not the analysis itself.
22. Added missing legend/title to ***DATABASE_DISBOUT** output files.
23. Corrected time step output for the SPG controlled time steps to `d3hsp`.

§7.21 Restarts

1. Fix simple/small restart with ***LOAD_SEGMENT_FILE**. It was reading the incorrect family file.
2. During simple restart, only open an ***INTERFACE_LINKING** file on processor 0 in MPP.
3. Fixed compatibility of ***DEFINE_SPH_TO_SPH_COUPLING** with full deck restart.
4. Fixed bug in full deck restarts of models containing cylindrical joints. Some joint history data was not recovered properly.
5. Bug fix for all restart jobs run with double precision executables. The bug was a memory error that might lead to erroneous results and existed in all double precision executables since June of 2012.
6. Fix seg fault with full deck restart when initial run is 6 degrees-of-freedom while the full deck restart run is 3 degrees-of-freedom.
7. Fixed small-deck restart for ***LOAD_SEISMIC_SSI**, ***INTERFACE_SSI**, and ***LOAD_SPCFORC**.
8. Fixed bug causing an error during full deck restart for ***CONTACT_TIEBREAK_SURFACE_TO_SURFACE**.

§7.22 Sensors

1. Fix old typo in *SENSOR related to tied contacts, which should have converted them to GROUPABLE but was only working correctly for the first contact in the model.
2. For *SENSOR_CONTROL:
 - Fixed a bug that failed to recover failed elements (TYPE=ELESET)
 - Fixed a bug for TYPE = PRESC-MOT that happens when the node or rigid part to which the boundary condition is to be applied has a big ID.
 - Fixed a bug for TYPE = BAGVENTPOP. The bug caused venting to not turn off for *AIRBAG_HYBRID.
 - Fixed a bug for TYPE = SPC that was triggered when trying to turn on SPC in the middle of the analysis.
3. Fixed a bug for MTYPE = BNDOUT of *SENSOR_DEFINE_MISC.
4. Fixed an MPP bug for FTYPE = JOINT of *SENSOR_DEFINE_FORCE that might result in a seg fault.
5. Fixed an MPP bug for *SENSOR_DEFINE_FORCE with TYPE = JOINTSTIF.
6. Fixed a bug for *SENSOR_DEFINE_ELEMENT with TYPE=STRAIN that occurs when STRFLG of *DATABASE_EXTENT_BINARY is "0".
7. Fix for *SENSOR_DEFINE_FORCE with nonzero CRD and long=s. Error in structured read happened (STR+211) with MPP and np > 1.

§7.23 SPH

1. Fix for *DEFINE_ADAPTIVE_SOLID_TO_SPH now that we allow up to 8 particles per direction, per element. It was not working correctly if there were more than 5 particles per direction.
2. Bug fix in *DEFINE_ADAPTIVE_SOLID_TO_SPH with tet elements. The parametric coordinates were incorrect, so particles were created in wrong locations.

§7.24 Thermal Solver

1. For $MXDMP \neq 0$ (matrix dumping for $SOLVER > 11$) on ***CONTROL_THERMAL_SOLVER**, fixed incorrect output in MPP.
2. An issue was fixed when ***DEFINE_CURVE_FUNCTION** is used in thermal calculations.
3. LS-DYNA echoes a reasonable error message when a view factor file is missing and ***BOUNDARY_RADIATION_OPTION1_VF_READ** is used.
4. An input trap was added when using a wrong TYPE option in ***BOUNDARY_RADIATION_SET**.
5. Avoid an error termination when ***LOAD_THERMAL_LOAD_CURVE** is used with adaptivity.
6. A reasonable message is echoed if the d3heat file used with keyword ***LOAD_THERMAL_D3PLOT** is not specified in the command line, and LS-DYNA will exit gracefully.
7. Enabled ***LOAD_THERMAL** to work for beam nodes when nodal releases are defined at those nodes. The thermal loading was incorrect.

§7.25 XFEM (eXtended Finite Element Method)

1. Fixed a bug in defining a pre-crack for XFEM. Added a warning message when changing precrack defining points so the pre-crack cannot pass nodal points.
2. Fixed a bug in 2D XFEM ($ELFORM = 52$ in ***SECTION_SHELL_XFEM**) so it works when applied to plasticity materials.

§7.26 Miscellaneous

1. To improve memory performance, removed initial large allocation that was zeroing all of memory and thereby physically allocating it on Linux. This change restores behavior that existed in releases prior to R11.2
2. Store file name and format flag for ***INTERFACE_COMPONENT_FILE** into the structured file. With this change, if we convert keyword input to structured input and run from the structured input,

Listing 1

```
1 *PARAMETER
2 $   PRMR1      VAL1      PRMR2      VAL2
3 RuPLscXXa1 &R1uP11SARuPLscXXt1      0.0
```

LS-DYNA will know the required file format for the component data. Previously, LS-DYNA always produced a non-lsda file when running from a structured file.

3. Fix ordering issue with ***DEFINE_FUNCTIONs** in the initialization phase (structured input) such that all of these are before ***DEFINE_CURVE_FUNCTIONs**. This prevents a ***DEFINE_CURVE_FUNCTION** that reference a ***DEFINE_FUNCTION** from being before the ***DEFINE_FUNCTION**.
4. Fix memory issue caused by ***USER_INTERFACE_FRICTION**.
5. Fixed a bug that happens when BOX or VOL is used for ***SET_NODE_GENERAL** in models containing ***NODE_TRANSFORM**.
6. Fix indexing error related to ***SET_SEGMENT_GENERAL**, which could have resulted in incorrect values being applied for the DA1-DA4 parameters, or possibly a segfault during input processing.
7. Fixed bug causing memory allocation problems for PGI binaries running models with thick shells output to the e1out file
8. Corrected some bugs related to ***INTERFACE_LINKING** that could lead to segmentation faults.
9. Fix bugs causing memory allocation issues for executables built with the PGI compiler.
10. Fixed bug in reading a character parameter (***PARAMETER**) in a comma-delimited input. It did not conform to the usual comma-delimited input conventions.
11. Fixed bug in reading data cards of ***PARAMETER** that occurs if there is no space between a symbol and a value like in [Listing 1](#).
12. For ***DEFINE_CURVE_FUNCTION**:
 - Fixed an MPP bug for FX, FY, FZ, TX, TY and TZ in that might result in erroneous result.

- Fixed a bug for function SENSOR that sometimes failed a job and printed a bogus message.
 - Fixed a bug for PIDCTL that occurs during thermal-only simulations.
13. Fixed a bug that happens in double precision of PGI machines associated with ***INCLUDE_NASTRAN**.
 14. For ***DEFINE_PRESSURE_TUBE**:
 - Fixed bug in equilibrium equations for Continuous Galerkin (MTD = 0).
 - Removed bottleneck in MPP when using many rigid parts in the model.
 15. Keyword: ***MODULE** Fixed bug in argument lists for user subroutines when using ***MODULE**. Before this fix, some user subroutines (umatc, usrshl, usrsld, usrfrc, usrtie, usrtbrk, usr_nunonl, mortar_*, rebar_*, loadsetud) may not have worked correctly.
 16. Fixed a problem with ***NODE_TRANSFORM** when used with spot weld assembly generation. Spot weld elements were getting distorted.
 17. Fixed seg fault that occurs during the output of a d3plot file after a solid element is deleted. This behavior occurred when the material model was a user material.
 18. Fix 'reference load curve ID # cannot be found' error, KEY+825, when load curves for ***DEFINE_TABLE_2D/3D** are contained in a different include files after the include file that contains the ***DEFINE_TABLE_2D/3D** statement.
 19. Fixed bug causing error 'BEAM INT ID ### does not exist' (KEY+308) when using ***INTEGRATION_BEAM** with ***INCLUDE_TRANSFORM**.
 20. Fixed issue with ***DEFORMABLE_TO_RIGID_AUTOMATIC** when CODE = 4 and OFFSET > 0.0. Offset was not working correctly.
 21. For ***SET_OPTION_COLLECT**, fixed a bug in which the code stops with an error message when the first set of this kind is empty.

22. Few fixes and improvements for keyword memory handling of larger models, mainly affecting keywords ***SET_NODE**, ***SET_SOLID**, ***SECTION_SOLID**, ***INITIAL_STRESS_SHELL**, and ***AIRBAG_REFERENCE_GEOMETRY**. Time spent in keyword reading phase is reduced.
23. Activate error/warning for failed/unconstrained spot weld assembly solids (***DEFINE_HEX_SPOTWELD_ASSEMBLY**). Up to now, this error or warning (depending on SPOTSTP from ***CONTROL_CONTACT**) was only issued for single spot weld solids.

Release Notes for R13.0.0

November 19, 2021

Contents

1 Release notes revision	3
2 License	4
3 Documentation	4
4 Notes	4
5 New Features	4
5.1 Acoustics, NVH, and Frequency Domain	4
5.2 ALE	5
5.3 Constraint Cards	6
5.4 Control Cards	6
5.5 Discrete Element Method	6
5.6 Dual CESE Compressible Fluid Solver	6
5.6.1 Dual CESE Compressible Multiphase CFD Solvers	8
5.7 Elements	11
5.8 EM (Electromagnetic Solver)	11
5.9 ICFD (Incompressible Fluid Solver)	12
5.10 Implicit (Mechanical) Solver	12
5.11 Isogeometric Analysis (IGA)	12
5.12 Load Cards	13
5.13 Materials and Equations-of-State	14
5.14 MPP	15
5.15 Multiscale	15
5.16 Output	15
5.17 Peridynamics	16

5.18	SPG (Smooth Particle Galerkin)	16
5.19	SPH	17
5.20	Thermal Solver	17
5.21	XFEM (eXtended Finite Element Method)	17
6	Improvements and Enhancements	18
6.1	Acoustics, NVH, and Frequency Domain	18
6.2	Adaptivity	19
6.3	Airbag Cards	20
6.4	ALE	20
6.5	*BATTERY (Electrochemistry Solver)	23
6.6	Blast	24
6.7	Boundary Cards	24
6.8	CESE Compressible Fluid Solver	25
6.9	*CHEMISTRY	25
6.10	Constraint Cards	25
6.11	Contact	26
6.12	Control Cards	28
6.13	Discrete Element Method	30
6.14	Dual CESE Compressible Fluid Solver	31
6.15	Elements	32
6.16	EFG (Element Free Galerkin)	33
6.17	EM (Electromagnetic Solver)	33
6.18	Fatigue Cards	34
6.19	Forming Analysis	34
6.20	ICFD (Incompressible Fluid Solver)	35
6.21	Implicit (Mechanical) Solver	36
6.22	Initial Cards	38
6.23	Isogeometric Analysis (IGA)	38
6.24	Load Cards	39
6.25	Materials and Equations-of-State	39
6.26	MPP	46
6.27	Multiscale	47
6.28	Output	47
6.29	Restarts	48
6.30	Sensors	48
6.31	SPG (Smooth Particle Galerkin)	49
6.32	SPH	49
6.33	Thermal Solver	50
6.34	Miscellaneous	51

7 Bug Fixes	53
7.1 Acoustics, NVH, and Frequency Domain	53
7.2 Adaptivity	53
7.3 Airbag Cards	54
7.4 ALE	55
7.5 Boundary Cards	56
7.6 CESE Compressible Fluid Solver	56
7.7 *CHEMISTRY	57
7.8 Constraint Cards	57
7.9 Contact	57
7.10 Control Cards	61
7.11 Discrete Element Method	62
7.12 Elements	62
7.13 Initial Cards	64
7.14 Implicit (Mechanical) Solver	64
7.15 Isogeometric Analysis (IGA)	66
7.16 Load Cards	66
7.17 Materials and Equations-of-State	66
7.18 MPP	70
7.19 Multiscale	71
7.20 Output	71
7.21 Restarts	74
7.22 Sensors	74
7.23 SPH	75
7.24 Thermal Solver	75
7.25 XFEM (eXtended Finite Element Method)	76
7.26 Miscellaneous	76

§1 Release notes revision

This file constitutes revision 1 of the release notes for LS-DYNA version R13.0.0. In this revision, we fixed some typos. ***DATABASE_D3MAX** was incorrectly called ***DATABASE_BINARY_D3MAX**. Three notes were not included in the previous version. These are item 11. in §6.10, item 11. in §6.28, and item 15. in §6.34.

We also previously stated that there would be a Volume IV User’s manual. The contents that would have been in Volume IV are in Volumes I and II.

§2 License

The string "REVISION 13" must appear in the LS-DYNA license file in order to run version R13. Please contact your LS-DYNA distributor or your Ansys sales representative if you have to have your license updated.

§3 Documentation

Until such time the R13.0 User's manual is posted at www.lstc.com/download/manuals, the DRAFT User's Manual may be used as a temporary substitute. Note that there may be features mentioned in the DRAFT User's manual which are not implemented for R13.0.0.

§4 Notes

The remainder of this file provides a brief description of new features, enhancements, and bug fixes in version R13.0.0. Some of the bug fixes may have also been included in other recent releases.

The items are arranged by category in the sections New Features, Improvements and Enhancements, and Bug Fixes. Understand that in many cases, a particular item may pertain to more than one category. In the interest of brevity, each item is listed only once, under a single category.

§5 New Features

§5.1 Acoustics, NVH, and Frequency Domain

1. Added keyword ***CONTROL_ACOUSTIC_SPECTRAL** to request an acoustic spectral element analysis.
2. Added keyword ***CONTROL_IMPLICIT_SSD_DIRECT** to request a direct complex solution to steady state vibration.
3. Added new boundary condition keywords for acoustics: ***BOUNDARY_ACOUSTIC_COUPLING_SPECTRAL**, ***BOUNDARY_ACOUSTIC_FREE_SURFACE**, ***BOUNDARY_ACOUS-**

TIC_IMPEDANCE_COMPLEX, ***BOUNDARY_ACOUSTIC_IMPEDANCE_MECHANICAL**, ***BOUNDARY_ACOUSTIC_INTERFACE**, ***BOUNDARY_ACOUSTIC_NON_REFLECTING**, ***BOUNDARY_ACOUSTIC_PRESCRIBED_MOTION**, and ***BOUNDARY_ACOUSTIC_PRESCRIBED_SPECTRAL**

4. Added keyword ***CONTROL_ACOUSTIC_COUPLING** for altering defaults parameters for ***BOUNDARY_ACOUSTIC_COUPLING_MISMATCH** and ***BOUNDARY_ACOUSTIC_COUPLING_SPECTRAL**.
5. Added keyword ***LOAD_ACOUSTIC_SOURCE** for specifying loading sources for acoustic spectral analyses (***CONTROL_ACOUSTIC_SPECTRAL**) and SSD analyses (***CONTROL_IMPLICIT_SSD_DIRECT**).
6. Added keyword ***INTERFACE_ACOUSTIC** to create an interface file of boundary motions that can be used by ***BOUNDARY_ACOUSTIC_INTERFACE** for weakly coupled acoustic fluid-structure interaction in **SSD_DIRECT** analyses.
7. Added ***DATABASE_ACEOUT** and ***DATABASE_HISTORY_ACOUSTIC** to control the output for acoustic analyses.

§5.2 ALE

1. Added support for S-ALE using multi-materials to work with the thermal solver.
2. Added option **COVER_LAG** to ***ALE_STRUCTURED_MESH_MOTION** to expand the S-ALE mesh to cover and follow a Lagrangian structure.
3. Added field **ISYM** for the **FOLLOW_GC** option of ***ALE_STRUCTURED_MESH_MOTION** to constrain the S-ALE mesh expansion / contraction along one or more symmetry planes.
4. 2D S-ALE is now fully supported with major bug fixes and algorithm enhancements. Prior to this release, 2D existed but was experimental and under development.

§5.3 Constraint Cards

1. Added feature invoked with ***CONSTRAINED_SOIL_PILE** that enables modeling of piled foundations in soil/structure interaction models. Applications include seismic engineering and staged construction modeling. See the R13 keyword manual for further details.
2. First experimental version to support solid element parts connected by ***CONSTRAINED_SPR2**.

§5.4 Control Cards

1. Implemented a feature for finding an approximate reference configuration given a measured (deformed) geometry and boundary conditions. It is invoked with ***CONTROL_REFERENCE_CONFIGURATION**.

§5.5 Discrete Element Method

1. Added keywords ***DEFINE_DE_MESH_BEAM** and ***DEFINE_DE_MESH_SURFACE** for coupling beam and shell structure elements to ICFD elements through DES shadow elements on the surface of the structure elements.
2. Added keyword ***DEFINE_DE_BOND_OVERRIDE** to support the modification of bond properties within a certain node set, part set or part.

§5.6 Dual CESE Compressible Fluid Solver

1. With R13, multiphase models are introduced in the DUALCESE framework for the first time. An expanded description of these solvers will be given in §5.6.1.
2. Dual CESE Boundary Conditions:
 - ***DUALCESE_BOUNDARY_CYCLIC**. Cyclic or periodic boundary condition for parallel or rotational sectors
 - ***DUALCESE_BOUNDARY_SOLID_WALL_ROTATE**. Rotating solid wall boundary conditions
 - ***DUALCESE_BOUNDARY_SLIDING**. Sliding boundary condition for special cases involving the moving mesh FSI solver where

the nodes move along a surface that maintains its shape, like the surface swept out by the motion of the outside surface of a piston

3. Dual CESE equations-of-state:

- REFPROP v10.0 is now used (replaces REFPROP v9.1 support), and makes more engineering-type real gas and real liquid equations-of-state available to the user (***DUALCESE_EOS_REFPROP**)
- New equations-of-state that are intended to be used with the new multiphase / multifluid dual CESE solvers described later:
 - ***DUALCESE_EOS_COCHRAN_CHAN**. Condensed-phase explosive EOS of type Cochran-Chan
 - ***DUALCESE_EOS_JWL**. Condensed-phase explosive EOS of JWL-type
 - ***DUALCESE_EOS_STIFFENED_GAS**. A stiffened gas type fluid EOS
 - ***DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED**. A generalized Van der Waals type of EOS

4. Dual CESE Reaction Rate:

- ***DUALCESE_REACTION_RATE_IG**. A reaction rate law for the Ignition and Growth model to describe the conversion of reactants to products in the modeling of a condensed-phase explosive
- ***DUALCESE_REACTION_RATE_IG_REDUCED**. A reduced form reaction rate law for the Ignition and Growth model to describe the conversion of reactants to products in the modeling of a condensed-phase explosive
- ***DUALCESE_REACTION_RATE_P_DEPEND**. An explicitly pressure-dependent reaction rate law for describing the conversion of reactants to products in the modeling of a condensed-phase explosive

5. New multiphase solver for multiphase flow problems with two immiscible materials. Keywords associated with this are:

***DUALCESE_EOS_SET**

***DUALCESE_PART_MULTIPHASE**

***DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE**

***DUALCESE_INITIAL_TWO-PHASE**

***DUALCESE_INITIAL_TWO-PHASE_SET**

6. A new hybrid multiphase model solver useful for modeling multiphase / multifluid systems that involve condensed phase explosives. Note that the reaction rate laws and condensed-phase explosive equations-of-state need to be employed for the case of explosion modeling. Keywords associated with this solver are:

***DUALCESE_EOS_SET**

***DUALCESE_PART_MULTIPHASE**

***DUALCESE_BOUNDARY_PRESCRIBED_HYBRID**

***DUALCESE_INITIAL_HYBRID**

***DUALCESE_INITIAL_HYBRID_SET**

7. Support the ***MESH** (ICFD) mesher with boundary layer meshing. ***MESH** keywords are now available for use in creating ***DUALCESE** volume meshes.
8. Immersed boundary method (IBM) FSI with material erosion is now supported for the 3D case, where the eroding elements can be from the following structural element types: solid, thick shell, and thin shell elements.

§5.6.1 Dual CESE Compressible Multiphase CFD Solvers

Our multiphase solver is based on a hybrid formulation, i.e., a mixture of multiphase formulation and augmented Euler formulation. It is designed for the numerical simulation of condensed-phase explosives. It can handle two miscible components, namely the reactant and the products of reaction, and one inert immiscible component, such as the confiner in the case of the rate-stick.

We also implemented a solver for one of the hybrid formulation reduced models, i.e., a two-phase model (a 5-equations model). In this model, only two immiscible materials can be included. Another reduced model, i.e., a fluid-mixture limiting case, will be implemented in the near future.

Notable features of this hybrid model include that it can provide the accurate recovery of the temperature field across all components, and also can sustain large density differences across material interfaces. It

can accommodate realistic EOS and arbitrary (pressure or temperature based) reaction-rate laws.

The main applications of these multiphase solvers include the numerical simulation of condensed-phase commercial- and military-grade explosives. Of particular interest is the propagation of detonations in compliantly-confined charges (rate-stick-type problems) and the sensitization of commercial explosives by means of collapsing micro-balloons. For example,

- simulation of combustion and transition to detonation of condensed-phase explosives
- propagation of detonations in compliantly-confined charges (rate-stick-type problem)
- sensitization simulation of commercial explosives by means of collapsing micro-balloons
- shock-induced cavity collapse in liquid explosives

Currently available EOSs: (we will implement more EOSs based on user requirements in the future)

- Mie-Gruneisen type EOSes: Ideal-gase EOS, JWL EOS, and Cochran-Chan EOS
- generalized Van der Waals EOS, including standard Van der Waals EOS and a stiffened gas EOS

Currently available reaction rate laws: (we will implement more models based on user requirements in the future)

- Ignition and Growth (I&G) model
- Simplified Ignition and Growth (I&G) model
- Simple pressure dependent reaction rate law

Note: Multiphase flow and structure interaction (FSI) capabilities will be added in the near future.

Keywords involved with these new multiphase solvers:

***DUALCESE_BOUNDARY_PRESCRIBED_HYBRID**

***DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE**

***DUALCESE_EOS_COCHRAN_CHAN**
***DUALCESE_EOS_JWL**
***DUALCESE_EOS_STIFFENED_GAS**
***DUALCESE_EOS_VAN_DER_WAALS_GENERALIZED**
***DUALCESE_EOS_IDEAL_GAS**
***DUALCESE_EOS_SET**
***DUALCESE_INITIAL_HYBRID**
***DUALCESE_INITIAL_HYBRID_SET**
***DUALCESE_INITIAL_TWO-PHASE**
***DUALCESE_INITIAL_TWO-PHASE_SET**
***DUALCESE_PART_MULTIPHASE**
***DUALCESE_REACTION_RATE_IG**
***DUALCESE_REACTION_RATE_IG_REDUCED**
***DUALCESE_REACTION_RATE_P_DEPEND**

Solver Characteristics:

- Explicit
- Double precision
- Dynamic memory handling
- SMP and MPP
- 3D solver / special case 2D solver and 2D axisymmetric solver

Solver Main Features:

- dual CESE method (Conservation Element / Solution Element) used for solving the system of multiphase partial differential equations.
- Highly accurate shock wave capturing

§5.7 Elements

1. Added two solid element formulations ***SECTION_SOLID**:
 - Element formulation 62 is an 8 node fully integrated solid brick element that uses an assumed strain field of incompatible modes.
 - Element formulation -18 is an enhanced assumed strain hex/brick formulation for coarse mesh accuracy. It contains 13 incompatible modes to avoid shear and volumetric locking. It is primarily intended for implicit analysis of thin components.

§5.8 EM (Electromagnetic Solver)

1. Erosion of elements can be combined with the batmac model (***EM_RANDLES_BATMAC**) for nail penetration and localized short study
2. Added support for ***EM_RANDLES_EXOTHERMIC_REACTION** to batmac (***EM_RANDLES_BATMAC**)
3. Added contribution of positive current collector to Joule heat rate in batmac (***EM_RANDLES_BATMAC**).
4. Added support for pentahedral elements in ***EM_RANDLES_SOLID**.
5. Added ability to replace the ECM (Randles) model in batmac with a user defined model (***EM_RANDLES_BATMAC**)
6. Implemented NREL's four-equation model in the thermal solver for batter thermal runaway study. This is invoked with ***LOAD_HEAT_EXOTHERMIC_REACTION**.
7. Introduction of mortar contact to treat FEM part in Eddy current solver (***EM_CONTROL_CONTACT**)
8. Support ***EM_CIRCUIT_SOURCE** with inductive heating solver
9. Support the 2D axisymmetric solver with the inductive heating solver
10. Added the monolithic solver (***EM_SOLVER_FEMBEM_MONOLITHIC**) for simulations involving permanent magnets and ferromagnetic materials.
11. Added ***EM_PERMANENT_MAGNET** to define a permanent magnet.

12. Addition of several cell models as well as a user defined cell model. See keywords ***EM_EP_CELLMODEL_FITZHUGH-NAGUMO**, ***EM_EP_CELLMODEL_FENTONKARMA**, ***EM_EP_CELLMODEL_TENTUSSCHER**, and ***EM_EP_CELLMODEL_DEFINEFUNCTION**
13. The EM solver can be coupled with the explicit thermal solver.

§5.9 ICFD (Incompressible Fluid Solver)

1. Added new keyword ***ICFD_BOUNDARY_PRESCRIBED_LEVELSET** to prescribe the fluid height on a boundary.
2. Added keyword ***ICFD_BOUNDARY_NAVIERSLIP** to impose a Navier-slip boundary condition. It works like a regular free slip condition except we are calculating a shear term and adding it to the right-hand side (just like the turbulent law of the wall)
3. Added keyword ***ICFD_BOUNDARY_FSI_FIXED** to allow a solid surface to "slide" over a fluid surface without the fluid surface moving. This boundary condition is mostly useful for conjugate heat transfer applications.

§5.10 Implicit (Mechanical) Solver

1. Added implicit incompressible control volumes for simplified analysis of pressure vessels.
2. For ***CONTROL_IMPLICIT_SOLVER**:
 - LSOLVR = 30 (MUMPS) can now be used for nonsymmetric problems.
 - The fill-reducing method invoked with ORDER = 4 (LS-GPart) is now supported for eigenvalue problems.

§5.11 Isogeometric Analysis (IGA)

1. Added IGA shell element formulations 5 and 6 (***SECTION_IGA_SHELL**) that are analogous to shell element formulations 25 and 26 (***SECTION_SHELL**).

2. Added ***IGA** keyword family for isogeometric analysis. This set of keywords sets up and controls the isogeometric-related capabilities of LS-DYNA. It is fully compatible with standard CAD. Connections are automatic through topology. It supports unstructured splines. It includes extensions for NURBS volumes. It supports boundary conditions, contacts, constraints, and connections (such as welds with HAZ, RBEs, bolts) on exact CAD geometry.
3. Extended mechanical coupling of shell patches to T-joints with ***IGA_TIED_EDGE_TO_EDGE**
4. Geometry can be input using elements in Bezier decomposed form with ***IGA_INCLUDE_BEZIER**.
5. Introduced a stabilization scheme for light-control points for trimmed shell NURBS patches in SMP and MPP (see ***IGA_SHELL** and ***ELEMENT_SHELL_NURBS_PATCH**). The basis functions in the trimmed geometry have reduced support which can result in adverse numerical effects in both implicit and explicit.
6. Full MPP support for tied-edge-to-edge contact, for tying multiple NURBS patches along their topological connected interfaces with ***IGA_TIED_EDGE_TO_EDGE**.
7. Allow the use of ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) for IGA through interpolation elements. (SMP/MPP)
8. Enable implicit contact in MPP via interpolation elements which applies to both ***ELEMENT_SHELL/SOLID_NURBS_PATCH** and ***IGA_SHELL/SOLID**.
9. Added support for using rigid walls with IGA.
10. Enable Dirichlet boundary conditions at a parametric edge using Nitsche's method with ***BOUNDARY_PRESCRIBED_MOTION**.
11. New meshing scheme for interpolation shells that allows for non-convex domains.
12. Enable nodal boundary conditions (fully MPP support) with ***IGA_POINT_UVW**.

§5.12 Load Cards

1. Added keyword ***LOAD_PYRO_ACTUATOR** to provide a simple way to simulate hood lift pyrotechnics.

§5.13 Materials and Equations-of-State

1. For the ACTIVE module of ***MAT_295** (***MAT_ANISOTROPIC_HYPERELASTIC**, included Martins et al. 2006 model for skeletal tissues for both explicit and implicit (ACTYPE = 5).
2. Implemented new material model ***MAT_COHESIVE_GASKET** (***MAT_326**) for analyzing gaskets. It is used with cohesive elements. User defined loading/unloading determine the out-of-plane behavior, while the in-plane behavior is linear elastic.
3. Added material model ***MAT_263** / ***MAT_LOU-YOON_ANISOTROPIC_PLASTICITY**. This material is based on the anisotropic yield function proposed by Lou and Yoon, 2017. It includes an anisotropic form of the Drucker function. This material is more accurate in representing material anisotropy than the other general anisotropic models. It can be accurate when predicting earing during cup-drawing simulations. It is faster than ***MAT_133**. This material is supported for shell and solid elements.
4. New thermal material ***MAT_T17** / ***MAT_THERMAL_CHEMICAL_REACTION_ORTHOTROPIC**:
 - Material incorporates the chemical reaction kinetics from ***MAT_T06** and relies on the same input data.
 - Orthotropic thermal behavior. Three load curves for the thermal conductivity values in the different directions are expected in the input for the filler and for each species.
5. Added material ***MAT_249_CRASH**. It is an anisotropic material model for crashworthiness analyses of reinforced composite materials. The overall stress is found by adding the fiber and matrix stresses. The matrix is modeled with a relatively simple orthotropic elastic plastic formulation. The reinforcement is modeled with a hyperelastic constitutive model with up to three distinct fiber directions. It uses a phenomenological description of damage and failure based on tabular input data. Driving forces for damage and failure are fiber elongation and reorientation.
6. Added materials ***MAT_ACOUSTIC_COMPLEX** (***MAT_090_COMPLEX**), ***MAT_ACOUSTIC_DAMP** (***MAT_090_DAMP**), and ***MAT_ACOUSTIC_POROUS_DB** (***MAT_090_POROUS**) for modeling acoustic fluids. See the materials user's manual for details.

§5.14 MPP

1. Added keyword ***CONTROL_MPP_REBALANCE** to control features associated with dynamic load balancing.

§5.15 Multiscale

1. New keywords ***INCLUDE_MULTISCALE** and ***DEFINE_MULTISCALE** to generalize ***INCLUDE_MULTISCALE_SPOTWELD** and ***DEFINE_MULTISCALE_SPOTWELD** for multiscale modeling applications. For instance, these keywords may be used to couple meso-scale solder joint models with a macro-scale printed-circuit-board for IC reliability analysis.
2. Added Representative Volume Element (RVE)-based micromechanical analysis for composite materials invoked with ***RVE_ANALYSIS_FEM**. It is intended for predicting nonlinear macroscopic constitutive behaviors based on 2D or 3D material microstructure. It works in the double precision versions for both SMP and MPP.
3. Added ***DATABASE_RVE** to output RVE homogenization results to the rveout file
4. Added new keyword ***INCLUDE_COSIM** for two-scale co-simulation. The two-scale co-simulation method allows for running two separate LS-DYNA MPP jobs simultaneously: one for large-scale and the other for meso-scale. Two jobs run at different time step sizes and synchronize automatically at every large scale time step.

§5.16 Output

1. Output of ***ALE_STRUCTURED_FSI** now can be controlled with the following DATABASE keywords: ***DATABASE_FSI**, ***DATABASE_FSI_SENSOR**, ***DATABASE_BINARY_FSIFOR**, ***DATABASE_SLEOUT**
2. Implemented ***DATABASE_D3MAX** to output maximum stresses and strains during a transient analysis. These results may be output to d3part if OUTPUT = 2. A Butterworth low-pass filter may also be applied to the results (IFILT = 1).

§5.17 Peridynamics

1. Added peridynamics laminate model to capture in-plane and delamination failure modes of Unidirectional Fiber Reinforced Polymer (UD-FRP) laminates in SMP and MPP. With this model you specify each lamina as a separate part and the laminate as a part set. ***SET_PERI_LAMINATE** assembles the laminate and specifies the fiber angles for each lamina. Peridynamic parts require `ELFORM = 48` on ***SECTION_SOLID_PERI**. The laminate must be meshed as 4-node surface elements. It is discretized by layers of these elements along the thickness direction. To represent material failure, adjacent elements cannot share nodes, meaning the total number of nodes is 4 times the total number of peridynamic laminate elements. ***MAT_ELASTIC_SOLID_PERI** specifies the transverse isotropic material properties for one lamina and failure parameters for the laminate.

§5.18 SPG (Smooth Particle Galerkin)

1. Added MC-SPG invoked with `ITB = 3` on ***SECTION_SOLID_SPG**. MC-SPG uses a momentum consistent smoothing algorithm to stabilize SPG. Input field `NMC` for this keyword sets the damping coefficient for MC-SPG. Four spline functions specified with `ISPLINE` are available for MC-SPG: box-type cubic spline function (`ISPLINE = 4`), box-type WF3 function (`ISPLINE = 5`), elliptic-type spline function (`ISPLINE = 14`), and elliptic-type WF3 function (`ISPLINE = 15`).

We also added two damage models specifically for MC-SPG. Setting `IDAM = 11` specifies a pre-damage model for controlling the crack growth speed. `ALPHA_B` is used to control the crack growth speed. Alternatively, bond-based damage is available by setting `IDAM = 13`. In this case, shear strain and stretch are used to calculate the damage accumulation. The damage is then applied to the flow stress. This type of damage is useful for simulating shear bands in cutting simulations.

2. Two-stage SPG analysis, particularly for strength analysis of mechanical joints after joining process analysis, such as flow drill screw (FDS), self-piercing riveting (SPR), wood/concrete screwing, etc. This analysis requires two new keywords ***INTERFACE_SPG_1** and ***INTERFACE_SPG_2**.
3. Support small restart for both SMP and MPP SPG calculations.

4. Developed a fully implicit ISPG formulation invoked with ***SECTION_FPD**. ISPG is a new generation incompressible Navier-Stokes CFD solver for the accurate and efficient simulation of solder reflowing in the electronics packaging process. This method can simulate complex solder reflow scenarios including solder pad, mask, copper trace, etc. It accurately and efficiently handles surface tension and wall adhesion. It can be coupled to the thermal and structure (implicit and explicit) solvers for both SMP and MPP. Applications of this coupling include large scale thermal-mechanical PCB and solder reflow analysis.

§5.19 SPH

1. Added optional calculation of drag force in incompressible SPH with ***DEFINE_SPH_AMBIENT_DRAG**.
2. Added keyword ***DEFINE_SPH_MASSFLOW_PLANE** to track mass of SPH particles going through a set of elements.
3. Added keyword ***DEFINE_SPH_VICINITY_SENSOR** to track mass of SPH particles coming in the vicinity of set of elements. Added capability to selectively trace them in LS-PrePost.
4. Added SPH Interface file for visualization of wetness results. This file is activated with `isph = "filename"` on the command line. The output interval is controlled with ***DATABASE_BINARY_ISPHFOR**.

§5.20 Thermal Solver

1. Added keyword ***BOUNDARY_TEMPERATURE_PERIODIC_SET** to specify rotational periodic, reflective periodic, and sliding periodic boundary conditions for the thermal solver.

§5.21 XFEM (eXtended Finite Element Method)

1. Implemented crack propagation in junction structures for both SMP and MPP solvers (***SECTION_SHELL_XFEM** with `ELFORM = 54`).
2. ***MAT_240** can now be used to model the XFEM cohesive zone for ductile fracture.
3. Added stress triaxiality based failure strain to ductile XFEM for shells (see `FAILCR = -4` on ***SECTION_SHELL_XFEM**).

4. Added directional center of effective plastic strain for crack propagation option (PROPCR = 3 in ***SECTION_SHELL_XFEM**).

§6 Improvements and Enhancements

§6.1 Acoustics, NVH, and Frequency Domain

1. The following are improvements for ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**:
 - For METHOD = 2 (Variational Indirect BEM), matrix assembly is now multithreaded, leading to considerable performance improvements in SMP.
 - Enabled Kirchhoff method and Rayleigh method for panel contribution analysis.
 - The **PANEL_CONTRIBUTION** keyword option can be used with the MATV keyword option. Mismatching BEM/FEM mesh is supported.
2. Enabled ***BOUNDARY_ACOUSTIC_MAPPING** to map the user provided velocity history file (RESTRT = 3 on ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**) to the BEM mesh. The mesh in the history file may be different from the BEM mesh. This also works in MPP.
3. Implemented d3acc output (***DATABASE_FREQUENCY_BINARY_D3ACC**) for MPP.
4. Added input fields HALF1 and HALF2 to ***FREQUENCY_DOMAIN_ACOUSTIC_FRINGE_PLOT_SPHERE** to create complete, half or quarter sphere mesh.
5. Updated ***BOUNDARY_ACOUSTIC_MAPPING** so that when a BEM node overlaps with a structural node, no interpolation is needed and the nodal velocity is just copied for the BEM acoustic computation.
6. For ***FREQUENCY_DOMAIN_ACOUSTIC_FEM**,
 - Enabled ***FREQUENCY_DOMAIN_PATH** for modal acoustics.
 - Enabled mode selection by ***FREQUENCY_DOMAIN_MODE** for modal acoustics.
7. For ***FREQUENCY_DOMAIN_RESPONSE_SPECTRUM**,

- Implemented an option to run response spectrum analysis with user defined mode combination (MCOMB = 99).
 - Implemented ABS method (absolute value method) to response spectrum analysis (MCOMB = 8).
 - Implemented an option to use user defined coefficients in 100-40-40 rule for multidirectional load spectrum cases (MPRS = 2).
 - Implemented 100-40-40 rule for multidirectional response combination (MPRS = 1).
 - Implemented adding static response results to d3spcm (BINARY = 3 in ***DATABASE_FREQUENCY_BINARY_D3SPCM**).
8. For ***FREQUENCY_DOMAIN_SSD**,
- Implemented keyword option SUBCASE to include multiple load cases in one run. Also added keyword option SUBCASE to the various databases (D3SSD, NODOUT_SSD and ELOUT_SSD) so the output frequencies for each loading case can be different.
 - Implemented the output of nodout_ssd for direct SSD (***FREQUENCY_DOMAIN_SSD_DIRECT**)

§6.2 Adaptivity

1. Added support for the file INTFOR during in-core adaptivity
2. For ***USER_INTERFACE_FRICTION** support *h*-adaptivity (MPP), meaning if ***CONTROL_ADAPTIVE** with ADPTYP = 1 or 2 is used, then history variables uhnew of subroutine usrfrc are transferred correctly from the old to new mesh in each adaptive refinement step.
3. The following are improvements for 3D *r*-adaptivity (ADPTYP = 7 in ***CONTROL_ADAPTIVE**)
 - For local refinement, LS-DYNA smooths out the mesh size transition from the refined area to the coarse area.
 - 3D TET adaptivity (ADPOPT = 2 in ***PART**) now only accepts TET initial meshes
 - When the auto remesher fails, you can now perform manual remeshing.
4. 3D tetrahedral adaptivity now supports defining moving adaptive boxes (***DEFINE_BOX_ADAPTIVE**).

5. Load curve, LCADP, used by 3D r -adaptivity is now output to d3hsp if it is nonzero.
6. Some improvements to the curvature calculation for ADPTYP = 2 on ***CONTROL_ADAPTIVE**. The mesh refinement happens closer to the curved region.
Improvement to adaptivity algorithm associated with ***CONTROL_ADAPTIVE_CURVE**. More elements than necessary were being refined.
7. Six-node solid elements are allowed to be refined during adaptivity for sandwich parts with multiple layers.

§6.3 Airbag Cards

1. ***AIRBAG_PARTICLE**

- Avoid asynchronous particle to fabric contact collision sorting for parallel processing which may select collision pairs in a different order and produce various results.
 - Solid parts can be included in the CPM airbag part list
 - Added to the CPM interface forces file the ratio of the impact from the initial air inside the bag to the impact from the inflator particles, p_{air} . This output allows you to visualize the initial air effect for IAIR = 2 and 4.
 - Added keyword option JET to ***AIRBAG_PARTICLE**. This option calculates the thrust force for an external vent and gives a better reaction force on the structure
 - Store maximum temperature for nonlinear C_p curve to keep the value monotonic for the energy calculation.
2. Extended ***DEFINE_CPM_VENT** so that the part pressure of specified parts (PID1 and PID2) can be used to evaluate the flow rate through internal vents without requiring a chamber definition. It has also been enhanced to be used for external vents for both the with chamber and without chamber cases.

§6.4 ALE

1. Added support for ***ALE_STRUCTURED_MESH_TRIM** in 2D. It previously only worked in 3D.

2. For keywords ***ALE_STRUCTURED_MESH_TRIM** and ***ALE_STRUCTURED_MESH_VOLUME_FILLING**, created new algorithm for determining if a point lies inside or outside of a structure. The new logic works well for handling structures with narrow spaces between them or with narrow openings (like car doors).
3. Added field ICASE to ***ALE_STRUCTURED_MESH_CONTROL_POINTS** to optionally select from two special methods of automatically setting up the mesh spacing.
4. Reimplemented ***CONTROL_EXPLOSIVE_SHADOW_SET** for S-ALE to achieve better accuracy and much shorter run time.
5. Added ***ALE_STRUCTURED_MULTI-MATERIAL_GROUP** for a more user-friendly way to set up ALE fluids in a S-ALE model. 3D and 2D are both supported.
6. Enhanced vacuum treatment in S-ALE for better stability. This will cause a small discrepancy in the results, but this version is more robust and has a more stable time step size.
7. Added support for S-ALE with MPP pre-decomposition.
8. For ***ALE_STRUCTURED_VOLUME_FILLING**, if specifying how to fill the mesh with GEOM = PART or PARTSET, the part/part set of the Lagrangian structure can now be meshed with thick shell or solid elements (shell in 2D). Previously only shell element parts were supported. All the volume occupied by these parts will be filled with the assigned fluid.
9. ***EOS_MURNAGHAN** supported for ***ALE_AMBIENT_HYDROSTATIC** and ***INITIAL_HYDROSTATIC_ALE**.
10. ***ALE_STRUCTURED_FSI** works in 2D. It previously only worked in 3D.
11. Algorithm enhancements for S-ALE Fluid-Structure Interaction (FSI) with ***ALE_STRUCTURED_FSI** to ensure a more stable FSI with less leakage, especially for explosion simulations. These improvements include
 - better leakage detection and removal;
 - constraining ALE nodes immersed inside LAG solid/thick shells;

- enhancements in determining the spring stretch more accurately based on different cases (explosion/sloshing/airbags);
 - better estimation of the fractional penalty stiffness; and
 - better choice of penalty spring direction.
12. For ***ALE_STRUCTURED_FSI**, added variable friction coefficient as a function of contact pressure and relative velocity (see field FRIC).
 13. ***DATABASE_FSI** supports structure segments from thick shell parts.
 14. For ***BOUNDARY_AMBIENT** AMBTYP = 5 (***LOAD_BLAST_ENHANCED** receptor), we've made the following enhancements:
 - Support multiple detonation sources
 - Obtain element density from ***LOAD_BLAST_ENHANCED** for better accuracy
 - Obtain nodal velocity from the area-weighted average of the segment velocity squared instead of segment velocity. This better conserves kinetic energy.
 15. Improved the stability of ***EOS_TABULATED_COMPACTION** when used with ALE/S-ALE.
 16. Enhanced ***BOUNDARY_NON_REFLECTING** when used with ALE/S-ALE to prevent artificial influx from outside.
 17. Extended the feature invoked with CNTTYP = 1 of ***INITIAL_VOLUME_FRACTION_GEOMETRY** to fill ALE elements with the material of a solid part in 3D (or shell part in 2D). The boundaries of the part specify the geometry of the container to be filled. The material in the part can optionally fill this container. This feature is useful for setting up an ALE model with a complex dummy structure that outlines a volume of ALE elements to fill.
 18. Added field TBEG to ***INITIAL_ALE_MAPPING** to shift the initial time of the current run from the map file. The current run usually starts at $t = 0$.
 19. LS-DYNA creates the mapping file read by ***INITIAL_ALE_MAPPING** during the last cycle. If a sensor switch defined by ***TERMINATOR_SENSOR** reaches a critical condition, the run is about to be terminated. Under these conditions, LS-DYNA will output the mapping file as well.

20. ***ALE_MESH_INTERFACE** now generates a mapping file that can be used in a 2nd run to initialize a Lagrangian (not ALE) tetrahedral model. ***INITIAL_LAG_MAPPING** reads this file and maps data from the last cycle of the ALE run to the Lagrangian tetrahedra.
21. Implemented a multiple ideal gases version of the flux corrected transport invoked with METH = 6 in ***CONTROL_ALE** for Structured ALE.
22. Reimplemented CTYPE = 3 in ***CONSTRAINED_LAGRANGE_IN_SOLID** to constrain ALE materials and structures to move together in the structure normal direction. This version, unlike the previous version, is supported in MPP.
23. ***DEFINE_CURVE_FUNCTION** can be used instead of ***DEFINE_CURVE** to compute the mesh velocities in ***ALE_REFERENCE_SYSTEM_CURVE**.
24. For ELFORM = -8 on ***SECTION_ALE1D**, single point constraints are automatically applied to the nodes on the beam mesh ends if EBC > 0 on ***CONTROL_ALE**. These constraints are the same as applying ***BOUNDARY_SPC_NODE** to the ends.

§6.5 ***BATTERY (Electrochemistry Solver)**

For the current updates, the 1D electrochemistry (EC) and its coupled thermal and mechanical (ECTM) solvers can be summarized as follows:

1. 1D EC can run up to almost 0
2. The solver can run without any limitation of storage (memory).
3. The convergence ability of the solver is improved by applying a higher-order method in the time marching scheme.
4. The material properties of the anode, cathode, and electrolyte are updated for practical applications.
5. The MPP capability for the ECTM solver is improved, and some instability issues are resolved.
6. Users can directly print out an input file for the source term of the LS-DYNA thermal solver by providing an output file name to the ***BATTERY_ECHEM_THERMAL** keyword card. The data then written by

the 1D EC solver to this ascii file has the format of time in the first column, and heat flux in the second column. The thermal solver can then simulate the battery thermal effects to test if the thermal solver is working correctly when the heat flux from the battery is input to it. Please refer to the thermal solver manual pages for a detailed description of the source term input.

The keywords influenced by the current updates and improvements are:

- ***BATTERY_ECHEM_CONTROL_SOLVER**
- ***BATTERY_ECHEM_MAT_ANODE**
- ***BATTERY_ECHEM_MAT_CATHODE**
- ***BATTERY_ECHEM_MAT_ELECTROLYTE**
- ***BATTERY_ECHEM_MAT_THERMAL**

§6.6 Blast

1. Output mole per particle for HE and resident air to `abstat_pbm` so the user can adjust the particle ratio

§6.7 Boundary Cards

1. Enabled ***BOUNDARY_CYCLIC** to set the electric voltage of node set NSID1 equal to the voltage of node set NSID2 for piezoelectric materials.
2. Enabled segments of thick shells to be subject to a non-reflecting boundary condition with ***BOUNDARY_NON_REFLECTING**.
3. Added support for thermal thick shells with ***BOUNDARY_TEMPERATURE_TRAJECTORY**.
4. Simplified the boundary condition redundancy check to improve the performance of ***BOUNDARY_TEMPERATURE_TRAJECTORY**.
5. Implemented the parameters AD and AS from ***BOUNDARY_NON_REFLECTING** in ***BOUNDARY_NON_REFLECTING_2D** to activate dilatational and shear waves, respectively.
6. Improvement for ***BOUNDARY_SPC_SYMMETRY_PLANE_SET**: The default for TOL is 0.2 for shell elements, so for these elements the tolerance does not need to be specified.

7. Extended ***BOUNDARY_SPC_SYMMETRY_PLANE** to support thick shells.

§6.8 CESE Compressible Fluid Solver

1. Removed the CFL < 1.0 limit. Note that there is no such limit in the dual CESE solvers.
2. Turned off any ***CESE_DATABASE_...** output when running CESE cavitation problems, and warn the user about this limitation.
3. Allow cyclic boundary condition segment sets to contain more than 9999 segments.
4. For the case where a problem attempts to use immersed boundary FSI with the non-inertial rotating reference frame solver, a fatal error is emitted informing the user that this is not supported.

§6.9 *CHEMISTRY

1. Add code to properly deal with missing transport data in chemistry models when using ***CHEMISTRY** with ***CESE**.

§6.10 Constraint Cards

1. Added the ID keyword option to ***CONSTRAINED_LOCAL**.
2. Support constraining the electric voltage of piezoelectric materials (DOF = 7) with ***CONSTRAINED_LINEAR_GLOBAL**.
3. Allow ***CONSTRAINED_NODE_SET** to constrain the electric potential of the piezoelectric material.
4. Rigid body stoppers invoked with ***CONSTRAINED_RIGID_BODY_STOPPERS** is supported in implicit by way of Lagrangian Multipliers and inequality constraints.
5. Added "state variable" of ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) to binout/swforc output.
6. Improved two rarely occurring issues with ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3):
 - More accurate search of associated shell element nodes, helpful in cases where shell elements are bent around corners.

- Better solution for rotation vector if both normal vectors are colinear.
7. ***CONSTRAINED_TIED_NODES_FAILURE** now supports damage values from ***MAT_107** and ***MAT_110** to be used as the failure variable for EPPF.
 8. PENALTY option of ***CONSTRAINED_BEAM_IN_SOLID** now supports 3D *r*-adaptivity (ADPTYP = 7 on ***CONTROL_ADAPTIVE**).
 9. Added deactivating ***CONSTRAINED_BEAM/SHELL_IN_SOLID** during dynamic relaxation.
 10. Specify whether ***CONSTRAINED_BEAM/SHELL_IN_SOLID** are the PENALTY versions in the d3hsp file.
 11. Added DBFLG to apply displacement boundary conditions on the patch for ***CONSTRAINED_NODE_TO_NURBS_PATCH**. When DBFLG = 1, the nodes specified in NSID will be connected to the patch by discrete beams. These nodes can be used to impose displacement boundary conditions on the patch.

§6.11 Contact

1. Add option TETFAC to optional Card F on ***CONTACT_OPTION** to scale the computed volume of tet elements for the mass calculation in SOFT = 2 contact. Better agreement is achieved with TETFAC between 3 and 5 when contact forces are compared between hex and tet meshes. This input field is supported for 4-node and 10-node tets.
2. Enabled the SWRADF factor on ***CONTROL_CONTACT** to work with solid elements and solid element assembly spot welds.
3. Added a tapered segment option (SWTPR on optional Card F of ***CONTACT_OPTION**) for segment-to-segment (SOFT = 2) contact. This option uses tapered segments around spot welds when SPOTHIN on ***CONTROL_CONTACT** is used to thin segment at spot welds. The segment thickness tapers from full thickness to the reduced thickness at nodes shared with thinned segments.
4. Enabled the offset option with CODE = 4 of ***DEFORMABLE_TO_RIGID_AUTOMATIC** to work with segment-to-segment (SOFT = 2) contact.

5. Enabled PSTIFF on ***CONTACT** to work with a contact birth time. Also, modified how friction parameters are used in segment-to-segment (SOFT = 2) contact when ***PART_CONTACT** is used. If a segment pair in contact has nonzero friction parameters from only one of the segments, the nonzero friction parameters will be used for the pair. If friction parameters are defined for both parts, then the average is used. Previously, the average was used without checking values, so zero values could be used in the average causing the coulomb coefficient to be half of the expected value.
6. Added support to segment-to-segment (SOFT = 2) contact friction for the viscous friction parameters VC on ***DEFINE_FRICTION**.
7. Improved ***DEFINE_FRICTION_SCALING** in segment-to-segment (SOFT = 2) contact so that it averages the friction coefficients only if nonzero scale factors are defined for both segments of a pair. Previously, a scale factor of 1.0 was assumed for undefined scale factors, and an average was done including that 1.0. Now only the defined scale factor is used.
8. Enabled the ***DEFINE_FRICTION_SCALING** option to be used with orthotropic friction options when using segment-to-segment (SOFT = 2) contact.
9. Enabled FRCFRQ on Optional Card A of ***CONTACT_OPTION** to work with MPP contact when SOFT = 0 or 1.
10. Enabled the use of ***LOAD_SEGMENT_CONTACT_MASK** with segment-to-segment (SOFT = 2) contact. This works with any supported contact keyword in all versions, SMP, MPP and HYBRID.
11. Enabled ***ELEMENT_SHELL_SOURCE_SINK** when used with segment-to-segment (SOFT = 2) contact.
12. Added an option for orthotropic friction when using segment-to-segment (SOFT = 2) contact. If friction properties on either side are set to zero or left blank, then properties of the other side will be used, and the material direction will be calculated from those segments.
13. Added input parameter ENDTOL to ***CONTACT_GUIDED_CABLE** to dictate how far beyond the end of a cable element a sliding node must pass before it starts contacting the next cable element. The default, which is 0.002 times the cable element length, normally works

fine. But if one or more of the cable elements is especially long, the transition can cause noise. In that case specifying the tolerance manually using ENDTOL may be desirable.

14. The following are now supported for 3D Mortar contact (***CONTACT_..._MORTAR**):

- Orthotropic friction, governed by four static friction coefficients that models sliding interaction between fiber/fiber, fiber/cross-fiber, crossfiber/fiber and crossfiber/crossfiber directions. Manual provides full details.
- A linear relationship between contact pressure and penetration depth, as opposed to the parabolic relation which is default. The new option is activated by specifying IMGAP < 0, and can presumably be used to reduce penetrations for small to moderate contact pressures.
- Redecomposition and full deck restarts.
- User defined tiebreak and tied weld for modeling lamination/delamination.
- LCFST for a friction coefficient that depends on the temperature.

15. The following are now supported for 2D Mortar contact (***CONTACT_2D_..._MORTAR**):

- MPP
- User defined friction
- Frictional energy calculations
- Interference contact using TDPEN, which is similar to MPAR1 and IGNORE = 3 for 3D Mortar contact

16. Added ***CONTACT_DRAWBEAD_BENDING**. This keyword option invokes a new contact algorithm for drawbeads that includes a weakening effect on the blank as it travels through the beads. It also stretches the blank as it goes through the beads.

§6.12 Control Cards

1. For ***CONTROL_PZELECTRIC**:

- Added the field NDTRFK to form a new dielectric matrix for every NDTRFK steps.

- Allow optional output of averaged electric flux and electric field with input field EPZMSG.
2. Allow time step factor TSSFAC on ***CONTROL_TIMESTEP** to be a time-dependent curve.
 3. For ***CONTROL_STAGED_CONSTRUCTION** and ***DEFINE_STAGED_CONSTRUCTION_PART**, added staged construction for thick shell ELFORM = 3 and 5. Previously, the staged construction keywords had no effect on these elements. Also, the manual did not state that ***DEFINE_STAGED_CONSTRUCTION** was applicable to thick shells; this has been corrected.
 4. Pore fluid capability ***CONTROL_PORE_FLUID** added for thick shell elements.
 5. Add new option LISPSID on ***CONTROL_SHELL** to define a part set for which the stresses from ***INITIAL_STRESS_SHELL** are with respect to the local coordinate system.
 6. Add new keyword ***CONTROL_CONSTRAINED** for setting constraint related properties. Currently it only sets the behavior of the initialization check for SPR2/SPR3.
 7. Added UMCHK on ***CONTROL_MAT** to potentially issue an error if only unmodified user material subroutines are called.
 8. Added 5 more materials supported for the ISTUPD = 4 option of ***CONTROL_SHELL: *MAT_120, *MAT_187, *MAT_224, *MAT_225, and *MAT_251**.
 9. Added new detailed materials energy option MATEN on ***CONTROL_ENERGY**. Currently this feature is supported material models 3, 4, 15, 19, 24, 63, 81, 82, 98, 104, 105, 106, 107, 123, 124, 188, 224, 225, 240, and 251 for shell and solid elements. Setting MATEN = 2 causes the detailed material energies to be computed. For this calculation, the internal energy is split into the elastic, plastic, and damage portions. They are computed and reported as "mat_energy_elastic", "mat_energy_plastic", and "mat_energy_damage" in glstat and matsum.
 10. Support ICOH = 1 of ***CONTROL_SOLID** for ***MAT_169**. This is automatic deletion if neighboring elements fail. It is already available for cohesive elements but is now also available for this material.

§6.13 Discrete Element Method

1. Allow user control over the number of neighboring DES to search for across processors by specifying MAXNEI on ***CONTROL_DISCRETE_ELEMENT** (default = 20)
2. Added new drag coefficient for external ballistic drag force calculation for shotgun pallet analysis (CD = -2 on ***DEFINE_DE_DRAG_FLOW**).
3. Added the keyword option TRANSDUCER to ***DEFINE_DE_TO_SURFACE_COUPLING** to collect coupling forces from different active coupling interfaces.
4. Support moving DES active region (see ***DEFINE_DE_ACTIVE_REGION**).
5. Automatically generate DES element between start and end nodes specified with NID and NID2 on ***ELEMENT_DISCRETE_SPHERE**
6. Added injecting particles with multiple predefined radii by specifying IMULTI on ***DEFINE_DE_INJECTION**. The composition of the particles (radius and mass percentage) can be specified by setting IMULTI > 1.
7. Added optional input fields LCVX, LCVY, and LCVZ to allow the particle injection velocity to vary through time instead of being constant.
8. Added flag ISKIP to ***CONTROL_DISCRETE_ELEMENT** to skip the particle to particle interaction computation.
9. Added input field MAXGAP to ***DEFINE_DE_TO_SURFACE_TIED** to determine the maximum gap between the DES and master surface.
10. For ***DEFINE_DE_INJECT_BONDED**, optimized the DES bond injection model to increase the injection mass capacity with the same injection plane.
11. Added passing more wear history variables, including contact surface area and accumulated surface wear, to the user defined wear interface.

§6.14 Dual CESE Compressible Fluid Solver

1. Added an afterburn energy field to the ideal gas EOS that is used only in the new hybrid model multiphase solver ***DUALCESE_EOS_IDEAL_GAS**.
2. For dual CESE moving mesh FSI problems, added a second argument to the ***DUALCESE_BOUNDARY_FSI** card that is the reference pressure. This permits one-sided FSI calculations.
3. Also for dual CESE moving mesh FSI problems, using a displacement threshold, allow the expensive mesh morphing calculation to be avoided when the FSI displacements are all below the specified threshold (***DUALCESE_CONTROL_MESH_MOV**).
4. Added support to several initialization and prescribed boundary cards to allow them to use the ***DEFINE_FUNCTION** capability. For the initialization cards, the function used can only be a function of x , y , and z (position). For the boundary conditions, these functions can involve the following variables defined at the point of evaluation: x , y , z (position), v_x , v_y , v_z (fluid velocity), density, temp (temperature), pres (pressure), time, and dt (dual CESE time step). This functionality applies to:

***DUALCESE_INITIAL**

***DUALCESE_INITIAL_SET**

***DUALCESE_INITIAL_HYBRID**

***DUALCESE_INITIAL_HYBRID_SET**

***DUALCESE_INITIAL_TWO-PHASE**

***DUALCESE_INITIAL_TWO-PHASE_SET**

***DUALCESE_BOUNDARY_PRESCRIBED**

***DUALCESE_BOUNDARY_PRESCRIBED_HYBRID**

***DUALCESE_BOUNDARY_PRESCRIBED_TWO-PHASE**

***DUALCESE_BOUNDARY_PRESCRIBED_VN**

5. For code speed purposes, improved the treatment of wedge/prism elements.
6. Many corrections to d3plot output have been made, along with corresponding enhancements to LSPP4.9. This applies mostly to the

moving mesh FSI and multiphase solvers. Also, the dual CESE keyword input decks can now be read into LSPP4.9 and modified after that in LSPP4.9.

7. Instead of leading to a code failure, unused mesh nodes in the dual CESE mesh are now ignored. Also, the user is now informed about elements defined in the dual CESE mesh that are invalid, and the simulation run is terminated if any are found.
8. Corrected the overallocation of some state arrays and otherwise reduced the memory footprint of the dual CESE solvers. This includes re-reading (from an internal file) a large array each time it is needed for d3plot output with the moving mesh solvers.

§6.15 Elements

1. Updated ***ELEMENT_BEAM_SOURCE** to allow resultant beam elements (***SECTION_BEAM**, ELFORM=2) with bending resistance to be pulled out of a beam source. This feature is supported for ***MAT_ELASTIC** and ***MAT_RESULTANT_PLASTICITY**.
2. Improve MPP decomposition (and therefore efficiency) for ***ELEMENT_BEAM_SOURCE**. All beam elements inside the source still go on one processor, but the others that are initially outside are equally distributed now.
3. Added optional pull-in force limiter (LMTFRC) to ***ELEMENT_SEATBELT_PRETENSIONER** for type 4, 6 and 7 pretensioners.
4. For ***ELEMENT_SEATBELT_SLIPRING** added the output of slipring friction coefficient, normal force, swap angle, skew angle and attached belt force to SBTOUT.
5. For ELFORM = -29 and 29 on ***SECTION_SHELL**, increased mass scaling factor for stability.
6. Enabled the ORIENTATION option on ***ELEMENT_BEAM** to be used with form 9 spot weld beam. It was not available before.
7. Enabled thick shell form 3 to work with material models that use an equation of state.
8. For ***ELEMENT_DIRECT_MATRIX_INPUT**:

- Enhanced the reading of superelement input files to support connections using nodes, scalar nodes, and internal dofs. Also accept dof labels of "0" for scalar nodes.
 - Modified MPP decomposition logic for collecting nodes when reading binary formatted files for superelements.
9. Add the possibility to define 3-Point Simpson integration rule through the thickness per layer of a composite shell defined via ***PART_COMPOSITE**. This can be defined via an irregular optional Card 2 (OPTCARD).
 10. COHOFF on ***SECTION_SOLID** can be used to place the relative location of cohesive interface between shells. It is only relevant for ELFORM = 20/22.
 11. Added element quality check for solid cohesive elements (***SECTION_SOLID**, ELFORM=19/20) to detect inverted elements in the first cycle.
 12. Added keyword option MISC for ***SECTION_SHELL** with first new parameter THKSCL. This allows the scaling of shell element thickness for all elements in the corresponding section at once.

§6.16 EFG (Element Free Galerkin)

1. Pressure smoothing (IPS = 1 on ***SECTION_SOLID_EFG**) now supports usermat when using EFG solid formulation 42.

§6.17 EM (Electromagnetic Solver)

1. Added more strain related options for ***DEFINE_FUNCTION** in ***EM_RANDLES_SHORT**.
2. Added option to ***EM_MAT_002** to allow EOS to be a function of temperature for the permeability
3. Added input field NPERIO to ***EM_CONTROL** so you can choose the period for calculating the average Joule heat rate for the induced heating solver.

§6.18 Fatigue Cards

1. For ***FATIGUE_{OPTION}**, extended strain based fatigue (STRSN = 1) to MPP (solid elements).
2. Implemented ***FATIGUE_D3PLOT** for MPP.

§6.19 Forming Analysis

1. Dynain file is output when using ***INTERFACE_SPRINGBACK_SEAMLESS** upon termination.
2. Added field DEPTH to ***DEFINE_CURVE_TRIM_2D**. If the distance between the element and the curve is greater than DEPTH - 1, than the element is not trimmed. In this way, we can prevent the trimming through to the opposite side of the part.
3. Changes to lancing (***ELEMENT_LANCING**):
 - Lancing can happen at different times.
 - Lancing can occur in an arbitrary direction. Previously lancing could only occur in the z-direction.
 - Lancing can have depth.
4. Added input field I2DYNAIN to ***CONTROL_FORMING_MAXID**. When I2DYNAIN = 1, ***CONTROL_FORMING_MAXID** is output to a dyanin file with the new maximum node and elements IDs. This feature simplifies post-processing for multi-step forming processes by ensuring that the new element and node IDs generated in subsequent steps are bigger than those in previous steps.
5. Added keyword ***CONTROL_FORMING_HOME_GAP** to calculate the minimum gap between the upper and lower tools. If the gap is smaller than the blank thickness, then problems can occur during forming simulations.
6. Output keyword ***INITIAL_STRAIN_SHELL/SOLID** after trimming. This will not affect the simulations after trimming.
7. Improved algorithm for searching in ***INCLUDE_STAMPED_PART** which significantly sped it up.
8. Extended ***DEFINE_BOX_NODES_ADAPTIVE** so that tube adaptivity can be applied to part sets, not just parts (set PID < 0). This extensions is usefule for simulating the forming of tailor welded blanks.

9. ***CONTROL_FORMING_REMOVE_ADAPTIVE_CONSTRAINTS** is used to remove the adaptive constraint by refining the master element with triangle elements. However, the newly generated elements do not have forming history information. Per customer request, we have added the history information to the newly generated elements.
10. ***MAT_036** is supported for the one-step method (***CONTROL_FORMING_ONESTEP**). Note that the stress-strain is only allowed to be specified with a curve (HR = 3).
11. Modified trimming of sandwich parts. We assumed sandwich parts were made up of three layers: a bottom shell layer, a solid layer, and a top shell layer. Some customers cover the solid elements not only from the top and bottom but also the sides. With this improvement, the shell elements on the side will also be trimmed.
12. Added NRBST = 2 to ***CONTROL_FORMING_PROJECTION** so that the blank nodes move in the tool's normal direction when there is initial penetration. This feature was developed for contact between a guide pin and blank.
13. Added keyword option POSITIVE to ***CONTROL_FORMING_AUTOPosition_PARAMETER** which ensures that the calculated distance to reposition will always be a positive value.
14. Improvement for ***CONTROL_ADAPTIVE_CURVE**: This keyword is used to refine the elements along the trimming curve. When nodal temperature is included, the newly added nodes do have an initial temperature which can cause problems for subsequent calculations. With this fix, the newly added nodes have the average temperature of their constraining nodes.

§6.20 ICFD (Incompressible Fluid Solver)

1. Added input field AVER to ***ICFD_CONTROL_MESH** to select the automatic volume mesher version. Version 14 is the default, but version 16 is also available.
2. Added mesh movement method where the movement is controlled by solving a linear elasticity problem using a constant size which is invoked by setting MMSH = 20 on ***ICFD_CONTROL_MESH_MOV**.

This method is useful for avoiding large distortion in rotating problems that involve large discrepancies in mesh sizes (typically in cases involving boundary layer mesh).

3. Added generating level set surface with a sphere (STYPE = 3) to ***ICFD_INITIAL_LEVELSET**. Also, multiple definition of ***ICFD_INITIAL_LEVELSET** is allowed as long as the different shapes generated do not intersect.
4. Added one and two parameter Pierson-Moskowitz spectrum for irregular wave generation invoked with WTYPE = 6 and 7, respectively, on ***ICFD_BOUNDARY_FSWAVE**.
5. Made boundary conditions at free surfaces more stable and accurate. Stability has particularly increased in simulations that combine FSI with free surfaces.
6. Added input field TPS to ***ICFD_CONTROL_EMBEDSHELL** to control the fluid escape through triple points.
7. Added option OWC = 3 to ***ICFD_CONTROL_FSI** to perform two-way coupling in which the forces weakly couple (no sub-stepping) with the implicit mechanical solver
8. Added backflow stabilization for pressure boundary conditions
9. Added fields KS and CS to ***ICFD_BOUNDARY_PRESCRIBED_-TURBULENCE** to prescribe local roughness height and roughness constant
10. Added input field DTT to ***ICFD_CONTROL_TIME**, so you can set a thermal time step that is larger than the one used for the velocity/-pressure solve.
11. Added input flag DTBL to ***ICFD_CONTROL_TIME** to specify whether boundary layer elements should be included in the automatic time step calculation.

§6.21 Implicit (Mechanical) Solver

1. For ***CONTROL_IMPLICIT_SOLVER**:
 - We upgraded MUMPS (LSOLVR = 30) from version 5.2.0 to 5.3.5.

- For intermittent eigenvalue analysis the direct solvers used by the eigenvalue analysis were forced for the entire execution. We now save and restore the solver option selected by the user to use during the non-eigenvalue execution of the run. This enables the use of iterative solvers.
2. For ***CONTROL_IMPLICIT_EIGENVALUE**:
 - Performance improvements for Block Lanczos (EIGMTH = 2) in the matrix-vector multiplies in MPP.
 - Performance improvements for LOBPCG (EIGMTH = 102) in the preconditioner calculation.
 - Enabled additional keyword controls (optional Card 3) for MPP execution. These controls were initially only available for SMP.
 - Rewrote how data is output to the d3plot, d3eigv, and similar files to vastly improve performance of the output. This saves 15% of the wall clock time for large eigenvalue problems where lots and lots of eigenmodes are computed and written to d3eigv.
 3. Enhanced ***PART_MODES** to work in single precision.
 4. Some contact types were being turned off for the inertia relief computation ***CONTROL_IMPLICIT_INERTIA_RELIEF**. These contacts are now enabled.
 5. Allow shell element 18 to be used in modal dynamics ***CONTROL_IMPLICIT_MODAL_DYNAMICS** when reading the modes from d3eigv.
 6. Enhanced the implicit solver's processing of nodal mass matrices to include the local coordinate transformation.
 7. For ***CONTROL_IMPLICIT_SOLUTION**:
 - Prior to this revision implicit linear analysis (NSOLVR = 1) used the forces at the end of the linear step to compute resultant forces. This did not match expectations of users. We changed the computation of resultant forces, such as those to bndout, to use the force at the beginning of the linear step.
 - Shell types 4 and 16 are supported for the nonsymmetric solver (LCPACK = 3), for the few materials that need this feature (***MAT_SAMP_LIGHT** currently being the only one).

§6.22 Initial Cards

1. Enabled ***INITIAL_VELOCITY_GENERATION** to work for solid element forms 24, 25, 26, and 61. Previously, only the first 8 nodes had velocities initialized.
2. For ***INITIAL_STRESS_SECTION**:
 - **IZSHEAR = 2** now supports all cross section types. Previously cross sections specified with ***DATABASE_CROSS_SECTION_SET** could not be used.
 - Bolt preload can now start from a nonzero stress state.
3. Added support for initializing solids with ***INITIAL_HISTORY_NODE(_SET)**. So far ELFORMs -2, -1, 1, 2, 10, and 13 are supported.

§6.23 Isogeometric Analysis (IGA)

1. Print out the first 100 smallest time steps and the element connectivities for IGA shell and solid elements. In SMP, the time step information is output to the d3hsp file; in MPP the time step information is printed out to each message file on each processor.
2. Support consistency flag (CONST in ***CONTROL_PARALLEL**) in SMP for IGA thick shells specified with ***ELEMENT_SHELL_NURBS_PATCH**.
3. Add cross section force to new IGA shell/solid (see ***DATABASE_CROSS_SECTION**).
4. ***DEFINE_STOCHASTIC_VARIATION** is supported for isogeometric analysis with shell elements in combination with some material models that allow the STOCHASTIC option (***MAT_015**, ***MAT_024**, ***MAT_081** and ***MAT_123**).
5. ***MAT_ADD_DAMAGE_GISSMO_STOCHASTIC** is supported for isogeometric analysis with shell elements.
6. Material models ***MAT_122 (*MAT_HILL_3R)** and ***MAT_122_TABULATED (*MAT_HILL_3R_TABULATED)** are supported for isogeometric analysis with shell elements.
7. Support ***MAT_030** with IGA shells and solids.

8. Support ***MAT_SAMP-1** and ***MAT_SAMP_LIGHT** (***MAT_-187/*MAT_187L**) with IGA shells
9. Enable restart possibility for IGA elements.
10. Improved multi-patch coupling for thin (rotation-free) IGA shells with initial kinks.
11. Expand modeling capabilities of HAZ for IGA shells

§6.24 Load Cards

1. Accept thick shells in the input for ***LOAD_SEGMENT_FILE**
2. Implemented ***LOAD_SEISMIC_SSI_DECONV** for deconvolved earthquake input in soil-structure interaction and site-response analysis.
3. Added having ***LOAD_SEISMIC_SSI_AUX** read gmbin files created by ***INTERFACE_SSI_AUX** or ***INTERFACE_SSI_AUX_EMBEDDED** in a previous run. This allows multiple gmbin files to be used in the same run. Previously this was done as part of ***INTERFACE_SSI**, but that approach is now deprecated.
4. Added keyword ***LOAD_EXPANSION_PRESSURE** to apply a uniform pressure to a section of chamber that may change in size due to a moving edge. It works in both SMP and MPP.

§6.25 Materials and Equations-of-State

1. ***MAT_ADD_COHESIVE** now supports hyperelastic foam materials ***MAT_057** and ***MAT_083**.
2. For ***MAT_ADD_DAMAGE_DIEM**:
 - The FLD/MSFLD criteria now applies to solid elements, with certain mesh restrictions (consult manual for details). This feature allows for modeling membrane and bending failure relatively independent.
 - Exponential damage function for accurate post-instability behavior is available.
 - Add new parameter P5 to refer to a curve/table ID that defines
 - regularization factor vs. element size (curve) or

- regularization factor vs. element size vs. abscissa value of criterion (table)

Plastic strain at onset of damage can be scaled by that factor. This is similar to LCREGD in ***MAT_ADD_DAMAGE_GISSMO**.

3. Enabled GISSMO to be used with ***DEFINE_HAZ_PROPERTIES**. It is necessary to use ***MAT_ADD_DAMAGE_GISSMO_STOCHASTIC** to make it work.
4. Support VOLFRAC in ***MAT_ADD_EROSION** so that it works correctly with failure criteria in ***MAT_ADD_DAMAGE_DIEM/GISSMO** and ***MAT_ADD_EROSION** itself. Before it only worked with failure in materials, such as FAIL in ***MAT_024**.
5. For ***MAT_ADD_GENERALIZED_DAMAGE**:
 - Add new option IFLG3 = 2. This is a special case called "Domain of Shell-to-Solid Equivalence" or DSSE, where two damage variables are used but the second one is not really affecting the stresses.
 - Improved robustness for when DMGEXP < 1.
6. New features for ***MAT_ADD_INELASTICITY** include
 - (i) orthotropic field overlay for thermal expansion and failure
 - (ii) temperature shifts in hypo-viscoelasticity,
 - (iii) new hyper-viscoelastic laws,
 - (iv) support of higher order tetrahedra and
 - (v) norton-bailey creep.
7. For ***MAT_ADD_PZEELECTRIC**:
 - Added support for 20-node solid elements (ELFORM = 23 in ***SECTION_SOLID**) and 10-node tetrahedron elements (ELFORM = 16 in ***SECTION_SOLID**).
 - Implemented axisymmetric and plane-strain formulation.
 - Added iterative solver.
 - Added optional dielectric loss parameters.
8. Extended to anisotropic behaviour for ***MAT_ADD_SOC_EXPANSION** by specifying LCIDY, MULTY, LCIDZ and MULTZ similar to ***MAT_ADD_THERMAL_EXPANSION**.

9. Added a negative MACF option for solid element orthotropic material models which causes the axis swapping to be done prior to the element rotation. This allows the rotation to be about any axis and not just the *c*-axis.
10. Added new option RATEOP = 5 to ***MAT_015** with VP = 1: strain rate dependent term based on Couque (2014).
11. For ***MAT_024**:
 - Properly support VP = 3 in implicit analysis for solid and shell elements.
 - LCSS with VP = 3 now supports ***DEFINE_TABLE_XD/COMPACT** up to a level of 9, meaning yield stress can be a function of plastic strain, strain rate, and up to seven history variables. If ***MAT_ADD_DAMAGE_DIEM** is used together with ***MAT_024** with VP = 3, then P1 of ***MAT_ADD_DAMAGE_DIEM** also supports a table up to level 9.
12. For ***MAT_030 / *MAT_SHAPE_MEMORY**:
 - Updated beam, shell and solid element tangent stiffness to account for plasticity.
 - Added warning message for too low plastic yield stress with respect to phase transformation limits.
 - Added curve/table dependence on plastic strain/temperature for transition stress parameters.
13. Output "maximum effective strain" as history variable number 29 for ***MAT_034** (FORM = 14, -14, and 24).
14. Add ***MAT_036** as a potential weld partner material for PRUL=2/3 of ***DEFINE_CONNECTION_PROPERTIES**.
15. Support thermal option (thermal structure coupling) for ***MAT_USER_DEFINED_MATERIAL_MODELS** with SPH particles.
16. Added rate filtering option, RFILTF, to ***MAT_064** with VP = 0 for solid elements.
17. For ***MAT_068**:

- Added velocity dependent hardening. Hardening curve IDs LCPD{R,S,T} and LCPM{R,S,T} can now refer to tables, meaning velocity dependent yield forces/moments can be defined.
 - Added velocity dependent failure. Negative input of {F,M,U,T}FAIL{R,S,T} can now refer to curves, meaning velocity dependent failure forces/moments/displacements/rotations can be defined.
18. Added a new option for solid element spot weld failure. Setting TTOPT = 2 on ***MAT_SPOTWELD (*MAT_100)** causes the failure calculate to use average forces and moments on the opposite weld faces for the failure calculation. This makes the welds invariant in terms of how the weld elements are numbered. This works for single brick welds and solid weld assemblies.
 19. Added keyword option UNIAXIAL for ***MAT_100** and ***MAT_100_DA**. In addition to setting the transverse shear stresses to zero ($E < 0$), the transverse shear strain increments get zeroed as well. This can reduce occasional instabilities.
 20. For ***MAT_100_DA**, improved the material tangent in the plastic regime to yield better convergence in implicit analysis.
 21. Implemented erosion flag for ***MAT_JOHNSON_COOK/*MAT_110** for shells.
 22. For ***MAT_112 / *MAT_FINITE_ELASTIC_STRAIN_PLASTICITY**, implemented material tangent stiffness for solid elements.
 23. For ***MAT_122_3D / *MAT_HILL_3R_3D**, updated plasticity algorithm to increase implicit stability.
 24. Added non-linear viscoelastic model to ***MAT_123** ($VP = 1$) for shell and solid elements. Nonzero LCEMOD activates this model. This viscoelastic approach is adopted from ***MAT_187**.
 25. Add strain rate filtering parameter, SRFILT, to ***MAT_124**. This enables strain rate filtering using an exponential moving average. Works for LCSRC or LCSRT being nonzero and SRFLAG = 0 or 1.
 26. Added support for ***MAT_MODIFIED_HONEYCOMB / *MAT_126** with solid element types 0 and 9 to work with ***CONSTRAINED_TIED_NODES_FAILURE**.

27. Added option to specify the energy release rates for modes I and II (GIC and GIIC) in ***MAT_138** as element size dependent curves.
28. For ***MAT_DAMAGE_3** / ***MAT_153**:
 - Extended to allow for temperature dependence, including for elastic moduli, hardening parameters, and back stress parameters.
 - Enhanced accuracy for implicit simulations.
29. Improved robustness of ***MAT_154** / ***MAT_DESHPANDE_FLECK_FOAM** when using DERFI = 0 for the iteration of the equivalent stress.
30. Implemented a viscoelastic option for ***MAT_SIMPLIFIED_RUBBER/FOAM** / ***MAT_181** for solid formulation 13, tetrahedron with nodal pressure.
31. Resolved single precision issue with small BETA values in ***MAT_187** to increase accuracy of results if LCEMOD > 0.
32. Improved tangents for ***MAT_SAMP_1** and ***MAT_SAMP_LIGHT** for the implicit solver.
33. Added new parameter RATEOP to ***MAT_SAMP_LIGHT** (***MAT_187L**) to choose type of effective strain rate calculation.
34. Improvements to ***MAT_SAMP_LIGHT** to better deal with highly nonlinear inputs, such as sharp curves and logarithmic strain rate interpolation.
35. For ***MAT_199** / ***MAT_BARLAT_YLD2004**, added possibility to have Young's modulus and yield stress as functions of plastic strain and temperature.
36. For ***MAT_BOLT_BEAM**, added feature (see input parameter HOLSHR) to enable modeling of a shear deformation mode in which the bolt shank tears through the plates. After the clearance gap has been closed, further shear deformation would enlarge the hole in the plates rather than deform the bolt itself. The force-deformation relation of this mechanism is still governed by LCSHR, but the deformation (i.e. enlargement of the hole) is tracked separately in each of four principal directions. Thus, enlargement of the hole in the positive *Y* direction has no effect on the position of the edge of the hole in the negative *Y* direction.

37. Align element failure behavior in ***MAT_224_GYS** with the one in ***MAT_224**, e.g., NUMINT = -200 should prevent element erosion at all.
38. Added two options to ***MAT_224_GYS**:
 - Keyword option LOG_INTERPOLATION invokes logarithmic interpolation of strain rates in tables LCK1, LCCR, and LCSR.
 - Added BETA < 0 with abs(BETA) referring to a curve, table, table_3d, or table_4d as in ***MAT_224**.
39. Young's modulus for ***MAT_226** can be a function of plastic strain.
40. Added keyword option FUNCTIONS to ***MAT_240** to include connection partner properties. With FUNCTIONS, the following parameters can/must be defined as ***DEFINE_FUNCTION** IDs: EMOD, GMOD, G1C_0, T0, FG1, G2C_0, S0, and FG2 (and GMOD3, G3C_0, R0, and FG3 if keyword option 3MODES is also used). The arguments of these functions include thicknesses and maximum yield stresses of both weld partners (from tied contact) as well as the strain rate and element area. A similar capability is already available for ***MAT_169**.
41. Added new option for yield/damage initiation criterion of ***MAT_240**. Parameter INICRT can now be given as negative value, invoking a flexible exponent in the yield and damage initiation criteria.
42. Added new history variable number 15 to ***MAT_240**: "maximum strain value".
43. The coefficients α_i and m in ***MAT_242** can now be functions of effective plastic strain.
44. Added parameter IHIS to ***MAT_249** to allow initialization of fiber directions element-wise with ***INITIAL_STRESS_SHELL**.
45. Support ***MAT_251** in implicit analysis for solid and shell elements.
46. Add ***MAT_251** as potential weld partner material for PRUL=2/3 of ***DEFINE_CONNECTION_PROPERTIES**. Therefore use values from constant inputs PHASE1 and PHASE2 to find the associated curve in a ***DEFINE_TABLE_3D** or ***DEFINE_TABLE_4D**.
47. For ***MAT_254**:

- Added evolution equations for internal variables (grain size, dislocation density, globularization,...)
 - Accept user define functions for yield stress computations of individual phases
48. Add new option to define rate dependent strength in ***MAT_GLASS (*MAT_280)**. It is invoked by $FT < 0$, with curve ID $|FT|$ specifying tensile strength vs. strain rate.
 49. Minor change for ***MAT_280** with ENGCRIT and RADCRIT. If used together with ***MAT_ADD_EROSION** (where these criteria are also available), ***MAT_280** only "wins" if those parameters are nonzero.
 50. For ***MAT_295 / *MAT_ANISOTROPIC_HYPERELASTIC**, improved initial effective moduli computation used for both stable time step estimates as well as hourglass control.
 51. For the ACTIVE module of ***MAT_295 / *MAT_ANISOTROPIC_HYPERELASTIC**:
 - Made the default of the active stress scaling factor unity.
 - Enabled activation/deactivation for all active models used in conjunction with electrophysiology coupling.
 - Included an activation threshold on the calcium concentration for active models 1-3 for electrophysiology coupling.
 - Increased number of history variables.
 52. For ***MAT_SEATBELT**, improved the robustness of the 2D belt for $FORM \geq 2$.
 53. Consolidation and enhancements of all orthotropic thermal materials (***MAT_T02, *MAT_T04, *MAT_T08, *MAT_T11-15, *MAT_T17**):
 - New material axes definition AOPT = 3 implemented, defining material direction based on element normal and a vector d. First material direction is orthogonal to both of them.
 - Materials now all accept all the available options (AOPT = 0 through 4).
 54. For ***MAT_T08** and ***MAT_T10**, heat generation, specific heat, and thermal conductivity (TGRLC, LCC, LCK*i*) can be load curves or tables depending on mechanical history variables.

55. NOCOPY = 1 in ***CONTROL_SOLUTION** invokes a keep-in-memory treatment of material history variables, thus avoiding the overhead of copying to/from local buffers. This may result in a significant reduction in simulation time, depending on how much this gathering/scattering of data takes in relation to the rest of the computations. The best result is seen when "fast" materials, like ***MAT_024**, is used together with gissmo/diem. This option works only for a selection of materials and elements; consult manual for details. Open-MP threading (ncpu > 1 or ncpu < -1) is not supported.
56. Added new user interface to define number of history variables on per part basis. New subroutine "umat_setnhvpp" is added to dyn21umat.F and called during initialization.
57. The new implementation of ***DEFINE_CURVE_STRESS** allows you to change the number of output stress pairs based on the value from LCINT of ***CONTROL_SOLUTION**.

§6.26 MPP

1. Add MPP rebalance and in-core adaptive support for ***DEFINE_CURVE_FUNCTIONS** that reference nodal quantities
2. The following are enhancements for redecomposition (***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION**):
 - Added start time field (STIME) to MPP redecomposition.
 - Added keyword option ONCE to ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION** to allow for performing the redecomposition only once and then keep the job running with the adjusted decomposition
 - Support redecomposition problem with dynamic relaxation with polar symmetry
 - Support implicit joints in redecomposition.
 - Add new feature to ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION** (input field WEIGHT) to estimate element cost based on element in contact and stress state to achieve better load balancing.
 - Support new partition of the model based on the previous redecomposition step's element cost (SAMPT).

3. Allow a later case to run using a predecomposition file from an earlier case.
4. Restructure MPP beam weld scale factor calculation for HAZ to have better consistency with SMP and different MPP ranks

§6.27 Multiscale

1. Added support for Hughes-Liu beams (ELFORM = 1 on ***SECTION_BEAM**) in the macroscale model of the multiscale spot weld. Previously only spot weld beams (ELFORM = 9) were supported.
2. Material models that support Hughes-Liu beams are available in the macroscale model of the spot weld when that beam formulation is used.

§6.28 Output

1. Remove discrete and seat belt visualization beams from the eout file: they are NULL, have no meaningful data, and were filtered out anyway.
2. For ***DATABASE_RECOVER_NODE**:
 - Added support for one-point nodal pressure tetrahedron element (formulation 13).
 - Added support for solid formulations -1 and -2 for the elemental extrapolation method (METHOD = 1).
3. Added echoes in d3hsp for all default values set by ***CONTROL_THERMAL_FORMING**.
4. Enabled output of moments to the section force file, secforc, for 2D plane strain beam element form 7.
5. Added FILTER keyword option to ***DATABASE_SECFORC** for Butterworth filtering to smooth the forces. This is similar to the filter for ***DATABASE_NCFORC_FILTER**.
6. Added an optional card to ***DATABASE_BINARY_D3PART** to output element data for certain elements to d3part. Fields 1-4 of the optional card specify the set IDs of the solid elements, beam elements, shell elements, and thick shell elements for which the data will be

output. This option will work in addition to PSETID defined in Card 1.

7. Added forces from ***LOAD_ERODING_PART_SET** to nodfor output.
8. Output 1D spherical part masses to d3hsp when ELFORM = -8 is specified on ***SECTION_ALE1D**.
9. Added ***DATABASE_ALE_OPERATION** to output user-defined ALE values from existing ALE data that can be fringed in d3plot and/or output in a history .xy file.
10. For ***DEFINE_MATERIAL_HISTORIES**
 - Plastic Energy Density is available as an output quantity.
 - Added keyword option NAMES. This will allow you to specify user-defined names for history variables.
11. Added a new way to specify circular cross sections for ***DATABASE_CROSS_SECTION_PLANE**. When RADIUS < 0.0, the radius is the |RADIUS|, and XCT and XCH will be node IDs for specifying the location and the normal vector of the circular plane.

§6.29 Restarts

1. Support LCTM of ***CONTROL_TIMESTEP** in small restart.

§6.30 Sensors

1. For ***SENSOR_CONTROL**:
 - Added TYPE = EM to control the EM solver.
 - Add TYPE = CONSTRL to control ***CONSTRAINED_LOCAL**.
 - Added TYPE = PZBC to control the application of the piezoelectric boundary condition.
2. Added keyword option UPDATE to all versions of ***SENSOR_DEFINE_OPTION** so that you can define how often a sensor will be updated.
3. Added VID = M and MMOMENT to ***SENSOR_DEFINE_FORCE** for measuring the magnitude of force and moment, respectively.

4. Allow as many as sensors as needed for ***SENSOR_SWITCH_CALC-LOGIC**.

§6.31 SPG (Smooth Particle Galerkin)

1. MPP version is now available.
2. The maximum number of neighboring particles is increased to 2048 in SPG dynamic memory allocation.
3. To model fiber reinforced composites with the Immersed SPG method, added a special MPP decomposition checkup to make sure that the immersed beam node and the corresponding SPG solid are on the same CPU. By doing this, it will guarantee the coupling between SPG solids and FEM beams when using immersed SPG.
4. Add the input field ALPHA_S to ***SECTION_SOLID_SPG** to control the surface particles' quadrature fraction. This is used to overcome shear locking in thin structures.
5. Added support for ***MAT_005**, ***MAT_014**, ***MAT_059**, ***MAT_124**, and ***MAT_193**.
6. New input format for the data cards of ***CONSTRAINED_IMMERSED_IN_SPG** to make setup easier

§6.32 SPH

1. Reduced memory footprint and hid some of the MPI communication latency for very large SPH models.
2. Improved performance of SPH models with a large ratio of inactive particles (injection, moving box,...) by creating a sublist of active particles.
3. Added option to explicitly calculate pressure on structure particles (see ***CONTROL_SPH_INCOMPRESSIBLE**).
4. Added field ISHOW to ***DEFINE_BOX_SPH** to add a dummy shell for visualization in LSPP.
5. Added time step condition based on viscous forces for incompressible SPH.

6. Added support of the Thermal Speedup Factor (TSF) on ***CONTROL_THERMAL_SOLVER** to the explicit SPH thermal calculation.
7. Added support for multiple ***DEFINE_SPH_MESH_BOX** keyword definitions.
8. Adaptive SPH (FORM = 9 and 10 on ***CONTROL_SPH**) supported in MPP.
9. Support thermal solver for thermal-structure coupling with adaptive SPH.
10. Added SPH particle shifting stabilization algorithm. This algorithm is invoked with ISHIFT on Card 3 of ***CONTROL_SPH**. It shifts particles slightly across streamlines to reduce particle clustering in the maximum compression and stretching directions.

§6.33 Thermal Solver

1. For ***CONTROL_THERMAL_FORMING**:
 - Support setting the default thermal contact parameters for contact types ***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE**, ***CONTACT_ONE_WAY_SURFACE_TO_SURFACE**, ***CONTACT_SURFACE_TO_SURFACE** and ***CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE**. Previously only forming contacts were supported.
 - Added ability to define a default ***USER_INTERFACE_CONDUCTIVITY** for contact pairs without the THERMAL_FRICTION extension with ***CONTROL_THERMAL_FORMING** by setting FORMULA < 0.
 - Added initial temperature condition check for thermal thick shells. Default initial temperatures will be assigned to layers without initial temperatures to avoid unwanted temperature gradient at the 1st time step.
2. For ***CONTROL_THERMAL_SOLVER**, performance improvement in SMP of the GMRES solver for conjugate heat transfer (SOLVER = 17).
3. New temperature scaling option for ***BOUNDARY_TEMPERATURE_RSW** and ***LOAD_THERMAL_RSW**. If load curve ID in the input is negative, the scaling operation acts on the difference between

default and prescribed values in order to ensure a smooth temperature evolution in the weld nugget and heat affected zone.

4. For ***LOAD_THERMAL_BINOUT** enhancements for handling multiple LSDA temperature files:
 - If one lsdA file has reached the final point in time and the mechanical run continues, keep the respective temperatures constant
 - If all lsdA files have reached the end time while the mechanical run continues, write a message and keep all temperatures constant.

§6.34 Miscellaneous

1. Significantly sped up some load curve evaluations, particularly if there are many points in the curve definition, by changing the search from linear to binary.
2. Added two sense switches:
 - SWE terminates dynamic relaxation and continues LS-DYNA with the transient analysis.
 - `endtime=t` causes the termination time to be changed to time t .
3. Added ECHG and ECHGBC to ***DEFINE_CURVE_FUNCTION** to output the reactive charge of piezoelectric materials.
4. Allow character parameters specified with ***PARAMETER** to be used as part of a larger string.
5. For ***USER_NONLOCAL_SEARCH**, check same material model is used within the part set for averaged and surrounding elements.
6. For ***DEFINE_PRESSURE_TUBE**:
 - Added Discontinuous Galerkin (MTD = 1) and Discontinuous Galerkin for the Isothermal Euler Equations (MTD = 2) to the solution methods.
 - Cavities (CAV_i) can now be visualized on beam elements in `d3plot` when using automatically generated shell/solid tubes (TYPE \neq 0).
7. For ***DEFINE_QUASAR_COUPLING**:
 - Output initial state from LS-DYNA for QUASAR

- Allow user control of the frequency of data exchange between the two solvers with FRCFRQ
 - Support multiple quasar coupling interfaces
 - Added new option (IOPT) to allow user select different data exchanged between solvers
 - Cleanup and speedup quasar data collection and message passing routine
 - New feature to exclude node set from original list
 - Support local coordinates system for Quasar ROM model mapping to LS-DYNA global coordinates system.
8. Implemented ***DEFINE_DRIFT_REMOVE** for MPP.
 9. Added a new flag, IMONFLAG, to field 5 of ***DEFINE_HAZ_TAILOR_WELDED_BLANK**, to allow load curves, ISW and IFW, from Card 2 of ***DEFINE_HAZ_PROPERTIES** to either increase monotonically (IMONFLAG = 0) or arbitrarily (IMONFLAG = 1).
 10. For ***RAIL_TRAIN**:
 - Output file containing wheel/rail forces is now csv-format by default, enabling easier post-processing (see input parameter FMT)
 - New option for wheel flanges to be on the outside of the rails rather than on the inside (see input parameter LATDIR)
 - New option for friction resisting lateral sliding across the rail (see input parameter FRIC).
 11. Add keyword ***DEFINE_TABLE_COMPACT**. It allows a more compact notation of 2D-, 3D-, 4D-, ... tables. Everything takes place in the keyword reader, where a ***DEFINE_TABLE_{X}D** is created internally.
 12. Extended the ***USER_LOADING** interface to support thick shell elements.
 13. The following are enhancements for specifying the FMI co-simulation settings (***COSIM_FMI_CONTROL**):
 - Allow different time units for the solvers
 - Allow the FMU to have a time step that is smaller than LS-DYNA's time step

14. Added more helpful warning and error messages in `fmu_msg` for co-simulations.
15. Added the keyword option `DEFORM` on ***RIGIDWALL_GEOMETRIC_CYLINDER** for changing the shape of the cylinder.

§7 Bug Fixes

§7.1 Acoustics, NVH, and Frequency Domain

1. For ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**:
 - Fixed Rayleigh method (`METHOD = 0`) for the `HALF_SPACE` keyword option when using triangular elements.
 - Added the missing incident wave in the Rayleigh method for pressure output in fringe plots.
 - Fixed bug in restarting BEM using `d3dump`.
2. Fixed a bug in running FRF ***FREQUENCY_DOMAIN_FRF** with pressure load (`VAD1 = 4`).
3. For ***FREQUENCY_DOMAIN_RANDOM_VIBRATION**
 - Fixed a bug in dumping `d3psd`, `d3rms` and `d3zcf` when `neips` and `neiph` in ***DATABASE_EXTENT_BINARY** are not zero.
 - Fixed a bug in random vibration analysis with random pressure load (`VAFLAG = 2`) when the normal direction of the surface elements is not in the x , y , or z axis direction.
4. For ***FREQUENCY_DOMAIN_RESPONSE_SPECTRUM**, fixed a bug in combining ground motion with structural relative motion, when the load is defined by time history (`LCTYP = 10,11,12`).
5. Fixed a bug in SSD analysis ***FREQUENCY_DOMAIN_SSD** with pressure load (`VAD = 1`) when the normal direction of the surface elements is not in x , y , or z axis direction.

§7.2 Adaptivity

1. Fix for MPP in-core adaptivity when initial refinement is active. In some cases it was not refining elements it should have, resulting in invalid adaptive constraint situations.

2. Fix for MPP in-core adaptivity which was generating inconsistent user IDs for shared nodes in some cases, resulting in invalid DYNAIN files (and probably other unknown issues).
3. Fix bug in in-core adaptivity for forming blanks that occurs when some of the elements in the blank failed during the simulation.
4. Fixed remeshing of parts with voids for 2D *r*-adaptivity ($|ADPTYP| = 8$ on ***CONTROL_ADAPTIVE**). Previously, the voids were remeshed with the elements. With this fix, the voids remain as voids and are not remeshed with the elements.
5. Fixed bug causing error message STR+389 (solid elements referencing a non-existing part) that occurs when using ***RIGIDWALL_GEOMETRIC_FLAT_MOTION_ID_DISPLAY** with adaptivity.
6. Fixed bug in 3D *r*-adaptivity invoked with ***CONTROL_ADAPTIVE** ($ADPTYP = 7$) and ***CONTROL_REMESHING** when used with cases. The bug caused edges and corners to be lost during *r*-adaptivity after the first case.
7. Fixed bug that caused incorrect output to d3plot for 3D *r*-adaptivity ($ADPTYP = 7$ on ***CONTROL_ADAPTIVE**). It was incorrect when using interactive adaptivity $IAT = 3$ (***CONTROL_REMESHING**) in implicit analysis.
8. Fixed a bug to obtain correct resultant forces for prescribe motion constraints applied to a box region of space (***BOUNDARY_PRESCRIBED_MOTION_SET_BOX**) when 3D *r*-adaptivity ($ADPTYP = 7$ on ***CONTROL_ADAPTIVE**) is enabled. The results are incorrect when the region in the box is adapted.
9. Fixed bug in 3D *r*-adaptivity so the stresses in non-adapted parts are correct.

§7.3 Airbag Cards

1. Fix inconsistent results between runs which was caused by incorrect CV decomposition data on local processors.
2. Fix ***AIRBAG_HYBRID** structure input translation error under long = s
3. Fix mixing of reference geometry and reference shell geometry

4. The following are bug fixes for CPM airbags (***AIRBAG_PARTICLE**):
 - Fixed error in vent mass flow rate calculation after the switch from a corpuscular particle method airbag to a uniform pressure airbag. This error occurs with or without a chamber definition (CHM on Card 6).
 - Fixed issue with CPM/UP switch when multiple chambers are interacting.
 - Fixed issue associated with the initial air particle assignment for a multiple airbags model. The bug caused results to depend on the order of the airbags in the input.
 - Fixed issue in reading XM_i (Card 12) in the ***AIRBAG_PARTICLE** input that occurred when the ID associated with ***DEFINE_CPM_-GAS_PROPERTIES** is at least 9 digits long.
 - Fixed bug for multiple CPM bags that occurred when NPRLX was the number of cycles for some of the bags and the end time for the other bags. This issue occurred only in MPP.
 - Fixed species porous leakage output in `abstat_cpm`

§7.4 ALE

1. Fixed a bug that occurs when ***BOUNDARY_NON_REFLECTING_-2D** is used with S-ALE. The bug caused errors when the geometry was a ring. It also caused errors in MPP.
2. Bug fixes for ***ALE_STRUCTURED_MESH_TRIM**, ***ALE_STRUCTURED_MESH_VOLUME_FILLING**, and ***ALE_STRUCTURED_MESH_MOTION** in Windows for issues leading to crashes.
3. For ***ALE_STRUCTURED_FSI**:
 - Fixed bugs that were causing crashes or incorrect results when the structure contained triangular segments.
 - Fixed bugs that were causing crashes or instabilities (time step decreasing) due to memory issues.
4. Fixed the data output by ***ALE_STRUCTURED_MESH** to the mapping file. This file is read by ***INITIAL_ALE_MAPPING**.

§7.5 Boundary Cards

1. Fixed a bug associated with ***BOUNDARY_NON_REFLECTING** which might cause a failure in locating a segment belonging to a 10-node tetrahedron element.
2. Fixed behavior of ***BOUNDARY_NON_REFLECTING** in implicit.
3. For ***BOUNDARY_PRESCRIBED_MOTION_SET_BOX**, fixed an MPP bug that happens when the node set is part of a rigid body.
4. For ***BOUNDARY_TEMPERATURE_TRAJECTORY**:
 - Fixed bug causing incorrect error message "redundant temperature boundary conditions". It occurred when the model contains only solids and THSHEL = 1 on ***CONTROL_SHELL**.
 - Fixed the local coordinate system bug which results in an incorrect normal calculation when NSID2 < 0 and the surface normal jumps.
 - Fixed bug that caused an incorrect welding torch position when multiple welding paths exist.
5. Corrected a precision conversion problem associated with using super-elements in explicit single precision with velocity boundary conditions.

§7.6 CESE Compressible Fluid Solver

1. Corrected an MPP d3p1ot bug when outputting surface fluid or moving mesh or conjugate heat transfer boundary surface mechanical solver variables.
2. Fixed ***CESE_DATABASE_POINTOUT** for the 2D case where the user sets some z -coordinates to be nonzero. This practice is not recommended since it can lead to other problems, especially for FSI cases. Also, please note that ***CESE_DATABASE_POINTOUT** ignores the ***CESE_EOS** cards, making the assumption that the EOS is a perfect gas EOS with $\gamma = 1.4$.
3. For the ***CESE** moving mesh solvers, correct the nodal connectivity for the edge/prism element case.
4. Several bug fixes were made that make these solvers more stable.

§7.7 *CHEMISTRY

1. For TBX flow, fixed the problem of negative temperatures when using ***CHEMISTRY** with ***CESE**.

§7.8 Constraint Cards

1. For single precision fixed gradual underflow issue that occurred with ***CONSTRAINED_INTERPOLATION_GLOBAL**.
2. Fixed issue with ***CONSTRAINED_JOINT_STIFFNESS_TRANSLATIONAL**. When such a joint is activated with a sensor and JNTF = 2 in ***CONTROL_RIGID**, you can have a scenario where lower (called negative in the manual) stop displacements are positive, or upper (called positive in the manual) stop displacements are negative. The logic in the force calculation did not cover this scenario correctly.
3. Fixed bug associated with the penalty formulation of ***CONSTRAINED_SHELL/SOLID_IN_SOLID** in the MPP version that was causing LS-DYNA to crash.
4. Fixed incorrect initial velocity of ***CONSTRAINED_NODAL_RIGID_BODY** that occurs when using ***INITIAL_VELOCITY_GENERATION** with NX = -999. The rotation axis was incorrect.
5. Corrected error in the implementation of ***CONSTRAINED_INTERPOLATION** that was leading to a core dump.
6. Fix for ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) with huge number of associated nodes. A memory issue was likely to occur, if more than 200 nodes were found in the domain of influence of the upper or lower shell element parts.
7. Fixed bug for ***CONSTRAINED_BEAM_IN_SOLID** and ***CONSTRAINED_SHELL_IN_SOLID**: coupling between beams/shells and thick shell elements did not work correctly in the MPP version.
8. Defining multiple ***CONSTRAINED_BEAM_IN_SOLID** instantiations with 3D adaptivity did not work correctly.

§7.9 Contact

1. Fix for MPP dynamic load balancing with ***CONTACT_DRAWBEAD** which was mixing up the node order along the drawbead if the nodes as input were not in increasing number order.

2. Fix initial tying in MPP groupable tied contact which was not honoring the part specific thickness from ***PART_CONTACT**. This issue could result in some nodes not being tied that should be.
3. Fix for MPP tied contact birth/death times when given by load curves.
4. Include all slave nodes in the bounding box for ***CONTACT_SINGLE_SURFACE** in MPP when performing bucket sort – even those not from segments.
5. Fixes for load balancing of eroding contacts
6. Fix for MPP eroding contact to properly handle element deletion during small deck restarts
7. Fix for MPP groupable drawbead behavior when the contact is off the edge of a thin triangle. The (u, v) coordinates were not being truncated along the $u+v=1$ line. Thus, the computed mass was coming out negative due to extrapolation. This resulted in NaNs during the friction calculation.
8. Fix bug in input processing of ***CONTACT_BEAM_TO_SURFACE** in MPP
9. Fix for MPP groupable ***CONTACT_DRAWBEAD**, which had an overly restrictive bucket sort and so could miss some of the ramp up force a drawbead should create. Also fix incorrect handling of viscous damping for this contact.
10. Add some missing initialization code for groupable ***CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET** which could cause instabilities when these contacts have a nonzero birth time.
11. Fix for cohesive element failure issue with MPP groupable tied contact. Groupable tied contact does not initialize until the first cycle, so checking the "is this node tied" flags before then was causing premature element failure.
12. Changes for MPP $\text{SOFT} = 0$ and 1 contact with birth time, to check for any contact (shell) segments that may have failed before the contact became active. Previously, these checks only happened once the contact was active and so neglected to delete contact segments

from shells that failed before the contact birth time. This fix covers both groupable and non-groupable ***CONTACT_AUTOMATIC_SINGLE_SURFACE** and ***CONTACT_AUTOMATIC_..._TO_SURFACE**.

13. Fix for ***CONTACT_ERODING** in MPP, which was not treating exposed segments correctly
14. Free hash table related to 2 sided force transducers in MPP, thereby fixing a resource exhaustion issue that caused them to stop working after 9 adaptive steps (or redecomposition steps).
15. Suppress warning saying force transducers aren't supported with full deck restarts. They have no history so re-initializing them from scratch works just fine.
16. Fix bug in MPP groupable contact that could cause incorrect behavior for contact nodes that should be deleted due to element failure.
17. Fixed an MPP bug for surface-to-surface contact with $SOFT = 1$ that happens the contact involves 2D seat belt elements that are initially inside a retractor.
18. Enabled segment-to-segment ($SOFT = 2$) eroding contact to work correctly when elements erode before the contact birth time.
19. Fixed force transducers when used with groupable MPP segment-to-segment ($SOFT = 2$) contact. A memory error termination was likely.
20. Fixed ***TERMINATION_CONTACT** to make it work when used with $THRES > 0$ or $DOF = 2$ or 3 . Some data was scrambled causing wrong behavior.
21. Fixed force transducers when used with MPP contact with $SOFT = 0$ or 1 . The fixed types are **AUTOMATIC_GENERAL**, **SURFACE_TO_SURFACE**, **NODES_TO_SURFACE**, and **ONE_WAY_SURFACE_TO_SURFACE**. Forces were sometimes being assigned to the wrong interfaces or to nonexistent interfaces, in which case a memory error could have caused a segmentation fault.
22. Fixed MPP solutions with 2D force transducers (***CONTACT_2D_FORCE_TRANSUDCER**) that have both a slave and master surface defined.

23. Fixed a flaw in the segment-to-segment (SOFT = 2) contact thickness calculation. It could cause shell segments near welds to reduce to zero thickness when SPOTHIN is active on ***CONTROL_CONTACT**, shell thickness updates are active, and either OPTT > 0 ***PART_CONTACT** or OPTT = 0 and SFT = 0 on ***PART_CONTACT** and either SST > 0 or MST > 0 on ***CONTACT**.
24. Previously ***CONTACT_AUTOMATIC_GENERAL** in SMP missed the contact between beams generated with ***ELEMENT_BEAM_SOURCE**.
25. Added field TIEDIED to optional card D on ***CONTACT_OPTION**. Setting this field to 1 causes an incremental update of the contact normal. Incremental updates were added to prevent spurious stresses that occur even with no loading when using ***CONTACT_TIEBREAK_NODES_TO_SURFACE**.
26. Fixed bug causing ***CONTACT_TIED_SURFACE_TO_SURFACE** to be ineffective after dynamic relaxation when death time is set to -9999.
27. Fixed bug causing an input error saying the element has zero nodal thickness (STR+230) when using triangular elements with shell type 23 (8-node quadratic quadrilateral shell) for contact.
28. Fix issue causing an immediate core dump during runtime or memory error upon termination when using SOFT = 0 or 1 tied contacts with SOFT = 2 surface-to-surface and single surface contacts and running with SMP.
29. Fixed incorrect duplication of higher order shell(23 and 24) parts for contact.
30. Fixed bug that was resulting in contact not being detected when the slave box (SBOXID) is defined with a ***DEFINE_BOX_LOCAL** and the master box (MBOXID) is defined with a ***DEFINE_BOX** as specified on Card 1 of ***CONTACT_OPTION**.
31. Fixed seg fault or incorrect frictional behavior when using the THERMAL_FRICTION keyword option for ***CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE** and ***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE** and running with SMP.

32. Fixed a bug that occurred during initialization for MPP when there were a large number of contacts in the model. The bug caused a segmentation fault.
33. Fix a bug that that entered the code in 2015 for ***CONTACT_SURFACE_TO_SURFACE** in SMP. LS-DYNA was performing one-way contact instead of two-way contact. As a result, the contact force was half of what it should be.
34. Extended the logic for collecting the resultant forces reported to the INTFOR file for ***CONTACT_TIED_SURFACE_TO_SURFACE** when using implicit in MPP. Before the resultant forces were being reported as zeroes for this contact type.
35. Added a warning that ***CONTACT_TIEBREAK_NODES_TO_SURFACE_ID** is not supported for implicit computations.

§7.10 Control Cards

1. Issue warning message when the number of iterations reaches the upper limit, MAXITR, in ***CONTROL_PZELECTRIC**.
2. Fixed incorrect mass scaling, DTMS2<0 in ***CONTROL_TIMESTEP**, when solid type 48 (***SECTION_SOLID_PERI**) is used in the presence of the keyword ***INITIAL_FOAM_REFERENCE_GEOMETRY**.
3. For any kind of implicit eigenvalue computation, LS-DYNA before ignored the value of TET10S8 on ***CONTROL_OUTPUT** and only wrote the corner nodes of the elements to d3plot for the connectivity. Now it accepts the setting of TET10S8, so the full connectivity can be written to the d3eigv database.
4. For staged construction, fixed bug related to unexpected forces being present in dormant elements. The bug affected solids (hourglass type 6 only), thick shells (all hourglass types), and shells (stiffness-method hourglass types). When a part is dormant (meaning the construction stage when the part becomes active has not yet been reached), stresses and forces are calculated but are scaled down (see input parameter FACT on ***CONTROL_STAGED_CONSTRUCTION**). The scaling down had been erroneously omitted for the hourglass forces pertaining to the element types and hourglass settings mentioned above.
5. For ***CONTROL_PORE_FLUID**:

- Fixed bug that existed in R12 but not previous versions such as R11. In some models containing `*CONTROL_PORE_FLUID`, the output to `d3plot` was wrong. The pore pressure variables could be written to the wrong position in the file, resulting in stresses or other extra history variables being overwritten with pore pressure, and the extra history variables that were expected to contain pore pressure data instead contained zero. The bug apparently occurred only with small simple models and did not seem to affect large complex models.
- Fixed bug affecting `ATYPE = 4` (steady state consolidation), which could occasionally occur when thin elements are sandwiched between thick elements. The calculation could become unstable in the first few time steps, with pore pressure oscillating between high positive and negative values. A work-around for the bug was to use the curve `ACURVE` on `*BOUNDARY_PORE_FLUID` to set the analysis type to time-dependent consolidation (`ATYPE = 3`) for at least 100 time steps before switching to `ATYPE = 4`.
- Fixed two bugs affecting pore fluid analysis with flow through tied contact (`FTIED` on `*CONTROL_PORE_FLUID`):
 - If a node on the tied contact reached the suction limit (see `SUCLIM` on `*BOUNDARY_PORE_FLUID`), the suction limit pressure then spread along the tied contact in a non-physical way.
 - Pressures could become unstable at nodes shared across MPP processors if the node belonged to more than one tied contact.

§7.11 Discrete Element Method

1. Fix torque evaluation from rolling friction for DES surface coupling (`*DEFINE_DE_TO_SURFACE_COUPLING`)
2. Fix bug of negative bond damage that occurred with `*DEFINE_DE_ACTIVE_REGION`.

§7.12 Elements

1. Bug fix for `PARM3` for discrete beams defined with `*ELEMENT_-BEAM_THICKNESS`. The local coordinate ID specified by `PARM3` was ignored by the discrete beam material models `*MAT_069`, `*MAT_070`, `*MAT_071`, `*MAT_074`, `*MAT_094`, `*MAT_197`, `*MAT_-205` and `*MAT_208`. The global coordinate system was used instead.

2. Fixed a 2D belt bug that happens when a belt has two retractors (***ELEMENT_SEATBELT_RETRACTOR**).
3. For ***USER_NONLOCAL_SEARCH**:
 - Fixed bug in memory allocation with respect to number of requested user element history variables.
 - Fixed bug related to reading of history variable lines of the user element keyword.
 - Fixed bug that may cause segmentation fault for user element history variables ($NUELHSV > 0$).
4. Fixed the elbow beam (***SECTION_BEAM**, $ELFORM = 14$) when the inside pressure is zero. The bug could cause a wrong result.
5. Reduced round-off error in the initial volume calculation for thick shell forms 3, 5, and 7 so that they can correctly initialize the pressure to zero when used with equation-of-state materials.
6. Fix incorrect mass/inertia when using rigid bodies affected by ***ELEMENT_MASS_PART** and ***CONSTRAINED_RIGID_BODIES**. The mass contributions from ***ELEMENT_MASS_PART** were not added to the merged rigid bodies from ***CONSTRAINED_RIGID_BODIES**.
7. Fixed spurious error message KEY+1567, saying 'ELEMENT ### is not found,' that occurs using ***ELEMENT_SHELL_COMPOSITE** if ***MAT_ADD_THERMAL_EXPANSION** is present in the input.
8. Corrected the setting of nodal inertias for solid element formulation 3 for eigenvalue computations. It was zero and now is the standard computed value.
9. Corrected the computation of the critical time step for explicit when using superelements (***ELEMENT_DIRECT_MATRIX_INPUT**).
10. Fix for ***PART_STACKED_ELEMENTS**: memory overwrite. Models could fail with "Error 10183 (KEY+183), part ... not defined", even if that part did not exist at all.
11. Fixed a bug writing out dynain and reading in ***INITIAL_STRESS_SHELL** for shell type 55.

§7.13 Initial Cards

1. For ***INITIAL_FOAM_REFERENCE_GEOMETRY**, fix an SMP bug that occurs when $NCPU > 1$.
2. Fixed SMP bug where $ISTIFF < 0$ in ***INITIAL_STRESS_SECTION** could cause a segmentation fault.
3. Fix for ***INITIAL_AXIAL_FORCE_BEAM** with beam element 1 (H-L beam). Available since R12.0, this feature could have lead to various side effects, such as severe beam deformations.

§7.14 Implicit (Mechanical) Solver

1. Fixed incorrect handling of multiple right-hand-sides during the forward / back solve for the Power Method. The Power Method extracts buckling modes (see ***CONTROL_IMPLICIT_BUCKLING**).
2. Corrected the processing of energy and other such statistics in file `glstat` for the multistep linear solution method ($NSOLVR = -1$ on ***CONTROL_IMPLICIT_SOLUTION**). Prior to the bug fix, the statistics were additive, meaning the statistics for the new time step were added to the ones from the previous time step. Now the statistics are reset at the beginning of the new time step, so they just reflect the current time step.
3. For ***CONTROL_IMPLICIT_EIGENVALUE**:
 - Removed implicit input/output files at the end of any eigencomputation. These were scratch files that were of no interest to the user.
 - Corrected an error associated with the the nonsymmetric eigensolver which impacted any problem with damping.
4. Fixed the generation of superelements using ***CONTROL_IMPLICIT_MODES** in MPP when nodes involved were shared across processors.
5. Fixed application of implicit constraint modes ***CONTROL_IMPLICIT_MODES** in MPP for the case where a constraint mode is also a shared node.

6. Correct how the load curve IDs for IAUTO and IMASS were being processed for implicit keywords ***CONTROL_IMPLICIT_DYNAMICS** and ***CONTROL_IMPLICIT_AUTO** with the options DYN or SPR enabled. Before if the curve IDs did not match the input order (such as an ID of 97125 for the curve input, then LS-DYNA would give an error. This is not how curves are normally processed. Now they are being processed like all other curves in LS-DYNA where the curve ID and order of input are independent.
7. For ***CONTROL_IMPLICIT_GENERAL**, adjust logic of when last intermittent eigenvalue computation is performed during an explicit simulation if the last one is performed near the termination time. For instance, if the termination time is 10.00 and the explicit time step is 0.01, then the explicit calculation terminates between 9.99 and 10.00. Therefore, if you wanted an eigenvalue computation performed at 10.00, it would not happen because the calculation would have already terminated. This change causes the intermittent eigenvalue computation to occur before the termination time.
8. During MPP decomposition, the nodes on an interface (see ***INTERFACE**) may be shared by multiple processes. We fixed a bug in implicit where we were double counting the interface forces for the shared nodes.
9. Correct a long standing problem associated with computing the La-Grange Multipliers for constraint processing in Implicit Mechanics. The displacements on the independent dofs were inappropriately being zeroed during this computation leading to inaccurate reporting of resultant forces for Implicit Mechanics.
10. In the previous releases, we had reduced the size of d3eigv database for models with no rotational dofs like solid element only models. Modal Dynamics ***CONTROL_IMPLICIT_MODAL_DYNAMICS** was still trying to read the nonexistent data for rotational dofs and became lost. This revision corrects that oversight.
11. Restored the ordering reuse logic to the MPP implementation of implicit dynamics. This is an important time saving feature for very large models with multiple time steps.

§7.15 Isogeometric Analysis (IGA)

1. Fixed a bug that causes the deletion of the wrong interpolation shell elements (IGA).

§7.16 Load Cards

1. Corrected effective seismic input calculation for tetrahedral and pentahedral elements.

§7.17 Materials and Equations-of-State

1. Fixed the use of equation of state materials with thick shells, when ***PART_COMPOSITE** or ***INTEGRATION_SHELL** was not used to define an integration rule.
2. Fix for ***MAT_ADD_DAMAGE_GISSMO** with $LCSDG < 0$ to avoid incorrect element deletion (only SMP/Hybrid with $ncpu > 1$).
3. Correction for ***MAT_ADD_DAMAGE_GISSMO** and ***MAT_ADD_GENERALIZED_DAMAGE** in full deck restart. Element size for $LCREGD$ and $FADEXP < 0$ was re-computed after restart, but it should not.
4. Fix for ***MAT_ADD_GENERALIZED_DAMAGE** with $long = s$. The long input format was not working with this keyword before.
5. For ***MAT_ADD_PZELECTRIC**:
 - Fixed a bug that happens when coupling coefficient for shear, P_{ijk} , is nonzero.
 - Fixed a bug for strain-based input that happens when a part and its material have different IDs.
6. Fixed bug to make 2D shell type 13 with $NIP = 4$ work correctly with ***MAT_ADD_THERMAL_EXPANSION** for implicit.
7. Fixed ***MAT_NONLOCAL** when used with eroding elements. The smoothing calculation was using uninitialized data due to a failure to set initial values for the eroded elements prior to smoothing. This error could be subtle or cause an error termination.
8. Fixed the use of material type 26, 126, or 201 with thick shell elements when $AOPT = 2$ or 3. The material directions were being miscalculated causing wrong stress.

9. Fixed incorrect initial strains when using ***MAT_MOONEY-RIVLIN_-RUBBER** / ***MAT_027** with ***INITIAL_FOAM_REFERENCE_GEOOMETRY** and running with single precision version.
10. For ***MAT_030** / ***MAT_SHAPE_MEMORY**
 - Fixed issue where shells failed to converge for plasticity.
 - Fixed issue where incorrect curve ID was used in LCSS for solids.
 - Fixed issue that could cause a segmentation fault when using this material.
11. for ***MAT_053** extrapolate load curve for von Mises stress as a function of negative volumetric strain using the first 2 points of the load curve if the volumetric strain is less than the first point of the curve. This fix prevents an error termination.
12. For ***MAT_LAMINATED_COMPOSITE_FABRIC (*MAT_058)**:
 - Fix contact stiffness in case EA, EB, or GAB are defined with load curves or tables:
 - Fixed bug for using ≥ 8 digit load-curve ID for GAB
 - Fixed bug for LCDFAIL check. Prior to this fix the LCDFAIL option did not work for shells!
13. Fix for ***MAT_076** with temperature shift. Initial temperature in an uncoupled analysis is not always zero (such as coming from ***LOAD_THERMAL**).
14. Fix for combination of ***MAT_077** / ***MAT_181** with VFLAG = 1 and ***INCLUDE_TRANSFORM**. Normalized shear relaxation moduli G_i (unitless) should not be converted.
15. Fixed incorrect effect of BVFLAG in card 2 field 1 of ***MAT_FU_CHANG_FOAM** / ***MAT_083** on initial time step size.
16. Fixed bug in ***MAT_WINFRITH_REINFORCEMENT** / ***MAT_084_REINF**: Reinforcement when defined with specific groups of elements (Card 1a) did not work in MPP. It would error terminate.
17. Fixed bug in ***MAT_100 (*MAT_SPOTWELD)** for OPT = 0/-1 when using ***DEFINE_CURVES** for defining force/moment resultants as a function of the effective strain rate, by setting NRR, NRS, NRT, MRR, MSS, MTT as negative values.

18. Fixed bug in damage evolution of ***MAT_SPOTWELD_DAMAGE-FAILURE**. Prior to this fix an earlier and wrong damage evolution could have happened. This is for beam elements only.
19. Fix initial temperature increment in ***MAT_106** for shell elements. In a structural-only analysis, the "old" temperature should be set to the initial temperature, e.g. from ***LOAD_THERMAL_LOAD_CURVE**.
20. Fixed incorrect computation of *phel* for ***MAT_110** / ***MAT_JOHNSON_HOLMQUIST_CERAMICS** that occurs when you input *phel* = 0 in the data cards.
21. For ***MAT_JOHNSON_HOLMQUIST_CONCRETE** / ***MAT_111**, bug fix in the erosion criteria.
22. For ***MAT_157_IHIS** / ***MAT_ANISOTROPIC_ELASTIC_PLASTIC**:
 - Fixed bug for IHIS = 2 where history variables were not read correctly.
 - Corrected the update of transversal (through the thickness) strain. Prior to this correction, the thickness strain was not computed correctly and did not fulfill the condition that was set with the Lankford parameters (*R*-values). This applies to shells only.
23. For ***MAT_172**:
 - Fixed bug affecting shear capacity check option *TYPESC* = 2. Note that the shear capacity calculated by ***MAT_172** affects only the output history variables, not the deformations or stresses. The constants in the equations have been corrected to match ACI 318-05M equations 11-4 and 11-8. Before this change, any axial compressive stress would have caused the reported shear capacity to be greater than required by 11-4. Furthermore, minor corrections have been made to other terms to match the metric edition of ACI 318 - previously the constants had been converted from imperial units with a greater level of precision than in ACI 318-05M.
 - Fixed bug in implicit solution that occurred when *TYPEC* = 3 or 6 and *PR36* was left zero: a crash occurred on the first time step. A work-around was to set nonzero *PR36*.
24. Fix problem with *long=s* for ***MAT_196** to avoid error termination.

25. Fixed bug in ***MAT_BOLT_BEAM** that existed in R12 but not earlier versions such as R11. The combination of LCSHR being a DEFINE_TABLE with AXSHFL = 0 did not work correctly, i.e. the shear force plastic limit would not be the one expected based on LCSHR and the current shear deformation.
26. Make option E < 0 of ***MAT_224_GYS** work correctly, which was not the case before.
27. For ***MAT_CAZACU_BARLAT** / ***MAT_233**, removed a redundant term in the hardening modulus calculation when the material hardening is temperature and strain rate sensitive.
28. Fixes for ***MAT_PIECEWISE_LINEAR_PLASTIC_THERMAL (*MAT_255)**:
 - Avoid error termination if huge number of curves (> 9999) is used in a model.
 - Avoid error termination with large curve IDs for E < 0 and PR < 0.
29. Fixes for ***MAT_258**: repair visco-plasticity, use correct thickness of shell type 16, and eliminate potential dependence on ncpu.
30. For ***MAT_LAMINATED_FRACTURE_DAIMLER_PINHO (*MAT_261)**:
 - Check for fiber kinking, even if FMAT = 1.0. This allows for proper behavior if fiber compression and matrix tension is present at the same time.
 - Define minimal values for fracture toughnesses if not defined to avoid division by zero.
 - Make sure that stress will not exceed the stress at damage initiation (i.e. $\sigma_{\text{kink}} > \sigma_{\text{kink}_0}$ / $\sigma_t > \sigma_{t_0}$). The damage variable has to be at least high enough to limit the stress. There might be special scenarios where the stress keeps growing while the driving strain is not (which would lead to no damage accumulation).
31. Fixed issue with ***MAT_264** / ***MAT_TABULATED_JOHNSON-COOK_ORTHO_PLASTICITY** which gives incorrect results when running with adaptivity.
32. Fixed bug in ***MAT_274** where shell thickness strain was not updated properly.

33. For ***MAT_295** / ***MAT_ANISOTROPIC_HYPERELASTIC**:
- Fixed a bug which let the simulation run even if negative determinants were reported in the stress update routine in MPP.
 - For the ANISO module, fixed a bug for fiber models other than the Holzapfel-Gasser-Ogden; the compression tension switch is now uniformly based on the total fiber stretch.
 - For the ANISO module, fixed a bug when using fiber stress vs stretch relations defined by a curve.
 - Fixed a bug in the keyword reader of the ACTIVE module.
34. For ***MAT_SEATBELT**
- Fixed a bug for 2D seat belts that can result in seg faults when there is no retractor in the model.
 - Fixed a bug when initializing initial strain for seat belt elements initially inside a retractor.
35. For ***MAT_SPH_VISCOUS**:
- Fixed bugs to make this material model work properly with SPH particles.
 - Fixed bugs of the user subroutine inside ***MAT_SPH_VISCOUS** for the user defined viscosity to make this option work properly.
36. Fixed bug causing an error termination with error message STR+1475 that occurred when mixing materials with resultant and non-resultant materials in the same ***INTEGRATION_SHELL** or ***PART_COMPOSITE**.

§7.18 MPP

1. Fix failure during MPP tied contact initialization caused by ***AIRBAG_REFERENCE_GEOMETRY**.
2. Fix MPP to support general functions when performing pre-decomposition based runs.
3. Fix dynamic load balancing issue that could cause deadlock when processors were being removed from contact interfaces.

4. Fix for MPP pre-decomposition in models with airbags. When pre-decomposition is done for more processors than the calculation, the airbag ownership was not being properly handled leading to problems including possibly seg fault.
5. Fix bug in ***TERMINATION_DELETED_SOLIDS**, which wasn't terminating properly in MPP.
6. Fix some initialization checks related to ***PART_AVERAGED**, and prevent execution of related code on processors that have none of these beams (MPP only)
7. Add MPP support for ***TERMINATION_BODY**, which was never properly implemented in MPP.
8. Fix for possible divide by 0 during dynamic load balancing and in-core adaptivity on processors with no contact interfaces
9. Fix for MPP treatment of ***CONSTRAINED_SHELL_TO_SOLID** which in rare cases, depending on the specific decomposition, could have generated NaNs during initialization when scaling the node rotations for stability.
10. Fix for MPP handling of ***DEFINE_CURVE_FUNCTION**. Specifically, if the built-in function PIDCTL was used, along with MPP pre-decomposition, the behavior was unpredictable.
11. For ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION**, fix full deck restart with deleted rigidwall.

§7.19 Multiscale

1. Fixed bugs associated with including several different mesoscale spot weld models, each included with ***INCLUDE_MULTISCALE_SPOTWELD**.
2. Fixed the offset option for the contact between the beams in the macroscale spot weld and the master interface when the offset distance is larger than the beam length

§7.20 Output

1. Fixed issue in MPP that resulted in incorrect reporting of rotational masses to `massout`.

2. Fixed bug that caused shell data to be skipped in the d3thdt output routines ***DATABASE_BINARY_D3THDT**. This issue resulted in corrupted d3thdt files.
3. For thermal-only simulations, honor the setting of BINARY in ***DATABASE_CURVOUT**. It was previously only outputting an ASCII format even if an LSDA format was requested.
4. Removed very old diagnostic write statements that could have potentially corrupted the deforc file in any model using ***MAT_WOOD**.
5. For linear DK shell elements (***SECTION_SHELL**, ELFORM = 18) the moments are now correctly written to the cross section output generated by ***DATABASE_SECFORC**. Before, incorrect values were reported.
6. Fixed error in elout for beam, solid, shell and tshell elements that occurred if a beam ID had 9 digits.
7. Fixed error in nodout that occurred if a node ID had 9 digits.
8. Restored how encrypted files are echoed to d3hsp back to how it was done in version R9.3.1.
9. Fix for DCOMP = 5 and 6 on ***DATABASE_EXTENT_BINARY** option when PSET on ***DATABASE_BINARY_D3PLOT** option is also used. This issue was corrupting the d3plot file.
10. Fixed a bug pertaining to the gathering of nodal volumes in MPP for solid element formulation 13. This fixes issues with erratic stress distributions in d3plot.
11. Honor the setting of IEVERP in ***DATABASE_EXTENT_BINARY** for thermal only calculations.
12. For ***DATABASE_CROSS_SECTION**:
 - Fixed a bug that happens when a cross section contains 2D seat belts.
 - Fixed bug where cross-sectional area was not displayed properly in secforc for local coordinate systems.
 - Fixed bug causing NaN in secforc when using beam type 6 with zero area and length.

- With this fix, forces/moments are being output to `secforc` for beam type 13.
13. For ***DATABASE_RECOVER_NODE**:
 - Fix an MPP bug that might happen when there are 10-node tetrahedron element in the model.
 - Issue warning message when no vertex is detected for Zienkiewicz-Zhu method (`METHOD = 0`).
 14. Fixed a bug that happens when `NEWLEG` of ***CONTROL_OUTPUT** is > 0 and `SBTOUT` (seat belt output with ***DATABASE_SBTOUT**) is not requested.
 15. Fixed segmentation fault when writing out dump file after dynamic relaxation when using a large number of layers in ***ELEMENT_SHELL_COMPOSITE**. This affects MPP only.
 16. Fixed bug so that the strain tensor is output to `d3part` when `STRFLG > 0` in ***DATABASE_EXTENT_D3PART** and `STRFLG = 0` in ***DATABASE_EXTENT_BINARY**. Previously, the strains were not being output to `d3part`.
 17. Fixed zero torsion output to `swforc` for beam spot welds.
 18. Fixed incorrect ***INITIAL_INTERNAL_DOF_SOLID_TYPE3** output to the `dyna.in` file. It caused negative volume when used to initialize stresses in solid types 3 and 4.
 19. Fixed pressure computation in `intfor` so that it matches the pressure computed using forces from `ncforc` and the segment area.
 20. Fixed a bug in the moment computation for the moments output to `rcforc` (***DATABASE_RCFORC_MOMENT**). This bug only affects post-processing, not the analysis itself.
 21. Fixed possible bug in output of stresses for non-isotropic materials when using `CMPFLG = 1` on ***DATABASE_EXTENT_BINARY** and ***DEFINE_MATERIAL_HISTORIES**. Prior to this fix the output of stresses might not have been properly transformed into the material frame. This only affects the post-processing and not the analysis itself.
 22. Added missing legend/title to ***DATABASE_DISBOUT** output files.

23. Corrected time step output for the SPG controlled time steps to d3hsp.

§7.21 Restarts

1. Fix simple/small restart with ***LOAD_SEGMENT_FILE**. It was reading the incorrect family file.
2. During simple restart, only open an ***INTERFACE_LINKING** file on processor 0 in MPP.
3. Fixed compatibility of ***DEFINE_SPH_TO_SPH_COUPLING** with full deck restart.
4. Fixed bug in full deck restarts of models containing cylindrical joints. Some joint history data was not recovered properly.
5. Bug fix for all restart jobs run with double precision executables. The bug was a memory error that might lead to erroneous results and existed in all double precision executables since June of 2012.
6. Fix seg fault with full deck restart when initial run is 6 degrees-of-freedom while the full deck restart run is 3 degrees-of-freedom.
7. Fixed small-deck restart for ***LOAD_SEISMIC_SSI**, ***INTERFACE_SSI**, and ***LOAD_SPCFORC**.
8. Fixed bug causing an error during full deck restart for ***CONTACT_TIEBREAK_SURFACE_TO_SURFACE**.

§7.22 Sensors

1. Fix old typo in ***SENSOR** related to tied contacts, which should have converted them to GROUPABLE but was only working correctly for the first contact in the model.
2. For ***SENSOR_CONTROL**:
 - Fixed a bug that failed to recover failed elements (TYPE=ELESET)
 - Fixed a bug for TYPE = PRESC-MOT that happens when the node or rigid part to which the boundary condition is to be applied has a big ID.
 - Fixed a bug for TYPE = BAGVENTPOP. The bug caused venting to not turn off for ***AIRBAG_HYBRID**.

- Fixed a bug for TYPE = SPC that was triggered when trying to turn on SPC in the middle of the analysis.
3. Fixed a bug for MTYPE = BNDOUT of ***SENSOR_DEFINE_MISC**.
 4. Fixed an MPP bug for FTYPE = JOINT of ***SENSOR_DEFINE_FORCE** that might result in a seg fault.
 5. Fixed an MPP bug for ***SENSOR_DEFINE_FORCE** with TYPE = JOINTSTIF.
 6. Fixed a bug for ***SENSOR_DEFINE_ELEMENT** with TYPE=STRAIN that occurs when STRFLG of ***DATABASE_EXTENT_BINARY** is "0".
 7. Fix for ***SENSOR_DEFINE_FORCE** with nonzero CRD and long=s. Error in structured read happened (STR+211) with MPP and np > 1.

§7.23 SPH

1. Fix for ***DEFINE_ADAPTIVE_SOLID_TO_SPH** now that we allow up to 8 particles per direction, per element. It was not working correctly if there were more than 5 particles per direction.
2. Bug fix in ***DEFINE_ADAPTIVE_SOLID_TO_SPH** with tet elements. The parametric coordinates were incorrect, so particles were created in wrong locations.

§7.24 Thermal Solver

1. For MXDMP \neq 0 (matrix dumping for SOLVER > 11) on ***CONTROL_THERMAL_SOLVER**, fixed incorrect output in MPP.
2. An issue was fixed when ***DEFINE_CURVE_FUNCTION** is used in thermal calculations.
3. LS-DYNA echoes a reasonable error message when a view factor file is missing and ***BOUNDARY_RADIATION_OPTION1_VF_READ** is used.
4. An input trap was added when using a wrong TYPE option in ***BOUNDARY_RADIATION_SET**.
5. Avoid an error termination when ***LOAD_THERMAL_LOAD_CURVE** is used with adaptivity.

6. A reasonable message is echoed if the d3heat file used with keyword ***LOAD_THERMAL_D3PLOT** is not specified in the command line, and LS-DYNA will exit gracefully.
7. Enabled ***LOAD_THERMAL** to work for beam nodes when nodal releases are defined at those nodes. The thermal loading was incorrect.

§7.25 XFEM (eXtended Finite Element Method)

1. Fixed a bug in defining a pre-crack for XFEM. Added a warning message when changing pre-crack defining points so the pre-crack cannot pass nodal points.
2. Fixed a bug in 2D XFEM (ELFORM = 52 in ***SECTION_SHELL_XFEM**) so it works when applied to plasticity materials.

§7.26 Miscellaneous

1. To improve memory performance, removed initial large allocation that was zeroing all of memory and thereby physically allocating it on Linux. This change restores behavior that existed in releases prior to R11.2
2. Store file name and format flag for ***INTERFACE_COMPONENT_FILE** into the structured file. With this change, if we convert keyword input to structured input and run from the structured input, LS-DYNA will know the required file format for the component data. Previously, LS-DYNA always produced a non-lsda file when running from a structured file.
3. Fix ordering issue with ***DEFINE_FUNCTIONS** in the initialization phase (structured input) such that all of these are before ***DEFINE_CURVE_FUNCTIONS**. This prevents a ***DEFINE_CURVE_FUNCTION** that reference a ***DEFINE_FUNCTION** from being before the ***DEFINE_FUNCTION**.
4. Fix memory issue caused by ***USER_INTERFACE_FRICTION**.
5. Fixed a bug that happens when BOX or VOL is used for ***SET_NODE_GENERAL** in models containing ***NODE_TRANSFORM**.
6. Fix indexing error related to ***SET_SEGMENT_GENERAL**, which could have resulted in incorrect values being applied for the DA1-DA4 parameters, or possibly a segfault during input processing.

Listing 1

```
1 *PARAMETER
2 $   PRMR1      VAL1      PRMR2      VAL2
3 RuPLscXXa1 &R1uP11SARuPLscXXt1      0.0
```

7. Fixed bug causing memory allocation problems for PGI binaries running models with thick shells output to the `e1out` file
8. Corrected some bugs related to ***INTERFACE_LINKING** that could lead to segmentation faults.
9. Fix bugs causing memory allocation issues for executables built with the PGI compiler.
10. Fixed bug in reading a character parameter (***PARAMETER**) in a comma-delimited input. It did not conform to the usual comma-delimited input conventions.
11. Fixed bug in reading data cards of ***PARAMETER** that occurs if there is no space between a symbol and a value like in [Listing 1](#).
12. For ***DEFINE_CURVE_FUNCTION**:
 - Fixed an MPP bug for FX, FY, FZ, TX, TY and TZ in that might result in erroneous result.
 - Fixed a bug for function SENSOR that sometimes failed a job and printed a bogus message.
 - Fixed a bug for PIDCTL that occurs during thermal-only simulations.
13. Fixed a bug that happens in double precision of PGI machines associated with ***INCLUDE_NASTRAN**.
14. For ***DEFINE_PRESSURE_TUBE**:
 - Fixed bug in equilibrium equations for Continuous Galerkin (MTD = 0).
 - Removed bottleneck in MPP when using many rigid parts in the model.
15. Fixed bug in argument lists for user subroutines when using ***MODULE**. Before this fix, some user subroutines (`umatc`, `usrshl`, `usrslid`, `usrfrc`, `usrtie`, `usrtbrk`, `usr_nunonl`, `mortar_*`, `rebar_*`, `loadsetud`) may not have worked correctly.

16. Fixed a problem with ***NODE_TRANSFORM** when used with spot weld assembly generation. Spot weld elements were getting distorted.
17. Fixed seg fault that occurs during the output of a d3plot file after a solid element is deleted. This behavior occurred when the material model was a user material.
18. Fix 'reference load curve ID # cannot be found' error, KEY+825, when load curves for ***DEFINE_TABLE_2D/3D** are contained in a different include files after the include file that contains the ***DEFINE_TABLE_2D/3D** statement.
19. Fixed bug causing error 'BEAM INT ID ### does not exist' (KEY+308) when using ***INTEGRATION_BEAM** with ***INCLUDE_TRANSFORM**.
20. Fixed issue with ***DEFORMABLE_TO_RIGID_AUTOMATIC** when CODE = 4 and OFFSET > 0.0. Offset was not working correctly.
21. For ***SET_OPTION_COLLECT**, fixed a bug in which the code stops with an error message when the first set of this kind is empty.
22. Few fixes and improvements for keyword memory handling of larger models, mainly affecting keywords ***SET_NODE**, ***SET_SOLID**, ***SECTION_SOLID**, ***INITIAL_STRESS_SHELL**, and ***AIRBAG_REFERENCE_GEOMETRY**. Time spent in keyword reading phase is reduced.
23. Activate error/warning for failed/unconstrained spot weld assembly solids (***DEFINE_HEX_SPOTWELD_ASSEMBLY**). Up to now, this error or warning (depending on SPOTSTP from ***CONTROL_CONTACT**) was only issued for single spot weld solids.