

# R17.0.0 Release Notes

June 9, 2026

- 1 Release notes revision
- 2 License
- 3 Documentation
- 4 Notes
- 5 New
  - 5.1 Acoustics, NVH, and Frequency Domain
  - 5.2 Adaptivity
  - 5.3 Airbags
  - 5.4 ALE
  - 5.5 Constraint Cards
  - 5.6 Discrete Element Method
  - 5.7 Dual CESE Compressible Fluid Solver
  - 5.8 Elements
  - 5.9 EM (Electromagnetic Solver)
  - 5.10 Fatigue Cards
  - 5.11 Forming Analysis
  - 5.12 ICFD (Incompressible Fluid Solver)
  - 5.13 Isogeometric Analysis (IGA)
  - 5.14 Load Cards
  - 5.15 Materials and Equations-of-State
  - 5.16 MPP
  - 5.17 Multiscale
  - 5.18 SPG (Smooth Particle Galerkin)
  - 5.19 SPH
  - 5.20 Thermal Solver
  - 5.21 Miscellaneous
- 6 Improvements and Enhancements
  - 6.1 Acoustics, NVH, and Frequency Domain
  - 6.2 Adaptivity

- 6.3 Airbags
- 6.4 ALE
- 6.5 Battery Cards (Electrochemistry Solver)
- 6.6 Blast
- 6.7 Boundary Cards
- 6.8 Constraint Cards
- 6.9 Contact
- 6.10 Control Cards
- 6.11 Dual CESE Compressible Fluid Solver
- 6.12 Elements
- 6.13 EM (Electromagnetic Solver)
- 6.14 Forming Analysis
- 6.15 ICFD (Incompressible Fluid Solver)
- 6.16 Implicit (Mechanical) Solver
- 6.17 Initial Cards
- 6.18 Isogeometric Analysis (IGA)
- 6.19 Load Cards
- 6.20 Materials and Equations-of-State
- 6.21 Mesh Cards
- 6.22 MPP
- 6.23 Multiscale
- 6.24 Output
- 6.25 Sensors
- 6.26 SPG (Smooth Particle Galerkin)
- 6.27 SPH
- 6.28 Thermal Solver
- 6.29 Miscellaneous
- 7 Bug Fixes
  - 7.1 Acoustics, NVH, and Frequency Domain
  - 7.2 Adaptivity
  - 7.3 Airbags
  - 7.4 ALE
  - 7.5 Battery Cards (Electrochemistry Solver)
  - 7.6 Boundary Cards
  - 7.7 CESE Compressible Fluid Solver
  - 7.8 Constraint Cards
  - 7.9 Contact
  - 7.10 Control Cards

- 7.11 Discrete Element Method
- 7.12 Dual CESE Compressible Fluid Solver
- 7.13 Elements
- 7.14 EM (Electromagnetic Solver)
- 7.15 ICFD (Incompressible Fluid Solver)
- 7.16 Implicit (Mechanical) Solver
- 7.17 Initial Cards
- 7.18 Isogeometric Analysis (IGA)
- 7.19 Load Cards
- 7.20 Materials and Equations-of-State
- 7.21 MPP
- 7.22 Multiscale
- 7.23 Output
- 7.24 Restarts
- 7.25 Seat Belts
- 7.26 Sensors
- 7.27 SPH
- 7.28 Thermal Solver
- 7.29 User-Defined Features
- 7.30 Miscellaneous

## **1 Release notes revision**

This file constitutes revision 0 of the release notes for Ansys LS-DYNA<sup>®</sup> version R17.0.0.

## **2 License**

The string “REVISION 17” must appear in the license file to run version R17.0.0. Please contact your distributor or your Synopsys sales representative if you need to have your license updated.

## **3 Documentation**

Until such time as the R17.0 User's Manual is posted at <https://lsdyna.ansys.com/manuals-download/>, the DRAFT User's Manual may be used as a temporary substitute. Note that some features mentioned in the DRAFT User's Manual may not be implemented for R17.0.0.

## 4 Notes

The remainder of this file briefly describes new features, enhancements, and bug fixes in version R17.0.0. Other recent releases may also include some of these updates.

We arranged the items by category. Note that in many cases, a particular item may pertain to more than one category. In the interest of brevity, we list each item only once, under a single category.

## 5 New

### 5.1 Acoustics, NVH, and Frequency Domain

1. Added the ability to run an FRF analysis based on reduced eigenvector output using the REDUCED keyword option of **\*FREQUENCY\_DOMAIN\_FRF**.

### 5.2 Adaptivity

1. Enabled the ability to select the tetrahedral volume remesher for 3D *r*-adaptivity (ADPTYP = 7 on **\*CONTROL\_ADAPTIVE**). Two remeshers are available: the classic remesher and an alternative remesher. This feature is controlled by ADPTGEN on **\*CONTROL\_ADAPTIVE**.
2. Added the ability to compute the microstructure evolution in a 3D *r*-adaptive (ADPTYP = 7 on **\*CONTROL\_ADAPTIVE**) part during

hot forging simulations. This feature is controlled by ADPMIC on **\*CONTROL\_ADAPTIVE** and requires **\*MAT\_ADD\_MICRO** to input material constants.

## 5.3 Airbags

1. Added keyword **\*DEFINE\_CPG\_ADAPTIVE** to adapt particle size based on distance from the wall.
2. Added keyword **\*DEFINE\_DE\_TO\_CPG\_COUPLING** to define a coupling interface between discrete element spheres and a CPG airbag. Coupling can be two-way, one-way with the DEM data passed to CPG, and one-way with the CPG data passed to DEM. Drag forces and heat fluxes are exchanged between CPG and DEM. Associated energies and works can be tracked in `abstat_cpg`.
3. Added the ability to use the  $k-\varepsilon$  turbulence model with CPG airbags; see ITURB on **\*CONTROL\_CPG**.

## 5.4 ALE

1. Added keyword **\*DATABASE\_BINARY\_FSILNK\_FULL** to store the full force from FSI in the ALE interface linking database. The full force, instead of pressure, can then be loaded onto the Lagrangian structure in subsequent Lagrangian-only runs.

## 5.5 Constraint Cards

1. Added harmonic gear joint modeling to **\*CONSTRAINED\_JOINT** and **\*CONSTRAINED\_JOINT\_COOR** with the HARMONIC keyword option.

## 5.6 Discrete Element Method

1. Added thermal coupling between Discrete Element Spheres (DES) and

solid or shell elements using keywords **\*CONTROL\_DISCRETE\_ELEMENT**, **\*DEFINE\_DE\_BY\_PART**, and **\*DEFINE\_DE\_TO\_SURFACE\_COUPLING**. New input fields provide the atmospheric temperature, DES temperature, thermal conductivity between shell/solid elements and DES, surface temperature, specific heat capacity, and the thermal expansion coefficient. The DES can rigidly expand or shrink to simulate soft materials under the influence of temperature.

## 5.7 Dual CESE Compressible Fluid Solver

1. Support subcycling when using the IBM-FSI dual CESE solvers in conjunction with the structural mechanics solver (see SUBCYC on **\*DUALCESE\_SOLVER\_SELECTION**). Subcycling allows for more efficient simulations by enabling multiple steps of the structural solver between dual CESE fluid solver updates. While this feature can significantly speed up the simulation, you should be mindful of the frequency of FSI feedback updates to maintain accuracy; excessively large steps between updates may impact results. When enabled, the number of mechanics time steps between fluid time steps is controlled by an error tolerance. The error tolerance has a default value of 0.001 but can be input as a constant value or a load curve that modifies it in time (see **\*DUALCESE\_SOLVER\_SELECTION**). The maximum allowed single time step structural displacement equals the error tolerance multiplied by the maximum element length in the dual CESE mesh. A maximum number of subcycle steps can also be specified.
2. Added the pyramid element as a new element type available in the 3D case. This is used as a transition element type by many mesh generators.
3. Made conjugate heat transfer available for the IBM-FSI single-fluid solver (ICHT on **\*DUALCESE\_MESH\_PART**) when coupled with the structural thermal solver.
4. Added keyword **\*DUALCESE\_INITIAL\_REGION**, which uses **\*DEFINE\_REGION** to enable more flexible problem setup for the single fluid solvers, with and without the IBM-FSI capabilities.

5. Ported the chemically reacting flow solver from the CESE solver into the dual CESE solver framework. This feature involves the following new keyword card:

- **\*DUALCESE\_SOLVER\_CHEMICALLY\_REACTING**
- **\*DUALCESE\_INITIAL\_CHEMISTRY**
- **\*DUALCESE\_INITIAL\_CHEMISTRY\_SET**
- **\*DUALCESE\_INITIAL\_CHEMISTRY\_REGION**
- **\*DUALCESE\_CHEMISTRY\_D3PLOT**

The dual CESE chemically reacting flow solver is available for the most important features. The solver options ported from the CESE version include 3D, 2D, and 2D axisymmetric versions, with and without the IBM-FSI capability. The dual CESE version includes the following added capabilities:

- Coupling with dual CESE point sources
- Coupling with dual CESE porous media
- binout output of species mass fractions
- Initialization based upon **\*DEFINE\_REGION**
- Restart support
- Output of species mass fractions in the Ansys EnSight™ Gold format, invoked with the new dual CESE EnSight keyword cards.

Note that the only chemistry solver available for use with the dual CESE chemically-reacting flow solver is **\*CHEMISTRY\_CONTROL\_FULL**.

6. Added keyword **\*DUALCESE\_BLOCKMESH** to enable the quick creation of a single block-structured mesh with variable element spacing in each coordinate direction. This feature provides a quick way to set up the mesh with its boundary segment sets for testing dual CESE use in new applications. Note that this feature is not intended to replace higher-quality meshes that will give much better simulation results.
7. Added the ability to output most dual CESE results in the EnSight Gold format. This is an alpha-level feature; additional capabilities are under development. This output mechanism gives more flexibility in optimizing output for post-processing. In addition, it provides support for dual CESE meshes containing pyramid meshes and those generated

by a future solver based upon adaptive meshing. The following keywords are available for this feature:

- **\*DUALCESE\_ENSIGHT\_TIMESEQ**
- **\*DUALCESE\_ENSIGHT\_DOMAIN**
- **\*DUALCESE\_ENSIGHT\_REGION**

## 5.8 Elements

1. Redesigned superelement (**\*ELEMENT\_DIRECT\_MATRIX\_INPUT**) support to handle large rotations and explicit analyses with mass scaling. Furthermore, the visualization of the superelement is slightly improved.

## 5.9 EM (Electromagnetic Solver)

1. Added  $EMSOL = 7$  to **\*EM\_CONTROL**. It invokes a frequency domain solver that solves the Helmholtz equation for the electric field. This solver is for the high frequency domain and targets applications that include high frequency electromagnetic waves, radar, antenna, MRI and other similar applications. Excitation fields are typically plane waves (radar, scattering) or current sources (antennas, coils, wires). See **\*EM\_EXTERNAL\_FIELD** and **\*EM\_CIRCUIT\_SOURCE** for the excitation fields.
2. Added  $EMSOL = 8$  to **\*EM\_CONTROL**. It invokes an FEM solver in the time domain that solves the quasistatic electrostatic equation for dielectric breakage and capacitor stud applications.
3. Added  $EMSOL = 9$  to **\*EM\_CONTROL**. It invokes a radiofrequency (RF) heating solver in the frequency domain. In previous versions, this solver could already be used by using  $EMSOL = 3$  and simply defining a second line to **\*EM\_MAT\_001**, but the increased capabilities of the EM solver made us reorganize this capability into its own  $EMSOL$  for consistency. Warning: for consistency also, when this solver is used, the amplitude defined in the boundary condition is now the peak amplitude. Previous versions asked for the RMS value instead.

4. Added keyword **\*EM\_EXTERNAL\_ECVEL**. This keyword allows you to add the Eulerian velocity effects to your Eddy current analyses when working in an Eulerian framework (mesh not moving but imposed velocity known). This can typically be useful in rotating cases to avoid costly BEM matrix reassemblies.
5. Added `FTYPE = 4` to **\*EM\_EXTERNAL\_FIELD**, which enables defining a planar excitation wave when `EMSOL = 7` is selected in **\*EM\_CONTROL**.
6. Added keyword **\*EM\_SOLVER\_BEM\_REASSEMBLY**. This keyword enables keeping part of the previous BEM matrices during reassembly by specifying the list of parts that do not move or deform either themselves or relative to each other (typically fixed rigid bodies). This feature can save a lot of calculation time in transient applications, such as magnetic metal forming and magnet snapping.
7. Added **\*EM\_EP\_TENTUSSCHER\_STIMULUS2** to apply stimulation to electro-physiology cell models using a load curve. This keyword is supported for all electro-physiology cell models.
8. Added keyword **\*EM\_EP\_CELLMODEL\_STEP**. This ionic cell model is a simple step function with a resting potential, depolarization potential, and duration. This model can be helpful for Eikonal solves where you only care about the duration of the active phase.
9. Added the Trovato cellular action potential model for electro-physiology invoked with **\*EM\_EP\_CELLMODEL\_TROVATO**.
10. Support using an external variable (**\*LOAD\_EXTERNAL\_VARIABLE**) in the **\*DEFINE\_FUNCTION** used by **\*EM\_RANDLES\_SHORT**. With this functionality, the value of an external variable can be used to trigger short circuits in a battery model.

## 5.10 Fatigue Cards

1. Added keyword **\*DATABASE\_FATIGUE\_RAINFLOW\_MATRIX** to provide a rainflow matrix plot for fatigue analysis.

2. Added keyword **\*TERMINATION\_FATIGUE** to terminate the calculation early based on fatigue damage. The feature is only supported for **\*FATIGUE\_ELOUT**.
3. Added **\*FATIGUE\_D3PART** to run a fatigue analysis with d3part instead of d3plot.
4. Added fatigue damage computation for proportional loading with **\*FATIGUE\_STATIC**.

## 5.11 Forming Analysis

1. Added the SET\_LIST keyword option to **\*INCLUDE\_STAMPED\_PART\_SOLID\_TO\_SOLID** to map more than one solid part. With this option, the data input includes a card for listing the target parts. All parts in the source file are considered for mapping to the target.
2. Added the ability to include a transitional draw bead for obtaining the transitional bead force (**\*CONTROL\_FORMING\_DRAWBEAD\_TRANSITION**). In a stamping simulation, a line bead is commonly used. Usually, the line bead has a constant bead force in the numerical model. This keyword enables obtaining the transitional bead force. The transitional bead force changes linearly from one end to the other. If the end of the transitional draw bead has a neighboring draw bead, then the force on that end takes the value from its neighboring draw bead. If the end is open, the bead force is zero on that end.
3. Added **\*INTERFACE\_COMPENSATION\_TRIMMING\_CURVE** to update the trimming curves when performing compensation.

## 5.12 ICFD (Incompressible Fluid Solver)

1. Added the ability to select the structural parts for computing the immersed interface (see PID on **\*ICFD\_CONTROL\_IMMERSED**) for the RIIS immersed method (IMMTH = 4 on **\*ICFD\_CONTROL\_IMMERSED**) and the discontinuous FEM approximation method (IMMTH = 1 on **\*ICFD\_CONTROL\_IMMERSED**).

2. Added keyword **\*ICFD\_CONTROL\_IMMERSED\_FSI** to select the immersed coupling type.
3. Added **IMMTH = 4** to **\*ICFD\_CONTROL\_IMMERSED** to enable the RIIS immersed method without DEM particles.
4. Added keywords **\*ICFD\_DATABASE\_BIN** and **\*ICFD\_INITIAL\_BIN**. These two keywords are used together for two-step simulation workflows in which the results output from the first simulation **\*ICFD\_DATABASE\_BIN** are then used to initialize a subsequent analysis with **\*ICFD\_INITIAL\_BIN**. The geometries must match between the two steps, but the meshes can be different. This feature can be useful in a variety of problems. For example, some FSI problems benefit from a starting point that already features a developed flow. In conjugate heat transfer problems, it could be used to focus on the transient thermal behavior while assuming steady flow conditions for velocity.
5. Added **\*ICFD\_DATABASE\_VOL** to print out general geometry information for the fluid domain, such as total volume and center of gravity.
6. Added keyword **\*ICFD\_INITIAL\_VISCOELASTIC** to initialize components of the conformation tensor.
7. Added HVAC modeling ability to ICFD (see HVAC on **\*ICFD\_MODEL\_SPECIES\_TRANSPORT**). The HVAC solver couples to the Boussinesq Navier-Stokes momentum equation. See HVAC on **\*ICFD\_MODEL\_SPECIES\_TRANSPORT** to enable this model. See **\*ICFD\_BOUNDARY\_PRESCRIBED\_WATERVAPOR** for setting the water vapor mass fraction on a boundary and **\*ICFD\_INITIAL\_WATERVAPOR** for initializing the water vapor mass fraction.
8. Made further developments to the multi-species transport solver by supporting more than two species, adding chemical reactions, and enabling fluid density that depends on species concentrations. **DENSRCT = 1** on **\*ICFD\_MODEL\_SPECIES\_TRANSPORT** enables specifying a function, **DENSFID**, to provide the fluid density as a function of species concentrations. **DENSRCT = 2** enables modeling reactive species transport with the reaction rates for each

species given as functions. Additionally, the mass diffusion for each species can be defined. **\*ICFD\_BOUNDARY\_PRESCRIBED\_SP-TRANSP\_CONC** has been modified to provide the concentration of each species at the boundary. Lastly, **\*ICFD\_DEFINE\_SP-TRANSPSOURCE** has been modified to specify the species for which the volumetric species source applies (see SPIND).

9. Added the ability to compute and track the hemolysis index and scalar shear stresses with keyword **\*ICFD\_DEFINE\_HEMOLYSIS\_INDEX**.

## 5.13 Isogeometric Analysis (IGA)

1. Support steady-state and transient thermal conduction analysis for isogeometric boundary fitted/untrimmed structured splines (**\*IGA\_3D\_NURBS\_XYZ**) and unstructured splines (**\*IGA\_3D\_BASIS\_TRANSFORM\_XYZ**) when using IGA solid elements.
2. Support conjugate heat transfer, coupling IGA with the ICFD solver (SMP only) for isogeometric boundary fitted/untrimmed structured splines (**\*IGA\_3D\_NURBS\_XYZ**) and unstructured splines (**\*IGA\_3D\_BASIS\_TRANSFORM**). This feature is supported for IGA solid elements.

## 5.14 Load Cards

1. Added keyword **\*LOAD\_SEISMIC\_FREE\_FIELD**, a boundary condition for the outer faces of the soil block where earthquake loading is applied to soil-structure interaction models. The “free field” is included as a separate soil column within the same model. The boundary condition applies damping similar to a non-reflecting boundary condition except that the motion relative to the free field is used to calculate the damping forces. The method is that described as “free-field boundary elements” in PEER Report No. 2019/02 Section 3.5.2.

## 5.15 Materials and Equations-of-State

1. Added material model **\*MAT\_331/\*MAT\_IMPACT\_-ORTHOTROPIC**. It is used for modeling the behavior of composite materials subjected to hypervelocity impact.
2. Added **\*EOS\_043/\*EOS\_LEE-TARVER**. This equation of state is an implementation of the Lee-Tarver ignition and growth model, a phenomenological framework describing how shocked, heterogeneous explosives transition from unreacted solids to detonation products. This formulation implements the Lee-Tarver Ignition and Growth model originally developed within the Ansys® Autodyn solver and has been adapted for broader application. This type of equation of state is classified as a reactive burn model. Compared to **\*EOS\_IGNITION\_AND\_GROWTH\_OF\_REACTION\_IN\_HE**, EOS Form 43 requires fewer parameters, while maintaining consistency with much of the external literature.
3. Added equation of state **\*EOS\_AERATED\_WATER (\*EOS\_044)**. With this equation of state, water is modeled as a homogeneous air-water mixture, with pressure depending on the compression ratio (barotropic).

## 5.16 MPP

1. Added keyword **\*CONTROL\_MPP\_DECOMPOSITION\_TRANSFORM\_KEYWORD**. Adding this keyword will cause all **\*CONTROL\_MPP\_DECOMPOSITION\_TRANSFORMATION** instantiations to apply not just to the primary decomposition region, but to all the specific decomposition regions created by other **\*CONTROL\_MPP\_DECOMPOSITION\_...** keywords. Note that this feature has been available since R12 through the pfile command `decomposition { transform_keyword }`.
2. Added keywords **\*CONTROL\_MPP\_DECOMPOSITION\_ADJUST\_PART\_COST** and **\*CONTROL\_MPP\_DECOMPOSITION\_ADJUST\_MTYPE\_COST**. The effect of these keywords is to alter the decomposition cost of each element of the given part or

material type. If a part appears in more than one instance of this card, the first values are kept and a warning is issued. Parts have priority over material type as they are more specific.

## 5.17 Multiscale

1. Changed the solver used in solving the unit cell model (**\*INCLUDE\_UNITCELL**) from LAPACK to a compressed-sparse-column-based sparse solver. Using the LAPACK solver in explicit analysis requires solving a large-scale inverse problem, which is both time and memory intensive. The compressed-sparse-column-based sparse solver improves efficiency for unit cell analyses in explicit.
2. Initial release of the MAPDL LS-DYNA co-simulation feature. This feature is limited to modeling solder reflow. It uses adaptive ISPG to model the solder balls. See the beta documentation for LS-DYNA on the Ansys help website for details.

## 5.18 SPG (Smooth Particle Galerkin)

1. Added **\*ISPG\_TO\_ISPG\_CONTACT** to specify sets of ISPG where the parts in each set interact with each other through contact during the solder reflow simulation process.
2. Added keyword **\*ISPG\_TRANS\_CPL** to define strong coupling between an ISPG fluid and rigid bodies. It is intended for models that involve force-driven adhesive flow or self-alignment due to strong capillary action.
3. Added keyword **\*MAT\_ISPG\_CROSS\_WLF\_MODEL (\*MAT\_ISPG\_05)** for ISPG. The Cross WLF viscosity model combines the Williams-Landel-Ferry (WLF) equation with a power-law dependence on shear rate. It is particularly useful for modeling the thermal dependence of polymers, and it captures the non-Arrhenius behavior of viscosity near the glass transition temperature.
4. Added **\*DEFINE\_SPG\_TO\_SPG\_COUPLING** to define a

coupling interface between two SPG parts.

## 5.19 SPH

1. Added **\*BOUNDARY\_SPH\_TRACK\_SURFACE** to track SPH particles on the boundary of an SPH domain. `ICONT = 1` can be used to reduce the number of particles considered candidates for nodes-to-surface contacts.

## 5.20 Thermal Solver

1. Added **\*LOAD\_HEAT\_LASER\_ABSORPTION** to simulate the heating of a translucent, heat absorbing material by a moving laser source.
2. Made sliding contacts with `AUTOMATIC_SINGLE_SURFACE` in the name available for contacts that involve thermal heat transfer (`THERMAL` and `THERMAL_FRICTION` keyword options).

## 5.21 Miscellaneous

1. Including the command line option `sharecrv` causes the raw load curve data, which is the same on all processors, to be put into shared memory on each compute node. Doing this reduces the amount of memory required for execution. We recommend using this command line option only if necessary, as it might have a slight impact on speed. This feature is only available in the MPP version and is not supported with PlatformMPI.
2. Added keywords **\*HASH** and **\*HASH\_END** for generating a checksum of a portion of the input. This checksum provides an easy confirmation that a block of input has not been modified. It is included in a table at the bottom of `d3hsp`. This table also includes hashes for all the input files and some of the generated output files. Note that these keywords require using the hash command line option to have an effect.

3. Added **\*DEFINE\_TWINBUILDER\_COUPLING** to define a node/node set that interacts with an Ansys Twin Builder™ ROM model. This keyword is still under development.
4. Made coupling the LS-DYNA solver with the Ansys Fluent® solver available through the Ansys System Coupling™ software. This feature is beta and still under development. See **\*COSIM\_SYC\_CONTROL** and **\*COSIM\_SYC\_INTERFACE**.

## 6 Improvements and Enhancements

### 6.1 Acoustics, NVH, and Frequency Domain

1. Added acoustic intensity in directivity output (AC\_intensity\_directivity\_n) invoked with **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_DIRECTIVITY**.
2. Added load type VAD = 15 to **\*FREQUENCY\_DOMAIN\_SSD** to provide a rotating structure with a mass imbalance as the excitation input.
3. Support using **\*FREQUENCY\_DOMAIN\_PATH** for reduced FRF (**\*FREQUENCY\_DOMAIN\_FRF\_REDUCED**).
4. Support outputting the acoustic fringe plot and directivity plot for variational indirect BEM (METHOD = 2) when RESTRT = 1 on **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM**.
5. Enabled computing the acoustic radiated power with **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM\_POWER** using variational indirect BEM (METHOD = 2).
6. Support RESTRT = -3 and -30 for the Rayleigh method (METHOD = 0 on **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM**).
7. Support the fringe plot (**\*FREQUENCY\_DOMAIN\_ACOUSTIC\_-**

- FRINGE\_PLOT**) including field on the boundary of the boundary element mesh for collocation BEM (METHOD = 3 on **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM**).
8. Support the reflection coefficient, RC, on **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM** for the Rayleigh method (METHOD = 0) and collocation BEM (METHOD = 3).
  9. Added the ability to specify the nodal normal velocities in the user-provided files **bevel** and **bevelf** instead of the velocity components. In the first line of these files, the third parameter is now NORM which flags the kind of velocity in the file. NORM = 0 means that the velocity components are provided, while NORM = 1 means the normal velocity is provided.
  10. Support **restrt = -3** (using **bevelf**) and **-30** (using **bevelf.lsd**) on **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM** for collocation BEM with the Burton-Miller formulation (METHOD = 4 on **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM**).
  11. Support acoustic directivity (**\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM**) output for collocation BEM (METHOD = 3 on **\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM**).

## 6.2 Adaptivity

1. Added in-core adaptivity support when ADPOPT is defined in Card 2 of **\*PART**. Parts with ADPOPT = 0 will be skipped when INMEMORY = 1 in **\*CONTROL\_ADAPTIVE**, consistent with the adaptive behavior when INMEMORY = 0.
2. Support in-core adaptivity with tabulated friction coefficients (FS = -2 on Card 2 of **\*CONTACT\_...**) in MPP, where the friction coefficients can be functions of pressure and relative velocity at the contact interface.

## 6.3 Airbags

1. When vents are defined in **\*AIRBAG\_CPG**, skip applying coupling forces to vent elements. If PPOP is used, coupling forces are applied to vent elements until the vent is marked as open.
2. Made a change to the evaluation of porosity curves defined in **\*MAT\_FABRIC** when associated with **\*AIRBAG\_CPG**. Previously, porous velocity was reset to zero if the pressure was out of bounds of the provided porosity curve. We now use the last available slope to extrapolate the porous velocity, similar to the behavior of **\*AIRBAG\_PARTICLE**. Ideally, the porosity curve should include the entire range of pressures expected in the simulation.
3. Added capability to automatically refine the particle size around inflators and vents. See HLENV and HLENO on **\*AIRBAG\_CPG**.
4. Added shear friction force to structure coupling for CPG airbags.
5. Added part output of heat transfer and heat transfer rate in binout for CPG airbags.
6. For **\*AIRBAG\_CPG**, added bag integrity checks to see if the external mesh is closed and normal vectors are consistent.
7. Added PPOP to **\*AIRBAG\_CPG** to open the vents and keep them open based on a pressure differential.
8. Added an error message if the fabric porosity option, FVOPT, on **\*MAT\_FABRIC** is not 7 or 8 when used with CPG airbags (**\*AIRBAG\_CPG**).
9. Added support for the FLC porosity option define in **\*MAT\_FABRIC** for CPG airbags.
10. Allow the density to be defined as a function of time ( $RO < 0.0$ ) for **\*AIRBAG\_LINEAR\_FLUID**.
11. Added the ability for **\*SENSOR\_DEFINE\_FORCE** with FTYPE = CPM to return either the average airbag pressure or the part pressure. Previously, only the average airbag pressure was returned. Now, the part pressure can be returned instead by setting field CPMPID to a part ID.

12. Support specifying HCONV on **\*AIRBAG\_PARTICLE** or **\*DEFINE\_CPM\_Npdata** with a **\*DEFINE\_CURVE\_FUNCTION**.
13. Support specifying the total mass flow rate into a CPM airbag (**\*AIRBAG\_PARTICLE**) with a **\*DEFINE\_CURVE\_FUNCTION** using LCMCF on **\*DEFINE\_CPM\_GAS\_PROPERTIES**. Note that LCMCF cannot be used when using the MOLEFRACTION keyword option for **\*AIRBAG\_PARTICLE**.
14. For **\*AIRBAG\_PARTICLE** opening a vent with PPOP is based on the instantaneous part pressure which is very noisy and triggers venting unexpectedly. To remove the noise, a moving average scheme is now used, leading to a more reliable signal.
15. Support FVOPT = 1 and 2 on **\*MAT\_034** for CPM airbags (**\*AIRBAG\_CPM**).

## 6.4 ALE

1. Support **\*CONSTRAINED\_EULER\_IN\_EULER** for S-ALE.
2. When using **\*ALE\_COUPLING\_NODAL\_...** to couple ALE materials with SPH particles generated by **\*DEFINE\_ADAPTIVE\_SOLID\_TO\_SPH**, the coupling is disabled for inactive embedded SPH nodes inside the solid elements.
3. Added OPTION SALEBOXL to **\*SET\_NODE/SEGMENT/SOLID/SHELL\_GENERAL** to generate nodes/segments/solids/shells inside a box in S-ALE local coordinates.
4. Support the NLPTC option for **\*DATABASE\_BINARY\_FSILNK**.
5. Calculate and report erosion energy for failed S-ALE materials.
6. Made improvements to the S-ALE volume filling by part/segment set algorithm:
  - to deal with cases where the Lagrangian structure's mesh is coarse, and
  - process special cases where the Lagrangian segment's coordinates

are exactly the same as the S-ALE nodes.

7. Force dump the d3sale database at the same interval as d3plot, so you can post-process the d3plot/d3sale combinations during the run.
8. Implemented keyword option AXIVOL for **\*ALE\_STRUCTURED\_-MULTI-MATERIAL\_GROUP** to use the volume-weighted axisymmetric ALE 2D formulation (ELFORM = 15 in **\*SECTION\_-ALE2D**).
9. Added advection method Van Leer with Half Index Shift modified to conserve total energy over each advection step invoked with METH = 4 on **\*CONTROL\_ALE**.
10. Added INITRNS = 1 to **\*CONTROL\_REFINE\_ALE2D** to initially refine a mesh region with mesh transitions.
11. Support **\*SET\_SOLID\_ADD** and **\*SET\_SHELL\_ADD** for S-ALE.
12. Implemented **\*SET\_SHELL\_INTERSECT** for S-ALE 2D meshes.
13. Added flag ICUT to **\*INITIAL\_VOLUME\_FRACTION\_GEOMETRY** for CNTTYP = 1 and 2 to activate computing the exact volume instead of an estimate when filling ALE elements cut by the Lagrangian structure.
14. Support computing lighting times of explosive S-ALE elements in a set with **\*INITIAL\_DETONATION\_SET**. Previously, **\*INITIAL\_DETONATION** was supported for S-ALE but not with the SET keyword option.
15. Implemented **\*ALE\_BURN\_SWITCH\_MMG** for S-ALE.
16. Exclude segments loaded by **\*LOAD\_SEGMENT** from the sliding conditions built along mesh boundaries for EBC = -2 on **\*CONTROL\_ALE**.
17. Support prescribing motion of S-ALE groups with **\*ALE\_PRESCRIBED\_MOTION**.
18. For **\*BOUNDARY\_SALE\_MESH\_FACE** with OPTION =

- AMBIENT, added the ability to specify the “reservoir” material, internal energy load curve, and the relative volume load curve.
19. Added `BCTYPE = PRES` to `*BOUNDARY_SALE_MESH_FACE` to apply time-dependent pressures along an S-ALE mesh boundary face.
  20. Added `BCTYPE = FLOWVEL` to `*BOUNDARY_SALE_MESH_FACE` to apply velocities along an S-ALE mesh boundary face.
  21. For `*CONSTRAINED_LAGRANGE_IN_SOLID`, interpolate coupling forces at middle edge and face center nodes for higher-order elements, namely, 40-node pentahedra, 20-node tetrahedra, 8-node shells, 6-node shells, 21-node pentahedra, 15-node tetrahedra, 64-node hexahedra, 10-node tetrahedra, 27-node hexahedra, and 20-node hexahedra.
  22. Automatically replace `*ALE_MAPPING` when `RW = 1` and `TBEG = 0` with `*INITIAL_ALE_MAPPING`. The behavior in the simulation stays the same.
  23. Added writing the mapping file with `*ALE_MAPPING` if the velocity at node ND is nonzero.
  24. Updated the approach for computing 3D volume fractions with using `METHOD = 1` on `*ALE_MAPPING_FROM_LAGRANGIAN`.
  25. Support `*ALE_MAPPING_FROM_LAGRANGIAN` in 2D. For this case, volume fractions and other data from 2D Lagrangian elements are written to a mapping file that can be used to initialize 2D ALE meshes.
  26. Support the mapping file produced from `*ALE_MAPPING_FROM_LAGRANGIAN` in S-ALE.
  27. Support initializing S-ALE FSI (`*ALE_STRUCTURED_FSI`) from a mapping file with `*INITIAL_ALE_MAPPING`.
  28. Support mapping the penalty penetrations from `*CONSTRAINED_LAGRANGE_IN_SOLID` in a 2D to 2D mapping with `*INITIAL_ALE_MAPPING`.

29. Support mapping coupling penetrations from **\*CONSTRAINED\_LAGRANGE\_IN\_SOLID** that come from several mapping files using **\*INITIAL\_ALE\_MAPPING** with `map2=`.
30. Support producing mapping data with **\*ALE\_MESH\_INTERFACE** for S-ALE:
  - Map S-ALE data to Lagrangian tetrahedrons
  - Map S-ALE 2D data to Lagrangian triangles
  - Output `ale[re]meshmat[int,vol].k` for S-ALE 3D runs
  - Output `ale[re]meshmat[int,vol].k` for S-ALE 2D runs
31. Added `N = 2` to `NOWRT` of **\*ALE\_MESH\_INTERFACE** to write a hexahedral mesh in 3D or quadrilateral mesh in 2D for selected materials.

## 6.5 Battery Cards (Electrochemistry Solver)

1. Increased the limit on the `NCYCLE` parameter of **\*BATTERY\_ECHEM\_CONTROL\_SOLVER**, and put out an error message if the maximum value (now 100) is exceeded.

## 6.6 Blast

1. Added smoothing factor to smooth out the blast load from the particle blast method (PBM). To use this, set `SMOOTH` on **\*DEFINE\_PARTICLE\_BLAST** to a value greater than 1. The unsmoothed blast load is used if the factor is the default value of 1.

## 6.7 Boundary Cards

1. Support specifying prescribed displacement constraints using **\*BOUNDARY\_SPC\_NODE** or **\*BOUNDARY\_SPC\_SET** in a cylindrical or spherical coordinate system. Previously, only the global Cartesian system was supported.

2. Support defining translational prescribed motions specified with **\*BOUNDARY\_PRESCRIBED\_MOTION** in a cylindrical or spherical coordinate system.
3. Added birth time flag to **\*BOUNDARY\_PRESCRIBED\_FINAL\_GEOMETRY**. This feature enables the definition of sequential multilinear paths in one single run.
4. Implemented a more efficient hydrodynamic wave effect solver (**\*BOUNDARY\_MCOL**).

## 6.8 Constraint Cards

1. Added flag SKPCNRB to **\*CONTROL\_CONSTRAINED** to skip the stress calculation of a part based on element type if the part is part of a nodal rigid body (**\*CONSTRAINED\_NODAL\_RIGID\_BODY**).
2. For **\*CONSTRAINED\_SOIL\_PILE**:
  - Connection points are now switched off automatically following failure of the soil solid element or the pile beam element.
  - Damping of the perpendicular motion can now be specified (see DAMPH).
3. Support beam resultant output (forces and moments) to the elout database for **\*CONSTRAINED\_INTERPOLATION\_SPOTWELD MODEL = 3** (SPR3 with strongly objective formulation) connector beams.
4. Made improvements to SPRSRCH = 1 on **\*CONTROL\_CONSTRAINED**. Better search results can be obtained, especially when solid element parts are involved. For solid element parts, nodes on the surface facing the SPR connector are detected, leading to automatically finding enough nodes.
5. For SPR2 and SPR3, added option to modify the nodal weights for force distribution. Setting the tens digit of INTP to 1 causes nodal areas to be considered and adds a participation factor for better consideration

- of distances between SPR and the associated nodes.
6. Added two more arguments to the **\*DEFINE\_FUNCTION** definition for SPR3 (**\*CONSTRAINED\_INTERPOLATION\_SPOTWELD**): normal and tangential velocity.
  7. Added three new output quantities for SPR3 to **binout/swforc**: “velocity,” “velocity\_axial,” and “velocity\_shear.”
  8. Added field PID to **\*CONSTRAINED\_TIED\_NODES\_FAILURE**. With  $PID > 0$  and  $NSID = 0$ , the mesh of the corresponding part is automatically separated (creating new coincident nodes) and reassembled using this tied constraint with failure. If some original nodes are part of node sets, such as for boundary conditions, these sets get automatically extended as well. This is currently only supported for shell elements ( $ETYPE = 0$ ).

## 6.9 Contact

1. Made improvements to MPP single surface (**\*CONTACT\_AUTOMATIC\_SINGLE\_SURFACE**) contact and penetration tracking that reduce force spikes in difficult meshes.
2. Detect empty shell sets being used in **\*CONTACT** definitions, and flag these as errors during keyword processing.
3. Made modifications to the MPP tiebreak code for **\*CONTACT\_AUTOMATIC\_...\_TIEBREAK** with  $OPTION < 0$ . The behavior here now agrees well with SMP and with **BEAM\_OFFSET** tied contact.
4. Added some mesh checking for solid elements during contact input processing. When contact is defined using solid parts, the process that determines the outside surface (and the interior segments for erodingcontact) now checks for problems and error terminates when nonphysical solid meshes are input (for which the “outside” is indeterminate). In particular, it checks for cases where elements are duplicated, or connected in impossible ways (such as two solid elements being on the same side of a common face). Note that older versions of the code did not do these checks, and so would run models

which might now be rejected. These checks are not optional and cannot be turned off because they expose actual physical mesh errors that should be corrected to get correct results.

5. Added MPP contact support for **\*CONTACT\_TIED\_SHELL\_EDGE\_TO\_SOLID\_CONSTRAINED\_OFFSET**.
6. Improved handling of nodes that are on the reference side of constraint-based tied contact interfaces while also being constrained with **\*INTERFACE\_LINKING\_...** For MPP non-groupable contact, this is already being handled well. But for groupable contact, all reference nodes were being removed from the LINKING interface, even if they did not have any tracked nodes tied to them. Now only those reference nodes actually having constraints due to tracked nodes being tied to them will be removed from the linking interface.
7. For **\*CONTACT\_AUTOMATIC\_...\_TIEBREAK**, force automatic surface reorientation for these contacts.
8. Fixed failure to tie the surfaces when using the MPP version of the **\*CONTACT\_AUTOMATIC\_SURFACE\_TO\_SURFACE\_TIED\_WELD** contact in the case where the temperature is driven by **\*LOAD\_THERMAL\_LOAD\_CURVE**.
9. Added support for contact option TCSO (optional Card D) to the MPP versions of **SINGLE\_SURFACE**, **AUTOMATIC\_SINGLE\_SURFACE**, and **AIRBAG\_SINGLE\_SURFACE**.
10. In constraint-based tied contacts, massless nodes are now allowed on the tracked side of the contact (SURFA).
11. Modified segment-to-segment (SOFT = 2) contact. A problem was observed with edge-to-edge contact checking when the sliding option (SBOPT = 4 or 5) was active. Segments were not developing contact forces, so edges were penetrating other edges. Since the sliding option was developed for surface sliding, a decision was made to skip the sliding option logic when a contact pair was contacting edge-to-edge in the previous cycle. This change will affect results when the sliding option is used with DEPTH = 5, 15, 25, 35, or 55.

12. Modified the bucket sort behavior of segment-to-segment (SOFT = 2) eroding contact. The default bucket sort frequency is still 10, but if the number of cycles (BSORT on **\*CONTACT\_...** or NSBCS on **\*CONTROL\_CONTACT**) is set to a value larger than 10, then the larger value is used. However, if a new contact segment is added due to the deletion of a solid or thick shell element, then another bucket sort is added in no more than 5 cycles. It will be less than 5 cycles if the scheduled bucket sort is less than 5 cycles ahead, but 5 cycles if the scheduled cycle is more than 5 cycles ahead. In the latter case, the next scheduled cycle will be set to BSORT or NSBCS cycles after the added bucket sort.
13. Added parameter IBEAMS2S to optional Card G of **\*CONTACT\_ - ...**. If you want to have beam segments included in a segment-to-segment (SOFT = 2) contact interface, include the beam part or parts in the contact surface and set IBEAMS2S = 1. If IBEAMS2S is not set or is set to 0, the beam segments will not be created and beams will not be checked for contact. In other words, the default behavior is to omit beams from contact.
14. Tuned segment-to-segment (SOFT = 2) contact to make it more accurate. This improvement enables rigid body translation at any distance from the origin without producing contact energy.
15. Improved beam contact in single-surface segment-to-segment (SOFT = 2) contact by optimizing the code that removes neighbor beams from consideration. This should make the contact faster and a less like to generate spurious energy with flexible beams such as when fabric fibers are simulated.
16. Implemented asynchronous inter-processor communication for Mortar contact. Communication between processors in Mortar contact is now initiated as early as possible and overlapped with local computation, using non-blocking MPI operations. This potentially reduces idle time and improves scalability for large parallel jobs.
17. Map user-defined friction and wear history variables for Mortar contact across adaptive remeshing steps. When *r*-adaptive remeshing (mesh rezoning) is combined with Mortar contact, wear depth/history and user-defined friction history variables are now correctly interpolated from the

- old mesh to the new mesh at each rezoning step. Previously, all contact history was discarded at each rezone, causing an incorrect contact state during the remainder of the simulation.
18. Support nonzero birth time (BT on Card 2 of **\*CONTACT\_...**) for all non-groupable implicit tied contacts.
  19. For implicit tied contacts, improved the tied contact logic so that stacked plies with constrained tied contacts in between are not subjected to release, provided that there is no dependency when viewed as a fully coupled system.
  20. Made a slight modification to the output value for “contact gap” in the intfor database for **\*CONTACT\_AUTOMATIC\_...\_TIEBREAK** with **OPTION = 13** or **14**. This value not only shows damage, but also a small margin to damage initialization:  $\text{contact gap} = 0.99 \times \text{damage} + 0.01 \times \min(1, \delta_m/\delta_{m1})$ .

## 6.10 Control Cards

1. Added option **MAEF = 2** on **\*CONTROL\_MAT**. In addition to turning off **\*MAT\_ADD\_EROSION** or **\*MAT\_ADD\_DAMAGE\_DIEM/GISSMO** globally (**MAEF = 1**), this also deactivates failure in materials 24, 103, 105, 106, 114, 123, 124, 155, 195, 225, 251, and 255 by setting **FAIL = 0** internally for all corresponding **\*MAT** cards in a model.
2. Added field **PSMAEF** to **\*CONTROL\_MAT**. **PSMAEF** is a part set ID that specifies which parts the deactivation of failure/damage with **MAEF** applies.

## 6.11 Dual CESE Compressible Fluid Solver

1. Improved the bottom flux calculation at the FSI-IBM interface, leading to better stability.
2. Added load curve support for the dual CESE porous media capability (see **LC\_KP** and **LC\_BETA** on **\*DUALCESE\_DARCY-FORCH-**

- HEIMER\_EQ**). This gives you a way to obtain a variable permeability for the porous media that is immersed in the dual CESE fluid. This feature is available in the single-fluid IBM-FSI, two-phase IBM-FSI, and chemically-reacting flow IBM-FSI solvers.
3. Added the restart capability for all dual CESE solvers. With this, large and long-running dual-CESE-based applications can be restarted from restart files created at intermediate times during a simulation.
  4. Using the REFPROP EOS is available by requesting the required support files from ACE.
  5. Made significant reductions in internal memory use in several aspects of the dual CESE solvers. With these improvements, much larger meshes can be used on a given compute resource.
  6. Implemented an alternate method for determining the number of buckets in the bucket sort search by allowing for a scaling factor multiplying the number of fluid nodes or elements on the current processor. This scale factor is input with SFNBCKT on **\*DUALCESE\_MESH\_GEOMETRY**. A fractional number (such as 0.6) reduces the number of buckets below the number of fluid nodes or elements.
  7. Added check for invalid segments or faces in the definitions of a dual CESE segment set. These segments typically occur in surfaces used for boundary condition definitions.

## 6.12 Elements

1. Added option ITORM = 3 on **\*SECTION\_BEAM** which uses the average of the rotational inertias of ITORM = 2. This frame invariant approach is directly applicable in explicit and implicit analyses. Additionally, ITORM may be input negative to get a user-defined rotational inertia computed from the absolute value of ITORM and the nodal translation mass. Both cases may require a reduced step size for stability in explicit analysis.
2. Added the ability to specify two shear areas for the Belytschko-Schwer resultant beam (ELFORM = 2) with SAS and SAT on **\*SECTION\_**

## BEAM.

3. Zero density is now allowed in cohesive solid elements. It has always been the case that the cohesive element time step is computed using the assembled nodal mass. Therefore, the cohesive elements did not strictly need to have any density. To allow for zero cohesive density, we changed the mass scaling computation to be based on the cohesive element mass. Before, it was based on the cohesive element density.
4. Added option  $L = 3$  as the 10s digit of ICOH on **\*CONTROL\_SOLID**. The tens digit determines how to compute how much nodal mass is associated with each cohesive element. For  $L = 3$ , instead of assuming that each node is shared by two or four cohesive elements, the actual distribution of the nodal mass is computed. This leads to identical time steps at edges and inside of the cohesive surface.
5. Added check for 5-noded pyramid elements and added a warning about it.
6. Improved single precision accuracy of the internal energy computation in shell type 16.  $ISTUPD = 5$  and  $6$  on **\*CONTROL\_SHELL** invoke this improved accuracy version.
7. Made a modification to the cohesive element quality check, option COHEQC on **\*CONTROL\_SOLID**. This check is only performed for cohesive elements that have materials with  $ROFLG = 0$ .

## 6.13 EM (Electromagnetic Solver)

1. For **\*EM\_CIRCUIT**, an optional phase time can be defined in the C/T0 field if  $EMSOL = 4$  or  $9$ .
2. It is now possible to define a general contact resistance law when no **\*EM\_CONTACT** has been defined by setting  $CONTID = 0$  on **\*EM\_CONTACT\_RESISTANCE**. This is useful in self contact cases.
3. Added  $BIOT = 1$  for  $EMSOL = 3$  on **\*EM\_CONTROL** to calculate Lorentz forces in conductive beam elements using the resistive heat

solver (diffusive effects are neglected). This feature can be useful in high voltage cable applications to study their oscillation behavior in cases of strong current discharge.

4. Added BIOT = 1 and 2 for EMSOL = 1 on **\*EM\_CONTROL** to output results from a magnetostatic solve in a binary file (BIOT = 1) which can then be reread in a subsequent analysis (BIOT = 2) to do some Biot-Savart calculations and extract the magnetic flux on nonmagnetic surfaces. This feature can be useful for studying the effects of magnets or ferromagnets on their environment.
5. For EMSOL = 4 on **\*EM\_CONTROL**, it is now possible to define a complex permeability to include hysteresis effects using **\*EM\_MAT\_002**.
6. Made improvements that led to a reduction in the calculation time when using **\*EM\_DATABASE\_PARTDATA**.
7. For **\*EM\_EOS\_PERMEABILITY**, added option to have the *B-H* curves as a function of either the signed Von Misses stress or more complex laws for magnetostriction problems.
8. Support using **\*DEFINE\_CURVE\_FUNCTION** for the load curves specified in **\*EM\_ISOPOTENTIAL\_CONNECT**.
9. For **\*EM\_ISOPOTENTIAL\_CONNECT**, implemented imposing an optional phase shift if EMSOL = 4, 7, or 9 on **\*EM\_CONTROL**.
10. Add keyword **\*EM\_OUTPUT\_BINARY**. This keyword enables outputting certain EM solver results in a binary format instead of ASCII. For now, only the “em\_randles” input family and the “em\_isopot” output family are supported (for large battery cases).
11. For **\*EM\_RANGLES\_...**, added support for **\*DEFINE\_CURVE** in the definition of R0CHA, R0DIS, etc. when a negative value is entered. LS-PrePost seemed to have assumed that this was already supported (which was causing errors and confusion for users).
12. Added STYPE = 3 to **\*EM\_SOLVER\_FEM** to invoke a hybrid approach between the direct solver and the iterative approach. At the initial time, a direct solve is done and factorization is kept. Subsequent

- solves either use the direct solver (if entry matrices have not changed) or switch to an iterative conjugate gradient approach if they changed unless convergence is not achieved in which case a new factorization and direct solve is attempted. This approach saves significant computational time in large battery models that use temperature-dependent Randles circuits.
13. The EM solver supports the **\*INCLUDE\_TRANSFORM** offsets specified in Card 2b.1.
  14. Added EM functions EM\_ISO HIST and EM\_ISO CHIST to **\*DEFINE\_CURVE\_FUNCTION** to access the global quantities calculated by **\*EM\_ISOPOTENTIAL** and **\*EM\_ISPOTENTIAL\_CONNECT**, respectively.
  15. In the 2D axisymmetric Eddy current solver, the Lorentz force calculation now better includes the element deformation, resulting in higher accuracy.
  16. Optimized the FEM and BEM matrix assemblies, resulting in reduced calculation times.
  17. Optimized the internal operation that calculates the current density, resulting in a slight reduction in calculation times (visible mostly in large models that use FEM-based solvers).
  18. Improved the overall robustness of the EM solver when using FEM-based solvers. We added automatic criteria in the FEM system solve that check whether the incoming matrices have changed. If they have changed, a refactorization is automatically done if the direct solver is used. We made this change after noticing that using birth times or death times for boundary conditions led to incorrect results unless the birth/death times coincided with a FEM matrix reassembly.
  19. We now take the average of the interaction between face A against face B and face B against face A when assembling the symmetric BEM matrices ( $Q$ ,  $W$ , ...), rather than simply the first one that was calculated (for example, face A against face B). This makes the result more consistent in case of different element orderings as round off errors depending on the order of calculations could introduce micro changes in

- results. As a result, this new approach may cause some small differences in results compared to previous versions but should overall be more consistent.
20. Randles circuit output files can now have IDs over 999999.
  21. Changed the way Joule heating is calculated. We are now directly using the scalar product of the current density and electric field rather than the squared electric field divided by the conductivity. We made this change because the accuracy was poor in areas that had elements with high conductivity right next to elements with poor conductivity. The new way should overall not introduce large changes in thermal coupling problems, but it is possible that results differ slightly when compared with previous versions (and should offer higher accuracy).
  22. Modified the d3plot for the frequency-based solvers (EMSOL = 4, 7, or 9 on **\*EM\_CONTROL**): complex vectors are now directly output. To read them properly, use a LS-PrePost version 4.13 or later.
  23. Always output the magnetic surface force to d3plot. Previously it was only output when permanent magnets were present.
  24. Added output of EM data on nodes (**\*EM\_DATABASE\_NODOUT**) for the pure Eikonal solver (EMSOL = 14 on **\*EM\_CONTROL**)
  25. Support the RE+ solver (see EMSOL = 16 on **\*EM\_CONTROL**) for the ECG computation (**\*EM\_EP\_ECG**).
  26. Added stimulus on beams in the multifront time stepping Eikonal solver (**\*EM\_EP\_EIKONAL**).
  27. Added solvers RE- and RE+ (EMSOL = 15 and 16 on **\*EM\_CONTROL**) to the capability of solving the external potential,  $\phi_e$ . Previously, this was only supported for the monodomain solver.  $\phi_e$  is solved when LAMBDA is defined in **\*EM\_MAT\_003** or **\*EM\_MAT\_005**. Note that for **\*EM\_MAT\_005** this feature is only supported for EMSOL = 15 and 16 and not for the monodomain solver.
  28. Added capability to model the electromagnetic properties of tissue using **\*EM\_MAT\_001** with MTYPE = 2. This feature is only

supported for the EP monodomain solvers (see EMSOL = 11 and 13 on **\*EM\_CONTROL**). Note that the tissue must not have a fiber orientation.

29. Support **\*EM\_EP\_CELLMODEL\_DEFINE\_FUNCTION** and **\*EM\_EP\_CELLMODEL\_TROVATO** for the spline ionic solver (IONSOLVR = 2 on **\*EM\_CONTROL\_EP**) when using Eikonal solvers EPSOL = 14, 15, or 16.

## 6.14 Forming Analysis

1. Added field IDRAW to **\*CONTROL\_FORMING\_ONESTEP\_AUTO\_CONSTRAINT** to specify or let the solver approximate the draw direction. In the one step method, if the panel is put at an arbitrary orientation, the boundary conditions might be different and the obtained results can vary. This feature leads to more consistent results.
2. Added the ability to provide the pad as a reference for positioning the final blank with **\*CONTROL\_FORMING\_ONESTEP\_AUTO\_CONSTRAINT**. The pad is specified with PADID and must be flat. The unfolded blank is repositioned such that the (x,y) mass centers match.
3. Made the results from the one-step solution method (**\*CONTROL\_FORMING\_ONESTEP**) more consistent when using GISSMO (**\*MAT\_ADD\_DAMAGE\_GISSMO**) for the damage model.
4. Made the following improvements in stamping simulations that use solid elements:
  - enable auto constraints for springback analysis and
  - enable constraints on coordinates.
5. Added the ability to include the effect of a post in springback compensation with **\*INTERFACE\_COMPENSATION\_3D**. The information on Card 1a.3 (VX, VY, VZ, and GAP0) provides the direction the blank would need to move to be on the post and the gap between them, so that the gap can be removed. With this the correct springback value can be used to compensate the die. If springback is

performed using single point constraints, this card is not needed.

## 6.15 ICFD (Incompressible Fluid Solver)

1. Added support for adaptive meshing when using the RIIS immersed method ( $IMMTH = 2$  and  $4$  on **\*ICFD\_CONTROL\_IMMERSED**).
2. Improved immersed interface detection using structural normals and fluid edge intersections.
3. Enhanced accuracy of force computation for immersed methods, including beam element fixes and structural side detection.
4. Improved mapping algorithms after remeshing for robustness and efficiency.
5. Simplified the setup and usage of the RIIS immersed method using DEM particles for the interface ( $IMMTH = 2$  on **\*ICFD\_CONTROL\_IMMERSED**). This method no longer requires **\*ICFD\_CONTROL\_DEM\_COUPLING**. Instead it uses **\*ICFD\_CONTROL\_IMMERSED\_FSI** like all the other immersed methods.
6. Added keyword option **CSYS** to **\*ICFD\_DATABASE\_DRAG** to define a local reference frame in which the forces will be output. This ability can be useful in certain categories of problems, such as container dropping or rotating problems.
7. Made it possible to dynamically turn on and off the Navier Stokes solve using  $TDEATH < 0.0$  on **\*ICFD\_CONTROL\_TIME**. If less than  $0.0$ ,  $|TDEATH|$  refers to a load curve ID for the curve giving an indicator as a function of time. A positive value for the indicator means the Navier Stokes solve is on, while a negative value means it is off. Equations for the mesh movement and temperature are still solved, regardless of the indicator. This feature can be useful in various scenarios, such as transient conjugate heat transfer where the fluid may no longer vary significantly after a certain time or conversely is only turned on after a certain birth time.
8. Added the Giesekus model ( $VTYPE = 2$ ) to **\*ICFD\_MODEL\_VIS-**

## COELASTIC.

9. Enabled specifying an initial level set surface with a function (**\*DEFINE\_FUNCTION**) using `STYPE = 5` on **\*ICFD\_INITIAL\_LEVELSET**.
10. Added the ability to include two materials in a volume for level sets with **\*ICFD\_PART\_VOL**.
11. Added support for a more general way to define the residence time injection region with `ISHAPE = 4` on **\*ICFD\_DEFINE\_RESIDENCETIMESOURCE**. This method involves specifying one or more infinite planes that must result in a concave region. It can be combined with other source shapes and allows for nesting and combining them through a set of union operations.
12. Added `VAR = 10` to **\*ICFD\_CONTROL\_ADAPT** to use species concentrations and a combination of species concentrations and the velocity field in the error calculation for species transport problems.

## 6.16 Implicit (Mechanical) Solver

1. Upgraded MUMPS (`LSOLVR = 30` on **\*CONTROL\_IMPLICIT\_SOLVER**) from version 5.7.3 to 5.8.1.
2. The LSDA database for the output of eigenvectors on a node set (see `IPARM3` for `EIGMTH = 2, 3, 5, 6, 102, and 103` on **\*CONTROL\_IMPLICIT\_EIGENVALUE**) is now named `EigenVectors` for `EIGMTH = 2, 3, 5, 6, 102, and 103`. Previously, the file name depended on the value of `EIGMTH` (`LanczosEigenVectors` for `EIGMTH = 2, 3, 5, and 6`, `LOBPCGEigenVectors` for `EIGMTH = 102`, and `FastLnzEigenVectors` for `EIGMTH = 103`).
3. Added error checking for the nodes in the node sets for constraint and attachment modes in **\*CONTROL\_IMPLICIT\_MODES**. Warning messages for bad nodes (e.g., massless nodes) are issued, and the nodes are removed from the computation.

## 6.17 Initial Cards

1. Implemented TBEG on **\*INITIAL\_LAG\_MAPPING** to control the start time of **\*INITIAL\_LAG\_MAPPING**.

## 6.18 Isogeometric Analysis (IGA)

1. Support using **\*IGA\_POINT\_UVW** with VAD1 = 12 (torsion) in **\*FREQUENCY\_DOMAIN\_FRF**.
2. Support applying a pressure excitation load on parametric faces (**\*IGA\_FACE\_UVW**) defined on outer surfaces of untrimmed solids with **\*FREQUENCY\_DOMAIN\_FRF** and **\*FREQUENCY\_DOMAIN\_SSD**.
3. Support imposing boundary conditions on the outer surfaces of untrimmed solid elements with **\*BOUNDARY\_PRESCRIBED\_MOTION\_FACE\_UVW** and **\*BOUNDARY\_PRESCRIBED\_MOTION\_SET\_FACE\_UVW**.
4. Support defining multiple parts on a single unstructured patch. The LIST and RANGE options can be used to list local element IDs in **\*IGA\_FACE\_XYZ** and **\*IGA\_VOLUME\_XYZ** for each part.
5. Enabled output of information for IGA solid elements to **matsum**.

## 6.19 Load Cards

1. For **\*LOAD\_BODY\_...**, moved keyword data out of the structured file and into LSDA, to prevent loss of precision, particularly in the scale factor value.
2. Support **\*LOAD\_BEAM** for discrete beam elements (ELFORM = 6 on **\*SECTION\_BEAM**). The load, which is evenly distributed to the beam nodes, is then interpreted as force and not force per unit length.
3. Support defining nodal forces applied with **\_NODE** in a cylindrical or spherical coordinate system.

## 6.20 Materials and Equations-of-State

1. For **\*MAT\_ADD\_DAMAGE\_DIEM**, **\*MAT\_ADD\_DAMAGE\_GISSMO**, and **\*MAT\_ADD\_EROSION**, **NUMFIP < 0** now also applies to beam elements.
2. Made a minor change for **GISSMO** output with **\*DEFINE\_MATERIAL\_HISTORIES**. Label “Damage” with attribute **A1 = 2** gives alternative damage parameter (history variable # **ND + 13**).
3. Added option **FFUNC** to **\*MAT\_ADD\_EROSION** to define a freely selectable failure criterion using **\*DEFINE\_FUNCTION**. Arguments to the function include stresses, strains, strain rates, temperature, etc.
4. For **\*MAT\_ADD\_EROSION**, **LCFLD** can now refer to a **\*DEFINE\_TABLE\_3D** to define major strain as a function of temperature (**TABLE\_3D**), strain rate (**TABLE**), and minor strain (**CURVE**).
5. Added option **LCREG < 0** to **\*MAT\_ADD\_GENERALIZED\_DAMAGE**. **|LCREG|** refers to a table that contains regularization factor as a function of element size curves for different triaxialities.
6. Added flag **ILC** to enable the Norton-Bailey creep law material constants included when **LAW = 5** with **MODEL = 3** or **LAW = 6** with **MODEL = 4** to be specified as load curves giving the constants as functions of temperature.
7. Turned off pore air flow (**\*MAT\_ADD\_PORE\_AIR**) simulation during dynamic relaxation.
8. For **\*MAT\_ADD\_PZELECTRIC**, the electric voltage of nodes connected via tied contacts is now calculated by interpolating the voltage values of the segment they are attached to.
9. Added capability to use 3D piezo properties (**\*MAT\_ADD\_PZELECTRIC**) directly in axisymmetric analyses, eliminating the need for manual property transformations.
10. Fixed bug affecting the output of the electric field and flux from

- \*MAT\_ADD\_PZLECTRIC** during implicit analysis. The bug led to incorrect results.
11. Implemented reduced integration for the piezoelectric material (**\*MAT\_ADD\_PZLECTRIC**) stiffness matrix.
  12. Added the option to define **\*MAT\_ADD\_THERMAL\_EXPANSION** per material instead of per part.
  13. Added parameter TREF to **\*MAT\_ADD\_THERMAL\_EXPANSION**. By default, the tangent coefficient of thermal expansion is used, meaning instantaneous change in length per degree of temperature change. TREF activates the secant coefficient of thermal expansion, meaning average change in length per degree of temperature change over a temperature range, relative to a reference temperature. This is similar to the choice of LCALPH > 0 or LCALPH < 0 in **\*MAT\_106**.
  14. Capped the exponential term in **\*MAT\_JOHNSON\_COOK** (**\*MAT\_015**) when VP = 1.0 to avoid random crashes.
  15. Added the ability to provide the plastic strain term,  $A + B \times \bar{\epsilon}^{pn}$ , in the Johnson-Cook (**\*MAT\_015**) flow stress formula as a load curve that depends on the effective plastic strain,  $\bar{\epsilon}^p$ . This is only available for solid elements (including SPG). It is available for both no keyword option and the STOCHASTIC keyword option. It is also available for both the rate-independent and viscoplastic (VP = 0 and 1) formulations.
  16. Made the Cockcroft-Latham damage models implemented for **\*MAT\_107** available for **\*MAT\_015** (see DMODEL).
  17. Implemented multi-directional failure (see MDFAIL) in **\*MAT\_196**.
  18. Implemented flag BNDLCO in **\*MAT\_RIGID** and option IPRT < 0 in **\*CONSTRAINED\_NODAL\_RIGID\_BODY** so that a rigid body's forces and moments written to bndout are in the body's local coordinate system.
  19. Added check of anisotropic, elastic material parameters (Young's moduli and Poisson's ratios) in **\*MAT\_021** for consistency. This check is already performed for most of the other anisotropic materials.

20. Added option  $VP = 4.0$  to **\*MAT\_024** to invoke filtering/smoothing of the plastic strain rate. It is like  $VP = 1.0$  but supports **\*DEFINE\_TABLE\_COMPACT** like  $VP = 3.0$ .
21. Added compressive plasticity and Drucker-Prager based plasticity to **\*MAT\_030**/**\*MAT\_SHAPE\_MEMORY** (see IDPP). Also added support for total strain-stress curves for experimental input data.
22. Added table evaluation flag, TFLG, to **\*MAT\_036E**. If  $TFLG = 0.0$ , tables LCH00/45/90 depend on strain and strain rate (default). If  $TFLG = 1.0$ , tables LCH00/45/90 depend on strain and history variable #12 or can be 3D tables that depend on strain, strain rate, and history variable #12. History variable #12 is an arbitrary user-defined variable that can be set with **\*INITIAL\_STRESS\_SHELL** or **\*INITIAL\_HISTORY\_NODE**. Additionally, the effective plastic strain rate is output as history variable #6.
23. For **\*MAT\_036\_NLP**, changed the second history variable to maximum failure. Previously the strain ratio was output.
24. Added a general Blatz-Ko strain energy functional model and the Ciambella model to **\*MAT\_038**. See flag IMODEL to change the model.
25. Allow the combination of **\*MAT\_077\_O** with  $N = -1$  (thermal option) and solid element formulation 13. This was not previously supported.
26. Enabled **\*MAT\_091** ( $REF = 1$ ) to be used with **\*INITIAL\_FOAM\_REFERENCE\_GEOMETRY**.
27. Allow logarithmic interpolation in some  $E$  (Young's modulus) as a function of strain rate curves. This is available for  $LCEMOD > 0$  in **\*MAT\_123**, **\*MAT\_187**, and **\*MAT\_187L**. If the first strain rate is negative, all values are expected to represent the natural logarithm of a strain rate.
28. Support **\*MAT\_125** for 2D implicit analysis.
29. Applied strain rate factor, RFAC, to LCSR on **\*MAT\_MODIFIED\_HONEYCOMB**.

30. Improved using the strain-rate-dependent tables (for LCA, etc.) for **\*MAT\_MODIFIED\_HONEYCOMB (\*MAT\_126)**, resulting in a smoother and more robust response curve.
31. Added output of the formability index as history variable #9 for shell elements with material **\*MAT\_133**.
32. Added field CTS to **\*MAT\_COHESIVE\_MIXED\_MODE (\*MAT\_138)** to flag whether to use a consistent tangent stiffness in implicit simulations. A consistent tangent stiffness improves convergence. Additionally, this flag enables changing the cohesive law formulation from the original to the Ortiz and Pandolfi cohesive law. The Ortiz and Pandolfi cohesive law is often efficient and stable in implicit finite element analyses. It considers the effect of mode mixity on the peak traction but neglects the effect of mode mixity on the critical energy release rate, so it can be considered as a simplified version of the original **\*MAT\_138** cohesive law.
33. For **\*MAT\_172 (\*MAT\_CONCRETE\_EC2)**:
  - The shear capacity check now works for beam elements (ELFORM = 1 on **\*SECTION\_BEAM**). Previously, it only worked with shells and thick shells.
  - Added shear capacity check options per Eurocode:
    - TYPESC = 5 for sections without shear reinforcement;
    - TYPESC = 6 for sections with shear reinforcement.
34. Made an improvement to **\*MAT\_187** to avoid failed iterations in the plasticity algorithm: Added a convergence criterion that helps in high triaxiality cases ( $|p| > \sigma_{vm}$  where  $p$  is pressure and  $\sigma_{vm}$  is the von Mises stress).
35. Improved the behavior of **\*MAT\_187** with LCID-C > 0 and poorly defined LCID-P. Extrapolation into the negative region could have led to a Poisson's ratio that is less than -1. Now, a constant value is used instead of extrapolating.
36. Made improvements for shells with **\*MAT\_SAMP\_LIGHT (\*MAT\_187L)**, such as always using Brent's method if LCID-T and LCID-C

- are different tables with an improved start value for Brent.
37. For **\*MAT\_209** (**\*MAT\_HYSTERETIC\_BEAM**), added the possibility for different nonlinear shear inputs for positive versus negative direction shear. This capability can be relevant when shear reinforcement is set at an angle to the through-thickness direction. The new parameters are on Card 4.1, which is read only if LCSHS is input as negative.
  38. Updated **\*MAT\_213** to version 1.3.8, which includes the following new features:
    - Added a Point Cloud Failure Criterion.
    - Updated the flow rule coefficient checks. They are now checked to be positive-definite.
  39. For **\*MAT\_224**, made the plastic work conversion to a thermal heat source available for axisymmetric solids.
  40. Added option to **\*MAT\_240** to make elastic moduli rate dependent. EMOD (GMOD) < 0 references a curve giving the Young's (shear) modulus as a function of strain rate. Logarithmic interpolation is used if the first abscissa value is negative. This feature is available for **\*MAT\_240** without the THERMAL, 3MODES, or FUNCTIONS keyword options.
  41. For **\*MAT\_240\_FUNCTIONS**, add the following function arguments: tangential separations, normal separation, plastic tangential separations, plastic normal separation, and cohesive thickness.
  42. Added new history variables for **\*MAT\_240\_THERMAL**. History variable #16 is FG1, #17 is FG2, and #25 is the state variable.
  43. Enabled **\*MAT\_251** to work with solid tetrahedron type ELFORM = 13.
  44. Added keyword option STOCHASTIC to **\*MAT\_251**. The scaling factor for the yield stress is on history variable #9. The current yield stress can be found on history variable #5.
  45. Support the external variable feature (**\*LOAD\_EXTERNAL\_VARI-**

- ABLE**) for shell elements with **\*MAT\_254**. Previously, only solid elements were supported for this material.
46. For **\*MAT\_264**:
    - added 3D orthotropic failure models (IFTRA > 0),
    - improved evaluation of tables, and
    - corrected the plastic strain increment for temperature changes.
  47. Support BETA < 0 in **\*MAT\_264**. |BETA|, in this case, refers to a curve, table, 3D table, or 4D table as in **\*MAT\_224** and **\*MAT\_224\_GYS** to define a non-constant Taylor-Quinney coefficient.
  48. Added a less than 0.0 case for BETAC and BETAT on **\*MAT\_272**. These cases enable load curves to be defined to give the strain rate dependence of the dynamic increment factor,  $F_p$ .
  49. Added new option for the critical energy criterion of **\*MAT\_280** (**\*MAT\_GLASS**): ENGCRIT < 0 refers to a **\*DEFINE\_FUNCTION** ID equal to |ENGCRIT|. This function has one argument: the minimum distance of the center of impact to the edge of the windshield. Thus, the critical energy value can now depend on this distance.
  50. For **\*MAT\_ANAND\_VISCOPLASTICITY**, added support for temperature-dependent Young's modulus and Poisson's ratio. For this feature, YM/PR should be defined as negative IDs (-ID), where the ID refers to a curve that defines the Young's modulus/Poisson's ratio as a function of temperature.
  51. Enabled **\*MAT\_296** for pentahedral elements.
  52. Support Smoothed Particle Galerkin (SPG) for **\*MAT\_303**.
  53. Added the ability to automatically initialize a 2D random short fiber orientation state for shell elements. This state refers to the condition where reinforcing fibers are distributed uniformly in all in-plane directions, resulting in isotropic material behavior in all in-plane directions. To obtain this state, set  $A_{xx} = A_{yy} = 0.5$  in **\*INITIAL\_STRESS\_SHELL** for the fiber orientation tensor. All other components of the tensor are zero.

54. Support shell element formulation 26 for **\*MAT\_303**.
55. Extended the structural damage  $D_2$  below the gelation point to the compression regime for **\*MAT\_307**. This change ensures that the damage from compression is neutralized as soon as the gelation point is reached, but the damage from viscous fingering is not.

## 6.21 Mesh Cards

1. Added field ISIDE to **\*MESH\_BL** as flag for specifying whether the boundary layer is defined on only one side of the embedded surface (**\*MESH\_EMBEDSHELL**).
2. Added specifying a minimum radius (RMIN) when using the sphere option on **\*MESH\_SIZE\_SHAPE** with METHOD = 1.

## 6.22 MPP

1. Updated the MPP decomposition code to support **\*INITIAL\_INTERNAL\_DOF\_SOLID\_....**
2. Support load balancing for tabulated friction coefficients (FS = -2 on Card 2 of **\*CONTACT\_...**). Previously the nodal information required for tabulated friction coefficients was not transferred during load balancing.

## 6.23 Multiscale

1. Added a periodic displacement boundary condition for RVEs with zero-thickness cohesive interface elements invoked with BC = 3 on **\*RVE\_ANALYSIS\_FEM**. It is intended for material interface debonding prediction.

## 6.24 Output

1. Made various improvements to the accuracy of the energy reported in the rcfrc file for MPP contacts.
2. Added **\*DATABASE\_SLEOUT\_DR**, **\*DATABASE\_MATSUM\_DR**, and **\*DATABASE\_GLSTAT\_DR** to output sleout, matsum, and glstat during dynamic relaxation.
3. Improved the computational efficiency of **\*DEFINE\_MATERIAL\_HISTORIES** label 'Max/Principal Stress Range'.
4. Added output of the cohesive separations in the location of the strain tensor when the ones digit (*L*) of STRFLG is 1 on **\*DATABASE\_EXTENT\_BINARY**.
5. Echo warning message, KEY+1655, when part set IDs in **\*DATABASE\_EXTENT\_SSSTAT\_ID** and **\*DATABASE\_EXTENT\_SSSTAT** are zero.
6. Write the d3dat data when sense switch 4 (sw4) is provided and during an error termination. Note that d3dat is only output when running CPG problems or if the input deck contains **\*DATABASE\_BINARY\_D3DAT**.
7. Added field COR to **\*DATABASE\_TRACER\_GENERAL** to correct the tracer position if it is found outside the tracked mesh during initialization.
8. Added flag IAVER to **\*DATABASE\_ALE** to provide the option to average variable histories over the elements in the set and ALE groups.
9. Enabled **\*DATABASE\_FSI\_SENSOR** to work without coupling to a Lagrangian structure for ALE and S-ALE models.
10. Allow specifying the sensor location without providing SEGMID and by just setting ND1, ND2, and ND3 on **\*DATABASE\_FSI\_SENSOR**.
11. The following types of elements are now supported for setting tracers in (see ELEM) with **\*DATABASE\_TRACER\_GENERAL**:
  - 2D beams (ELFORM = 7 or 8 on **\*SECTION\_BEAM**)

- Discrete beams (ELFORM = 6 on **\*SECTION\_BEAM**)
  - Belytschko-Schwer beams with cross-section integration (ELFORM = 4 or 5 on **\*SECTION\_BEAM**)
  - Belytschko-Schwer resultant beams (ELFORM = 2 on **\*SECTION\_BEAM**)
  - Trusses (ELFORM = 3 on **\*SECTION\_BEAM**)
  - Hughes-Liu beams with tubular sections (ELFORM = 1 with CST = 1 on **\*SECTION\_BEAM**)
12. Support the output of the plastic strain tensor for more materials (STRFLG = 11 on **\*DATABASE\_EXTENT\_BINARY**): **\*MAT\_003**, **\*MAT\_010**, **\*MAT\_018**, **\*MAT\_098**, **\*MAT\_103**, **\*MAT\_107**, **\*MAT\_124**, and **\*MAT\_187**.
  13. Print information about the expiration date in the encrypted vendor block with DATE.
  14. Added nodal moments (x, y, and z) to the nodfor file (**\*DATABASE\_NODFOR**).
  15. Added option ISETS to **\*DATABASE\_BINARY\_D3PROP** to write all **\*SET** data to d3prop or the end of the first d3plot file.
  16. Parts with **\*MAT\_PML** are no longer sub-divided into multiple parts and will appear in d3plot as they appear in the input file.
  17. l2a will now extract aceout from the binout file.
  18. Added NFAIL to **\*DATABASE\_D3MAX** which determines whether failed elements are included in d3max.
  19. Added OUTPUT = 8 and 9 to **\*DATABASE\_D3MAX** to write the maximum stress/strain results, the minimum stress/strain results, and the absolute maximum stress/strain results using the current and original geometry, respectively.

## 6.25 Sensors

1. Added MTYPE = PULLEY to **\*SENSOR\_DEFINE\_MISC** to trace

the pulley's force.

2. Enabled **\*SENSOR\_DEFINE\_FORCE** to report the cross-sectional moment for **FTYPE = X-SECTION**.
3. Added flag **SIDR** to **\*SENSOR\_CONTROL** to control the use of sensor control during dynamic relaxation.

## 6.26 SPG (Smooth Particle Galerkin)

1. Support planar 2D ISPG (**ELFORM = 50** on **\*SECTION\_ISPG**) for combined flow and thermal analysis (**SOLN = 1** on **\*ISPG\_CONTROL\_SOLUTION**).
2. Added damage model **IDAM = 8** (**\*SECTION\_SOLID\_SPG**). For this model, **FS** is the critical value of the damage value taken from the material model, and **STRETCH** is not used.
3. Added support for SPG parts during dynamic relaxation.
4. Support SPG for orthotropic material models 2, 22, 26, 54, 58, 59, 103, 104, and 274.

## 6.27 SPH

1. Made the SPH mass flow plane (**\*DEFINE\_SPH\_MASSFLOW\_PLANE**) results consistent between SMP and MPP.
2. Added error trap for invalid **IOPT** values in **\*DEFINE\_ADAPTIVE\_SOLID\_TO\_SPH**.
3. ISPH structure particles (meaning particles embedded on shell meshes to represent the boundaries) are no longer shifted when **ISHIFT = 1** in **\*CONTROL\_SPH**.

## 6.28 Thermal Solver

1. For axisymmetric and 2D solids, the thermal thick shell option (THSHEL on **\*CONTROL\_SHELL**) is deactivated now, and a warning is printed, instead of throwing an error.
2. Added the node list of the current segment as a subroutine argument to the user flux interface (subroutine `usrflux`).

## 6.29 Miscellaneous

1. Changed the logic of parameter evaluation in keyword titles/headings. If the parameter name exists, we evaluate the parameter; otherwise, we consider the “&” symbol to be part of the string and no error is returned.
2. Added new options “SET” and “DSET” to **\*SET\_SEGMENT\_GENERAL**. Option SET adds the listed **\*SET\_SEGMENT** sets to the current set, while option DSET deletes the listed **\*SET\_SEGMENT** sets from the current set.
3. Added command line option `heartbeat = ncycleh` to write a status line to an ASCII file named `heartbeat`. If `ncycleh > 0`, a line is appended to the file every `ncycleh` cycles. If `ncycleh < 0`, the file contains a single line that is overwritten every `ncycleh` cycles.
4. Updated the `l2a` utility to handle the case where you add or remove rigid walls via a restart, but append to the existing `lsda` output file. The `glstat` file parsing code was breaking if the number of rigid walls suddenly changed.
5. Added new option `SET_PART` for **\*SET\_NODE\_GENERAL**, so that a (list of) **\*SET\_PART** can be used to provide the node set. This allows much greater flexibility than the existing PART option, since instead of listing the individual parts, you can list part sets and so take advantage of part lists created with **\*SET\_PART\_GENERATE**, **\*SET\_PART\_ADD**, and **\*SET\_PART\_GENERAL**.
6. Support **\*INTERFACE\_COMPONENT\_...** and **\*INTERFACE\_LINKING\_...** during dynamic relaxation. If both runs have dynamic relaxation, or neither does, then the linked interface follows the motion of the component interface through the whole simulation as expected. If

- only the first run has dynamic relaxation, then the motion of the linked interface during the second run follows the “dynamic relaxation converged” state of the first run. If only the second run has dynamic relaxation, then during dynamic relaxation the linked interface does not move at all but is tied to the initial coordinates of the first run.
7. Added option **MATTYPE** for **\*SET\_PART\_GENERAL**. With this option you can specify numeric material types (1 for **\*MAT\_ELASTIC** and so on). For each material type, ALL parts with the specified material type are added to the part set.
  8. Added  $A7 = 2$  to the **MIRROR** option of **\*DEFINE\_TRANSFORMATION** to change the element connectivity or segment node numbering to mirror the element/segment normal. The initial stress/strain is reordered as well to match the mirrored element coordinate system. This is available for linear shell, solid, and thick shell elements.
  9. Enabled the curve imported from the FMU to control thermal time steps (**\*COSIM\_FMI\_INTERFACE**).
  10. Enabled exporting LS-DYNA’s nodal temperature to the FMU (see **FIELD = TMPT** on **\*COSIM\_FMI\_INTERFACE**).
  11. Time spent on FMU (**\*COSIM\_FSI\_...**) is now reported in d3hsp.
  12. Added option **STRUCT** to **\*DAMPING\_PART\_MASS** to exclude lumped mass from **\*ELEMENT\_MASS**.
  13. Added additional functionality to easily model bolted joints (**\*DEFINE\_BOLTED\_JOINTS**). See the manual for details.
  14. Added the ability to have the file read in by **\*INCLUDE\_STAMPED\_PART** be in LSDA format. Previously, the file could only be in ASCII format. Initial stresses, strains, and history variables are all supported.
  15. Implemented specifying a death time, **DEATH**, in both **\*INTERFACE\_LINKING\_SEGMENT** and **\*INTERFACE\_LINKING\_NODE**, after which the linking constraint is deactivated. This feature allows the linking to be switched off at a specified point during the analysis.

# 7 Bug Fixes

## 7.1 Acoustics, NVH, and Frequency Domain

1. Fixed a bug in computing acoustic pressure for nodes on the BEM mesh (**\*FREQUENCY\_DOMAIN\_ACOUSTIC\_BEM**) in models that include an incident wave. Previously the contribution from the incident wave was counted twice, causing the results from **d3acs** and **d3acp** were not consistent.
2. Fixed bug when restarting a job that uses the acoustic spectral element method (**\*CONTROL\_ACOUSTIC\_SPECTRAL**). The bug caused the restart to fail.

## 7.2 Adaptivity

1. Fixed an issue where excessive mesh refinement occurred when **INMEMORY = 1** in **\*CONTROL\_ADAPTIVE**. The minimum mesh size is now consistent with the behavior when **INMEMORY = 0**.
2. Resolved a memory access violation occurring at the initial stage of the adaptive cycle in MPP solid adaptivity. This issue previously led to “Negative Volume” errors during the calculation phase.

## 7.3 Airbags

1. Fixed compatibility issue when using **\*PART\_COMPOSITE** and **\*AIRBAG\_CPG** in the same input deck. Automatically generated part IDs could conflict in some scenarios.
2. Fully support multiple **\*AIRBAG\_CPG** definitions.
3. Fixed issue when there are multiple orifice cards (**NORIF > 1** on **\*AIRBAG\_CPG**) referencing the same gas for CPG airbags. Each orifice was injecting the entire gas mass, instead of distributing it across orifices.

4. Fixed part temperature output to `abstat_cpg` for internal parts. It was erroneously reporting double the correct value. It was an output issue only and did not affect the simulation.
5. For CPG airbags, fixed the porosity leakage calculation. The user-provided load curve gives gas velocity as a function of absolute pressure, but the previous implementation assumed this was referring to the pressure differential between the bag pressure and atmospheric pressure.
6. Fixed issue when internal parts are inflators for CPG airbags. Orifice particles became duplicated, resulting in an incorrect boundary condition being applied.
7. Fixed a bug in `*AIRBAG_INTERACTION` that produced incorrect, unusually too low temperature results when `LCID` is defined.
8. Fixed bug in CPG airbags (`*AIRBAG_CPG`) that occurred when multiple airbags had vents. The issue was incorrectly causing the error “Part number is used as a vent but is not found in the parts defining the airbag.”

## 7.4 ALE

1. Fixed memory conflict in models containing both S-ALE and SPH parts that led models to crash and/or resulted in incorrect results.
2. Fixed memory error that occurred when S-ALE models contain solid elements without a corresponding part definition. The solid elements were incorrectly treated as S-ALE elements, leading to the error.
3. Fixed error in the recorded and reported forces and pressures to the files created with `*DATABASE_FSI` and `*DATABASE_FSI_SENSOR` for S-ALE. Other than the incorrect output, the FSI runs are still valid and trustworthy.
4. Fixed issue with initial velocity assignment (`VID` on `*ALE_STRUCTURED_MESH_VOLUME_FILLING`) to S-ALE nodes when using S-ALE volume filling that led to the incorrect assignment of

velocity to nodes outside the filling geometry instead of only inside the geometry.

## 7.5 Battery Cards (Electrochemistry Solver)

1. For battery models, corrected the output of the thermal data printed to an ASCII file when running with multiple processors.

## 7.6 Boundary Cards

1. Fixed bug when using **\*BOUNDARY\_FLUIDM** with **\*CONSTRAINED\_NODAL\_RIGID\_BODY** and **\*MAT\_RIGID (\*MAT\_020)** in MPP. The bug caused jobs to crash.
2. Fixed an issue with **\*BOUNDARY\_MCOL** in reading multi-database (.mco) files correctly.

## 7.7 CESE Compressible Fluid Solver

1. Corrected the detection of triangular faces for the CESE conjugate heat transfer boundary condition.
2. Fixed a chemistry initialization step broken in R15 needed for the chemically-reacting flow solver.
3. Fixed **\*MESH**-based tetrahedral mesh generation for CESE. This fix includes correcting the handling of an error message and correcting the misnaming of two **\*CESE** keywords in the keyword checking database.

## 7.8 Constraint Cards

1. The energy reported in files such as `glstat` previously did not include energy absorbed in **\*CONSTRAINED\_SOIL\_PILE**, but now it is included in “spring and damper energy”.

2. When using **\*CONSTRAINED\_SOIL\_PILE**, as the axial displacement of the beam element (“pile”) relative to the solids (“soil”) increased, (a) an unstable perpendicular response could occur (shown by spurious oscillation and energy growth), (b) the connection points could drift away from their intended positions on the circle centered on the pile node, and (c) spurious torsional moments could develop in the pile beam. These issues have been fixed, and stability can be further improved by the new capability to specify perpendicular damping (DAMPH). Note also that the issues listed above did not occur when the axial displacements were small.
3. For SPR3 with the strongly object formulation (MODEL = 3 on **\*CONSTRAINED\_INTERPOLATION\_SPOTWELD**) with failure activated, fixed issue that caused the failure state not to propagate to the associated element in the output databases. The element appeared intact in post-processing even after failure had occurred.
4. Fixed incorrect values written to the binary output databases for the reaction moments at a rotational-motor joint (**\*CONSTRAINED\_JOINT\_...\_MOTOR**).

## 7.9 Contact

1. Fixed an issue in the MPP groupable **AUTOMATIC\_GENERAL** beam-to-beam contact that could result in the application of forces that were too low.
2. Fixed an issue in MPP that could cause contact forces for SOFT = 2 with DEPTH = 45 contacts with enabled friction to not be correctly attributed to corresponding force transducers.
3. Addressed a consistency issue of the hybrid solver in **\*CONTACT\_AUTOMATIC\_SINGLE\_SURFACE** for SOFT = 0 and 1.
4. Fixed output of MPP groupable contact timing information in the cont\_profile.csv file, which was never properly supported and so gave incorrect results for “wall clock time.”

5. In some cases, an MPP tied contact with birth time was not being treated as “groupable,” which is required for birth time to work properly for tied contacts.
6. Fixed issue in determining the beam element contact thickness due to not honoring the thickness scaling factor sometimes. Specifically, **\*CONTACT\_AUTOMATIC\_BEAMS\_TO\_SURFACE** was not honoring the SFSAT parameter.
7. Fixed issue in **\*INITIAL\_CONTACT\_WEAR** where wear depth was not saved to the dynain file.
8. With nonzero friction, the stonewall energy was not correct for **\*RIGIDWALL\_PLANAR**.
9. Fixed implicit eigenvalue analyses that use segment-to-segment (SOFT = 2) contact. The stiffness was incorrect causing wrong values.
10. Fixed the thick segment pair check of MPP single-surface eroding segment-to-segment (SOFT = 2) contact. The thick segment check was not working which could cause possible spurious contact forces and internal energy in shell parts when elements have large thickness-to-length ratios.
11. Made the **\*CONTACT\_ENTITY** IDs output to binout for both SMP/MPP be the same as the IDs output to the ASCII output for SMP.
12. Fixed issue causing hanging with hybrid executables when using eroding contact.
13. Fixed bug for case where the model contains more than one **\*CONTACT\_TIED\_SHELL\_EDGE\_...** keyword (MPP only). Whether the bug showed any symptoms depended on the number of processors and the model contents. Symptoms could include crashing on the first calculation cycle.
14. Fixed the contact-segment search region being too small for Mortar contact. The bucket sort range used when searching for nearby contact segments could be underestimated in some cases, causing segments to be overlooked. This led to missing contact forces where contact should have been detected.

15. Fixed segment-to-segment ( $\text{SOFT} = 2$ ) contact that involved higher-order elements (quadratic or higher). The issue resulted in incorrect behavior with errors.

## 7.10 Control Cards

1. Fixed bug related to pore fluid flowing through a tied contact (see FTIED on **\*CONTROL\_PORE\_FLUID**). The bug occurred when some (but not all) nodes on the tied contact are either (a) subject to **\*BOUNDARY\_PWP** with  $\text{IPHRE} = 1$ , or (b) belong to drained parts while others belong to parts undergoing time-dependent consolidation. Possibly the bug could also occur with other flavors of **\*BOUNDARY\_PWP**. Symptoms were nonphysical large pore pressures developing at the surrounding nodes on the tied contact.

## 7.11 Discrete Element Method

1. Fixed bug in **\*DEFINE\_DE\_ACTIVE\_REGION** with  $\text{NFREQ} = 1$  that led to incorrect results.

## 7.12 Dual CESE Compressible Fluid Solver

1. Corrected the REFPROP EOS table look-up on all processors other than processor 0.

## 7.13 Elements

1. Fixed a unit conversion issue for **\*ELEMENT\_SEATBELT\_PRE-TENSIONER** affecting the PTLCID curve of types 4, 6, 7, 8, and 9 when using transformation factors in **\*INCLUDE\_TRANSFORM**.
2. Fixed the incorrect part mass and body mass for thick shells with **\*ELEMENT\_MASS\_PART**.

## 7.14 EM (Electromagnetic Solver)

1. Fixed incorrect behavior in MPP and incorrect units used when using `CRTYPE = 0` on `*EM_CONTACT_RESISTANCE`.
2. Stopped calculating the contact resistance Joule heating term unless `JHRTYPE = 1` when `CRTYPE = 1` on `*EM_CONTACT_RESISTANCE`.
3. Fixed issue with `TSTYPE = 5` on `*EM_CONTROL_TIMESTEP` when `THCPL = 2` in `*EM_CONTROL_COUPLING` that caused incorrect behavior.
4. In Eddy current 2D axisymmetric cases, the displacement of conductors is now properly taken into account when outputting results of magnetic flux in the air (see `*EM_DATABASE_POINTOUT`).
5. Fixed `MTYPE = 3` on `*EM_PERMANENT_MAGNET`, which was not working properly in MPP. Additionally we fixed a missing translation to the origin point operation with `MTYPE = 6`.
6. For `*EM_RANDLES_TSHELL`, fixed issue in the part assignment and node ordering that was causing incorrect results in MPP.
7. Made a small fix to the **P** matrix assembly for tetrahedral elements in magnetostatic cases. This fix improved the results.
8. Fixed the output frequency for the file output with `*EM_DATABASE_NODOUT`. It was being output one timestep ahead.
9. Fixed issue in `*EM_EP_EIKONAL` so that `EIKPST` is now effectively considered in the time stepping Eikonal, and the Eikonal solve is done only on the corresponding part set. If `EIKPST = 0`, the Eikonal is done on the whole conducting mesh.

## 7.15 ICFD (Incompressible Fluid Solver)

1. Corrected the force computation for the RIIS immersed method (`IMMTH = 2` on `*ICFD_CONTROL_IMMersed`) on beam

- elements.
2. Fixed the coupling between ICFD and DEM particles when the force is computed using the fluid pressure gradient (FORM = 2 on **\*ICFD\_CONTROL\_DEM\_COUPLING**). This bug caused incorrect solutions.
  3. Fixed the average pressure field output in icfd\_nodout.dat (**\*ICFD\_DATABASE\_NODOUT**).
  4. Fixed the mortar contact approach in **\*ICFD\_BOUNDARY\_CONJ\_HEAT**. Some problems (mainly 3D) were incorrect (mainly due to issues in normal vector orientations) as well as not working properly in combination with the nonlinear thermal solver. Also thermal mortar contact did not previously work when using **\*ICFD\_BOUNDARY\_FSI\_FIXED**.
  5. Fixed a divergence issue when using **\*ICFD\_CONTROL\_GAP** with the steady-state solver.
  6. Fixed issues in data output with **\*ICFD\_DATABASE\_NODOUT** and output of Paraview vtu format with **\*ICFD\_CONTROL\_OUTPUT** (OUTL = 7).
  7. Fixed bug in reading multiple cards for **\*ICFD\_BOUNDARY\_PRESCRIBED\_MOVEMESH**.

## 7.16 Implicit (Mechanical) Solver

1. Fixed bug in the LOBPCG eigenvalue extraction method (EIGMTH = 102 on **\*CONTROL\_IMPLICIT\_EIGENVALUE**) that led to a floating-point exception in some configurations.
2. Fixed bug that manifested when using **\*DAMPING\_FREQUENCY\_RANGE\_DEFORM** in implicit dynamic analysis (IMASS = 1 on **\*CONTROL\_IMPLICIT\_DYNAMICS**). Symptoms of the bug could include poor convergence behavior in nonlinear solutions or runaway oscillation in linear solutions.

3. During implicit dynamic analysis ( $IMASS = 1$  on **\*CONTROL\_IMPLICIT\_DYNAMICS**), the interpolation of load curves used for time-dependent features, such as loading, boundary conditions, etc., could sometimes be performed incorrectly, meaning the amount of load applied was not the value expected from the current value of the load curve. The bug depended on how the time step was controlled. A workaround for previous releases was to use **LCTM** on **\*CONTROL\_TIMESTEP** to define a reasonable value for the time step; this workaround was effective even though **LCTM** doesn't actually control the time step of implicit solutions. The issue affected R15 and R16 but not R14 or previous releases.
4. Corrected logic for using implicit intermittent eigenvalue computation with an explicit time simulation (**\*CONTROL\_IMPLICIT\_GENERAL** with  $IMFLAG = 6$ ) to properly handle the eigenvalue computation at the end time. Explicit time stepping may not hit the end time exactly, leading to skipping the intermittent eigenvalue computation and failure occurring.

## 7.17 Initial Cards

1. When the initial keyword file contains "long" format (20-character fields), any **dynain** file written by LS-DYNA is also in long format and the keywords within the **dynain** file are supposed to be marked as such with a "+" (such as **\*NODE+**), but the "+" was omitted for **\*INITIAL\_STRAIN\_BEAM** and **\*INITIAL\_STRESS\_BEAM**. This omission meant that the **dynain** file could not be read back in for the next stage of analysis.
2. Fixed issue with the internal energy computation when using **\*INITIAL\_FOAM\_REFERENCE\_GEOMETRY**. If used in a run with dynamic relaxation, the energy could become negative.

## 7.18 Isogeometric Analysis (IGA)

1. Fixed a bug associated with refinement of periodic knot vectors (**\*IGA\_REFINE\_SHELL** and **\*IGA\_REFINE\_SOLID**).

## 7.19 Load Cards

1. Fixed **\*LOAD\_SEGMENT\_SET\_NONUNIFORM** ignoring nonzero arrival time, AT.
2. Fixed bug that could cause a segmentation fault when running hybrid/mpp binaries with an input deck that contains **\*LOAD\_BLAST\_ENHANCED**.

## 7.20 Materials and Equations-of-State

1. Made fixes to using **\*MAT\_ADD\_DAMAGE\_DIEM** with **\*INCLUDE\_TRANSFORM**. Conversion of logarithmic rates in table P1 did not work, and the output to dyna.inc was not using the integer value for  $Q1 < 0$ .
2. Fixed the combination of **\*INCLUDE\_TRANSFORM** and curves/tables of some materials (were not correctly considered):
  - **\*MAT\_063**: LCID (rate dependent table)
  - **\*MAT\_123**: LCEMOD (rate dependent curve)
  - **\*MAT\_136**: LCID (rate dependent table)
  - **\*MAT\_187**: LCEMOD (rate dependent curve)
  - **\*MAT\_251**: LCSS (rate dependent table)
  - **\*MAT\_280**: FT < 0 (rate dependent curve)
  - **\*MAT\_ADD\_GISSMO**: LCSRS (rate dependent curve)
3. Write missing fatigue data from **\*MAT\_ADD\_FATIGUE** for generated visualization beams. Lack of this data caused input processing failures in models that have visualization beams with material models that included fatigue.
4. Fixed bug in **\*MAT\_004** for beam elements. The reference temperature used by the material model now coincides with the temperature at time  $t = 0$ . Previously,  $T = 0.0$  was always used, leading to unphysical initial axial forces for nonzero expansion coefficients and nonzero initial temperatures.

5. Fixed issue with **\*MAT\_015** shell elements with initial temperature from **dynain**. Initialization of the internal energy was needed as well to get the correct initial temperature.
6. Fixed **\*MAT\_015** with  $VP = 1$  and initial stresses from **\*INITIAL\_-STRESS\_SHELL**. Plastic flow in cycle 0 could lead to problems.
7. Fixed MPP full deck restart issue with **\*MAT\_024** that were not properly initializing some time step information on restart.
8. Fixed incorrect shell element deletion behavior for **\*MAT\_024** for  $VP = 1, 2, \text{ or } 4$ . The element was deleted only when the last integration point failed. The corrected behavior is that the element should only be deleted when all the integration points failed.
9. Changed the plasticity algorithm for **\*MAT\_030** in the case where the yield stress is provided as load curve input. The old implementation might result in an unphysical time-step-size dependence for larger explicit simulations.
10. Fixed stress initialization of **\*MAT\_030** beams. The deformation gradient coming from the history history variables in the **dynain** was incorrectly set to zero in the stress computation at  $t = 0$ .
11. Fixed bug that could caused the analysis to crash if solid elements with  $ELFORM = 1$  and material **\*MAT\_060** were used with implicit time integration. The bug did not affect **\*MAT\_060** solids with  $ELFORM = 2$  or  $-2$ , nor did it affect **\*MAT\_060** shells.
12. Fixed the unloading behavior in **\*MAT\_063** with  $MODEL = 0$ . This fix can change results, so we added  $MODEL = -1$  to call the old version.
13. For **\*MAT\_076\_MOISTURE**:
  - Fixed reading initial moisture (parameter  $M0$ ) value from the input deck.
  - Corrected expansion due to moisture change.
  - Fixed moisture stiffness scaling.
  - Fixed the temperature shift by taking the correct initial temperature value into account. Previously, the initial temperature was always assumed to be zero.

- Fixed the initial temperature always being assumed to be zero in an uncoupled analysis.
14. Fixed bug for **\*MAT\_100** with  $SIGY < 0$ . If this material was used with solid element spot welds with  $SIGY$  referring to a table, the strain rate was wrong.
  15. Fixed issue with the function evaluation in some spot weld models. This issue applies to **\*MAT\_100\_DA** with **\*DEFINE\_CONNECTION\_PROPERTIES** ( $PRUL = 2$ ), **\*MAT\_169**, **\*MAT\_240\_FUNCTIONS**, and **\*DEFINE\_MULTI\_SHEET\_CONNECTORS**. **\*DEFINE\_FUNCTION** arguments  $sm1$  and  $sm2$  (maximum engineering yield stresses) could have been wrong if determined from a curve/table in the weld partner material. This has been broken since March 2024.
  16. Corrected the damage and stress calculations for the extended Cockcroft-Latham model in **\*MAT\_107**.
  17. Fixed bug in **\*MAT\_MODIFIED\_HONEYCOMB** that caused the element volume to not be updated for solid element formulations 0 and 9, leading to an incorrect compression status and stiffness. Note that fixing this bug leads to a different stiffness and to different results.
  18. Fixed issue when using **\*MAT\_PLASTICITY\_COMPRESSION\_TENSION\_EOS** (**\*MAT\_155**) with S-ALE that led to incorrect results due to the internal energy not being updated.
  19. Fixed issue in the initialization and stress update of **\*MAT\_179** when tables are used for LCID1 and LCID2.
  20. Fixed bug in **\*MAT\_187L** when  $RATEOP = 1$  and LCID-C is a table. A potential change of the Ducker-Prager yield criterion slope during return mapping was not correctly taken into account, leading to wrong stresses.
  21. Fixed bug in **\*MAT\_SAMP\_LIGHT** (**\*MAT\_187L**) with solids and difficult return mapping. An update of a history variable (deviatoric plastic strain) was missing when Brent's method was used. This could have led to wrong stress results.

22. Fixed plasticity algorithm issue in **\*MAT\_199**. Discontinuous behavior was observed in some cases due to an unfortunate condition in principal stress computations.
23. Fixed the latent heat treatment for solid elements with **\*MAT\_254**. In certain cases, latent heat was ignored.
24. Fixed issue when using large IDs ( $> 2^{24}$ ) for LCSS in **\*MAT\_261**.
25. For curves used by **\*MAT\_270**, only offset the curve ID by the value specified by **\*INCLUDE\_TRANSFORM** if the original curve ID is nonzero.
26. For **\*MAT\_295**, fixed a bug with input relying on curves and/or functions, including **FTYPE = 1** or **2** invoking **FCID**, **ACTIVE** models invoking **ACID**, and **ACTYPE = 5** invoking **FSEID**, **FLID**, **FVID**, and **ALPHAID**.
27. Fixed incorrect behavior when using **\*MAT\_...\_STOCHASTIC** with **VAR\_S = VAR\_F = 2** in Card 1 of **\*DEFINE\_STOCHASTIC\_VARIATION**. All the elements with the material were either being deleted at cycle 1 or not failing.

## 7.21 MPP

1. Fixed a problem in segment-to-segment (**SOFT = 2**) MPP contact of MPI messages that were not received causing the code to hang. The fix is to change the message size from the buffer size to the message size. This message failure was only observed in a large model with a very large buffer.
2. Fixed issue with **\*DEFINE\_STOCHASTIC\_ELEMENT\_SHELL\_VARIATION** in MPP. The MPP decomposition data distribution was not working for this keyword.

## 7.22 Multiscale

1. Fixed a bug in two-scale cosimulation (**\*INCLUDE\_COSIM**) that

caused reading `*SET_SOLID` to fail for solid-in-solid coupling if the last line of solid IDs does not contain all 8 entries.

## 7.23 Output

1. Fixed rare case where a parameter expression (`*PARAMETER_EXPRESSION`) might not get echoed to the `d3hsp` file. This lack of echoing only happened when the `*PARAMETER_EXPRESSION` instantiation was the last keyword in an include file.
2. Use the same nodal interpolation to recover stresses for solid `elform 23` in `d3plot` as in `eloutdet` for `SOLSIG ≠ 0` on `*CONTROL_OUTPUT`.
3. Fixed a bug that occurred when `*DATABASE_PWP_OUTPUT` was used in models without pore air. The bug was erroneously causing the acceleration values at all nodes to be zero.
4. Fixed an MPP bug that caused incorrect output to `nodout` for the rotational DOF of a node constrained to a rigid part belonging to another core.
5. Fixed a bug for `NODOUT` of accelerometer, `*ELEMENT_SEATBELT_ACCELEROMETER`, that happened when `IGRAV = 1`.
6. Fixed a bug that led to printing wrong values for function `PIDCTL` (`*DEFINE_CURVE_FUNCTION`) to `curvout`.
7. Fixed issue for shell `ELFORM = 14` where output to `nodfor`, `spcfor`, and `secfor` was not consistent with `ELFORM = 15` which reports force per radian.
8. Fixed issue for `*DATABASE_NODAL_FORCE_GROUP` where legends did not match groups in `binout/ascii` files.
9. Fixed bug in `*DEFINE_MATERIAL_HISTORIES` that could cause solid integration point strain (`NINTSLD = 8` on `*DATABASE_EXTENT_BINARY`) output to binary databases to be overwritten.
10. Fixed incorrect eroded kinetic energy when an inactive node also

belongs to a rigid body. The eroded kinetic energy was incorrectly including nodes shared with a rigid body, such as **\*CON-STRAINED\_NODAL\_RIGID\_BODY**. This fix corrects the eroded KE in both `glstat` and `matsum`.

11. Fixed eroded kinetic energy differences between `glstat` and `matsum`.
12. Fixed incorrect internal energy in `glstat`.
13. Fixed issue with `MATEN = 2` on **\*CONTROL\_ENERGY** in implicit analysis. The new energy increment should only be added if implicit converged. Previously, the results of `mat_energy_elastic/plastic` could become way too high.
14. Fixed potentially missing stress output to `d3plot` with **\*PART\_COMPOSITE**. Stresses could have been zero for some integration points if `CMPFLG = 1` (**\*DATABASE\_EXTENT\_BINARY**) and the material of first integration point is orthotropic but others are isotropic.
15. Fixed a bug in writing `d3max` (**\*DATABASE\_D3MAX**) that occurred when plastic strain tensor output is requested in **\*DATABASE\_EXTENT\_BINARY**.

## 7.24 Restarts

1. Fixed an MPP full deck restart reading synchronization error when the first simulation has 6 degrees of freedom and the second has only 3.
2. Fixed an issue where ALE data being output to the `d3full` file could in some cases corrupt the `d3full` file, resulting in unpredictable behavior when restarting from the `d3full` file.
3. Fixed issue with communicating certain information between MPP ranks for solid element formulation 13 (tetrahedral elements) while running a restart.
4. The numbering of dump files (`d3dump` and `d3full`) was broken. The correct behavior, where the first file is always 01, is now restored.
5. Fixed bug in obtaining the `d3plot` output interval using a load curve that

- occurred after a full deck restart.
6. Fixed issue causing the beam axial strain to not be transferred in a full restart.

## 7.25 Seat Belts

1. Fixed a bug in 2D seat belts that occurred when the formulation parameter, FORM, is 12, 13, or 14. This bug only occurred for double precision. It sometimes led to instability.
2. Fixed a bug in the seat belt's force results provided to d3plot. The old force was along the global coordinate system. The new results are along the axial direction.
3. Fixed an MPP bug that occurred in the new decomposition of a 2D belt when 1D and 2D seat belts coexist in a model.

## 7.26 Sensors

1. Fixed bug that occurred when using **\*SENSOR\_DEFINE\_FORCE** with **FTYPE = X-SECTION** in conjunction with **\*SENSOR\_CONTROL** with **TYPE = ELESET**. The reported section force could be erroneous.
2. Fixed a bug that caused incorrect rigid body kinematics values to be reported by **\*SENSOR\_DEFINE\_MISC** with **MTYPE = RIGIDBODY** when the input deck also included **\*SENSOR\_CONTROL** with **TYPE = ELESET**.

## 7.27 SPH

1. Fixed incorrect values used for SPH heat capacity in heat transfer simulations.
2. Fixed issue with **ISHIFT = 1** on **\*CONTROL\_SPH** causing incorrect shifts to be assigned to particles.

3. Made fixes to avoid floating point overflows during high velocity simulations that use the following materials with SPH:
  - **\*MAT\_022/\*MAT\_COMPOSITE\_DAMAGE**
  - **\*MAT\_054-055/\*MAT\_ENHANCED\_COMPOSITE\_DAMAGE**
  - **\*MAT\_059/\*MAT\_COMPOSITE\_FAILURE\_SPH\_MODEL**
4. Corrected multiple issues that did not account for ghost particles being used with the thermal solver and/or **\*DEFINE\_ADAPTIVE\_SOLID\_TO\_SPH**.
5. Corrected issues causing particles to incorrectly affect non-neighboring particles while using FORM = 12 on **\*CONTROL\_SPH**.
6. Fixed issue with allocating more memory for neighbor search when tracer interpolation (**\*DATABASE\_TRACER**) was enabled for SPH.
7. Corrected **\*INITIAL\_STRESS\_SPH** output written to the dynain file.
8. Addressed issues with kinetic and internal energies coming from inactive embedded particles. In some cases, their energies were not removed, while in others they were removed twice.
9. Fixed issue with **\*BOUNDARY\_SPH\_NOSLIP** leading to incorrect pressures for 2D SPH simulations when using MPP single precision versions.
10. Properly account for IDEOFF in **\*INCLUDE\_TRANSFORM** when NEND in **\*ELEMENT\_SPH** is provided.
11. Corrected particles reported to the sldsph\_frag file when **\*DEFINE\_ADAPTIVE\_SOLID\_TO\_SPH** is used.
12. Added missing particle radii to restart files for axisymmetric cases (IDIM = -2 in **\*CONTROL\_SPH**). This issue would lead to incorrect particle radii for restarts.
13. Apply no-slip fictitious velocities to SPH tracer output (**\*DATABASE\_TRACER**).

## 7.28 Thermal Solver

1. Fixed the input trap for **\*BOUNDARY\_RADIATION\_[OPTION]\_VF\_READ** to check for the correct view factor file name. The error message was also modified.
2. Fixed initialization issue for **\*LOAD\_HEAT\_GENERATION\_SET\_OPTION** that occurred if the same input deck contained the SOLID and SHELL keyword options. This issue led to a segmentation fault.

## 7.29 User-Defined Features

1. Fixed issue where the usermat function GETUSRVER was always called when using sharelib usermat. This caused issues for shared object files where the subroutine was not present. It is now only called if found in the shared object.

## 7.30 Miscellaneous

1. Corrected the center of mass of **\*PART\_INERTIA** if **\*PART\_MOVE** is specified.
2. Fixed the local coordinate system calculations in MPP when using **\*DEFINE\_COORDINATE\_NODES** with DIR input as Y or Z. These cases were not working correctly and could produce incorrect output.
3. Fixed a bug that caused problems when co-simulating LS-DYNA-generated FMUs (**\*COSIM\_FSI\_...**) with Adams. The bug was due to an incorrect, reversed sequence of I/O variable definitions within the modelDescription.xml file.
4. Fixed issue for **\*DEFINE\_CURVE\_FUNCTION** where several functions could overwrite each other if no local coordinate system was attached to nodes when using the functions AX and AX2.

5. Fixed bug in **\*SET...\_COLLECT** that led to an error when a set was empty.
6. If a set of type **GENERAL** referenced a set of type **LIST\_GENERATE** defined later in the input deck, the set was not properly defined. We removed this input deck order dependence.
7. Fixed bug for the combination of **\*DAMPING\_FREQUENCY\_RANGE\_DEFORM** with **\*DAMPING\_PART\_STIFFNESS** acting on solid elements with ELFORMs 1, 10, and 15. The results for these element types were showing greater damping than expected.
8. Fixed incorrect velocity transformations after deformable-to-rigid (**\*DEFORMABLE\_TO\_RIGID**) switch with selective or consistent mass scaling present. When elements switched from deformable to rigid (or the other way around) in a model that used selective mass scaling (SMS) or consistent mass scaling, the velocities and inertia computed at the switch point were incorrect, leading to wrong motion of the newly formed rigid/deformable body.
9. Avoid problems with too many entries in **\*INCLUDE\_PATH** or **\*MODULE\_PATH** by increasing the limit for the number of total cards defining paths in a model from 50 to 100 and adding an error check for more than 100 entries.
10. Made a few fixes for **\*DEFINE\_STOCHASTIC\_VARIATION**. This feature was not working when PID refers to a part set with more than eight parts. VAR\_S/F = 3 or 4 (curves) were not working in single precision since December 2023.
11. Fixed combination of **\*INCLUDE\_TRANSFORM** and **\*DEFINE\_TABLE\_{X}D** with  $X \geq 3$ . IDFOFF was not correctly applied to the table IDs on Card 2.
12. Fixed issue for **\*PARAMETER\_EXPRESSION+** and **PRMR = "C"**. With the long format, the expression string could be 140 characters, but there was an internal limit of 72 characters.