

R16.0.0 Release Notes

March 11, 2025

- 1 Release notes revision
- 2 License
- 3 Documentation
- 4 Notes
- 5 New
 - 5.1 Acoustics, NVH, and Frequency Domain
 - 5.2 Airbags
 - 5.3 ALE
 - 5.4 Dual CESE Compressible Fluid Solver
 - 5.5 EM (Electromagnetic Solver)
 - 5.6 Forming Analysis
 - 5.7 ICFD (Incompressible Fluid Solver)
 - 5.8 Initial Cards
 - 5.9 Isogeometric Analysis (IGA)
 - 5.10 Load Cards
 - 5.11 Materials and Equations-of-State
 - 5.12 Multiscale
 - 5.13 Output
 - 5.14 Peridynamics
 - 5.15 SPG (Smooth Particle Galerkin)
 - 5.16 User-Defined Features
 - 5.17 Miscellaneous
- 6 Improvements and Enhancements
 - 6.1 Acoustics, NVH, and Frequency Domain
 - 6.2 Adaptivity
 - 6.3 Airbags
 - 6.4 ALE
 - 6.5 Boundary Cards
 - 6.6 Constraint Cards
 - 6.7 Contact
 - 6.8 Control Cards
 - 6.9 Discrete Element Method
 - 6.10 Dual CESE Compressible Fluid Solver
 - 6.11 Elements
 - 6.12 EM (Electromagnetic Solver)

- 6.13 Forming Analysis
- 6.14 ICFD (Incompressible Fluid Solver)
- 6.15 Implicit (Mechanical) Solver
- 6.16 Initial Cards
- 6.17 Isogeometric Analysis (IGA)
- 6.18 Materials and Equations-of-State
- 6.19 Mesh Cards
- 6.20 MPP
- 6.21 Output
- 6.22 Seat Belts
- 6.23 Sensors
- 6.24 SPG (Smooth Particle Galerkin)
- 6.25 SPH
- 6.26 Thermal Solver
- 6.27 Miscellaneous
- 7 Bug Fixes
 - 7.1 Adaptivity
 - 7.2 Airbags
 - 7.3 Boundary Cards
 - 7.4 CESE Compressible Fluid Solver
 - 7.5 Constraint Cards
 - 7.6 Contact
 - 7.7 Control Cards
 - 7.8 Dual CESE Compressible Fluid Solver
 - 7.9 Elements
 - 7.10 EM (Electromagnetic Solver)
 - 7.11 Forming Analysis
 - 7.12 ICFD (Incompressible Fluid Solver)
 - 7.13 Implicit (Mechanical) Solver
 - 7.14 Initial Cards
 - 7.15 Isogeometric Analysis (IGA)
 - 7.16 Materials and Equations-of-State
 - 7.17 Output
 - 7.18 Restarts
 - 7.19 Sensors
 - 7.20 SPH
 - 7.21 Thermal Solver
 - 7.22 User-Defined Features
 - 7.23 Miscellaneous

1 Release notes revision

This file constitutes revision 0 of the release notes for Ansys LS-DYNA®

version R16.0.0.

2 License

The string “REVISION 16” must appear in the license file to run version R16.0.0. Please contact your distributor or your Ansys sales representative if you need to have your license updated.

3 Documentation

Until such time the R16.0 User’s Manual is posted at <https://lsdyna.ansys.com/manuals/>, 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 implemented for R16.0.0.

4 Notes

The remainder of this file briefly describes new features, enhancements, and bug fixes in version R16.0.0. Other recent releases may also include some of these updates.

We arranged the items by category. Understand 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 keyword ***FREQUENCY_DOMAIN_ACOUSTIC_FREQUENCY** to define output frequencies and interpolation schemes for BEM acoustic calculation. This keyword overrides FMIN, FMAX, and NFREQ defined in ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**.

5.2 Airbags

1. Introduction of the new Continuum-based Particle Gas (CPG) solver: A particle-based compressible gas solver for accurate airbag deployment simulations. See ***AIRBAG_CPG** and related keywords.
2. Added ***AIRBAG_SALE**. It is a wrapper keyword intended for easily setting up an airbag inflation model. It contains new algorithms for solving “contact locking” and alleviating constrained flow in long tubes.

The following keywords (only available for S-ALE) are related component keywords to ***AIRBAG_SALE** added in this release:

- ***MAT_ALE_GAS_MIXTURE_ADV**. It defines inlet gas properties and can hold more than eight gas species.
- ***ALE_STRUCTURED_POINT_SOURCE**. It gives the inlet gas orifices.
- **ABEXT** and **ABINT** keyword options to ***ALE_STRUCTURED_FSI**. They invoke special algorithms exclusively used in FSI with the exterior and interior parts of airbags.

5.3 ALE

1. Added ***DATABASE_SALE**, which has two flags to control S-ALE mesh dumping and the post-processing database output.

5.4 Dual CESE Compressible Fluid Solver

1. Added ***DUALCESE_MAT_LIQUID** to support simple models of viscous liquids.
2. ***DUALCESE_GRAVITY** is a new keyword that allows liquids to experience the load provided with ***LOAD_BODY...** With this keyword, gravity source terms are active for the following 2D and 3D solvers: single fluid with (FSI-IBM or FSI-MMM) or without FSI.
3. Added ***DUALCESE_FSI_EXCLUDE_PARTSET** as a convenience tool for excluding sets of structural parts created with ***SET_PART** from the FSI calculation when using the dual CESE IBM-FSI solvers.

5.5 EM (Electromagnetic Solver)

1. Introduced a time-based periodic solver (**EMSOL = 5** on ***EM_CONTROL**) that is similar to the time-based inductive heating solver, except it

only takes resistive effects into account (no induction/diffusion). This solver provides a very fast solve in resistive heating cases that involve periodic behavior.

5.6 Forming Analysis

1. Added keyword ***DEFINE_FORMING_ADAPTIVE_PART_INITIAL**. This keyword is for refining blank parts in local spot regions at time zero. It is intended for refining the mesh of the blank in small regions for spot welding simulations or where the blank contacts rigid pins in gravity loading processes.

5.7 ICFD (Incompressible Fluid Solver)

1. Introduced a viscoelastic solver. See ***ICFD_MODEL_VISCOELASTIC** to set viscoelastic properties. See also ***ICFD_BOUNDARY_PRESCRIBED_VISCOELASTIC**.
2. Introduced a higher-order level-set method. See SLLS of ***ICFD_CONTROL_ADVECTION** and SRL of ***ICFD_CONTROL_LEVELSET**.
3. Introduced the Resistive Immersed Implicit Surface (RIIS) method (**IMMTH = 2** on ***ICFD_CONTROL_IMMERSED**).

5.8 Initial Cards

1. Added ***INITIAL_STRAIN_BEAM**. When an input deck includes ***DATABASE_DISBOU** and requests writing a dynain file, the output dynain file now includes ***INITIAL_STRAIN_BEAM** to enable the transfer of the discrete beam displacements across multi-stage analyses.

5.9 Isogeometric Analysis (IGA)

1. Added in-core refinement for structured isogeometric shells and solids. ***IGA_REFINE_SHELL** and ***IGA_REFINE_SOLID** enable specifying various refinement types for structured isogeometric shells and solids defined using the ***IGA_SHELL** and ***IGA_SOLID**, respectively.
2. Support subcycling (***CONTROL_SUBCYCLE_{K}_{L}** and ***CONTROL_SUBCYCLE_MASS_SCALED_PART**) for IGA solid elements (***IGA_SOLID**). These techniques are suited for reducing the

computational cost for models involving large spatial variations in mesh density and/or material characteristics.

5.10 Load Cards

1. Added ***LOAD_EXTERNAL_VARIABLE** for defining the spatial and temporal distribution of an external variable on a nodal basis. Input can be based on tabular data for nodes or node sets, or data provided by an LSDA file. The keyword input references part IDs and material property indices. The material properties referenced by material index and part are then determined based on the external variable, meaning the external variable serves as an independent parameter for the respective load curve or table evaluation. The supported materials are ***MAT_106**, ***MAT_251**, ***MAT_254**, ***MAT_To8**, and ***MAT_T10**.

5.11 Materials and Equations-of-State

1. Added basic incremental failure model: ***MAT_ADD_BASIC_INCREMENTAL_FAILURE**. It is a simplified version of GISSMO with a smaller input parameter set and automatic failure curve/surface generation. It is intended as an alternative to something like “FAIL” in ***MAT_024** (which is too simple) and a fully calibrated GISSMO or DIEM (which is sometimes too elaborate).
2. Added ***MAT_ADD_EXTVAR_EXPANSION** to define material expansion based on the evolution of an external variable (***LOAD_EXTERNAL_VARIABLE**). It accounts for nonlinear (load curve) and orthotropic behavior. This expansion property is available for solid formulations 1, 2, -1, and -2 and shell formulations 2, 16, and -16.
3. Added a new glass strength prediction model by Rudshaug et al. (2023) invoked with ***MAT_GLASS_SPM** (***MAT_280_SPM**). It is a Monte-Carlo-based fracture-initiation predictor that combines the theories of linear elastic fracture mechanics and sub-critical crack growth to generate a representative sample of virtual glass plates that are monitored during the simulation. The model results in representative fracture strength distributions that span the probable fracture initiation instances with respect to time, location, and stress level.
4. Added new material ***MAT_317**/***MAT_RRR_POLYMER**.

5.12 Multiscale

1. Added solid-in-solid immersed coupling to ***INCLUDE_COSIM**, where the local solid model occupies the same space as the global solid model at the coupling interface. The set of nodes in the local model follows the translational motion of the set of solid elements in the global model. In return, the nodes of the global solid elements obtain the constraint forces. To use this feature, specify the flag ITS as 3 in both ***SET_SOLID** and ***SET_NODE**.

5.13 Output

1. Select visualization data, including for CPG airbags and SPH, can be written to an LSDA binary file named <JOBID>.d3dat, where <JOBID> is the job ID provided in the run. ***DATABASE_BINARY_D3DAT** specifies the output frequency of this data.

5.14 Peridynamics

1. Made performing a thermally-induced residual stress analysis available for peridynamics. This analysis is performed during dynamic relaxation and requires providing the temperature field with a function or a file called peritprofile.txt. See the manual page for ***MAT_292** for details.

5.15 SPG (Smooth Particle Galerkin)

1. Added IMERGE to ***ISPG_CONTROL_ADAPTIVITY** to activate a merging algorithm for parts that are sufficiently close together. In MPP, this merging feature requires parts to be on the same processor. To do this, use ***ISPG_DEFINE_MERGER_MPP**.
2. Added the keyword ***ISPG_DAMPING_SURF** to define Rayleigh damping related to fluid surface tension on all the ISPG parts.
3. ISPG 3D supports a new curing-degree-dependent viscosity model via the keyword ***ISPG_MAT_CASTRO_MACOSKO_CARREAU**.
4. Added new keyword ***ISPG_CONTROL_D3ISPG** to control outputting the curing, viscosity, temperature, stresses, and strain rates for ISPG.
5. For adaptive ISPG, support thermal-flow coupled analysis. We added the following features for this:
 - Added ***ISPG_CONTROL_SOLUTION** to specify the analysis

solution procedure when performing combined thermal and flow analysis.

- Added ***MAT_THERMAL_ISPG** to define isotropic thermal properties.
- Added keyword ***ISPG_INITIAL_TEMPERATURE_SET** to define initial nodal point temperatures using ISPG part ID.
- Added keyword ***ISPG_BOUNDARY_CONVECTION_SET** to apply a convection boundary condition on an ISPG part for a coupled thermal and flow analysis.
- Extended ***DEFINE_ISPG_TO_SURFACE_COUPLING** to model heat exchange between the ISPG and structural parts through a heat transfer coefficient (HTC).

5.16 User-Defined Features

1. Added ***USER_GET_HISTORY** for gathering material history variables on selected elements.

5.17 Miscellaneous

1. Added ***DEFINE_CABLE** to simplify the creation and data collection of electric cables for failure modeling.
2. When using `dynain` and `dynain.lsd` files to transfer the state of a system between simulations, it is important to provide reference configuration information for hyperelastic material models. Providing this information is particularly tricky when switching between implicit and explicit, for instance, or if the way the deformation gradient is stored changes because of a different set of control parameters. To this end, we introduced ***NODE_REFERENCE**, ***ELEMENT_SOLID_REFERENCE**, and ***ELEMENT_SHELL_REFERENCE** that are part of the `dynain(.lsd)` content whenever hyperelastic materials are present in a model. These keywords allow for a seamless approach to dealing with hyperelasticity in a **dynain** context, meaning that the `RFLAG` on ***INTERFACE_SPRINGBACK_LSDYNA** becomes redundant. Users will typically never insert these keywords in a simulation, but previous simulations always generate them. Seamless transfer requires properly setting up the first simulation, for which the reference configuration typically coincides with the configuration at time zero or is specified using ***INITIAL_FOAM_REFERENCE_GEOMETRY** for solids or ***AIRBAG_REFERENCE_GEOMETRY** for shells.

6 Improvements and Enhancements

6.1 Acoustics, NVH, and Frequency Domain

1. Improved SMP load balancing for METHOD 2, 211, 3, and 311 on ***FREQUENCY_DOMAIN_ACOUSTIC_BEM**. These changes improved performance, particularly for a large number of threads/cores.
2. Support IGA parametric points (***IGA_POINT_UVW**) and IGA physical faces (***IGA_FACE_XYZ**) for setting excitation loads on ***FREQUENCY_DOMAIN_SSD** (see NTYP on Card 7).
3. Extended transient Enight animation support (MASEPLT = 4 on ***CONTROL_ACOUSTIC_SPECTRAL**) to 12th-order spectral acoustic elements.
4. Enabled fast computing of the mcf file (***CONTROL_FREQUENCY_DOMAIN**) without writing d3ssd (BINARY = 0 in ***DATABASE_FREQUENCY_BINARY_D3SSD**).
5. Added option NDD = 0 to ***FREQUENCY_DOMAIN_ACOUSTIC_BEM** to use a recommended domain decomposition.

6.2 Adaptivity

1. Implemented an option to match the tied surfaces of two adaptive parts with tied contact in the first adaptive set. See ICONOPT = -1 on ***CONTROL_ADAPTIVE**.
2. Added printout of what triggers the adaptivity in EFG interactive adaptivity.
3. Control extra d3plot output in 3D *r*-adaptive simulation with ADPD3P on ***CONTROL_ADAPTIVE**. Note that this flag existed previously, but the behavior has changed. The default behavior changed from outputting the extra data. Extra data was output starting in R13 without any controls. ADPD3P was added in R15 to turn off outputting this extra data. With R16, this extra data is not output by default and by setting ADPD3P = 1; ADPD3P = -3 (set to not output the data in R15) was removed; and ADPD3P = 2 and 3 were added to output this extra data.

6.3 Airbags

1. Added an automatic checker during the keyword phase to check airbag quality for S-ALE. The results are output to saleabchk.msg.
2. Support TYPE = CURVE on ***SENSOR_CONTROL** with ***AIRBAG_PARTICLE**.
3. Support coupling CPM (***AIRBAG_PARTICLE** with the thermal solver.
4. Support unlimited gas species for ***AIRBAG_PARTICLE**. Internally, there had been a limit of 40 species.
5. Support including solid parts when defining the airbag (SID1 and SID2 on ***AIRBAG_PARTICLE**). Previously, the airbag could only be defined with shell parts. Deleted segments are removed from the coupling.

6.4 ALE

1. Enabled using GEOM = ALL on ***ALE_STRUCTURED_MESH_VOLUME_FILLING** to assign velocity using VID to all S-ALE nodes. Previously, VID worked with all GEOM settings except ALL.
2. Improved algorithm to cure pressure oscillation during the ALE advection process when using ***MAT_ELASTIC_PLASTIC_HYDRO** with ***EOS_TABULATED_COMPACTON**.
3. Support beam erosion for S-ALE FSI in 2D.
4. Implemented using an element set to specify the ALE fluid subjected to the porous flow condition (SIDTYP = 2 on ***LOAD_BODY_POROUS**).
5. If ALE groups fully fill the ambient elements (see ***BOUNDARY_AMBIENT**), MMG = 0 keeps these fillings as sources instead of replacing them.
6. Added command line option map2=[listofmappingfiles.txt] for ***INITIAL_ALE_MAPPING** to map ALE data from several mapping files created by previous ALE runs with the same dimensions and identical ALE groups. Note that listofmappingfiles.txt needs to be a .txt file that lists the names of the mapping files. In this case, map= is not used.
7. Made ***ALE_FSI_TO_LOAD_NODE** available for S-ALE FSI.
8. For ***BOUNDARY_ALE_MAPPING** with INI = 1, write S-ALE data to a mapping file that can be read by ***INITIAL_ALE_MAPPING** and ***BOUNDARY_ALE_MAPPING**.
9. Added OPTION = AMBIENT to ***BOUNDARY_SALE_MESH_FACE** as

a macro for applying ***BOUNDARY_AMBIENT** with **AMBTYP = 4** along the boundary faces.

10. Applying **OPTION = AMBIENT** and **OPTION = NONREFL** on ***BOUNDARY_SALE_MESH_FACE** to the same mesh faces causes the elements attached to the faces to turn into ambient elements with a non-reflecting boundary condition on the faces connecting the ambient elements to the rest of the S-ALE mesh.
11. Enabled moving joined S-ALE meshes.
12. Enabled **DVOL** on ***SET_SOLID_GENERAL** to work for S-ALE elements.
13. Enabled **DVOL** on ***SET_SEGMENT_GENERAL** to work for S-ALE segments.
14. Enabled **DVOL** on ***SET_NODE_GENERAL** to work for S-ALE nodes.

6.5 Boundary Cards

1. For ***BOUNDARY_PWP**:
 - **LCLEAK** now works with the different settings of **ITOTEX** to model water exchange with a reservoir.
 - Added input field **TYPMP** which, if nonzero, changes the effect of **LCPUMP**. It enables the node to try to follow the boundary condition pressure defined by **LC**, **CVAL**, and **ITOTEX**, with **LCPUMP** setting the maximum volume flow rate.

6.6 Constraint Cards

1. Check that there are no conflicts between the IDs used for ***RIGIDWALL_PLANAR** and ***CONSTRAINED_NODAL_RIGID_BODY**.
2. Support solid elements for ***CONSTRAINED_COORDINATE**.
3. Added **TITLE** keyword option to ***CONSTRAINED_EXTRA_NODES** and ***CONSTRAINED_RIGID_BODIES**.
4. Setting **|CMO| = 2** on ***CONSTRAINED_NODAL_RIGID_BODY_SPC** enables applying the constraints or prescribed motion (***BOUNDARY_PRESCRIBED_MOTION**) of the rigid body about a specified coordinate.
5. Added an **SPR3** with a strongly objective formulation (see **MODEL = 3** of ***CONSTRAINED_INTERPOLATION_SPOTWELD**).

6. Added ability to position the spot weld (see SPROFF) between the two sheets for spot welds defined with
`[*CONSTRAINED_INTERPOLATION_SPOTWELD]{.kwstyle}`.
7. Added new search method for ***CONSTRAINED_SPR2** and ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) invoked by flag SPRSRCH on ***CONTROL_CONSTRAINED**. The option includes the nodes inside the search radius and all nodes from elements inside the search radius.
8. Modified SPR2/SPR3 search. The height of the search cylinder (THICK) can be too small if there is a substantial gap or THICK is defined way too large, causing problems with bent sheets. The modified search uses the measured distance with some tolerance instead of using THICK from the input. It is only supported for the recently added option SPRSRCH = 1.
9. Added option STIFF2 < 0 to ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3). |STIFF2| is a load curve ID for shear stiffness as a function of load angle. This curve allows a better fit for the elastic-mixed-mode response.
10. Add damping option, DAMP, to ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) to apply viscous damping.

6.7 Contact

1. Added MPP support for ***TERMINATION_CONTACT**.
2. Added an initialization error termination check for MPP only due to an issue with contact. Before the decomposition, contact segments are mapped to the element they come from. Any not found will cause error termination.
3. Added support for beam elements in segment-to-segment (SOFT = 2) contact. To use beam elements, simply use parts or part sets to define surface A and/or surface B and include beam parts in the surface definition. This works with all contact keywords that are compatible with segment-to-segment contact. However, it must be used with DEPTH options that support edge-to-edge contact, which currently are DEPTH = 5, 15, 25, 35, and 55. If other DEPTH options are input, they are switched internally to DEPTH = 55.
4. Improved segment-to-segment (SOFT = 2) contact when shell element segments contact solid element segments. There were spurious forces in some instances that caused nonphysical deformation.

5. Implemented edge treatment options SRNDE = 1 and 2 (on Optional Card E) for ***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE** in SMP for SOFT = 0 and 1.
6. The ORTHO_FRICTION keyword option in ***CONTACT_..._MORTAR** allows for 3D tables, thereby adding a temperature dependence in addition to sliding velocity and contact pressure.
7. Thermal edge contact is supported and automatically enabled when the underlying mechanical contact is of ***CONTACT_..._MORTAR** type.
8. Support output of tied area to **binout** for ***CONTACT_..._MORTAR_TIED** types.
9. Account for the OFFSET option on ***ELEMENT_BEAM** and NSLOC/NTLOC on ***SECTION_BEAM** when determining contact with ***CONTACT_AUTOMATIC_..._MORTAR**. However, the contact still uses an approximate tubular section.
10. Support ***LOAD_THERMAL_BINOUT** with ***CONTACT_..._MORTAR**.
11. Added MINFO < 0 on ***CONTROL_OUTPUT**, where -MINFO is a time-step frequency for output penetration and memory information from ***CONTACT_..._MORTAR**. This negative option applies also to explicit analysis.
12. Added TIEOPT to ***CONTROL_CONTACT** to invoke constrained tied contacts in explicit algorithms that circumvent some of the shortcomings in the standard implementations. It is mandatory (and automatically applied) for selective mass scaling (IMSCAL ≠ 0 on ***CONTROL_TIMESTEP**) to properly support all tied contact combinations. The primary justification for this option is the completion of tied contact support in selective mass scaling. However, even without that, this option offers some advantages for constrained tied contacts, such as:
 - SMP and MPP constrained tied contacts become identical, each finding the same node-segment pairs.
 - The segment sides of different tied contact interfaces can share nodes, which previously required the MPP GROUPABLE option.
 - The SHELL_EDGE_TO_SOLID_CONSTRAINED_OFFSET option becomes available, which is otherwise silently converted to a SHELL_EDGE_TO_SOLID option without offset.
 - The tied contacts automatically detect the nature of involved nodes, thereby internally converting invalid constraints to more appropriate ones. Thus, “any” tied contact option is valid for “any” situation. For instance, the TIED_SHELL_EDGE_TO_SURFACE contact behaves as a

- TIED_SHELL_EDGE_TO_SOLID for shell-to-solid ties, leaving the nonphysical rotational degree of freedom of the solid untouched.
- For TIEOPT = 2, any occurrence of SHELL_EDGE_TO_SURFACE is converted to SHELL_EDGE_TO_SOLID to avoid coupling between rotational degrees of freedom on the segment side to the translational degrees of freedom on the node side. This option potentially avoids instabilities related to the former option.
13. Added STROBJ to ***CONTROL_CONTACT** to turn on strong objectivity for single surface contacts with SOFT = 0 or 1.

6.8 Control Cards

1. Setting SRTFLG = 1 on ***CONTROL_ACCURACY** causes the parts, contacts, nodal rigid bodies, and elements to be processed in an order sorted by the respective user IDs, regardless of the order that these appear in the input file. This feature helps ensure consistent results if, for some reason, the order of the above entities changes in the input file. SRTFLG can also be activated or deactivated through the command line options SRTFLG=1 or SRTFLG=0. The command line option overrides the value specified in the keyword.
2. Added options ESORT = 11, 12, 13, and 14 to ***CONTROL_SOLID**. These are the same as ESORT = 1, 2, 3, and 4, respectively, except tetrahedrons are sorted to type 13.
3. Added ability to request additional output with ***CONTROL_EOS - USER_LIBRARY** for plotting 3D surface graphs of seslib data.
4. Added the ASCII keyword option to ***CONTROL_EOS_USER_LIBRARY** for reading Los Alamos sesame tables instead of seslib and handling several ASCII table files.
5. Switch to LAMSHT = 3 when LAMSHT = 1 on ***CONTROL_SHELL**. LAMSHT = 1 is only implemented for a few material models, and even for them, it does not seem to work properly.
6. Extended UMCHK on ***CONTROL_MAT** to check for default user-defined friction (only MPP).
7. Reduce memory requirement of detailed energies for materials (MATEN = 2 on ***CONTROL_ENERGY**) by only requesting it after MPP decomposition.

6.9 Discrete Element Method

1. Added defining a box with SBOX to ***DEFINE_DE_TO_SURFACE_COUPLING** to exclude segments belonging to the surface that are outside the box during initialization from the coupling.

6.10 Dual CESE Compressible Fluid Solver

1. Made the NASG EOS (***DUALCESE_EOS_NASG**) available to the single fluid solvers, including both with and without FSI.
2. Added viscous shear to the feedback forces for the viscous 2D and 3D FSI-IBM solver.
3. Introduced a porous media model for supersonic flow with thin-shell porous structures in the dual CESE FSI-IBM and two-phase FSI-IBM solvers. This model solves the Darcy-Forchheimer equation as a jump condition across the porous media interfaces. This new porous media model is available for the following two-phase and single-fluid FSI-IBM dual CESE solvers: 2D, 2D axisymmetric, and 3D.
4. Reversed the modification done in R15 to the local CFL number formula for a new time-step-size calculation. In other words, we restored the R14 treatment of this calculation.

6.11 Elements

1. Added option ASA to ***SECTION_BEAM** for resultant beam type 2. ASA flags automizing the shear area approximation for tubular (SECTION_09) and solid box (SECTION_11) shapes.
2. Extended the STRETCH field in ***CONTROL_SHELL** to check the three edges of triangular shells in the same manner that it checks the diagonals of a quadrilateral. Previously, this option ignored triangular shells.
3. The definition of baseline orientation vectors with ***PART_FIELD** for finite element solids ELFORM = 1 is now supported for the following material models: 22, 23, 26, 33, 40, 54, 55, 58, 59, 91, 92, 103, 104, 122, 126, 133, 142, 143, 157, 199, 213, 215, 219, 221, 233, 245, 261, 262, 263, 264, 278, and 291.
4. Extended to beams with ELFORM = 13 the existing option of using BEAMIP on ***DATABASE_EXTENT_BINARY** to cause writing the

moments at both ends of resultant beams (ELFORM = 2) to d3plot and d3thdt.

5. Enhanced the check for cohesive elements to properly reorientate them so that the orientation matches neighboring elements by changing the connectivity.
6. Added experimental options ISTUPD = 5 and 6 on ***CONTROL_SHELL**. These are the same as 1 and 4, respectively, but associated thickness changes are double precision in single precision binaries for shell elements with ELFORM = 2, 4, and 16. These options should make results more comparable (single precision versus double precision) in long-lasting explicit analyses.
7. Added option to change some cohesive errors to warnings invoked with COHEQC = 10, 11, and 12 on ***CONTROL_SOLID**.
8. Support stiffness proportional damping for cohesive elements. ***DAMPING_PART_STIFFNESS** did not have any effect on them until now. Also, support RYLEN = 2 from ***CONTROL_ENERGY** for cohesive elements to include the damping energy in the internal energy computation.
9. Added warning for the use of pentahedral thick shell (TSHELL) elements:

```
Thick shell element ... is a degenerated hexahedron (nodes N3=N4
and N7=N8). Such pentahedral thick shells are currently not
supported and should therefore be avoided.
```

6.12 EM (Electromagnetic Solver)

1. Added CIRCTYP = 30 to ***EM_CIRCUIT** to include Diode modeling in the circuit. It is available with the Eddy current solver only (EMSOL = 1 in ***EM_CONTROL**).
2. An imposed current condition (see CIRCTYP IN ***EM_CIRCUIT**) when using the Resistive heating solver is now done via a Neumann boundary condition rather than an internal transformation into a voltage condition (after doing an initial solve to get the mesh resistance and using Ohm's law). We made this change after realizing that the previous way could cause issues in cases that involved multiple circuit branches. Resistance and current output values may be affected.
3. Added the option to turn off the time interpolation (INTERP on ***EM_CONTROL_COUPLING**) of the EM forces between two EM steps. This can be useful in magnetostatic steps (especially rotating problems) where

the default linear time interpolation of the EM force at each mechanical step can introduce a larger error than having no interpolation at all.

4. Added an optional third line to [***EM_CONTROL_COUPLING**]{kwstyle] for providing three node IDs defining a local coordinate system for the EM solver to follow during the force interpolation phase given to the mechanical solver. This feature may be recommended in rotating magnetostatic problems for increased accuracy.
5. Added an optional second line to ***EM_CONTROL_SOLUTION** for defining two local coordinate systems and specifying that BEM matrix recomputations can only occur based on specific translation and rotation criteria.
6. Fixed bug in magnet ID ordering (***EM_PERMANENT_MAGNET**). Previous versions assumed the user defined the magnet IDs in their keyword file in an ordered fashion (from small ID to large) which is not always the case.
7. For ***EM_RANGLES_BATMAC**, added the option to have temperature-dependent equilibrium voltage as a function of state of charge tables.
8. For Randles circuits (***EM_RANGLES_...**), state of charge values in battery cells are now hard capped at 100.
9. Enabled specifying a part ID instead of segment sets for the associated Rogowsky coil in 2D axisymmetric Eddy current problems (see PID on ***EM_CIRCUIT_ROGO**. This ability simplifies creating the input for this case.
10. Added **CONTYPE = 2** to ***EM_CIRCUIT_CONNECT** to have the circuit connect constraint take precedence over the constraint (if it exists} defined in ***EM_CIRCUIT** for the second circuit. In contrast, for **CONTYPE = 1**, the constraint for the second circuit takes precedence over the circuit connect constraint.
11. Support beam elements with the eikonal solvers (solver types 14 and 15 on ***EM_CONTROL**) so that beams can be coupled with tetrahedral elements.
12. For eikonal models (***EM_EP_EIKONAL**), **SIGMA** in the data cards of EM materials ***EM_MAT_001** and ***EM_MAT_003** corresponds to the wave velocity. Note that ***EM_MAT_001** is supported for beam elements, while ***EM_MAT_003** is supported for tetrahedral elements.
13. Added ionic model using splines instead of solving the ODEs at each node. See **IONSOL = 2** on ***EM_CONTROL_EP**. One spline is built for each

ionic model once and for all at the beginning of the run, and is then used at each node, starting at the activation time of the node. This method can be used either for the eikonal solvers (EMSOL = 14 or 15 in ***EM_CONTROL**) or the mono/bi domains (EMSOL = 11 or 12 on ***EM_CONTROL_EP**). It saves a lot of cpu time in the ionic model part. An optional Card 2 on ***EM_CONTROL_EP** can be included to give the end time (SPLTEND) and time step for building the spline (SPLDT).

14. Added ability to add delays between 2 depolarizations/repolarizations on ***EM_EP_ISOCH** to avoid detecting successive repol/depol in cases where the transmembrane potential oscillates.

6.13 Forming Analysis

1. In the mapping process of ***INCLUDE_STAMPED_SOLID_TO_SOLID**, perform strain averaging around the neighboring elements.

6.14 ICFD (Incompressible Fluid Solver)

1. For ***ICFD_BOUNDARY_WINDKESSEL**, added detailed Windkessel circuits types 7, 8, 9, 10. We also reworked this feature to make it more robust and less time-step-dependent. We added WTYPE = 6 to specify a user-defined model.
2. Added the radius factor to the flux output (see ***ICFD_DATABASE_FLUX**) in 2D axisymmetric analyses to make it consistent with 3D output.
3. For ***ICFD_CONTROL_ADAPT**, added an option, DT, to control the remeshing frequency via a time step (transient analysis only).
4. For the steady state solver, a velocity boundary condition is now considered constant during the steady state analysis (the first value of the ***DEFINE_CURVE** will be picked).
5. For weak 2-WAY FSI with the implicit structural solver (OWC = 3 on ***ICFD_CONTROL_FSI**), enabled the implicit solver to perform substepping until convergence with a single CFD step.
6. For ***ICFD_CONTROL_ADAPT**:
 - Improved mesh size stability and smoothness in adaptive meshing.
 - Introduced a level-set-based adaptation algorithm. See VAR < 0.
 - The mesh size in the adaptation algorithm can be computed based on relative errors. See MTH = 3 and 4.

- Added new option VAR = 5 to only refine close to immersed interfaces.
7. For ***ICFD_CONTROL_TURBULENCE**:
 - Added a linear blending between the laminar and turbulent zones with the log law (WLAW = 1).
 - Made stability improvements in the k - ϵ model.
 8. Support the advection of two species instead of just one (see ***ICFD_MODEL_SPECIES_TRANSPORT**, ***ICFD_BOUNDARY_PRESCRIBED_SPTRANSP_CONC**, and ***ICFD_INITIAL_SPTRANSP**).
 9. Support initializing species concentrations by ***ICFD_PARTs** and ***ICFD_PART_VOLs** (see ***ICFD_INITIAL_SPTRANSP**) in the species transport flow solver. Previously, initialization only worked globally.
 10. For ***ICFD_MODEL_POROUS**, added support to have hybrid porous/non-porous membrane/fabrics in the same model. A mixing of porous and non-porous deforming embedded shells is now allowed.

6.15 Implicit (Mechanical) Solver

1. The option ITORM on ***SECTION_BEAM** is now supported for implicit transient analyses. Before, it could only be used for eigenvalue analyses.
2. Made the sectoral symmetry method (EIGMTH = 111 on ***CONTROL_IMPLICIT_EIGENVALUE**) available in MPP. This method only works for jobs running on one processor, but it no longer requires SMP to run and gives the same results as for SMP.
3. For LSOLVR = 30 (MUMPS) on ***CONTROL_IMPLICIT_SOLVER**:
 - Upgraded MUMPS from version 5.6.2 to 5.7.3.
 - Support the solver in PGI builds.
4. Support ORDER = 3 (ParMETIS) on ***CONTROL_IMPLICIT_SOLVER** in PGI builds.
5. For METHOD = 102 (LOBPCG) on ***CONTROL_IMPLICIT_EIGENVALUE**:
 - Improved performance (up to 30% faster for large problems and/or large numbers of requested modes).
 - Made interval computations (LFLAG/RFLAG) available in MPP.
 - Number modes in the eigout file from 1 instead of the index in the entire spectrum for interval computations (LFLAG/RFLAG).

- Support it in PGI builds.
- 6. Support modal analysis (***CONTROL_IMPLICIT_EIGENVALUE**) with rotational dynamics terms (***CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS**) for MPP executables when the number of MPI ranks is 1. Full MPP capability is the subject of future work.
- 7. Made ***DATABASE_ADAMS** available again. This feature outputs a Modal Neutral File, d3mnf, for later use with Hexagon's Adams software. It was available up to R7 upon customer request and required an external library. It is now available with all MPP builds without additional software.
- 8. Added MXMODES to ***CONTROL_IMPLICIT_INERTIA_RELIEF** for setting the maximum allowed eigenvectors to be computed.
- 9. Added printing information about the JFNK nonlinear solver (NSOLVR = 13) if NLPRINT = 3 on ***CONTROL_IMPLICIT_SOLUTION**.
- 10. Output time history results for beam elements undergoing modal dynamics (***CONTROL_IMPLICIT_MODAL_DYNAMIC**) to elout and d3thdt. Previously, the beam results were written as zero to both elout and d3thdt.
- 11. For implicit, imposing spc on too many rigid body nodes results in an overdetermined set of constraints. Previously, doing this would result in an error termination. Now, we process these constraints during the initialization and translate them into the equivalent combination of nodal and rigid body constraints that allow the simulation to continue with expected results.
- 12. For implicit, stacked use of tied contacts (think laminates/composites) may result in an overdetermined set of constraints that could lead to an error termination or automatic deletion of important ties just to start the simulation. Now, we use a dependency graph to delete only the tied contacts that must be deleted, resulting in a more expected behavior. It may still result in an error termination if the dependencies cannot be resolved.
- 13. IACC = 1 on ***CONTROL_ACCURACY** is now the default for nonlinear implicit simulations. To recover the old behavior, set IACC = -1.
- 14. Equipped incompressible control volumes (***DEFINE_CONTROL_VOLUME**) with two new features:
 - Enable providing a source term.
 - Exclude resulting forces on flow areas because the flow areas typically represent separators between domains for enclosing volumes and have no physical counterparts.

6.16 Initial Cards

1. Implemented ***INITIAL_FOAM_REFERENCE_GEOMETRY** for axisymmetric elements (ELFORM = 14 and 15 on ***SECTION_SHELL**).
2. Implemented initializing velocities with ***INITIAL_VELOCITY_GENERATION_START_TIME** under a switch condition (***SENSOR_SWITCH**). When $STIME < 0$, $|STIME|$ refers to a sensor switch ID.
3. Implemented defining the high explosive as a part or element set (HETYP = 1 or 2) on ***INITIAL_DETONATION_GEOMETRY**.
4. Added selecting elements to ignite with a box for ***INITIAL_DETONATION_GEOMETRY**.

6.17 Isogeometric Analysis (IGA)

1. Added support for implicit MPP contact for trimmed IGA solids (***IGA_SOLID**).
2. Added support for MPP eigenvalue extraction for trimmed IGA solids (***IGA_SOLID**).
3. Map the friction definition (***DEFINE_FRICTION_TABLE**) of IGA contact parts to the related interpolation element parts to properly reflect the friction behavior in the contact evaluation.
4. In case multiple faces (***IGA_FACE_XYZ**) are defined on the same underlying surface patch (***IGA_2D_NURBS_XYZ**) and the individual faces are topologically connected with ***IGA_EDGE_UVW** and ***IGA_EDGE_XYZ**, check that the generated interpolation mesh for the individual faces matches at the shared interface and that the corresponding interpolation nodes are merged. This check solves an issue that may appear if the faces are in the same contact definition, and edge contact may happen between the free boundaries of the individual faces.
5. Add support for the parameters THICK, PSTRN, STRAIN, and STRESS in ***INCLUDE_STAMPED_PART** for IGA shells (***IGA_SHELLS**).
6. Added support for ***INCLUDE_TRANSFORM** for IGA-related keywords.
7. Add support to use ***IGA_POINT_UVW** for trimmed IGA solids (***IGA_SOLIDS**). This can be used to define constrained nodal rigid bodies or similar boundary conditions.

8. Support ***CONSTRAINED_SPR2** for (trimmed) IGA shell/solid (***IGA_SHELL/SOLID**) parts.
9. Improve force sharing for shared interpolation nodes along MPP domain boundaries for IGA solid parts (***IGA_SOLID**).
10. Add support for ***LOAD_THERMAL** for IGA solid elements (***IGA_SOLID**).
11. Properly map IGA shell (***IGA_SHELL**) thickness to related interpolation elements to have the correct thickness reflected in contact.

6.18 Materials and Equations-of-State

1. For ***MAT_001** / ***MAT_ELASTIC**
 - Changed the bound of Poisson's ratio from (0,0.5) to (-1,0.5).
 - Allow the Young's modulus to be a function of pressure.
2. Setting $|CMO| = 2$ on ***MAT_RIGID** (***MAT_020**) enables applying the constraints or prescribed motion (***BOUNDARY_PRESCRIBED_MOTION**) of the rigid body about a specified coordinate.
3. Enabled **IRATE** on ***CONTROL_IMPLICIT_DYNAMICS** for Hughes-Liu beam elements (**ELFORM = 1**) with ***MAT_024**.
4. For ***MAT_024**, allow simultaneous use of keyword options **_STOCHASTIC** and **_2D** for shell elements.
5. Enabled fabric material formulation 24 (**FORM** on ***MAT_034**/***MAT_FABRIC** and ***MAT_B01**/***MAT_SEATBELT**) to be used in implicit analysis. Previously, parts input with formulation 24 were internally switched to formulation 14.
6. Made the Poisson's ratio behavior of fabric material formulation 24 (**FORM** on ***MAT_034**/***MAT_FABRIC** and ***MAT_B01**/***MAT_SEATBELT**) more accurate for large strains. Results for problems where fabric material strains remain small should change little.
7. Added implicit tangent to ***MAT_058** (***MAT_LAMINATED_COMPOSITE_FABRIC**).
8. Made ***MAT_074** work with ***INITIAL_STRESS_BEAM**.
9. Enabled ***MAT_083_PATH_DEPENDENT** to work with ***INITIAL_FOAM_REFERENCE_GEOMETRY**.

10. Made the latest features and bug fixes in ***MAT_106** available for type 13 solid (tetrahedron) elements.
11. Added support for plastic strain tensor output with STRFLG = 11 on ***DATABASE_EXTENT_BINARY** for ***MAT_106**, ***MAT_123**, ***MAT_224**, and ***MAT_251**.
12. Added flag PDV to ***MAT_148** / ***MAT_GAS_MIXTURE**. This flag sets the method for computing the element energy update. The default (PDV = 0) and original implementation is the ideal gas gamma law, which violates energy conservation when temperature-dependent C_v/C_p are used with ALE formulations. PDV = 1 invokes pressure work for this calculation.
13. For ***MAT_SOIL_SANISAND** (***MAT_207**), improved stability for elements at low pressure.
14. Updated ***MAT_213** / ***MAT_COMPOSITE_TABULATED_PLASTICITY_DAMAGE** to version 1.3.7. The new version:
 - Supports all thick shell formulations.
 - Includes a simplified Material Model (SMM). The simplified material model is a modified version of the existing linear ***MAT_213** that includes post-peak damage curves (includes damage after linear elastic behavior). It is implemented for solid, thin shell, and thick shell elements (only ELFORM = 5 for thick shells). The simplified material model can be used for materials with linear elastic behavior (tension-compression asymmetry handled). Setting all flow rule coefficients to zero activates it. The simplified material model has no rate and temperature dependency and no plasticity. The simplified material model is used to cut the computational cost.
 - Performs Poisson's ratio checks to determine the compatibility with material orthotropy. Poisson's ratio checks are skipped when all three Poisson's ratios are zero to avoid an error message.
 - Issues a warning message instead of an error message when the effective stress values are not in ascending order after curve conversion for stress-strain and damage curves.
15. For ***MAT_LAMINATED_FRACTURE_DAIMLER_PINHO** (***MAT_261**), improved the behavior when the compressive stress reached XC prior to the actual failure criterion. In this case, we start computing damage values (dkink) to limit the stress within bounds, but we did not compute the failure strain in case the failure criterion is finally met, which led to a brittle failure due to lack of damage evolution. Now, we compute the failure strain in this scenario before reaching the failure criterion, which enables damage evolution and, thus, softening behavior before failure.

16. Added parameter DOSPOT to ***MAT_270**, which activates thinning of tied shell elements when SPOTHIN > 0 on ***CONTROL_CONTACT**.
17. Added grouping option, GRPFT, to ***MAT_280** to scale down the tensile strength of several parts with FTSCAL at the same time.
18. Added plasticity to ***MAT_291** (***MAT_SHAPE_MEMORY_ALLOY**).
19. Added the FIBAND (FIBer distribution based ANisotropic Damage) model to ***MAT_DMN_COMPOSITE_FRC** (***MAT_303**) for predicting the damage and failure of injection-molded short-fiber-reinforced composites with any fiber distribution.
20. Enabled support for using ***MAT_305** / ***MAT_HOT_PLATE_ - ROLLING** with 1-point pressure tetrahedrons (type 13).
21. For ***MAT_307** / ***MAT_GENERALIZED_ADHESIVE_CURING**:
 - Added diffusion control for curing kinetics. Different options are available that can be combined with curing kinetics models 2, 3, and 4.
 - Allow initializing degree of cure with ***INITIAL_STRESS_SOLID** cards with IDOC < 0.0.

6.19 Mesh Cards

1. Implemented automatically renumbering the ID of ***MESH_VOLUME** if the same ID exists in ***MESH_SURFACE_ELEMENT** already to avoid output confusion. This change is only for output purposes. Results are unaffected.
2. For ***MESH_SIZE_SHAPE**, a ***DEFINE_FUNCTION** can now be used to define the mesh size as a function of the (x, y, z) coordinates. This capability enables defining a gradual coarsening or refinement.

6.20 MPP

1. Added environment variable LSTC_DIR_LOCAL to specify the local directory for scratch files from an environment variable. This directory should be on a local disk on each processor for performance reasons. Note that the pfile command local has priority over this environment variable.

6.21 Output

1. Include energy due to ***INTERFACE_LINKING_SEGMENT** in the external work reported to the glstat file.
2. Added ***DATABASE_RCFORC_DR** to output rforc during dynamic relaxation.
3. The mass and inertia properties output to d3hsp are now computed using double precision regardless of binary precision.
4. ***DATABASE_BINARY_ISPHFOR** now supports the FILE keyword option.
5. Added ***DATABASE_FSI_DR** to save the dbfsi data from the 1st dynamic relaxation phase.
6. Enabled converting the binary file created with ***DATABASE_BINARY_FSI_LNK** into a keyword file with ***LOAD_SEGMENTS** and ***DEFINE_CURVES** generated from the data in that file. To generate this keyword file, run LS-DYNA with only the `fsilnk=[fsifile]` command line option, where fsifile is the name of the binary file.
7. For ***DATABASE_D3MAX**:
 - Added support for rigid body solid elements in d3max.
 - Added option to use original geometry instead of deformed geometry (OUTPUT = 6 and 7) in d3max output.

6.22 Seat Belts

1. Add the contribution of the shell force to the section force when the shell is a 2D seat belt.
2. For MPP decompositions, every 2D seat belt is assigned a separate core to speed them up.

6.23 Sensors

1. Enable ***SENSOR_DEFINE_MISC** to trace seat belt slip rings (MTYPE = SLIPRING).
2. Added the options PULLOUT and FORCE for TYPE = RETRACTOR on ***SENSOR_DEFINE_MISC** to trace the retractor's total pull-out and the force of the attached belt, respectively.
3. Added TFIRE to ***SENSOR_CPM_AIRBAG** to control the activation

time of the mass flow rate and temperature curves.

6.24 SPG (Smooth Particle Galerkin)

1. Made coupling between SPG particles and a surface (***DEFINE_SPG_TO_SURFACE_COUPLING**) available in MPP.
2. Added option **SFPN** to ***DEFINE_ISPG_TO_SURFACE_COUPLING** to scale the thickness of the solid structure that is coupled with the ISPG fluid. This option improves contact detection if the structure is very thin and the length of the solid elements along the normal direction is much smaller than the length in the in-plane direction.
3. Support the **TITLE** keyword option for ***SECTION_ISPG**, ***MAT_ISPG_...**, ***MAT_THERMAL_ISPG**, ***DEFINE_ISPG_TO_SURFACE_COUPLING** for ISPG adaptivity.

6.25 SPH

1. Repeated solid or shell part IDs in ***DEFINE_ADAPTIVE_SOLID_TO_SPH** will now raise an error. Specifying the same parent part more than once will cause: (1) overlapping particles if **NQ** is the same in both instances, or (2) particles of different sizes that are too close.
2. For ***DEFINE_ADAPTIVE_SOLID_TO_SPH**, select history variables are now scaled by volume ratios between SPH and their parent solid/shell element. This lack of scaling could cause simulations to destabilize when $NQ > 1$ while using some materials or equations of state.

6.26 Thermal Solver

1. Added field **NIDOFF** to ***LOAD_THERMAL_BINOUT**. It represents an offset in the global nodal IDs between the **lsda** file and the model and makes it possible to use the thermal results for a component that is included in the model with a nodal ID shift through ***INCLUDE_TRANSFORM**.

6.27 Miscellaneous

1. Enhanced internal treatment of curves defined with ***DEFINE_CURVE**. These curves have a regularly spaced internal representation. With this

update, the way these curves are stored has been changed, resulting in significant memory reductions, particularly when different curves have different values of LCINT. Evaluating these curves is also more efficient.

For those implementing user-defined materials, note that the curve array `crv` is NOT AVAILABLE anymore beginning with R16. Please use new curve/table evaluation routines instead:

- `curveval`
 - `curveval_v`
 - `tableval`
 - `tableval_v`
2. Improved implicit co-simulation with ***COSIM_FMI_INTERFACE** by exchanging data in every iteration.
 3. Added command line option `stdout=` to control the messages to stdout. See the manual for details.
 4. Added parameter `IORIEN` to ***DEFINE_PRESSURE_TUBE** to use the third beam node or ***BEAM_ORIENTATION** for solid/shell orientation.
 5. ***DAMPING_FREQUENCY_RANGE_DEFORM** now works with `ELFORM = 13` beam elements.

7 Bug Fixes

7.1 Adaptivity

1. Fixed a bug that may cause memory errors when `INMEMORY` is on and `ADPTYP` is greater than 2 in ***CONTROL_ADAPTIVE**.
2. Fixed bugs in nodal mass calculations when `INMEMORY` is on in ***CONTROL_ADAPTIVE**. These bugs affect solution accuracy, especially in metal forming deep drawing simulations.
3. Fixed bugs in the initialization of load curves that may cause memory errors after an adaptive cycle.
4. Fixed bugs in the initialization of contacts that may cause errors when `INMEMORY` is on in ***CONTROL_ADAPTIVE**.
5. Fixed bugs in shell size evaluation that may induce inconsistencies in adaptivity criteria between in-core and out-of-core adaptivity.

7.2 Airbags

1. Fixed bug when the inflator mass flow rate is input with a total mass flow rate curve and species based on mole fraction for `iair = 2` and `4`.

7.3 Boundary Cards

1. Fixed a bug for `LCBCHK` on `*BOUNDARY_PRESCRIBED_MOTION_-SET_BOX`. The bug was due to mistakenly assuming all related nodes to be inside the checked box initially, leading to unexpected behavior.
2. Fixed issue with `*BOUNDARY_PWP` during a full-deck restart. The boundary conditions were not being obeyed in the restart run.
3. For `*BOUNDARY_FLUIDM`:
 - Fixed an incorrect error message issued when coincident segments are encountered. It would mistakenly issue an IGA shell error message.
 - Corrected flawed auto-curvature estimates when they were requested in MPP solutions.
 - Corrected MPP `nrank > 1` solutions. Indexing of owned implicit degrees of freedom was wrong. Typically, the eigenvalue solution would crash or not converge.

7.4 CESE Compressible Fluid Solver

1. For the CESE chemically-reacting flow solver:
 - Corrected some time step calculation issues,
 - Adjusted the code to not allow the internal energies to go below the minimum value represented in the user's chemkin data, and
 - Corrected an IBM FSI bug in the CESE IBM FSI 2D axisymmetric version of the solver.

7.5 Constraint Cards

1. Fixed a bug in the `*CONSTRAINED_BEAM_IN_SOLID` general slip-force law (`AXFOR = 999`) to consider negative slip.
2. Fixed issue that caused very slow initialization in models that have BOTH a large number of nodal rigid bodies (`*CONSTRAINED_NODAL_-RIGID_BODY`) AND a large number of `*BOUNDARY_PRESCRIBED_MOTION` cards.

3. Fixed problem with ***CONSTRAINED_SPR2/SPR3** and ***SET_N-ODE_ADD**. The following error occurred: node set for spr # 123 is not found. Generating the visualization beams somewhat later resolves it.

7.6 Contact

1. Fixed issue with MPP contact near the decomposition boundary between processors. It occurred if some of the contact nodes were involved in a ***CONSTRAINED_TIED_NODES_FAILURE** constraint that failed. The failure was not being properly treated.
2. Fixed issue for MPP groupable contacts with birth times. If some processors ended up with no currently active groupable contacts, while others did have currently active groupable contacts, the code could hang.
3. When using the DISPLAY keyword option of ***RIGIDWALL_GEOMETRIC/PLANAR_...**, contacts defined by BOX or SURFATYP/SURFBTYP = 6 included the segments of the rigid wall display parts. The nodes of the display parts could become tracked by other rigid walls.
4. Enabled ***ELEMENT_SHELL_SOURCE_SINK** when used with MPP segment-to-segment (SOFT = 2) contact in the MPP version. Previously, the job would error terminate.
5. Fixed an error in implicit segment-to-segment (SOFT = 2) contact stiffness when DEPTH = 5, 15, 35, or 55 and contact is between edge pairs. The stiffness was uninitialized, causing unpredictable behavior.
6. Fixed a conflict between ***CONTACT_EXCLUDE_INTERACTION** and the thick segment pair check in MPP segment-to-segment (SOFT = 2) contact. The conflict led to excluding the wrong segment pairs.
7. Fixed damping in segment-to-segment (SOFT = 2) contact when using contact stiffness scaling by part. The damping was not adjusted correctly for the scaled stiffness values. This fix may change results a little but should not cause big changes.
8. Consider IDDOFF from ***INCLUDE_TRANSFORM** for REGION on optional Card E of ***CONTACT_...**

7.7 Control Cards

1. Consistent mass treatment of rigid bodies (RBSMS = 1 on ***CONTROL_-RIGID**) did not work with rigid-to-deformable switching.

2. Selective mass scaling (IMSCL on ***CONTROL_TIMESTEP**) did not work properly on constrained tied contacts using backup tied contact (IPBACK = 1 on optional Card E of ***CONTACT_...**).
3. When ATYPE = 3 or 4 on ***CONTROL_PORE_FLUID**, pore fluid is modeled flowing through the solid element mesh. Cracks in the mesh act as impermeable barriers, except that fluid can flow through tied contacts (***CONTACT_TIED...**) if FTIED = 1. In previous releases, this did not work if the tied contact was “groupable.”
4. For ***CONTROL_STAGED_CONSTRUCTION**:
 - Fixed unpredictable errors when combined with certain material models for which one or more of the material model’s history variables were initialized with a nonzero value. The material models these errors occurred with were material types 213, 277, 278, 307, and ***MAT_USER_DEFINED** with NHV (number of history variables) set greater than 99.
 - When writing dynain files during staged construction analysis, beam elements that have ELFORM = 3 with ***MAT_NULL** were always skipped. This behavior is the correct action for autogenerated elements created for displaying spring and seatbelt elements from the d3plot file but not for user-created elements. A distinction is now made between these two categories.
 - A bug affected models containing beam elements with ELFORM = 6 if the parts containing these elements change from dormant to active during the simulation (STGA > 0 on ***DEFINE_STAGED_CONSTRUCTION_PART**) and ***DATABASE_DISBOU** is present. Symptoms occurred after the beams became active and could affect parts of the model not physically connected to the beams. The symptoms were unpredictable and could differ according to the model’s contents and number of CPUs.

7.8 Dual CESE Compressible Fluid Solver

1. Corrected the setup for the use of ***DEFINE_CURVE_FUNCTION** with the dual CESE solvers.
2. For dual CESE d3plot IBM-FSI nodal outputs, if the nodes on a fluid boundary are also very close to a structural part, they are now treated for plotting as if they were effectively inside the structure.

7.9 Elements

1. Fixed issue with ***ELEMENT_DIRECT_MATRIX_INPUT** combined with ***CONTROL_IMPLICIT_MODAL_DYNAMIC** that led to a crash.
2. Fixed internal energy computation for cohesive elements. Shear components were too small by a factor of 2.
3. Fixed damping energy computation (**RYLEN = 2** on ***CONTROL_ENERGY**) for Hughes-Liu beam elements. Necessary strain increments were not rotated to the global system. Global z-components were not included in the energy computation.

7.10 EM (Electromagnetic Solver)

1. Made fixes when using the EM 2D axisymmetric solver (***EM_2DAXI**) with EM contact. Memory issue could cause some cases to fail.
2. Bug fix for **EMSOL = 2** (periodic inductive heating solver) on ***EM_CONTROL**. Previously, models that involved imposed voltage and also had EM contact present would fail during the solve.

7.11 Forming Analysis

1. Fixed bugs in memory allocation that may cause memory errors in metal forming simulations.

7.12 ICFD (Incompressible Fluid Solver)

1. Fixed issue that caused the presence of ***ICFD_DATABASE_FLUXSURF** to affect the generated mesh.
2. Fixed update of the stiffness matrix for turbulent flows.
3. Fixed bug in the computation of the volume region of ***ICFD_DEFINE_POROUS_REGION** and ***ICFD_CONTROL_GAP**.

7.13 Implicit (Mechanical) Solver

1. Fixed MPP implicit issue that caused some nodes not to tie for ***CONTACT_AUTOMATIC_TIEBREAK** when using **NSOLVR = 12** on ***CONTROL_IMPLICIT_SOLVER**.

2. For METHOD = 102 (LOBPCG) on ***CONTROL_IMPLICIT_EIGEN-VALUE**:
 - Enforce maximum number of modes, NEIG, in interval computations (LFLAG/RFLAG). Before, the entire interval of modes was returned.
 - Fixed a bug leading to crashes in some cases when performing interval computations (LFLAG/RFLAG).
 - Fixed bug that caused incorrect results for very small problems.
 - Improved some log messages and fixed some erroneous timers.

7.14 Initial Cards

1. Fixed incorrect element offsets for ***INITIAL_STRAIN_SHELL** and ***INITIAL_STRAIN_TSHELL** when they are in an included file with nonzero element offsets.
2. Fixed issue in dynain where ***INITIAL_{STRESS/STRAIN}_SHELL** did not use “+” to indicate long format.
3. Fixed issue where ***INCLUDE_TRANSFORM** did not work properly for ***INITIAL_STRESS_SECTION**.
4. Fixed the incorrect initialization of the additional thickness stretch degree of freedom of shells (ELFORM = 25, 26, and 27) when initialized with ***INITIAL_VELOCITY**.
5. Fixed some ***INITIAL_STRAIN_SHELL** output in the dynain file (written with ***INTERFACE_SPRINGBACK_LSDYNA**). The parametric coordinate *T* of through thickness-integration points could have been wrong, mainly in the case of using ***PART_COMPOSITE**.

7.15 Isogeometric Analysis (IGA)

1. Properly update control point coordinates prior to writing the dynain file (***INTERFACE_SPRINGBACK_LSDYNA**). The local integration point coordinates were not grabbed correctly, leading to a low-quality mapping.
2. Fixed potential mapping errors with ***INCLUDE_STAMPED_PART** from standard FE parts to IGA shell parts ***IGA_SHELL** in cases where the number of in-plane integration points and through-thickness integration points differ.
3. Properly exclude interpolation elements for the computation of mass and inertia properties written to d3hsp and d3plot. This computation only

affected the output but not the actual analysis.

4. Fixed some bugs in the mapping of the results from the IGA solid elements (***IGA_SOLID**) to the related interpolation elements that may have led to strange post-processing results in the interpolation elements.

7.16 Materials and Equations-of-State

1. Fixed the following bugs that occurred for ***MAT_ADD_DAMAGE_-DIEM** when in a file included with ***INCLUDE_TRANSFORM**:
 - Conversion of rates in table P1 did not work.
 - Values in table Q1 < 0 were converted incorrectly.
 - Output to dyna.inc was not using integers for IDs (P1, P5, Q1<0, Q4)
2. Fixed issue for ***MAT_ADD_DAMAGE_DIEM** by not extrapolating in strain rate tables given by P1.
3. Fixed incorrect table evaluation of LCSDG on ***MAT_ADD_DAMAGE_-GISSMO** when it is a ***DEFINE_TABLE_4D**. The table evaluation was not working correctly for the strain rates, and the temperature was zero in a purely mechanical analysis.
4. Fixed issue for ***MAT_ADD_DAMAGE_GISSMO** for the unlikely event of using shells with Lode parameter dependent FADEXP < 0 (***DEFINE_TABLE_3D**) or LCREGD<0 (***DEFINE_TABLE_3D**). Shell elements do not need a Lode dependence anyway.
5. Fixed error in the load curve input added during the input phase for ***MAT_ADD_SOC_EXPANSION**. For all load curves, an offset OFFFA = 1 was accidentally added during the input phase.
6. Fixed not correctly accounting for temperature in the evaluation of the load curves for ***MAT_ADD_THERMAL_EXPANSION**.
7. Fixed combination of tetrahedron type 13 and material models with an equation of state (***EOS_...**). A wrong new volume value was used in the energy update, leading to unstable results. This issue affected ***MAT_015** (EOS mandatory), ***MAT_224** (when using EOS), and ***MAT_260** (when using EOS).
8. Stopped erroneously converting parameter C on ***MAT_024** when ***MAT_024** with VP = 3 is in a file included with ***INCLUDE_TRANSFORM**.
9. Corrected problem with the number of history variables in user materials

(*MAT_USER_...) with solid elements when options INTOUT/NODOUT of *DATABASE_EXTENT_BINARY are set to STRAIN or ALL. It was already fixed in 2014 for explicit (subroutine urmathn), but now it is also fixed for implicit (urtanh).

10. For *MAT_058 (*MAT_LAMINATED_COMPOSITE_FABRIC) and *MAT_262 (*MAT_LAMINATED_FRACTURE_DAIMLER_CAMANHO), fixed bug in table evaluation for EA, EB, and GAB < 0 for shell elements that could have led to a wrong initial stiffness evaluation.
11. For *MAT_WINFRITH_CONCRETE (*MAT_084), fixed bug that affected the combination of RATE = 8 with DILRAT greater than zero, when cracks were undergoing sliding (shear displacement). Symptoms could include stresses being noisier than expected or crack opening displacements being tracked incorrectly in some elements.
12. Fixed the transformation of DAMP on *MAT_089 when *MAT_089 is in a file included with *INCLUDE_TRANSFORM.
13. Fixed issue with *MAT_098 solids initialized by dynain (*INITIAL_STRESS_SOLID). The initial volume was not properly scattered to all integration points.
14. For beam and hex spot welds, fixed a wrong interpretation of the values MRR, MSS, and MTT on *MAT_100 (*MAT_SPOTWELD), which were cyclically shifted. This error could have affected material definitions, where MRR ≠ MSS ≠ MTT.
15. Fixed issue with time averaging using NF on *MAT_100. We reduced the number of cycles for averaging when the total number of cycles thus far is less than NF. Initial stresses or forces could have been reduced inappropriately without this fix. This fix applies to spot weld solids, solid assemblies, and spot weld beams.
16. Align implicit behavior of *MAT_100 solids with and without SIGY < 0.
17. Added missing initialization of solid spot weld failure table when using solid spot welds in implicit with IACC = 1 on *CONTROL_ACCURACY. Without this fix, options like *MAT_100 with OPT = 10, *MAT_100_DA with *DEFINE_CONNECTION_PROPERTIES, or *MAT_240_FUNCTIONS did not work correctly.
18. Fixed bugs in the plasticity algorithm of *MAT_106 / *MAT_ELASTIC_VISCOPLASTIC_THERMAL that may result in a floating error in element processing.
19. Fixed issue with *MAT_124 (*MAT_PLASTICITY_COMPRESSION_

TENSION) for solid elements when $SRFLAG = 2$. We increased the maximum number of iterations from 20 to 100 for the (slow) Ridders' method. Potential nonconvergence led to error termination.

20. Enable ***MAT_126** to be used with ***MAT_ADD_GENERALIZED_DAMAGE**, meaning add ***MAT_126** to the list of available materials for $IFLG2 = 1$ (material system).
21. For ***MAT_135**, $SIGMA_0 < 0$ (yield curve) did not work for WTM ($FLG = 2$), option NUMFI was not working as expected, and the failure mode on history variable #7 was missing.
22. Fixed bug in ***MAT_COHESIVE_GENERAL** (***MAT_186**) that resulted in negative damage (meaning healing) when the mode-mixity changed under load reversals. This issue was most apparent for brittle and weak materials.
23. For ***MAT_187** in a file included with ***INCLUDE_TRANSFORM**, fixed the conversion of logarithmic strain rates in LCIDT.
24. Fixed issue with ***MAT_SOIL_BRICK** (***MAT_192**) that could cause a crash when running SMP with two or more processors.
25. Limit SLIMXT to 0.99 on ***MAT_213** (***MAT_4A_MICROMECC**). This restriction resolves the fiber stress dropping to 0.
26. Fixed thickness computation in ***MAT_240** with $THICK = 0.0$ for pentahedral cohesive elements ($ELFORM = 21$ and 22).
27. Allow using ***MAT_PML_NULL** (***MAT_246**) in a model with ALE, SALE, or SPH.
28. Fixed a bug in the latent heat calculation for the 3D version of ***MAT_254**. Special handling for axisymmetric solids was missing, which could result in segmentation violations.
29. Fixed stability issue in ***MAT_273** / ***MAT_CDPM** causing a segmentation fault in some cases for MPP single precision.
30. Fixed bug in using large curve IDs (> 16.7 million) in ***MAT_S15** (***MAT_SPRING_MUSCLE**). The bug led to an error reporting the load curve as undefined.

7.17 Output

1. Fixed a bug for ***DATABASE_CURVOUT** for thermal-structural coupling

analysis that caused the database to be wrong.

2. Fixed incorrect mass reporting for cohesive shells type 46 and 47 in d3hsp.
3. Labels Principal/Max Stress Range on ***DEFINE_MATERIAL_HISTORY** did not work with dynamic relaxation.
4. Fixed issue with ***DATABASE_PWP_FLOW** when using automatic time factoring (see ***CONTROL_PORE_FLUID** with TARG, FMIN, FMAX). The fluid flow rate tended to rise or drop suddenly every 100 time steps during the period when the time factor was changing. This bug could also have affected the displacement time histories of nodes in the consolidating material, again with a pattern that repeated every 100 time steps.
5. Correct spcforc calculation for nodes with ***ELEMENT_MASS_MATRIX**.

7.18 Restarts

1. Fixed (full) restart issue for MPP where elements were not deleted when using ***DELETE_ELEMENT_SOLID**.

7.19 Sensors

1. Fixed a bug for ***SENSOR_DEFINE_FORCE** when TYPE = JOINT. The sensed results could depend on the joint definition sequence.
2. Fixed a bug for ***SENSOR_DEFINE_FORCE** when TYPE = PRESCMOT. The sensed force was not correct.
3. Fixed a bug for TYPE = BELTPRET on ***SENSOR_CONTROL** that caused the pretensioner to not turn off correctly for pretensioner types 4 through 8.

7.20 SPH

1. Fixed bugs in the code that removes inactive SPH particles during redecompositions when REMSPH > 0 in ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION**. Some issues would cause the program to crash, while others would cause SPH injections to terminate prematurely.
2. Fixed memory issues that could cause SPH simulations to crash when any

of the following keywords were used: ***DEFINE_SPH_MESH_OBJ**, ***DEFINE_ADAPTIVE_SOLID_TO_SPH**, ***RIGIDWALL**, or ***CONTACT_AUTOMATIC_NODES_TO_SURFACE**.

3. Fixed bug that caused issues when running a simple restart of SPH simulations with $MEMSPH < 0$ in ***CONTROL_SPH**. The bug led to different issues depending on the feature used, including crashes or error terminations.

7.21 Thermal Solver

1. Fixed higher order (27 node hexahedrons) elements in the thermal solver. Without the fix, simulations often resulted in segmentation violation or error termination.

7.22 User-Defined Features

1. Fixed (simple) restarts for sharelib usermat when using ***MODULE_LOAD** or the command line option `module=`. It was not working.
2. Fixed issue for sharelib usermat where ***MODULE_USE** could not be used for thermal materials.
3. Fixed format issue with ***USER_INTERFACE_FRICTION** when `long=s`.

7.23 Miscellaneous

1. Fixed bug in interpreting part sets. This bug was only triggered if the input deck had part sets that contained composite materials. In this situation, if MPP decomposition referenced any part sets, the decomposition might not be what was expected. In some cases (even if the decomposition did not reference any part set), the program crashed during input processing. If no part sets were used in the decomposition, and the program did not crash during startup, the rest of the calculation was fine. There were no adverse effects on the actual simulation.
2. When $OFFA > 0$ on ***DEFINE_CURVE**, the old discretization code assumed that the first abscissa value on the input curve was 0.0, and inserted two extra curve values at (0, 0) and ($OFFA \times 0.999$, 0). However, this assumption caused a reversal in the resulting curve if the first abscissa value on the curve was < 0 . If the first abscissa value on the curve was > 0 , $OFFA \times 0.999$ might be quite a bit less than the shifted curve location. The

revised behavior is not to insert these two points at all if the original starting abscissa is $< -OFFA$, and otherwise to insert the two points at $(0, 0)$ and $(\text{starting abscissa} + 0.999 \times OFFA, 0)$.

3. For ***COSIM_FMI_INTERFACE**:
 - Fixed a bug that might result in a segmentation fault when ***SET_SEGMENT** exists in the input file.
 - Fixed a bug causing incorrect solutions that occurred when an imported curve was used to prescribe displacement ($VAD = 2$ in ***BOUNDARY_PRESCRIBED_MOTION**).
4. Fixed a bug that occurred when using ***INCLUDE_TRANSFORM** together with ***NODE_TRANSFORM** for a node set defined by BOX or VOL. The model was not transformed correctly.
5. Fixed issue with ***PART_AVERAGED**. A segmentation fault during initialization was possible if nodes were not numbered in a certain way. The numbering should not matter, and now it does not.
6. Made fixes for keyword format signs +, -, and %. Mixing those different format signs did not work in every case. See the section “GENERAL CARD FORMAT” for a description.
7. Fixed bug with ***DEFINE_TABLE_2D** in a ***INCLUDE** file. One symptom of the bug was incorrect output in dyna.inc.
8. Avoid error termination when ***DEFINE_CONNECTION_PROPERTIES** is used with $(D)SIGY < 0$.
9. Fixed issue with ***DEFINE_MULTI_SHEET_CONNECTORS**. Different parts/materials for joining elements JNTxx did not work.