R13.2.0 Release Notes

July 3, 2025

1 Release notes revision

This file constitutes revision 0 of the release notes for Ansys LS-DYNA® version R13.2.0.

2 License

The string "REVISION 13" must appear in the license file to run version R13.2.0. Please contact your distributor or your Ansys sales representative if you need to have your license updated.

3 Documentation

The R13.0 User's Manuals provide documentation for R13.2.0. To download these manuals, go to https://lsdyna.ansys.com/manuals/. For features mentioned in these notes missing from the R13.0 User's Manuals, please refer to the DRAFT User's Manuals at https://lsdyna.ansys.com/manuals/. Please note that not all features in the DRAFT User's Manuals are available in version R13.2.0.

4 Notes

The remainder of this file briefly describes new features, enhancements, and bug fixes in version R13.2.0. 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 Airbags

1. Support CPM in the dynamic relaxation phase. CPM particles can move randomly, but the interaction forces between particles and the structure are disabled.

5.2 Elements

1. Implemented fast shell type 30 based on fully integrated shell type 16 but only with two in-plane integration points (1 and 3).

5.3 MPP

1. Added keyword ***CONTROL_MPP_IO_NOTIEDIO** to invoke using memory instead IO for storing the tied_nodes temp files to avoid problems on distributed file systems. The equivalent pfile command is `general { notiedio }`.

5.4 User-Defined Features

1. Added user subroutine for time step definition in explicit analysis. Field DTUSR on ***CONTROL_TIMESTEP** invokes using subroutine `utimestep` in dyn21.F. This subroutine can be used to synchronize time steps between coupled codes.

6 Improvements and Enhancements

6.1 Airbags

1. Improved the behavior for determining the pressure difference for PPOP on ***AIRBAG_PARTICLE**. Previously, before reaching the pop pressure, the part pressure of the vent was zero. The algorithm determining the pressure difference for PPOP then depended on how the airbag was modeled. If an additional part was included near the vent, the part pressure of that part was used. If not, the algorithm looked to see if a chamber was defined to determine the pressure at the vent. If neither a chamber nor an additional part was defined, a noisy model was used to determine the pressure at the vent. With this improvement, the pressure of the vent part is calculated in the same way as a regular part before reaching the pop pressure without requiring an additional part. This algorithm has increased accuracy compared to not including an additional part or chamber. It also no longer requires an additional part or a chamber. Chamber definitions are now ignored, but additional parts are still used.

2. Added keyword options JET_LENGTH and JET_TIME to ***AIRBAG_-PARTICLE** to improve deployment kinematics.

3. Added field VNDAMP, a damping coefficient, to ***AIRBAG_PARTICLE** to suppress the motion in the fabric's normal direction.

4. LCPC23 on ***AIRBAG_PARTICLE** can be a ***DEFINE_CURVE_-FUNCTION** or ***DEFINE_FUNCTION**.

5. Added VD*i* = -3 and -4 on ***AIRBAG_PARTICLE** to enable an offset from the shell's center of the inflator position for shell-based inflators. This option can give consistent results between inflators defined by shells and nodes.

6.2 Boundary Cards

1. Made improvements to MPP handling of non-reflecting boundary (***BOUNDARY_NON_REFLECTING**) segments during dynamic relaxation.

6.3 Constraint Cards

1. Add two optional fields to ***CONSTRAINED_INTERPOLATION_-SPOTWELD** (SPR3): Scale factors SCARN and SCARS for tensile and shear strength. They scale RN and RS and the displacements in LCF, LCUPF, and LCUPR. The purpose of these scale factors is to obtain a scaled

force-displacement curve in both directions. These scale factors can also refere to curves by setting them to values less than 0. These curves allow the scale factors to be functions of the state of the neighboring SPR3 (such as in a multi-sheet connection). Thus, the scale factors can reduce the strength as the SPR3 approaches failure.

2. Added option SPRSMD to ***CONTROL_CONSTRAINED**. This flag sets the "shear moment distribution" for SPR3 with:

   - SPRSMD = 0: Distributed as force pairs (new default), or
   - SPRSMD = 1: Distributed as nodal moments (old behavior).

   This flag acts as a bug fix with a fallback option.

3. Added flag COHEQC to ***CONTROL_SOLID** to control the COHesive Element Quality Check. We added this check in 2020, which caused an error termination if a bad quality (inverted element) is detected in cycle 0 or 1. This flag changes the behavior of the check:

   - COHEQC = 0: Error termination as before (default)
   - COHEQC = 1: Warning and continue
   - COHEQC = 2: Warning with element erosion, continue

4. Added new interpolation method for ***CONSTRAINED_SPR2** and ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3). INTP = 3 applies quadratic weighting, meaning the scale factor increases with the squared distance from the center to achieve higher stresses at the connector periphery.

5. Enhanced the performance of ***CONSTRAINED_INTERPOLATION_-SPOTWELD** (SPR3) and ***CONSTRAINED_SPR2** by replacing brute force search by a bucket sort approach. This change should reduce initial CPU time for large models with a huge number of such connector definitions.

6. Added support for using ***CONSTRAINED_SPR2** with ***DEFINE_-HAZ_PROPERTIES**. Up to now, ***DEFINE_HAZ_PROPERTIES** was only working with ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3).

7. For ***CONSTRAINED_SPR2** and ***CONSTRAINED_INTERPOLA-TION_SPOTWELD** (SPR3), enhanced the computation of normal directions for nodes in the connector domain using the least square method to determine the regression plane with a normal vector. This computation was already done in the global system with three different cases, leading to discontinuities when the case switched. Now, we do this computation in the local system of the SPR connector, which prevents discontinuities.

6.4 Contact

1. Improved segment-to-segment (SOFT = 2) contact behavior. This update primarily affects hybrid consistency and eroding contact when the thick segment check removes shell pairs from contact.

2. Added ability to process the contacts in MPP with an ordering based on the contact IDs instead of the input ordering. In this way, if all contacts in the model have user IDs assigned, the contact processing should occur in an order independent of the actual input order of the contacts. This means you can rearrange your include files and not get any differences in the contact processing. Note that for R13.2, this behavior is optional and disabled by default, but for R15 and later major releases, this behavior is always enabled. To enable this feature in version R13.2, either include ***CONTROL_MPP_DECOMPOSITION_CONTACT_SORT** in the input deck or add the line `decomp { contact_sort}` to the pfile.

3. Added field IGTOL to MPP 2 on ***CONTACT_…**. This field applies to ***CONTACT_AUTOMATIC_SINGLE_SURFACE** and ***CONTACT_AUTOMATIC_GENERAL** in MPP. IGTOL is a scale factor used to help determine an "ignore" tolerance based on the segment and node thicknesses.

4. Changed the behavior of the bit flag GRP on ***CONTROL_MPP_CONTACT_GROUPABLE** for segment-to-segment (SOFT = 2) contacts. Segment-to-segment contacts are excluded from the normal meanings of bit = 1 (on) and bit = 4 (off). Setting the 16 bit here does turn on groupable for segment-to-segment contacts, allowing for further testing.

5. Made a minor change to MPP groupable ***CONTACT_…_INTERFERENCE** so that it catches deeper initial penetrations, which makes it match the non-groupable contact behavior better.

6. Changed the initialization of MPP groupable constraint-based tied contacts to match the behavior of the non-groupable variety. Previously, if a reference segment contained rigid nodes, the segment was skipped during the global search. As a result, some tracked nodes were tied to segments that were further away. With this update, the nearest segment is always used, and if it contains rigid nodes, the tracked node is not constrained at all. This behavior matches that of non-groupable contacts. It also works much better than previously when using the IPBACK flag because it allows a penalty contact to tie the tracked node to the nearest segment rather than having the constraint contact tie it to a segment further away (and possibly move the tracked node in the process).

7. Improved MPP full deck restart handling when the initial and current run

have a different number of contact interfaces.

8. Improved segment-to-segment (SOFT = 2) contact accuracy noticeably during rigid body translation.

9. Improved the accuracy of the segment-to-segment (SOFT = 2) contact intersection check to prevent false positives.

10. Added large default values for the normal and shear failure of SMP ***CONTACT_AUTOMATIC_..._TIEBREAK** type contact. This change enables these contact types to be used even if both values are input as 0.0. Before this change, they were behaving as regular sliding-type contacts.

11. 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.

12. Fine-tuned OpenMP directives for ***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE**, ***CONTACT_AUTOMATIC_SINGLE_SURFACE**, and ***CONTACT_ERODING_(OPTION)** to speed up hybrid performance.

13. Added option SSFTYP = 2 for ***CONTACT_...** with SOFT = 2 to invoke using the smaller of the SSF values from ***PART_CONTACT**. Already existing SSFTYP = 1 takes the larger value.

6.5 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. Improved the keyword processing speed for ***CONTROL_SPOTWELD_BEAM** when only a subset of the beam spot welds is converted to hexahedral elements.

3. Added the possibility to consider the effect of dynamic viscosity on the critical time step (DTDYNV on ***CONTROL_TIMESTEP**).

4. Implemented a new algorithm for subcycling (***CONTROL_SUBCYCLE_...**) that should provide a more robust and faster execution.

5. Added option CRVP to ***CONTROL_SOLUTION** to possibly improve curve evaluation performance in special cases (see the manual description) and perhaps additionally reduce the memory requirement of rediscretized curves.

6. Reduced the memory requirement of detailed energies for materials (MATEN = 2 on ***CONTROL_ENERGY**) by only requesting it after MPP decomposition.

6.6 Dual CESE Compressible Fluid Solver

1. Improved the robustness of the moving mesh FSI solver.

6.7 Elements

1. Added option RINRT to ***CONTROL_SOLID** to compute rotational inertia for the nodes of solid elements. This is to ensure consistent results if constraints are applied which assume rotational degrees of freedom, as with contact type ***CONTACT_TIED_SHELL_EDGE_TO_SURFACE**. Without this new option, a model-averaged portion of rotational inertia is used, which is sufficient in most situations but might lead to inconsistencies between different model assemblies in the case of rotational motion.

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. Relaxed check for material orientation definition to avoid writing wrong and unnecessary warning messages (STR+1438) for shells and thick shells when using ***PART_COMPOSITE(_TSHELL)**, ***ELEMENT_T/SHELL_COMPOSITE**, or ***MAT_USER_DEFINED_MATERIAL_MODEL** with anisotropy.

6.8 Forming Analysis

1. Added optional check for the correct material type of the target to ***INCLUDE_STAMPED_PART** with field MTYPE. MTYPE provides the expected material type of the target part. If MTYPE does not match the material type of the target part, an error is thrown. It is useful when mapping history variables to avoid an unintended change of material type.

2. Parameters TENSOR and THKSCL of ***INCLUDE_STAMPED** can be defined for keyword option MATRIX.

6.9 Implicit (Mechanical) Solver

1. Changed MPP implicit's treatment of ***INTERFACE_LINKING** to match the treatment in SMP better.

2. Improved the handling of interface linking (***INTERFACE_LINKING_-FILE** for 2D problems by the implicit solver by ignoring the linking between constraining and linked nodes on certain degrees of freedom if the two nodes have the same global single point constraints.

3. Added an additional line to d3hsp about the new default option for the implicit linear equation solver, which is 7.

4. Enhanced error handling of implicit rotational dynamics (***CONTROL_-IMPLICIT_ROTATIONAL_DYNAMICS**) and sectoral symmetry (***CONTROL_IMPLICIT_SECTORAL_SYMMETRY**) during keyword input.

5. Added logic to left justify the matrix names from keyword input and the dmig files (NASTRAN) for ***ELEMENT_DIRECT_MATRIX_INPUT** when reading them. This improvement makes input easier for the user as they do not have to have the same number of blanks before the matrix names.

6. Made a minor adjustment in the output of implicit memory requirements for the case of intermittent eigenvalue analysis.

6.10 Initial Cards

1. Allow the initialization of history variables based on nodes (***INITIAL_-HISTORY_NODE(_SET)**) for the case that the same node is part of various definitions of node set. For instance, if a node is part of two node sets where history variable #5 is initialized for the first node set and history variable #7 is initialized for the second, both history variables are now initialized for the node.

6.11 Isogeometric Analysis (IGA)

1. Enabled constraint-type tied contacts (***CONTACT_TIED_SHELL_-**

EDGE_TO_SURFACE and ***CONTACT_TIED_SHELL_EDGE_-TO_SURFACE_CONSTRAINED_OFFSET**) for IGA shells (***IGA_-SHELL**). For this to work correctly, the reference segments should be from the ***IGA_SHELL** parts, and the tracked nodes should be standard FE-nodes (such as from cohesive elements and solid spot weld hexahedral elements). This feature is only available in MPP as groupable contact (GRPABLE = 1).

6.12 Materials and Equations-of-State

1. Issue warning if ***MAT_ADD_COHESIVE** is used with non-cohesive elements.

2. Added option for the shear criterion of ***MAT_ADD_DAMAGE_DIEM** (DITYP = 1). For shells, there are now two possibilities for the computation of maximum shear stress: three-dimensional (P3 = 0) and two-dimensional (P3 = 1). The first one matches the behavior before November 2018 and the original paper.

3. Change the behavior of ***MAT_ADD_DAMAGE_DIEM** with parameter Q4 to make the damage evolution more comparable to the "P5 approach." The regularization factor specified with a curve or table using Q4 scales the damage evolution parameter specified with Q1. The factor depends on element size but can also depend on either the abscissa value of the P1 criterion used or the plastic strain rate.

4. Added VOLFRAC to ***MAT_ADD_DAMAGE_DIEM/GISSMO**. VOLFRAC in these keywords has the same functionality as VOLFRAC in ***MAT_ADD_EROSION**. It is the volume fraction that needs to fail before deleting a higher-order or IGA element.

5. Account for additional costs of ***MAT_ADD_DAMAGE_DIEM/GISS-MO** in MPP decomposition. The cost calculation already accounted for shells and the default cost estimate but now also accounts for solids and the newcost decomposition option in the pfile.

6. Added option LCSOFT to ***MAT_ADD_DAMAGE_GISSMO**. This field can refer to a load curve or table with ID |LCSOFT|, giving the soft reduction factor for failure strain in crashfront elements. It can be defined as a function of triaxiality (curve) or triaxiality and element size (table).

7. Added option to ***MAT_ADD_DAMAGE_GISSMO** for triaxiality-dependent regularization. Two new triaxiality values, RGTR1 and RGTR2, between 0 (shear) and 2/3 (biaxial) together with SHRF = 1 and BIAXF = 1 describe a trapezoidal tub-shaped regularization. This behavior would

already be possible with LCREGD < 0, but the new approach is faster.

8. Made a minor efficiency improvement for ***MAT_ADD_DAMAGE_GISSMO** when LCREGD is a table by only computing the regulation factor for currently plastic integration points.

9. Enabled solid element types 0 and 9 to be used with ***MAT_ADD_EROSION**. These are the special corotational formulations for ***MAT_126**.

10. Enabled part-wise definition of the nonlocal energy criterion of ***MAT_ADD_EROSION** and ***MAT_280**. This enhancement relates to the input parameters ENGCRT and RADCRT that exist for both materials. Up to now, only one global energy value could be defined (see Remark 1i in ***MAT_ADD_EROSION** for details). Now each material card definition is taken into account separately.

11. For ***MAT_ADD_GENERALIZED_DAMAGE**:

    - Added ***MAT_058_SOLID** to the list of available materials for IFLG2 = 1 (material system).
    - Added IFLG1 = 3 to make specific components of the total strain rate tensor as the damage driver.
    - Added IFLG4 = 1 to restrict damage drivers only to grow.

12. Modified longitudinal tensile failure (XT) of ***MAT_ENHANCED_COMPOSITE_DAMAGE** (***MAT_054**) in implicit solutions when DFAILT = 0. This option was developed to fail slowly over many cycles, which works OK for explicit solutions but prevents failure in implicit solutions. Failure is now immediate in implicit solutions.

13. Allow large IDs (more than eight digits) for load curves, such as GAB and LCDFAIL, in ***MAT_058** (***MAT_LAMINATED_COMPOSITE_FABRIC**.

14. Added option MODEL = 2 to ***MAT_063**. This choice can be seen as a tension-compression asymmetric version of MODEL = 1 with one additional new parameter.

15. Improved the viscoplastic algorithm in ***MAT_063** for MODEL ≥ 1. This change affects RFILTF = 0. Brent's method is used to avoid non-convergence.

16. Added new option to ***MAT_063** with MODEL = 1 or 2. LCID can now refer to a ***DEFINE_TABLE_3D** to make the yield stress a function of strain (CURVE), strain rate (TABLE), and value of history variable #8 (TABLE_3D). History variable #8 can be set with ***INITIAL_STRESS_SOLID** or ***INITIAL_HISTORY_NODE**. This feature can be used to

model foam with spatially varying porosity/density.

17. Extended the list of function arguments for ***MAT_169** and ***MAT_240_-FUNCTIONS**. Material properties can be defined with ***DEFINE_-FUNCTIONs** for both materials. We added the Young's moduli of the connection partners (flanges) to the list of arguments.

18. Added viscoplastic option invoked by RATEOP = 1 to ***MAT_SAMP_-LIGHT** (***MAT_187L**).

19. For ***MAT_215** (***MAT_4A_MICROMEC**, added check for proper fiber orientation definition and issue an error message.

20. Added ability to define the exponent for the mixed-mode failure to ***MAT_240** with XMU. This is the same approach as in ***MAT_138**.

21. Added six new history variables to ***MAT_240** invoked by RFILTF ≠ 0. History variables 18 to 23 give the normal stress, tangential stress, normal force, tangential force, element area, and mode mixity.

22. Support IRATE from ***CONTROL_IMPLICIT_DYNAMICS** in ***MAT_-240**.

23. Added new options for POSTV and IHIS in ***MAT_249** and ***MAT_249_-CRASH**:

    - Output of fiber directions in the material coordinate system
    - Initialization of fiber directions in the material coordinate system

24. Added ability to initialize history variables 11, 12, and 13 (hardening variables R_i) of ***MAT_258** based on the initial plastic strain coming from ***INITIAL_STRESS_SHELL**.

25. Added alternative damage formulation to ***MAT_280** invoked by setting FRACEN. The existing model drops the stresses in a few cycles, while the alternative approach is based on an actual damage model with linear softening.

26. Added user-specified or stochastic scaling of tensile strength to ***MAT_-280**. A scale factor for FT on history variable #13 can be defined with ***INITIAL_STRESS_SHELL** or with new keyword option STOCHASTIC. Note that both this scale factor and FTSCL scale FT, but the application of this scale factor does not depend on RATENL.

27. Add user-specified scaling of fracture energy to ***MAT_280**. A scale factor for FRACEN on history variable #14 can be defined with ***INITIAL_-STRESS_SHELL**.

28. Added grouping option, GRPFT, to ***MAT_280** to scale down the tensile strength of several parts with FTSCL at the same time.

6.13 MPP

1. Added MPP support for ***TERMINATION_CONTACT**.

6.14 Output

1. For shells, added writing the parameter NLOC from ***SECTION_SHELL** to hisnames.xml.

2. Added field NSKIP to ***DATABASE_BINARY_RUNRSF/D3DUMP** for MPP/HYBRID to reduce the frequency of producing full deck restart files. This option saves CPU time since processor 0 must collect all the information from the processors.

3. Added ***DATABASE_RCFORC_DR** to output resultant interface forces during dynamic relaxation.

4. Support using ***DEFINE_MATERIAL_HISTORIES** for output to d3part.

6.15 Thermal Solver

1. Increased number of digits when writing ***LOAD_HEAT_GENERATION** IDs to the structured LSDA file. Now, it is possible to have more than 10^6 cards.

2. Allow for exchange of history variables between mechanical and thermal user material models for axisymmetric solids (***SECTION_SHELL** ELFORMs 14 and 15).

3. Added parameter TMPOFF to ***LOAD_THERMAL_BINOUT** for each thermal loading condition. It defines a temperature offset between thermal results in the binout files and the thermal loading for the mechanical solver. Thus, it enables running the thermal simulation in Kelvin and the mechanical simulation in Celsius or vice versa.

6.16 User-Defined Features

1. Added PARAM1 = UCTRL and PARAM1 = UTIME for TYPE = UTIL in ***MODULE_USE**.

2. Added keyword options SP and DP to ***MODULE_LOAD** to allow loading different shared objects depending on the precision of the binary of the executable.

3. Added restart capabilities for dynamic linking with ***MODULE**.

4. Added output of the sharelib module file name. It is now printed to d3hsp, messag, and screen.

5. Made EOS type information accessible through userinterface.F90.

6.17 Miscellaneous

1. Use the values from the first occurrence if multiple instances of ***DEFINE_FRICTION** have the same ID in an input deck. Additionally, if a part appears in both a part and part set, then the first occurrence is used.

2. Added command line option `shell=16to30/shell=30to16` for switching between shell formulations 16 and 30 without changing the input deck. LS-DYNA adjusts the cost of different formulations during decomposition for better parallel load balance.

3. Enabled sensor switch `endtime` with the d3kil file.

4. Added command line option `stdout=` to control the messages to stdout. See the manual for details.

5. Added adiabatic index GAMMA to ***DEFINE_PRESSURE_TUBE**.

6. Added cavity pressure correction factor CFAC to ***DEFINE_PRESSURE_TUBE**.

7. Addition to option TRANSL2ND of ***DEFINE_TRANSFORMATION**: if distance a3 is zero, then the distance between a1 and a2 is directly used for the translation distance.

8. Extended the list of function arguments for ***DEFINE_CONNECTION_PROPERTIES** with PRUL ≥ 2 or ***DEFINE_MULTI_SHEET_CONNECTORS** to include the Young's moduli of the connection partners (flanges). This extended list of arguments applies to the ***DEFINE_FUNCTION**s that determine the material and failure properties.

7 Bug Fixes

7.1 Acoustics, NVH, and Frequency Domain

1. Bug fix to properly read the ***FREQUENCY_DOMAIN_SSD** cards if the long format input option `long=s` is used.

7.2 Airbags

1. Fixed bug for ***INITIAL_AIRBAG_PARTICLE_POSITION**.
2. Report airbag inflator energy to external work under glstat. This external work was zero in versions R12 and later.

7.3 ALE

1. Fixed bug when using ***ALE_STRUCTURED_MESH_VOLUME_-FILLING** in 2D cases when an initial velocity is specified. This issue could cause the run to crash.
2. Fixed issue when the filling geometry (***ALE_STRUCTURED_MESH_-VOLUME_FILLING** contains two opposing parallel segments that lie very close to each other, whcih could cause the run to crash.
3. Added a check to prevent a crash while using ***ALE_STRUCTURED_-MESH_VOLUME_FILLING** when no Lagrange surface segment is found inside S-ALE mesh.
4. Fixed issue with constraining the nodal motion of S-ALE nodes in a solid element structure using ***ALE_STRUCTURED_FSI**. The constraints on the S-ALE nodes were not being released when the S-ALE nodes moved outside of the structure.
5. Fixed missing coupling segments for coupling with S-ALE (***ALE_-STRUCTURED_FSI**) when generating surface segments from wedge or tetrahedral structure solid elements.
6. Fixed missing coupling with a newly exposed structure surface segment in the coupling between S-ALE and an eroding structure using ***ALE_-STRUCTURED_FSI** that occurred if the segment lies right at the MPP domain decomposition boundary.

7. Fixed incorrect results when coupling 2D S-ALE with an eroding structure using ***ALE_STRUCTURED_FSI**.

8. Fixed lack of coupling between S-ALE and the structure when using ***ALE_STRUCTURED_FSI** if the Lagrange part was defined by ***PART_STACKED_ELEMENTS**.

9. For S-ALE, the mass of the switch material (SWID on ***DATABASE_FSI**) accumulated twice in the dbfsi file.

10. Fixed incorrect results in the fsifor for S-ALE fluid-structure interaction (***ALE_STRUCTURED_FSI**).

11. Fixed ***SENSOR_CONTROL** with TYPE = SPC not properly working for nodes in S-ALE.

12. Fixed issue in MPP when coupling S-ALE (***ALE_STRUCTURED_FSI**) to a Lagrange structure that can erode and has a hexahedral element sharing a segment with two wedges. The issue led the run to crash.

13. Fixed crash that occurred when an input deck includes both S-ALE and solid elements with ***MAT_SPOTWELD** as the material.

14. Fixed bug that caused a great slow down in the S-ALE advection.

15. Fixed issue in S-ALE that caused an incorrect hourglass energy.

16. Fixed bugs in S-ALE 3D to 3D mapping that led to incorrect results.

17. Fixed part ID collision that occurred when using S-ALE with Lagrange parts containing ***ELEMENT_SHELL_OFFSET_COMPOSITE**.

18. For ***ALE_STRUCTURED_MESH_TRIMMING**:

    - Fixed issue that caused the run to crash when generating the solid set for the trimmed mesh.
    - Fixed issue that caused the run to crash when generating the segment set for the trimmed mesh.

19. Added a check to avoid the Reynolds number being incorrectly calculated as infinity when using ***ALE_COUPLING_NODAL_DRAG**.

7.4 CESE Compressible Fluid Solver

1. Fixed FSI coupling involving a structural part that uses solid element type 13 (tetrahedral) elements. Using these elements previously led to a crash.

7.5 Constraint Cards

1. Fixed a memory error that may cause the simulation to crash when using ***CONSTRAINED_COORDINATE**.

2. Do not echo warning INI+485 when node 4 is not defined for ***CONSTRAINED_JOINT_TRANSLATIONAL_MOTOR** because it is not required.

3. Fixed bug for the combination of ***CONSTRAINED_NODE_SET** and ***DAMPING_PART_MASS** on MPP. The symptom of the bug was that the constraint was not obeyed for all of the nodes in the set, and warning MPP+156 was issued.

4. Fixed bug for ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3). If the connection opened up more than 90 degrees (very high bending), the computation of the lower sheet's normal direction could be wrong, leading to erroneous results.

5. Fixed bug for SPOTHIN on ***CONTROL_CONTACT** when applied to ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3). MPP synchronization was missing for scaled contact thickness.

6. Fixed issue in the internal energy computation for ***CONSTRAINED_SPR2** to avoid nonsensical results when damage reaches 1.

7. Fixed the format of the structured input for ***CONSTRAINED_SHELL_TO_SOLID** with `long = s`.

7.6 Contact

1. Fixed an issue with frictional energy output for groupable surface-to-surface contacts. The issue could result in slightly incorrect energies being reported or in a segmentation fault under the following conditions:

   - ***CONTACT_AUTOMATIC_SURFACE_TO_SURFACE** interface with GRPABLE = 1
   - FRCENG = 1 in ***CONTROL_CONTACT**
   - PKP_SEN = 1 in ***DATABASE_EXTENT_BINARY**

2. Fixed inconsistency issues in the hybrid solver caused by ***CONTACT_TIED_SHELL_EDGE_TO_SURFACE_OFFSET** and ***CONTACT_AUTOMATIC_SINGLE_SURFACE**.

3. Fixed issue in determining the beam element contact thickness due to not

honoring the thickness scaling factor in some cases. Specifically, ***CONTACT_AUTOMATIC_BEAMS_TO_SURFACE** was not honoring the SFSAT parameter.

4. Fixed error in the reporting of untied nodes in some groupable tied contacts. If a node was close to tying in a groupable tied contact but not close enough, the warning was not being properly issued. For nodes very far from tying, the warning was still correct. This bug affects all groupable tied contacts.

5. Fixed possible deadlock that could have happened in MPP if there was a tied contact with birth time that became active during dynamic relaxation. After dynamic relaxation, when the birth time was again reached, deadlock could occur in some cases.

6. Fixed MPP support of orthotropic friction for contact type ***CONTACT_ONE_WAY_SURFACE_TO_SURFACE**. The bug could cause the generation of an error similar to this:

```
*** Error 70341 (OTH+341) (processor # 0)
    Attempt to resolve pointer for unallocated memory
    block index 14
    Please report this message to LST
```

7. Fixed reporting issue during initialization of MPP groupable tied contact. In some cases, nodes that were not being tied were not reported if rigid body nodes were involved.

8. Fixed warning message for nodes not being tied in MPP for tied contact. In some cases where this message should have been written to the messag file:

```
*** Warning 50129 (MPP+129)
    Tracked node is not constrained
    since it is not found on a segment.
```

   a message like this was being issued with garbage values for the connectivity:

```
*** Warning 50146 (MPP+146)
    segment contains rigid body nodes
```

9. Fixed communication error in MPP ***CONTACT_AUTOMATIC_SINGLE_SURFACE** that occurred when IFTORQ is enabled. This combination could lead to segmentation faults due to message size mismatch.

10. Removed “only output once” restriction on “node not tied” warning

messages for MPP groupable contact. Warning messages were not being printed for all of the nodes not being tied; only a subset had messages.

11. Fixed bug in penalty-based tied contacts that caused the optional contact stiffnes to not be taken into account when the SURFB segments originated from solid facets.

12. Fixed segment-to-segment (SOFT = 2) contact when used with PSTIFF > 0 and selective mass scaling. Segment masses were being scaled incorrectly causing excessive stiffness.

13. Fixed a problem with the birth time option of MPP segment-to-segment (SOFT = 2) contact when the contact interface uses PSTIFF = 1. An error occurred when not all cores participated in the contact.

14. Fixed a segmentation fault error when ***CONTACT_2D_AUTOMATIC** is coupled with the thermal solution, and the structural solution is explicit.

15. Fixed the thermal option of ***CONTACT_2D_AUTOMATIC_...** when it is used with ***SENSOR_SWITCH** to turn it off. Before the fix, thermal energy would continue to transfer across the interface even when it was switched off.

16. Fix an issue with excessive forces when automatic tiebreak (***CONTACT_-AUTOMATIC_..._TIEBREAK**) fails. This affects OPTION = 6, 7, 8, 9, 10, 11, 13, 14, and 101-105 (user-defined).

17. Fixed application of viscous damping (VDC) to automatic tiebreak with OPTION = 9, 11, 13, or 14. Before failure, these options use CN × area as the contact stiffness, not the regular penalty value. Now, they also use this for the additional damping stress.

18. Consider IDDOFF from ***INCLUDE_TRANSFORM** for REGION on ***CONTACT_...** optional Card E.

7.7 Control Cards

1. Fixed bug that leads to an error termination when initializing a model with IDRFLG = 2 in ***CONTROL_DYNAMIC_RELAXATION** and using shell *h*-adaptivity in the transient phase. The error occurred during the first adaptive remeshing.

2. Fixed using ***DEFORMABLE_TO_RIGID** with consistent mass scaling (RBSMS = 1 on ***CONTROL_RIGID**. The consistent mass treatment of rigid bodies did not work with rigid-to-deformable switching.

3. Pore pressure (***CONTROL_PORE_FLUID**) analysis did not work with tetrahedral elements with element formulation 10.

4. For staged construction (***CONTROL_STAGED_CONSTRUCTION**):

   - Fixed incorrect calculation of the internal energy for dormant solid elements. Dormant solid elements are solid elements (***ELEMENT_SOLID**) that belong to parts that are not yet active according to ***DEFINE_STAGED_CONSTRUCTION_PART**. Because the internal energy was wrong, the energy balance ("energy ratio") showed up as far different from unity. While this issue could appear alarming to users, it affected only the output values of energy and did not indicate any problem with the calculation itself (meaning stresses, displacements, etc.).
   - When writing the dynain file from staged construction analysis, ELFORM = 3 beam elements with material ***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.

5. Selective mass scaling (IMSCL ≠ 0 on ***CONTROL_TIMESTEP**) and consistent mass scaling (RBSMS = 1 on ***CONTROL_RIGID**) did not work with deformable to rigid switching (***DEFORMABLE_TO_RIGID**).

6. Fixed problem with DRCPSID on ***CONTROL_SHELL**. Even if the part set referenced some parts and did not reference others, the parts could still have been grouped together improperly. This improper grouping caused the drill constraint to be applied where it should not or not applied where it should. This incorrect grouping only happened if all other properties of the two parts were the same.

7. Added some missing part set offsets on control cards: apply IDSOFF from ***INCLUDE_TRANSFORM** where necessary.

8. Correctly read and use GMDT specified in ***CONTROL_OUTPUT** for ***INTERFACE_SSI_AUX**. Previously, the specified value of GMDT was being ignored.

7.8 Elements

1. Fixed issue with ***ELEMENT_SHELL_SOURCE_SINK** for MPP, which could lead to a deadlock if not all processors were involved invthe source-sink algorithm.

2. Fixed bug to properly read in solid elements with 64 nodes using ***ELEMENT_SOLID_H64** and solid elements with 40 nodes using ***ELEMENT_SOLID_P40**.

3. Fixed an energy growth problem that could occur with MPP segment-to-segment (SOFT = 2) contact using ***CONTACT_ERODING_SINGLE_SURFACE** when the surface includes thick shell elements with short edge lengths.

4. Fixed an error that could occur in MPP segment-to-segment (SOFT = 2) contact with keyword ***CONTACT_ERODING_SURFACE_TO_SURFACE** or ***CONTACT_ERODING_SINGLE_SURFACE** when either the sliding option is invoked by setting SBOPT = 4 or 5 in optional Card A or edge-to-edge checking is invoked by setting DEPTH = 5, 15, 25, or 35 on optional Card A. These options were not working well due to bad data.

5. Fixed the MCID option of ***ELEMENT_SHELL** which was calculating the wrong direction.

6. Corrected a bug that caused spurious damping to be applied when superlement (***ELEMENT_DIRECT_MATRIX_INPUT**) attachment nodes were on a rigid body. The bug affected implicit dynamic transient analysis (IMASS = 1 on ***CONTROL_IMPLICIT_DYNAMICS**).

7. The presence of ***ELEMENT_BEAM_PULLEY** in the model caused beams made of certain material types to disappear from the model on the first time step, irrespective of whether they were attached to pulley elements. Affected material types include 191, 209, and potentially others that refer to the initial length of the element.

8. Fixed bug for ***ELEMENT_BEAM_PULLEY** and ***ELEMENT_BEAM_SOURCE** related to the forces in the attached elements. This issue could result in oscillating forces. Also, if the cable elements started with a nonzero force (such as in multi-stage problems), the first time state in the new run showed zero forces in the elements attached to the pulley.

9. Corrected indexing error when using two or more superelements (***ELEMENT_DIRECT_MATRIX_INPUT**) with ***BOUNDARY_SPC** applied to the nodes of at least one superelement that was not the first one. This issue only impacts explicit time integration. The error would have shown up during matrix assembly of the superelement system with the error message including the term LCPACK.

10. Fixed a long-standing bug for mass scaling of tetrahedral elements. The algorithm computed the added mass based on the current element mass instead of the original element mass using the initial volume. A comparison with hexahedral elements revealed that the mass scaling was incorrect,

especially for large volume changes (like when foam gets compressed). In such cases, the added mass was too small during compression and too large during unloading.

11. Fixed issue with mass scaling of cohesive solids (***SECTION_SOLID** with ELFORM = 19 to 22). Inverted elements (negative volume) could lead to a huge negative added mass. Replacing current volume by initial volume fixes this issue.

12. Fixed issue in the shell thickness update specified with PSTUPD on ***CONTROL_SHELL**. This has not been working properly for ELFORM = 16 since May 2021.

7.9 EM (Electromagnetic Solver)

1. Fixed bug that occurred when using PRECOND = 4 (Block Low-Rank LLT factorization) on ***EM_SOLVER_BEM** that led to wrong solutions for SMP and Hybrid executables.

2. Fixed issue leading to a potential crash when electromagnetic contact (***EM_CONTROL_CONTACT**) was used with source circuits (***EM_CIRCUIT_SOURCE**).

3. Fixed issue leading to a potential crash in BatMac models (***EM_RANDLES_BATMAC**) when the tabs were connected with voltage isopotential (***EM_ISOPOTENTIAL_CONNECT**).

7.10 ICFD (Incompressible Fluid Solver)

1. Fixed a memory issue in ***ICFD_DATABASE_DRAG_VOL**.

7.11 Implicit (Mechanical) Solver

1. Made fix in the output to the **d3eigv**. For ***ELEMENT_TSHELL_COMPOSITE** and ***ELEMENT_SHELL_COMPOSITE**, the parts in d3eigv file referenced internally created part IDs and not the original user part IDs.

2. Corrected error while loading the consistent mass terms (***CONTROL_IMPLICIT_CONSISTENT_MASS** into the mass matrix for IGA.

3. Fixed array reallocation issue for implicit mechanics with inertia relief (***CONTROL_IMPLICIT_INERTIA_RELIEF**.

4. Made two corrections to the output for ***CONTROL_IMPLICIT_-MODES**:
    - Reset the number of eigenvalues in the case of fewer computed than requested due to an interval specification so that the output matches the number computed.
    - Ensure the output or not of stress is consistent to keep the d3mode file correct.
5. Corrected error in loading masses into the mass matrix for eigenvalue problems (***CONTROL_IMPLICIT_EIGENVALUE**). Some nodal rotational inertias could have been skipped.
6. Fixed error in the Lanczos eigensolver when using only one processor.
7. Corrected implicit handling of the prescribed motion constraint between two rigid bodies (***BOUNDARY_PRESCRIBED_MOTION** with VAD = 5 and NODE1 and NODE2 specified) in MPP with more than one rank. The issue occurred when the rigid bodies were on different processors and led to an error.
8. Corrected use of Absolute Memory Restriction (ABSMEM on ***CONTROL_IMPLICIT_SOLVER**) for the MPP eigensolver. The absolute memory specification was not being honored.

7.12 Initial Cards

1. Fixed issue for ***INITIAL_{STRESS/STRAIN}_SHELL where** where "+" was missing in the long format dynain output.
2. Fixed issue for ***INITIAL_STRESS_SECTION** where IZSHEAR = 1 was not rotationally invariant.
3. Fixed issue for ***INITIAL_STRESS_SECTION** where unit scaling with ***INCLUDE_TRANSFORM** did not work.

7.13 Isogeometric Analysis (IGA)

1. Fixed bug in the initialization of laminated shell theory (LAMSHT ≠ 0 on ***CONTROL_SHELL**) for IGA shells.

7.14 Load Cards

1. Fixed issue where ***LOAD_BODY** acted on the added mass when using mass scaling with MS1ST = 1 on ***CONTROL_TIMESTEP**.

2. Fixed spurious error 20246 (STR+246) due to round-off error when using ***LOAD_MOVING_PRESSURE** in single precision.

7.15 Materials and Equations-of-State

1. Fixed incorrect error when using the TITLE keyword option with ***MAT_-ADD_COHESIVE**.

2. Fix for **_TITLE option** of ***MAT_ADD_DAMAGE_DIEM/GISSMO**: special characters, such as <, led to error termination.

3. Fix for large table ID ($> 2^{24}$) of Q1 < 0 on ***MAT_ADD_DAMAGE_-DIEM** when using a single precision executable.

4. Fix for ***MAT_ADD_DAMAGE_DIEM** do not extrapolate in strain rate tables (2D) given by P1.

5. Fixed issue with the combination of ***MAT_ADD_EROSION** and ***MAT_ADD_DAMAGE_DIEM**. In some instances, a criterion from ***MAT_ADD_EROSION** did not trigger failure because ***MAT_ADD_-DAMAGE_DIEM** overwrote the parameter NCS (number of conditions to satisfy) with the strain increment DEPS.

6. Let damage grow beyond 1.0 if ***MAT_ADD_DAMAGE_GISSMO** is used with DTYP = 0. The upper limit of 1.0 was added in 2016/2017 (after R9). This upper limit makes sense for real softening (DTYP > 0), but for DTYP = 0 values over 1.0 could be used for post-processing.

7. Fix for ***MAT_ADD_DAMAGE_GISSMO** with LP2BI ≠ 0. The Lode was not correctly being replaced by the bending indicator for LCREDG and FADEXP if needed.

8. Fix for ***MAT_ADD_DAMAGE_GISSMO**. Damage is driven by plastic strain (DTYP = 0 or 1) or a history variable (DTYP ≥ 10). If that history variable is not monotonically increasing (plastic strain always is), damage could grow even below the already reached maximum value. This is now prevented by remembering that maximum value. See also IFLG4 = 1 on ***MAT_ADD_GENERALIZED_DAMAGE**.

9. Fix for ***MAT_ADD_DAMAGE_GISSMO** when used in a full deck restart. Some variables (element size, regularization factor, fading exponent) were not correctly initialized for new parts.

10. Fixed issue for ***MAT_ADD_DAMAGE_GISSMO** for the unlikely event of using shells with Lode parameter dependent FADEXP < 0 (TABLE_3D) or LCREGD < 0 (TABLE_3D). Shell elements do not need Lode dependence anyway.

11. Fixed bug in IFLG3 = 1 of ***MAT_ADD_GENERALIZED_DAMAGE** for solid elements. This single damage parameter option was not working as intended.

12. Fixed the computation of the shear modulus for ***MAT_016** needed for contact stiffness.

13. Fixed issue in ***MAT_024_LOG_INTERPOLATION** with VP = 1 (table) for shell elements when using the implicit solver. The algorithm used the same interpolation as in the negative first strain rate approach, but here we have "regular" strain rates with a logarithmic interpolation.

14. Fixed issue for ***MAT_024_STOCHASTIC** with VP = 1, LCSS being a table, and shell elements. Scaling the yield stress could have been wrong.

15. Fixed bug in ***MAT_024** with MITER = 2 on ***CONTROL_SHELL**. Yield stress on history variable #5 could be zero if SIGY was not defined. That issue could lead to NaNs in the constitutive matrix for implicit.

16. Fixed convergence issue for ***MAT_030** for beams when plasticity occurred in the austenite phase.

17. Fixed issue with ***MAT_034** (***MAT_FABRIC**) combined with ***DEFINE_STAGED_CONSTRUCTION_PART** or ***LOAD_STIFFEN_PART**. The stiffness of ***MAT_FABRIC** elements did not follow the scaling requested on ***DEFINE_STAGED_CONSTRUCTION** and ***LOAD_STIFFEN_PART**.

18. Fixed issue for ***MAT_036** with table LCID > 0 and HR = 3 (strain rates) or HR = 8 (temperature). FCTTIM and FCTTEM from ***INCLUDE_TRANSFORM** were not taken into account for unit conversion.

19. For ***MAT_040** fixed potential divide by zero in dynamic relaxation with TRAMP = 0.

20. Fixed issue for ***MAT_USER_DEFINED_MATERIAL_MODELS** where strain energy was not calculated correctly for discrete beams.

21. Fixed bug for the combination of ***MAT_USER_DEFINED_MATERIAL_MODELS** with IORTHO = 1 and shell types 23 or 24. The number of integration points was incorrect for orthotropic user materials with these shells, leading to an error.

22. For ***MAT_054**:
    - Prevent failed integration points from reviving again in cyclic loading.
    - The internal energy was not computed correctly for solid elements when the SLIM limits were used.

23. For ***MAT_058** (***MAT_LAMINATED_COMPOSITE_FABRIC**) with shell elements when EA, EB, or GAB < 0:
    - Use the initial, undamaged stiffness to compute the sound speed/time step from the curves/tables. Prior to this fix, constant, hard-coded values were used, leading to an incorrect time step when shells with this material determine it.
    - Fixed bug in the table evaluation of EA, EB, or GAB < 0 for shell elements. Prior to this fix, the table may not have been evaluated correctly by not using the correct load curve associated with a specific strain rate.

24. Fixed behavior of ***MAT_081** with solid elements when using nonzero LCDM and EPPFR. If the plastic strain exceeds EPPFR, the element is deleted irrespective of LCDM. This changed behavior matches the behavior of this material for shell elements.

25. Revised the combination of ***MAT_083_LOG_LOG_INTERPOLATION** and implicit accuracy flag IACC = 1 (***CONTROL_ACCURACY**). The logarithmic interpolation was automatically deactivated. It is now enabled again.

26. For ***MAT_GENERAL_JOINT_DISCRETE_BEAM** (***MAT_097**) fixed the positioning of the history variables for neighboring parts and failure. Prior to this release, these could become offset, such as when using ***DEFINE_MATERIAL_HISTORIES**.

27. Fixed false initial element failure with ***MAT_100**. If used with MPP groupable tied contact (such as ***CONTACT_TIED_SHELL_EDGE_TO_SOLID**), elements were deleted prematurely in cycle 0.

28. Put yield stress on history variable #4 for ***MAT_106**. The manual already claimed it would be there, but that was only true for rare cases (shells with LCSIGY and ideal plasticity).

29. Prevent initial plastic strain from being overwritten for ***MAT_110**. It was falsely replaced by volumetric strain due to ***INITIAL_STRESS_SOLID**. We proactively corrected the same issue for related materials 111, 210, and 241.

30. Fixed the internal energy computation for ***MAT_123** with VP=1 and

ELFORM 13 tetrahedral solids. The energy could have been wrong (e.g., negative) if all elements in a vector block were elastic.

31. Fixed issue with ***MAT_124** for solid elements when SRFLAG = 2. We increased the maximum number of iterations from 20 to 100 for the (slow) Ridders' method. Potential non-convergence led to error termination.

32. Fixed issue with the unit conversion of table values (strain rates) in ***MAT_133**. Table HARD < 0 should take FCTTIM from ***INCLUDE_TRANSFORM** into account.

33. Fixed issue with ***MAT_133** when used with plain strain or axisymmetric elements (ELFORM = 13, 14, or 15 on ***SECTION_SHELL**). This combination did not work before. The stresses were most likely zero all the time.

34. Allow the use of ***MAT_133** and ***MAT_135** with thick shells 3, 5, and 7. Previously, this use resulted in a spurious error.

35. Fixed a bug in ***MAT_135**, which prevented the computation of eigenvalues.

36. Fixed a bug in the strain-rate-dependent strength evaluation in ***MAT_157** for the case where the natural logarithm of the strain rate is specified. An incorrect evaluation occurred when the strain rate was 0.

37. When using IHIS on ***MAT_157** (***MAT_ANISOTROPIC_ELASTIC_PLASTIC**) for the initialization of stiffness and/or material orientations (q_1 and q_2) together with laminated shell theory turned on (LAMSHT = 3/5 on ***CONTROLSHELL**), made the following fixes:

    - added proper initialization of parameters for laminated shell theory (shells),
    - corrected the contact stiffness (shells and solids), and
    - corrected values for hourglass stabilization (solids).

38. Fixed issue for ***MAT_159** when used with thick shell types 3, 5, or 7. Initialization of history variables was missing.

39. Fixed improper Poisson's ratio checks of material types 161, 162, and 219. Some valid materials error terminated unnecessarily.

40. Fixed false initial element failure with ***MAT_169** due to ICOH = 1 on ***CONTROL_SOLID**. If used with MPP groupable tied contact (e.g., ***CONTACT_TIED_SHELL_EDGE_TO_SOLID**), elements were deleted prematurely in cycle 0.

41. Allow large function IDs ($> 2^{24}$) for some ***MAT_169** entries when using single precision executables. This bug affects the negative input of TENMAX, GCTEN, SHRMAX, GCSHR, SHRP, SDFAC, and SGFAC.

42. Fixed triggering an incorrect error, KEY+315, for ***MAT_169** when SHRP is set to 0.9.

43. Fixed bug in ***MAT_172** and ***MAT_203** affecting thick shells (***ELEMENT_TSHELL**) only. Certain functions requiring knowledge of element size did not work correctly:

    - If CHARL > 0, the post-crack tensile stress did not decay correctly with crack opening. The response could even be perfectly plastic with no softening.
    - If AGGSZ > 0 and CHARL = 0, the maximum shear stress transferred across open cracks did not follow the expected relationship with crack opening.

44. Fixed bugs in ***MAT_179** affecting the stress extraction from tables and the energy initialization.

45. 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 weakened materials.

46. Fixed bug for ***MAT_193** that caused the time step for elements with this material to be too large, potentially causing an unstable response.

47. ***MAT_197** (***MAT_SEISMIC_ISOLATOR**) was generating spurious moments on node 2 of the discrete beam. This bug existed in R12, R13, and R14.0. A workaround was to apply rotational SPCs to node 2 of the isolator, but this method is only recommended if the new SPCs would not modify the behavior of the structure.

48. For ***MAT_213** (***MAT_COMPOSITE_TABULATED_PLASTICITY_DAMAGE**):

    - Addressed an issue with regard to the damage variables. In some rare cases, the damage variables could have been incorrectly retrieved from the element integration point history. An indication of this issue was a non-monotonic curve of damage variables over time plotted from the element integration point history.
    - Fixed an issue in the Generalized Tabulated Failure Criterion. If this criterion is used with strain-rate-dependent data and more than two curves are used in the corresponding tables, it could produce wrong results.

- Fixed an issue with post-processing. LS-PrePost expects the plastic strain to be in a specific position in the history data to be plotted as such. Before, the plastic strain was only available as a history variable written out when NEIPH was set on ***DATABASE_EXTENT_BINARY**. This fix is also needed if ***MAT_213** is used with ***CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION**.
- Resolved an issue when using this material with solid element type 2. After a discussion with all contributors to this material, we removed it from the B-bar treatment of solid element type 2. For some test cases, there were instabilities observed when using this material with the B-bar feature.

49. Fixed issue in ***MAT_4A_MICROMEC** (***MAT_215**) when used with shell elements for simulations that include dynamic relaxation. The problem occurred when the part reached plastic deformation during dynamic relaxation. This issue led to error termination.

50. Using a 3D table for LCI on ***MAT_215** could lead to incorrect results if the triaxiality started with a negative value.

51. Fixed issue with ***MAT_224**: allow adiabatic heating even if there is some external temperature somewhere else in the model.

52. Fixed bug for ***MAT_224** and ***MAT_224_GYS**. If LCI refers to a ***DEFINE_TABLE_3D**, results could have been wrong if the triaxility table started with a negative value.

53. Fixed issue for ***MAT_224** when used in a full deck restart. Temperature and element size were not correctly initialized for newly added parts.

54. Fix for penalty stiffness of ***MAT_240_THERMAL**: avoid potential zero stiffness by determining the maximum values from curves EMOD and GMOD.

55. Fixed the thickness computation in ***MAT_240** with THICK = 0.0 for pentahedral cohesive elements (ELFORM = 21 and 22).

56. For ***MAT_244** and ***MAT_248**, the hard phases (2 to 5) can now only accumulate plastic strain if they are currently present in the microstructure composition.

57. Fixed bug for the combination of ***MAT_251** and ***MAT_ADD_DAMAGE_GISSMO** with history options. If a history variable is provided via HISVN = 6 (material 251) and HISVN = -6 (GISSMO) and an initial value is specified on ***INITIAL_STRESS_SHELL/SOLID**, the history value of ***MAT_251** was overwritten by zero. It is now kept.

58. Fixed a bug in the latent heat calculation for the 3D version of ***MAT_254** for axisymmetric solids.

59. For ***MAT_262** (***MAT_LAMINATED_FRACTURED_DAIMLER_-CAMANHO**) with shell elements when EA, EB, or GAB < 0:

    - Use the initial, undamaged stiffness to compute the sound speed/time step from the curves/tables. Prior to this fix, incorrect values were used, leading to an incorrect time step when shells with this material determine it.
    - Fixed bug in the table evaluation of EA, EB, or GAB < 0 for shell elements. Prior to this fix, the table may not have been evaluated correctly by not using the correct load curve associated with a specific strain rate.

60. Fixed issue for ***MAT_274** where AOPT < 0 was not activated.

61. Fixed issue for ***MAT_280** with RATENL > 0 or FT < 0 to avoid negative strain rates.

62. Fixed issue with ***MAT_280** when used with thick shell elements. Sudden crack propagation through the thickness with option NIPF = 1 did not work as intended.

63. Fixed calcium ion (Ca^{2+}) interpolation for electrophysiology coupling with active models ACTYPE = 1 - 3 on ***MAT_295** when using solid element formulations 10 and 13. The issue caused incorrect results.

64. Fixed undesired transformations of ***MAT_SPRING/DAMPER** data. Factors from ***INCLUDE_TRANSFORM** could have been applied to these keywords when they were not in the include file to be transformed.

65. Fixed bug in using large curve IDs (> 16.7 million) in ***MAT_S15** (***MAT_-SPRING_MUSCLE**.

7.16 MPP

1. Fixed an MPP file handling issue that appeared in adaptive problems on systems running with no globally shared working directory. This bug exhibited itself as an error like this:

```
*** Error 21305 (STR+1305)
    Cannot open required auxiliary input file dyna.lsda
    on processor
```

2. Fixed an MPP hanging issue that could happen if some processors have

type 13 tets but some do not.

3. Fixed ordering issue that could cause MPP processing failures for models with a mix of ***CONSTRAINED_INTERPOLATION** and ***CONSTRAINED_INTERPOLATION_LOCAL.**

4. Fixed MPP decomposition issue that prevented any model from running that contained both SPH and IGA elements.

7.17 Output

1. Fixed error in temperature output for d3part that occurred for thermal analyses.

2. Fixed incorrect sign for the pressure in the dbfsi file that occurred if the segments in the segment set specified in ***DATABASE_FSI** have the opposite normal vectors from the ones defined in ***CONSTRAINED_LAGRANGE_IN_SOLID** or ***ALE_STRUCTURED_FSI**.

3. Fixed behavior output to nodout for shared nodes that belong to an accelerometer in MPP. For a shared node, the largest MPP rank that owns the node will output to nodout. This behavior is not correct if the node belongs to an accelerometer. In this case, the largest rank that owns the accelerometer outputs to the database.

4. Fixed issue for ***INTERFACE_SPRINGBACK** where reference geometry was not written correctly to ASCII file dynain_geo for MPP.

5. Fixed error in the INTFOR output when INMEMORY = 1 on ***CONTROL_ADAPTIVE**.

6. Fix a bug that caused zero energy in matsum for parts with ***ELEMENT_TSHELL_COMPOSITE**.

7. Fixed error in the output files when QUADSLD and CUBSLD of ***DATABASE_EXTENT_BINARY** are greater than 0. This bug affects MPP only. It corrupted the output files, leading LS-PrePost to crash.

8. Fixed incorrect moments/inertia/area output to secforc.

9. Fixed bug in outputting the contact force file ***DATABASE_BINARY_INTFOR**. If the analysis started from a nonzero time (such as in a staged construction analysis), the contact force file contained a very large number of states instead of being written at the time interval requested by the user.

10. Fixed issue that caused beam element topology written to the dynain file to

be corrupted when pulley elements are present.

11. Fixed issue in the averaged nodal stress/strain output specified with ***DATABASE_EXTENT_BINARY**. These are the options NODOUT = STRESS_GL, STRAIN_GL, and ALL_GL. It was not correctly working with tetrahedral or pentahedral solids.

12. Fix memory reporting in d3hsp. Previously, the total memory was reported to be the same as the "Additional dynamically allocated memory."

7.18 Restarts

1. Mortar contact did not work properly in small restarts.

2. When switching to implicit after an explicit phase, including during a small restart, a bug led to nonconvergence.

7.19 SPH

1. Corrected issue when ISHIFT = 1 in ***CONTROL_SPH** that could lead the program to crash.

2. Temperatures provided with the ***LOAD_THERMAL_VARIABLE_NODE** are now properly updated for SPH.

3. SMP axisymmetric cases were performing renormalization when FORM was anything other than zero in ***CONTROL_SPH**.

4. Select encrypted binary files were not opened properly during redecompositions when REMSPH = 1 in ***CONTROL_SPH_DECOMPOSITION_REDECOMPOSITION**, which likely meant that removal of inactive particles was not properly handled.

7.20 Thermal Solver

1. Fixed bug in solvers 12 through 16 of ***CONTROL_THERMAL_SOLVER** that led to the temperature staying stuck at the initial value.

2. Fixed a bug in SOLVER = 17 of ***CONTROL_THERMAL_SOLVER** that caused incorrect results in single precision.

3. Fixed issue where mechanical history variables could not be read by thermal materials for tetrahedrons and pentahedrons.

4. Fixed issue where the number of mechanical history variables that could be read by thermal materials was limited to 117.

5. Fixed bug for a full deck restart with thermal solver. When the time stepping is controlled by load curves (for dt, dtmax, dtmin, or tmax) in the restart input, the load curve data was probably incorrectly read.

6. Fixed a bug that could cause incorrect temperature results when a model contains both anisotropic and isotropic thermal materials.

7. Fixed a bug when filtering the dangling nodes out of the temperature boundary conditions. The bug sometimes resulted in very strange temperature distributions in MPP.

8. Fixed an inconsistency in the birth/death times for temperature boundary conditions when using the thermal solver. The birth/death condition is evaluated at the end time of the time step, but before the fix, the time integration parameter (TIP on ***CONTROL_THERMAL_SOLVER**) erroneously affected the condition.

7.21 User-Defined Features

1. Fixed issue where ***MODULE_USE** did not work for thermal materials.

2. Fixed issue for ***MODULE** where MMALE did not work correctly when a user material (or equation of state) spread to a part that initially did not have a user material.

3. Fixed two potential problems with `LD_LIBRARY_PATH` for MPP sharelib versions that led to the user subroutine not being loaded:

   - Using '::' (double colons) to separate paths
   - Including repeated paths, which has been broken since 2020

4. Fixed combination of ***PART_AVERAGED** and ***MODULE**. A missing part ID could lead to a crash or error (SOL+1378: `No user subroutine (usrmat) is loaded for part 0`).

5. Fixed format issue with ***USER_INTERFACE_FRICTION** when using `long = s`.

7.22 Miscellaneous

1. Fixed MPP support for part numbers greater than five digits in ***DE-**

FORMABLE_TO_RIGID_AUTOMATIC.

2. Fixed reinitialization issue with the flag that controls the output of the "relax" file during dynamic relaxation. It was not properly set up for ***CASE** support, so only the first ***CASE** having dynamic relaxation was writing this file.

3. Fixed segmentation fault triggered by CAV*i* < 0 on ***DEFINE_PRESSURE_TUBE**.

4. Fixed a bug in which ***RIGIDWALL_GEOMETRIC** does not recognize the INTERIOR option if it is placed after the DEFORM option.

5. Fixed issue with printing titles for ***DEFINE_CURVE_TITLE** to d3hsp. This did not work correctly if the ***DEFINE_CURVE_TITLE** directly preceded or directly followed a ***DEFINE_CURVE** without a title in the input deck.

6. Fixed issue with ***PART_STACKED_ELEMENTS** when used with ***SET_PART**. Part sets are modified if they contain PIDREF to add the newly generated parts instead. But the removal of PIDREF itself was missing.

7. 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 doesn't.

8. Fixed issue with ***DEFINE_SD_ORIENTATION**. In large models (> 16.8 million nodes), it was possible that the internal node IDs of NID1 and NID2 were the same, leading to error STR+509 (`orientation vector 8: nodes are coincident`).

9. Fixed issue with ***USER_INTERFACE_FRICTION**. Determining element history variables could go wrong if triangular shell elements were involved. Also, a large number of history variables were not taken care of correctly.

10. Made fixes for keyword format signs +, -, and %. Mixing those different format signs did not work in every case. See the section "GENERAL CARD FORMAT" in the manual for a description.

11. Avoid error termination when ***DEFINE_CONNECTION_PROPERTIES** is used with (D)SIGY < 0.

12. If a velocity curve was specified for ***DEFINE_GROUND_MOTION**, then it was incorrectly rejected for having unequal time steps (equal time steps are required), and a new velocity curve was generated unnecessarily.

13. Fixed incomplete output for warning SOL+1213. The recommended maximum time step for ***DAMPING_FREQUENCY_RANGE** was not being printed.