INTRODUCTION

CHRONOLOGICAL HISTORY

DYNA3D originated at the Lawrence Livermore National Laboratory [Hallquist 1976]. The early applications were primarily for the stress analysis of structures subjected to a variety of impact loading. These applications required what was then significant computer resources, and the need for a much faster version was immediately obvious. Part of the speed problem was related to the inefficient implementation of the element technology which was further aggravated by the fact that supercomputers in 1976 were much slower than today's PC. Furthermore, the primitive sliding interface treatment could only treat logically regular interfaces that are uncommon in most finite element discretizations of complicated three-dimensional geometries; consequently, defining a suitable mesh for handling contact was often very difficult. The first version contained trusses, membranes, and a choice of solid elements. The solid elements ranged from a one-point quadrature eight-noded element with hourglass control to a twenty-noded element with eight integration points. Due to the high cost of the twenty node solid, the zero energy modes related to the reduced 8-point integration, and the high frequency content which drove the time step size down, higher order elements were all but abandoned in later versions of DYNA3D. A two-dimensional version, DYNA2D, was developed concurrently.

A new version of DYNA3D was released in 1979 that was programmed to provide near optimal speed on the CRAY-1 supercomputers, contained an improved sliding interface treatment that permitted triangular segments and was an order of magnitude faster than the previous contact treatment. The 1979 version eliminated structural and higher order solid elements and some of the material models of the first version. This version also included an optional element-wise implementation of the integral difference method developed by Wilkins et al. [1974].

The 1981 version [Hallquist 1981a] evolved from the 1979 version. Nine additional material models were added to allow a much broader range of problems to be modeled including explosive-structure and soil-structure interactions. Body force loads were implemented for angular velocities and base accelerations. A link was also established from the 3D Eulerian code, JOY [Couch, et. al., 1983] for studying the structural response to impacts by penetrating projectiles. An option was provided for storing element data on disk thereby doubling the capacity of DYNA3D.

The 1982 version of DYNA3D [Hallquist 1982] accepted DYNA2D [Hallquist 1980] material input directly. The new organization was such that equations of state and constitutive models of any complexity could be easily added. Complete vectorization of the material models had been nearly achieved with about a 10 percent increase in execution speed over the 1981 version.

In the 1986 version of DYNA3D [Hallquist and Benson 1986], many new features were added, including beams, shells, rigid bodies, single surface contact, interface friction, discrete springs and dampers, optional hourglass treatments, optional exact volume integration, and VAX/ VMS, IBM, UNIX, COS operating systems compatibility, that greatly expanded its range of applications. DYNA3D thus became the first code to have a general single surface contact algorithm.

In the 1987 version of DYNA3D [Hallquist and Benson 1987] metal forming simulations and composite analysis became a reality. This version included shell thickness changes, the Belytschko-Tsay shell element [Belytschko and Tsay, 1981], and dynamic relaxation. Also included were non-reflecting boundaries, user specified integration rules for shell and beam elements, a layered composite damage model, and single point constraints.

New capabilities added in the 1988 DYNA3D [Hallquist 1988] version included a cost-effective resultant beam element, a truss element, a C^0 triangular shell, the BCIZ triangular shell [Bazeley et al. 1965], mixing of element formulations in calculations, composite failure modeling for solids, noniterative plane stress plasticity, contact surfaces with spot welds, tie break sliding surfaces, beam surface contact, finite stonewalls, stonewall reaction forces, energy calculations for all elements, a crushable foam constitutive model, comment cards in the input, and one-dimensional slidelines.

By the end of 1988 it was obvious that a much more concentrated effort would be required in the development of this software if problems in crashworthiness were to be properly solved; therefore, Livermore Software Technology Corporation (LSTC) was founded to continue the development of DYNA3D as a commercial version called LS-DYNA3D which was later shortened to LS-DYNA. The 1989 release introduced many enhanced capabilities including a one-way treatment of slide surfaces with voids and friction; cross-sectional forces for structural elements; an optional user-specified minimum time step size for shell elements using elastic and elastoplastic material models; nodal accelerations in the time history database; a compressible Mooney-Rivlin material model; a closed-form update shell plasticity model; a general rubber material model; unique penalty specifications for each slide surface; external work tracking; optional time step criterion for 4-node shell elements; and internal element sorting to allow full vectorization of right-hand-side force assembly.

Over the years, considerable progress has been made as may be seen in the chronology of the developments discussed below. In 2019, LSTC was acquired by Ansys, and this software was renamed Ansys LS-DYNA®.

Capabilities added in 1989-1990:

- arbitrary node and element numbers,
- fabric model for seat belts and airbags,
- composite glass model,
- vectorized type 3 contact and single surface contact,
- many more I/O options,
- all shell materials available for 8 node thick shell,
- strain rate dependent plasticity for beams,
- fully vectorized iterative plasticity,
- interactive graphics on some computers,
- nodal damping,
- shell thickness taken into account in shell type 3 contact,
- shell thinning accounted for in type 3 and type 4 contact,
- soft stonewalls,
- print suppression option for node and element data,
- massless truss elements, rivets – based on equations of rigid body dynamics,
- massless beam elements, spot welds – based on equations of rigid body dynamics,
- expanded databases with more history variables and integration points,
- force limited resultant beam,
- rotational spring and dampers, local coordinate systems for discrete elements,
- resultant plasticity for C^0 triangular element,
- energy dissipation calculations for stonewalls,
- hourglass energy calculations for solid and shell elements,
- viscous and Coulomb friction with arbitrary variation over surface,
- distributed loads on beam elements,
- Cowper and Symonds strain rate model,
- segmented stonewalls,
- stonewall Coulomb friction,
- stonewall energy dissipation,
- airbags (1990),
- nodal rigid bodies,

- automatic sorting of triangular shells into C^0 groups,
- mass scaling for quasi static analyses,
- user defined subroutines,
- warpage checks on shell elements,
- thickness consideration in all contact types,
- automatic orientation of contact segments,
- sliding interface energy dissipation calculations,
- nodal force and energy database for applied boundary conditions,
- defined stonewall velocity with input energy calculations,

Capabilities added in 1991-1992:

- rigid/deformable material switching,
- rigid bodies impacting rigid walls,
- strain-rate effects in metallic honeycomb model 26,
- shells and beams interfaces included for subsequent component analyses,
- external work computed for prescribed displacement/velocity/accelerations,
- linear constraint equations,
- MPGS database,
- MOVIE database,
- Slideline interface file,
- automated contact input for all input types,
- automatic single surface contact without element orientation,
- constraint technique for contact,
- cut planes for resultant forces,
- crushable cellular foams,
- urethane foam model with hysteresis,
- subcycling,
- friction in the contact entities,
- strains computed and written for the 8 node thick shells,
- “good” 4 node tetrahedron solid element with nodal rotations,
- 8 node solid element with nodal rotations,
- 2×2 integration for the membrane element,

- Belytschko-Schwer integrated beam,
- thin-walled Belytschko-Schwer integrated beam,
- improved TAURUS database control,
- null material for beams to display springs and seatbelts in TAURUS,
- parallel implementation on Crays and SGI computers,
- coupling to rigid body codes,
- seat belt capability.

Capabilities added in 1993-1994:

- Arbitrary Lagrangian Eulerian brick elements,
- Belytschko-Wong-Chiang quadrilateral shell element,
- Warping stiffness in the Belytschko-Tsay shell element,
- Fast Hughes-Liu shell element,
- Fully integrated thick shell element,
- Discrete 3D beam element,
- Generalized dampers,
- Cable modeling,
- Airbag reference geometry,
- Multiple jet model,
- Generalized joint stiffnesses,
- Enhanced rigid body to rigid body contact,
- Orthotropic rigid walls,
- Time zero mass scaling,
- Coupling with USA (Underwater Shock Analysis),
- Layered spot welds with failure based on resultants or plastic strain,
- Fillet welds with failure,
- Butt welds with failure,
- Automatic eroding contact,
- Edge-to-edge contact,
- Automatic mesh generation with contact entities,
- Drawbead modeling,
- Shells constrained inside brick elements,

- NIKE3D coupling for springback,
- Barlat's anisotropic plasticity,
- Superplastic forming option,
- Rigid body stoppers,
- Keyword input,
- Adaptivity,
- First MPP (Massively Parallel) version with limited capabilities.
- Built in least squares fit for rubber model constitutive constants,
- Large hysteresis in hyperelastic foam,
- Bilhku/Dubois foam model,
- Generalized rubber model,

Capabilities added in 1995:

- Belytschko - Leviathan Shell
- Automatic switching between rigid and deformable bodies.
- Accuracy on SMP machines to give identical answers on one, two or more processors.
- Local coordinate systems for cross-section output can be specified.
- Null material for shell elements.
- Global body force loads now may be applied to a subset of materials.
- User defined loading subroutine.
- Improved interactive graphics.
- New initial velocity options for specifying rotational velocities.
- Geometry changes after dynamic relaxation can be considered for initial velocities..
- Velocities may also be specified by using material or part ID's.
- Improved speed of brick element hourglass force and energy calculations.
- Pressure outflow boundary conditions have been added for the ALE options.
- More user control for hourglass control constants for shell elements.
- Full vectorization in constitutive models for foam, models 57 and 63.
- Damage mechanics plasticity model, material 81,
- General linear viscoelasticity with 6 term prony series.

- Least squares fit for viscoelastic material constants.
- Table definitions for strain rate effects in material type 24.
- Improved treatment of free flying nodes after element failure.
- Automatic projection of nodes in CONTACT_TIED to eliminate gaps in the surface.
- More user control over contact defaults.
- Improved interpenetration warnings printed in automatic contact.
- Flag for using actual shell thickness in single surface contact logic rather than the default.
- Definition by exempted part ID's.
- Airbag to Airbag venting/segmented airbags are now supported.
- Airbag reference geometry speed improvements by using the reference geometry for the time step size calculation.
- Isotropic airbag material may now be directly for cost efficiency.
- Airbag fabric material damping is specified as the ratio of critical damping.
- Ability to attach jets to the structure so the airbag, jets, and structure to move together.
- PVM 5.1 Madymo coupling is available.
- Meshes are generated within LS-DYNA3D for all standard contact entities.
- Joint damping for translational motion.
- Angular displacements, rates of displacements, damping forces, etc. in JNTFORC file.
- Link between LS-NIKE3D to LS-DYNA3D via *INITIAL_STRESS keywords.
- Trim curves for metal forming springback.
- Sparse equation solver for springback.
- Improved mesh generation for IGES and VDA provides a mesh that can directly be used to model tooling in metal stamping analyses.
- Capabilities added in 1996-1997 in Version 940:
- Part/Material ID's may be specified with 8 digits.
- Rigid body motion can be prescribed in a local system fixed to the rigid body.
- Nonlinear least squares fit available for the Ogden rubber model.
- Least squares fit to the relaxation curves for the viscoelasticity in rubber.
- Fu-Chang rate sensitive foam.
- 6 term Prony series expansion for rate effects in model 57-now 73

- Viscoelastic material model 76 implemented for shell elements.
- Mechanical threshold stress (MTS) plasticity model for rate effects.
- Thermoelastic-plastic material model for Hughes-Liu beam element.
- Ramberg-Osgood soil model
- Invariant local coordinate systems for shell elements are optional.
- Second order accurate stress updates.
- Four noded, linear, tetrahedron element.
- Co-rotational solid element for foam that can invert without stability problems.
- Improved speed in rigid body to rigid body contacts.
- Improved searching for the a_3, a_5 and a10 contact types.
- Invariant results on shared memory parallel machines with the *a_n* contact types.
- Thickness offsets in type 8 and 9 tie break contact algorithms.
- Bucket sort frequency can be controlled by a load curve for airbag applications.
- In automatic contact each part ID in the definition may have unique:
  - Static coefficient of friction
  - Dynamic coefficient of friction
  - Exponential decay coefficient
  - Viscous friction coefficient
  - Optional contact thickness
  - Optional thickness scale factor
  - Local penalty scale factor
- Automatic beam-to-beam, shell edge-to-beam, shell edge-to-shell edge and single surface contact algorithm.
- Release criteria may be a multiple of the shell thickness in types a_3, a_5, a10, 13, and 26 contact.
- Force transducers to obtain reaction forces in automatic contact definitions. Defined manually via segments, or automatically via part ID's.
- Searching depth can be defined as a function of time.
- Bucket sort frequency can be defined as a function of time.
- Interior contact for solid (foam) elements to prevent "negative volumes."
- Locking joint
- Temperature dependent heat capacity added to Wang-Nefske inflator models.
- Wang Hybrid inflator model [Wang, 1996] with jetting options and bag-to-bag venting.

- Aspiration included in Wang's hybrid model [Nusholtz, Wang, Wylie, 1996].
- Extended Wang's hybrid inflator with a quadratic temperature variation for heat capacities [Nusholtz, 1996].
- Fabric porosity added as part of the airbag constitutive model.
- Blockage of vent holes and fabric in contact with structure or itself considered in venting with leakage of gas.
- Option to delay airbag liner with using the reference geometry until the reference area is reached.
- Birth time for the reference geometry.
- Multi-material Euler/ALE fluids,
    - 2nd order accurate formulations.
    - Automatic coupling to shell, brick, or beam elements
    - Coupling using LS-DYNA contact options.
    - Element with fluid + void and void material
    - Element with multi-materials and pressure equilibrium
- Nodal inertia tensors.
- 2D plane stress, plane strain, rigid, and axisymmetric elements
- 2D plane strain shell element
- 2D axisymmetric shell element.
- Full contact support in 2D, tied, sliding only, penalty and constraint techniques.
- Most material types supported for 2D elements.
- Interactive remeshing and graphics options available for 2D.
- Subsystem definitions for energy and momentum output.
- Boundary element method for incompressible fluid dynamics and fluid-structure interaction problems.

Capabilities added during 1997-1998 in Version 950:

- Adaptive refinement can be based on tooling curvature with FORMING contact.
- The display of drawbeads is now possible since the drawbead data is output into the D3PLOT database.
- An adaptive box option, *DEFINE_BOX_ADAPTIVE, allows control over the refinement level and location of elements to be adapted.
- A root identification file, ADAPT.RID, gives the parent element ID for adapted elements.

- Draw bead box option, *DEFINE_BOX_DRAWBEAD, simplifies drawbead input.
- The new control option, CONTROL_IMPLICIT, activates an implicit solution scheme.
- 2D Arbitrary-Lagrangian-Eulerian elements are available.
- 2D automatic contact is defined by listing part ID's.
- 2D r-adaptivity for plane strain and axisymmetric forging simulations is available.
- 2D automatic non-interactive rezoning as in LS-DYNA2D.
- 2D plane strain and axisymmetric element with 2x2 selective-reduced integration are implemented.
- Implicit 2D solid and plane strain elements are available.
- Implicit 2D contact is available.
- The new keyword, *DELETE_CONTACT_2DAUTO, allows the deletion of 2D automatic contact definitions.
- The keyword, *LOAD_BEAM is added for pressure boundary conditions on 2D elements.
- A viscoplastic strain rate option is available for materials:
    - *MAT_PLASTIC_KINEMATIC
    - *MAT_JOHNSON_COOK
    - *MAT_POWER_LAW_PLASTICITY
    - *MAT_STRAIN_RATE_DEPENDENT_PLASTICITY
    - *MAT_PIECEWISE_LINEAR_PLASTICITY
    - *MAT_RATE_SENSITIVE_POWERLAW_PLASTICITY
    - *MAT_ZERILLI-ARMSTRONG
    - *MAT_PLASTICITY_WITH_DAMAGE
    - *MAT_PLASTICITY_COMPRESSION_TENSION
- Material model, *MAT_PLASTICITY_WITH_DAMAGE, has a piecewise linear damage curve given by a load curve ID.
- The Arruda-Boyce hyper-viscoelastic rubber model is available, see *MAT_-ARRUDA_BOYCE.
- Transverse-anisotropic-viscoelastic material for heart tissue, see *MAT_-HEART_-TISSUE.
- Lung hyper-viscoelastic material, see *MAT_LUNG_TISSUE.
- Compression/tension plasticity model, see *MAT_PLASTICITY_COMP-RESSION_TENSION.
- The Lund strain rate model, *MAT_STEINBERG_LUND, is added to Steinberg-Guinan plasticity model.

- Rate sensitive foam model, *MAT_FU_CHANG_FOAM, has been extended to include engineering strain rates, etc.
- Model, *MAT_MODIFIED_PIECEWISE_LINEAR_PLASTICITY, is added for modeling the failure of aluminum.
- Material model, *MAT_SPECIAL_ORTHOTROPIC, added for television shadow mask problems.
- Erosion strain is implemented for material type, *MAT_BAMMAN_DAMAGE.
- The equation of state, *EOS_JWLB, is available for modeling the expansion of explosive gases.
- The reference geometry option is extended for foam and rubber materials and can be used for stress initialization, see *INITIAL_FOAM_REFERENCE_GEOMETRY.
- A vehicle positioning option is available for setting the initial orientation and velocities, see *INITIAL_VEHICLE_KINEMATICS.
- A boundary element method is available for incompressible fluid dynamics problems.
- The thermal materials work with instantaneous coefficients of thermal expansion:
    - *MAT_ELASTIC_PLASTIC_THERMAL
    - *MAT_ORTHOTROPIC_THERMAL
    - *MAT_TEMPERATURE_DEPENDENT_ORTHOTROPIC
    - *MAT_ELASTIC_WITH_VISCOSITY
- Airbag interaction flow rate versus pressure differences.
- Contact segment search option, [bricks first optional]
- A through thickness Gauss integration rule with 1-10 points is available for shell elements. Previously, 5 were available.
- Shell element formulations can be changed in a full deck restart.
- The tied interface which is based on constraint equations, TIED_SURFACE_TO_SURFACE, can now fail if FAILURE, is appended.
- A general failure criterion for solid elements is independent of the material type, see *MAT_ADD_EROSION
- Load curve control can be based on thinning and a flow limit diagram, see *DEFINE_CURVE_FEEDBACK.
- An option to filter the spotweld resultant forces prior to checking for failure has been added the option, *CONSTRAINED_SPOTWELD, by appending FILTERED_FORCE, to the keyword.
- Bulk viscosity is available for shell types 1, 2, 10, and 16.

- When defining the local coordinate system for the rigid body inertia tensor a local coordinate system ID can be used. This simplifies dummy positioning.
- Prescribing displacements, velocities, and accelerations is now possible for rigid body nodes.
- One way flow is optional for segmented airbag interactions.
- Pressure time history input for airbag type, LINEAR_FLUID, can be used.
- An option is available to independently scale system damping by part ID in each of the global directions.
- An option is available to independently scale global system damping in each of the global directions.
- Added option to constrain global DOF along lines parallel with the global axes. The keyword is *CONSTRAINED_GLOBAL. This option is useful for adaptive remeshing.
- Beam end code releases are available, see *ELEMENT_BEAM.
- An initial force can be directly defined for the cable material, *MAT_CABLE_DISCRETE_BEAM. The specification of slack is not required if this option is used.
- Airbag pop pressure can be activated by accelerometers.
- Termination may now be controlled by contact, via *TERMINATION_CONTACT.
- Modified shell elements types 8, 10 and the warping stiffness option in the Belytschko-Tsay shell to ensure orthogonality with rigid body motions in the event that the shell is badly warped. This is optional in the Belytschko-Tsay shell and the type 10 shell.
- A one point quadrature brick element with an exact hourglass stiffness matrix has been implemented for implicit and explicit calculations.
- Automatic file length determination for D3PLOT binary database is now implemented. This insures that at least a single state is contained in each D3PLOT file and eliminates the problem with the states being split between files.
- The dump files, which can be very large, can be placed in another directory by specifying

  d=**/home/user /test/d3dump**

  on the execution line.
- A print flag controls the output of data into the MATSUM and RBDOUT files by part ID's. The option, PRINT, has been added as an option to the *PART keyword.
- Flag has been added to delete material data from the D3THDT file. See *DATABASE_EXTENT_BINARY and column 25 of the 19th control card in the structured input.

- After dynamic relaxation completes, a file is written giving the displaced state which can be used for stress initialization in later runs.

Capabilities added during 1998-2000 in Version 960:

Most new capabilities work on both the MPP and SMP versions; however, the capabilities that are implemented for the SMP version only, which were not considered critical for this release, are flagged below. These SMP unique capabilities are being extended for MPP calculations and will be available in the near future. The implicit capabilities for MPP require the development of a scalable eigenvalue solver, which is under development for a later release of LS-DYNA.

- Incompressible flow solver is available. Structural coupling is not yet implemented.
- Adaptive mesh coarsening can be done before the implicit springback calculation in metal forming applications.
- Two-dimensional adaptivity can be activated in both implicit and explicit calculations. (SMP version only)
- An internally generated smooth load curve for metal forming tool motion can be activated with the keyword: *DEFINE_CURVE_SMOOTH.
- Torsional forces can be carried through the deformable spot welds by using the contact type: *CONTACT_SPOTWELD_WITH_TORSION (SMP version only with a high priority for the MPP version if this option proves to be stable.)
- Tie break automatic contact is now available via the *CONTACT_AUTOMATIC_…_TIEBREAK options. This option can be used for glued panels. (SMP only)
- *CONTACT_RIGID_SURFACE option is now available for modeling road surfaces (SMP version only).
- Fixed rigid walls PLANAR and PLANAR_FINITE are represented in the binary output file by a single shell element.
- Interference fits can be modeled with the INTERFERENCE option in contact.
- A layered shell theory is implemented for several constitutive models including the composite models to more accurately represent the shear stiffness of laminated shells.
- Damage mechanics is available to smooth the post-failure reduction of the resultant forces in the constitutive model *MAT_SPOTWELD_DAMAGE.
- Finite elastic strain isotropic plasticity model is available for solid elements. *MAT_FINITE_ELASTIC_STRAIN_PLASTICITY.
- A shape memory alloy material is available: *MAT_SHAPE_MEMORY.

- Reference geometry for material, *MAT_MODIFIED_HONEYCOMB, can be set at arbitrary relative volumes or when the time step size reaches a limiting value. This option is now available for all element types including the fully integrated solid element.
- Non orthogonal material axes are available in the airbag fabric model. See *MAT_-FABRIC.
- Other new constitutive models include for the beam elements:
  - *MAT_MODIFIED_FORCE_LIMITED
  - *MAT_SEISMIC_BEAM
  - *MAT_CONCRETE_BEAM
- for shell and solid elements:
  - *MAT_ELASTIC_VISCOPLASTIC_THERMAL
- for the shell elements:
  - *MAT_GURSON
  - *MAT_GEPLASTIC_SRATE2000
  - *MAT_ELASTIC_VISCOPLASTIC_THERMAL
  - *MAT_COMPOSITE_LAYUP
  - *MAT_COMPOSITE_LAYUP
  - *MAT_COMPOSITE_DIRECT
- for the solid elements:
  - *MAT_JOHNSON_HOLMQUIST_CERAMICS
  - *MAT_JOHNSON_HOLMQUIST_CONCRETE
  - *MAT_INV_HYPERBOLIC_SIN
  - *MAT_UNIFIED_CREEP
  - *MAT_SOIL_BRICK
  - *MAT_DRUCKER_PRAGER
  - *MAT_RC_SHEAR_WALL
- and for all element options a very fast and efficient version of the Johnson-Cook plasticity model is available:
- *MAT_SIMPLIFIED_JOHNSON_COOK
- A fully integrated version of the type 16 shell element is available for the resultant constitutive models.
- A nonlocal failure theory is implemented for predicting failure in metallic materials. The keyword *MAT_NONLOCAL activates this option for a subset of elasto-plastic constitutive models.

- A discrete Kirchhoff triangular shell element (DKT) for explicit analysis with three in plane integration points is flagged as a type 17 shell element. This element has much better bending behavior than the C0 triangular element.
- A discrete Kirchhoff linear triangular and quadrilateral shell element is available as a type 18 shell. This shell is for extracting normal modes and static analysis.
- A C0 linear 4-node quadrilateral shell element is implemented as element type 20 with drilling stiffness for normal modes and static analysis.
- An assumed strain linear brick element is available for normal modes and statics.
- The fully integrated thick shell element has been extended for use in implicit calculations.
- A fully integrated thick shell element based on an assumed strain formulation is now available. This element uses a full 3D constitutive model which includes the normal stress component and, therefore, does not use the plane stress assumption.
- The 4-node constant strain tetrahedron element has been extended for use in implicit calculations.
- Relative damping between parts is available, see *DAMPING_RELATIVE (SMP only).
- Preload forces are can be input for the discrete beam elements.
- Objective stress updates are implemented for the fully integrated brick shell element.
- Acceleration time histories can be prescribed for rigid bodies.
- Prescribed motion for nodal rigid bodies is now possible.
- Generalized set definitions, i.e., SET_SHELL_GENERAL etc. provide much flexibility in the set definitions.
- The command "sw4." will write a state into the dynamic relaxation file, D3DRLF, during the dynamic relaxation phase if the D3DRLF file is requested in the input.
- Added mass by PART ID is written into the MATSUM file when mass scaling is used to maintain the time step size, (SMP version only).
- Upon termination due to a large mass increase during a mass scaled calculation a print summary of 20 nodes with the maximum added mass is printed.
- Eigenvalue analysis of models containing rigid bodies is now available using BCSLIB-EXT solvers from Boeing. (SMP version only).
- Second order stress updates can be activated by part ID instead of globally on the *CONTROL_ACCURACY input.
- Interface frictional energy is optionally computed for heat generation and is output into the interface force file (SMP version only).

- The interface force binary database now includes the distance from the contact surface for the FORMING contact options. This distance is given after the nodes are detected as possible contact candidates. (SMP version only).
- Type 14 acoustic brick element is implemented. This element is a fully integrated version of type 8, the acoustic element (SMP version only).
- A flooded surface option for acoustic applications is available (SMP version only).
- Attachment nodes can be defined for rigid bodies. This option is useful for NVH applications.
- CONSTRAINED_POINTS tie any two points together. These points must lie on a shell element.
- Soft constraint is available for edge to edge contact in type 26 contact.
- CONSTAINED_INTERPOLATION option for beam to solid interfaces and for spreading the mass and loads. (SMP version only).
- A database option has been added that allows the output of added mass for shell elements instead of the time step size.
- A new contact option allows the inclusion of all internal shell edges in contact type *CONTACT_GENERAL, type 26. This option is activated by adding "_INTERIOR" after the GENERAL keyword.
- A new option allows the use deviatoric strain rates rather than total rates in material model 24 for the Cowper-Symonds rate model.
- The CADFEM option for ASCII databases is now the default. Their option includes more significant figures in the output files.
- When using deformable spot welds, the added mass for spot welds is now printed for the case where global mass scaling is activated. This output is in the log file, d3hsp file, and the messag file.
- Initial penetration warnings for edge-to-edge contact are now written into the MESSAG file and the d3hsp file.
- Each compilation of LS-DYNA is given a unique version number.
- Finite length discrete beams with various local axes options are now available for material types 66, 67, 68, 93, and 95. In this implementation the absolute value of SCOOR must be set to 2 or 3 in the *SECTION_BEAM input.
- New discrete element constitutive models are available:
  - *MAT_ELASTIC_SPRING_DISCRETE_BEAM
  - *MAT_INELASTIC_SPRING_DISCRETE_BEAM
  - *MAT_ELASTIC_6DOF_SPRING_DISCRETE_BEAM
  - *MAT_INELASTIC_6DOF_SPRING_DISCRETE_BEAM
- The latter two can be used as finite length beams with local coordinate systems.

- Moving SPC's are optional in that the constraints are applied in a local system that rotates with the 3 defining nodes.
- A moving local coordinate system, CID, can be used to determine orientation of discrete beam elements.
- Modal superposition analysis can be performed after an eigenvalue analysis. Stress recovery is based on type 18 shell and brick (SMP only).
- Rayleigh damping input factor is now input as a fraction of critical damping, i.e. 0.10. The old method required the frequency of interest and could be highly unstable for large input values.
- Airbag option “SIMPLE_PRESSURE_VOLUME” allows for the constant CN to be replaced by a load curve for initialization. Also, another load curve can be defined which allows CN to vary as a function of time during dynamic relaxation. After dynamic relaxation CN can be used as a fixed constant or load curve.
- Hybrid inflator model utilizing CHEMKIN and NIST databases is now available. Up to ten gases can be mixed.
- Option to track initial penetrations has been added in the automatic SMP contact types rather than moving the nodes back to the surface. This option has been available in the MPP contact for some time. This input can be defined on the fourth card of the *CONTROL_CONTACT input and on each contact definition on the third optional card in the *CONTACT definitions.
- If the average acceleration flag is active, the average acceleration for rigid body nodes is now written into the D3THDT and NODOUT files. In previous versions of LS-DYNA, the accelerations on rigid nodes were not averaged.
- A capability to initialize the thickness and plastic strain in the crash model is available through the option *INCLUDE_STAMPED_PART, which takes the results from the LS-DYNA stamping simulation and maps the thickness and strain distribution onto the same part with a different mesh pattern.
- A capability to include finite element data from other models is available through the option, *INCLUDE_TRANSFORM. This option will take the model defined in an INCLUDE file: offset all ID's; translate, rotate, and scale the coordinates; and transform the constitutive constants to another set of units.

Features added during 2001-2002 for the 970 release of LS-DYNA:

Some of the new features, which are also listed below, were also added to later releases of version 960. Most new explicit capabilities work for both the MPP and SMP versions; however, the implicit capabilities for MPP require the development of a scalable eigenvalue solver and a parallel implementation of the constraint equations into the global matrices. This work is underway. A later release of version 970 is planned in 2003 that will be scalable for implicit solutions.

Below is list of new capabilities and features:

- MPP decomposition can be controlled using *CONTROL_MPP_DECOMPOSITION commands in the input deck.
- The MPP arbitrary Lagrangian-Eulerian fluid capability now works for airbag deployment in both SMP and MPP calculations.
- Euler-to-Euler coupling is now available through the keyword *CONSTRAINED_EULER_TO_EULER.
- Up to ten ALE multi-material groups may now be defined. The previous limit was three groups.
- Volume fractions can be automatically assigned during initialization of multi-material cells. See the GEOMETRY option of *INITIAL_VOLUME_FRACTION.
- A new ALE smoothing option is available to accurately predict shock fronts.
- DATABASE_FSI activates output of fluid-structure interaction data to ASCII file DBFSI.
- Point sources for airbag inflators are available. The origin and mass flow vector of these inflators are permitted to vary with time.
- A majority of the material models for solid materials are available for calculations using the SPH (Smooth Particle Hydrodynamics) option.
- The Element Free Galerkin method (EFG or meshfree) is available for two-dimensional and three-dimensional solids. This new capability is not yet implemented for MPP applications.
- A binary option for the ASCII files is now available. This option applies to all ASCII files and results in one binary file that contains all the information normally spread between a large number of separate ASCII files.
- Material models can now be defined by numbers rather than long names in the keyword input. For example the keyword *MAT_PIECEWISE_LINEAR_PLASTICITY can be replaced by the keyword: *MAT_024.
- An embedded NASTRAN reader for direct reading of NASTRAN input files is available. This option allows a typical input file for NASTRAN to be read directly and used without additional input. See the *INCLUDE_NASTRAN keyword.
- Names in the keyword input can represent numbers if the *PARAMETER option is used to relate the names and the corresponding numbers.
- Model documentation for the major ASCII output files is now optional. This option allows descriptors to be included within the ASCII files that document the contents of the file.
- ID's have been added to the following keywords:

- *BOUNDARY_PRESCRIBED_MOTION
- *BOUNDARY_PRESCRIBED_SPC
- *CONSTRAINED_GENERALIZED_WELD
- *CONSTRAINED_JOINT
- *CONSTRAINED_NODE_SET
- *CONSTRAINED_RIVET
- *CONSTRAINED_SPOTWELD
- *DATABASE_CROSS_SECTION
- *ELEMENT_MASS

- Penetration warnings for the contact option, ignore initial penetration, î are added as an option. Previously, no penetration warnings were written when this contact option was activated.
- Penetration warnings for nodes in-plane with shell mid-surface are printed for the AUTOMATIC contact options. Previously, these nodes were ignored since it was assumed that they belonged to a tied interface where an offset was not used; consequently, they should not be treated in contact.
- For the arbitrary spot weld option, the spot welded nodes and their contact segments are optionally written into the d3hsp file. See *CONTROL_CONTACT.
- For the arbitrary spot weld option, if a segment cannot be found for the spot welded node, an option now exists to error terminate. See *CONTROL_CONTACT.
- Spot weld resultant forces are written into the SWFORC file for solid elements used as spot welds.
- Solid materials have now been added to the failed element report.
- A new option for terminating a calculation is available, *TERMINATION_CURVE.
- A 10-noded tetrahedron solid element is available with either a 4 or 5 point integration rule. This element can also be used for implicit solutions.
- A new 4 node linear shell element is available that is based on Wilson's plate element combined with a Pian-Sumihara membrane element. This is shell type 21.
- A shear panel element has been added for linear applications. This is shell type 22. This element can also be used for implicit solutions.
- A null beam element for visualization is available. The keyword to define this null beam is *ELEMENT_PLOTEL. This element is necessary for compatibility with NASTRAN.
- A scalar node can be defined for spring-mass systems. The keyword to define this node is *NODE_SCALAR. This node can have from 1 to 6 scalar degrees-of-freedom.

- A thermal shell has been added for through-thickness heat conduction. Internally, 8 additional nodes are created, four above and four below the mid-surface of the shell element. A quadratic temperature field is modeled through the shell thickness. Internally, the thermal shell is a 12 node solid element.
- A beam OFFSET option is available for the *ELEMENT_BEAM definition to permit the beam to be offset from its defining nodal points. This has the advantage that all beam formulations can now be used as shell stiffeners.
- A beam ORIENTATION option for orienting the beams by a vector instead of the third node is available in the *ELEMENT_BEAM definition for NASTRAN compatibility.
- Non-structural mass has been added to beam elements for modeling trim mass and for NASTRAN compatibility.
- An optional checking of shell elements to avoid abnormal terminations is available. See *CONTROL_SHELL. If this option is active, every shell is checked each time step to see if the distortion is so large that the element will invert, which will result in an abnormal termination. If a bad shell is detected, either the shell will be deleted or the calculation will terminate. The latter is controlled by the input.
- An offset option is added to the inertia definition. See *ELEMENT_INERTIA_OFFSET keyword. This allows the inertia tensor to be offset from the nodal point.
- Plastic strain and thickness initialization is added to the draw bead contact option. See *CONTACT_DRAWBEAD_INITIALIZE.
- Tied contact with offsets based on both constraint equations and beam elements for solid elements and shell elements that have 3 and 6 degrees-of-freedom per node, respectively. See BEAM_OFFSET and CONSTRAINED_OFFSET contact options. These options will not cause problems for rigid body motions.
- The segment-based (SOFT = 2) contact is implemented for MPP calculations. This enables airbags to be easily deployed on the MPP version.
- Improvements are made to segment-based contact for edge-to-edge and sliding conditions, and for contact conditions involving warped segments.
- An improved interior contact has been implemented to handle large shear deformations in the solid elements. A special interior contact algorithm is available for tetrahedron elements.
- Coupling with MADYMO 6.0 uses an extended coupling that allows users to link most MADYMO geometric entities with LS-DYNA FEM simulations. In this coupling MADYMO contact algorithms are used to calculate interface forces between the two models.
- Release flags for degrees-of-freedom for nodal points within nodal rigid bodies are available. This makes the nodal rigid body option nearly compatible with the RBE2 option in NASTRAN.

- Fast updates of rigid bodies for metalforming applications can now be accomplished by ignoring the rotational degrees-of-freedom in the rigid bodies that are typically inactive during sheet metal stamping simulations. See the keyword: *CONTROL_RIGID.
- Center of mass constraints can be imposed on nodal rigid bodies with the SPC option in either a local or a global coordinate system.
- Joint failure based on resultant forces and moments can now be used to simulate the failure of joints.
- CONSTRAINED_JOINT_STIFFNESS now has a TRANSLATIONAL option for the translational and cylindrical joints.
- Joint friction has been added using table look-up so that the frictional moment can now be a function of the resultant translational force.
- The nodal constraint options *CONSTRAINED_INTERPOLATION and *CONSTRAINED_LINEAR now have a local option to allow these constraints to be applied in a local coordinate system.
- Mesh coarsening can now be applied to automotive crash models at the beginning of an analysis to reduce computation times. See the new keyword: *CONTROL_COARSEN.
- Force versus time seatbelt pretensioner option has been added.
- Both static and dynamic coefficients of friction are available for seat belt slip rings. Previously, only one friction constant could be defined.
- *MAT_SPOTWELD now includes a new failure model with rate effects as well as additional failure options.
- Constitutive models added for the discrete beam elements:
  - *MAT_1DOF_GENERALIZED_SPRING
  - *MAT_GENERAL_NONLINEAR_6DOF_DISCRETE_BEAM
  - *MAT_GENERAL_NONLINEAR_1DOF_DISCRETE_BEAM
  - *MAT_GENERAL_SPRING_DISCRETE_BEAM
  - *MAT_GENERAL_JOINT_DISCRETE_BEAM
  - *MAT_SEISMIC_ISOLATOR
- for shell and solid elements:
  - *MAT_PLASTICITY_WITH_DAMAGE_ORTHO
  - *MAT_SIMPLIFIED_JOHNSON_COOK_ORTHOTROPIC_DAMAGE
  - *MAT_HILL_3R
  - *MAT_GURSON_RCDC
- for the solid elements:

  - *MAT_SPOTWELD
  - *MAT_HILL_FOAM
  - *MAT_WOOD
  - *MAT_VISCOELASTIC_HILL_FOAM
  - *MAT_LOW_DENSITY_SYNTHETIC_FOAM
  - *MAT_RATE_SENSITIVE_POLYMER
  - *MAT_QUASILINEAR VISCOELASTIC
  - *MAT_TRANSVERSELY_ANISOTROPIC_CRUSHABLE_FOAM
  - *MAT_VACUUM
  - *MAT_MODIFIED_CRUSHABLE_FOAM
  - *MAT_PITZER_CRUSHABLE FOAM
  - *MAT_JOINTED_ROCK
  - *MAT_SIMPLIFIED_RUBBER
  - *MAT_FHWA_SOIL
  - *MAT_SCHWER_MURRAY_CAP_MODEL

- Failure time added to MAT_EROSION for solid elements.
- Damping in the material models *MAT_LOW_DENSITY_FOAM and *MAT_-LOW_DENSITY_VISCOUS_FOAM can now be a tabulated function of the smallest stretch ratio.
- The material model *MAT_PLASTICITY_WITH_DAMAGE allows the table definitions for strain rate.
- Improvements in the option *INCLUDE_STAMPED_PART now allow all history data to be mapped to the crash part from the stamped part. Also, symmetry planes can be used to allow the use of a single stamping to initialize symmetric parts.
- Extensive improvements in trimming result in much better elements after the trimming is completed. Also, trimming can be defined in either a local or global coordinate system. This is a new option in *DEFINE_CURVE_TRIM.
- An option to move parts close before solving the contact problem is available, see *CONTACT_AUTO_MOVE.
- An option to add or remove discrete beams during a calculation is available with the new keyword: *PART_SENSOR.
- Multiple jetting is now available for the Hybrid and Chemkin airbag inflator models.
- Nearly all constraint types are now handled for implicit solutions.
- Calculation of constraint and attachment modes can be easily done by using the option: *CONTROL_IMPLICIT_MODES.
- Penalty option, see *CONTROL_CONTACT, now applies to all *RIGIDWALL options and is always used when solving implicit problems.

- Solid elements types 3 and 4, the 4 and 8 node elements with 6 degrees-of-freedom per node are available for implicit solutions.
- The warping stiffness option for the Belytschko-Tsay shell is implemented for implicit solutions. The Belytschko-Wong-Chang shell element is now available for implicit applications. The full projection method is implemented due to it accuracy over the drill projection.
- Rigid to deformable switching is implemented for implicit solutions.
- Automatic switching can be used to switch between implicit and explicit calculations. See the keyword: *CONTROL_IMPLICIT_GENERAL.
- Implicit dynamics rigid bodies are now implemented. See the keyword *CONTROL_IMPLICIT_DYNAMIC.
- Eigenvalue solutions can be intermittently calculated during a transient analysis.
- A linear buckling option is implemented. See the new control input: *CONTROL_IMPLICIT_BUCKLE
- Implicit initialization can be used instead of dynamic relaxation. See the keyword *CONTROL_DYNAMIC_RELAXATION where the parameter, IDFLG, is set to 5.
- Superelements, i.e., *ELEMENT_DIRECT_MATRIX_INPUT, are now available for implicit applications.
- There is an extension of the option, *BOUNDARY_CYCLIC, to symmetry planes in the global Cartesian system. Also, automatic sorting of nodes on symmetry planes is now done by LS-DYNA.
- Modeling of wheel-rail contact for railway applications is now available, see *RAIL_TRACK and *RAIL_TRAIN.
- A new, reduced CPU, element formulation is available for vibration studies when elements are aligned with the global coordinate system. See *SECTION_SOLID and *SECTION_SHELL formulation 98.
- An option to provide approximately constant damping over a range of frequencies is implemented, see *DAMPING_FREQUENCY_RANGE.

Features added during 2003-2005 for the 971 release of LS-DYNA:

Initially, the intent was to quickly release version 971 after 970 with the implicit capabilities fully functional for distributed memory processing using MPI. Unfortunately, the effort required for parallel implicit was grossly underestimated, and, as a result, the release has been delayed. Because of the delay, version 971 has turned into a major release. Some of the new features, listed below, were also added to later releases of version 970. The new explicit capabilities are implemented in the MPP version and except for one case, in the SMP version as well.

INTRODUCTION

Below is list of new capabilities and features:

- A simplified method for using the ALE capability with airbags is now available with the keyword *AIRBAG_ALE.
- Case control using the *CASE keyword, which provides a way of running multiple load cases sequentially within a single run
- New option to forming contact: *CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURFACE_SMOOTH, which use fitted surface in contact calculation.
- Butt weld definition by using the *CONSTRAINED_BUTT_WELD option which makes the definition of butt welds simple relative to the option: *CONSTRAINED_GENERALIZED_WELD_BUTT.
- H-adaptive fusion is now possible as an option with the control input, *CONTROL_ADAPTIVE.
- Added a parameter on, *CONTROL_ADAPTIVE, to specify the number of elements generated around a 90 degree radius. A new option to better calculate the curvature was also implemented.
- Added a new keyword: *CONTROL_ADAPTIVE_CURVE, to refine the element along trimming curves
- Birth and death times for implicit dynamics on the keyword *CONTROL_IMPLICIT_DYNAMICS.
- Added an option to scale the spot weld failure resultants to account for the location of the weld on the segment surface, see *CONTROL_SPOTWELD_BEAM.
- Added an option which automatically replaces a single beam spot weld by an assembly of solid elements using the same ID as the beam that was replaced, see *CONTROL_SPOTWELD_BEAM.
- Boundary constraint in a local coordinate system using *CONSTRAINED_LOCAL keyword.
- A cubic spline interpolation element is now available, *CONSTRAINED_SPLINE.
- Static implicit analyses in of a structure with rigid body modes is possible using the option, *CONTROL_IMPLICIT_INERTIA_RELIEF.
- Shell element thickness updates can now be limited to part ID's within a specified set ID, see the *CONTROL_SHELL keyword. The thickness update for shells can now be optionally limited to the plastic part of the strain tensor for better stability in crash analysis.
- Solid element stresses in spot welds are optionally output in the local system using the SWLOCL parameter on the *CONTROL_SOLID keyword.
- SPOTHIN option on the *CONTROL_CONTACT keyword cards locally thins the spot welded parts to prevent premature breakage of the weld by the contact treatments.

- New function: *CONTROL_FORMING_PROJECT, which can initial move the penetrating slave nodes to the master surface
- New function *CONTROL_FORMING_TEMPLATE, which allows user to easily set up input deck. Its function includes auto-position, define travel curve, termination time, and most of the forming parameters for most of the typical forming process.
- New function *CONTROL_FORMING_USER, *CONTROL_FORMING_POSITION, and *CONTROL_FORMING_TRAVEL, when used together, can allow the user to define atypical forming process.
- Added new contact type *CONTACT_GUIDED_CABLE.
- Circular cut planes are available for *DATABASE_CROSS_SECTION definitions.
- New binary database FSIFOR for fluid structure coupling.
- Added *DATABASE_BINARY_D3PROP for writing the material and property data to the first D3PLOT file or to a new database D3PROP.
- DATABASE_EXTENT_BINARY has new flags to output peak pressure, surface energy density, nodal mass increase from mass scaling, thermal fluxes, and temperatures at the outer surfaces of the thermal shell.
- Eight-character alphanumeric labels can now be used for the parameters SECID, MID, EOSID, HGID, and TMID on the *PART keyword.
- Two NODOUT files are now written: one for high frequency output and a second for low frequency output.
- Nodal mass scaling information can now be optionally written to the D3PLOT file.
- Added option, MASS_PROPERTIES, to include the mass and inertial properties in the GLSTAT and SSSTAT files.
- Added option in *CONTROL_CPU to output the cpu and elapsed time into the GLSTAT file.
- Added an option, IERODE, on the *CONTROL_OUTPUT keyword to include eroded energies by part ID into the MATSUM file. Lumped mass kinetic energy is also in the MATSUM file as part ID 0.
- Added an option, TET10, on the *CONTROL_OUTPUT keyword to output ten connectivity nodes into D3PLOT database rather than 4.
- New keyword, *ELEMENT_SOLID_T4TOT10 to convert 4 node tetrahedron elements to 10 node tetrahedron elements.
- New keyword, *ELEMENT_MASS_PART defines the total additional non-structural mass to be distributed by an area weighted distribution to all nodes of a given part ID.
- New keyword option, SET, for *INTIAL_STRESS_SHELL_SET allows a set of shells to be initialized with the state of stress.

- New option allows the number of cpu's to be specified on the *KEYWORD input.
- Tubular drawbead box option for defining the elements that are included in the drawbead contact, see *DEFINE_BOX_DRAWBEAD.
- New function: *DEFINE_CURVE_DRAWBEAD, allow user to conveniently define drawbead by using curves (in x, y format or iges format)
- New function: *DEFINE_DRAWBEAD_BEAM, which allows user to conveniently define drawbead by using beam part ID, and specify the drawbead force.
- Analytic function can be used in place of load curves with the option *DEFINE_-CURVE_FUNCTION.
- Friction can now be defined between part pair using the *DEFINE_FRICTION input.
- New keyword: *DEFINE_CURVE_TRIM_3D, to allow trimming happens based on blank element normal, rather than use pre-defined direction
- A new trimming algorithm was added: *DEFINE_CURVE_TRIM_NEW, which allow seed node to be input and is much faster then the original algorithm.
- A new keyword, *DEFINE_HEX_SPOTWELD_ASSEMBLY, is available to define a cluster of solid elements that comprise a single spot weld.
- The definition of a vector, see *DEFINE_VECTOR, can be done by defining coordinates in a local coordinate system.
- The definition of a failure criteria between part pairs is possible with a table defined using the keyword, *DEFINE_SPOTWELD_FAILURE_RESULTANTS.
- A new keyword, *DEFINE_CONNECTION_PROPERTIES is available for defining failure properties of spot welds.
- Added *DEFINE_SET_ADAPTIVE to allow the adaptive level and element size to be specified by part ID or element set ID.
- Static rupture stresses for beam type spot welds can be defined in the keyword input, *DEFINE_SPOTWELD_RUPTURE_STRESS.
- Section properties can be define in the *ELEMENT_BEAM definitions for resultant beam elements using the SECTION option.
- Physical offsets of the shell reference surface can be specified on the shell element cards, see the OFFSET option on *ELEMENT_SHELL.
- File names can be located in remote directories and accessed through the *INCLUDE_PART keyword.
- New features to *INCLUDE_STAMPED_PART: two different mirror options, user-defined searching radius.
- *INTIAL_STRESS_SECTION allows for stress initialization across a cross-section, which consists of solid elements.

- An option, IVATN, is available for setting the velocities of slaved nodes and parts for keyword, *INITIAL_VELOCITY_GENERATION.
- Twenty-two built-in cross-section are now available in the definition of beam integration rules, see *INTEGRATION_BEAM.
- The possibility of changing material types is now available for shells using the user defined integration rule, see *INTEGRATION_SHELL.
- The interface springback file created by using the keyword, *INTERFACE_-SPRINGBACK is now optionally written as a binary file.
- An optional input line for *KEYWORD allows the definition of a prefix for all file names created during a simulation. This allows multiple jobs to be executed in the same directory.
- Body force loads can now be applied in a local coordinate system for *LOAD_-BODY.
- A pressure loading feature allows moving pressures to be applied to a surface to simulate spraying a surface with stream of fluid through a nozzle. See keyword *LOAD_MOVING_PRESSURE.
- Thermal expansion can be added to any material by the keyword, *MAT_ADD_-THERMAL_EXPANSION.
- Curves can now be used instead of eight digitized data points in the material model *MAT_ELASTIC_WITH_VISCOSITY_CURVE
- New options for spot weld failure in *MAT_SPOTWELD, which apply to beam and solid elements.
- Failure criteria based on plastic strain to failure is added to material *MAT_ANISOTROPIC_VISCOPLASTIC.
- Strain rate failure criterion is added to material *MAT_MODIFIED_PIECEWISE_-LINEAR_PLASTICITY.
- Strain rate scaling of the yield stress can now be done differently in tension and compression in material with separate pressure cut-offs in tension and compression in material model *MAT_PLASTICITY_TENSION_COMPRESSION.
- The RCDC model is now available to predict failure in material *MAT_PLASTICITY_WITH_DAMAGE.
- Two additional yield surfaces have been added to material *MAT_MODIFIED_-HONEYCOMB to provide more accurate predictions of the behavior of honeycomb barrier models.
- Unique coordinate systems can be assigned to the two nodal points of material *MAT_1DOF_GENERALIZED_SPRING.
- Poisson's ratio effects are available in foam defined by load curves in the material *MAT_SIMPLIFIED_RUBBER/FOAM

- Failure effects are available in the rubber/foam material defined by load curves in the *MAT_SIMPLIFIED_RUBBER/FOAM_WITH_FAILURE.
- The material option *MAT_ADD_EROSION now allows the maximum pressure at failure and the minimum principal strain at failure to be specified.
- Strains rather than displacements can now be used with the material model for discrete beams, *MAT_GENERAL_NONLINEAR_6DOF_DISCRETE_BEAM.
- New option for *MAT_TRANSVERSELY_ANISOTROPIC_ELASTIC_PLASTIC_(ECHANGE), which allow two ways to change the Young's modulus during forming simulation.
- New Material model: *MAT_HILL_3R: includes the shear term in the yield surface calculation by using Hill's 1948 an-isotropic material model.
- New Material model: *MAT_KINEMATIC_HARDENING_TRANSVERSELY_ANISOTROPIC: which integrates Mat #37 with Yoshida's two-surface kinematic hardening model.
- Improved formulation for the fabric material, *MAT_FABRIC for formulations 2, 3, and 4. The improved formulations are types 12, 13, and 14.
- Constitutive models added for truss elements:
  - *MAT_MUSCLE
- For beam elements
  - *MAT_MOMENT_CURVATURE
- For shell elements
  - *MAT_RESULTANT_ANISOTROPIC
  - *MAT_RATE_SENSITIVE_COMPOSITE_FABRIC.
  - *MAT_SAMP-1
  - *MAT_SHAPE_MEMORY is now implemented for shells.
- for shell and solid elements:
  - *MAT_BARLAT_YLD2000 for anisotropic aluminum alloys.
  - *MAT_SIMPLIFIED_RUBBER_WITH_DAMAGE
  - *MAT_VISCOELASTIC_THERMAL
  - *MAT_THERMO_ELASTO_VISCOPLASTIC_CREEP
- for the solid elements:
  - *MAT_ARUP_ADHESIVE
  - *MAT_BRAIN_LINEAR_VISCOELASTIC.
  - *MAT_CSCM for modeling concrete.

  - *MAT_PLASTICITY_COMPRESSION_TENSION_EOS for modeling ice.
  - *MAT_COHESIVE_ELASTIC
  - *MAT_COHESIVE_TH
  - *MAT_COHESIVE_GENERAL
  - *MAT_EOS_GASKET
  - *MAT_SIMPLIFIED_JOHNSON_COOK is now implemented for solids.
  - *MAT_PLASTICITY_WITH_DAMAGE is now implemented for solids.
  - *MAT_SPOTWELD_DAIMLERCHRYSLER

- User defined equations-of-state are now available.
- There is now an interface with the MOLDFLOW code.
- Damping defined in *DAMPING_PART_STIFFNESS now works for the Belytschko –Schwer beam element.
- The option *NODE_TRANSFORMATION allows a node set to be transformed based on a transformation defined in *DEFINE_TRANSFORMATION.
- Parameters can be defined in FORTRAN like expressions using *PARAMETER_EXPRESSION.
- A part can be moved in a local coordinate system in *PART_MOVE.
- A simplified method for defining composite layups is available with *PART_COMPOSITE
- The rigid body inertia can be changed in restart via *CHANGE_RIGID_BODY_INERTIA.
- A part set can now be defined by combining other part sets in *SET_PART_ADD.
- Termination of the calculation is now possible if a specified number of shell elements are deleted in a give part ID. See *TERMINATION_DELETED_SHELLS.
- Added hourglass control type 7 for solid elements for use when modeling hyperelastic materials.
- Shell formulations 4, 11, 16, and 17 can now model rubber materials.
- Added a new seatbelt pretensioner type 7 in which the pretensioner and retractor forces are calculated independently and added.
- A new composite tetrahedron element made up from 12 tetrahedron is now available as solid element type 17.
- Shell thickness offsets for *SECTION_SHELL now works for most shell elements, not just the Hughes-Liu shell.
- The Hughes-Liu beam has been extended to include warpage for open cross-sections.
- A resultant beam formulation with warpage is available as beam type 12.

- Two nonlinear shell elements are available with 8 degrees-of-freedom per node to include thickness stretch.
- Tetrahedron type 13, which uses nodal pressures, is now implemented for implicit applications.
- Cohesive solid elements are now available for treating failure.
- Seatbelt shell elements are available for use with the all seatbelt capabilities.
- Superelements can now share degrees-of-freedom and are implemented for implicit applications under MPI.
- A user defined element interface is available for solid and shell elements.
- Thermal shells are available for treating heat flow through shell elements.
- EFG shell formulations 41 and 42 are implemented for explicit analysis.
- EFGPACK is implemented in addition to BCSLIB-EXT solver on the keyword *CONTROL_EFG.
- EFG MPP version is available for explicit analysis.
- EFG fast transformation method is implemented in the EFG solid formulation.
- EFG Semi-Lagrangian kernel and Eulerian kernel options are added for the foam materials.
- EFG 3D adaptivity is implemented for the metal materials.
- EFG E.O.S. and *MAT_ELASTIC_FLUID materials are included in the 4-noded background element formulation.
- Airbag simulations by using ALE method can be switched to control volume method by *ALE_CV_SWITCH.
- *MAT_ALE_VISCOUS now supports Non-Newtonian viscosity by power law or load curve.
- *DATABASE_BINARY_FSIFOR outputs fluid-structure interaction data to binary file.
- *DATABASE_FSI_SENSOR outputs ALE element pressure to ASCII file dbsor.
- *MAT_GAS_MIXTURE supports nonlinear heat capacities.
- *INITIAL_VOLUME_FRACTION_GEOMETRY uses an enhanced algorithm to handle both concave and convex geometries and substantially reduce run time.
- A new keyword *DELETE_FSI allows the deletion of coupling definitions.
- Convection heat transfer activates by *LOAD_ALE_CONVECTION in ALE FSI analysis.
- *ALE_FSI_SWITCH_MMG is implemented to switch between ALE multi-material groups to treat immersed FSI problems.

- Type 9 option is added in *ALE_REFERENCE_SYSTEM_GROUP to deal complex ALE mesh motions including translation, rotation, expansion and contraction, etc.

    - New options in *CONSTRAINED_LAGRANGE_IN_SOLID
    - Shell thickness option for coupling type 4.
    - Bulk modulus based coupling stiffness.
    - Shell erosion treatment.
    - Enable/disable interface force file.

- New coupling method for fluid flowing through porous media are implemented as type 11 (shell) and type 12 (solid) in *CONSTRAINED_LAGRANGE_IN_SOLID.
- *ALE_MODIFIED_STRAIN allows multiple strain fields in certain ALE elements to solve sticking behavior in FSI. (MPP underdevelopment)
- *ALE_FSI_PROJECTION is added as a new constraint coupling method to solve small pressure variation problem. (MPP underdevelopment)
- *BOUNDARY_PRESCRIBED_ORIENTATION_RIGID is added as a means to prescribe as a function of time the general orientation of a rigid body using a variety of methods. This feature is available in release R3 and higher of Version 971.
- *BOUNDARY_PRESCRIBED_ACCELEROMETER_RIGID is added as a means to prescribe the motion of a rigid body based un experimental data gathered from accelerometers affixed to the rigid body. This feature is available in release R3 and higher of Version 971.

Capabilities added during 2008-2011 for Version 971R6 of LS-DYNA:

During the last four years the implicit capabilities are now scalable to a large number of cores; therefore, LS-DYNA has achieved a major goal over 15 years of embedding a scalable implicit solver. Also, in addition to the progress made for implicit solutions many other new and useful capabilities are now available.

- The keyword *ALE_AMBIENT_HYDROSTATIC initializes the hydrostatic pressure field in the ambient ALE domain due to an acceleration like gravity.
- The keyword *ALE_FAIL_SWITCH_MMG allows switching an ALE multi-material-group ID (AMMGID) if the material failure criteria occurs.
- The keyword *ALE_FRAGMENTATION allow switching from the ALE multi-material-group ID, AMMGID, (FR_MMG) of this failed material to another AMMGID (TO_MMG). This feature may typically be used in simulating fragmentation of materials.
- The keyword *ALE_REFINE refines ALE hexahedral solid elements automatically.

- The keyword *BOUNDARY_ALE_MAPPING maps ALE data histories from a previous run to a region of elements. Data are read from or written to a mapping file with a file name given by the prompt "map=" on the command line starting the execution.
- The keyword *BOUNDARY_PORE_FLUID is used to define parts that contain pore fluid where defaults are given on *CONTROL_PORE_FLUID input.
- With the keyword, *BOUNDARY_PRESCRIBED_FINAL_GEOMETRY, the final displaced geometry for a subset of nodal points is defined. The nodes of this subset are displaced from their initial positions specified in the *NODE input to the final geometry along a straight line trajectory. A load curve defines a scale factor as a function of time that is bounded between zero and unity corresponding to the initial and final geometry, respectively. A unique load curve can be specified for each node, or a default load curve can apply to all nodes.
- The keyword, *BOUNDARY_PWP defines pressure boundary conditions for pore water at the surface of the software.
- The keyword, *CONSTRAINED_JOINT_COOR, defines a joint between two rigid bodies. The connection coordinates are given instead of the nodal point IDs used in *CONSTRAINED_JOINT.
- The keyword, *CONSTRAINED_SPR2, defines a self-piercing rivet with failure. This model for a self-piercing rivet (SPR2) includes a plastic-like damage model that reduces the force and moment resultants to zero as the rivet fails. The domain of influence is specified by a diameter, which should be approximately equal to the rivet's diameter. The location of the rivet is defined by a single node at the center of two riveted sheets.
- Through the keyword, *CONTROL_BULK_VISCOSITY, bulk viscosity is optional for the Hughes-Liu beam and beam type 11 with warpage. This option often provides better stability, especially in elastic response problems.
- The display of nodal rigid bodies is activated by the parameter, PLOTEL, on the *CONTROL_RIGID keyword.
- The mortar contact, invoked by appending the suffix MORTAR to either FORMING_SURFACE_TO_SURFACE, AUTOMATIC_SURFACE_TO_SURFACE or AUTOMATIC_SINGLE_SURFACE, is a segment to segment penalty based contact. For two segments on each side of the contact interface that are overlapping and penetrating, a consistent nodal force assembly taking into account the individual shape functions of the segments is performed. In this respect the results with this contact may be more accurate, especially when considering contact with elements of higher order. By appending the suffix TIED to the CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR keyword or the suffix MORTAR to the CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK keyword, this is treated as a tied contact interface with tiebreak failure in the latter case. Only OPTION = 9 is supported for the mortar tiebreak contact. The mortar

contact is intended for implicit analysis in particular but is nevertheless supported for explicit analysis as well.

- In the database, ELOUT, the number of history variables can be specified for output each integration point in the solid, shell, thick shell, and beam elements. The number of variables is given on the *DATABASE_ELOUT keyword definition.
- A new option is available in *DATABASE_EXTENT_BINARY. Until now only one set of integration points were output through the shell thickness. The lamina stresses and history variables were averaged for fully integrated shell elements, which results in less disk space for the D3PLOT family of files, but makes it difficult to verify the accuracy of the stress calculation after averaging. An option is now available to output all integration point stresses in fully integrated shell elements: 4 x # of through thickness integration points in shell types 6, 7, 16, 18-21, and 3 x # of through thickness integration points in triangular shell types 3, and 17.
- The keyword *DATABASE_PROFILE allows plotting the distribution or profile of data along x, y, or z-direction.
- The purpose of the keyword, *DEFINE_ADAPTIVE_SOLID_TO_SPH, is to adaptively transform a Lagrangian solid Part or Part Set to SPH particles when the Lagrange solid elements comprising those parts fail. One or more SPH particles (elements) will be generated for each failed element to. The SPH particles replacing the failed element inherit all of the properties of failed solid element, e.g. mass, kinematic variables, and constitutive properties.
- With the keywords beginning with, *DEFINE_BOX, a LOCAL option is now available. With this option the diagonal corner coordinates are given in a local coordinate system defined by an origin and vector pair.
- The keyword, *DEFINE_CURVE_DUPLICATE, defines a curve by optionally scaling and offsetting the abscissa and ordinates of another curve defined by the *DEFINE_CURVE keyword.
- The keyword, *DEFINE_ELEMENT_DEATH, is available to delete a single element or an element set at a specified time during the calculation.
- The purpose of the keyword, *DEFINE_FRICTION_ORIENTATION, is to allow for the definition of different coefficients of friction (COF) in specific directions, specified using a vector and angles in degrees. In addition, COF can be scaled according to the amount of pressure generated in the contact interface.
- With the new keyword, *DEFINE_FUNCTION, an arithmetic expression involving a combination of independent variables and other functions, i.e.,

  f(a,b,c) = a*2 + b*c + sqrt(a*c)

  is defined where a, b, and c are the independent variables. This option is implemented for a subset of keywords.

- *ELEMENT_SEATBELT_SLIPRING
- *LOAD_BEAM
- *LOAD_MOTION_NODE
- *LOAD_MOVING_PRESSURE
- *LOAD_NODE
- *LOAD_SEGMENT
- *LOAD_SEGMENT_NONUNIFORM
- *LOAD_SETMENT_SET_NONUNIFORM
- *BOUNDARY_PRESCRIBED_MOTION

- If a curve ID is not found, then the function ID's are checked.
- The keyword, *DEFINE_SPH_TO_SPH_COUPLING, defines a penalty based contact to be used for the node to node contacts between SPH parts.
- The keyword, *DEFINE_TABLE_2D, permits the same curve ID to be referenced by multiple tables, and the curves may be defined anywhere in the input.
- The keyword, *DEFINE_TABLE_3D, provides a way of defining a three-dimensional table. A 2D table ID is specified for each abscissa value defined for the 3D table.
- The keyword, *ELEMENT_BEAM_PULLEY, allows the definition of a pulley for truss beam elements (see *SECTION_BEAM, ELFORM = 3). Currently, the beam pulley is implemented for *MAT_001 and *MAT_156. Pulleys allow continuous sliding of a string of truss beam element through a sharp change of angle.
- The purpose of the keyword, *ELEMENT_MASS_MATRIX, is to define a 6x6 symmetric nodal mass matrix assigned to a nodal point or each node within a node set.
- The keyword, *ELEMENT_DISCRETE_SPHERE, allows the definition of a discrete spherical element for discrete element calculations. Each particle consists of a single node with its mass, mass moment of inertia, and radius. Initial coordinates and velocities are specified via the nodal data.
- The two keywords, *ELEMENT_SHELL_COMPOSITE and *ELEMENT_TSHELL_COMPOSITE, are used to define elements for a general composite shell part where the shells within the part can have an arbitrary number of layers. The material ID, thickness, and material angle are specified for the thickness integration points for each shell in the part
- The keyword, *EOS_USER_DEFINED, allows a user to supply their own equation-of-state subroutine.
- The new keyword *FREQUENCY_DOMAIN provides a way of defining and solving frequency domain vibration and acoustic problems. The related keyword cards given in alphabetical order are:

  - *FREQUENCY_DOMAIN_ACOUSTIC_BEM_{*OPTION*}

- *FREQUENCY_DOMAIN_ACOUSTIC_FEM
- *FREQUENCY_DOMAIN_FRF
- *FREQUENCY_DOMAIN_RANDOM_VIBRATION
- *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM
- *FREQUENCY_DOMAIN_SSD

- The keyword, *INITIAL_AIRBAG_PARTICLE, initializes pressure in a closed airbag volume, door cavities for pressure sensing studies, and tires.
- The keyword *INITIAL_ALE_HYDROSTATIC initializes the hydrostatic pressure field in an ALE domain due to an acceleration like gravity.
- The keyword *INITIAL_ALE_MAPPING maps ALE data histories from a previous run. Data are read from a mapping file with a file name given by the prompt "map=" on the command line starting the execution.
- The keyword, *INITIAL_AXIAL_FORCE_BEAM, provides a simplified method to model initial tensile forces in bolts.
- The keyword, *INITIAL_FIELD_SOLID, is a simplified version of the *INITIAL_-STRESS_SOLID keyword which can be used with hyperelastic materials. This keyword is used for history variable input. Data is usually in the form of the eigenvalues of diffusion tensor data. These are expressed in the global coordinate system.
- The equation-of-state, *EOS_MIE_GRUNEISEN, type 16, is a Mie-Gruneisen form with a p-α compaction model.
- The keyword, *LOAD_BLAST_ENHANCED, defines an air blast function for the application of pressure loads due the explosion of conventional charge. While similar to *LOAD_BLAST this feature includes enhancements for treating reflected waves, moving warheads and multiple blast sources. The loads are applied to facets defined with the keyword *LOAD_BLAST_SEGMENT. A database containing blast pressure history is also available (see *DATABASE_BINARY_BLSTFOR).
- The keyword, *LOAD_ERODING_PART_SET, creates pressure loads on the exposed surface composed of solid elements that erode, i.e., pressure loads are added to newly exposed surface segments as solid elements erode.
- The keyword, *LOAD_SEGMENT_SET_ANGLE, applies traction loads over a segment set that is dependent on the orientation of a vector. An example application is applying a pressure to a cylinder as a function of the crank angle in an automobile engine
- The keyword, *LOAD_STEADY_STATE_ROLLING, is a generalization of *LOAD_BODY, allowing the user to apply body loads to part sets due to translational and rotational accelerations in a manner that is more general than the *LOAD_BODY capability. The *LOAD_STEADY_STATE_ROLLING keyword may be invoked an arbitrary number of times in the problem as long as no part

has the option applied more than once and they can be applied to arbitrary meshes. This option is frequently used to initialize stresses in tire.

- The keywords INTERFACE_SSI, INTERFACE_SSI_AUX, INTERFACE_SSI_-AUX_EMBEDDED and INTERFACE_SSI_STATIC are used to define the soil-structure interface appropriately in various stages of soil-structure interaction analysis under earthquake ground motion.
- The keyword, *LOAD_SEISMIC_SSI, is used to apply earthquake loads due to free-field earthquake ground motion at certain locations — defined by either nodes or coordinates — on a soil-structure interface. This loading is used in earthquake soil-structure interaction analysis. The specified motions are used to compute a set of effective forces in the soil elements adjacent to the soil-structure interface, according to the effective seismic input–domain reduction method.
- The keyword *DEFINE_GROUND_MOTION is used to specify a ground motion to be used in conjunction with *LOAD_SEISMIC_SSI.
- Material types *MAT_005 and *MAT_057 now accept table input to allow the stress quantity versus the strain measure to be defined as a function of temperature.
- The material option *MAT_ADD_EROSION, can now be applied to all nonlinear shell, thick shell, fully integrated solids, and 2D solids. New failure criteria are available.
- The GISSMO damage model, now available as an option in *MAT_ADD_EROSION, is a phenomenological formulation that allows for an incremental description of damage accumulation, including softening and failure. It is intended to provide a maximum in variability for the description of damage for a variety of metallic materials (e.g. *MAT_024, *MAT_036, …). The input of parameters is based on tabulated data, allowing the user to directly convert test data to numerical input.
- The keyword, *MAT_RIGID_DISCRETE or MAT_220, eliminates the need to define a unique rigid body for each particle when modeling a large number of rigid particles. This gives a large reduction in memory and wall clock time over separate rigid bodies. A single rigid material is defined which contains multiple disjoint pieces. Input is simple and unchanged, since all disjoint rigid pieces are identified automatically during initialization.
- The keyword, *NODE_MERGE, causes nodes with identical coordinates to be replaced during the input phase by the node encountered that has the smallest ID.
- The keyword, *PART_ANNEAL, is used to initialize the stress states at integration points within a specified part to zero at a given time during the calculation. This option is valid for parts that use constitutive models where the stress is incrementally updated. This option also applies to the Hughes-Liu beam elements, the integrated shell elements, thick shell elements, and solid elements.

- The keyword, *PART_DUPLICATE, provides a method of duplicating parts or part sets without the need to use the *INCLUDE_TRANSFORM option.
- To automatically generate elements to visualize rigid walls the DISPLAY option is now available for *RIGIDWALL_PLANAR and *RIGIDWALL_GEOMETRIC.
- A one point integrated pentahedron solid element with hourglass control is implemented as element type 115 and can be referenced in *SECTION_SOLID. Also, the 2 point pentahedron solid, type 15, no longer has a singular mode.
- The keyword *SECTION_ALE1D defines section properties for 1D ALE elements.
- The keyword *SECTION_ALE2D defines section properties for 2D ALE elements.
- The keywords *SET_BEAM_INTERSECT, *SET_SHELL_INTERSECT, *SET_SOLID_INTERSECT, *SET_NODE_INTERSECT, and *SET_SEGMENT_INTER-SECT, allows the definition of a set as the intersection, ∩, of a series of sets. The new set, SID, contains all common members.
- The keyword, *SET_SEGMENT_ADD, is now available for defining a new segment set by combining other segment sets.
- The two keywords, *DEFINE_ELEMENT_GENERALIZED_SHELL and *DEFINE_ELEMENT_GENERALIZED_SOLID, are used to define general shell and solid element formulations to allow the rapid prototyping of new element formulations. They are used in combination with the new keywords *ELEMENT_GENERLIZED_SHELL and *ELEMENT_GENERALIZED_SOLID.
- The two keywords, *ELEMENT_INTERPOLATION_SHELL and *ELEMENT_INTERPOLATION_SOLID, are used to interpolate stresses and other solution variables from the generalized shell and solid element formulations for visualization. They are used together with the new keyword *CONSTRAINED_NODE_INTERPOLATION.
- The keyword, *ELEMENT_SHELL_NURBS_PATCH, is used to define 3D shell elements based on NURBS (Non-Uniform Ration B-Spline) basis functions. Currently four different element formulations, with and without rotational degrees of freedom are available.
- The keyword LOAD_SPCFORC is used to apply equivalent SPC loads, read in from the d3dump file during a full-deck restart, in place of the original constraints in order to facilitate the classical non-reflecting boundary on an outside surface.

Capabilities added in 2012 to create Version 971R6.1, of LS-DYNA:

- A new keyword *MAT_THERMAL_DISCRETE_BEAM defines thermal properties for ELFORM 6 beam elements.
- An option *CONTROL_THERMAL_SOLVER, invoked by TSF < 0, gives the thermal speedup factor via a curve. This feature is useful when artificially scaling velocity in metal forming.

- A nonlinear form of Darcy's law in *MAT_ADD_PORE_AIR allows curves to define the relationship between pore air flow velocity and pore air pressure gradient.
- An extension to the PART option in *SET_SEGMENT_GENERAL allows reference to a beam part. This allows for creation of 2D segments for traction application.
- Options “SET_SHELL”, “SET_SOLID”, “SET_BEAM”, “SET_TSHELL”, “SET_-SPRING” are added to *SET_NODE_GENERAL so users can define a node set using existing element sets.
- Options “SET_SHELL”, “SET_SOLID”, “SET_SLDIO”, “SET_TSHELL”, “SET_-TSHIO” are added to *SET_SEGMENT_GENERAL so users can use existing element sets to define a segment set.
- *BOUNDARY_PRESCRIBED_MOTION_SET_BOX prescribes motion to nodes that fall inside a defined box.
- IPNINT > 1 in *CONTROL_OUTPUT causes d3hsp to list the IPNINT smallest element timesteps in ascending order.
- Section and material titles are echoed to d3hsp.
- A new parameter, MOARFL, in *DEFINE_CONNECTION_PROPERTIES permits reduction in modeled area due to shear.
- A new option HALF_SPACE in *FREQUENCY_DOMAIN_ACOUSTIC_BEM enables treatment of a half-space in boundary element method, frequency domain acoustic analysis.
- A shell script “kill_by_pid” is created during MPP startup. When executed, this script will run “kill -9” on every LS-DYNA process started as part of the MPP job. This is for use at the end of submission scripts, as a “fail safe” cleanup in case the job aborts.
- A new parameter IAVIS in *CONTROL_SPH selects the artificial viscosity formulation for the SPH particles. If set to 0, the Monaghan type artificial viscosity formulation is used. If set to 1, the standard artificial viscosity formulation for solid elements is used which may provide a better energy balance but is less stable in specific applications such as high velocity impact.
- Contact friction may be included in *CONTACT_2D_NODE_TO_SOLID for SPH.
- A new keyword *ALE_COUPLING_NODAL_CONSTRAINT provides a coupling mechanism between ALE solids and non-ALE nodes. The nodes can be from virtually any non-ALE element type including DISCRETE_SPHERE, EFG, and SPH, as well as the standard Lagrangian element types. In many cases, this coupling type may be a better alternative to *CONSTRAINED_LAGRANGE_IN_SOLID.
- The keyword *ALE_ESSENTIAL_BOUNDARY assigns essential boundary conditions to nodes of the ALE boundary surface. The command can be repeated multiple times and is recommended over use of EBC in *CONTROL_ALE.

- The keyword *DELETE_ALECPL in a small restart deck deletes coupling defined with *ALE_COUPLING_NODAL_CONSTRAINT. The command can also be used to reinstate the coupling in a later restart.
- *DEFINE_VECTOR_NODES defines a vector with two node points.
- *CONTACT_AUTOMATIC_SINGLE_SURFACE_TIED allows for the calculation of eigenvalues and eigenvectors for models that include *CONTACT_AUTOMATIC_SINGLE_SURFACE.
- A new parameter RBSMS in *CONTROL_RIGID affects rigid body treatment in Selective Mass Scaling (*CONTROL_TIMESTEP). When rigid bodies are in any manner connected to deformable elements, RBSMS = 0 (default) results in spurious inertia due to improper treatment of the nodes at the interface. RBSMS = 1 alleviates this effect but an additional cost is incurred.
- A new parameter T10JTOL in *CONTROL_SOLID sets a tolerance for issuing a warning when J_min/J_max goes below this tolerance value (i.e., quotient between minimum and maximum Jacobian value in the integration points) for tetrahedron type 16. This quotient serves as an indicator of poor tetrahedral element meshes in implicit that might cause convergence problems.
- A new option MISMATCH for *BOUNDARY_ACOUSTIC_COUPLING handles coupling of structural element faces and acoustic volume elements (ELFORMs 8 and 14) in the case where the coupling surfaces do not have coincident nodes.
- A porosity leakage formulation in *MAT_FABRIC (*MAT_034, FLC < 0) is now available for particle gas airbags (*AIRBAG_PARTICLE).
- *BOUNDARY_PRESCRIBED_ACCELEROMETER is disabled during dynamic relaxation.
- A new parameter CVRPER in *BOUNDARY_PAP defines porosity of a cover material encasing a solid part.
- A parameter TIEDID in *CONTACT_TIED_SURFACE_TO_SURFACE offers an optional incremental normal update in SMP to eliminate spurious contact forces that may appear in some applications.
- A new option SPOTSTP = 3 in *CONTROL_CONTACT retains spot welds even when the spot welds are not found by *CONTACT_SPOTWELD.
- The SMP consistency option (ncpu < 0) now pertains to the ORTHO_FRICTION contact option.
- Forces from *CONTACT_GUIDED_CABLE are now written to ncforc (both ASCII and binout).
- Discrete beam materials 70, 71, 74, 94, 121 calculate axial force based on change in length. Output the change in length instead of zero axial relative displacement to ASCII file disbout (*DATABASE_DISBOUT).
- *DATABASE_RCFORC_MOMENT is now supported in implicit.

- After the first implicit step, the output of projected cpu and wall clock times is written and the termination time is echoed.
- *DATABASE_MASSOUT is upgraded to include a summary table and to optionally add mass for nodes belonging to rigid bodies.
- Generate and store resultant forces for the LaGrange Multiplier joint formulation so as to give correct output to jntforc (*DATABASE_JNTFORC).
- Control the number of messages for deleted and failed elements using parameter MSGMAX in *CONTROL_OUTPUT.
- Nodal and resultant force output is written to nodfor for nodes defined in *DATABASE_NODAL_FORCE_GROUP in *FREQUENCY_DOMAIN_SSD analysis (SMP only).
- Ncforc data is now written for guided cables (*CONTACT_GUIDED_CABLE) in MPP.
- Jobid handling is improved in l2a utility so that binout files from multiple jobs, with or without a jobid-prefix, can be converted with the single command "l2a -j *binout*". The output contains the correct prefix according to the jobid.
- ALE_MULTI-MATERIAL_GROUP (AMMG) info is written to matsum (both ASCII and binout).
- Shell formulation 14 is switched to 15 (*SECTION_SHELL) in models that include axisymmetric SPH.
- *ELEMENT_BEAM_PULLEY is permitted with *MAT_CABLE_DISCRETE_BEAM.
- A warning during initialization is written if a user creates DKT triangles, either by ELFORM = 17 on *SECTION_SHELL or ESORT = 2 on *CONTROL_SHELL, that are thicker than the maximum edge length.
- Account is taken of degenerate acoustic elements with ELFORM 8. Tria and quad faces at acoustic-structure boundary are handled appropriately according to shape.
- The compression elimination option for 2D seatbelts, CSE = 2 in *MAT_SEATBELT is improved.
- Detailed material failure (*MAT_ADD_EROSION) messages in messag and d3hsp are suppressed when number of messages > MSGMAX (*CONTROL_OUTPUT).
- Implement SMP consistency (ncpu < 0) in *MAT_COHESIVE_GENERAL (*MAT_186) solids and shells.
- Viscoelastic model in *MAT_077_O now allows up to twelve terms in Prony series instead of standard six.
- Large curve IDs for friction table (*CONTACT_… with FS = 2) are enabled.
- Efficiency of GISSMO damage in *MAT_ADD_EROSION is improved.

- *MAT_ADD_PERMEABILITY_ORTHOTROPIC is now available for pore pressure analysis (*…_PORE_FLUID).
- For *MAT_224 solids and shells, material damage serves as the failure variable in *CONSTRAINED_TIED_NODES_FAILURE.
- The behavior of *MAT_ACOUSTIC is modified when used in combination with dynamic relaxation (DR). Acoustic domain now remains unperturbed in the DR phase but hydrostatic pressure from the acoustic domain is applied to the structure during DR.
- Option for 3D to 2D mapping is added in *INITIAL_ALE_MAPPING.
- *CONTACT_ERODING_NODES_TO_SURFACE contact may be used with SPH particles.
- Total Lagrangian SPH formulation 7 (*CONTROL_SPH) is now available in MPP.
- The output formats for linear equation solver statistics now accommodate very large numbers as seen in large models.
- *CONTROL_OUTPUT keyword parameter NPOPT is now applicable to thermal data. If NPOPT = 1, then printing of the following input data to d3hsp is suppressed:
    - *INITIAL_TEMPERATURE
    - *BOUNDARY_TEMPERATURE
    - *BOUNDARY_FLUX
    - *BOUNDARY_CONVECTION
    - *BOUNDARY_RADIATION
    - *BOUNDARY_ENCLOSURE_RADIATION
- Beam energy balance information is written to TPRINT file.
- MPP performance for LS-DYNA/MADYMO coupling is improved.
- Shell adaptivity (*CONTROL_ADAPTIVE) is improved to reduce the number of elements along curved surfaces in forming simulations.
- One-step unfolding (*CONTROL_FORMING_ONESTEP) is improved to accommodate blanks with small initial holes.
- Efficiency of FORM 3 isogeometric shells is improved.
- The processing of *SET_xxx_GENERAL is faster.
- *KEYWORD_JOBID now works even when using the *CASE command.
- Parts may be repositioned in a small restart by including *DEFINE_TRANSFORMATION and *NODE_TRANSFORM in the small restart deck to move nodes of a specified node set prior to continuing the simulation.

INTRODUCTION

Capabilities added during 2012/2013 to create LS-DYNA R7.0:

- Three solvers, EM, CESE, and ICFD, and a volume mesher to support the latter two solvers, are new in Version 7. Brief descriptions of those solvers are given below. Keyword commands for the new solvers are in Volume III of the LS-DYNA Keyword User's Manual. These new solvers are only included in double precision executables.
- Keyword family: *EM_, the keywords starting with *EM refer to and control the Electromagnetic solver problem set up:
  - EM Solver Characteristics:
    - Implicit
    - Double precision
    - Dynamic memory handling
    - SMP and MPP
    - 2D axisymmetric solver / 3D solver
    - Automatic coupling with structural and thermal LS-DYNA solvers
    - FEM for conducting pieces only, no air mesh needed (FEM-BEM system)
    - Solid elements for conductors, shells can be insulators
  - EM Solver Main Features:
    - Eddy Current (a.k.a Induction-Diffusion) solver
    - Induced heating solver
    - Resistive heating solver
    - Imposed tension or current circuits
    - Exterior field
    - Magnetic materials (beta version)
    - Electromagnetic contact
    - EM Equation of states (Conductivity as a function of temperature)
  - EM Solver Applications (Non-exhaustive) :
    - Electromagnetic forming
    - Electromagnetic welding
    - Electromagnetic bending
    - Inductive heating
    - Resistive heating
    - Rail-gun
    - Ring expansions
- Keyword family: *CESE_, the keywords starting with *CESE refer to and control the Compressible CFD solver problem set up:

- CESE Solver Characteristics:
  - Explicit
  - Double precision
  - Dynamic memory handling
  - SMP and MPP
  - 3D solver / special case 2D solver and 2D axisymmetric solver
  - Automatic coupling with structural and thermal LS-DYNA solvers
  - Eulerian fixed mesh or moving mesh (Either type input with *ELEMENT_SOLID cards or using *MESH cards)
- CESE Solver Main Features:
  - The CESE (Conservation Element / Solution Element) method enforces conservation in space-time
  - Highly accurate shock wave capturing
  - Cavitation model
  - Embedded (immersed) boundary approach or moving (fitting) approach for FSI problems
  - Coupled stochastic fuel spray solver (See *STOCHASTIC keywords)
  - Coupling with chemistry (See *CHEMISTRY keywords) solver
- CESE Solver Applications (Non-exhaustive):
  - Shock wave capturing
  - Shock/acoustic wave interaction
  - Cavitating flows
  - Conjugate heat transfer problems
  - Many different kinds of stochastic particle flows, e.g, dust, water, fuel.
  - Chemically reacting flows, e.g, detonating flow, supersonic combustion.

- Keyword family: *ICFD_, the keywords starting with *ICFD refer to and control the incompressible CFD solver problem set up:
  - ICFD Solver Characteristics:
    - Implicit
    - Double precision
    - Dynamic memory handling
    - SMP and MPP
    - 2D solver / 3D solver
    - Makes use of an automatic volume mesh generator for fluid domain (See *MESH keywords)
    - Coupling with structural and thermal LS-DYNA solvers

- ICFD Solver Main Features:

  - Incompressible fluid solver
  - Thermal solver for fluids
  - Free Surface flows
  - Two-phase flows
  - Turbulence models
  - Transient or steady-state problems
  - Non-Newtonian fluids
  - Boussinesq model for convection
  - Loose or strong coupling for FSI (Fluid-structure interaction)
  - Exact boundary condition imposition for FSI problems

- ICFD Solver Applications (Non-exhaustive) :

  - External aerodynamics for incompressible flows
  - Internal aerodynamics for incompressible flows
  - Sloshing, Slamming and Wave impacts
  - FSI problems
  - Conjugate heat transfer problems

- Keyword family: *MESH_, the keywords starting with *MESH refer to and control the tools for the automatic volume mesh generator for the CESE and ICFD solvers.

  - Mesh Generator Characteristics:

    - Automatic
    - Robust
    - Generic
    - Tetrahedral elements for 3D, Triangles in 2D
    - Closed body fitted mesh (surface mesh) needs to be provided for volume generation

  - Mesh Generator Main Features:

    - Automatic remeshing to keep acceptable mesh quality for FSI problems (ICFD only)
    - Adaptive meshing tools (ICFD only)
    - Anisotropic boundary layer mesh
    - Mesh element size control tools
    - Remeshing tools for surface meshes to ensure mesh quality

  - Mesh Generator Applications :

    - Used by the Incompressible CFD solver (ICFD).
    - Used by the Compressible CFD solver (CESE).

Other additions to Version 7 include:

- Add new parameter VNTOPT to *AIRBAG_HYBRID, that allows user more control on bag venting area calculation.
- Allow heat convection between environment and CPM bag (*AIRBAG_PARTICLE) bag. Apply proper probability density function to part's temperature created by the particle impact.
- *AIRBAG_PARTICLE and *SENSOR_SWITCH_SHELL_TO_VENT allows user to input load curve to control the venting using choking flow equation to get proper probability function for vents. Therefore, this vent will have the same vent rate as real vent hole.
- Add new option NP2P in *CONTROL_CPM to control the repartition frequency of CPM particles among processors (MPP only).
- Enhance *AIRBAG_PARTICLE to support a negative friction factor (FRIC or PFRIC) in particle to fabric contact. Particles are thus able to rebound at a trajectory closer to the fabric surface after contact.
- Use heat convection coefficient HCONV and fabric thermal conductivity KP to get correct effective heat transfer coefficient for heat loss calculation in *AIRBAG_PARTICLE. If KP is not given, H will be used as effective heat transfer coefficient.
- Extend CPM inflator orifice limit from 100 to unlimited (*AIRBAG_PARTICLE).
- Support dm_in_dt and dm_out_dt output to CPM chamber database (*DATABASE_ABSTAT) to allow user to study mass flow rate between multiple chambers.
- Previously, the number of ships (rigid bodies) in *BOUNDARY_MCOL, as specified by NMCOL, was limited to 2. Apparently, this was because the code had not been validated for more than 2 rigid bodies, but it is believed that it should not be a problem to remove this restriction. Consequently, this limit has been raised to 10, with the caveat that the user should verify the results for NMCOL > 2.
- Implemented a structural-acoustic mapping scheme (*BOUNDARY_ACOUSTIC_MAPPING), for mapping transient structural nodal velocity to acoustic volume surface nodes. This is useful if the structure finite element mesh and the acoustic boundary/finite element mesh are mismatched.
- *CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURACE_ORTHO_FRICTION can now be defined by part set IDs when supplemented by *DEFINE_FRICTION_ORIENTATION. Segment sets with orientation per *DEFINE_FRICTION_ORIENTATION are generated automatically.
- Contact force of *CONTACT_ENTITY is now available in intfor (*DATABASE_BINARY_INTFOR).
- *CONTACT_FORCE_TRANSDUCER_PENALTY will now accept node sets for both the slave and master sides, which should allow them to work correctly for eroding materials. BOTH sides should use node sets, or neither.

- Added option to create a backup penalty-based contact for a tied constraint-based contact in the input (IPBACK on Card E of *CONTACT).
- New option for *CONTACT_ENTITY. If variable SO is set to 2, then a constraint-like option is used to compute the forces in the normal direction. Friction is treated in the usual way.
- *CONTACT_ENTITY: allow friction coefficient to be given by a "coefficient vs time" load curve (input < 0 -> absolute value is the load curve ID). Also, if the friction coefficient bigger or equal 1.0, the node sticks with no sliding at all.
- Minor tweak to the way both MPP and SMP handle nodes sliding off the ends of beams in *CONTACT_GUIDED_CABLE.
- Frictional energy output in sleout (*DATABASE_SLEOUT) supported for *CONTACT_…_MORTAR.
- Tiebreak damage parameter output as "contact gap" in intfor file for *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK_MORTAR, OPTION = 9.
- Added MPP support for *CONTACT_2D_AUTOMATIC_SINGLE_SURFACE and *CONTACT_2D_AUTOMATIC_SURFACE_TO_SURFACE.
- Added keyword *CONSTRAINED_MULTIPLE_GLOBAL for defining multi-node constraints for imposing periodic boundary conditions.
- Enhancement for *CONSTRAINED_INTERPOLATION_SPOTWELD (SPR3): calculation of bending moment is more accurate now.
- If *CONSTRAINED_NODAL_RIGID_BODY nodes are shared by several processors with mass scaling on, the added mass is not summed up across processors. This results in an instability of the NRB. (MPP only)
- *ALE_REFINE has been replaced and expanded upon by the *CONTROL_REFINE family of commands. These commands invoke local mesh refinement of shells, solids, and ALE elements based on various criteria.
- Shells or solids in a region selected for refinement (parent element) are replaced by 4 shells or 8 solids, respectively. *CONTROL_REFINE_SHELL applies to shells, *CONTROL_REFINE_SOLID applies to solids and *CONTROL_REFINE_ALE and *CONTROL_REFINE_ALE2D applies to ALE elements. Each keyword has up to 3 lines of input. If only the 1st card is defined, the refinement occurs during the initialization. The 2nd card defines a criterion CRITRF to automatically refine the elements during the run. If the 3rd card is defined, the refinement can be reversed based on a criterion CRITM. All commands are implemented for SMP and MPP, but SMP is not parallelized.
- *CONTROL_REFINE_MPP_DISTRIBUTION distributes the elements required by the refinement across the MPP processes.

- Eliminate automatic writing of a d3plot plot state after each 3D tetrahedral remeshing operation (*CONTROL_REMESHING) to reduce volume of output.
- Generate disbout output (*DATABASE_DISBOUT) for MPP and SMP binout files.
- Extend *DATABASE_MASSOUT to include option to output mass information on rigid body nodes.
- Added new keyword *CHANGE_OUTPUT for full deck restart to override default behavior of overwriting existing ASCII files. For small restart, this option has no effect since all ASCII output is appended to the result of previous run already.
- Added new option (NEWLENGD) to 2nd field of 3rd card of *CONTROL_OUTPUT to write more detailed legend in ASCII output files. At present, only rcforc and jntforc are implemented.
- Increased default binary file size scale factor (x=) from 7 to 1024. That means the default binary file size will be 1 Gb for single version and 2 Gb for double version.
- Add echo of new "max frequency of element failure summaries" flag (FRFREQ in *CONTROL_OUTPUT) to d3hsp file.
- Support LSDA/binout output for new pllyout file (*DATABASE_PLLYOUT, *ELEMENT_BEAM_PULLEY) in both SMP and MPP.
- Allow degenerated hexahedrons (pentas) for cohesive solid elements (ELFORM = 19, 20) that evolve from an extrusion of triangular shells. The input of nodes on the element cards for such a pentahedron is given by: N1, N2, N3, N3, N4, N5, N6, N6.
- Add new option to activate drilling constraint force for shells in explicit calculations. This can be defined by parameters DRCPSID (part set) and DRCPRM (scaling factor) on *CONTROL_SHELL.
- Add SMP ASCII database "pllyout" (*DATABASE_PLLYOUT) for *ELEMENT_BEAM_PULLEY.
- *FREQUENCY_DOMAIN_ACOUSTIC_BEM:
  - Added an option to output real part of acoustic pressure in time domain.
  - Enabled BEM acoustic computation following implicit transient analysis.
  - Implemented coupling between steady state dynamics and collocation acoustic BEM.
  - Implemented Acoustic Transfer Vector (ATV) to variational indirect BEM acoustics.
  - Enabled boundary acoustic mapping in BEM acoustics.
- *FREQUENCY_DOMAIN_ACOUSTIC_FEM:
  - Added boundary nodal velocity to binary plot file d3acs.
  - Implemented pentahedron elements in FEM acoustics.
  - Enabled using boundary acoustic mapping in FEM acoustics.

- *FREQUENCY_DOMAIN_FRF:
    - Updated FRF to include output in all directions (VAD2 = 4).
    - Added treatment for FRF with base acceleration (node id can be 0).
- *FREQUENCY_DOMAIN_RANDOM_VIBRATION:
    - Updated calculation of PSD and RMS von Mises stress in random vibration environment, based on Sandia National Laboratories report, 1998.
- *FREQUENCY_DOMAIN_RANDOM_VIBRATION_FATIGUE:
    - Implemented an option to incorporate initial damage ratio in random vibration fatigue.
- *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM:
    - Implemented double sum methods (based on Gupta-Cordero coefficient, modified Gupta-Cordero coefficient, and Rosenblueth-Elorduy coefficient).
    - Updated calculating von Mises stress in response spectrum analysis.
    - Implemented treatment for multi simultaneous input spectra.
    - Improved double sum methods by reducing number of loops.
- *FREQUENCY_DOMAIN_SSD:
    - Added the option to output real and imaginary parts of frequency response to d3ssd.
    - Added the option to output relative displacement, velocity and acceleration in SSD computation in the case of base acceleration. Previously only absolute values were provided.
- Implemented keyword *FREQUENCY_DOMAIN_MODE_{OPTION} so that user can select the vibration modes to be used for frequency response analysis.
- Implemented keyword *SET_MODE_{OPTION} so that user can define a set of vibration modes, to be used for frequency response analysis.
- Implemented keyword *FREQUENCY_DOMAIN_PATH to define the path of binary databases containing mode information, used in restarting frequency domain analysis, e.g. frf, ssd, random vibration.
- Compute normal component of impulse for oblique plates in *INITIAL_MINE_IMPULSE. The feature is no longer limited to horizontal plates.
- Disable license security for *INITIAL_IMPULSE_MINE. The feature is no longer restricted.

- Enabled hourglass type 7 to work well with *INITIAL_FOAM_REFERENCE_GEOMETRY so that initial hourglass energy is properly calculated and foam will spring back to the initial geometry.
- Accommodate erosion of thin shells in *LOAD_BLAST_ENHANCHED.
- *LOAD_VOLUME_LOSS has been changed such that after the analysis time exceeds the last point on the curve of volume change fraction versus time, the volume change is no longer enforced.
- *LOAD_BODY_POROUS new option AOPT added to assign porosity values in material coordinate system.
- Added *LOAD_SEGMENT_FILE.
- Add new sensor definition, *SENSOR_DEFINE_ANGLE. This card traces the angle formed between two lines.
- *SENSOR_DEFINE_NODE can be used to trace the magnitude of nodal values (coordinate, velocity or accleration) when VID is “0” or undefined.
- Add two new parameters to *SENSOR_DEFINE_ELEMENT, scale factor and power, so that user can adjust the element-based sensor values (strain, stress, force, …).
- Change history variables 10-12 in *MAT_054/*MAT_ENHANCED_COMPOSITE_DAMAGE (thin shells only) to represent strains in material coordinate system rather than in local element coordinate system. This is a lot more helpful for post-processing issues. This change should not lead to different results other than due to different round-off errors.
- New features and enhancements to *MAT_244/*_MAT_UHS_STEEL:
    - Added implicit support for MAT_244.
    - Changed the influence of the austenite grain size in Mat244 according to Li et al.
    - Changed the start temperatures to fully follow WATT et al and Li et al.
    - Hardness calculation is now improved when noncontinuous cooling is applied i.e., tempering.
    - Added temperature dependent Poisson ratio and advanced reaction kinetics.
    - Added new advanced option to describe the thermal expansion coefficients for each phase.
    - Added option to use Curve ID or a Table ID for describing the latent heat generation during phase transormations.
    - Added support for table definition for Youngs modulus. Now you can have one temperature dependent curve for each of the 5 phases
- Added support for implicit to *MAT_188.

- Added material model *MAT_273/*MAT_CDPM/*MAT_CONCRETE_DAMAGE_PLASTIC_MODEL. This model is aimed at simulations where failure of concrete structures subjected to dynamic loadings is sought. The model is based on effective stress plasticity and has a damage model based on both plastic and elastic strain measures. Implemented for solids only but both for explicit and implicit simulations. Using an implicit solution when damage is activated may trigger a slow convergense. IMFLAG = 4 or 5 can be useful.
- Added an option in *MAT_266 (*MAT_TISSUE_DISPERSED) so that the user can tailor the active contribution with a time dependent load curve instead of using the internal hardcoded option. See ACT10 in the User's Manual.
- *MAT_173/*MAT_MOHR_COULOMB is available in 2D.
- Enable *MAT_103 and *MAT_104 to discretize the material load curves according to the number of points specified by LCINT in *CONTROL_SOLUTION.
- Implement Prony series up to 18 terms for shells using *MAT_076/*MAT_GENERAL_VISCOELASTIC.
- Added *DEFINE_STOCHASTIC_VARIATION and the STOCHASTIC option for *MATs 10, 15, 24, 81, 98 for shells, solids, and type 13 tets. This feature defines a stochastic variation in the yield stress and damage/failure of the aforementioned material models.
- Add Moodification for *DEFINE_CONNECTION_PROPERTIES, PROPRUL = 2: thinner weld partner is first partner, PROPRUL = 3: bottom (nodes 1-2-3-4) weld partner is first partner.
- Add spotweld area to debug output of *DEFINE_CONNECTION_PROPERTIES which is activated by *CONTROL_DEBUG.
- Add support of *MAT_ADD_EROSION option NUMFIP < 0 for standard (non-GISSMO) failure criteria. Only for shells.
- Improve implicit convergence of *MAT_ADD_EROSION damage model GISSMO by adding damage scaling (1-D) to the tangent stiffness matrix.
- Provide plastic strain rates (tension/compression, shear, biaxial) as history variables no. 16, 17, and 18 for *MAT_187.
- Add new variables to user failure routine matusr_24 (activated by FAIL < 0 on *MAT_024 and other materials): integration point numbers and element id.
- Add new energy based, nonlocal failure criterion for *MAT_ADD_EROSION, parameters ENGCRT (critical energy) and RADCRT (critical radius) after EPSTHIN. Total internal energy of elements within a radius RADCRT must exceed ENGCRT for erosion to occur. Intended for windshield impact.
- Add new option to *MAT_054 for thin shells: Load curves for rate dependent strengths and a rate averaging flag can be defined on new optional card 9.

- Add new option for *MAT_MUSCLE: Input parameter SSP < 0 can now refer to a load curve (stress vs. stretch ratio) or a table (stress vs. stretch ratio vs. normalized strain rate).
- Expand list of variables for *MAT_USER_DEFINED_MATERIAL_MODELS by characteristic element size and element ID.
- Enable *MAT_USER_DEFINED_MATERIAL_MODELS to be used with tetrahedron element type 13. New sample routines “umat41_t13” and “umat41v_t13” show corresponding pressure calculation in the elastic case.
- Add a new feature to *MAT_125 allowing C1 and C2 to be used in calculation of back stress. When plastic strain < 0.5%, C1 is used, otherwise C2 is used as described in Yoshida's paper.
- Extend non-linear strain path (_NLP_FAILURE) in *MAT_037 to implicit.
- *MAT_173/*MAT_MOHR_COULOMB now works in ALE. A new option has been added to suppress the tensile limit on hydrostatic stress recommended for ALE multi-material use.
- Upgraded *MAT_172/*MAT_CONCRETE_EC2.
    - Corrections to DEGRAD option.
    - Concrete and reinforcement types 7 and 8 have been added to reflect changes to Eurocode 2.
    - Extra history variables for reinforcement stress and strain are now output as zero for zero-fraction reinforcement directions.
- Added RCDC model for solid *MAT_082.
- Added Feng's failure model to solid *MAT_021.
- Added *MAT_027 for beams.
- Added *DEFINE_HAZ_PROPERTIES and *DEFINE_HAZ_TAILOR_WELDED_-BLANK for modifying material behavior near a spot weld.
- Added fourth rate form to viscoplastic Johnson-Cook model (*MAT_015).
- Added option to *MAT_224 to not delete the element if NUMINT = -200.
- New damage initiation option 3 in multi fold damage criteria in *MAT_ADD_-EROSION. Very similar to option 2 but insensitive to pressure.
- Added rotational resistance in *MAT_034/*MAT_FABRIC. Optionally the user may specify the stiffness, yield and thickness of and elastic-perfectly-plastic coated layer of a fabric that results in a rotational resistance during the simulation.
- FLDNIPF < 0 in *MAT_190/*MAT_FLD_3-PARAMETER_BARLAT for shell elements means that failure occurs when all integration points within a relative distance of -FLDNIPF from the mid surface has reached the fld criterion.

- A computational welding mechanics *MAT_270/*MAT_CWM material is available that allows for element birth based on a birth temperature as well as annealing based on an annealing temperature. The material is in addition a thermo-elasto-plastic material with kinematic hardening and temperature dependent properties.
- Added *MAT_271/*MAT_POWDER, a material for manufacturing (i.e., compaction and sintering) of cemented carbides. It is divided into an elastic-plastic compaction model that is supposed to be run in a first phase, and a viscoelastic sintering model that should be run in a second phase. This model is for solid elements.
- For IHYPER = 3 on a *MAT_USER_DEFINED_… shell material, the deformation gradient is calculated from the geometry instead of incremented by the velocity gradient. The deformation gradient is also passed to the user defined subroutines in the global system together with a transformation matrix between the global and material frames. This allows for freedom in how to deal with the deformation gradient and its transformations in orthotropic (layered) materials.
- The Bergstrom-Boyce viscoelastic rubber model is now available in explicit and implicit analysis as *MAT_269/*MAT_BERGSTROM_BOYCE_RUBBER. The Arruda-Boyce elastic stress is augmented with a Bergstrom-Boyce viscoelastic stress corresponding to the response of a single entangled chain in a polymer gel matrix.
- Added a new parameter IEVTS to *MAT_USER_DEFINED_MATERIAL_MODELS (*MAT_041-050). IEVTS is optional and is used only by thick shell formulation 5. It points to the position of E(a) in the material constants array. Following E(a), the next 5 material constants must be E(b), E(c), v(ba), v(ca), and v(cb). This data enables thick shell formulation 5 to calculate an accurate thickness strain, otherwise the thickness strain will be based on the elastic constants pointed to by IBULK and IG.
- Implemented enhancements to fabric material (*MAT_034), FORM = 14. Stress-strain curves may include a portion for fibers in compression. When unload/reload curves with negative curve ID are input (curve stretch options), the code that finds the intersection point now extrapolates the curves at their end rather than simply printing an error message if an intersection point cannot be found before the last point in either curve.
- Map 1D to 3D by beam-volume averaging the 1D data over the 3D elements (*INITIAL_ALE_MAPPING).
- In a 3D to 3D mapping (*INITIAL_ALE_MAPPING), map the relative displacements for the penalty coupling in *CONSTRAINED_LAGRANGE_IN_SOLID.
- The [name].xy files associated with *DATABASE_ALE_MAT are now created when sense switches sw1, sw2, quit, or stop are issued.
- *ALE_ESSENTIAL_BOUNDARY is available in 2D.
- *DATABASE_FSI is available for 2D (MPP).

- *ALE_ESSENTIAL_BOUNDARY implemented to apply slip-only velocity BC along ALE mesh surface.
- *CONTROL_ALE flag INIJWL = 2 option added to balance initial pressure state between ALE Soil and HE.
- Include SPH element (*ELEMENT_SPH) in time step report.
- Time step and internal energy of 2D axisymmetric SPH elements are calculated in a new way more consistent with the viscosity force calculation.
- Only apply viscosity force to x and y components of 2D axisymmetric SPH element, not on hoop component.
- MAXV in *CONTROL_SPH can be defined as a negative number to turn off velocity checking.
- Improve calculation of 2D axisymmetric SPH contact force in *DEFINE_SPH_TO_SPH_COUPLING.
- Added the following material models for SPH particles: *MAT_004/*MAT_ELASTIC_PLASTIC_THERMAL (3D only) and *MAT_106/*MAT_ELASTIC_VISCOPLASTIC_THERMAL
- Added a new parameter DFACT for *DEFINE_SPH_TO_SPH_COUPLING. DFACT invokes a viscous term to damp the coupling between two SPH parts and thereby reduce the relative velocity between the parts.
- Added BOUNDARY_CONVECTION and BOUNDARY_RADIATION for explicit SPH thermal solver.
- *CONTROL_REMESHING_EFG:
  - Add eroding failed surface elements and reconstructing surface in EFG adaptivity.
  - Add a control parameter for monotonic mesh resizing in EFG adaptivity.
  - Add searching and correcting self-penetration for adaptive parts in 3D tetrahedron remeshing.
- Enhance 3D axisymmetric remeshing with 6-node/8-node elements
- *CONTROL_REMESHING:
  - Use RMIN/RMAX along with SEGANG to determine element size.
  - Remove the restriction that the reference point of computational model must be at original point $(0,0,0)$.
  - Rewrite the searching algorithm for identifying the feature lines of cross-sections in order to provide more stable remeshing results.
- Improve rigid body motion in EFG shell type 41.

- Support EFG pressure smoothing in EFG solid type 42 for *MAT_ELASTIC_VISCOPLASTIC_THERMAL.
- Add visco effect for implicit EFG solid type 42.
- Add new EFG solid type 43 (called Meshfree-Enriched FEM, MEFEM) for both implicit and explicit. This element formulation is able to relieve the volumetric locking for nearly incompressible material (eg. rubber) and performs strain smoothing across elements with common faces.
- EFG shell adaptivity no longer requires a special license.
- Application of EFG in an implicit analysis no longer requires a special license.
- Add *SENSOR_CONTROL for prescribed motion constraints in implicit.
- Update *INTERFACE_LINKING_NODE in implicit to catch up with explicit, including adding scaling factors.
- Add support for *DATABASE_RCFORC_MOMENT for implicit.
- Enhance Iterative solvers for Implicit Mechanics.
- Add, after the first implicit time step, the output of projected cpu and wall clock times. This was already in place for explicit. Also echo the termination time.
- Add variable MXDMP in *CONTROL_THERMAL_SOLVER to write thermal conductance matrix and right-hand side every MXDMP time steps.
- Add keyword *CONTROL_THERMAL_EIGENVALUE to calculate eigenvalue(s) of each thermal conductance matrix.
- Added thermal material model *MAT_THERMAL_ORTHOTROPIC_TD_LC. This is an orthotropic material with temperature dependent properties defined by load curves.
- Changed structured file format for control card 27 (first thermal control card). Several input variables used i5 format limiting their value to 99,999. A recent large model exceeded this limit. The format was changed to i10. This change is not backward compatible. Old structured input files will no longer run unless control card 27 is changed to the new i10 format. This change does not affect the KEYWORD file.
- Add thermal material *MAT_T07/*MAT_THERMAL_CWM for welding simulations, to be used in conjunction with mechanical counterpart *MAT_270/*MAT_-CWM.
- Modify decomposition costs of *MAT_181 and *MAT_183.
- Introduce new timing routines and summary at termination.
- Echo “MPP contact is groupable” flag to d3hsp
- Bodies using *MAT_RIGID_DISCRETE were never expected to share nodes with non-rigid bodies, but this now works in MPP.

- There is no longer any built-in limitation on the number of processors that may be used in MPP.
- Echo contents of the MPP pfile (including keyword additions) to the d3hsp and mes0000 files.
- Add new keyword *CONTROL_MPP_PFILE, which allows for insertion of text following this command to be inserted into the MPP pfile (p = pfile).
- Change in MPP treatment of *CONSTRAINED_TIEBREAK. They now share a single MPI communicator, and a single round of communication. This should improve performance for problems with large numbers of these, without affecting the results.
- Added two input variables for *CONTROL_FORMING_ONESTEP simulation, TSCLMIN is a scale factor limiting the thickness reduction and EPSMAX defines the maximum plastic strain allowed.
- Added output of strain and stress tensors for onestep solver *CONTROL_FORMING_ONESTEP, to allow better evaluation of formability.
- Improved *CONTACT_AUTO_MOVE: before changes the termination time, and it causes problems when several tools need to be moved. Now *CONTACT_AUTO_MOVE does not change the termination time, but changes the current time. In this way, several tools can be moved without the need to worry about the other tool's move. This is especially useful in multi-flanging and hemming simulations.
- Made improvements to previously undocumented keyword *INTERFACE_BLANKSIZE, including adding the options INITIAL_TRIM, and INITIAL_ADAPTIVE. This keyword was developed for blank size development in sheet metal forming. Generally, for a single forming process, only the option DEVELOPMENT is needed and inputs are an initial estimated blank shape, a formed blank shape, and a target blank shape in either mesh or boundary coordinates. Output will be the calculated/corrected initial blank shape. Initial blank mesh and formed blank mesh can be different (e.g. adaptive). For a multi-stamping process involving draw, trimming and flanging, all three options are needed. Related commands for blank size estimation are *CONTROL_FORMING_ONESTEP, and for trim line development, *CONTROL_FORMING_UNFLANGING.
- Made improvements and added features to previously undocumented keyword *CONTROL_FORMING_UNFLANGING, this keyword unfolds flanges of a deformable blank, e.g., flanged or hemmed portions of a sheet metal part, onto a rigid tooling mesh using the implicit static solver. It is typically used in trim line mapping during a draw die development process. The roots of the flanges or hemmed edges are automatically processed based on a user input of a distance tolerance between the flanges/hemmed edges and rigid tool. It includes the ability to handle a vertical flange wall. Other keywords related to blank size development are, *CONTROL_FORMING_ONESTEP, and *INTERFACE_BLANKSIZE_DEVELOPMENT.

- Added keyword *CONTROL_FORMING_OUTPUT which allows control of d3plot output by specifying distances to tooling home. It works with automatic position of stamping tools using *CONTROL_FORMING_AUTOPOSITION_PARAMETER.
- Added the LOCAL_SMOOTH option to *INTERFACE_COMPENSATION_NEW which features smoothing of a tool's local area mesh, which could otherwise become distorted due to, e.g., bad/coarse mesh of the original tool surface, tooling pairs (for example, flanging post and flanging steel) do not maintain a constant gap and several compensation iterations. This new option also allows for multiple regions to be smoothed. Local areas are defined by *SET_LIST_NODE_SMOOTH.
- Added output to rcforc for *DEFINE_DE_TO_SURFACE_COUPLING.
- Implement traction surface for *DEFINE_DE_TO_SURFACE_COUPLING.
- Add keyword *DATABASE_BINARY_DEMFOR with command line option dem = dem_int_force. This will turn on the DEM interface force file for DEM coupling option. The output frequency is controlled by the new keyword.
- Add new feature *DEFINE_DE_INJECTION to allow DEM particle dropping from user defined plane.
- Add new option_VOLUME to *ELEMENT_DISCRETE_SPHERE. This will allow DEM input based on per unit density and use *MAT card to get consistent material properties.
- Added FORM = -4 for *ELEMENT_SHELL_NURBS_PATCH. Rotational dofs are automatically set at control points at the patch boundaries, whereas in the interior of the patch only translational dofs are present. This helps for joining multiple nurbs patches at their C0-boundaries.
- Disabled FORM = 2 and 3 for *ELEMENT_SHELL_NURBS_PATCH. These formulations are experimental and not fully validated yet.
- Added energy computation for isogeometric shells (*ELEMENT_SHELL_NURBS_PATCH) to matsum.
- Allow isogeometric shells (*ELEMENT_SHELL_NURBS_PATCH) to behave as rigid body (*MAT_RIGID).
- Added “g” as abbreviation for gigawords in specification of memory on execution line, e.g, memory = 16g is 16 billion words.
- Suppress non-printing characters in *COMMENT output.
- Add command line option “pgpkey” to output the current public PGP key used by LS-DYNA. The output goes to the screen as well as a file named “lstc_pgpkey.asc” suitable for directly importing into GPG.
- When reading the NAMES file, allow a “+” anywhere on a line to indicate there will be a following line, not just at the end. This was never intended, but worked before r73972 and some customers use it that way.

- Check for integer overflow when processing command line arguments and the memory value on the *KEYWORD card.
- Added new capability for *INTERFACE_LINKING_NODE to scale the displacements of the moving interface.
- Support for *KEYWORD_JOBID with internal *CASE driver.
- *DAMPING_FREQUENCY_RANGE now works for implicit dynamic solutions. An error check has been added to ensure that the timestep is small enough for the damping card to work correctly.
- Added new option *DAMPING_FREQUENCY_RANGE_DEFORM to damp only the deformation instead of the global motion.
- Added *DEFINE_VECTOR_NODES. A vector is defined using two node IDs.
- Add sense switch "prof" to output current timing profile to message (SMP) file or mes#### (MPP) files. Also, for MPP only, collect timing information from processor and output to prof.out when sense switch "prof" is detected.

Capabilities added during 2013/2014 to create LS-DYNA R7.1:

- Add MUTABLE option for *PARAMETER so that parameter values can be redefined later in the input deck.
- Change MPP treatment of two-sided *CONTACT_FORCE_TRANSDUCER so that proper mass and moment values can be output to the rcforc file.
- MPP support for non-zero birthtime for *CONTACT_SINGLE_EDGE.
- Add new command line option "ldir=" for setting a *local* working directory. In MPP, this has the same effect as setting the "directory { local }" pfile option (and it overrides that option). For SMP, it indicates a directory where local, working files should be placed.
- Add support for SMOOTH option in MPP groupable contact.
- Add new keyword card *CONTROL_REQUIRE_REVISION to prevent the model from being run in old versions of LS-DYNA.
- Add part set specification for dynamic relaxation with implicit using *CONTROL_DYNAMIC_RELAXATION. This is a new feature specified with idrflg = 6 on *CONTROL_DYNAMIC_RELAXATION. This allows implicit to be used for the dynamic relaxation phase for models involving parts being modeled with SPH and/or ALE while excluding those parts from the dynamic relaxation phase.
- Add new feature for implicit automatic time step control to cooperate with thermal time step control. On *CONTROL_IMPLICIT_AUTO, IAUTO = 2 is the same as IAUTO = 1 with the extension that the implicit mechanical time step is limited by the active thermal time step.

- On *CONTROL_IMPLICIT_SOLUTION, add negative value of MAXREF for implicit mechanics. Nonlinear iteration will terminate after |MAXREF| iterations. With MAXREF < 0 convergence is declared with a warning. Simulation will continue. Positive values of MAXREF still cause failure of convergence to be declared leading to either a time step reduction or an error termination.
- Add *CONTROL_IMPLICIT_MODAL_DYNAMIC keywords and features. This elevates the modal dynamic features of IMASS = 2 on *CONTROL_IMPLICIT_DYNAMICS. It also adds additional features of damping and mode selection and stress computations.
- New material model *MAT_DRY_FABRIC / MAT_214, which can be used in modeling high strength woven fabrics with transverse orthotropic behavior.
- Add *ALE_COUPLING_NODAL_PENALTY, penalty-based nodal coupling with ALE.
- Add type 8 *ELEMENT_SEATBELT_PRETENSIONER which takes energy-time curve, instead of pull-in or force curve.
- Add type 9 *ELEMENT_SEATBELT_PRETENSIONER for energy-based buckle / anchor pretensioner.
- Add *DATABASE_BINARY_FSILNK. This feature stores coupling pressure from *CONSTRAINED_LAGRANGE_IN_SOLID in a binary time history file for use in a separate model that does not include ALE.
- Add *LOAD_SEGMENT_FSILNK. Use pressure loads stored in aforementioned binary time history file to load model that does not have ALE elements.
- Add new keyword *DEFINE_SPH_DE_COUPLING to allow SPH particles to contact discrete element spheres (DES).
- Add MOISTURE option to *MAT_076 solids. Allows moisture content to be input as a function of time. Material parameters are then scaled according to the moisture and a moisture strain is also introduced.
- Add *RIGIDWALL_FORCE_TRANSDUCER to output forces from rigidwalls acting on node sets.
- Add LOG_INTERPOLATION option to *MAT_024. This offers an alternate means of invoking logarithmic interpolation for strain rate effects. The other way is to input the natural log of strain rate in the table LCSS.
- Add capability in *MAT_ADD_EROSION (NUMFIP < -100) to set stress to zero in each shell integration point as it reaches the failure criterion. When |NUMFIP|-100 integration points have failed, the shell is eroded. In contrast, when NUMFIP > 0, failed integration points continue to carry full load as though they were unfailed until element erosion occurs.

- Add new keyword, *PARAMETER_TYPE, for use by LS-PrePost when combining keyword input files. The appropriate offset is applied to each ID value defined using *PARAMETER_TYPE, according to how that ID is used.
- Allow use of load curve to specify damping as a function of time in *DAMPING_-RELATIVE.
- Add a segment based (SOFT = 2) contact option to include the overlap area in the contact stiffness calculation. This is good for improving the friction calculation and possibly for implicit convergence. The option is turned on by setting FNLSCL > 0 and DNLSCL = 0. As DNLSCL = 0, the contact stiffness is not non-linear. This new option is also useful when used with another improvement that was made to the FS = 2 friction coefficient by table lookup option in segment based contact. When the above mentioned FNLSCL > 0, option is used, the FS = 2 option is now very accurate.
- Add a new RCDC damage option, *MAT_PLASTICITY_WITH_DAMAGE_ORTHO_RCDC1980 which is consistent with the WILKINS paper. It uses the principal values of stress deviators and a different expression for the A_d term.
- Add a TIETYP option to *CONTACT_2D_AUTOMATIC. By default the tied contact automatically uses constraint equations when possible for 2D tied contact. If a conflict is detected with other constraints, or to avoid 2-way constraints, penalty type ties are used when constraints are not possible. The TIETYP option, when set to 1, causes all ties to use the penalty method. This is useful if in spite of the code's best efforts to avoid problems, there is still a conflict in the model.
- Add a scale factor for scaling the frictional stiffness for contact. The parameter is FRICSF on optional card E and it's only supported for segment based (SOFT = 2) contact. This was motivated by a rubber vs. road skidding problem where the friction coefficent had static, dynamic and decay parameters defined. The growth of the frictional force was too slow so the static coulomb value could not be achieved. By scaling the frictional stiffness higher, the coulomb value could approach the static value.
- Add keyword *CONTACT_2D_AUTOMATIC_FORCE_TRANSDUCER. Like the 3D force transducers, it does no contact calculation but only measures the contact forces from other contact definitions. When only a slave side is defined, the contact force on those segments is measured. Currently, two surface force transducers are not available.
- Add options to *MAT_058:
    - Load curves for rate dependent strain values (E11C, E11T, …) can be defined on new optional card 9.
    - Load curves for rate dependent strengths (XC, XT, …) and a rate averaging flag can be defined on new optional card 8.
    - Abscissa values in above curves are taken to be natural log of strain rate when the first value is negative.

  - Add optional transverse shear damage to *MAT_058.

- Add MAT_261 and MAT_262 for general use. *MAT_261 is *MAT_LAMINATED_FRACTURE_DAIMLER_PINHO. *MAT_262 is *MAT_LAMINATED_FRACTURE_DAIMLER_CAMANHO.
- Add pentahedra cohesive solid element types (TYPE = 21 & 22). Type = 21 is the pentahedra version of Type = 19 and Type = 22 is the pentahedra version of Type = 20. Using ESORT > 0 in *CONTROL_SOLID will automatically sort out the pentahedra elements (19 to 21 and 20 to 22).
- Add *DEFINE_DE_BY_PART to define control parameters for DES by part ID, including damping coefficient, friction coefficient, spring constant, etc. If defined, it will overwrite the parameters in *CONTROL_DISCRETE_ELEMENT.
- Add new feature for *MAT_030 (*MAT_SHAPE_MEMORY) as optional 3rd card. Curves or tables (strain rate dependency) can be defined to describe plastic loading and unloading behavior.
- New feature for *ELEMENT_BEAM_PULLEY. Beam elements BID1 and BID2 can now both be defined as “0” (zero). In that case, adjacent beam elements are automatically detected. Therefore, the first two beam elements with nodal distance < 10^{-6} to the pulley node (PNID) will be chosen.
- Add new feature to *MAT_ADD_EROSION's damage model GISSMO. By default, damage is driven by equivalent plastic strain. Now, users can optionally define another history variable as driving quantity by setting DMGTYP.
- Add volumetric plastic strain to *MAT_187 as history variable 6.
- Add internal energy calculation for *ELEMENT_BEAM_PULLEY.
- Add viscoplastic option to *MAT_157: new parameter VP on Card 5, Column 6.
- Add new keyword *MAT_ADD_COHESIVE which is intended to make 3D material models available for cohesive elements.
- Add new parameters to *MAT_CABLE_DISCRETE / *MAT_071. MXEPS (Card 2, Column 4) is equal the maximum strain at failure and MXFRC (Card 2, Column 5) is equal to the maximum force at failure
- Add *MAT_124 as potential weld partner material for PROPRUL = 2/3 of *DEFINE_CONNECTION_PROPERTIES.
- Add new material *MAT_TOUGHENED_ADHESIVE_POLYMER (TAPO) or *MAT_252 for epoxy-based, toughened, ductile adhesives.
- Add new option to *MAT_002_ANIS: parameter IHIS on Card 4, Column 8. IHIS = 0: terms C11, C12, … from Cards 1, 2, and 3 are used. IHIS = 1: terms C11, C12, … initialized by *INITIAL_STRESS_SOLID's extra history variables.
- Add new option to *MAT_102. Instead of constant activation energy Q, one can define a load curve LCQ on Card 2, Column 7:

  - LCQ.GT.0: Q as function of plastic strain
  - LCQ.LT.0: Q as function of temperature

- Add new option to *MAT_071 (MAT_CABLE_DISCRETE_BEAM). New parameter FRACL0 (Card 2, Column 3) is fraction of initial length that should be reached over time period of TRAMP. That means the cable element length gets modified from L0 to FRACL0 × L0 between t = 0 and t = TRAMP.
- Add internal energy calculation for SPR models *CONSTRAINED_INTERPOLATION_SPOTWELD (SPR3) and *CONSTRAINED_SPR2. Their contribution was missing in energy reports like glstat.
- Add new failure model OPT = 11 to *MAT_SPOTWELD/*MAT_100 for beam elements.
- Add three new failure criteria for shell elements to *MAT_ADD_EROSION on optional card 4, columns 6-8:
  - LCEPS12: load curve in-plane shear strain limit vs. element size.
  - LCEPS13: load curve cross-thickness shear strain limit vs. element size.
  - LCEPSMX: load curve in-plane major strain limit vs. element size.
- Add new capability to *MAT_ADD_EROSION damage model GISSMO. Strain rate scaling curve LCSRS can now contain natural logarithm values of strain rates as abscissa values. This is automatically assumed when the first value is negative.
- Add new parameter NHMOD to *MAT_266. The constitutive model for the isotropic part can now be chosen:
  - NHMOD = 0: original implementation (modified Neo-Hooke)
  - NHMOD = 1: standard Neo-Hookeon (as in umat45)
- New keyword *DEFINE_TABLE_MATRIX is an alternative way of defining a table and the curves that the table references from a single unformatted text file, e.g., as saved from an Excel spreadsheet.
- Change long format so that all data fields are 20 columns and each line of input can hold up to 200 columns. In this way, the number of input lines is the same for long format as for standard format.
  - 8 variables per line in long format = 160 columns
  - 10 variables per line in long format = 200 columns
- Add a new option (SOFT = 6) in *CONTACT_FORMING_NODES_TO_SURFACE for blank edge and guide pin contact.
- Add user-defined criteria for mesh refinement (or coarsening) in *CONTROL_REFINE_….

- Add new contact option that currently only works for MPP SINGLE_SURFACE contact with SOFT = 0 or 1. If SRNDE (field 4 of optional card E) is a 1, then free edges of the contact definition will be rounded WITHOUT extending the segments. Rather than having cylindrical caps on the ends of the segments, the "corners" of the squared off thickness are rounded over.
- Add geometric contact entity type -3 "finite cylinder".
- Add irate = 2 to *CONTROL_IMPLICIT_DYNAMICS to turn off rate effects for both implicit and explicit.
- Add quadratic 8-node and 6-node shells (shell formulations 23 and 24).
- Add LOG_LOG_INTERPOLATION option for table defining strain rate effects in *MAT_083, *MAT_181, and *MAT_183.
- Add automatic generation of null shells for quadratic shell contact (*PART_DUPLICATE_NULL_OVERLAY).
- Add beam contact forces to rcforc output (*DATABASE_RCFORC).
- Add SHL4_TO_SHL8 option to *ELEMENT_SHELL to automatically convert 4-node shells to 8-node quadratic shells.
- Add 3-node beam element with quadratic interpolation that is tailored for the piping industry. It includes 12 degrees of freedom, including 6 ovalization degrees of freedom, per node for a total of 36 DOF. An internal pressure can be given that can stiffen and elongate the pipe.
    - ELFORM = 14 in *SECTION_BEAM.
    - *ELEMENT_BEAM_ELBOW.
    - NEIPB in *DATABASE_EXTENT_BINARY to direct output of elbow loop-stresses to d3plot. Otherwise, output goes to ASCII file elbwls.k.
    - Supported by a subset of material models including mats 3, 4, 6, 153, 195.
- Add discrete element option DE to *DATABASE_TRACER.
    - Includes variable RADIUS. average result of all
        - RADIUS > 0: Reports the average result of all DE particles in a spherical volume having radius equal to RADIUS and centered at the tracer.
        - RADIUS < 0: Reports result of the closest particle to the tracer.
    - If a tracer node NID is given, then the tracer moves with this node. The node must belong to a DES.
- Add new options *PART_COMPOSITE_LONG and *ELEMENT_SHELL_COMPOSITE_LONG. In contrast to "COMPOSITE", one integration point is defined per card. This is done to allow for more information, such as "ply id".

- Add support of *MAT_ADD_EROSION option NUMFIP < 0 for standard (non-GISSMO) failure criteria. Only for shells.
- Add viscoplastic behavior to *MAT_157, i.e., parameter LCSS can now refer to a table with strain rate dependent yield curves.
- Add singular finite element with midside nodes for 2D plane strain fracture analysis (ELFORM = 55 in *SECTION_SHELL). This is an 8-noded element and can induce a singular displacement field by moving mid-side nodes to quarter locations.
- If HCONV < 0 in *AIRBAG_PARTICLE, |HCONV| is a curve of heat convection coefficient vs. time.
- Add new option DECOMPOSITION for *AIRBAG_PARTICLE -- MPP only.This will automatically invoke the recommended decomposition commands, *CONTROL_MPP_DECOMPOSITION_BAGREF (if applicable) and *CONTROL_-MPP_DECOMPOSITION_ARRANGE_PARTS, for the bag.
- Add new blockage option for vents in *AIRBAG_PARTICLE:
    - blockage considered
        - .eq.0: no
        - .eq.1: yes
        - .eq.2: yes, exclude external vents
        - .eq.3: yes, exclude internal vents
        - .eq.4: yes, exclude all vents
- Add option in *CONTROL_CPM to consider CPM in the time step size calculation.
- When using *AIRBAG_PARTICLE with IAIR = 2, user should keep mole / particle similar between inflator gas and initial air particles to ensure the correct elastic collision. If different by more than 10%, code will issue warning message and provide the suggested initial air particle number.
- Enable *DEFINE_CURVE_FUNCTION for *SECTION_POINT_SOURCE_MIXTURE and *SECTION_POINT_SOURCE.
- Make *BOUNDARY_PRESCRIBED_MOTION_SET compatible with *CONTROL_REFINE
- Change *BOUNDARY_ACOUSTIC_COUPLING_MISMATCH to rank order opposing acoustic faces and structural segments by proximity, thereby accelerating the preprocessing stage, enhancing reliability and allowing some liberalization of the search parameters.
- Implement hemispherical geometry for particle blast (*DEFINE_PBLAST_GEOMETRY).
- Add explosive type for *PARTICLE_BLAST.

- For particle-based blast *PARTICLE_BLAST:
    - Include random distribution of initial air molecules
    - Modify algorithm to account for the non-thermally-equilibrated state of high velocity gas.

- Improve particle contact method for particle-based blast loading *PARTICLE_-BLAST.
- *CONTACT now works for parts refined using *CONTROL_REFINE_SOLID or *CONTROL_REFINE_SHELL.
- Improve calculation of shell element contact segment thicknesses, particularly at material boundaries.
- MPP: Add output to rcforc file for *CONTACT_AUTOMATIC_TIEBREAK to record the # of nodes tied, and the total tied area.
- MPP: Add calculation of "contact gap" for master side of FORMING contact.
- MPP: Add support for table-based friction (FS = 2.0) to groupable contact.
- Implement splitting-pinball contact, Belytschko & Yeh (1992, 1993). This new contact option is invoked by setting SOFT = 2, SBOPT = 3 and DEPTH = 45. A penetration check method based on LS-PrePost version 4.0 is implemented for the new bilinear-patch-based contact, SOFT = 2, DEPTH = 45 & Q2TRI = 0. The new method provides more accurate intersection information when Q2TRI = 0.
- Add support for birth time for *CONTACT_2D_AUTOMATIC_TIED.
- Improve the segment based single surface contact search for thick segment pairs that are too close together. The code was not working well with triangluar segments. This change affects models with shell segments that have thickness greater than about 2/3 of the segment length.
- Enable segment based quad splitting options to work when shell sets or segment sets are used to define the surface that will be split. This is really a bug fix because there was no check to prevent this and the result was writing past the allocated memory for segment connectivites.
- Allow *CONSTRAINED_INTERPOLATION to use node set to define the independent nodes.
- Add a length unit to the tolerance used for the checking of noncoincident nodes in *CONSTRAINED_JOINTs excluding spherical joints. The old tolerance was 1.e-3. The new tolerance is 1.e-4 times the distance between nodes 1 and 3. The error messages were changed to warnings since this change might otherwise cause existing models to stop running.
- Add d3hsp output for *CONSTRAINED_INTERPOLATION_SPOTWELD (SPR3) and *CONSTRAINED_SPR2. Can be deactivated by setting NPOPT = 1 on *CONTROL_OUTPUT.

- Support NFAIL1 and NFAIL4 of *CONTROL_SHELL in coupled thermal-mechanical analysis, i.e. erode distorted elements instead of error termination.
- PTSCL on *CONTROL_CONTACT can be used to scale contact force exerted on shell formulations 25, 26, 27 as well as shell formulations 2, 16 (IDOF = 3).
- Use SEGANG in *CONTROL_REMESHING to define positive critical angle (unit is radian) to preserve feature lines in 3D tetrahedral remeshing (ADPOPT = 2 in *PART).
- For 3D solid adaptive remeshing including ADPOPT = 2 and ADPOPT = 3 (*PART), the old mesh will be used automatically if the remesher fails generating a new mesh.
- Add option INTPERR on *CONTROL_SHELL (Optional Card 3, Column 8). By default, warning messages INI+143/144/145 are written in case of non-matching number of integration points between *INITIAL_STRESS_SHELL and *SECTION_SHELL. Now with INTPERR = 1, LS-DYNA can terminate with an error.
- Add variable D3TRACE on *CONTROL_ADAPTIVE: The user can now force a plot state to d3plot just before and just after an adaptive step. This option is necessary for tracing particles across adaptive steps using LS-PrePost.
- By putting MINFO = 1 on *CONTROL_OUTPUT, penetration info is written to message files for mortar contact., see also *CONTACT_…. Good for debugging implicit models, not available for explicit.
- Change the default scale factor for binary file sizes back to 70. This value can be changed using “x=” on the execution line. In version R7.0, the default value of x is 4096, and that sometimes leads to difficulty in postprocessing owing to the large size of the d3plot file(s).
- Enable *CONTROL_OUTPUT flag, EOCS, which wasn't having any effect on the shells output to elout file.
- *DATABASE_FSI_SENSOR: Create sensors at solid faces in 3D and at shell sides in 2D.
- *DATABASE_PROFILE: Implement the option DIR = 4 to plot data with curvilinear distributions and the flag UPDLOC to update the profile positions.
- In *CONTROL_SHELL, add options for deletion of shells based on:
  - diagonal stretch ratio (STRETCH)
  - w-mode amplitude in degrees (W-MODE)
- New element formulation ELFORM = 45 in *SECTION_SOLID: Tied Meshfree-enriched FEM (MEFEM). This element is based on the 4-noded MEFEM element (ELFORM = 43, *SECTION_SOLID). Combined with *CONSTRAINED_TIED_-NODES_FAILURE, *SET_NODE_LIST and cohesive model, this element can be used to model dynamic multiple-crack propagation along the element boundaries.

- New high order tetrahedron CPE3D10 based on Cosserat Point theory can be invoked by specifying element formulation ELFORM = 16 and combining this with hourglass formulation IHQ = 10. See *SECTION_SOLID and *HOURGLASS.
- Add database D3ACS for collocation acoustic BEM (*FREQUENCY_DOMAIN_ACOUSTIC_BEM) to show the surface pressure and normal velocities.
- Implement biased spacing for output frequencies for random vibration (*FREQUENCY_DOMAIN_RANDOM_VIBRATION).
- Add frequency domain nodal or element velocity output for acoustic BEM (*FREQUENCY_DOMAIN_ACOUSTIC_BEM).
- Implement boundary acoustic mapping to acoustic BEM in MPP (*BOUNDARY_ACOUSTIC_MAPPING). This is enabled only for segment sets at present.
- Implement panel contribution analysis capability to Rayleigh method (*FREQUENCY_DOMAIN_ACOUSTIC_BEM_PANEL_CONTRIBUTION).
- Implement a scheme to map velocity boundary condition from dense BEM mesh to coarse mesh to speed up the computation (*FREQUENCY_DOMAIN_ACOUSTIC_BEM).
- Add user node ID for acoustic field points in D3ATV (*FREQUENCY_DOMAIN_ACOUSTIC_BEM). Now D3ATV is given for multiple field points, and multiple frequencies.
- Add database D3ATV for acoustic transfer vector binary plot (*FREQUENCY_DOMAIN_ACOUSTIC_BEM_ATV, *DATABASE_FREQUENCY_BINARY_D3ATV).
- Implement acoustic panel contribution analysis to collocation BEM and dual collocation BEM (*FREQUENCY_DOMAIN_ACOUSTIC_BEM).
- Enable *FREQUENCY_DOMAIN_MODE in response spectrum analysis (*FREQUENCY_DOMAIN_RESPONSE_SPECTRUM).
- Implement an option to read in user-specified nodal velocity history data for running BEM acoustics (*FREQUENCY_DOMAIN_ACOUSTIC_BEM).
- Extend Kirchhoff acoustic method to MPP (*FREQUENCY_DOMAIN_ACOUSTIC_BEM).
- Extend response spectrum analysis to multiple load spectra cases (*FREQUENCY_DOMAIN_RESPONSE_SPECTRUM).
- Add BAGVENTPOP for *SENSOR_CONTROL. This allows user more flexibilty controlling the pop-up of the venting hole of *AIRBAG_HYBRID and *AIRBAG_WANG_NEFSKE
- Add command *SENSOR_DEFINE_FUNCTION. Up to 15 *DEFINE_SENSORs can be referenced in defining a mathematical operation.

- LAYER of *SENSOR_DEFINE_ELEMENT can now be an integer “I” representing the Ith integration point at which the stress/strain of the shell or tshell element will be monitored.
- Add control of *LOAD_MOVING_PRESSURE by using *SENSOR_CONTROL.
- Add thick shells to the ETYPE option list of *SENSOR_DEFINE_ELEMENT.
- Add *CONTROL_MPP_MATERIAL_MODEL_DRIVER in order to enable the Material Model Driver for MPP (1 core).
- Add table input of thermal expansion coefficient for *MAT_270. Supports temperature-dependent curves arranged according to maximum temperature.
- Add table input of heat capacity for *MAT_T07. Supports temperature dependent curves arranged according to maximum temperature.
- Add two more kinematic hardening terms for *MAT_DAMAGE_3/MAT_153, c2 & gamma2.
- Add materials *MAT_CONCRETE_DAMAGE_REL3/*MAT_072R3 and *MAT_-CSCM_CONCRETE/*MAT_159 to Interactive Material Model Driver.
- Enable *MAT_JOHNSON_COOK/*MAT_015 for shell elements to work with coupled structural / thermal analysis.
- Allow *MAT_SOIL_AND_FOAM/*MAT_005 to use positive or negative abscissa values forload curve input of volumetric strains.
- Add *MAT_ACOUSTIC elform = 8 support for pyramid element case using 5-pt integration.
- Add support to *MAT_219 (*MAT_CODAM2) for negative AOPT values which point to coordinate system ID's.
- Modify *MAT_224 so it uses the temperatures from the thermal solution for a coupled thermal-mechanical problem.
- Add alternative solution method (Brent) for *MAT_015 and *MAT_157 in case standard iteration fails to converge.
- Add shell element IDs as additional output to messag file for *MAT_036's warning “plasticity algorithm did not converge”.
- For *MAT_USER_DEFINED_MATERIAL_MODELS, the subroutines crvval and tabval can be called with negative curve / table id which will extract values from the user input version of the curve or table instead of the internally converted “100-point” curve / table.
- In the damage initiation and evolution criteria of *MAT_ADD_EROSION (invoked by IDAM < 0), add the option Q1 < 0 for DETYP = 0. Here, |Q1| is the table ID defining the ufp (plastic displacement at failure) as a function of triaxiality and damage value, i.e., ufp = ufp(eta, D), as opposed to being constant which is the default.

- In *MAT_RHT, ONEMPA = -6 generates parameters in g, cm, and μS and ONEMPA = -7 generates parameters in g, mm, and mS
- In *MAT_SIMPLIFIED_RUBBER/FOAM, STOL > 0 invokes a stability analysis and warning messages are issued if an unstable stretch point is found within a logarithmic strain level of 100%.
- Implement *DATABASE_ALE to write time history data (volume fractions, stresses, …) for a set of ALE elements. Not to be confused with *DATABASE_-ALE_MAT.
- Implement *DELETE_PART in small restarts for ALE2D parts.
- Add conversion of frictional contact energy into heat when doing a coupled thermal-mechanical problem for SPH (variable FRCENG in *CONTROL_CONTACT). This option applys to all 3D contact types supported by SPH particles.
- For keyword *DEFINE_ADAPTIVE_SOLID_TO_SPH, add support of explicit SPH thermal solver for the newly generated SPH particles which were converted from solid elements. The temperatures of those newly generated SPH particles are mapped from corresponding solid elements.
- Implement DE to surface tied contact *DEFINE_DE_TO_SURFACE_TIED. The implementation includes bending and torsion.
- Implement keyword *DEFINE_DE_HBOND to define heterogeneous bond for discrete element spheres (DES). DES (*ELEMENT_DISCRETE_SPHERE) with different material models can be bonded.
- Implement keyword *INTERFACE_DE_BOND to define multiple failure models for various bonds within one part or between different parts through the keyword *DEFINE_DE_HBOND.
- Implement *DEFINE_DE_TO_BEAM_COUPLING for coupling of discrete element spheres to beam elements.
- Add variable MAXGAP in *DEFINE_DE_BOND to give user control of distance used in judging whether to bond two DES together or not, based on their initial separation.
- Add IAT = -3 in *CONTROL_REMESHING_EFG, which uses FEM remapping scheme in EFG adaptivity. Compared to IAT = -2, -1, 1, 2, IAT = -3 is faster and more robust but less accurate.
- Add control flag MM in *CONTROL_REMESHING_EFG to turn on/off monotonic mesh resizing for EFG 3D general remeshing (ADPOPT = 2 in *PART).
- *CONTROL_IMPLICIT_BUCKLING - Extend Implicit Buckling Feature to allow for Implicit problems using Inertia Relief. This involves adding the Power Method as a solution technology for buckling eigenvalue problems. Using the power method as an option for buckling problems that are not using inertia relief has been added as well.

- Extend Implicit Buckling to allow for Intermittent extraction by using negative values of NMODE on *CONTROL_IMPLICIT_BUCKLING similar to using negative values of NEIG on *CONTROL_IMPLICIT_EIGENVALUE.
- Extend implicit-explicit switching specified on *CONTROL_DYNAMIC_RELAXATION to allow explicit simulation for the dynamic relaxation phase and implicit for the transient phase.
- New implementation for extracting resultant forces due to joints for implicit mechanics.
- New implementation of extracting resultant forces due to prescribed motion for implicit mechanics.
- Add support for IGAP > 2 in implicit, segment based (SOFT = 2) contact.
- Add constraint-based, thermal nodal coupling for *CONSTRAINED_LAGRANGE_IN_SOLID. HMIN < 0 turns it on.
- Add FRCENG = 2 on CONTROL_CONTACT keyword.
    - if FRCENG = 1, convert contact frictional energy to heat.
    - if FRCENG = 2, do not convert contact frictional energy to heat.
- Add effect of thermal time scaling (TSF in *CONTROL_THERMAL_SOLVER) to 2D contact.
- Add new pfile decomposition region option: partsets. Takes a list of part sets (*SET_PART) from the keyword input and uses them to define a region, e.g., region { partsets 102 215 sy 1000 } This example would take partsets, scale y by 1000, and decompose them and distribute them to all processors.
- Reduce MPP memory usage on clusters.
- Add MPP support for *ELEMENT_SOURCE_SINK.
- Add new pfile options:
    - decomp { d2r_as_rigid }
    - decomp { d2ra_as_rigid }

  which cause materials appearing in "*DEFORMABLE_TO_RIGID" and "*DEFORMABLE_TO_RIGID_AUTOMATIC" to have their computational costs set as if they were rigid materials during the decomposition.
- Add option ISRCOUT to *INCLUDE_STAMPED_PART to dump out the transformed source/stamp mesh.
- *CONTROL_FORMING_OUTPUT: Allow NTIMES to be zero; support birth and death time; support scale factor in curve definition.
- Add a new option (INTFOR) to *CONTROL_FORMING_OUTPUT to control the output frequency of the INTFOR database.

- Add new features (instant and progressive lancing) in *ELEMENT_LANCING for sheet metal lancing simulation.
- Add a new keyword: *CONTROL_FORMING_INITIAL_THICKNESS.
- Add a new option for springback compensation: *INCLUDE_COMPENSATION_ORIGINAL_TOOLS.
- Add a new keyword: *INTERFACE_COMPENSATION_NEW_PART_CHANGE.
- Add a new keyword (*DEFINE_CURVE_BOX_ADAPTIVITY) to provide better control of mesh refinement along two sides of the curve.
- Isogeometric analysis: contact is available in MPP.
- Normalize tangent vectors for local coordinate system for the rotation free isogeometric shells.
- Add support for dumping shell internal energy density for isogeometric shells (*ELEMENT_SHELL_NURBS_PATCH) via interpolation shells.
- Add support for dumping of strain tensor (STRFLG.EQ.1) for isogeometric shells (*ELEMENT_SHELL_NURBS_PATCH) via interpolation shells.
- Add H-field, magnetization and relative permeability to d3plot output.
- *ICFD_INITIAL: Add a reference pressure (pressurization pressure) for when no pressure is imposed on the boundaries.
- Add the initialization of all nodes at once by setting PID = 0.
- Add the non-inertial reference frame implementation defined by the keyword *ICFD_DEFINE_NONINERTIAL.
- Add several new state variables to LSO. Please refer to the LSO manual to see how to print out the list of supported variables.
- Add support for FSI with thick shells.
- 2D shells are now supported for FSI in MPP. In the past only beams could be used in MPP and beams and shells could be used in SMP.
- The keyword ICFD_CONTROL_FSI has a new field to control the sensitivity of the algorithm to find the solid boundaries used in FSI calculations.
- The 2D mesh now generates semi-structured meshes near the boundaries.
- Add heat flux boundary condition using ICFD_BOUNDARY_FLUX_TEMP.
- Add divergence-free and Space Correlated Synthetic Turbulence Inlet Boundary Condition for LES (Smirnov et al.) using *ICFD_BOUNDARY_PRESCRIBED_-VEL.
- *ICFD_BOUNDARY_PRESCRIBED_VEL: Add inflow velocities using the wall normal and a velocity magnitude using the 3rd field VAD.
- Add the activation of synthetic turbulence using the 3rd field VAD.

- Add the option to control the re-meshing frequency in both keywords: see *ICFD_-CONTROL_ADAPT_SIZE and *ICFD_CONTROL_ADAPT.
- *ICFD_CONTROL_TURB_SYNTHESIS: control parameters for the synthetic turbulence inflow.
- *ICFD_BOUNDARY_PRESCRIBED_MOVEMESH: Allows the mesh to slide on the boundaries following the cartesian axis.
- Add a PART_SET option for *CESE_BOUNDARY_…_PART cards.
- Bring in more 2D mesh support, both from the PFEM mesher and a user input 2D mesh (via *ELEMENT_SOLID with 0 for the last 4 of 8 nodes).
- Enable the 2D ball-vertex mesh motion solver for the 2D CESE solver.
- Add new input cards:
  - *CESE_BOUNDARY_CYCLIC_SET
  - *CESE_BOUNDARY_CYCLIC_PART
- Add code for 2D CESE sliding boundary conditions.
- Add support in CESE FSI for 2D shells in MPP.
- Add support for CESE FSI with thick shells.
- Add 2D & 2D-axisymmetric cases in the CESE-FSI solver (including both immersed boundary method & moving mesh method) .
- Add the CSP reduced chemistry model with 0D, 2D, and 3D combustion. The 2D and 3D combustion cases couple with the CESE compressible flow solver.
- Add the G-scheme reduced chemistry model only for 0D combustion.
- Add two different reduced chemistry models.
  - The Computational Singular Perturbation (CSP) reduced model is implemented with existing compressible CESE solver. The CSP is now working on 0-dimensional onstant volume and pressure combustion, 2-D, and 3-D combustion problems.
  - The new reduced chemistry model, G-scheme, is implemented, but currently works only 0-dimensional problems such as constant combustors.
- Jobid can now be changed in a restart by including "jobid=" on the restart execution line. Previously, the jobid stored in d3dump could not be overwritten.
- Part labels (PID) can be up to 8 characters in standard format; 20 characters in long format.
- Labels for sections (SID), materials (MID), equations of state (EOSID), hourglass IDs (HGID), and thermal materials (TMID) can be up to 10 characters in standard format; 20 characters in long format.

- Create bg_switch and kill_by_pid for SMP. Both files will be removed at the termination of the run.
- Increase the overall length of command line to 1000 characters and length of each command line option to 50 characters.
- Increase MPP search distance for tied contacts to include slave and master thicknesses.
- For *CONTACT_AUTOMATIC_…_MORTAR, the mortar contact now supports contact with the lateral surface of beam elements.
- On *CONTACT_..._MORTAR, IGAP.GT.1 stiffens the mortar contact for large penetrations. The mortar contact has a maximum penetration depth DMAX that depends on geometry and input parameters; if penetration is larger than this value the contact is released. To prevent this release, which is unwanted, the user may put IGAP.GT.1 which stiffens the behavior for penetrations larger than 0.5*DMAX without changing the behavior for small penetrations. This should hopefully not be as detrimental to convergence as increasing the overall contact stiffness.
- For initialization by prescribed geometry in dynamic relaxation (IDRFLG = 2, *CONTROL_DYNAMIC_RELAXATION), add an option where displacements are not imposed linearly but rather according to a polar coordinate system. This option was added to accommodate large rotations.
- The flag RBSMS on *CONTROL_RIGID is now active for regular and selective mass scaling to consistently treat interfaces between rigid and deformable bodies
- Remove static linking for l2a as many systems do not have the required static libraries.
- Add IELOUT in *CONTROL_REFINE to handle how child element data is handled in elout (*DATABASE_HISTORY_SOLID and *DATABASE_HISTORY_-SHELL). Child element data are stored if IELOUT = 1 or if refinement is set to occur only during initialization.
- Include eroded hourglass energy in hourglass energy in glstat file to be consistent with KE & IE calculations so that the total energy = kinetic energy + internal energy + hourglass energy + rigidwall energy.
- Remove *DATABASE_BINARY_XTFILE since it is obsolete.
- When using *PART_AVERAGED for truss elements (beam formulation 3), calculate the time step based on the total length of the combined macro-element instead of the individual lengths of each element.
- Enable writing of midside nodes to d3plot or 6- and 8-node quadratic shell elements.
- Write complete history variables to dynain file for 2D solids using *MAT_NULL and equation-of-state.

- Shell formulations 25, 26, and 27 are now fully supported in writing to dynain file (*INTERFACE_SPRINGBACK_LSDYNA).
- Shell formulations 23 (quad) and 24 (triangle) can now be mixed in a single part. When ESORT = 1 in *CONTROL_SHELL, triangular shells assigned by *SECTION_SHELL to be type 23 will automatically be changed to type 24.
- Enable hyperelastic materials (those that use Green's strain) to be used with thick shell form 5. Previously, use of these materials (2, 7, 21, 23, 27, 30, 31, 38, 40, 112, 128, 168, and 189) with thick shell 5 has been an input error.
- Update acoustic BEM to allow using *DEFINE_CURVE to define the output frequencies (*FREQUENCY_DOMAIN_ACOUSTIC_BEM).
- When using *CONTROL_SPOTWELD_BEAM, convert *DATABASE_HISTORY_BEAM to *DATABASE_CROSS_SECTION and *INITIAL_AXIAL_FORCE_BEAM to *INITIAL_STRESS_CROSS_SECTION for the spot weld beams that are converted to hex spot welds.
- Improve output of *INITIAL_STRESS_BEAM data to dynain via *INTERFACE_SPRINGBACK_LSDYNA. Now, large format can be chosen, history variables are written, and local axes vectors are included.
- Update *MAT_214 (*MAT_DRY_FABRIC) to allow fibers to rotate independently.
- Enable regularization curve LCREGD of *MAT_ADD_EROSION to be used with FLD criterion, i.e. load curve LCFLD. Ordinate values (major strain) will be scaled with the regularization factor.
- Modify *MAT_ADD_EROSION parameter EPSTHIN:
  - EPSTHIN > 0: individual thinning for each IP from z-strain (as before).
  - EPSTHIN < 0: averaged thinning strain from element thickness (new).
- Enable regularization curve LCREGD of *MAT_ADD_EROSION to be used with standard (non-GISSMO) failure criteria. Users can now define a failure criterion plus IDAM = 0 plus LCREGD = scaling factor vs. element size to get a regularized failure criterion.
- *MAT_ADD_EROSION: equivalent von Mises stress SIGVM can now be a function of strain rate by specifying a negative load curve ID.
- *SECTION_ALE1D and *SECTION_ALE2D now work on multiple processors (SMP and MPP).
- *CONSTRAINED_LAGRANGE_IN_SOLD ctype 4/5 now converts friction energy to heat. Note it only works for ALE elform 12.

Capabilities added September 2013 – January 2015 to create LS-DYNA R8.0:

See release notes (published separately) for further details.

- Add RDT option for *AIRBAG_SHELL_REFERENCE_GEOMETRY.
- LCIDM and LCIDT of *AIRBAG_HYDRID can now be defined through *DEFINE_CURVE_FUNCTION.
- New variable RGBRTH in *MAT_FABRIC to input part-dependent activation time for airbag reference geometry.
- Negative PID of *AIRBAG_INTERACTION considers the blockage of partition area due to contact.
- Enhancements to *AIRBAG_PARTICLE:
  - New blockage (IBLOCK) option for vents.
  - External work done by inflator gas to the structure is reported to glstat.
  - Enhance segment orientation checking of CPM bag and chambers.
  - Allow user to excluded some parts surface for initial air particles.
  - Support compressing seal vent which acts like flap vent.
  - Support Anagonye and Wang porosity equation through *MAT_FABRIC.
  - Add keyword option _MOLEFRACTION.
- Add ID keyword option to *AIRBAG_REFERENCE_GEOMETRY and *AIRBAG_SHELL_REFERENCE which includes optional input of variables for scaling the reference geometry.
- Enable *DEFINE_CURVE_FUNCTION for *AIRBAG_SIMPLE_AIRBAG_MODEL.
- Calculate heat convection (HCONV) between environment and airbag in consistent fashion when TSW is used to switch from a particle airbag to a control volume.
- For *AIRBAG_PARTICLE, add ENH_V = 2 option for vent hole such that two-way flow can occur, i.e., flow with or against the pressure gradient.
- *BOUNDARY_ALE_MAPPING: add the following mappings: 1D to 2D, 2D to 2D, 3D to 3D.
- *SET_POROUS_ALE: new keyword to define the properties of an ALE porous media by an element set. The porous forces are computed by *LOAD_BODY_POROUS.
- *ALE_FSI_SWITCH_MMG: applies also now to 2D.
- *ALE_SWITCH_MMG: new keyword to switch multi-material groups based on criteria defined by the user with *DEFINE_FUNCTION.
- *CONTROL_ALE: Allow PREF (reference pressure) to be defined by materials.
- Implement *ALE_COUPLING_NODAL_DRAG to model the drag force coupling between discrete element spheres or SPH particles and ALE fluids.

- Implement *ALE_COUPLING_RIGID_BODY as an efficient alternative for constraint type coupling between ALE fluids and a Lagrangian rigid body.
- Error terminate if *BOUNDARY_SPC_NODE_BIRTH_DEATH is applied to a node that belongs to a rigid body.
- Modify *BOUNDARY_PRESCRIBED_ORIENTATION_VECTOR to accommodate bodies which undergo no changes in orientation.
- Add a new keyword *BOUNDARY_SPC_SYMMETRY_PLANE.
- Solid part or solid part set is now allowed for *PARTICLE_BLAST.
- Add ambient pressure boundary condition flag BC_P for *PARTICLE_BLAST.
- New command *DEFINE_PBLAST_GEOMETRY allows the high explosive domain for*PARTICLE_BLAST to be defined by various geometric shapes.
- Allow multiple *PARTICLE_BLAST definitions.
- Add *DATABASE_PBSTAT to output particle blast statistics.
- Output the initial volume and initial mass of HE particles and air particles for *PARTICLE_BLAST to d3hsp.
- Add the command *CESE_BOUNDARY_BLAST_LOAD to allow a blast described by the *LOAD_BLAST_ENHANCED command to be used as a boundary condition in CESE.
- Modify the FSI interface reflective boundary condition pressure treatment in some calculations for the moving mesh and immersed boundary solvers.
- Change the CESE derivatives calculation method to use the current values of flow variables.
- Add two new MAT commands for CESE solver, *CESE_MAT_000 and *CESE_-MAT_002.
- Add a non-inertial reference frame solver for fluid and FSI problems using the moving-mesh method.
- For the moving mesh CESE solver, replace the all-to-all communication for conjugate heat and FSI quantities with a sparse communication mechanism.
- Add structural element erosion capability to the immersed boundary method CESE FSI solver (serial capability only).
- Add 2D cyclic boundary conditions capability.
- Add a NaN detection capability for the CESE solver.
- Switch all CESE boundary conditions that use a mesh surface part to define the boundary to use the character string "MSURF" instead of "PART" in the option portion of the keyword name.

- Add missing temperature interpolation in time for imposing solid temperatures as a boundary condition in the CESE solver.
- Optimize the IDW-based mesh motion for the CESE moving mesh solver.
- Treat the input mesh as 3D by default for the CESE solver.
- All of the chemistry features mentioned below are coupled only to the CESE compressible flow solver when 2D or 3D calculations are involved.
- Chemical source Jacobians have been added.
- Introduce *CHEMISTRY_CONTROL_PYROTECHNIC and *CHEMISTRY_PROPELLANT_PROPERTIES for airbag applications. In conjunction with these commands, basic airbag inflator models are implemented.
- The pyrotechnic inflator model using NaN3/Fe2O3 propellant is newly implemented. To connect with the existing ALE airbag solver, two load curves, mass flow rate and temperature, are saved in "inflator_outfile" as a function of time. This model computes three sub-regions: combustion chamber, gas plenum, and discharge tank. Each region can be initialized with different *CHEMISTRY_COMPOSITION models, which means that user can compute Propellant+Gas hybrid mode.
- The following 0-dimensional combustion problems have been improved: constant volume, constant pressure, and CSP.
- For iso-combustion. temperature and species mass fractions as a function of time are displayed on screen and saved in "isocom.csv" to plot with LS-PrePost.
- Another chemical ODE integration method has been implemented.
- The output file of the pyrotechnic inflator is updated so that this file can be read from ALE solver for an airbag simulation.
- 2-D and 3-D TNT gaseous blast explosives, categorized as TBX (thermobaric explosives), are implemented for the Euler equation systems (CESE-only). Also, 3-D TNT blast + aluminum combustion for serial problems is now implemented.
- Implement a mix modeling method for use with CESE solvers.
- Modify *CHEMISTRY-related keyword commands to allow multiple chemistry models in the same problem.
- Add command *CHEMISTRY_MODEL which identifies the files that define a Chemkin chemistry model.
- Modify the following commands such that the files related to the chemistry model have been removed. These commands are only used to select the type of chemistry solver:
    - *CHEMISTRY_CONTROL_CSP
    - *CHEMISTRY_CONTROL_FULL
    - *CHEMISTRY_CONTROL_1D

- Modify *CHEMISTRY_DET_INITIATION where the files related to the chemistry model have been removed, and the Model ID used is inferred through a reference to a chemistry composition ID.
- Modify *CHEMISTRY_COMPOSITION and *CESE_CHEMISTRY_D3PLOT to add model ID.
- Add *CONTACT_TIED_SHELL_EDGE_TO_SOLID for transferring moments from shells into solids.
- Add frictional energy calculation for beams in *CONTACT_AUTOMATIC_GENERAL.
- Enhance ERODING contacts for MPP. The new algorithm uses a completely different approach to determining the contact surface. The old algorithm started from scratch when identifying the exterior of the parts in contact. The new algorithm is smarter about knowing what has been exposed based on what is eroded, and is faster.
- Force EROSOP = 1 for all ERODING type contacts, with a warning to the user if they had input it as 0.
- Add error check in case of a contact definition with an empty node set being given for the slave side.
- Modify output of ncforc (*DATABASE_NCFORC) in order to support output in a local coordinate system.
- For ERODING contacts, reduce memory allocated for segments so each interior segment is only allocated once.
- Add keyword *DEFINE_CONTACT_EXCLUSION (MPP only) to allow for nodes tied in some contacts to be ignored in certain other contacts.
- Rewrite meshing of *CONTACT_ENTITY to use dynamic memory, which removes the previous limit of 100 meshed contact entities. There is now no limit.
- Remove undocumented release condition for MPP's *CONTACT_AUTOMATIC_TIEBREAK, options 5 and greater.
- Add new experimental "square edge" option to select SOFT = 0,1 contacts. This new option applies only to AUTOMATIC_SINGLE_SURFACE and the segment-to-segment treatment of AUTOMATIC_GENERAL, and is invoked by setting SRNDE = 2 on *CONTACT's Optional Card E. This new option does not apply to SOFT = 2; SOFT = 2 square edge option is set using SHLEDG in *CONTROL_CONTACT.
- BT and DT in *CONTACT can be set to define more than one pair of birthtime/death-time for the contact by pointing to a curve or table. These pairs can be unique for the dynamic relaxation phase and the normal phase of the simulation.

- Add EDGEONLY option to *CONTACT_AUTOMATIC_GENERAL to exclude node-to-segment contact and consider only edge-to-edge and beam-to-beam contact.
- VDC defines the coefficient of restituion when variable CORTYP is defined. Available for *CONTACT_AUTOMATIC_NODES_TO_SURFACE, *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE, and *CONTACT_AUTOMATIC_SINGLE_SURFACE; SOFT = 0 or 1 only.
- Enhancements for *CONTACT_AUTOMATIC_GENERAL:
  - Add beam to beam contact option CPARM8 in *PART_CONTACT (MPP only).
  - Add option whereby beam generated on exterior shell edge will be shifted into the shell by half the shell thickness. In this way, the shell-edge-to-shell-edge contact starts right at the shell edge and not at an extension of the shell edge (see OPT2 = 10, CPARM8, MPP Card 1).
- Implement *CONTROL_CONTACT PENOPT = 3 option to *CONTACT_AUTOMATIC_NODES_TO_SURFACE and *CONTACT_ERODING_NODES_TO_SURFACE for SMP.
- Update segment based (SOFT = 2) contact to improve accuracy at points away from the origin. The final calculations are now done with nodal and segment locations that have been shifted towards the origin so that coordinate values are small.
- Enable user defined friction (*USER_INTERFACE_FRICTION; subroutine usrfrc) for MPP contact SOFT = 4.
- Unify automatic tiebreak messages for damage start and final failure. SMP and MPP should now give the same output to d3hsp and messag. This affects *CONTACT_AUTOMATIC_...TIEBREAK, OPTIONs 6, 7, 8, 9, 10, and 11.
- *CONTACT_ADD_WEAR: Associates wear calculations to a forming contact interface whose quantities can be posted in the intfor database file. Adaptivity is supported.
- *CONTACT_..._MORTAR:
  - Detailed warning outputs activated for mortar contact, also clarifies echoed data in d3hsp.
  - Contact thickness made consistent with other contacts in terms of priority between ISTUPD on CONTROL_SHELL, SST on CONTACT and OPTT on PART_CONTACT.
  - Efficency improvement of bucket sort in mortar contact allowing for significant speedup in large scale contact simulations.
- *CONTACT_..._MORTAR, *DEFINE_FRICTION, *PART_CONTACT:

- Mortar contact supports FS = -1.0, meaning that frictional coefficients are taken from *PART_CONTACT parameters.
- Mortar contact supports FS.EQ.-2 meaning that friction is taken from *DEFINE_FRICTION.

- *CONTACT_AUTOMATIC_SINGLE_SURFACE_MORTAR: Using IGNORE < 0 for single surface mortar contact will ignore penetrations of segments that belong to the same part.
- Friction factors are now a function of temperature for *CONTACT_..._THERMAL_FRICTION.
- *SET_POROUS_LAGRANGIAN: new keyword to define the porosity of Lagrangian elements in an element set. The porous forces are computed by *CONSTRAINED_LAGRANGE_IN_SOLID CTYPE = 11 or 12.
- *CONSTRAINED_LAGRANGE_IN_SOLID: CTYPE = 12 is now also available in 2D.
- Add helix angle option for *CONSTRAINED_JOINT_GEARS.
- Change keyword from *CONSTRAINED_BEARING to *ELEMENT_BEARING.
- Enhance explicit to use the implicit inertia relief constraints. This allows implicit-explicit switching for such problems.
- Add new input options to *CONTROL_IMPLICIT_INERTIA_RELIEF:
  - user specified number of nodes
  - user specified list of modes to constrain out.
- Implement *CONSTRAINED_BEAM_IN_SOLID. This feature is basically an overhauled constraint couping between beams and Lagrangian solids that includes features that make it more attractive in some cases than *CONSTRAINED_-LAGRANGE_IN_SOLID, for example, in modeling coupling of rebar in concrete.
- Allow *CONSTRAINED_INTERPOLATION to use node set to define the independent nodes.
- Add new feature MODEL.GE.10 to *CONSTRAINED_INTERPOLATION_SPOTWELD (SPR3). This allows parameters STIFF, ALPHA1, RN, RS, and BETA to be defined as *DEFINE_FUNCTIONs of thicknesses and maximum engineering yield stresses of connected sheets.
- Add failure reports for *CONSTRAINED_SPR2.
- Add more d3hsp output for *CONSTRAINED_INTERPOLATION_SPOTWELD and *CONSTRAINED_SPR2. Can be deactivated by setting NPOPT = 1 on *CONTROL_OUTPUT.

- Add option to *CONSTRAINED_JOINT: Relative penalty stiffness can now be defined as function of time when RPS < 0 refers to a load curve. Works for SPHERICAL, REVOLUTE, CYLINDRICAL in explixit analyses.
- Variable MODEL invokes new SPR4 option in *CONSTRAINED_INTERPOLATION_SPOTWELD.
- *CONSTRAINED_JOINT_GEARS: Gear joint now supports bevel gears and similar types, i.e., the contact point does not necessarily have to be on the axis between the gear centers.
- *CONSTRAINED_MULTIPLE_GLOBAL: Support multiple constraints defined on the extra DOFs of user-defined elements.
- Make the *CONTROL_SHELL PSNFAIL option work with the W-MODE deletion criterion for shells.
- New subcycling scheme activated for *CONTROL_SUBCYCLE and *CONTROL_SUBCYCLE_MASS_SCALED_PART. By default the ratio between the largest and smallest time step is now 16 and the external forces are evaluated every time step. The old scheme had a hard wired ratio of 8. The ratios can be optionally changed by *CONTROL_SUBCYCLE_K_L where K is the maximum ratio between time steps for internal forces and L is likewise the ratio for external forces.
- *DATABASE_PROFILE:
  - output kinetic and internal energy profiles,
  - output volume fraction profiles,
  - add a parameter MMG to specify the ALE group for which element data can be output.
- *DATABASE_ALE_MAT: can now use *DEFINE_BOX to compute the material energies, volumes and masses for elements inside boxes (instead of the whole mesh).
- *DATABASE_TRACER_GENERATE: new keyword to create ALE tracer particles along iso-surfaces.
- *DATABASE_FSI: add option to output moments created by FSI forces about each node in a node set. These moments about nodes are reported in dbfsi.
- Add *DATABASE_BEARING to write brngout data pertaining to *ELEMENT_BEARINGs.
- Include eroded hourglass energy in hourglass energy in glstat file to be consistent with KE & IE calculations so that the total energy = kinetic energy + internal energy + hourglass energy + rigidwall energy.
- Add support for new database pbstat (*DATABASE_PBSTAT) for *PARTICLE_BLAST.
  - internal energy and translational energy of air and detonation products

   - force/pressure of air and detonation products for each part

- *DATABASE_EXTENT_INTFOR: New parameter NWEAR on optional card governs the output of wear depth to the intfor database.
- Using CMPFLG = -1 in *DATABASE_EXTENT_BINARY will work just as CMPFLG = 1, except that for *MAT_FABRIC (form 14 and form -14) and *MAT_FABRIC_MAP the local strains and stresses will be engineering quantities instead of Green-Lagrange strain and 2nd Piola-Kirchhoff stress.
- For some materials and elements, thermal and plastic strain tensors can be output to d3plot database, see STRFLG in *DATABASE_EXTENT_BINARY.
- Add option for output of detailed (or long) warning/error messages to d3msg. See MSGFLG in *CONTROL_OUTPUT. Only a few "long" versions of warnings/errors at this time but that list is expected to grow.
- Add two new options for rigid body data compression in d3plot; see DCOMP in *DATABASE_EXTENT_BINARY.
- Add option to write revised legend to jntforc, secforc, rcforc, deforc and nodout files via input flag NEWLEG in *CONTROL_OUTPUT. This helps to avoid confusion over unassigned IDs and duplicated IDs.
- If any input data is encrypted and dynain is requested, the code issues an error message and stops the job.
- Solid part or solid part set is now allowed for *DEFINE_DE_TO_SURFACE_COUPLING.
- Implement *DELETE_PART for Discrete Element Sphere.
- The unit of contact angle changed from radian to degree for *CONTROL_DISCRETE_ELEMENT.
- Implement Archard's wear law to *DEFINE_DE_TO_SURFACE_COUPLING for discrete element spheres. Wear factor is output to DEM binout database.
- Add damping energy and frictional energy of discrete elements to "damping energy" and "sliding interface energy" terms in glstat.
- Introduce a small perturbation to the initial position of newly generated discrete elements for *DEFINE_DE_INJECTION. This allows a more random spatial distribution of the generated particles.
- *INTERFACE_DE_HBOND replaces *INTERFACE_DE_BOND. Used to define the failure models for bonds linking various discrete element (DE) parts within one heterogeneous bond definition (*DEFINE_DE_HBOND).
- *DEFINE_ADAPTIVE_SOLID_TO_DES: Embed and/or transform failed solid elements to DES (*ELEMENT_DISCRETE_SPHERE) particles. The DES particles inherit the material properties of the solid elements. All DES-based features are available through this transformation, including the bond models and contact

algorithms. This command is essentially to DES what *DEFINE_ADAPTIVE_-SOLID_TO_SPH is to SPH particles.

- Add EM orthotropic materials where the electric conductivity is a 3x3 tensor, see new card, *EM_MAT_003.
- Add new keyword family, *EM_DATABASE_... which triggers the output of EM quantities and variables. All EM related ASCII outputs now start with em_***. Keywords are:
  - EM_DATABASE_CIRCUIT
  - EM_DATABASE_CIRCUIT0D
  - EM_DATABASE_ELOUT
  - EM_DATABASE_GLOBALENERGY
  - EM_DATABASE_NODOUT
  - EM_DATABASE_PARTDATA
  - EM_DATABASE_POINTOUT
  - EM_DATABASE_ROGO
  - EM_DATABASE_TIMESTEP
- Add capability to plot magnetic field lines in and around the conductors at given times, see *EM_DATABASE_FIELDLINE. ASCII output files are generated (lspp_fieldLine_xx) and are readable by LSPP in order to plot the field lines. In the future, LSPP will be capable of directly generating the field lines.
- Add EM quantities in *DEFINE_CURVE_FUNCTION:
  - EM_ELHIST for element history (at element center).
  - EM_NDHIST for node history.
  - EM_PAHIST for part history (integrated over the part).
- Add *EM_EOS_TABULATED2 where a load curve defines the electrical conductivity vs time.
- Introduce capability to use the EM solver on (thin) shells: An underlying solid mesh (hexes and prisms) is built where the EM is solved and the EM fields are then collapsed onto the corresponding shell. The EM mat for shells is defined in *EM_MAT_004. This works for EM solvers 1, 2 and 3 and the EM contact is available for shells.
- Add different contact options in the *EM_CONTACT card.
- Add new methods to calculate electric contact resistance between two conductors for Resistive Spot Welding applications (RSW). See *EM_CONTACT_RESISTANCE.
- Add Joule Heating in the contact resistance (*EM_CONTACT_RESISTANCE). The Joule heating is evenly spread between the elements adjacent to the faces in contact.

- Add new circuit types 21 and 22 (see *EM_CIRCUIT) allowing users to put in their own periodic curve shape when using the inductive heating solver. This is useful in cases where the current is not a perfect sinusoidal.
- Provide default values for NCYCLEBEM and NCYCLEFEM (=5000) and set default value of NUMLS to 100 in *EM_CIRCUIT.
- Add two additional formulations, FORM = 3 and 4, to *PART_MODES.
- Add 20-node solid element, ELFORM = 23 in *SECTION_SOLID.
- Add H8TOH20 option to *ELEMENT_SOLID to convert 8-node to 20-node solids.
- Add option SOLSIG to *CONTROL_OUTPUT which will permit stresses and other history variables for multi-integration point solids to be extrapolated to nodes. These extrapolated nodal values replace the integration point values normally stored in d3plot. NINTSLD must be set to 8 in *DATABASE_EXTENT_BINARY when a nonzero SOLSIG is specified. Supported solid formulations are solid elements are: -1, -2, 2, 3, 4, 18, 16, 17, 23.
- Activate contact thickness input from *PART_CONTACT for solids.
- Made many enhancements for *PART_MODES for robustness and MPP implementation.
- Add new cohesive shell element (elform = 29) for edge-to-edge connectivity between shells. This element type takes bending into account and supports MPP and implicit solvers.
- Error terminate with message, STR+1296, if same node is defined multiple times in *ELEMENT_MASS_MATRIX.
- Add support for negative MAXINT option in *DATABASE_EXTENT_BINARY for thick shell elements.
- *ELEMENT_TSHELL: Add "BETA" as option for *ELEMENT_TSHELL to provide an orthotropic material angle for the element.
- Add Rayleigh damping (*DAMPING_PART_STIFFNESS) for triangular shell element types 3 and 17.
- Add new keyword *ELEMENT_BEAM_SOURCE. Purpose: Define a nodal source for beam elements. This feature is implemented for truss beam elements (ELFORM = 3) with material *MAT_001 and for discrete beam elements (ELFORM = 6) with material *MAT_071.
- Add new option to *DEFINE_ELEMENT_DEATH. New variable IDGRP defines a group id for simultaneous deletion of elements.
- Convert cohesive solid type 20 and 22 to incremental formulation to properly handle large rotations. Also use consistent mass.
- Add Smoothed Particle Galerkin (SPG) method for solid analysis (ELFORM = 47) and corresponding keyword option *SECTION_SOLID_SPG. SPG is a true

particle method in Galerkin formulation that is suitable for severe deformation problems and damage analysis.

- Enhance *ELEMENT_LANCING by supporting *PARAMETER, *PARAMETER_-EXPRESSION.
- Add a new feature, *CONTROL_FORMING_TRIMMING, for 2D and 3D trimming of a 3-layer, sandwich laminate blank via *DEFINE_CURVE_TRIM.
- Add 3D normal trimming of solid elements via *DEFINE_CURVE_TRIM_3D.
- Add new features for solid elements 2D trimming *DEFINE_CURVE_TRIM_-NEW:
  - Allow support of arbitrary trimming vector (previously only global z direction was allowed).
  - Improve trimming algorithm for speed up.
  - Allow trimming curves to project to either the top or bottom surface.
- Add a new AUTO_CONSTRAINT option to *CONTROL_FORMING_ONESTEP which is convenient for blank nesting.
- Add new features to *CONTROL_FORMING_SCRAP_FALL. Previously the user was required to define the trimmed blank properly. Now the blank is trimmed by the cutting edge of the trim steel, which is defined by a node set and a moving vector.
- Enhance *CONTROL_FORMING_SCRAP_FALL: Allow the node set (NDSET) on the trim steel edge to be defined in any order.
- Improve *CONTROL_FORMING_ONESTEP:
  - Reposition the initial part before unfolding, using the center element normal.
  - Add a message showing that the initial unfolding is in process.
- Add 2D trimming for solid elements *DEFINE_CURVE_TRIM_NEW, support *DEFINE_TRIM_SEED_POINT_COORDINATES.
- Add *CONTROL_FORMING_AUTOCHECK to detect and fix flaws in the mesh for the rigid body that models the tooling.
- Add new features to *CONTROL_FORMING_UNFLANGING:
  - The incoming flange mesh will be automatically checked for mesh quality and bad elements fixed.
  - Allow thickness offset of deformable flange to use the blank thickness from user's input.
  - Allow definition any node ID in the outer boundary of the flange, to speed up the search when holes are present in the part.
  - Add a new parameter CHARLEN to limit the search region.

  - Allow holes to exist in the flange regions.
  - Output a suggested flange part after unflanging simulation, with the failed elements deleted from the unflanged part.
  - Automatically define a node set and constraints for the flange boundary nodes through the user definition of three nodes.
  - Add output of forming thickness, effective strain and trim curves after unflanging simulation.

- Add a new keyword *CONTROL_FORMING_TRIM to replace *ELEMENT_-TRIM.
- Add a new keyword: *CONTROL_FORMING_UNFLANGING_OUTPUT: Failed elements are removed to come up with the trim curves.
- Add new features to *INTERFACE_BLANKSIZE_DEVELOPMENT including allowing for trimming between initial and final blank.
- Enhance *CONTROL_FORMING_OUTPUT for controlling the number of states.
- Add *CONTROL_FORMING_TRIM_MERGE to close a user specified (gap) value in the trim curves, so each trim curve will form a closed loop, which is required for a successful trimming.
- Add *CONTROL_FORMING_MAXID to set a maximum node ID and element ID for the incoming dynain file (typically the blank) in the current simulation.
- Enhance *FREQUENCY_DOMAIN_ACOUSTIC_BEM:
  - Update the boundary condition definition for BEM acoustics so that impedance and other user defined boundary conditions can be combined with time domain velocity boundary condition.
  - Implement Burton-Miller BEM to MPP.
  - Implement impedance boundary condition to Burton-Miller BEM.
  - Implement half space option (*FREQUENCY_DOMAIN_ACOUSTIC_-BEM_HALF_SPACE) to variational indirect BEM.
  - Implement half space option to acoustic scattering problems.
  - Extend acoustic ATV computation to elements, in addition to nodes.
  - Support element based ATV output in d3atv.
  - Add an option (_MATV) to run modal acoustic transfer vector. Implement MATV to MPP.
  - Implement running BEM Acoustics based on modal ATV (SSD excitation only).
- *FREQUENCY_DOMAIN_ACOUSTIC_FEM: Enable running FEM acoustics based on restarting SSD (*FREQUENCY_DOMAIN_SSD).
- Add *FREQUENCY_DOMAIN_ACOUSTIC_INCIDENT_WAVE to define the incident waves for acoustic scattering problems. To be used with *FREQUENCY_-DOMAIN_ACOUSTIC_BEM.

- Add *FREQUENCY_DOMAIN_ACOUSTIC_SOUND_SPEED to define frequency dependent complex sound speed, which can be used in BEM acoustics. By using complex sound speed, the damping in the acoustic system can be considered. To be used with *FREQUENCY_DOMAIN_ACOUSTIC_BEM.
- *FREQUENCY_DOMAIN_FRF: Add mode dependent rayleigh damping to frf and ssd (DMPMAS and DMPSTF).
- *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM:
  - Add output of nodout_spcm and elout_spcm, to get nodal results and element results at user specified nodes and elements.
  - Add von Mises stress computation.
- *FREQUENCY_DOMAIN_RANDOM_VIBRATION: Add semi-log, and linear-linear interpolation on PSD curves (parameter LDFLAG).
- *FREQUENCY_DOMAIN_SSD:
  - Add strain computation.
  - Add parameter LC3 to define the duration of excitation for each frequency.
  - Implement fatigue analysis option (_FATIGUE) based on ssd (sine sweep).
  - Add option to use *DAMPING_PART_MASS and *DAMPING_PART_STIFFNESS in SSD (DMPFLG = 1).
- Add *MAT_ADD_FATIGUE to define material's SN fatigue curve for application in vibration fatigue and SSD fatigue analysis.
- Add *FREQUENCY_DOMAIN_ACCELERATION_UNIT to facilitate the acceleration unit conversion.
- The icfd_mstats.dat file now outputs the ten worst quality element locations (ICFD solver).
- Add option in *ICFD_CONTROL_OUTPUT allowing terminal output to be written to messag file.
- Add keyword *ICFD_CONTROL_OUTPUT_SUBDOM to output only part of the domain. Available for vtk, dx and gmv formats.
- Add new keyword family, *ICFD_DATABASE_... which triggers the output of ICFD variables. All ICFD related output files now start with icfd_***.
- Add new keyword family *ICFD_SOLVER_TOL_... which allows the user to control tolerances and iteration number for the fractional step solve, the mesh movement solve, and the heat equation solve.
- Curves in *ICFD_BOUNDARY_PRESCRIBED_VEL each provide a scaling factor vs. x, y, or z coordinate, respectively. These scaling factors are applied to the velocity boundary condition.
- Enable free-slip condition for FSI walls (ICFD solver).

- Add new variable IDC to *ICFD_CONTROL_FSI that allows the modification of the scaling parameter that multiplies the mesh size to detect contact.
- Add automatic squeezing to the ICFD elements of the boundary layer when there are two very close surfaces with poor (coarse) mesh resolution.
- Add the initialization for all nodes using *ICFD_INITIAL with PID = 0.
- Add a curve (LCIDSF in *ICFD_CONTROL_TIME) that scales the CFL number as a function of time.
- Add a Heaviside function that allows the solution of simple multiphase problems (ICFD).
- Add the computation of the heat convection coefficient (ICFD).
- Add MPP support for y+ and shear for output (ICFD).
- Add uniformity index (ICFD).
- Add *ICFD_CONTROL_TAVERAGE to control the restarting time for computing the time average values.
- Implement the XMl format for vtk. See *ICFD_CONTROL_OUTPUT.
- Improve temperature stabilization for thermal problems (ICFD).
- Add the Generalized Flow Through Porous Media model monolithically coupled to the incompressible Navier-Stokes model. See keyword *ICFD_MAT for the new options.
- Add the Anisotropic version of the Generalized Flow in Porous Media. See *ICFD_MAT for details.
- Add the capability to define the porous properties using the Pressure-Velocity (P-V) experimental curves. See *ICFD_MAT.
- Compute drag forces around anisotropic/isotropic porous domains (ICFD).
- Extend implicit debug checking when LPRINT = 3 on *CONTROL_IMPLICIT_SOLVER.
- Add option for implicit dynamic relaxation so that only a subset of parts is active during the dynamic relaxation phase.
- Extend implicit time step control via IAUTO < 0 in *CONTROL_IMPLICIT_AUTO to linear analysis.
- Add self-piercing rivet capability to implicit (*CONSTRAINED_SPR2, *CONSTRAINED_INTERPOLATION_SPOTWELD).
- Add MTXDMP in *CONTROL_IMPLICIT_SOLVER to dump the damping matrix from implicit mechanics.
- Improve stress and strain computation induced by mode shapes. See MSTRES in *CONTROL_IMPLICIT_EIGENVALUE.

- Add variable MSTRSCL to *CONTROL_IMPLICIT_EIGENVALUE for user control of geometry scaling for the stress computation.
- Make SMP and MPP treatment of autospc constraints consistent. See AUTOSPC on *CONTROL_IMPLICIT_SOLVER.
- Enhance output for *ELEMENT_DIRECT_MATRIX_INPUT (superelements) to describe how they are attached to the LS-DYNA model.
- Enhance superelement computation (*CONTROL_IMPLICIT_MODES or *CONTROL_IMPLICIT_STATIC_CONDENSATION):
    - The computation of the inertia matrix in the presense of rigid bodies is correct.
    - Adjust superelement computation to accept initial velocities.
    - Add null beams for the visualization of superelements.
- Enhance implicit to allow the use of *CONSTRAINED_RIVET in conjunction with axisymmetric shell element problems.
- Add output of performance statistics for the MPP implicit eigensolver to mes0000.
- Add stress computation to modal dynamics (*CONTROL_IMPLICIT_MODAL_DYNAMIC).
- Allow nonsymmetric terms to the assembled stiffness matrix from some implicit features.
- Enhance implicit-explicit switching (IMFLAG < 0 in *CONTROL_IMPLICIT_GENERAL) so that curve |IMFLAG| can be defined using *DEFINE_CURVE_FUNCTION.
- Upgrade the implicit implementation of rack and pinion and screw joints so the joint is driven by relative motion of the assembly instead of absolute motion.
- Add *CONTACT_1D to implicit mechanics.
- *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS is added to study Rotordynamics using the implicit time integrator.
- *MAT_SEATBELT is supported for implicit by introducing bending stiffness.
- *INITIAL_LAG_MAPPING added to initialize a 3D Lagrangian mesh from the last cycle of a 2D Lagrangian simulation.
- *ELEMENT_SHELL_NURBS_PATCH:
    - Add support for dumping of strain tensor and shell internal energy density for isogeometric shells via interpolation shells.
    - Add conventional mass-scaling for isogeometric shells.
- *LOAD_BODY_POROUS: applies also now to 1D and 2D problems.

- Add *LOAD_SEGMENT_CONTACT_MASK, which currently works in MPP only. This feature masks the pressure from a *LOAD_SEGMENT_SET when the pressure segments are in contact with another material.
- Curve LCID of *LOAD_NODE can be defined by *DEFINE_CURVE_FUNCTION.
- *USER_LOADING: pass more data to user-defined loading subroutine loadud including nodal moment, nodal rotational displacement and velocity, and nodal translational mass and rotational inertia.
- Add load curves for dynamic relaxation for *LOAD_THERMAL_VARIABLE.
- *LOAD_SEGMENT_NONUNIFORM, *LOAD_SEGMENT_SET_NONUNIFORM: By specifying a negative load curve ID the applied load becomes a follower force, i.e., the direction of the load is constant with respect to a local coordinate system that rotates with the segment.
- Make several enhancements to *MAT_172.
- *MAT_HYPERELASTIC_RUBBER (*MAT_077_H) has new thermal option for material properties.
- Add *MAT_ORTHOTROPIC_PHASE_CHANGE, *MAT_ELASTIC_PHASE_CHANGE, and *MAT_MOONEY-RIVLIN_PHASE_CHANGE whereby elements change phase as they cross a plane in space.
- Add P1DOFF to 2D seatbelt material, *MAT_SEATBELT_2D, to specify a part ID offset for the internally created 1D seatbelt elements.
- All load curves for *MAT_067 can be defined via *DEFINE_FUNCTION.
- Enhance *MAT_CWM:
    - Add support for shell elements.
    - Add support for hardening curves. Yield stress can be supplied as table depending on plastic strain and temperature.
- Check diagonal elements of C-matrix of *MAT_002/MAT_*OPTION*TROPIC_ELASTIC and error terminate with message, STR+1306, if any are negative.
- Add a keyword option called MIDFAIL for *MAT_024, (MAT_PIECEWISE_LINEAR_PLASTICITY). When MIDFAIL appears in the keyword, failure by plastic strain will only be checked at the mid-plane. If the mid-plane fails, then the element fails. If there are an even number of integration points through the thickness, then the two points closest to the middle will check for failure and the element fails when both layers fail.
- Enable solid and solid assembly spot welds (*MAT_SPOTWELD) to use the NF parameter for force filtering.
- Add the shear angle in degrees as the first history variable for shell material *MAT_214 (DRY_FABRIC).

- Expand from 2 to 5 the number of additional cards that can be used for the user defined weld failure, OPT = 12 or OPT = 22 on *MAT_SPOTWELD. Now a total of 46 user variables are possible.
- Add a solid spot weld material option in *MAT_SPOTWELD to treat the stress state as uniaxial. This option is available for solid assemblies also.
- Add *MAT_FABRIC form 24 which is a modified version of form 14. The main improvement is that the Poisson's effects work correctly with the nonlinear curves for fiber stress. Also, the output of stress and strain to d3plot are engineering stress and strain instead of 2nd PK stress and Green's strain. Added an option to input curves in engineering stress and strain rather than 2nd PK stress vs. Green's strain. To use this, set DATYP = -2 on *DEFINE_CURVE.
- Increase maximum number of plies from 8 to 24 in a sublaminate with *MAT_CODAM2.
- Add *MAT_THERMAL_CHEMICAL_REACTION to model a material undergoing a chemical reaction such as an epoxy used in manufacturing composite materials.
- *MAT_058:
    - Add option to use nonlinear (elastic) stress-strain curves instead of constant stiffnesses (EA, EB, GAB).
    - Add option to use strain-rate dependent nonlinear (elastic) stress-strain curves instead of constant stiffnesses (EA, EB, GAB).
    - Add option to define proper poisson ratios PRCA and PRCB (also added in *MAT_158).
- Add option to use yield curve or table in *MAT_100 (*MAT_SPOTWELD) for solid elements.
- Add *MAT_157 for solid elements. This includes an optional variable IHIS that invokes *INITIAL_STRESS_SOLID to initialize material properties on an element-by-element basis. This was developed to allow a user to map/initialize anisotropic material properties from an injection molding simulation.
- *MAT_157 (shells):
    - Add anisotropic scale factor for plastic strain rate (VP = 1 only).
    - Improve local stress projection for VP = 1.
    - Add optional variable IHIS, similar to that described for solids above.
- Add strain rate dependence to *MAT_103 for solids via a table (isotropic hardening only).
- *MAT_136 (*MAT_CORUS_VEGTER): Implemented an alternative, implicit plasticity algorithm (define N.lt.0) for enhanced stability.
- *MAT_244 (*MAT_UHS_STEEL):

- In plasticity with non-linear hardening, temperature effects and strain rate effects are now dealt with the same way they are implemented in *MAT_106. In particular, strain rate now refers to the plastic strain rate.
- Allow for the definition of start temperatures for each phase change, for cooling and heating.
- Account for elastic transformation strains, given as a curve wrt temperature.
- Add feature to *MAT_244 for welding simulations. Similar to *MAT_270, material can be initialized in a quiet (ghost) state and activated at a birth temperature.

- Furthermore, annealing is accounted for.
- Modify formula for Pearlite phase kinetics based on Kirkaldy and Venugoplan (1983).
- *MAT_249 (*MAT_REINFORCED_THERMOPLASTIC): Implement new material formulation for shells, which is based on additive split of stress tensor.
    - For the thermoplastic matrix, a thermo-elasto-plastic material is implemented,where the temperature dependence is defined by load curves/tables in the input file.
    - Includes hyperelastic fiber contribution.
    - For any integration point, up to three different fiber directions can be defined. Their (non-linear) response to elongation and shear deformations can also be defined with load curves.
    - Includes input parameters for anisotropic transverse shear stiffness.
- *MAT_T07 (*MAT_THERMAL_CWM): Add HBIRTH and TBIRTH which are specific heat and thermal conductivity, resp., used for time t < TISTART.
- One additional parameter (exponent GAMMA) for B-K law of *MAT_138.
- MAT_187: Speed-up of load curve lookup for curves with many points.
- Add new option "MAGNESIUM" to *MAT_233. Differences between tension and compression are included.
- Add enhanced damage model with crack closure effects to *MAT_104.
- Some improvements for *MAT_075 (BILKHU/DUBOIS_FOAM): Volumetric strain rate can now be averaged over NCYCLE cycles, original input curve LCRATE is instead of a rediscretized curve, and averaged strain rate is stored as history variable #3.
- Add new history variables to *MAT_123: A mixed failure indicator as history variable #10 and triaxiality as #11.
- Decrease memory requirements for *MAT_ADD_EROSION by 50%.
- Add *MAT_098 for tetrahedral solid type 13.

- Add new history variable #8 to *MAT_157 for shell elements: "Anisotropic equivalent plastic strain".
- Add tangent stiffness to *MAT_224 for implicit analyses with solid and shell elements.
- Put internal enery on "plastic strain" location for *MAT_027 solids.
- Add new option *MAT_224: BETA < 0: strain rate dependent amount given by load curve ID = -BETA
- Add new flag to switch off all MAT_ADD_EROSION definitions globally.
- This will be the 1st parameter "MAEOFF" on new keyword *CONTROL_MAT.
- Add option to define a load curve for isotropic hardening in *MAT_135.
- *MAT_CDPM is reimplemented by its original developers (Peter Grassl and Dimitros Xenos at University of Glasgow) for enhanced robustness. A new parameter EFC is introduced governing damage in compression and the bilinear law is exchanged for an exponential one.
- *MAT_3-PARAMETER_BARLAT: HR = 7 is complemented with biaxial/shear hardening curves.
- *MAT_FABRIC_MAP:
  - A stress map material for detailed stress response in fabrics, stress can be prescribed through tables PXX and PYY corresponding to functions of biaxial strain states.
  - A compaction effect due to packing of yarns in compression is obtained by specifying BULKC (bulk modulus) and JACC (critical jacobian for the onset of compaction effect). This results ib increasing pressure that resists membrane elements from collapsing and/or inverting.
  - Strain rate effects can be obtained by specifying FXX and FYY which in effect scales the stress based on engineering strain rate. A smoothing effect is applied by using a time window DT.
  - A hysteresis option TH is implemented for stability, given in fraction dissipated energy during a cycle. Can also depend on the strain state through a table.
- *MAT_GENERAL_HYPERELASTIC_RUBBER, *MAT_OGDEN_RUBBER: By specifying TBHYS.LT.0 a more intuitive interpolation of the damage vs. deviatoric strain energy is obtained. It requires however that the damage and strain energy axes are swapped.
- *MAT_SIMPLIFIED_RUBBER: For AVGOPT.LT.0 the absolute value represents a time window over which the strain rates are averaged. This is for suppressing extensive noise used for evaluating stress from tables.

- *MAT_FABRIC: The bending stiffness contribution in material 34, ECOAT/SCOAT/TCOAT, is now supported in implicit calculations.
- Add *MAT_122_3D which is an extension of *MAT_122 to solid elements. This material model combines orthotropic elastic behavior with Hill's 1948 anisotropic plasticity theory and its applicability is primarily to composite materials.
- MPP groupable tied contact: Output messages about initial node movement due to projection like non-groupable routines do.
- MPP tied contact initialization:
    - Change a tolerance in groupable tied contact bucketsort to match the non-groupable code and fix the slave node thickness used for beam nodes during initial search in non-groupable contact to match groupable contact.
    - Update the slave node from beam thickness calculation for type 9,11, and 12 beams.
- For MPP, set a "last known location" flag to give some indication of where the processors were if an error termination happens. Each writes a message to their own message file. Look for a line that says "When error termination was triggered, this processor was".
- MPP BEAMS_TO_SURFACE contact: Remove "beam" node mass from the penalty stiffness calculation when soft = 1 is used, which matches SMP behavior.
- Make sure the pfile.log file gets created in case of termination due to *CONTROL_-STRUCTURED_TERM.
- Add two new decomposition region-related pfile options "nproc" and "%proc" so that any given decomposition region can be assigned to some subset of all the processors. nproc takes a single argument, which is a specific number of processors. %proc takes a single argument, which is a percentage of processors to use. The old options "lump" and "distribute" are still available and are mapped to the new options thusly:
    - lump => "nproc 1"
    - distribute => "%proc 100.0"
- Tweak MPP beam-to-beam contact routine for better handling of parallel beams.
- MPP: Add support for new solid and shell cost routines, invoked with the pfile option "decomp { newcost }". Will be expanded to include beams, thick shells, etc. in the future.
- MPP contact: add support for IGAP > 2 added to the SINGLE_SURFACE, AUTOMATIC_GENERAL, and *_TO_SURFACE contacts.
- Improve the way MPP computes slave node areas for AUTOMATIC_TIEBREAK contacts (and other that use areas). This should result in less mesh dependency in the failure condition of AUTOMATIC_TIEBREAK contacts.

- MPP: synchronize rigid body flags for shared nodes during rigid-to-deformable switching so that these nodes are handled consistently across processors.
- Add new pfile decomposition region option "partsets". Takes a list of part sets (SET_PART) from the keyword input and uses them to define a region.
- Apply decomposition transformation (if defined) to:
  - *CONTROL_MPP_DECOMPOSITION_PARTS_DISTRIBUTE
  - *CONTROL_MPP_DECOMPOSITION_PARTSET_DISTRIBUTE
  - *CONTROL_MPP_DECOMPOSITION_ARRANGE_PARTS.
- Honor TIEDPRJ flag on *CONTROL_CONTACT for MPP groupable tied interfaces.
- Increase initial search distance in MPP tied contact to include slave and master thicknesses.
- Tweak MPP_INTERFERENCE contact to better handle deep initial penetrations.
- MPP: Reorganize how *RIGIDWALL_PLANAR_FORCES is handled, which greatly improves scaling.
- Add new MPP pfile option: directory { local_dirs { path1 path2 path3 } which will assign different local working directories to different processors, to balance the I/O load.
- Miscellaneous MPP enhancements:
  - Restructure and reduce memory usage of 3D ALE searching of neighboring algorithm. Now, the code can handle hundreds of millions ALE elements during decomposition.
  - Support *PARTICLE_BLAST.
  - Support SPH 2D contact.
  - Greatly speed up reconstruction of eroding contact surface, (soft = 0,1) when using large number of cores.
- Add the following options for small restarts:
  - *CHANGE_VELOCITY_GENERATION,
  - *CHANGE_RIGIDWALL_option,
  - PSNFAIL option to *CONTROL_SHELL
- MPP full deck restart: Restore behavior consistent with SMP which is that only the nodes of materials being initialized (not all nodes) are initialized from d3full.
- MPP: add full deck restart support for AUTOMATIC_TIEBREAK contact types.
- Implement *DELETE_PART for seatbelt parts. The associated slipring, retractors and pretensioners will be deactivated as well.
- Add support for MPP restarts with USA coupling.

- Add NREP option to *SENSOR_CONTROL to repeat NREP cycles of switches given on Card 2.
- Implement *SENSOR_CONTROL TYPEs BELTPRET, BELTRETRA and BELTSLIP control the pretensioners, retractors and sliprings of a 2D seatbelt.
- Add function SENSORD to *DEFINE_CURVE_FUNCTION to return the value of a sensor.
- Replace *SENSOR_DEFINE_ANGLE with more general *SENSOR_DEFINE_MISC. MTYPEs include ANGLE, RETRACTOR, RIGIDBODY, and TIME.
- Add rcforc output for *CONTACT_2D_NODE_TO_SOLID (supported for ASCII output only; not binout).
- Add temperature output (when applicable) to sphout file (*DATABASE_SPHOUT).
- Add support of *MAT_ALE_VISCOUS for SPH particles. This allows modeling of non-viscous fluids with constant or variable viscosity, i.e, non-newtonian type fluid using SPH.
- Add support of *EOS for *MAT_272 with SPH particles.
- Add support of *MAT_255, *MAT_126, and *MAT_26 (with AOPT = 2 only) for SPH particles.
- Add new keyword command *SECTION_SPH_INTERACTION: Combined with CONT = 1 in *CONTROL_SPH card, this keyword is used to define the partial interaction between SPH parts through normal interpolation method and partial interaction through the contact option. All the SPH parts defined through this keyword will interact with each other through normal interpolation method automatically.
- Add support for *DATABASE_TRACER for axisymmetric SPH (IDIM = -2 in *CONTROL_SPH).
- ICONT in *CONTROL_SPH now affects *DEFINE_SPH_TO_SPH_COUPLING in the sense of enabling or disabling the coupling for deactivated particles.
- The commands *STOCHASTIC_TBX_PARTICLES and *CHEMISTRY_CONTROL_TBX are now available for use (along with the CESE solver) in TBX-based explosives simulations.
- Multi-nozzle injection mode is implemented for spray injection.
- Add logic to skip thermal computations during dynamic relaxation for a coupled thermal-structual problem (i.e. when SOLN = 2 on the *CONTROL_SOLUTION keyword). This does not affect the use of *LOAD_THERMAL keywords during dynamic relaxation.
- Implement *DEFINE_CURVE_FUNCTION for convection, flux, radiation boundary

- conditions in thermal-only analyses, both 2D and 3D.
- *BOUNDARY_CONVECTION, *BOUNDARY_FLUX, and thermal dynamics are implemented for 20 node brick element.
- Include the reading of thermal data to *INCLUDE_BINARY.
- Allow *DEFINE_FUNCTION_TABULATED to be used in any place that requires a function of 1 variable. Specifically, as a displacement scale factor with *INTERFACE_LINKING_NODE.
- Add new MUTABLE option for *PARAMETER and *PARAMETER_EXPRESSION to indicate that it is OK to redefine a specific parameter even if *PARAMETER_DUPLICATION says redefinition is not allowed. Also, only honor the first *PARAMETER_DUPLICATION card.
- Add functions DELAY and PIDCTL to *DEFINE_CURVE_FUNCTION for simulating PID (proportional-integral-derivative) controllers.
- *DEFINE_TABLE: Add check of table's curves for mismatching origin or end points.
- Update ANSYS library to version 16.0.
- Enhance report of "Elapsed time" in d3hsp.
- Add keyword *INCLUDE_UNITCELL to create a keyword file containing user-defined unit cell information with periodic boundary conditions.
- Add *INCLUDE_AUTO_OFFSET: the node and element IDs of the include file will be checked against IDs of the previously read data to see if there is any duplication. If duplicates are found, they will be replaced with another unique ID.

Capabilities added to create LS-DYNA R9.0:

See release notes (published separately) for further details.

- ***AIRBAG**
  - Disable CPM airbag feature during DR and reactivate in the transient phase.
  - *AIRBAG_WANG_NEFSKE_POP_ID pop venting based on RBIDP is now supported correctly (MPP only).
  - *AIRBAG_INTERACTION:
    - Fixed MPP airbag data sync error to allow final pressure among interacted airbags to reach equilibrium.
  - *AIRBAG_PARTICLE:
    - When IAR = -1 and Pbag or Pchamber is lower than Patm, ambient air will inflate the bag through external vents and also fabric porosity.

- Treat heat convection when chamber is defined.
- Output pres+ and pres- to CPM interface forces file for internal parts.
- Allow IAIR = 4 to gradually switch to IAIR = 2 to avoid instability.
- Allow using shell to define inflator orifice. The shell center and normal will be used as orifice node and flow vector direction.
- Bug fix for porous leakage for internal fabric parts using CPM.
- New feature to collect all ring vents into a single vent in order to correctly treat enhanced venting option. All the vent data will only be output to the first part defined in the part set.
- Evaluate airbag volume based on relative position to avoid truncation. The bag volume becomes independent of coordinate transformation.
- Support explicit/implicit switch and dynamic relaxation for *AIRBAG_PARTICLE.
- Support vent/fabric blockage for CPM and ALE coupled analysis.
- New option in *CONTROL_CPM to allow user defined smoothing of impact forces.
- Fixed bug affecting *AIRBAG_PARTICLE_ID with PGP encryption.

- ***ALE**

  - *ALE_REFERENCE_SYSTEM_GROUP: For prtype = 4, allow the ALE mesh to follow the center of mass of a set of nodes.
  - *CONTROL_ALE:

    - Add a variable DTMUFAC to control the time step related to the viscosity from
    - *MAT_NULL (if zero, the viscosity does not control the time step).
    - Implement a 2D version of BFAC and CFAC smoothing algorithm.

  - *ALE_SMOOTHING: Automatically generate the list of 3 nodes for the smoothing constraints and implement for MPP.
  - *SECTION_ALE2D, *SECTION_SOLID_ALE: Allow a local smoothing controlled by AFAC,...,DFAC.
  - *ALE_SWITCH_MMG: Allow the variables to be modified at the time of the switch.
  - *CONTROL_REFINE_ALE: Add a variable to delay the refinement after removal (DELAYRGN), one to delay the removal after the refinement (DELAYRMV), and one to prevent any removal in a certain radius around latest refinements (RADIUSRMV).
  - *ALE_STRUCTURED_MESH: Implemented structured ALE mesh solver to facilitate rectilinear mesh generation and to run faster.

- ***BOUNDARY**

  - *BOUNDARY_AMBIENT_EOS: Implement *DEFINE_CURVE_FUNCTION for the internal energy and relative volume curves.

- *BOUNDARY_AMBIENT: Apply ambient conditions to element sets.
- Fix for adaptivity dropping SPCs in some cases (MPP only).
- Added conflict error checking between rigid body rotational constraints and joints between rigid bodies (*CONSTRAINED_JOINT) with *BOUNDARY_PRESCRIBED_ORIENTATION.
- The first rigid body of the prescribed orientation cannot have any rotational constraints. Only spherical joints or translational motors can be used between the two rigid bodies of the prescribed orientation. For now explicit will be allowed to continue with these as warnings.Implicit will terminate at end of input checking.
- Instead of error terminating with warning message, STR+1371, when *BOUNDARY_PRESCRIBED_MOTION and *BOUNDARY_SPC are applied to same node and dof, issue warning message, KEY+1106, and release the conflicting SPC.
- Fix erroneous results if *SET_BOX option is used for *BOUNDARY_PRESCRIBED_MOTION.
- Fix *BOUNDARY_PRESCRIBED_ACCELEROMETER_RIGID for MPP. It may error terminate or give wrong results if more than one of this keyword are used.
- Fix segmentation fault when using *BOUNDARY_PRESCRIBED_ORIENTATION with vad = 2, i.e. cubic spline interpolation.
- Fix incorrect behavior if multiple *BOUNDARY_SPC_SYMMETRY_PLANE, i.e. > 1, are used.
- Fix incorrect motion if *BOUNDARY_PRESCRIBED_MOTION_RIGID_LOCAL is on a rigid part which is merged with a deformable part that has been switched to rigid using *DEFORMABLE_TO_RIGID.
- Fix incorrect external work when using *BOUNDARY_PRESCRIBED_MOTION with or without_RIGID option. The dof specified in *BPM was not considered when computing the external work. Also, when multiple *BPM applied to the same node/rigid body with different dof may also cause incorrect computation of external work.
- Fix incorrect velocities when using *BOUNDARY_PRESCRIBED_MOTION_RIGID_LOCAL and *INITIAL_VELOCITY_RIGID_BODY for rigid bodies.
- Implement check for cases where *MAT_ACOUSTIC nodes are merged with structural nodes on both sides of a plate element and direct the user to the proper approach to this situation - *BOUNDARY_ACOUSTIC_COUPLING.
- *BOUNDARY_ACOUSTIC_COUPLING with unmerged, coincident node coupling now implemented in MPP.
- MPP logic corrected so *MAT_ACOUSTIC and *BOUNDARY_ACOUSTIC_COUPLING features may be used with 1 MPP processor.
- Fixed bug for *BOUNDARY_PRESCRIBED_MOTION if part label option is used.

- **BLAST**

  - Improve *LOAD_BLAST_ENHANCED used with ALEPID option in *LOAD_BLAST_SEGMENT:
  - Rearrange the ambient element type 5 and its adjacent element into same processor to avoid communications.
  - Eliminate several n-by-n searches for segment set and ambient type 5 with its neighboring elements to speed up the initialization.
  - Change the name of keyword *DEFINE_PBLAST_GEOMETRY to *DEFINE_PBLAST_HEGEO. Both names will be recognized.

- ***CESE (Compressible Flow Solver)**

  - Modified the CESE moving mesh CHT interface condition calculation to deal with some occasional MPP failures that could occur with mesh corner elements.
  - Improved the CESE spatial derivatives approximation in order to bring better stability to the CESE solvers.
  - The 3D SMP and MPP CESE immersed boundary solvers now work with structural element erosion.
  - A new energy conservative conjugate heat transfer method has been added to the following 2D and 3D CESE Navier-Stokes equation solvers:

    - Fixed mesh (requires use of *CESE_BOUNDARY_CONJ_HEAT input cards)
    - Moving mesh FSI
    - Immersed boundary FSI

  - Prevent the fluid thermal calculation from using too short a distance between the fluid and structure points in the new IBM CHT solvers.
  - In the under resolved situation, prevent the CHT interface temperature from dipping below the local structural node temperature.
  - Add detection of blast wave arrival at CESE boundary condition face first sensing the leading edge of the pulse (used with *LOAD_BLAST_ENHANCED).
  - Set CESE state variable derivatives to more stable values for the blast wave boundary condition.
  - Corrected time step handling for the CESE Eulerian conjugate-heat transfer solver. This affected only the reported output time.
  - Added CESE cyclic BC capability to the moving mesh CESE solver.
  - Fixed some issues with 2D CESE solvers where the mesh is created via *MESH cards.
  - For the CESE solver coupled with the structural solver (FSI), corrected the time step handling.

- For the CESE mesh motion solvers, and the ICFD implicit ball-vertex mesh motion solver, added a mechanism to check if all of the imposed boundary displacements are so small that it is not necessary to actually invoke the mesh motion solver. This is determined by comparing the magnitude of the imposed displacement at a node with the minimum distance to a virtual ball vertex (that would appear in the ball-vertex method). The relative scale for this check can be input by the user via field 4 of the *CESE_CONTROL_-MESH_MOV card.
- Changed the NaN check capability for the CESE solvers to be activated only upon user request. This is input via a non-zero entry in field 7 of the *CESE_-CONTROL_SOLVER card.
- Much like the ICFD solver, added a mechanism to adjust the distance used by the contact detection algorithm for the *CESE_BOUNDARY_FSI cards, as well as the new moving mesh conjugate heat transfer solvers. This is available through field 6 of the *CESE_CONTROL_SOLVER card.
- Added a correction to the moving mesh CESE solver geometry calculation.
- Corrected the initial time step calculation for both the 2D and 3D moving mesh CESE solvers.
- For the moving mesh CESE solver, replaced the all-to-all communication for fsi quantities with a sparse communication mechanism.

- ***CHEMISTRY**

  - The immersed boundary FSI method coupled with the chemistry solver is released.

    - Only Euler solvers, both in 2D and 3D, are completed with full chemistry.
    - Using this technique, CESE FSI Immerged Boundary Method coupled to the chemistry solver can be applied to high speed combustion problems such as explosion, detonation shock interacting with structures, and so on.
    - Some examples are available on our ftp site.

- ***CONTACT**

  - Fix MPP groupable contact problem that could in some cases have oriented the contact surfaces inconsistently.
  - Fix bug in *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIED_-WELD.
  - Fix seg fault when using *CONTACT_AUTOMATIC_SINGLE_SURFACE_-TIED with consistency mode, .i.e. ncpu < 0, for SMP.
  - Fix false warnings, SOL+1253, for untied nodes using *CONTACT_AUTO-MATIC_SURFACE_TO_SURFACE_TIEBREAK and *CONTACT_AUTO-MATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK.

- Fix *CONTACT_TIED_SHELL_EDGE_TO_SURFACE when rigid nodes are not tied even when ipback = 1. This applies to SMP only.
- Issue warning if SOFT = 4 is used with an unsupported contact type, and reset it to 1.
- Change "Interface Pressure" report in intfor file from abs(force/area) to -force/area, which gives the proper sign in case of a tied interface in tension.
- Increase MPP contact release condition for shell nodes that contact solid elements in SINGLE_SURFACE contact.
- Fix for MPP IPBACK option for creating a backup penalty-based tied contact.
- Fix for MPP orthotropic friction in contact.
- Fix for MPP *CONTACT_SLIDING_ONLY that was falsely detecting contact in some cases.
- Skip constraint based contacts when computing the stable contact time step size.
- Add error trap if node set is input for slave side of single surface contact.
- MPP: some fixes for constrained tied contact when used with adaptivity. The behavior of the slave nodes in adaptive constraints was not correct if they were also master nodes of a tied interface. This has been fixed, and support for the rotations required for CONTACT_SPOTWELD have also been added.
- MPP: update to AUTOMATIC_TIEBREAK option 5 to release the slave nodes (and report them as having failed) when the damage curve reaches 0.
- Fix made to routine that determines the contact interface segments, which was not handling pentahedral thick shell elements correctly.
- MPP: fix for strange deadlock that could happen if a user defines a *CONTACT_FORCE_TRANSDUCER that has no elements in it and so gets deleted.
- MPP contact: add support for *DEFINE_REGION to define an active contact region. Contact occurring outside this region is ignored. This is only for MPP contact types:

    - AUTOMATIC_SINGLE_SURFACE
    - AUTOMATIC_NODES_TO_SURFACE
    - AUTOMATIC_SURFACE_TO_SURFACE
    - AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE

- MPP fix for table based friction in non-groupable contact.
- MPP: add frictional work calculation for beams in *CONTACT_AUTOMATIC_GENERAL.
- Added new option "FTORQ" for contact. Currently implemented only for beams in *CONTACT_AUTOMATIC_GENERAL in MPP. Apply torque to the nodes to compensate for the torque introduced by friction. Issue error message when users try to use SOFT = 2/DEPTH = 45 contact for solid elements.

- R-adaptivity, ADPTYP = 7 in *CONTROL_ADAPTIVE, is now available for SMP version of *CONTACT_SURFACE_TO_SURFACE, _NODES_TO_-SURFACE, _AUTOMATIC_SURFACE_TO_SURFACE, and _AUTOMAT-IC_NODES_TO_SURFACE (SOFT = 0 or 1 only).
- The options AUTOMATIC_SURFACE_TO_SURFACE_COMPOSITE has been added to model composite processing. The same option may be used to model certain types of lubrication, and AUTOMATIC_SURFACE_TO_-SURFACE_LUBRICATION may be used instead of the COMPOSITE option for clarity. (The two keyword commands are equivalent.)
- Added AUTOMATIC_SURFACE_TO_SURFACE_TIED_WELD to model the simulation of welding. As regions of the surfaces are heated to the welding temperature and come into contact, the nodes are tied.
- Added *CONTACT_TIED_SHELL_EDGE_TO_SOLID. This contact transmits the shell moments into the solid elements by using forces unlike the SHELL_EDGE_TO_SURFACE contact with solid elements. This capability is easier for users than *CONSTRAINED_SHELL_TO_SOLID. The input is identical to *CONTACT_TIED_SHELL_EDGE_TO_SURFACE (except for the keyword).
- Fix incorrect motion of displayed rigidwall between 0.0 < time < birth_time when birth time > 0.0 for *RIGIDWALL_GEOMETRIC_FLAT_MOTION_-DISPLAY. The analysis was still correct. Only the displayed motion of the rigidwall is incorrect.
- Fix corrupted intfor when using parts/part sets in *CONTACT_AUTO-MATIC_.... This affects SMP only.
- Fix incorrect stonewall energy when using *RIGIDWALL_PLANAR_OR-THO.
- Fix unconstrained nodes when using *CONTACT_TIED_SURFACE_TO_-SURFACE_CONSTRAINED_OFFSET resulting in warning message, SOL+540. This affects SMP only.
- Fix spurious repositioning of nodes when using *CONTACT_SURFACE_-TO_SURFACE for SMP.
- Enable MAXPAR from optional card A to be used in *CONTACT_TIED_-SURFACE_TO_SURFACE. It was originally hard-coded to 1.07.
- The shells used for visualisation of *RIGIDWALL_PLANAR_MOVING_-DISPLAY and *RIGIDWALL_PLANAR_MOVING_DISPLAY in d3plot were not moving with the rigidwall. This is now fixed.
- Fix incorrect frictional forces if ORTHO_FRICTION is used in *CONTACT_-AUTOMATIC_SURFACE_TO_SURFACE.
- Fix seg fault when using *CONTACT_ENTITY and output to intfor file with MPP, i.e. s = intfor in command line.
- Fix ineffective birth time for *CONTACT_TIED_NODES_TO_SURFACE.
- Fix untied contacts when using *CONTACT_TIED... with *MAT_ANISO-TROPIC_ELASTIC_PLASTIC/*MAT_157.
- Fix MPP hang up when using *CONTACT_ENTITY.

- Allow *CONTACT_AUTOMATIC_GENERAL to use MAXPAR from contact optional card A instead of using the hard coded value of 1.02. This will better detect end to end contact of beams. This applies to SMP only.
- Fix *CONTACT_TIED_SHELL_EDGE_TO_SURFACE for SMP which ignores MAXPAR in contact optional card A.
- Fix seg fault when using *CONTACT_GUIDED_CABLE.
- Fix segmentation fault when using *CONTACT_AUTOMATIC_SINGLE_-SURFACE_TIED in consistency mode, i.e. ncpu < 0 in command line, for SMP.
- Fix incorrect contacts when using *CONTACT_AUTOMATIC_GENERAL_-INTERIOR for beams with large differences in thickness and when the thinner beams are closer to each other than to the thicker beams. Affects SMP only.
- Fixed force transducers with MPP segment-based contact when segments are involved with multiple 2 surface force transducers. The symptom was that some forces were missed for contact between segments on different partitions.
- Fixed an MPP problem in segment-based contact that cased a divide by zero during the bucket sort. During an iteration of the bucket sort, all active segments were somehow in one plane which was far from the origin such that a dimension rounded to zero. The fix for this should affect only this rare case and have no effect on most models.
- Fixed thermal MPP segment-based contact. The message passing of thermal energy due to friction was being skipped unless peak force data was written to the intfor file.
- Fixed MPP segment based implicit contact. A flaw in data handling caused possible memory errors during a line search.
- Fixed implicit dynamic friction for segment-based contact. For sliding friction, the implicit stiffness was reduced to an infinitesimal value. Also, the viscous damping coefficient is now supported for implicit dynamic solutions.
- Fixed segment-based contact when the data has all deformable parts that are switched to rigid at the start of the calculation and then switched back to deformable prior to contact occurring. A flaw was causing contact to be too soft. This is now corrected.
- Fixed a flaw in segment-based contact with DEPTH = 25 that could allow penetration to occur.
- Improved edge-to-edge contact checking (DEPTH = 5, 25, 35) and the sliding option (SBOPT = 4, 5) in areas where bricks have eroded when using segment based eroding contact.
- Improved the initial penetration check (IGNORE = 2 on *CONTROL_CONTACT) of segment-based contact to eliminate false positives for shell segments. Previously, the search was done using mid-plane nodes and the gap or penetration adjusted to account for segment thicknesses after. The new way projects the nodes to the surface first and uses the projected surface to

measure penetration. For brick segments with zero thickness there should be no difference. For shell segments, the improved accuracy will me more noticeable for thicker segments.

- Improved segment-based contact when SHAREC = 1 to run faster when there are rigid bodies in the contact interface.
- Fixed a possible problem during initialization of segment-based contact. Options that use neighbor segment data such as the sliding option and edge-to-edge checking could access bad data if the same nodes were part of both the slave and master surfaces. This would not be a normal occurrence but could happen.
- Updated segment-based contact to improve accuracy at points away from the origin. The final calculations are now done with nodal and segment locations that have been shifted towards the origin so that coordinate values are small.
- The reporting of initial penetrations and periodic intersection reports by segment-based contact was corrected for MPP solutions which were reporting incorrect element numbers.
- Fixed memory errors in 2D automatic contact initialization when friction is used.
- Fixed 2D force transducers in the MPP version which could fail to report master surface forces. Also fixed 2 surface 2D force transducers when the smp parallel consistency option is active.
- Fixed *CONTACT_2D_AUTOMAITC_SINGLE_SURFACE and SURFACE_TO_SURFACE which could exhibit unpredictable behavior such as a force spike or penetration.
- Fixed a serious MPP error in the sliding option of *CONTACT_2D_AUTOMATIC that could lead to error termination.
- Fixed a problem with birth time for *CONTACT_2D_AUTOMATIC_TIED when used with sensor switching. Also, fixed a problem in the contact energy calculation that could lead to abnormal terminations. Finally, I made the process of searching for nodes to tie more robust as some problem was found with nodes being missed.
- Fixed a 2D automatic contact bug that occurred if a segment had zero length. An infinite thickness value was calculated by A/L causing the bucket sort to fail.
- Added support for *CONTACT_ADD_WEAR for smp and mpp segment based (SOFT = 2) contact. This option enables wear and sliding distance to be measured and output to the intfor file.
- Added support to segment-based contact for the SRNDE parameter on optional card E of *CONTACT.
- Added support to segment based eroding contact for SBOXID and MBOXID on card 1 of *CONTACT.
- Added support for *ELEMENT_SOURCE_SINK used with segment-based contact. With this update, inactive elements are no longer checked for contact.

- Added a segment-based contact option to allow the PSTIFF option on *CONTROL_CONTACT to be specified for individual contact definitions. The new parameter is PSTIFF on *CONTACT on optional card F, field 1. Prior to this change, setting PSTIFF on *CONTROL_CONTACT set all contact to use the alternate penalty stiffness method. With this update, PSTIFF on *CONTROL_CONTACT now sets a default value, and PSTIFF on card F can be used to override the default value for an individual contact interface.
- Added support for REGION option on optional card E of *CONTACT when using segment based, SOFT = 2 contact. This works for all supported keywords, SMP and MPP.
- Added master side output in the MPP version for 2-surface force transducers when used with segment based (soft = 2) contact.
- Added contact friction energy to the sleout database file for
    - 2D_AUTOMATIC_SURFACE_TO_SURFACE and
    - 2D_AUTOMATIC_SINGLE_SURFACE contact.
- Enabled segment-based contact (SMP and MPP) to work with type 24 (27-node) solid elements.
- Enabled the ICOR parameter on *CONTACT, optional card E to be used with segment based (SOFT = 2) contact.
- Fixed output to d3hsp for *CONTACT_DRAWBEAD using negative curve ID for LCIDRF
- Add slave node thickness and master segment thickness as input arguments to the *USER_INTERFACE_FRICTION subroutine usrfrc (SMP).
- Forming mortar contact can now run with deformable solid tools and honors ADPENE to account for curvatures and penetrations in adaptive step. This applies to h- as well as r-adaptivity.
- Single surface and surface-to-surface mortar contact accounts for rotational degrees of freedom when contact with beam elements. This allows for beams to "roll" on surfaces and prevents spurious friction energy to be generated when in contact with rotating parts.
- Maximum allowable penetration in forming and automatic mortar contacts is hereforth .5*(tslav+tmast)*factor where tmast = thickness of slave segment and tmast = thickness of master segment. The factor is hardwired to 0.95, but is subject to change. Prior to this it was .5*tslav, which seems inadequate (too small) in coping with initial penetrations in automotive applications using standard modeling approaches.
- Up to now, mortar contact has only acted between flat surfaces, now account is taken for sharp edges in solid elements (the angle must initially be larger than 60 degrees), may have to increase the corresponding stiffness in the future.
- When solid elements are involved in mortar contact the default stiffness is increased by a factor of 10. This is based on feedback from customers indicating that the contact behavior in those cases has in general been too soft.

This may change the convergence characteristics in implicit but the results should be an improvement from earlier versions.
- The OPTT parameter on *PART_CONTACT for the contact thickness of beams is now supported in mortar contact.
- *CONTACT_ADD_WEAR: A wear law, Archard's or a user defined, can be associated with a contact interface to assess wear in contact. By specifying WTYPE < 0 a user defined wear subroutine must be written to customize the wear law. For the Archard's wear law, parameters can depend on contact pressure, relative sliding velocity and temperature. Contacts supported are *CONTACT_FORMING_SURFACE_TO_SURFACE, *CONTACT_-FORMING_ONE_WAY_TO_SURFACE and *CONTACT_AUTOMATIC_-SURFACE_TO_SURFACE. To output wear data set NWEAR = 1 or NWEAR = 2 on *DATABASE_EXTENT_INTFOR. If NWEAR is set to 2 then the sliding distance is output to the intfor file, in addition to the wear depth. Otherwise only wear depth is output. Also, the parameter NWUSR specifies the number of user wear history variables to be output in case a user defined wear routine is used. By specifying CID (contact interface id) to a negative number, the wear depth will couple to the contact in the simulation in the sense that the penetration is reduced with wear. The effect is that contact pressure will be redistributed accordingly but is only valid for relatively small wear depths. A formulation for larger wear depths lie in the future which will require modification of the actual geometry.
- Fixed bug affecting *CONTACT_RIGID_NODE_SURFACE (broken at rev. 86847). The bug was in reading *NODE_RIGID_SURFACE.
- A bug fix in *CONTACT_DRAWBEAD_INITILIZE. - The bug was caused by a sudden increase in effective strain after the element passed the drawbead. When the increase in strain is too big, the search algorithm was not working reasonably in the material routine. At the drawbead intersection point, an element could be initialized twice by two bead curves, and cause abnormal thickness distribution.
- Fix a bug in *CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURFACE_SMOOTH which removes the limitation that the contact must be defined by segment set.
- SMOOTH option does not apply to FORMING_SURFACE_TO_SURFACE contact. When the SMOOTH option is used, we now write a warning message and disregard the SMOOTH option.

- ***CONSTRAINED**

  - *CONSTRAINED_LAGRANGE_IN_SOLID: Implement *CONSTRAINED_-LAGRANGE_IN_SOLID_EDGE in 2D.
  - Fixed bug in *DAMPING_RELATIVE. If the rigid part PIDRB is the slave part in *CONSTRAINED_RIGID_BODIES, the damping card did not work correctly. There is a work-around for previous LS-DYNA versions: set

PIDRB to the master part in *CONSTRAIEND_RIGID_BODIES, not the slave part.

- *CONSTRAINED_RIGID_BODY_INSERT: This keyword is for modeling die inserts. One rigid body, called slave rigid body, is constrained to move with another rigid body, called the master rigid body, in all directions except for one.
- A variety of enhancements for *CONSTRAINED_INTERPOLATION.
  - Enhanced the error message when nodes involved in the constraint have been deleted.
  - Removed printing of 0 node ID in MPP.
  - Added a warning if there are too many (now set at 1000) nonzeroes in a constraint row for *CONSTRAINED_INTERPOLATION or *CONSTRAINED_LINEAR to protect implicit's constraint processing. These constraints will be processed differently in future releases. We modified the constraint processing software to robustly handle constraint rows with thousands of nonzero entries. We added error checking for co-linear independent nodes as these constraints allow singularities in the model.
- Improved implicit's treatment of the constraints for *CONSTRAINED_BEAM_IN_SOLID.
- Added error checking on the values of the gear ratios in *CONSTRAINED_JOINTS.
- *CONSTRAINED_BEAM_IN_SOLID:
  - Thick shell elements supported.
  - Wedge elements supported.
  - Debonding law by user-defined subroutine (set variable AXFOR > 1000).
  - Debonding law by *DEFINE_FUNCTION (set variable AXFOR < 0).
- Error terminate with message, SOL+700, if CIDA and CIDB is not defined for *CONSTRAINED_JOINT_STIFFNESS_GENERALIZED.
- Fix incorrect constraints on rotary dof for adaptivity.
- Fix incorrect motion if NRBF = 2 in *DEFORMABLE_TO_RIGID_AUTOMATIC and if any of the *CONSTRAINED_NODAL_RIGID_BODY nodes belongs to a solid element.
- Fix input error when using large load curve ID for FMPH, FMT, FMPS in card 3 of *CONSTRAINED_JOINT_STIFFNESS with GENERALIZED or TRANSLATIONAL options.
- Fix seg fault if using tables for FMPH of *CONSTRAINED_JOINT_STIFFNESS and if the angle of rotation is less than the abscissa of the table or load curves.

- Fixed a problem with *CONSTRAINED_BEAM_IN_SOLID when used in a model that also uses segment based eroding contact in the MPP version. This combination now works.
- Improved the precision of spot weld constraints (*CONSTRAINED_SPOTWELD) to prevent possible divide by zeroes when the inertia tensor is inverted. This affects the single precision version only.
- Fix for damage function in *CONSTRAINED_INTERPOLATION_SPOTWELD, MODEL = 2.
- Add some user-friendly output (rigid body ID) to d3hsp for *CONSTRAINED_NODAL_RIGID_BODY_INERTIA.
- Add new option to *CONSTRAINED_SPR2 to connect up to 6 shell element parts (metal sheets) with only one rivet location node. This is invoked by defining extra part IDs for such a multi-sheet connection.
- Add more flexibility to *CONSTRAINED_SPR2: Load curve function exponent values originally hardwired as "8" can now be defined with new input parameters EXPN and EXPT.
- Fixed bug wherein the joint ID in *CONSTRAINED_JOINT_COOR was read incorrectly.
- Fixed duplicate ID for *CONSTRAINED_SPOTWELD, ..._NODE_SET, _POINTS and _SPR2.
- Fix keyword reader for SPR4 option in *CONSTRAINED_INTERPOLATION_SPOTWELD, where BETA2 was replaced by BETA3.
- Significantly reduce the memory demand in the initialization stage of *CONSTRAINED_MULTIPLE_GLOBAL for implicit analysis.
- The unit cell mesh and constraint generated by *INCLUDE_UNITCELL now supports job ID.

- ***CONTROL**

  - Terminate and print error KEY+1117 for cases that use *INCLUDE_TRANSFORM in 3d r-adaptvity. More work is needed to make this combination work.
  - Changed SOL+41 message ("reached minimum step") from an error to a warning and terminate normally. This message is triggered when the DTMIN criterion set in *CONTROL_TERMINATION is reached.
  - Fixed bug in which h-adaptivity missed some ADPFREQ-based adaptations when IREFLG < 0 (*CONTROL_ADAPTIVE).
  - Fixed bug: MS1ST in *CONTROL_TIMESTEP causes non-physical large mass and inertia on Nodal Rigid Bodies if Dynamic Relaxation is active. The error occurs at the start of the transient solution. The mass can become very large, so the model may appear to be over-restrained.
  - Add new input check for curves. After rediscretizing curves, check to see how well the original values can be reproduced. If the match is poor, write out See variable CDETOL in *CONTROL_OUTPUT.

- Added the ability to specify unique values LCINT for each curve, which override the value set in *CONTROL_SOLUTION. Note: the largest value of LCINT that appears will be used when allocating memory for each load curve, so a single large value can cause significant increases in the memory required for solution.
- The DELFR flag in *CONTROL_SHELL has new options for controlling the deletion of shell elements. This feature is aimed at eliminating single, detached elements and/or elements hanging on by one shared node.
- Fix spurious deletion of elements when using TSMIN.ne.0.0 in *CONTROL_TERMINATION, ERODE = 1 in *CONTROL_TIMESTEP and initialized implicitly in dynamic relaxation.
- Fix spurious error, STR+755, if using *DAMPING_FREQUENCY_RANGE with *CONTROL_ADAPTIVE.
- Add new feature to *CONTROL_SOLUTION, LCACC, to truncate load curve to 6 significant figures for single precision & 13 significant figures for double precision. The truncation is done after applying the offset and scale factors.
- Fix "*** termination due to mass increase ***' error when using mass scaling with *ELEMENT_MASS_PART.
- Fix input error 'node set for nodal rigid body # not found' when using *PART_INERTIA with *CONTROL_SUBCYCLE.
- Fixed the negative DT2MS option on *CONTROL_TIMESTEP for thick shell types 5, 6, and 7.
- Fixed bug in *CONTROL_CHECK_SHELL if PSID.lt.0 (part set ID) is used
- Add new option NORBIC to *CONTROL_RIGID to bypass the check of rigid body inertia tensors being too small.
- Add new option ICRQ to *CONTROL_SHELL for continuous treatment of thickness and plastic strain across element edges for shell element types 2, 4, and 16 with max. 9 integration points through the thickness.
- Add new option ICOHED to *CONTROL_SOLID. If this value is set to 1, solid cohesive elements (ELFORM 19-22) will be eroded when neighboring (nodewise connected) shell or solid elements fail.
- Beam release conditions are now properly supported in selective mass scaling, see IMSCL on *CONTROL_TIMESTEP.
- Modified MSGMAX in *CONTROL_OUPUT: MSGMAX is the maximum number of each error/warning message

    - > 0 number of message to screen output, all messages written to messag/d3hsp
    - < 0 number of messages to screen output and messag/d3hsp
    - = 0 the default is 50

- Fix bugs in 3D solid adaptivity (*CONTROL_ADAPTIVE, ADPTYP = 7) so that the solid adaptivity will still work when there are any of the following in the model:

    - thick shells (*SECTION_TSHELL),
    - massless nodes,
    - *LOAD_SEGMENT_*{OPTION}*.

  - Added PARA = 2 to *CONTROL_PARALLEL which activates consistent force assembly in parallel for SMP. An efficient parallel algorithm is implemented for better performance when the consistency flag is turned on. It shows better scaling with more cpus. This option is overridden by parameter "para=" on the execution line.

- **DEM (Discrete Element Method)**

  - Added output of following DES history variables to d3plot:

    - nodal stress and force
    - pressure
    - density
    - force chain
    - damage calculation when *DEFINE_DE_BOND is defined

  - Added output of following DES history variables to demtrh (*DATABASE_-TRACER_DE):

    - coordination number
    - porosity and void ratio
    - stress
    - pressure
    - density

  - Output ASCII format for demrcf if BINARY.eq.3.
  - Implement gauss distribution of DE sphere radius for *DEFINE_DE_INJECTION. The mean radius is 0.5 × (rmin + rmax) and standard deviation is 0.5 × (rmax - rmin).
  - For DE sphere, implement the stress calculation for REV (Representative Elementary Volume) using *DATABASE_TRACER_DE and specific RADIUS.
  - Add *BOUNDARY_DE_NON_REFLECTING for defining non-reflecting boundary conditions for DE spheres.
  - For *CONTROL_DISCRETE_ELEMENT, add the option to create the liquid bridge if the initial distance between two DE spheres is smaller than predefined gap.
  - Added *DATABASE_DEMASSFLOW, see *DEFINE_DE_MASSFLOW_-PLANE, for measuring the mass flow of DE spheres through a surface. The surface is defined by part or part set. Output file is 'demflow'.
  - Add *DEFINE_DE_INJECTION_ELLIPSE, to define a circular or elliptical injection plane.

- Add *DEFINE_PBLAST_AIRGEO for *PARTICLE_BLAST which defines initial geometry for air particles.
- Add DEM stress calculation when coupling with segment (*DEFINE_DE_TO_SURFACE_COUPLING).
- Fix error in demtrh file output (Windows platform only).

• **EFG (Element Free Galerkin)**

- Fix bug for ELFORM = 41 implicit when there are 6-noded/4-noded elements.

• ***ELEMENT**

- Fix a 2d seatbelt bug triggered by having both 1d and 2d seatbelts, and a 1d pretensioner of type 2, 3 or 9.
- Fix MPP bug initializing multiscale spot weld in the unexpected case where the spot weld beam is merged with the shells rather than tied via contact.
- Fix bug for *INCLUDE_UNITCELL.
- *CONTROL_REFINE_...: Implement the parent-children transition in *CONTACT_2D_SINGLE_SURFACE when a shell refinement occurs.
- Fix error traps for *ELEMENT_SEATBELT_..., for example, error termination due to convergence failure in retractors. These error traps worked but could lead to a less graceful termination than other LS-DYNA error traps.
- Correct calculation of wrap angle in seatbelt retractor.
- Add MPP support for *ELEMENT_LANCING.
- *ELEMENT_SEATBELT:
  - Fix an MPP belt bug that can happen when buckle pretensioner is modeled as a type-9 pretensioner.
  - 2D belt and 1D belt now can share the same *MAT_SEATBELT.
  - The section force for 2d belt is recoded to provide more robust and accurate results.
  - The loading curve LLCID of *MAT_SEATBELT can be a table defining strain-rate dependent stiffness curve.
  - IGRAV of *ELEMENT_SEATBELT_ACCELEROMETER can be a curve defining gravitation flag as a function of time.
- Add *NODE_THICKNESS to override shell nodal thickness otherwise determined via *SECTION_SHELL, *PART_COMPOSITE, or *ELEMENT_SHELL_THICKNESS.
- Fix input error when using *DEFINE_ELEMENT_DEATH with BOXID > 0 for MPP.
- Implement subcycling for thick shells.
- Fix ineffective *DEFINE_HAZ_PROPERTIES when solid spotwelds and hex spotweld assemblies are both present.

- Fix incorrect beta written out for *ELEMENT_SHELL_BETA in dynain file when *PART_COMPOSITE keyword is present in the original input.
- Fix NaN output to elout_det and spurious element deletion if NODOUT = STRAIN or STRAIN_GL or ALL or ALL_GL.
- Fix incorrect reading of TIME in card 3 of *ELEMENT_SEATBELT_SENSOR SBSTYP = 3 when long = s in command line.
- *PART_COMPOSITE: Increased the explicit solution time step for thin shell composite elements. The existing method was overly conservative. The new method is based on average layer stiffness and density.
- In conjunction with the above change in composite time step calculation, increase nodal inertia in the rare cases of *PART_COMPOSITE in which the bending stability is not satisfied by the membrane stability criterion. The inertia is only increased in the cases where it is necessary; for most models this change has no effect, but this can occur in the case of sandwich sections with stiffer skins around a less stiff core.
- Corrected rotational inertia of thin shells when layers have mixed density and the outer layers are more dense than inner layers. The fix will mostly affect elements that are very thick relative to edge length.
- Fixed default hourglass control when the *HOURGLASS control card is used but no HG type is specified. We were setting to type 1 instead of 2. Also, fixed the default HG types to match the User's Manual for implicit and explicit.
- Fixed the part mass that was reported to d3hsp when *ELEMENT_SHELL_SOURCE_SINK is used. The inactive elements were being included causing too high mass.
- Prevent inactive shell elements (from *ELEMENT_SHELL_SOURCE_SINK) from controlling the solution time step.
- Fixed the reported strain tensor in elements created by *ELEMENT_SHELL_SOURCE_SINK when strain output is requested. The history was being retained from the previous elements with the same ID.
- Fixed torsion in linear beam form 13. A failure to add the torsional moment at node 2 caused an inability to reach equilibrium in the torsional mode.
- Fixed solid element 4 so that rigid body translation will not cause strain and stress due to round-off error.
- Mixed parallel consistency when used with solid element type 20. A buffer was not being allocated leading to a memory error.
- Changed the MPP behavior of discrete beams (ELFORM = 6) when attached to elements that fail. They were behaving like null beams, in the sense that it was possible for beam nodes to become dead due to attached elements failing, and discrete beams would be no longer visualized even if the beams themselves had not failed. With this change, the MPP discrete beams now behave like other beams in that the beams have to fail before they are removed. MPP and SMP behavior is now consistent.
- Improved the precision of the type 2 Belytschko Schwer resultant beam to prevent energy growth in single precision.

- Fixed the NLOC option on *SECTION_SHELL for the BCIZ triangle elements (ELFORM = 3) and the DKT triangle elements (ELFORM = 17). The offset was scaled by the solution time step so typically the offset was much smaller than expected.
- Fixed elout stress output for shell element forms 23 and 24. The in-plane averaging was incorrect causing wrong output.
- Changed *ELEMENT_TSHELL so that both the COMPOSITE and BETA options can be read at the same time. Prior to the fix, only the first one would be read.
- Fixed all thick shells to work with anisotropic thermal strains which can be defined by *MAT_ADD_THERMAL. Also, this now works by layer for layered composites.
- Fixed implicit solutions with thick shells with *MAT_057 when there are also solid elements in the model that use *MAT_057. Thick shells support only the incremental update of the F tensor but a flag was set incorrectly in the material model.
- Fixed *MAT_219 when used with thick shell types 3, 5, and 7. A failure to initialize terms for the time step caused a possible wrong time step.
- Fixed orthotropic user defined materials when used with thick shell elements. The storing of the transformation matrix was in the wrong location leading to wrong stress and strain.
- For thick shell composites that use element forms 5 and 7, the user can now use laminated shell theory along with the TSHEAR = 1 on *SECTION_TSHELL to get a constant shear stress through the thickness with a composite.
- Fixed the initialization of *MAT_CODAM2/*MAT_219 when used with thick shell forms 3, 5, or 7. The 3D thick shell routine uses only 2 terms for the transformation and therefore needs unique initialization of the transformation data.
- Fixed thick shell types 3 and 5 when used in implicit solutions with *MATs 2, 21, 261, and 263. The material constitutive matrix for *MATs 2 and 21 was not rotated correctly causing wrong element stiffness. The constitutive matrix for *MATs 261 and 263 was not orthotropic. Also, for *MAT_021, type 5 thick shell needed some material terms defined to correct the assumed strain.
- Fixed thick shell forms 3 and 5 when used in implicit solutions with nonisotropic materials. The stiffness matrix was wrong due to incorrect transformations.
- Also, fixed the implicit stiffness of thin and thick shells when used with laminated shell theory by assumed strain (LAMSHT = 3,4,5 on *CONTROL_ACCURACY). Elements were either failing to converge or converging more slowly due to the failure to adjust the stiffness matrix to be consistent with the assumed strain.
- Added support for *ELEMENT_SHELL_SOURCE_SINK to form 2 elements with BWC = 1 on *CONTROL_SHELL.

- Fixed the s-axis and t-axis orientation of beam spot welds in the MPP version when those beam weld elements are defined with a 3rd node. The 3rd node was being discarded prior to initializing the beam orientation so the s and t-axes were being randomly assigned as if the 3rd node had not been assigned. The effect on solutions is likely fairly minimal since beam material is isotropic and failure typically is too but may not be.
- Added Rayleigh damping (*DAMPING_PART_STIFFNESS) for thick shell formulations 1, 2, and 6. Previously, it was available for only the thick shells that call 3D stress updates, (forms 3 and 5), but now it is available for all thick shell formulations.
- Added new SCOOR options for discrete beam section 6 (*SECTION_BEAM). A flaw was found in how the discrete beam accounts for rigid body rotation when SCOOR = -3, -2, +2, and +3. A correction for this is made and introduced as new options, SCOOR = -13, -12, +12, and +13. A decision was made to leave the existing options SCOOR = -2, +2, -3 and +3 unchanged so that legacy data could run without changes.
- Enabled the ELFORM 18 linear DKT shell element to work with *PART_COMPOSITE and with an arbitrary number of through thickness integration points. It was limited to a single material and 10 Gauss points.
- Added the possibility to write *ELEMENT_SOLID_ORTHO into dynain file if requested. To activate this, add OPTCARD to *INTERFACE_SPRINGBACK and set SLDO = 1.
- Refine characteristic length calculation for 27-node solid (ELFORM 24). This change may increase the time step substantially for badly distorted elements.
- Implement selective reduced integration for 27-node solid (ELFORM 24).
- Allow part sets to be used in *DEFORMABLE_TO_RIGID_AUTOMATIC. Either PID is defined negative or "PSET" is set in column 3 (D2R) or 2 (R2D).
- Add new option STRESS = 2 to *INCLUDE_STAMPED_PART: no stresses and no history variables are mapped with that setting.
- New keyword *PART_STACKED_ELEMENTS provides a method to define and to discretize layered shell-like structures by an arbitrary sequence of shell and/or solid elements over the thickness.
- The geometric stiffness matrix for the Belytschko beam element type 2 has been extended to include nonsymmetric terms arising from nonzero moments. Provides "almost" quadratic convergence, still some terms missing to be added in the future. Also support a strongly objective version activated by IACC on *CONTROL_ACCURACY.
- The geometric stiffness for the Hughes-Liu element type 1 is fixed.
- Fix parsing error in *SECTION_BEAM_AISC.

- **EM (Electromagnetic Solver)**

  - Add the new EM 2d axi solver in SMP and MPP for EM solver 1 (eddy current). It is coupled with the mechanics and thermal solvers.

- The new EM 2d can be used with RLC circuits on helix/spiral geometries using *EM_CIRCUIT_CONNECT.
- Add EM contact into new EM 2d axi, in SMP and MPP.
- Add *EM_BOUNDARY support in new EM 2d axi solver.
- Introduce scalar potential in new EM 2d axi. The 2d axi can also be coupled with imposed voltage.
- Add new keyword *EM_CIRCUIT_CONNECT to impose linear constraints between circuits with imposed currents in 3d solvers. This allows for example to impose that the current in circuit 1 is equal to the current in circuit 2 even if the 2 corresponding parts are not physically connected.
- Add *EM_VOLTAGE_DROP keyword to define a voltage drop between 2 segment sets. This voltage drop constraint is coupled to the contact constraint so that the contact (voltage drop = 0) has priority over the *EM_-VOLTAGE_DROP constraint.
- Add *EM_CONTROL_SWITCH_CONTACT keyword to turn the EM contact detection on and off.
- NCYCLBEM/NCYCLFEM in *EM_SOLVER_... can now be different than 1 when EM_CONTACT detected.
- Add RLC circuit for type 3 solver (resistive heating).
- Add computation of mutuals/inductances in 2d axi for output to em_circuit.dat
- Add criteria on autotimestep calculation when RLC circuit used to take into account RLC period.
- Fix keyword counter in d3hsp.
- Better and clearer output to terminal screen.
- Support jobid for EM ascii file outputs.

- **Forming**

  - Improvements to trimming:

    - *DEFINE_CURVE_TRIM_NEW: if trim seed node is not defined, we will search a seed node based on nodes from the sheet blank and the inside/outside flag definition for the trimming curves.
    - Map strain tensors to triangular elements after trimming.

  - Add a new function to the trim of solid elements in normal (3-D) trimming case, related to *DEFINE_CURVE_TRIM_3D. If the trimming curve is close to the bottom side, set TDIR = -1. If the trimming curve is close to the upper side, set TDIR = 1.
  - Add to *ELEMENT_LANCING. Allow parametric expression for variables END and NTIMES.
  - A bug fix for *CONTROL_FORMING_AUTOPOSITION_PARAMETER_-SET: Fix distance calculation error when the target mesh is too coarse.
  - Improvements to springback compensations:

- Output the new trimming curve with *DEFINE_CURVE_TRIM_3D (previously *DEFINE_CURVE_TRIM), so that it can be easily converted to IGES curve by LS-PrePost. or used in another trimming calculation.
- Output each curve to IGES format in the following name format: newcurve_scp001.igs, newcurve_scp002.igs, newcurve_scp003.igs, etc.
- Output change in file "geocur.trm". This update will allow change from *DEFINE_CURVE_TRIM(_3D, _NEW), whatever is used for input.

- Add a new keyword: *DEFINE_FORMING_CONTACT to facilitate the forming contact definitions.
- Add a new keyword *DEFINE_FORMING_CLAMP, to facilitate clamping simulation.
- A new feature in mesh fusion, which allows a moving box to control the fusion, only if the center of the elements is inside the box can the elements be coarsened. Can be used in conjunction with *DEFINE_BOX_ADAPTIVE.
- Add a new feature to *DEFINE_BOX_ADAPTIVE: Moving box in adaptivity, useful in roller hemming and incremental forming.
- In mesh coarsening, if the node is defined in a node set, the connected elements will be kept from being coarsened. Previously, only *SET_NODE_-LIST was supported. Now option *SET_NODE_GENERAL is allowed.
- Add a new function: mesh refinement for sandwich part. The top and bottom layers are shell elements and the middle layer is solid elements. Set IFSAND to 1 in *CONTROL_ADAPTIVE.

  - Applies to both 8-noded and 6-noded solid elements.
  - Map stress and history variables to the new elements.

- New features related to blank size development *INTERFACE_BLANKSIZE_DEVELOPMENT:

  - Add *INTERFACE_BLANKSIZE_SYMMETRIC_PLANE to define symmetric plane in blank size development
  - Add *INTERFACE_BLANKSIZE_SCALE_FACTOR. For each trimming, different scale factors can be used to compensate the blanksize. This is especially useful when the inner holes are small. Includes an option of offset the target curve which is useful if multiple target curves (e.g., holes) and formed curves are far from each other.
  - Allow target curve to be outside of the surface of the blank.
  - Add sorting to the mesh so the initial mesh and the formed mesh do not need to have the same sequence for the nodes.
  - Add a new variable ORIENT, set to "1" to activate the new algorithm to potentially reduce the number of iterations with the use of *INTERFACE_BLANKSIZE_SCALE_FACTOR (scale = 0.75 to 0.9).

    - Fix smooth problem along calculated outer boundary.
    - Automatically determine the curve running directions (IOPTION = 2 and -2 now both give the same results).
    - Accept parameteric expression.

- A bug fix for springback compensation: *INCLUDE_COMPENSATION_-SYMMETRIC_LINES Fix reading problem of free format in the original coding.
- Add a new keyword *CONTROL_FORMING_BESTFIT. Purpose: This keyword rigidly moves two parts so that they maximally coincide. This feature can be used in sheet metal forming to translate and rotate a spring back part (source) to a scanned part (target) to assess spring back prediction accuracy. This keyword applies to shell elements only.
- Improvements to *CONTROL_FORMING_AUTOCHECK:

    - When IOFFSET = 1, rigid body thickness is automatically offset, based on the MST value defined in *CONTACT_FORMING_ONE_WAY_-SURFACE.
    - Add new variable IOUTPUT that when set to 1 will output the offset rigid tool mesh, and the new output tool file is: rigid_offset.inc. After output the simulation stops. See R9.0 Manual for further details.
    - When both normal check and offset are used, small radius might cause problem for offsetting. The new modification will check the normal again after offsetting the tool
    - When outputting the rigid body mesh, output the bead nodes also.
    - Changes to *CONTROL_FORMING_AUTOCHECK when used together with SMOOTH option: check and fix rigid body bad elements before converting the master part ID to segment set id to be used by SMOOTH option.
    - Set IOUTPUT.eq. 3 to output rigid body mesh before and after offset.
    - Fix problems offseting a small radius to a even smaller radius.
    - Remove T-intersection.

- For *CONTROL_IMPLICIT_FORMING, fix output messages in d3hsp that incorrectly identified steps as implicit dynamic when they were actually implicit static.
- Improve *CONTROL_FORMING_UNFLANGING:

    - Automatically calculate CHARLEN, so user does not need to input it anymore.
    - Allow nonsmooth flange edge.
    - Instead of using preset value of 0.4 (which works fine for thin sheet metal), blank thickness is now used to offset the slave node (flanges) from the rigid body (die).

- ***FREQUENCY_DOMAIN**
  - *FREQUENCY_DOMAIN_RANDOM_VIBRATION: Fixed a bug in dumping d3psd binary database, when both stress and strain are included.
  - *FREQUENCY_DOMAIN_SSD_ERP: Implemented the Equivalent adiated Power (ERP) computation to MPP.
  - *FREQUENCY_DOMAIN_ACOUSTIC_BEM:
    - Enabled running dual collocation BEM based on Burton-Miller formulation (METHOD = 4) with vibration boundary conditions provided by Steady State Dynamic analysis (*FREQUENCY_DOMAIN_SSD).
    - Added exponential window function for FFT (FFTWIN = 5).
    - Implemented a new forward and backward mixed radix FFT.
    - Implemented acoustic computation restart from frequency domain boundary conditions, in addition to time domain boundary conditions (RESTRT = 1).
    - Enabled out-of-core velocity data storage, to solve large scale problems.
    - Implemented option HALF_SPACE to Rayleigh method (METHOD = 0) to consider acoustic wave reflection.
    - Added velocity interpolation to take care of mismatching between acoustic mesh and structural mesh (*BOUNDARY_ACOUSTIC_MAPPING), for the case that the boundary conditions are provided by Steady State Dynamic analysis.
    - Added weighted SPL output to acoustic computation (DBA = 1,2,3,4).
    - Implemented radiated sound power, and radiation efficiency computation to collocation BEMs (METHOD = 3,4). Added new ASCII xyplot databases Press_Power and Press_radef to save the sound power and radiation efficiency results.
    - Enabled using both impedance and vibration (velocity) boundary conditions in acoustic simulation.
  - *FREQUENCY_DOMAIN_ACOUSTIC_FEM:
    - Added weighted SPL output to FEM acoustics (DBA = 1,2,3,4).
    - Implemented option EIGENVALUE to perform acoustic eigenvalue analysis; added ASCII database eigout_ac to save acoustic eigenvalue results; added binary plot database d3eigv_ac to save acoustic eigenvectors.
    - Enabled consideration of nodal constraints in acoustic eigenvalue analysis.
    - Enabled FEM acoustic analysis with frequency dependent complex sound speed.
    - Implemented pressure and impedance boundary conditions.

- *FREQUENCY_DOMAIN_ACOUSTIC_FRINGE_PLOT:
  - Added this keyword to 1) generate acoustic field points as a sphere or plate mesh (options SPHERE and PLATE), or 2) define acoustic field points mesh based on existing structure components (options PART, PART_SET and NODE_SET) so that user can get fringe plot of acoustic pressure and SPL. The results are saved in binary plot database d3acs (activated by keyword *DATABASE_FREQUENCY_BINARY_-D3ACS).
- *FREQUENCY_DOMAIN_RANDOM_VIBRATION:
  - Changed displacement rms output in d3rms to be the displacement itself, without adding the original nodal coordinates.
  - Implemented von mises stress PSD computation in beam elements.
  - Implemented fatigue analysis with beam elements.
  - Added strain output to binary plot databases d3psd and d3rms, and binout database elout_psd.
  - Added initial damage ratio from multiple loading cases (INFTG > 1).
- *FREQUENCY_DOMAIN_SSD:
  - Implemented option ERP to compute Equivalent Radiated Power. It is a fast and simplified way to characterize acoustic behavior of vibrating structures. The results are saved in binary plot database d3erp (activated by keyword *DATABASE_FREQUENCY_BINARY_D3ERP), and ASCII xyplot files ERP_abs and ERP_dB.
  - Implemented fatigue analysis based on maximum principal stress and maximum shear stress.

- **ICFD (Incompressible Flow Solver)**
  - *ICFD_BOUNDARY_FSWAVE: Added a boundary condition for wave generation of 1st order stokes waves with free surfaces.
  - *ICFD_DATABASE_DRAG_VOL: For computing pressure forces on volumes ID (useful for forces in porous domains), output in icfdragivol.dat and icfdragivol.#VID.dat.
  - *ICFD_CONTROL_DEM_COUPLING: Coupling the ICFD solver with DEM particles is now possible.
  - *ICFD_CONTROL_MONOLITHIC: Added a monolithic solver (=1) which can be selected instead of the traditional fractional step solver (=0).
  - *ICFD_CONTROL_POROUS: This keyword allows the user to choose between the Anisotropic Generalized Navier-Stokes model (=0) or the Anisotropic Darcy-Forchheimer model (=1) (for Low Reynolds number flows). The Monolithic solver is used by default for those creeping flows.
  - *ICFD_CONTROL_TURBULENCE:

- Modified existing standard k-epsilon.
- Added Realizable k-epsilon turbulence model.
- Added Standard 98 and 06 Wilcox and Menter SST 03 turbulence models.
- Added Several laws of the wall.
- Added Rugosity law when RANS turbulence model selected.

◦ *ICFD_MODEL_POROUS:

- Added Porous model 5 for anistropic materials defined by P-V experimental curves.
- Added porous model 6 for moving domain capabilities for Porous Media volumes using load curves for permeabilities directions.
- Added porous model 7 for moving domain capabilities for Porous Media volumes using ICFD_DEFINE_POINT for permeabilities directions.
- Added porous model 9 for a new Anisotropic Porous Media flow model (PM model ID = 9): It uses a variable permeability tensor field which is the result of solid dynamic problems. The model reads the solid mesh and the field state and maps elemental permeability tensor and solid displacements to the fluid mesh.

◦ *ICFD_MODEL_NONNEWT:

- Added a few models for non newtonian materials and temperature dependant viscosity :
  - model 1 : power law non newtonian (now also temperature dependant)
  - model 2 : carreau fluid
  - model 3 : cross fluid
  - model 4 : herschel-bulkley
  - model 5 : cross fluid II
  - model 6 : temperature dependant visc (sutherland)
  - model 7 : temperature dependant visc (power law)
  - model 8 : load curve dependant visc, model 8 is especially interesting since a DEFINE_FUNCTION can be used (for solidification applications).

◦ *ICFD_SOLVER_TOL_MONOLITHIC: Used to define atol, rtol, dtol and maxits linear solver convergence controls of the monolithic NS time integration

◦ *MESH_BL: Added support for boundary layer mesh creation by specifying the thickness, number of layers, first node near the surface and the strategy to use to divide and separate the elements inside the BL adding.

- *ICFD_BOUNDARY_PRESCRIBED_VEL: Added the support of DEFINE_-FUNCTION making the second line of the keyword obsolete.
- *ICFD_CONTROL_TIME: Min and Max timestep values can be set.
- *ICFD_DATABASE_DRAG:
  - Added frequency output.
  - Added option to output drag repartition percentage in the d3plots as a surface variable.
- *ICFD_CONTROL_IMPOSED_MOVE: This keyword now uses *ICFD_-PART and *ICFD_PART_VOL instead of *MESH_VOL for ID. It is now possible to impose a rotation on a part using Euler angles.
- *ICFD_CONTROL_OUTPUT:
  - Field 4 now to output mesh in LSPP format and in format to be run by the icfd solver (icfd_fluidmesh.key and icfd_mesh.key)
  - icfd_mesh.key now divides the mesh in ten parts, from best quality element decile to worst.
  - A new mesh is now output at every remeshing.
  - Added support for parallel I/O for Paraview using the PVTU format.
- *ICFD_DEFINE_POINT: Points can now be made to rotate or translate.
- *ICFD_MAT:
  - Nonnewtonian models and Porous media models are now selected in the third line by using the new ICFD_MODEL keyword family.
  - HC and TC can now be made temperature dependent.
- *ICFD_CONTROL_DIVCLEAN: Added option 2 to use a potential flow solver to initialize the Navier Stokes solver.
- *ICFD_CONTROL_FSI: Field 5 provides a relaxation that starts after the birthtime.
- *ICFD_CONTROL_MESH: Field 3 added a new strategy to interpolate a mesh size during the node insertion. In some cases it speeds up the meshing process and produces less elements. Field 4 changes the meshing strategy in 2d.
- *ICFD_CONTROL_SURFMESH: Added support for dynamic re-meshing/adaptation of surface meshing.
- *ICFD_BOUNDARY_PRESCRIBED_VEL:VAD = 3 now works with DOF = 4.
- SF can be lower than 0.
- PID can be over 9999 in *ICFD_DATABASE_FLUX.
- Fixed d3hsp keyword counter.
- Clarified terminal output.
- Y+ and Shear now always output on walls rather than when a turbulence model was selected.

- Added coordinate of distorted element before remeshing occurs. Output on terminal and messag file
- Fixed bug in conjugate heat transfer cases. When an autotimestep was selected in *ICFD_CONTROL_TIME, it would always only take the thermal timestep.
- An estimation of the CFL number is now output in the d3plot files. This is not the value used for the autotimestep calculation.
- Turbulence intensity is now output in the d3plots.
- Jobid now supported for ICFD ASCII File outputs.
- Fixed communication of turbulent constants in MPP.
- Fixed the Near Velocity field output.
- Increasing the limit of number of parts for the model.
- Temperature added as a surface variable in output.
- Fixed non-linear conjugate heat solver.

- **Implicit**

  - Fixed Implicit for the case of Multi-step Linear (*CONTROL_IMPLICIT_GENERAL with NSOLVR = 1) with Intermittent Eigenvalue Computation (*CONTROL_IMPLICIT_EIGENVALUE with NEIG < 0).
  - Recent fix for resultant forces for Multi-step Linear cause segmentation fault when Intermittent Eigenvalue Computation was also active.
  - Fix possible issue related to constrained contacts in MPP implicit not initializing properly.
  - Fixed label at beginning of implicit step to be correct for the case of controlling implicit dynamics via a load curve (*CONTROL_IMPLICIT_DYNAMICS).
  - Corrected the computation of modal stresses with local coordinate terms and for some shell elements (see MSTRES on *CONTROL_IMPLICIT_EIGENVALUE).
  - Corrected *CONTROL_IMPLICIT_INERTIA_RELIEF logic in MPP. In some cases the rigid body modes were lost.
  - Enhanced implicit's treatment of failing spotwelds (*CONSTRAINED_SPOTWELD).
  - Added additional error checking of input data for *CONTROL_IMPLICIT_MODAL_DYNAMICS_DAMPING.
  - Per user request we added the coupling of prescribed motion constraints for Modal Dynamics by using constraint modes. See *CONTROL_IMPLICIT_MODAL_DYNAMIC.
  - Added reuse of the matrix reordering for MPP implicit execution. This will reduce the symbolic processing time which is noticeable when using large numbers of MPP processes. Also added prediction of non-tied contact connections for standard contact and mortar contact. This allows reuse of the ordering when contact interfaces are changing very slightly but can increase the cost of the numerical factorization. Useful only for MPP using large

numbers of processes for large finite element models. This reuse checking happens automatically for MPP and is not required for SMP.

- Apply improvements to Metis memory requirements used in Implicit MPP.
- Enhanced Metis ordering software (ORDER = 2, the default, on *CONTROL_IMPLICIT_SOLVER).
- Added new keyword *CONTROL_IMPLICIT_ORDERING to control of features of the ordering methods for the linear algebra solver in MPP Implicit. Only should be used by expert users.
- The following 4 enhancements are applicable when IMFLAG > 1 on *CONTROL_IMPLICIT_GENERAL.

    - Implicit was modified to reset the time step used in contact when switching from implicit to explicit.
    - Adjusted implicit mechanical time step for the case of switching from explicit to implicit so as not to go past the end time.
    - Explicit with intermittent eigenvalue analysis was getting incorrect results after the eigenvalue analysis because an incorrect time step was used for the implicit computations. For this scenario implicit now uses the explicit time step.
    - The implicit time step is now reset for the dump file in addition to explicit's time.

- Implicit's treatment of prescribed motion constraints defined by a box had to be enhanced to properly handle potential switching to explicit.
- The following 6 enhancements are for matrix dumping (MTXDMP > 0 on *CONTROL_IMPLICIT_SOLVER) or for frequency response (*FREQUENCY_DOMAIN) computations.
- Corrected the collection of *DAMPING_PART_STIFFNESS terms for elements like triangles and 5, 6, and 7 node solid elements.
- Corrected Implicit's access of *DAMPING_PART_STIFFNES parameter when triangle and tet sorting is activated.
- Fixed Implicit's collecting of damping terms for beams that have reference nodes.
- There is an internal switch that turns off damping for beams if the run is implicit static. This switch needed to be turned off for explicit with intermittent eigenvalue analysis.
- Fixed collecting of stiffness damping terms for implicit. Corrected the loading of mass damping terms when collecting damping terms for post processing.
- Extend matrix dumping to include dumping the solution vector in addition to the matrix and right-hand-side.
- Adjusted Implicit's handling of sw1. and sw3. sense switches to properly handle dumping. If sw1. sense switch is issued when not at equilibrium, then reset time and geometry to that at the end of last implicit time step. If

sw3. sense switch is issued, then wait until equilibrium is reached before dumping and continuing.

- Enable the use of intermittent eigenvalue computation for models using inertia relief and/or rotational dynamics. See NEIG < 0 on *CONTROL_IMPLICIT_EIGENVALUE and *CONTROL_INERTIA_RELIEF and *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS. Due to round-off, an implicit intermittent eigenvalue computation was occasionally skipped. A fudge factor of 1/1000 of the implicit time step was added to compensate for round-off error in the summation of the implicit time. See NEIG < 0 on *CONTROL_IMPLICIT_EIGENVALUE.
- Added support for *CONSTRAINED_LINEAR for 2D implicit problems. It was already supported for standard 3D problems.
- Added warning for implicit when the product of ILIMIT and MAXREF (two parameters on *CONTROL_IMPLICIT_SOLUTION) is too small. For the special case when the user changes the default of ILIMIT to 1 to choose Full Newton and does not change MAXREF then MAXREF is reset to 165 and a warning is generated. Reinstate the option of MAXREF < 0.
- Fixed the display of superelements in LS-PrePost. Enhanced reading of Nastran dmig files to allow for LS-DYNA-like comment lines starting with '$'. Fixed a problem with implicit initialization in MPP with 2 or more superelements. See *ELEMENT_DIRECT_MATRIX_INPUT.
- Turned off annoying warning messages associated with zero contact elemental stiffness matrices coming from mortar contact. See *CONTACT_..._MORTAR
- Fixed construction of d3mode file in MPP. Involves proper computation of the reduced stiffness matrix. See *CONTROL_IMPLICIT_MODES
- Fixed up *PART_MODES to correctly handle constraint modes.

  - removed rigid body modes
  - correct construction of reduced stiffness matrix

- Enhanced the error handling for input for *PART_MODES.
- Modified open statements for binary files used by implicit to allow for use of *CASE.
- Removed internal use files such as spooles.res when not required for debugging.
- Fixed implicit static condensation and implicit mode computation to properly deal with the *CASE environment. See *CONTROL_IMPLICIT_-STATIC_CONDENSATION and *CONTROL_IMPLICIT_MODES. Sort node/dof sets for implicit_mode to get correct results. Properly handle cases with only solid elements.
- Add implicit implementation of the new "last location" feature for MPP error tracking.
- Fixed problem with implicit processing of rigid body data with deformable to rigid switching (*DEFORMABLE_TO_RIGID).

- Extended Implicit model debugging for LPRINT = 3 (*CONTROL_IMPLICIT_SOLVER) to isogeometric and other large elemental stiff matrices.
- Added beam rotary mass scaling to the modal effective mass computation. Enhanced implicit computation of modal effective mass that is output to file eigout with *CONTROL_IMPLICIT_EIGENVALUE. We had to account for boundary SPC constraints as well as beam reference nodes to get the accumulated percentage to add up to 100%.
- Fixed a problem reporting redundant constraints for MPP Implicit.
- Enhanced *CONTACT_AUTO_MOVE for implicit.
- Fixed Implicit handling of *CONSTRAINED_TIE-BREAK in MPP.
- Added support for implicit dynamics to *MAT_157 and *MAT_120.
- Skip frequency damping during implicit static dynamic relaxation.
- Added feature to simulate brake squeal. Transient and mode analysis can be combined to do the brake squeal study by intermittent eigenvalue analysis. Besides *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS, *CONTROL_IMPLICIT_SOLVER should also be used, setting LCPACK = 3 to enable nonsymmetric stiffness matrix. In the nonsymmetric stiffness matrix analysis such as brake squeal analysis, the damping ratio, defined as -2.0*RE(eigenvalue)/ABS(IMG(eigenvalue)), can be output to the eigout file and plotted in LS-PrePost. A negative damping ratio indicates an unstable mode.
- Add a warning message if the defined rotational speed is not the same as NOMEG in *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS.
- *CONTROL_IMPLICIT: Fixed a bug to initialize velocity correctly when using a displacement file in dynamic relaxation for implicit MPP.
- Nonlinear implicit solver 12 is made default implicit solver, which is aimed for enhanced robustness in particular relation to BFGS and line search.
- Parameter IACC available on *CONTROL_ACCURACY to invoke enhanced accuracy in selected elements, materials and tied contacts. Included is strong objectivity in the most common elements, strong objecitity and physical respons in most commont tied contacts and full iteration plasticity in *MATs 24 and 123. For more detailed information refer to the manual.
- Bathe composite time integration scheme implemented for increased stability and conservation of energy/momentum, see *CONTROL_IMPLICIT_DYNAMICS. Time integration parameter ALPHA on CONTROL_IMPLICIT_DYNAMICS is used for activation.
- For NLNORM.LT.0 all scalar products in implicit are with respect to all degrees of freedom, sum of translational and rotational (similar to NLNORM.EQ.4), just that the rotational dofs are scaled using ABS(NLNORM) as a characteristic length to appropriately deal with consistency of units.
- The message 'convergence prevented due to unfulfilled bc...' has annoyed users. Here this is loosened up a little and also accompanied with a check that the bc that prevents convergence is actually nonzero. Earlier this prevention has activated even for SPCs modelled as prescribed zero motion, which does not make sense.

- Implicit now writes out the last converged state to the d3plot database on error termination if not already written.
- Fixed bug for *CONTROL_IMPLICIT_MODAL_DYNAMIC if jobid is used.

- ***INITIAL**

  - Fix incorrect NPLANE and NTHICK for *INITIAL_STRESS_SHELL when output to dynain file for shell type 9.
  - Fix *INITIAL_STRAIN_SHELL output to dynain for shell types 12 to 15 in 2D analysis.
  - Write out strain at only 1 intg point if INTSTRN = 0 in *INTERFACE_-SPRINGBACK_LSDYNA and all strains at all 4 intg points if INTSTRN = 1 and nip = 4 in *SECTION_SHELL.
  - *INITIAL_EOS_ALE: Allow initialization of internal energy density, relative volume, or pressure in ALE elements by part, part set, or element set.
  - *INITIAL_VOLUME_FRACTION_GEOMETRY: Add option (FAMMG < 0) to form pairs of groups in *SET_MULTI-MATERIAL_GROUP_LIST to replace the first group of the pair by the second one.
  - *INITIAL_STRESS_DEPTH can now work with parts that have an Equation of State (EOS types 1, 4, 6 only). Note however that *INITIAL_STRESS_-DEPTH does not work with ALE.
  - Fix several instances of overwriting the initial velocities of any interface nodes read in from a linking file (SMP only).
  - *INITIAL_VOLUME_FRACTION_GEOMETRY: Add local coordinate system option for box.
  - The initial strain and energy is calculated for *INITIAL_FOAM_REFERENCE_GEOMETRY.
  - Add the option of defining the direction cosine using two nodes for *INITIAL_VELOCITY_GENERATION.
  - Fix incorrect transformation of *DEFINE_BOX which results in incorrect initial velocities if the box is used in *INITIAL_VELOCITY.
  - Fix incorrect initial velocity when using *INITIAL_VELOCITY with NX = -999.
  - Fix seg fault when using *INITIAL_INTERNAL_DOF_SOLID_TYPE4 in dynain file.
  - Do not transform the translational velocities in *INITIAL_VELOCITY or *INITIAL_VELOCITY_GENERATION if the local coordinate system ICID is defined.
  - Fix uninitialized velocities when using *INITIAL_VELOCITY_GENERATION with STYP = 2, i.e. part id, for *ELEMENT_SHELL_COMPOSITE/*ELEMENT_TSHELL_COMPOSITE.
  - Fix incorrect initialization of velocities if using *INITIAL_VELOCITY_GENERATION with STYP = 1, i.e. part set for shells with formulation 23 & 24.
  - Fix incorrect initial velocity and also mass output to d3hsp for shell types 23 & 24.

- Fix incorrect initial velocities when using *INITIAL_VELOCITY_GENERATION with irigid = 1 and *PART_INERTIA with xc = yc = zc = 0 and nodeid > 0 with *DEFINE_TRANSFORMATION.
- Fix incorrect stress initialization of *MAT_057/MAT_LOW_DENSITY_FOAM using dynain file with *INITIAL_STRESS_SOLID when NHISV is equal to the number of history variables for this mat 57.
- Fix seg fault when reading dynain.bin
- Fixed stress initialization (*INITIAL_STRESS_SECTION) for type 13 tetrahedral elements. The pressure smoothing was causing incorrect pressure values in the elements adjacent to the prescribed elements.
- Assign initial velocities (*INITIAL_VELOCITY) to beam nodes that are generated when release conditions are defined (RT1, RT2, RR1, RR2 on *ELEMENT_BEAM.)
- Added an option to retain bending stiffness in spot weld beams that have prescribed axial force. To use is, set KBEND = 1 on *INITIAL_AXIAL_-FORCE_BEAM.
- Fix for *INITIAL_STRESS_BEAM when used with spotweld beam type 9. It was possible that error/warning message INI+140 popped up even if number of integration points matched exactly.
- Fix for the combination of type 13 tet elements and *INITIAL_STRESS_SOLID. The necessary nodal values for averaging (element volume, Jacobian) were not correctly initialized. Now the initial volume (IVEFLG) is used to compute the correct initial nodal volume.

- **Isogeometric Elements**

  - Enable spc boundary condition to be applied to extra nodes of nurbs shell, see *CONSTRAINED_NODES_TO_NURBS_SHELL
  - Fix a bug for isogeometric element contact, IGACTC = 1, that happens when more than one NURBS patches are used to model a part so that a interpolated elements have nodes belonging to different NURB patches.
  - *ELEMENT_SOLID_NURBS_PATCH:

    - Enable isogeometric analysis for solid elements, it is now able to do explicit and implicit analysis, such as contact and eigenvalue analysis, etc.
    - Add mode stress analysis for isogeometric solid and shell elements so that the isogeometric element is also able to do frequency domain analysis.

  - Add reduced, patch-wise integration rule for C1-continuous quadratic NURBS. This can be used by setting INT = 2 in *ELEMENT_SHELL_-NURBS_PATCH.

- Add trimmed NURBS capability. Define NL trimming loops to specify a trimmed NURBS patch. Use *DEFINE_CURVE (DATTYP = 6) to specify define trimming edges in the parametric space.
- Fix bug in added mass report for *ELEMENT_SHELL_NURBS_PATCH in MPP.

- ***LOAD**

  - *LOAD_GRAVITY_PART and staged construction (*DEFINE_STAGED_-CONSTRUCTION_PART) were ignoring non-structural mass MAREA (shells) and NSM (beams). Now fixed.
  - Fix for *INTERFACE_LINKING in MPP when used with adaptivity.
  - Updates for *INTERFACE_LINKING so that it can be used with adaptivity, provided the linked parts are adapting.
  - Fix for *INTERFACE_LINKING when used with LSDA based files generated by older versions of the code.
  - *DEFINE_CURVE_FUNCTION:

    - Functions "DELAY", PIDCTL" and "IF" of are revised.
    - Add sampling rate and saturation limit to PIDCTL of *DEFINE_-CURVE_FUNCTION.
    - "DELAY" of *DEFINE_CURVE_FUNCTION can delay the value of a time-dependent curve by "-TDLY" time steps when TDLY < 0.

  - Add edge loading option to *LOAD_SEGMENT_SET_NONUNIFORM.
  - Fix insufficient memory error,SOL+659, when using *LOAD_ERODING_-PART_SET with mpp.
  - Fix incorrect loading when using *LOAD_ERODING_PART_SET with BOXID defined.
  - Fix incorrect pressure applied if the directional cosines, V1/V2/V3, for *LOAD_SEGMENT_SET_NONUNIFORM do not correspond to a unit vector.
  - Add *DEFINE_FUNCTION capability to *LOAD_SEGMENT_SET for 2D analysis.
  - Fix incorrect behavior when using arrival time, AT, or box, BOXID, in *LOAD_ERODING_PART_SET.
  - Fix error when runing analysis with *LOAD_THERMAL_CONSTANT_EL-EMENT_(OPTION) in MPP with ncpu > 1.
  - Fixed *LOAD_STEADY_STATE_ROLLING when used with shell form 2 when used with Belytschko-Wong-Chang warping stiffness (BWC = 1 *CONTROL_SHELL).
  - Add "TIMESTEP" as a code defined value available for *DEFINE_FUNCTION and *DEFINE_CURVE_FUNCTION. It holds the current simulation timestep.
  - Fixed issues involving *LOAD_THERMAL_D3PLOT.

- Allow extraction of node numbers in loadsetud for all values of LTYPE in *USER_LOADING_SET. Comments included appropriately in the code. Argument list of loadsetud is changed accordingly.
- Implemented SPF simulation (*LOAD_SUPERPLASTIC_FORMING) for 2d problems.
- Added effective stress as target variable for SPF simulation.
- Added box option for SPF simulation to limit target search regions.

- ***MAT**

  - Fix output to d3hsp for *MAT_HYPERELASTIC_RUBBER. Broken in r93028.
  - Error terminate with message, KEY+1115, if_STOCHASTIC option is invoked for *MATs 10,15,24,81,98, 123 but no *DEFINE_STOCHASTIC_VARIATION or *DEFINE_HAZ_PROPERTIES keyword is present in the input file.
  - Fix spurious error termination when using *DEFINE_HAZ_PROPERTIES with adaptivity.
  - Fixed *MATs 161 and 162 when run with MPP. The array that is used to share delamination data across processors had errors.
  - *MAT_261/*MAT_262: Fixed problem using *DAMPING_PART_STIFFNESS together with RYLEN = 2 in *CONTROL_ENERGY.
  - Added safety check for martensite phase kinetics in *MAT_244.
  - Fix for combination of *MAT_024_STOCHASTIC and shell elementstype 13, 14, and 15 (with 3d stress state).
  - Fix bug in *MATs 21 and 23 when used with *MAT_ADD_THERMAL_EXPANSION.
  - *MAT_ALE_VISCOUS: Implement a user defined routine in dyn21.F to compute the dynamic viscosity.
  - Add histlist.txt to usermat package. This file lists the history variables by material.
  - Bug in *MAT_089 fixed: The load curve LCSS specifies the relationship between "maximum equivalent strain" and the von Mises stress. The "maximum equivalent strain" includes both elastic and plastic components. The material model was not calculating this variable as intended, so was not following LCSS accurately. The error was likely to be more noticeable when elastic strains are a significant proportion of the total strain e.g. for small strains or low initial Youngs modulus.
  - Fixed bug affecting *MAT_119: unpredictable unloading behaviour in local T-direction if there are curves only for the T-direction and not for the S-direction.
  - Fixed bug in *MAT_172: Occured when ELFORM = 1 (Hughes-Liu shell formulation) was combined with Invariant Numbering (INN > 0 on *CONTROL_ACCURACY). In this case, the strain-softening in tension did not work: after cracking, the tensile strength remained constant.

- New option for *MAT_079: Load curve LCD defining hysteresis damping versus maximum strain to date. This overrides the default Masing behaviour.
- *MAT_172:
  - Added error termination if user inputs an illegal value for TYPEC. Previously, this condition could lead to abnormal terminations that were difficult to diagnose.
  - Fixed bug affecting ELFORM = 16 shells made of *MAT_172 – spurious strains could develop transverse to the crack opening direction.
- Fixed bug in *MAT_ARUP_ADHESIVE (*MAT_169). The displacement to failure in tension was not as implied by the inputs TENMAX and GCTEN. For typical structural adhesives with elastic stiffness of the order of 1000-10000 MPa, the error was very small. The error became large for lower stiffness materials.
- *MAT_SPR_JLR:
  - Modify output variables from *MAT_SPR_JLR (see Manual).
  - Fix bug that caused spurious results or unexpected element deletion if TELAS = 1.
- Fixed bug in *MAT_174 - the code could crash when input parameters EUR = 0 and FRACR = 0..
- Fix MPP problem when writing out aea_crack file for *MAT_WINFRITH.
- Include *MAT_196 as one that triggers spot weld thinning.
- *MAT_ADD_FATIGUE: Implemented multi slope SN curves to be used in random vibration fatigue (*FREQUENCY_DOMAIN_RANDOM_VIBRATION_FATIGUE) and SSD fatigue (FREQUENCY_DOMAIN_SSD_FATIGUE).
- Guard against possible numerical round off that in some cases might result in unexpected airflow in *MAT_ADD_PORE_AIR.
- Added new material *MAT_115_O/*MAT_UNIFIED_CREEP_ORTHO.
- *MAT_274: Added support for 2D-solids. New flag (parameter 8 on card 2) is used to switch normal with in-plane axis.
- *MAT_255: Fixed bug in plasticity algorithm and changed from total strain rate to plastic strain rate for stability. Added VP option (parameter 5 on card 2) for backwards compatibility: VP = 0 invokes total strain rate used as before.
- Added new cohesive material *MAT_279/*MAT_COHESIVE_PAPER to be used in conjunction with *MAT_274/*MAT_PAPER.
- User materials: Added support for EOS with user materials for tshell formulations 3 and 5.
- Fixed bug in dyna.str when using EOS together with shells and orthotropic materials.

- *MAT_122: A new version of *MAT_HILL_3R_3D is available. It supports temperature dependent curves for the Young's/shear moduli, Possion ratios, and Hill's anisotropy parameters. It also supports 2D-tables of yield curves for different temperatures. Implicit dynamics is supported. The old version is run if parameter 5 on card 3 is set to 1.0.
- Added the phase change option to *MAT_216, *MAT_217, *MAT_218 to allow material properties to change as a function of location. This capability is designed to model materials that change their properties due to material processing that is otherwise not modeled. For example, increasing the mass and thickness due to the deposition of material by spraying. It is not used for modeling phase changes caused by pressure, thermal loading, or other mechanical processes modeled within LS-DYNA.
- Fix internal energy computation of *MAT_ELASTIC_VISCOPLASTIC_-THERMAL/MAT_106.
- Fix incorrect results or seg fault for *MAT_FU_CHANG_FOAM/MAT_083 if KCON > 0.0 and TBID.ne.0.
- If SIGY = 0 and S = 0 in *MAT_DAMAGE_2/MAT_105, set S = EPS1/200, where EPS1 is the first point of yield stress input or the first ordinate point of the LCSS curve.
- Set xt = 1.0E+16 as default if user inputs 0.0 for *MAT_ENHANCED_COMPOSITE_DAMAGE/MAT_054. Otherwise, random failure of elements may occur. Implemented for thick shells and solids.
- Allow *MAT_ENHANCED_COMPOSITE_DAMAGE/MAT_054 failure mechanism to work together with *MAT_ADD_EROSION for shells.
- Fix incorrect erosion behavior if *MAT_ADD_EROSION is used with failure criteria defined for *MAT_123/MAT_MODIFIED_PIECEWISE_LINEAR_PLASTICITY.
- Fix non-failure of triangular elements type 4 using *MAT_ADD_EROSION with NUMFIP = -100.
- Implement scaling of failure strain for *MAT_MODIFIED_PIECEWISE_-LINEAR_PLASTICITY_STOCHASTIC/MAT_123_STOCHASTIC for shells.
- Fix incorrect behavior for *MAT_LINEAR_ELASTIC_DISCRETE_-BEAM/MAT_066 when using damping with implicit(statics) to explicit switching.
- Fix error due to convergence when using *MAT_CONCRETE_-EC2/MAT_172 in implicit and when FRACRX = 1.0 or FRACRY = 1.0
- Fix incorrect fitting results for *MAT_OGDEN_RUBBER/MAT_077_O if the number of data points specified in LCID is > 100.
- Fix incorrect fitting results for *MAT_MOONEY-RIVLIN-RUBBER/MAT_027 if the number of data points specified in LCID is > 100.
- Fix incorrect forces/moments when preloads are used for *MAT_-067/NONLINEAR_ELASTIC_DISCRETE_BEAM and the strains changes sign.

- Implement *MAT_188/MAT_THERMO_ELASTO_VISCO-PLASTIC_CREEP for 2D implicit analysis.
- Support implicit for *MAT_121/MAT_GENERAL_NONLIN-EAR_1DOF_DISCRETE_BEAM.
- Fix seg fault when using *DEFINE_HAZ_TAILOR_WELDED_BLANK with *DEFINE_HAZ_PROPERTIES.
- Fix ineffective *MAT_ADD_EROSION if the MID is defined using a alpha-numeric label.
- Fix seg fault when using *MAT_PIECEWISE_LINEAR_PLASTIC_THER-MAL/MAT_255 for solids.
- Zero the pressure for *MAT_JOHNSON_HOLMQUIST_JH1/MAT_241 after it completely fractures, i.e. D>=1.0, under tensile load.
- Fix incorrect element failure when using EPSTHIN and VP = 0 for *MAT_-123/MODIFIED_PIECEWISE_LINEAR_PLASTICITY.
- Fix error termination when using adaptive remeshing for 2D analysis with *MAT_015/JOHNSON_COOK and NIP = 4 in *SECTION_SHELL and ELFORM = 15.
- Fix erosion due to damage, max shear & critical temperature in elastic state for *MAT_MODIFIED_JOHNSON_COOK/MAT_107 for solids.
- Check diagonal elements of C-matrix of *MAT_002/MAT_*OPTION* TROP-IC_ELASTIC and error terminate with message, STR+1306, if any of them are negative.
- Fix plastic strain tensor update for *MAT_082/*MAT_PLASTICITY_-WITH_DAMAGE.
- Fix error when using *MAT_144/MAT_PITZER_CRUSHABLE_FOAM with solid tetahedron type 10.
- Fix out-of-range forces after dynamic relaxation when using VP = 1 for *MAT_PIECEWISE_LINEAR_PLASTICITY and non-zero strain rate parameters, C & P, and the part goes into plastic deformation during dynamic relaxation.
- Fixed unit transformation for GAMAB1 and GAMAB2 on *MAT_DRY_-FABRIC. We were incorrectly transforming them as stress.
- Fixed implicit solutions with shell elements that use *MAT_040 and laminated shell theory.
- Fixed the stress calculation in the thermal version of *MAT_077.
- Corrected the AOPT = 0 option of ortho/anisotropic materials when use with skewed solid elements. Previously, the material direction was initialized to be equivalent to the local coordinate system direction. This is not consistent with the manual for skewed elements which states that the material a-axis is in the 1-2 directions for AOPT = 0. This is now fixed and the manual is correct.
- Fixed the AOPT = 0 option of ortho/anisotropic materials for tetrahedral element forms 10, 13, and 44.
- Fixed *MAT_082 for solid elements. An error in the history data was causing possible energy growth or loss of partially damaged elements.

- Modified *MAT_FABRIC/*MAT_034 FORM = 24 so that Poisson's effects occur in tension only.
- Modified *MAT_221/*MAT_ORTHOTROPIC_SIMPLIFIED_DAMAGE to correct the damage behavior. Prior to this fix, damage was applied to new increments of stress, but not the stress history, so material softening was not possible.
- Fixed *MAT_106 when used with curves to define the Young's modulus and Poisson's ratio and when used with thick shell form 5 or 6. The assumed strain field was unreasonable which caused implicit convergence to fail.
- Added 2 new erosion criteria for *MAT_221/*MAT_ORTHOTROPC_SIMPLIFIED_DAMAGE. The new options are NERODE = 10: a or b directions failure (tensile or compressive) plus out of plane failure bc or ca. NERODE = 11: a or b directions failure (tensile only) plus out of plane failure bc or ca.
- Added a new option for shell *MAT_022/*MAT_COMPOSITE_DAMAGE. When ATRACK = 1, the material directions will follow not only element rotation, but also deformation. This option is useful for modeling layered composites, that have material a-directions that vary by layer, by allowing each layer to rotate independently of the others. Within each layer, the b-direction is always orthogonal to the a-direction.
- Fixed the TRUE_T option on *MAT_100 and *MAT_100_DA. If the weld connects shells with different thickness and therefore different bending stiffness, the scheme used by TRUE_T to reduce the calculated moment could behave somewhat unpredictably. With the fix, TRUE_T behaves much better, both for single brick welds and brick assemblies.
- Added a warning message and automatically switch DMGOPT > 0 to DMGOPT = 0 on *MAT_FABRIC when RS < EFAIL or RS = EFAIL. This prevents a problem where weld assemblies did not fail at all when RS = 0.
- *MATs 9, 10, 11, 15, 88, and 224 are now available for thick shells, however only *MATs 15, 88, and 224 are available for the 2D tshell forms 1,2, and 6.
- Added thick shell support for the STOCHASTIC option of *MATs 10, 15, 24, 81, and 98.
- Added support for *MAT_096 for several solid element types including ELFORMs 3, 4, 15, 18, and 23.
- Added a MIDFAIL keyword option for *MAT_024, (MAT_PIECEWISE_-LINEAR_PLASTICITY). With this option, element failure does not occur until the failure strain is reached in the mid plane layer. If an even number of layers is used, then the failure occurs when the 2 closest points reach the failure strain.
- Enabled *MATs 26 and 126 (HONEYCOMB) to be used with thick shell forms 3, 5, and 7. These was initialized incorrectly causing a zero stress.
- Enabled *MAD_ADD_EROSION to be used with beams that have user defined integration. Memory allocation was fixed to prevent memory errors.
- Enabled OPT = -1 on *MAT_SPOTWELD for solid elements.

- Enabled thick shells to use *MATs 103 and 104 in an implicit solution. These materials were lacking some data initialization so they would not converge.
- Enabled solid elements with user-defined orthotropic materials to work with the INTOUT and NODOUT options on *DATABASE_EXTENT_BINARY. The transformation matrix was stored in the wrong place causing strain and stress transformations to fail.
- Enabled *MAT_017 to run with thick shell forms 3 and 5. Neither element was initialized correctly to run materials with equations of state.
- Add degradation factors and strain rate dependent strength possibility for *MAT_054/*MAT_ENHANCED_COMPOSITE_DAMAGE solids.
- Fixed bug in *MAT_058/*MAT_LAMINATED_COMPOSITE_FABRIC when used with strain-rate dependent tables for stiffnesses EA, EB and GAB and LAMSHT = 3.
- Add strain rate dependency of ERODS in *MAT_058.
- Add possibility to use *DEFINE_FUNCTION for *MAT_SPOTWELD_DAMAGE_FAILURE (*MAT_100), OPT = -1/0. If FVAL = FunctionID, then a *DEFINE_FUNCTION expression is used to determine the weld failure criterion using the following arguments: func (N_rr, N_rs, N_rt, M_rr, M_ss, M_tt).
- Store tangential and normal separation (delta_II & delta_I) as history variables 1&2 of *MAT_138/*MAT_COHESIVE_MIXED_MODE.
- Add second normalized traction-separation load curve (TSLC2) for Mode II in *MAT_186/*MAT_COHESIVE_GENERAL.
- Fixed bug in using *MAT_157/*MAT_ANISOTROPIC_ELASTIC_PLASTIC with IHIS.gt.0 for shells. Thickness strain update d3 was not correct and plasticity algorithm failed due to typo.
- Fixed bug in *MAT_157 for solids: This affected the correct stress transformation for post-processing using CMPFLG = 1 in *DATABASE_EXTENT_BINARY.
- Fixed bug in *MAT_225 (*MAT_VISCOPLASTIC_MIXED_HARDENING) when using Table-Definition together with kinematic hardening.
- Add load curves for rate dependent strengths (XC, XT, YC, YT, SC) in *MAT_261/*MAT_LAMINATED_FRACTURE_DAIMLER_PINHO (shells only).
- Add table definition for LCSS for rate dependency in *MAT_261 (shells only).
- Add load curves for rate dependent strengths (XC, XCO, XT, XTO, YC, YT, SC) in *MAT_262/*MAT_LAMINATED_FRACTURE_DAIMLER_CAMANHO (shells only).
- Fixed bug when using *MAT_261 or *MAT_262 solids (ELFORM = 2).
- Add load curves for SIGY and ETAN for rate dependency of *MAT_262 (shells only)
- *MAT_021_OPTION

  - Fixed a bug for defining different orientation angles through the thickness of TSHELL elements (formulations 2 and 3)
  - Added new option CURING:
  - Two additional cards are read to define parameters for curing kinetics. Formulation is based on Kamal's model and considers one ODE for the state of cure.
  - State of cure does not affect the mechanical parameters of the material.
  - CTE's for othotropic thermal expansione can be defined in a table with respect to state of cure and temperature.
  - An orthotropic chemical shrinkage is accounted for.

- *MAT_REINFORCED_THERMOPLASTICS_OPTION (*MAT_249_OPTION):

  - Fiber shear locking can be defined wrt to the fiber angle or shear angle.
  - Output of fiber angle to history variables.
  - Simplified input: Instead of always reading 8 lines, now the user only has to specify data for NFIB fibers.
  - Added fiber elongation to history variables in *MAT_249 for posprocessing.
  - New Option UDFIBER (based on a user defined material by BMW):
    - Transversely isotropic hyperelastic formulation for each fiber family (see Bonet&Burton,1998).
    - Anisotropic bending behavior based on modified transverse shear stiffnesses.
    - Best suited for dry NCF's.

- *MAT_GENERALIZED_PHASE_CHANGE (*MAT_254):

  - New material that is a generalized version of *MAT_244 with application to a wider range of metals.
  - Up to 24 different phases can be included.
  - Between each of the phases, the phase transformation can be defined based on a list of generic transformation laws. For heating JMAK and Oddy are implemented. For cooling Koistinen-Marburger, JMAK and Kirkaldy can be chosen.
  - Constant parameters for the transformations are given as 2d tables, parameters depending on temperature (rate) or phase concentration employ 3d tables.
  - Plasticity model (temperature and strain rate dependent) similar to MAT_244.
  - Transformation induced strains.
  - TRIP algorithm included.
  - Temperature dependent mixture rules.

    - Parameter 'dTmax' that defines the maximum temperature increment within a cycle. If the temperature difference at a certain integration point is too high, local subcycling is performed.
    - Implemented for explicit/implicit analysis and for 2d/3d solid elements.

- *MAT_ADHESIVE_CURING_VISCOELASTIC (*MAT_277):

    - New material implementation including a temperature dependent curing process of epoxy resin based on the Kamal-Sourour-model.
    - Material formulation is based on *MAT_GENERAL_VISCOELASTIC.
    - Viscoelastic properties defined by the Prony series, coefficients as functions of state of cure.
    - Chemical and thermal shrinkage considered (differential or secant formulations).
    - Available for shell and solid elements.
    - Can be used in combination with *MAT_ADD_COHESIVE.
    - Implemented for explicit and implicit analysis.
    - An incremental and a total stress calculation procedure available.

- Enable *MAT_ADD_EROSION to be safely used with material models that have more than 69 history variables, for now the new limit is 119.
- Use correct element ID for output of failed solid elements when GISSMO (*MAT_ADD_EROSION) is used with *CONTROL_DEBUG.
- Improve performance of GISSMO (*MAT_ADD_EROSION with IDAM = 1), especially when used with *MAT_024, no other failure criteria, shell elements, and DMGEXP = 1 or 2. Allows speed-up of 10 to 20 percent.
- Add new keyword *MAT_ADD_GENERALIZED_DAMAGE. It provides a very flexible approach to add non-isotropic (tensorial) damage to standard materials in a modular fashion. Solely works with shell elements at the moment.
- Correct the computation of effective strain for options ERODS < 0 in *MAT_058 (*MAT_LAMINATED_COMPOSITE_FABRIC) and EFS < 0 in *MAT_261 and *MAT_262 (*MAT_LAMINATED_FRACTURE_DAIMLER_...). The shear strain term was twice the size as it should have been.
- Adjust stiffness for time step calculation in *MAT_076 and subsequent models (*MAT_176, *MAT_276, ...) to prevent rarely observed instabilities.
- Add output of original and fitted curves to messag and separate file (curveplot_<MID>) for *MAT_103.
- In *MAT_104 (*MAT_DAMAGE_1), stress-strain curve LCSS can now be used directly with all FLAG options (-1,0,1,10,11), no fitting.
- Correct strain calculation for anisotropic damage in *MAT_104 (*MAT_DAMAGE_1) with FLAG = -1.
- Initialize stress triaxiality of *MAT_107 (*MAT_MODIFIED_JOHNSON_COOK) to zero instead of 1/3.

- Avoid negative damage in *MAT_107 (*MAT_MODIFIED_JOHNSON_-COOK) with FLAG2 = 0 for solid elements.
- Rectify the characteristic element length in *MAT_138 (*MAT_COHESIVE_-MIXED_MODE) for solids type 21 and 22 (cohesive pentas) and shell type 29 (cohesive shell) for "curve" options T < 0 and S < 0.
- Correct/improve material tangent for *MAT_181 with PR > 0 (foam option).
- Add possibility to define logarithmically defined strain rate table LCID-T in material *MAT_187 (*MAT_SAMP-1).
- Fix missing offset when using *DEFINE_TRANSFORMATION with load curve LCID-P in *MAT_187 (*MAT_SAMP-1).
- Add reasonable limit for biaxial strength in *MAT_187 with RBCFAC > 0.5 to avoid concave yield surface.
- Improve performance of *MAT_187 to reach speed-up of 10 to 40 percent, depending on which options are used.
- Add new option for *MAT_224 (*MAT_TABULATED_JOHNSON_COOK). With BETA < 0 not only a load curve but now also a table can be referred to. The table contains strain rate dependent curves, each for a different temperature.
- Fix for implicit version of *MAT_224 (*MAT_TABULATED_JOHNSON_-COOK). Computations with shell elements should converge faster now.
- *MAT_224 (*MAT_TABULATED_JOHNSON_COOK) can now be used in implicit even with temperature dependent Young's modulus (parameter E < 0).
- Always store the Lode parameter as history variable #10 in *MAT_224 (*MAT_TABULATED_JOHNSON_COOK), not just for LCF being a table.
- Variable LCI of *MAT_224 / *MAT_224_GYS can now refer to a *DEFINE_-TABLE_3D. That means the plastic failure strain can now be a function of Lode parameter (TABLE_3D), triaxiality (TABLE), and element size (CURVE).
- For thick shells type 1 and 2, the element size in *MAT_224 is now correct.
- Add new option for definition of parameters FG1 and FG2 in *MAT_240 (*MAT_COHESIVE_MIXED_MODE_ELASTOPLASTIC_RATE).
- Add new option to *MAT_240: new load curves LCGIC and LCGIIC define fracture energies GIC and GIIC as functions of cohesive element thickness. GIC_0, GIC_INF, GIIC_0, and GIIC_INF are ignored in that case.
- Add new feature to *MAT_248 (*MAT_PHS_BMW). Estimated Hocket-Sherby parameters are written to history variables based on input functions and phase fractions.
- Add new option ISLC = 2 to *MAT_248 (*MAT_PHS_BMW) which allows to define load curves (cooling rate dependent values) for QR2, QR3, QR4, and all parameters on Cards 10 and 11.
- Add new option LCSS to *MAT_252 (*MAT_TOUGHENED_ADHESIVE_-POLYMER): A load curve, table or 3d table can now be used to define rate and temperature dependent stress-strain behavior (yield curve).

- Fix for *MAT_255, evaluation of 2d tables LCIDC and LCDIT. Negative temperatures were interpreted as logarithmic rates.
- Add new material model *MAT_280 (*MAT_GLASS) for shell elements. It is a smeared fixed crack model with a selection of different brittle, stress-state dependent failure criteria and crack closure effects.
- *DEFINE_FABRIC_ASSEMBLIES: Assemblies of *MAT_FABRIC part sets can be specified to properly treat bending of t-intersecting fabrics that are stitched or sewn together. See ECOAT, TCOAT and SCOAT on *MAT_FABRIC_... Bending can only occur within an assembly, aka a part set.
- *MAT_USER_DEFINED_MATERIAL_MODELS: In user defined material models, a logical parameter 'reject' can be set to .true. to indicate to the implicit solver that equilibrium iterations should be aborted. The criterion is the choise of the implementor, but it could be if plastic strain increases by more than say 5% in one step or damage increases too much, whatever that might render an inaccurate prediction and bad results. Setting this parameter for explicit won't do anything.
- IHYPER = 3 for user shell materials now supports thickness train update, see *MAT_USER_DEFINED_MATERIAL_MODELS.
- *MAT_SIMPLIFIED_RUBBER/FOAM: AVGOPT < 0 is now supported for the FOAM option, which activates a time averaged strain rate scheme to avoid noisy response.
- MAT_181 is now supported for 2D implicit simulations.
- *MAT_ADD_EROSION:

  - A number of extensions and improvements to the DIEM damage model were made, IDAM < 0.
  - General efficiency, it was slow, now it's GOT to be faster.
  - NCS can be used as a plastic strain increment to only evaluate criteria in quantifications of plastic strain.
  - NUMFIP < 0 is employing the GISSMO approach, number of layers for erosion.
  - A new ductile damage criterion based on principal stress added (DMITYP = 4).
  - MSFLD and FLD can be evaluated in mid or outer layers to separate membrane and bending instability (P2).
  - MSFLD and FLD can use an incremental or direct update of instability parameter (P3).
  - Output of integration point failure information made optional (Q2).
  - Specifying DCTYP = -1 on the damage evolution card will not couple damage to stress but the damage variable is only calculated and stored.

- *MAT_SMOOTH_VISCOELASTIC_VISCOPLASTIC, *MAT_275: An elastic-plastic model with smooth transition between elastic and plastic mode is available. It incorporates viscoelasticity and viscoplasticity and is based on hyper-elastoplasticity so it is valid for arbitrarily large deformations and rotations. A sophisticated parameter estimation is required to match test data, it is available for implicit and explicit analyisis but perhaps mostly suited for implicit.

- *MAT_FABRIC_MAP: Stress map material 34 is equipped with bending properties identical to that of the form 14 and form -14 version of the fabric. Coating properties are set in terms of stiffness, thickness and yield. The material is supported in implicit, including optional accounting for the non-symmetric tangent. Should be used with bending stiffness on, and convergence is improved dramatically if geometric stiffness is turned on.
- *MAT_084 with predefined units (CONM < 0) is now transformed correctly with INCLUDE_TRANSFORM.
- If LCIDTE = 0 in *MAT_121, then LS-DYNA was crashing on some platforms, including Windows. This is fixed.
- Fix initialization issues so that PML models can be run with *CASE commands.
- *MAT_027 is revised to avoid accuracy issues for single precision executables.
- The nearly imcompressible condition is enhanced for *MAT_027 shell elements.
- Add a new material model as a option for *MAT_165. *MAT_PLASTIC_NONLINEAR_KINEMATIC_B is a mixed hardening material model, and can be used for fatigue analysis.
- Output local z-stress in *MAT_037, when *LOAD_STRESS_SURFACE is used. This was previously calculated and saved as another history variable.
- Add a new material model *MAT_260 (2 forms).

    - Uses non-associated flow rule and Hill's yield surface; including strain rate effect and temperate effect. MIT failure criteria is also implemented.
    - Implemented for solids and shells.
    - Strain rate sensitivity for solids.
    - Option to directly input the Pij and Gij values.
    - Separate the material model *MAT_260 into *MAT_260A and *MAT_260B:
        - MAT260A=*MAT_STOUGHTON_NON_ASSOCIATED_FLOW
        - MAT260B=*MAT_MOHR_NON_ASSOCIATED_FLOW
    - Incorporates FLD into the fracture strain, so as to consider the mesh size effect.
    - Calculates the characteristic length of the element for *MAT_260B, so that an size-dependent failure criterial can be used.
    - When failure happens for half of the integration points through the thickness, the element is deleted.

- Add Formablitiy Index to *MAT_036, *MAT_037, *MAT_226.

- Add new history variables for Formability Index, affecting *MAT_036, *MAT_037, *MAT_125, *MAT_226. Those new history variables are FI, beta, effective strain. These comes after the 4 regular history variables.
- *MAT_036, *MAT_125: New option_NLP is added to evaluate formability under non-linear strain paths. User inputs a forming limit diagram (FLD), and Formablitiy Index (F.I.) will be automatically converted to effective stain vs. beta based space.

- **MPP**

  - Fix problem of MPP pre-decomposition that can occur if the local directory specified in the pfile has very different lengths in the initial run vs the actual run The difference resulted in a line count difference in the size of the structured files created, throwing off the reading of the file in the actual run.
  - Straighten out some silist/sidist issues in MPP decomp:

    - silist and sidist outside of a "region" in the pfile are no longer supported, and an error message is issued which suggests the use of "region { silist" instead.
    - They have been undocumented for several years (since "region" was introduced), and had other issues.

  - Fix the keywords, CONTROL_MPP_DECOMPOSITION_CONTACT_DISTRIBUTE and CONTROL_MPP_DECOMPOSITION_CONTACT_ISOLATE, which were not treating each contact interface individually (as the manual states), but collectively.
  - Fix for MPP decomp of part sets.
  - Fixed *CONTROL_MPP_PFILE (when used inside an include file) so that it honors ID offsets from *INCLUDE_TRANSFORM for parts, part sets, and contact ids referenced in "decomp { region {" specifications. Furthermore, such a region can contain a "local" designation, in which case the decomposition of that region will be done in the coordinate system local to the include file, not the global system. For example:
    *CONTROL_MPP_PFILE
    decomp { region {partset 12 local c2r 30 0 -30 0 1 0 1 0 0}}
    would apply the c2r transformation in the coordinate system of the include file, which wasn't previously possible. The local option can be useful even if there are no such transformations, as the "cubes" the decomposition uses will be oriented in the coordinate system of the include file, not the global system. Furthermore, the following decomposition related keywords now have a_LOCAL option, which has the same effect:

    - *CONTROL_MPP_DECOMPOSITION_PARTS_DISTRIBUTE_LOCAL

        - *CONTROL_MPP_DECOMPOSITION_PARTSET_DISTRIBUTE_LOCAL
        - *CONTROL_MPP_DECOMPOSITION_ARRANGE_PARTS_LOCAL
        - *CONTROL_MPP_DECOMPOSITION_CONTACT_DISTRIBUTE_LOCAL
          (**Note:** LOCAL option is only valid when *INCLUDE_TRANSFORM with *DEFINE_TRANSFORMATION is used)

    - Revert revision 86884, which was:

        - "MPP: change to the decomposition behavior of *CONTROL_MPP_DECOMPOSITION_PARTS_DISTRIBUTE
          *CONTROL_MPP_DECOMPOSITION_PARTSET_DISTRIBUTE
          *CONTROL_MPP_DECOMPOSITION_ARRANGE_PARTS
          in the case where a decomposition transformation is also used. Previously, any such regions were distributed without the transformation being applied. This has been fixed so that any given transformation applies to these regions also. So now the transformations will NOT apply to these keywords. Really, the "region" syntax should be used together with *CONTROL_MPP_PFILE as it is more specific.

    - Modify behavior of DECOMPOSITION_AUTOMATIC so that if the initial velocity used is subject to *INCLUDE_TRANSFORM, the transformed velocities are used.
    - Fix MPP decomposition issue with "decomp { automatic }" which was not honored when in the pfile.
    - Save hex weld creation orientation to the pre-decomposition file so that the subsequent run generates the welds in the same way.
    - Fix for MPP not handling element deletion properly in some cases at decomposition boundaries.
    - Add new pfile option "contact { keep_acnodes }" which does NOT exclude slave nodes of adaptive constraints from contact, which is the default behavior. (MPP only.)
    - MPP Performance-Related Improvements:

        - Allow user input of *LOAD_SEGMENT_FILE through familied files.
        - Bug fix for *LOAD_SEGMENT_FILE to get correct time history data for pressure interpolation.
        - Output two csv files for user to check MPP performance:
            - load_profile.csv: general load balance
            - cont_profile.csv: contact load balance
        - Allow user to control decomp/distribution of multiple airbags using *CONTROL_MPP_DECOMPOSITION_ARRANGE_PARTS
        - memory2 = option on *KEYWORD line

- Disable unreferenced curves after decomposition using *CONTROL_-MPP_DECOMPOSITION_DISABLE_UNREF_CURVES. This applies to the curves used in the following options to speed up the execution several times.
    - *BOUNDARY_PRESCRIBED_MOTION_NODE
    - *LOAD_NODE
    - *LOAD_SHELL_ELEMENT
    - *LOAD_THERMAL_VARIABLE_NODE

- Bug fix for *CONTROL_MPP_DECOMPOSITION_SHOW with *AIRBAG_-PARTICLE.
- Fix cpu dependent results when using function RCFORC() in *DEFINE_-CURVE_FUNCTION. This affects MPP only.
- Fix hang up when using *DEFINE_CURVE_FUNCTION with element function BEAM(id,jflag,comp,rm) and running MPP with np > 1.
- *CONTROL_MPP_DECOMPOSITION: The cpu cost for solid elements -1 and -2 are accounted for in the mpp domain decomposition.
- Fix bug in *CONTROL_MPP_IO (Windows platform only) related to insufficient administrative privileges for writing tmp file on root drive.
- Revise l2a utility on Windows platform to create identical node output format as Linux.

- **Output**

  - Fix for MPP external work when bndout is output and there are *BOUNDARY_PRESCRIBED_MOTION_RIGID commands in the input.
  - Fixed the output of forces and associated energy due to *LOAD_RIGID_-BODY for both explicit and implicit (*DATABASE_BNDOUT).
  - Fixed stress and strain output of thick shells when the composite material flag is set on *DATABASE_EXTENT_BINARY. The transformation was backwards.
  - If the size of a single plot state was larger than the d3plot size defined by x=<factor> on the execution line, the d3plot database may not be readable by LS-PrePost. This issue is now fixed.
  - *DATABASE_PROFILE: Output data profiles for beams (TYPE = 5) and add density as DATA = 20.
  - New option HYDRO = 4 on *DATABASE_EXTENT_BINARY. Outputs 7 additional variables: the same 5 as HYDRO = 2 plus volumetric strain (defined as Relative Volume - 1.0) and hourglass energy per unit initial volume.
  - Fix for binout output of swforc file which can get the data vs. ids out of sync when some solid spotwelds fail.
  - Fix for d3plot output of very large data sets in single precision.
  - Fix for output of bndout data for joints in MPP, which was writing out incorrect data in some cases.

- Added new option *INTERFACE_SPRINGBACK_EXCLUDE to exclude selected portions from the generated dynain file.
- Add a new option to *INTERFACE_COMPONENT_FILE to output only 3 degrees of freedom to the file, even if the current model has 6.
- Minor change to how pressure is computed for triangles in the INTFOR output.
- Fix MPP output issue with intfor file.
- Fixes for writing and reading of dynain data in LSDA format.
- Corrected the summation of rigid body moments for output to bndout for some special cases in MPP.
- Corrected the output to d3iter when 10 node tets are present (D3ITCTL on *CONTROL_IMPLICIT_SOLUTION).
- Enhanced implicit collection of moments for the rcforc file.
- For implicit, convert spc constraint resultant forces to local coordinate system for output. Also corrected Implicit's gathering of resultant forces due to certain SPC constraints.
- Fixed the gathering of resultant forces in implicit for prescribed motion on nodes of a constrained rigid body for output to bndout.
- Added output of modal dynamics modal variables to a new file moddynout. Output is controlled by *CONTROL_IMPLICIT_MODAL_DYNAMICS.
- Corrected the output of resultant forces for Implicit Linear analysis. Corrected the output of resultant forces for MPP executions. These enhancements affect a number of ASCII files including bndout.
- The following 4 enhancements are to the eigensolvers, including that used for *CONTROL_IMPLICIT_EIGENVALUE.

    - Standardized and enhance the warning/error messages for Implicit eigensolution for the case where zero eigenmodes are computed and returned in eigout and d3eigv.
    - Added nonsymmetric terms to the stiffness matrix for the implicit rotational dynamics eigenanalysis. This allows brake squeal analysis with the contact nonsymmetric terms from mortar contact now included in the analysis.
    - Updated implicit eigensolution for problems with nonsymmetric stiffness matrices. Fixed Rotational Dynamics eigensolution to work correctly when first order matrix (W) is null. (See *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS).
    - Added the eigensolution for problems with stiffness (symmetric or nonsymmetric), mass, and damping.

- Improve Implicit's treatment of constrained joints to account for rounding errors. Applicable to *CONSTRAINED_JOINT with *CONTROL_IMPLICIT_GENERAL.
- For implicit springback, zero out the forces being reported to rcforc for those contact interfaces disabled at the time of springback. Also enhance the

removal of contact interfaces for springback computations. For *INTERFACE_SPRINGBACK.

- *DATABASE_RECOVER_NODE is available to recover nodal stress.
- Fix a bug for detailed stress output, eloutdet, for SOLID type 18.
- Support new format of interface force files for ALE, DEM, and CPM. LS-PrePost can display the correct label for each output component.
- Added *DATABASE_NCFORC_FILTER option to allow the NCFORC data to be filtered using either single pass or double pass Butterworth filtering to smooth the output. Added the same filtering capability to *DATABASE_BINARY_D3PLOT. This capability is specified on the additional card for the D3PLOT option and does not require "_FILTER" in the keyword input.
- Fix incorrect mass properties for solids in SSSTAT file when using *DATABASE_SSSTAT_MASS_PROPERTIES.
- Fix seg fault during writing of dynain file if INSTRN = 1 in *INTERFACE_SPRINGBACK and STRFLG ≠ 0 in *DATABASE_EXTENT_BINARY and the *DATABASE_EXTENT_BINARY comes after *INTERFACE_SPRINGBACK. Also output warning message, KEY+1104.
- Fix zero strain values output to curvout for *DEFINE_CURVE_FUNCTION using function, ELHIST, for solid elements.
- Fix missing parts in d3part when MSSCL = 1 or 2 in *DATABASE_EXTENT_BINARY.
- Fix incorrect damping energy computation for glstat.
- Fix incorrect part mass in d3plot for shells, beams & thick shells.
- Fix incorrect curvout values when using BEAM(id,jflag,comp,rm) for *DEFINE_CURVE_FUNCTION and if the beam formulation is type 3, i.e. truss.
- Fix incorrect output to curvout file if using ELHIST in *DEFINE_CURVE_FUNCTION for shells.
- Output stresses for all 4 intg points to eloutdet for cohesive element types 19 & 20.
- Fix incorrect rotational displacement to nodout when REF = 2 in *DATABASE_HISTORY_NODE_LOCAL. Affects MPP only.
- Fix incorrect strains output to elout for shell type 5 and when NIP > 1.
- Fix incorrect acceleration output to nodout file when IACCOP = 1 in *CONTROL_OUTPUT and IGRAV = 1 in *ELEMENT_SEATBELT_ACCELEROMETER.
- Fix corrupted d3plot when RESPLT = 1 in *DATABASE_EXTENT_BINARY and idrflg.ge.5 in *CONTROL_DYNAMIC_RELAXATION.
- Fix missing element connectivities in nastin file when using *INTERFACE_SPRINGBACK_NASTRAN_NOTHICKNESS.
- Fix seg fault when using *DATABASE_BINARY_D3PART with *CONTACT_TIED_SHELL_EDGE_TO_SURFACE. This affects SMP only.
- Fix incorrect output to bndout when using multiple *LOAD_NODE_POINT for the same node and running MPP with ncpu > 1.
- Fix incorrect dyna.inc file when using *MAT_FU_CHANG_FOAM/MAT_83, *DEFINE_COORDINATE_NODES, and *CON-

STRAINED_JOINT_STIFFNESS_GENERALIZED with *INCLUDE_-TRANSFORM.

- Fix IEVERP in *DATABASE_EXTENT_D3PART which was not honored in writing out d3part files.
- Fix incorrect stresses written out to dynain for thick shells with formulations 1,2 and 4.
- Fix incorrect output to disbout data for discrete beams.
- Fix incorrect output to binary format of disbout. Affects SMP only.
- Fix error when writing initial stresses for thick shells to dynain. Affects MPP only.
- Fix thick shells strain output to dynain.
- Fix incorrect writing of material data to dyna.str for *MAT_SEATBELT when using long = s.
- Fix coordinate/disp output to d3plot of *CONSTRAINED_NODAL_-RIGID_BODY's pnode.
- Fixed the initial d3plot state in SMP runs when tied contact is used with theCNTCO parameter on *CONTROL_SHELL. The geometry was wrong in that state.
- Add cross section forces output (*DATABASE_SECFORC) for cohesive elements ELFORM type 19, 20, 21, and 22.
- Slight increase of precision for values in nodout file.
- Add new option FSPLIT to *INTERFACE_SPRINGBACK_LSDYNA to split the dynain file into two files (geometry and initial values).
- *DEFINE_MATERIAL_HISTORIES: New keyword for organizing material history outputs, currently only for solids, shells and beams and the d3plot output but to be extended to tshells and ascii/binout. The purpose is to customize the history variables that otherwise are output via NEIPS/NEIPH/NEIPB on *DATABASE_EXTENT_BINARY, to avoid variable conflict and large d3plots and thus facilitate post-processing of these variables. Currently available in small scale but to be continuously extended.
- Fixed bug affecting IBINARY = 1 (32 bit ieee format) in *DATABASE_FORMAT. This option was not working.
- Fixed incorrect printout of node ID for *ELEMENT_INERTIA.
- Increased the header length to 80 for the following files in binout: matsum, nodout, spcforc, ncforc
- Fixed bug in which d3msg was not written for SMP.
- The d3plot output for rigid surface contact was incorrect for MPP.
- Fixed bugs when using curve LCDT to control d3plot output.
- Fixed abnormal increase in d3plot size caused by outputting velocity and acceleration when data compression is on.
- Added new variable GEOM in *CONTROL_OUTPUT for choosing geometry or displacement in d3plot, d3part, and d3drlf.

- Added command line option "msg=" to output warning/error descriptions. See MSGFLG in *CONTROL_OUTPUT for alternate method of requesting such output. Accepted values for "msg=" are message# or all.

  - message#, e.g., KEY+101 or 10101. This option will print the error/warning message to the screen.
  - all. this option will print all error/warning messages to d3msg file.

- Fixed bug for *DATABASE_BINARY_D3PROP file if adaptivity used. The error caused blank d3prop output.
- *DATABASE_HISTORY_SHELL_SET combined with *CONTROL_ADAPTIVITY caused error 20211. The error involves the BOX option being used for shell history output.
- Added *INTEGRATION... data to d3prop.

- **Restarts**

  - Fix bug when deleted uniform pressure (UP) airbag during simple restart.
  - Fix for index error that could cause problems for accelerometers during full deck restart in MPP.
  - Fix for MPP output of LSDA interface linking file when restarting from a dump file.
  - Fix incorrect strains in d3plot after restart when STRLG > 1.
  - Fix incorrect velocity initialization for SMP full deck restart when using
  - *INITIAL_VELOCITY_GENERATION and *INITIAL_VELOCITY_GENERATION_START_TIME.
  - Fix incorrect behavior of *CONTACT_ENTITY in full deck restart.
  - Fix incorrect full deck restart analysis if initial run was implicit and the full deck restart run is explicit.
  - Fix ineffective boundary condition for *MAT_RIGID when using *CHANGE_RIGID_BODY_CONSTRAINT with *RIGID_DEFORMABLE_-R2D for small deck restart.
  - Fix initialization of velocities of *MAT_RIGID_DISCRETE nodes after restart using *CHANGE_VELOCITY_GENERATION.
  - Fix internal energy oscillation after full deck restart when using *CONTACT_TIED_SURFACE_TO_SURFACE_OFFSET with TIEDID = 1 in optional card D. This affects SMP only.
  - Corrected bug affecting full restart that included any change to node/element IDs. This bug has existed since version R6.
  - Fixed bug affecting d3plot times following fulldeck restart with curve in SMP.
  - Fixed bug in simple restart: *INTERFACE_COMPONENT_FILE forgets the filename and writes to infmak instead.

- ***SENSOR**

  - Enable full restart for *SENSOR.
  - Add optional filter ID to SENSORD of *DEFINE_CURVE_FUNCTION.
  - Enable LOCAL option of *CONSTRAINED_JOINT to be used with *SENSOR_DEFINE_FORCE.
  - Fix a MPP bug that happens when *SENSOR_DEFINE_NODE has a defined N2.
  - *SENSOR_CONTROL:

    - Fix a bug for TYPE = JOINTSTIF
    - Fix an MPP bug for TYPE = PRESC-MOT when the node subject to prescribed motion is part of a rigid body
    - Add TYPE = BELTSLIP to control the lockup of *ELEMENT_SEATBELT_SLIPRING.
    - Add TYPE = DISC-ELES to delete a set of discrete elements.

  - Add FTYPE = CONTACT2D to *SENSOR_DEFINE_FORCE to track the force from *CONTACT_2D.
  - Add the variable SETOPT for *SENSOR_DEFINE_NODE_SET and *SENSOR_DEFINE_ELEMENT_SET to sense and process data from a node set or element set, resp., resulting in a single reported value.
  - *SENSOR can be used to control *CONTACT_GUIDED_CABLE.
  - Fix a bug related to *SENSOR_DEFINE_FUNCTION triggered by more than 10 sensor definitions.

- **SPG (Smooth Particle Galerkin)**

  - *SECTION_SOLID_SPG (KERNEL = 1): The dilation parameters (DX, DY, DZ) of SPG Eulerian kernel are automatically adjusted according to the local material deformation to prevent tensile instability.

- **SPH (Smooth Particle Hydrodynamics)**

  - Retain user IDs of SPH particles in order to ensure consistent results when changing the order of include files.
  - Add feature to inject SPH particles, *DEFINE_SPH_INJECTION.
  - Added support of various material models for 2D and 3D SPH particles:

    - *MAT_098 (*MAT_SIMPLIFIED_JOHNSON_COOK)
    - *MAT_181 (*MAT_SIMPLIFIED_RUBBER)
    - *MAT_275 (*MAT_SMOOTH_VISCOELASTIC_VISCOPLASTIC)

  - Added support of *DEFINE_ADAPTIVE_SOLID_TO_SPH for 2D shell elements and 2D axisymmetric shell elements.

- When using *DEFINE_ADAPTIVE_SOLID_TO_SPH, eliminated duplicate kinetic energy calculation for SPH hybrid elements (both SPH particles and solid elements contributed kinetic energy into global kinetic energy).
- Added support of second order stress update (OSU = 1 in *CONTROL_ACCURACY keyword) for 2D and 3D SPH particles. This is necessary for simulation of spinning parts.
- Added ISYMP option in *CONTROL_SPH to define as a percentage of original SPH particles the amount of memory allocated for generation of SPH ghost nodes used in *BOUNDARY_SPH_SYMMETRY_PLANE.
- Fixed unsupported part and part set option in *BOUNDARY_SPH_FLOW.
- Fixed unsupported ICONT option from *CONTROL_SPH when combined with *BOUNDARY_SPH_FLOW.
- *DEFINE_SPH_TO_SPH_COUPLING: Output contact forces between two SPH parts (x, y, z and resultant forces) into sphout. The forces can be plotted by LS-PrePost.
- *CONTACT_2D_NODE_TO_SOLID: Added bucket sort searching algorithm to speed up the process of finding contact pairs between SPH particles and solid segments.

• **Thermal**

- Corrected a long standing bug in MPP thermal associated with spotwelds (*CONSTRAINED_SPOTWELD) using thermal linear solver option 11 or greater. The spotweld loads were not being loaded correctly due to an indexing issue in MPP.
- Fix for thermal with *CASE.
- Fix MPP support for thermal friction in SOFT = 4 contact.
- Fixed bug where thermal solver gives a non-zero residual even though no loads are present.
- Added SOLVER = 17 (GMRES solver) to *CONTROL_THERMAL_SOLVER for the conjugate heat transfer problem. The GMRES solver has been developed as an alternative to the direct solvers in cases where the structural thermal problem is coupled with the fluid thermal problem in a monolithic approach using the ICFD solver. A significant savings of calculation time can be observed when the problem reaches 1M elements. This solver is implemented for both SMP and MPP.
- *CONTACT_(option)_THERMAL (3D contact only): Add variable FTOSLV to specify fraction of frictional energy applied to slave surface. It follows that 1.-FTOSLV is applied to master surface. Default is 0.5 which gives a 50% - 50% split between the slave and master surfaces which was hardwired in prior releases.
- First release of AUTOMATIC_SURFACE_TO_SURFACE_TIED_WELD_-THERMAL. This will only work when used with BOUNDARY_THERMAL_WELD. This combination of keywords will activate a condition where sliding contact will become tied contact on cooldown when the temperature

of the segments in contact go above an input specified temperature limit during welding.
- *LOAD_THERMAL_D3PLOT: The d3plot data base was changed such that the 1st family member contains control words, geometry, and other control entities. Time state data begins in the 2nd family member. This change allows the new d3plot data structure to be read in by LS-DYNA when using the *LOAD_THERMAL_D3PLOT keyword. This change is not backward compatible. The old d3plot data structure will no longer be read correctly by LS-DYNA.
- Synchronize data in TPRINT for SMP and MPP:
  - Fixed output to tprint/binout for thermal contact.
  - Fixed part IDs for part energies.
  - Fixed format of TPRINT file generated by l2a.
- Fixed handling of start time defined with *CONTROL_START for thermal solver.
- Change the maximum number of *LOAD_HEAT_CONTROLLER definitions from 10 to 20.
- Added a third parameter to the TIED_WELD contact option. The parameter specifies heat transfer coefficient h_contweld for the welded contact. Before welding, the parameter from the standard card of the thermal contact is used.
- Parameter FRCENG supported for mortar contact to yield heat in coupled thermomechanical problems.

- **XFEM (eXtended Finite Element Method)**
  - Added ductile failure to XFEM using critical effective plastic strain as failure criterion.

- **Miscellaneous**
  - Support *SET_NODE_GENERAL PART with SPH and DES.
  - *DEFINE_POROUS_...: Compute the coefficients A and B with a user defined routine in dyn21.F.
  - Fixed bugs in Staged Construction (*DEFINE_STAGED_CONSTRUCTION_PART):
    - Staged construction not working on SMP parallel. Symptoms could include wrong elements being deleted.
    - Staged construction with beam elements of ELFORM = 2: when these beams are dormant, they could still control the time step.
    - Staged construction with *PART_COMPOSITE. The bug occurred when different material types were used for different layers within the same part, and that part becomes active during the analysis. The

symptom of the bug was that stresses and/or history variables were not set to zero when the part becomes active.

- Bugs fixed in *DAMPING_FREQUENCY_RANGE_DEFORM:
  - Incorrect results when large rigid body rotations occur.
  - If RYLEN on *CONTROL_ENERGY = 2, the energy associated with this damping should be included in the Internal Energy for the relevant part(s). This energy was being calculated only if there was also *DAMPING_PART_STIFFNESS in the model. Now fixed - the damping energy will be included in the internal energy whenever RYLEN = 2.
- Fixed NID option of *DEFINE_COORDINATE_VECTOR (bug occurred in MPP only).
- Fix lsda open mode to require only minimal permissions to avoid unnecessary errors, for example if using an interface linking file that is read only.
- Fix for DPART processing (*SET_..._GENERAL) for solid and thick shell elements.
- Fix for JOBID > 63 characters.
- Fix input processing problem (hang) that could happen in some unusual cases if encrypted *INCLUDE files are used.
- Fix interaction of *CASE with jobid = on command line, so the jobid on the command line is combined with the generated case ids instead of being ignored.
- *INCLUDE_NASTRAN:
  - Integration defaults to Lobatto for Nastran translator.
  - The default number of integration points is set to 5 for Nastran translator.
- Issue error message and terminate when illegal *DEFINE_TRANSFORMATION is specified.
- Add OPTION = POS6N to *DEFINE_TRANSFORMATION to define transformation with 3 reference nodes and 3 target nodes.
- Add OPTION = MIRROR to *DEFINE_TRANSFORMATION.
- Fix a bug that could occur when adapted elements are defined in a file included by *INCLUDE_TRANSFORM.
- Fix a bug that could occur when *BOUNDARY_SPC_SYMMETRIC_PLANE is used together with *INCLUDE_TRANSFORM.
- Fix a bug that occurs when *DEFINE_BOX is included by *INCLUDE_TRANSFORM.
- Make *SET_NODE_COLLECT work together with *NODE_SET_MERGE.
- Fix incorrect shell set generated when using *SET_SHELL_GENERAL with OPTION = PART.

- Add error trap for *SET_PART_LIST_GENERATE_COLLECT to catch missing part IDs.
- Fixed bug in *INCLUDE_TRANSFORM for adaptive case if JOBID is used.
- Fixed bug in memory allocation for *DEFINE_CURVE if total number of points in curve is more than 100.
- Fixed bug with *INCLUDE_TRANSFORM and *CONTROL_ADAPTIVITY due to an *INCLUDE inside *INCLUDE_TRANSFORM file. Added new files: adapt.inc# for *INCLUDE_TRANSFORM file. The *NODE, *ELEMENT_SHELL and *ELEMENT_SOLID are removed from include file.
- Fixed bug for DPART option in *SET_SEGMENT_GENERAL. DPART option was treated as PART option before.
- Fixed failure of *PARAMETER definition in long format.
- Fixed error in reading solid id for *SET_SOLID_GENERAL.
- Ignore any nonexistant part set IDs in *SET_PART_ADD.
- Fix bug in which sense switches sw2 and sw4 don't work when the output interval for glstat is small.
- Fixed bug if *DEFINE_CURVE is used to define adaptivity level.
- Three new keywords are implemented in support of user defined subroutines: *MODULE_PATH[_RELATIVE], MODULE_LOAD, MODULE_USE.

  - The MODULE feature allows users to compile user subroutines into dynamic libraries without linking to the LS-DYNA main executable.
  - The dynamic libraries are independent from the main executable and do not need to be recompiled or linked if the main executable is updated.
  - This feature loads multiple dynamic libraries on demand as specified in the keywords.
  - Without the MODULE feature, only one version of each umat (such as umat41) can be implemented. With the MODULE feature, most umat subroutines can be have multiple versions in multiple dynamic libraries, and used simultaneously.
  - The MODULE feature supports all user subroutines.
  - The LS-DYNA main executable may also run without any dynamic libraries if no user subroutines are required.

Capabilities added to create LS-DYNA R10:

See release notes (published separately) for further details.

- ***AIRBAG**

  - Enhance the robustness of *AIRBAG_INTERACTION to help avoid instability in MPP when the interaction involves more than two bags.
  - *AIRBAG_PARTICLE:

        - Adjust dm_out calculation of vent hole to avoid truncation error.
        - Fix bug in chamber output when there are multiple airbags and multiple chambers not in sequential order.
        - Bug fix for closed volume of airbag/chamber with intersecting tubes.
        - Add new feature to allow user to define local coordinates of jetting of particles through internal vents.
        - Support *SENSOR_CONTROL for CPM airbag.
        - CPM is not supported for dynamic relaxation. Disable CPM airbag feature during DR and reactivate airbag following DR.
        - Allow solid parts in definition of internal part set. The solid volume will be excluded from the airbag volume.
        - Allow additional internal part set for shells. The shell part should form a closed volume and its volume will be excluded from the airbag volume.

- ***ALE**

    - *LOAD_BLAST_SEGMENT: Automatically generate the ALE ambient elements attached to a segment or segment set.
    - *BOUNDARY_AMBIENT_EOS: implement *DEFINE_CURVE_FUNCTION for the internal energy and relative volume curves.
    - *CONTROL_ALE, *CONSTRAINED_LAGRANGE_IN_SOLID and *ALE_REFERENCE_SYSTEM: If NBKT < 0 in *CONTROL_ALE, call *DEFINE_CURVE to load a curve defining the number of cycles between bucket sorting in function of time. If NBKT > 0, the bucket sorting is activated if the mesh rotations and deformations are large.
    - *ALE_FSI_TO_LOAD_NODE: Implement a mapping of the FSI accelerations (penalty forces/masses) computed by *CONSTRAINED_LAGRANGE_IN_SOLID (ctype = 4) between different meshes.
    - DATABASE_FSI, *DATABASE_BINARY_FSIFOR and *DATABASE_BINARY_FSILNK: Add a parameter CID to output fsi forces in a local coordinate system.
    - Structured ALE (S-ALE) solver:

        - ALE models using rectilinear mesh can be directly converted to S-ALE models and run using S-ALE solver by assigning CPIDX = -1 in *ALE_STRUCTURED_MESH.
        - S-ALE progressive mesh generation via RATIO in *ALE_STRUCTURED_MESH_CONTROL_POINTS.

    - Recode ALE Donor Cell/Van Leer advection routines and restructure *CONSTRAINED_LAGRANGE_IN_SOLID communication algorithm. These give 30% improvement in run time.

- ***BOUNDARY**

- *BOUNDARY_PWP can now accept a *DEFINE_FUNCTION instead of a load curve. The input arguments are the same as for *LOAD_SEGMENT: (time, x, y, z, x0, y0, z0).
- Add option of "toffset" for *BOUNDARY_PRESCRIBED_ORIENTATION_-RIGID to offset the curves by the birth time.
- MPP now supports MCOL coupling, *BOUNDARY_MCOL.
- Fix bug of there being fully constrained motion of a rigid part when prescribing more than one translational dof with *BOUNDARY_PRESCRIBED_MOTION_RIGID while con2 = 7 in *MAT_RIGID, meaning all rotational dof are constrained.
- Instead of error terminating with warning message, STR+1371, when *BOUNDARY_PRESCRIBED_MOTION and *BOUNDARY_SPC is applied to same node and dof, issue warning message, KEY+1106, and release the conflicting SPC.
- Fix erroneous results if SET_BOX option is used for *BOUNDARY_PRESCRIBED_MOTION.
- Fix *BOUNDARY_PRESCRIBED_ACCELEROMETER_RIGID for MPP. It may error terminate or give wrong results if more than one of this keyword is used.
- Fix segmentation fault when using *BOUNDARY_PRESCRIBED_ORIENTATION with vad = 2, i.e. cubic spline interpolation.
- Added instruction *BOUNDARY_ACOUSTIC_IMPEDANCE for explicit calculations that applies an impedance boundary condition to the boundary of *MAT_ACOUSTIC element faces. This is a generalization of the non-reflecting boundary condition. Both *LOAD and *BOUNDARY_ACOUSTIC_IMPEDANCE may be used on the same faces, in which case the boundary acts like both and entrant and exit boundary.
- Fixed a problem with non-reflecting boundaries redefining the bulk modulus which caused contact to change behavior.
- Added support for acoustic materials with non-reflective boundaries.
- Fix the single precision version so that *INCLUDE_UNITCELL now has no problem to identify pairs of nodes in periodic boundaries.
- When using *INCLUDE_UNITCELL to generate Periodic Boundary Constraints (PBC) for an existing mesh, a new include file with PBCs is generated instead of changing the original mesh input file. For example, if users include a file named "mesh.k" through *INCLUDE_UNITCELL (INPT = 0), a new include file named "uc_mesh.k" is generated where all PBCs are defined automatically following the original model information in mesh.k.
- *INCLUDE_UNITCELL now supports long input format in defining the element IDs.
- Include SPC boundary conditions as part of H8TOH20 solid element conversion.
- Add a new option SET_LINE to *BOUNDARY_PRESCRIBED_MOTION: This option allows a node set to be generated including existing nodes and new nodes created from h-adaptive mesh refinement along the straight line

connecting two specified nodes to be included in prescribed boundary conditions.

- **BLAST**
  - *PARTICLE_BLAST and DES:
    - Consider eroding of shell and solid in particle_blast.
    - Support interface force file output for gas particle-structure coupling.
    - Bug fix for wet DES coupled with beam.
    - Support *SET_NODE_GENERAL PART with SPH or DES.
    - MPP now uses async communication for DES coupling to improve general performance.
    - Support for solid element when modeling irregular shaped charge with HECTYPE = 0/1 in *PARTICLE_BLAST.
    - Output adaptive generated DES and NODE to a keyword file.
  - Fix inadvertent detonation of HE part when there are more than one HE part and even though the HE part is not defined with *INITIAL_DETONATION.
  - Fixed explicit *BOUNDARY_USA_COUPLING to support *INITIAL_-STRESS and *INITIAL_STRAIN_ usage, typically from a dynain file.
  - Fixed explicit *BOUNDARY_USA_COUPLING to support *CONTROL_-DYNAMIC_RELAXATION IDRFLF = 5, so a static implicit calculation can be used to initialize/preload a model before conducting an explicit transient calculation. If inertia relief is used during the static phase, then it must be disabled with *CONTROL_IM-PLICIT_INERTIA_RELIEF for the explicit phase.
  - Support imperial unit system for *PARTICLE_BLAST. mass = lbf-s^2/in, length = inch, time = second, force = lbf, pressure = psi.
  - Add option to define detonation point using a node for *PARTICLE_BLAST.
  - Add interface force file output for *PARTICLE_BLAST with keyword *DATABASE_BINARY_PBMFOR and command line option "pbm=". This output of forces for gas-particle-structure coupling.
  - For *PARTICLE_BLAST, add built-in smoothing function for particle structure interaction.
  - For *PARTICLE_BLAST, when coupling with DEM, the DEM nodes that are inside HE domain are automatically deactivated.
  - Add support for solid elements when modeling irregular shaped charge with HECTYPE = 0/1 for *PARTICLE_BLAST. The original approach only supports shell elements and the initial coordinates of HE particle are at shell surface. The model had to relax several hundred time step to let particle fill in the interior space, which was not convenient. Using new approach, the initial positions of HE particles are randomly distributed inside the container by using solid element geometry. Both hex and Tet solids are supported.

- For particle blast method (PBM), consider reflecting plane as infinite.
- Change the name of keyword *DEFINE_PBLAST_GEOMETRY to *DEFINE_PBLAST_HEGEO.

- ***CESE (Compressible Fluid Solver)**

  - CESE time steps:

    - Modified the blast wave boundary condition treatment to make it more stable in blast wave calculations (with *LOAD_BLAST_ENHANCED).
    - The flow field calculation will be skipped if the structural time-step is much smaller than the fluid time step, until both time-steps reach the same order. This will save CPU time in some fluid/structure interaction (FSI) problem calculations.
    - In addition to depending upon the local CFL number, the fluid time step 'dt' calculation has been modified to also adjust dynamically to extreme flow conditions. This makes stiff flow problems more stable especially in 3D fluid problem calculations when the mesh quality is poor.

  - Moving mesh solvers:

    - Corrected several aspects of the implicit ball-vertex (BV) mesh motion solver for the following keywords:
      *ICFD_CONTROL_MESH_MOV
      *CESE_CONTROL_MESH_MOV.
    - The absolute tolerance argument is no longer used by the BV solver. As an example, the following is all that is needed for CESE moving mesh problems:
      ```
      *CESE_CONTROL_MESH_MOV
       $    ialg  numiter   reltol
              1       500   1.0e-4
      ```
    - Also corrected the CESE moving mesh solvers for a special case involving a wedge element. Also, fixed the d3plot output of wedge element connectivities for the CESE moving mesh solvers.

  - CESE d3plot output:

    - Added real 2D CESE output, and this is confirmed to work with LSP-P4.3 and later versions. This also works for d3plot output with the 2D CESE axisymmetric solver.
    - For all immersed-boundary CESE solvers, corrected the plotting of the Schlieren number and the chemical species mass fractions.
    - The following new CESE input cards are related to surface d3plot output:
      *CESE_SURFACE_MECHSSID_D3PLOT

*CESE_SURFACE_MECHVARS_D3PLOT
In conjunction with the above, new FSI and conjugate heat transfer output on solid (volume) mesh outside boundaries is now supported.

- CESE immersed-boundary method (IBM) FSI solvers:

  - *CESE_FSI_EXCLUDE is a new keyword for use with the CESE immersed boundary method FSI solvers. With it, unnecessary structural parts that are not actively participating in the FSI in the CESE IBM-FSI solver can now be excluded from the CESE FSI calculation. This is also supported for the case when some of the mechanics parts involve element erosion.

- CESE chemistry solvers:

  - In R10, we also updated several things in the FSI solver with chemistry called FSIC. In chemical reacting flow, a delta time between iterations is extremely important for code stabilization and thus, to get reasonable results. To this end, we optimized such an iterative delta time, which is based on the CFL number. This optimization is based on the gradient of the local pressure, which we think will dominate control of the CFL number.
  - Next, the total number of species are increased up to 60 species in chemical reacting flow, so that the reduced Ethylene (24~53 species) and Methane (20~60 species) combustion are possible with this version.
  - We will update more practical examples about FSIC problems including precise experimental validations.
  - Note that we can provide some related examples upon user request.
  - Other corrections of note include the following:
    Brought in enthalpy-related corrections to the CESE chemistry solvers.
    Fixed the conjugate heat transfer boundary condition for the 2D and 3D CESE fixed mesh chemistry solvers.
    Corrected the initialization of fluid pressure for CESE IBM chemistry solvers.
    Enabled output of the timing information for the CESE chemistry solvers.
    Added restart capability to the CESE chemistry solvers.

- ***CHEMISTRY**

  - New inflator models of Pyrotechnic and Hybrid type are updated. It is important to note that these are basically 0-dimensional models via the following two main keywords,
    *CHEMISTRY CONTROL_INFLATOR
    *CHEMISTRY_INFLATOR_PROPERTIES

- By using the *CHEMISTRY CONTROL_INFLATOR keyword, the user can select the type of the solver, output mode, running time, delta t, and time interval for output of time history data.
  For example, if we have a keyword set up as,
  *CHEMISTRY CONTROL_INFLATOR,
  ```
  $ isolver  ioutput  runtime    delt   p_time
           1          0        0.1  1.0e-6    5.0e-4
  ```
  with "isolver set to 1", the user can simulate a conventional Pyrotechnic inflator mode, while with "isolver" set to 2 or 3, Hybrid inflator simulation is possible.
- In addition, to continue an airbag simulation via an ALE or CPM method, the user can save the corresponding input data file by using "ioutput" option. For more details about airbag simulations using a saved data file, refer to the keyword manual.
- Also, note that the updated version has two options for the Hybrid models:
  - isolver = 2 => Hybrid model for the cold flow
  - isolver = 3 => Hybrid model for the heated flow.
- In the *CHEMISTRY_INFLATOR_PROPERTIES keyword, there are several cards to set up the required properties of an inflator model. The first two cards are for the propellant properties involved in inflator combustion.
  For example,
  ```
  $card1: propellants
  $ comp_id    p_dia  p_height   p_mass   p_tmass
            10     0.003     0.0013      2.0e-5   5.425e-3
  $card2: control parameters
  $ t_flame   pindex      A0    trise    rconst
        2473.        0.4  4.45e-5      0.0      0.037
  ```
  In the first card, the user can specify the total amount of propellant particles and their shape.
  Using the second card, the user can also specify the thermodynamics of the propellant and its burning rate.
  To support the options in card2, especially the second option, pindex, and the third, A0, we provide a standalone program upon request for the propellant equilibrium simulation.
  The remaining cards are for the combustion chamber, gas chamber, and airbag, respectively.

- ***CONTACT**
  - *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR_TIED_-WELD for modeling welding has been added. Surfaces are tied based on

meeting temperature and proximity criteria. Non-MORTAR version of this contact was introduced at R9.0.1.

- Fix issue setting contact thickness for rigid shells in ERODING contact.
- Add MPP support for *CONTACT_AUTOMATIC_GENERAL with adaptivity.
- Change "Interface Pressure" report in intfor file from abs (force/area) to -force/area, which gives the proper sign in case of a tied interface in tension.
- Rework input processing so that more than one *CONTACT_INTERIOR may be used, and there can be multiple part sets in each one.
- Minor change to how pressure is computed for triangles in the intfor database.
- Fix 2 bugs for contact involving high order shell elements:
  - When high order shell elements are generated by SHL4_TO_SHL8.
  - When using a large part ID like 100000001.
- Implement a split-pinball based contact option for neighbor elements in segment-based contact. Invoke this option by setting |SFNBR|>=1000. The new algorithm is more compatible with DEPTH = 45 so that there is no longer a need to split quads.
- The effect of shell reference system offsets on contact surface location is now properly considered when running MPP. The shell offset may be specified using NLOC in *SECTION_SHELL or in *PART_COMPOSITE, or by using the OFFSET option of *ELEMENT_SHELL. This effect on contact is only considered when CNTCO is set to 1 or 2 in *CONTROL_SHELL.
- Fix bug of zero forces in rcforc at time = 0.0 for *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE after dynamic relaxation when consistency is on in SMP.
- Fix input error when using many *RIGIDWALL_GEOMETRIC_... with _DISPLAY option.
- Fix input error when *CONTACT_ENTITY is attached to a beam part, PID.
- Fix error termination due to negative volume, SOL+509, even when *CONTACT_ERODING... is set. This affects MPP only.
- Check whether a slave/master node belongs to a shell before updating the nodal thickness when ISTUPD > 0.0 in *CONTROL_SHELL and SST/MST.ne.0.0 and in SSFT/SMFT = 0.0 card 3 of *CONTACT_..... For SMP only.
- Fix penetrating nodes when using *CONTACT_ERODING_-NODES_TO_-SURFACE with SOFT = 1 in *MAT_142/*MAT_
- Fix seg fault when using *CONTACT_AUTOMATIC_-SINGLE_SURFACE_TIED with consistency mode, .i.e. ncpu < 0, for SMP.
- Fix corrupted intfor when using parts/part sets in *CONTACT_AUTOMATIC_....This affects SMP only.
- Implement incremental update of normal option, invoked by TIEDID = 1, for *CONTACT_TIED_NODES_TO_SURFACE_CONSTRAINED_OFFSET for SMP.

- Fix unconstrained nodes when using *CONTACT_TIED_-SURFACE_TO_-SURFACE_CONSTRAINED_OFFSET resulting in warning message, SOL+540. This affects SMP only.
- Fix spurious repositioning of nodes when using *CONTACT_-SURFACE_-TO_SURFACE for SMP.
- Added support to segment-based contact for the SRNDE parameter on optional card E. This option allows round edge extensions that do not extend beyond shell edges and also square edges. The latter overlaps with the SHLEDG parameter on card D.
- Fixed a potential memory error that could occur during segment-based contact input.
- Fixed an error that could cause an MPP job to hang in phase 3. The error could occur when SOFT = 2 contact is used with the periodic intersection check and process 0 does not participate in the contact.
- Modified SOFT = 2 contact friction when used with *PART_CONTACT to define friction coefficients, and the two parts in contact have different coefficient values. With this change, the mu values used for contact will be the average of the values that are calculated for each part. Prior to this change, mu was calculated for only the part that is judged to be the master. This change makes the behavior more predictable and also makes it behave like the other contacts with SOFT = 0 and SOFT = 1.
- Added a warning message (STR+1392) for when trying to use the ORTHO_-FRICTION contact option with SOFT = 2 contact, because that option is not available. The contact type is switched to SOFT = 1.
- Fixed serious error in MPP *CONTACT_2D_AUTOMATIC_-SURFACE_-TO_SURFACE when used with node sets to define the contact surfaces. The master side was likely to trigger a spurious error about missing nodes that terminated the job.
- Switched segment based (SOFT = 2) non-eroding contact to prevent it from adding any new segments when brick element faces are exposed when other elements are deleted. There were two problems. The first is that the interface force file could not support NFAIL = 1 on *DATA-BASE_EXTENT_-INTFOR because the intfor file does not expect new segments to replace the old, so it just undeletes the old segments instead of adding the new. The second problem is that when non-eroding contact is used, we only have enough memory in fixed length arrays for the segments that exist at t = 0. When segments are deleted, I was using the space that they vacated to create new segments, but it was very likely that some segments could not be created when the number of open spaces was less than the number of new segments that are needed. In this case, some segments would not be created and there would be surfaces that could be penetrated with no resistance. This behavior is impossible to predict, so it seems better to prevent any new segments from being created unless eroding contact is used.
- Fixed rcforc output for MPP 2D automatic contact. The forces across processors were missed.

- Fixed a segment-based contact error in checking airbag segments. This affects only airbags that are defined by control volumes, that is defined by *AIRBAG. The symptom was a segmentation fault.
- Fixed SMP eroding segment based (SOFT = 2) contact which was not activating the negative volume checking of brick elements. The MPP contact and the other SMP contacts were doing this but not SMP SOFT = 2.
- Fixed support for CNTCO on *CONTROL_SHELL by segment based (SOFT = 2) contact. It was adjusting the contact surface only half of what it should have done.
- Fixed eroding segment-based contact when used with the CNTCO > 0 on *CONTROL_CONTACT. A segmentation fault was occurring.
- Modified MPP segment based (soft = 2) contact to use R8 buffers to pass nodal coordinates. This should reduce MPP scatter when decomposition changes.
- Added support for using a box to limit the contact segments to those initially in the box when using eroding segment-based contact. The box option has not been available for any eroding contact up until now. (SOFT = 2 and SBOXID, MBOXID on *CONTACT_ERODING_...).
- Fixed force transducers with MPP segment-based contact when segments are involved with multiple, 2-surface force transducers. The symptom was that some forces were missed for contact between segments on different partitions.
- Added support for *ELEMENT_SOURCE_SINK used with segment-based contact. With this update, inactive elements are no longer checked for contact.
- Fixed an MPP problem in segment-based contact that cased a divide by zero during the bucket sort. During an iteration of the bucket sort, all active segments were somehow in one plane which was far from the origin such that a dimension rounded to zero. The fix for this should affect only this rare case and have no effect on most models.
- Modified segment based (SOFT = 2) contact to make SMP and hybrid faster, particularly for larger numbers of processors.
- Fixed thermal MPP segment-based contact. The message passing of thermal energy due to friction was being skipped unless peak force data was written to the intfor file.
- Fixed likely memory errors in MPP problems with 2D automatic contact when friction is used.
- Support the VC parameter (coefficient for viscous friction) in the case of segment-based contact, which has previously been unsupported. This option will work best with FNLSCL > 0, DNLSCL = 0 on optional card D. The card D option causes the contact force to be proportional to the overlap area which causes even pressure distribution.
- Enabled segment-based contact (SMP and MPP) to work with type 24 (27-node) brick elements.

- Fixed MPP segment-based contact for implicit solutions. During a line search, some data was not restored correctly when the solver goes back to the last converged state. This caused possible memory errors.
- Fixed friction for MPP segment-based contact in the implicit solver. The sliding velocity was calculated incorrectly using the explicit time step rather than the implicit step.
- Fixed a bug in MPP *CONTACT_2D_AUTOMATIC..., where a flaw in code used during MPP initialization could cause segments to fail to detect penetration.
- Fixed the thick beam checking of *CONTACT_2D_AUTO-MATIC_SINGLE_SURFACE in the MPP version. There was a memory error that could occur if thick beams were in the model.
- New values for user friction element history (*USER_INTERFACE_FRICTION): material directions, relative velocity components and yield stress.
- Add new user-defined interface for tiebreak contact invoked by *CONTACT_-AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK_USER.
- MORTAR CONTACT

  - PENMAX and SLDTHK has taken over the meanings of SST and TKSLS in R9 and earlier, although in a different way. Now PENMAX corresponds to the maximum penetration depth for solid elements (if nonzero, otherwise it is a characteristic length). SLDTHK is used to offset the contact surface from the physical surface of the solid element, instead of playing with SST and TKSLS, which was rather awkward. This update also saves the pain of having to treat shells and solids in separate interfaces if these features are wanted. This changes the behavior in some inputs that did have SST turned on for solids, but a necessary measure to make the contact decent for future versions.
  - The characteristic length for solid elements has been revised to not result in too small sizes that would lead to high contact stiffnesses and less margin for maximum penetration.
  - SFS on CONTACT_..._MORTAR can be input as negative, then contact pressure is the -SFS load curve value vs penetration.
  - Smooth roundoffs of sharp edges in MORTAR contact has been extended to high order segments, meaning that edge contact is valid even in this case.
  - The MORTAR contact now honors the NLOC parameter for shells, see *SECTION_SHELL, adjusting the contact geometry accordingly. Note that CNTCO on *CONTROL_SHELL applies as if always active, meaning that if NLOC is on, then CNTCO will also be "on" for MORTAR contacts.
  - Output of contact gaps to the intfor file is now supported for MORTAR contact, see *DATABASE_EXTENT_INTFOR.

  - Transducer contacts, *CONTACT_..._FORCE_TRANSDUCER, are supported for MORTAR contact in SMP and MPP. A disclaimer is that the slave and master sets in the transducer have to be defined through parts or part sets. Warnings are issued if this is violated.
  - Option 2 is now supported for tiebreak MORTAR contact, *CONTACT_..._MORTAR_TIEBREAK, but only for small sliding. Options 4 and 7 are supported in the MORTAR tiebreak contact for any type of sliding.
  - For explicit analysis, the bucket sort frequency for MORTAR contact is 100, but can be changed through parameter BSORT on the CONTACT_..._MORTAR card or NSBCS on CONTROL_CONTACT. Note that the MPP bucket sort parameter does not apply. This assumes to improve the efficiency of MORTAR explicit contact significantly compared to R9 and earlier versions.
  - Dynamic friction is supported in MORTAR contact for explicit and implicit dynamic analysis. See FD and DC on *CONTACT_... card.
  - Wear calculations are supported for the MORTAR contact. See CONTACT_ADD_WEAR.
  - Triangular shell form 24 is supported with MORTAR forming contact and accounts for high order shape functions.
  - Automatic MORTAR contact now supports contact with end faces of beam elements and not just the lateral surfaces.
  - Mortar contact is available in 2D plane strain and axisymmetric simulations, but only for SMP implicit. See CONTACT_2D_...MORTAR.

- Wear computed from *CONTACT_ADD_WEAR can optionally be output to dynain on optional card of *INTERFACE_SPRINGBACK_LSDYNA. This will generate *INITIAL_CONTACT_WEAR cards for subsequent wear simulations, and LS-DYNA will apply this wear and modify geometry accordingly. Restrictions as described in the manual apply.
- Improve SOFT = 6 under *CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURFACE to allow users to define part ID and a node set is automatically generated.

• ***CONSTRAINED**

- Add frictional energy calculation for constraint-based rigid walls.
- *CONSTRAINED_BEAM_IN_SOLID:

  - Works with r-adaptivity now.
  - Can now constrain beams in tshells as well as solids.

- Fix a bug for *CONSTRAINED_LOCAL that might mistakenly constrain z-translation when RC = 0.

- The following options do not support MEMORY = auto properly. The MEMORY = auto option will be turned off in this section and report an error if additional memory allocation is needed.
  *CONSTRAINED_LINEAR_OPTION
  *CONSTRAINED_MULTIPLE_GLOBAL
- Switched translational joints with stiffness to use double precision storage for the displacement value so that the calculated forces are more accurate. This prevents round-off error that can become significant.
- Fixed *CONSTRAINED TIED_NODES_FAILURE when used with MPP single surface segment based contact. Non-physical contact between segments that share tied constraints was being penalized leading to failure of the constraints.
- The SPR models (*CONSTRAINED_SPR2, *CONSTRAINED_INTER-POLATION_SPOTWELD) now support the SPOTDEL option of *CONTROL_-CONTACT. That means if shell elements involved in the SPR domain fail, the SPR gets deactivated.

• ***CONTROL**

- Fix possible error termination with single precision MPP when PSFAIL.ne.0 in *CONTROL_SOLID and using solid formulation 10/13/44.
- Fix spurious deletion of elements when using TSMIN.ne.0.0 in *CONTROL_TERMINATION, erode = 1 in *CONTROL_TIMESTEP and initialized implicitly in dynamic relaxation.
- Added keyword *CONTROL_ACOUSTIC to calculate the nodal motions of *MAT_ACOUSTIC nodes for use in d3plot and time history files. Without this option the *MAT_ACOUSTIC mesh propagates pressure but does not deform because it uses a linear Eulerian solution method. The structural response is unaffected by this calculation; it is only for visualization and will roughly double the time spent computing acoustic element response.
- When IACC = 1 on *CONTROL_ACCURACY and for shell type 16/-16 in nonlinear implicit, shell thickness change due to membrane strain when ISTUPD > 0 in *CONTROL_SHELL is now included in the solution process and will render continuity in forces between implicit time steps. The output contact forces will reflect the equilibrated state rather than the state prior or after the thickness update.
- Fix bug when RBSMS in *CONTROL_RIGID, affecting mass scaled solutions, is used in conjunction with *ELEMENT_INERTIA and/or *PART_INERTIA, specifically with choices of IFLAG on *CONSTRAINED_RIGID_-BODIES and *CONSTRAINED_EXTRA_-NODES.
- Tshells added to the subcycling scheme (*CONTROL_SUBCYCLE).
- Tshells and spotweld beams are supported in selective mass scaling. See IMSCL in *CONTROL_TIMESTEP.
- Add a new keyword: *CONTROL_FORMING_SHELL_TO_TSHELL to convert shell elements to tshell elements.

- If a parent node has SPCs, the same SPC constraints will be applied to the corresponding tshell nodes.
- If adaptivity is invoked, *BOUNDARY_SPC_SET is automatically updated to include newly generated nodes.
- Allows the normal of the segment set to be changed.
- Can offset the generated tshells from the mid-surface of the parent shells.
- Automatically generate segment sets for the top and bottom surfaces, which can be used for contact.

- **DISCRETE ELEMENT METHOD**

  - Implement generalized law of erosion for *DEFINE_DE_TO_SURFACE_-COUPLING based on the following article in the journal "Wear": Magnee, A., Generalized law of erosion: application to various alloys and intermetallics, Wear, Vol. 181, 500, 1995.
  - Modify tangential force calculation to get better rigid body rotation behavior for *DEFINE_DE_BOND
  - Support restart feature for DEM interface force file and DATABASE output.
  - Instead of using bulk modulus, use mass and time step to estimate contact stiffness for SPH-DEM coupling. This should be better if DEM material is quite different from SPH material.
  - Fix *DEFINE_DE_MASSFLOW_PLANE bug if DE injection is defined.
  - Add CID_RCF to *DEFINE_DE_TO_SURFACE_COUPLING for force output in local coordinates to 'demrcf' file.
  - Update the *DEFINE_DE_BY_PART card so that it matches the capabilities of the *CONTROL_DISCRETE_ELEMENT card.
  - Add penalty stiffness scale factor, thickness scale factor, birth time and death time to *DEFINE_DE_TO_SURFACE_COUPLING.
  - Add dynamic coefficient of friction to *CONTROL_DISCRETE_ELEMENT.
  - Implement Finnie's wear law and user defined wear model to *DEFINE_-DE_TO_SURFACE_COUPLING.
  - Implement user-defined curve for DEM frictional coefficient as function of time.
  - Implement user-defined curve for contact force calculation for *CONTROL_DISCRETE_ELEMENT.
  - Fix inconsistent results between *DEFINE_DE_BY_PART and *CONTROL_DISCRETE_ELEMENT.

- ***ELEMENT**

  - Fixed bug affecting output from beam elements ELFORM = 2 when certain uncommon inputs are present. Forces and moments in the output files could be wrongly rotated about the beam axis. This affected the output files only, not the solution inside LS-DYNA. The error could occur under two circumstances: (a) if IST on *SECTION_BEAM is non-zero, the output forces and moments are supposed to be rotated into the beam's principal axis system, but this rotation could be applied to the wrong beam elements; and (b) when no ELFORM = 2 elements have IST, but the model also contains beams with ELFORM = 6 and RRCON = 1 on the SECTION_BEAM card, some of the ELFORM = 2 elements can have their output forces and moments rotated by 1 radian.

- Fix a bug affecting 2d seatbelt with time-dependent slipring friction.
- Fix erroneous 1d seatbelt slipring message.
- Fix seatbelt consistency issue in SMP (ncpu < 0).
- Add error message when 2d seatbelt part doesn't have shell formulation of 5 and *MAT_SEATBELT.
- Fix a bug for 2d seatbelt that could occur when a model has both 1d and 2d belts, and a 1d pretensioner of type 2, 3 or 9.
- Fix an MPP seatbelt bug that could occur when using a type 9 pretensioner.
- Allows shell formulation 9 to be used for 2d seatbelt. It was reset to formulation 5 by LS-DYNA, no matter what formulation was input. Now, only formulation 5 and 9 are accepted as input. Other formulations will incur error message.
- MPP now supports *ELEMENT_MASS_MATRIX_NODE_(SET).
- Added cohesive shell formulation -29. This formulation uses a cohesive midlayer where local direction q1 coincides with the average of the surrounding shell normals. This formulation is better suited for simulating normal shear.
- Cohesive shell formulation +/-29: Fixed absence of part mass in d3hsp.
- Make *TERMINATION_DELETED_SOLIDS work with hex spot weld failures.
- Fix incorrect load curve used if large value is used for FC < 0 and/or FCS < 0 in *ELEMENT_SEATBELT_SLIPRING.
- Fix incorrect velocity on accelerometer if
  - velocity is prescribed on the rigid body that the accelerometer is attached to, and
  - INTOPT = 1 in *ELEMENT_SEATBELT_ACCELEROMETER, and
  - *INITIAL_VELOCITY_GENERATION_START_TIME is used.
- Fix incorrect discrete spring behavior when used with adaptivity.
- Fix input error when using *DEFINE_ELEMENT_DEATH with BOXID > 0 for MPP.
- Modify tolerances on error messages SOL+865 and SOL+866 to prevent unnecessary error terminations when translational or rotational mass of a discrete beam was close to zero.
- Made the solid element negative volume warning SOL+630 for penta formulatgion 15 consistent with the volume calculation in the element. With this change, elements are deleted rather than the job terminating with error SOL+509.

- Fixed the default hourglass control for shell form 16. It was defaulting to type 5 hourglass control rather than 8.
- Fixed default hourglass control when the *HOURGLASS control card is used but no HG type is specified. We were setting to type 1 instead of 2. Also, fixed the default HG types to match the user's manual for implicit and explicit.
- Fixed the fully integrated membrane element (shell ELFORM = 9) when used with NFAIL4 = 1 on *CONTROL_SHELL and there are triangular elements in the mesh. Triangular elements were being deleted by the distorted element check.
- Fixed a divide by zero error that occurred with *SECTION_BEAM, ELFORM = 6, SCOOR = 12 or -12, and node 3 was omitted on *ELEMENT_-BEAM, and nodes 1 and 2 are along the global y-direction or z-direction.
- Fixed laminated shell theory for type 6 and 7 shell elements when made active by LAMSHT = 3 or 5 on *CONTROL_SHELL.
- Added an int.pt. variable for *PART_COMPOSITE_LONG and *PART_-COMPOSITE_TSHELL_LONG called SHRFAC which is a scale factor for the out-of-plane shear stress that allows the user to choose the stress distribution through thickness. This was motivated by test data that shows that for large differences is layer shear stiffness, the parabolic assumption is poor.
- Fixed implicit hourglass stiffness in viscoelastic materials when used with tshell forms 5 or 6. The stiffness was much too small.
- Modified tshell type 5 to use the tangent stiffness for calculating the Poisson's affects and hourglass control for *MAT_024. This makes the behavior softer during buckling which is much more realistic.
- Fixed a significant bug in segment based contact when SHLEDG = 1 and SBOPT = 3 or 5 and DEPTH < 45, and shell segments in contact have different thicknesses. A penetration check was using incorrect thicknesses causing contact to be detected too late, particularly for edge to surface contact.
- Improved the time step calculation for triangular tshell elements. The time step was too conservative for elements with significant thickness. This fix does not affect tshell type 7.
- Fixed all tshells to work with anisotropic thermal strains which can be defined by *MAT_ADD_THERMAL. Also, this now works by layer for layered composites.
- Enabled tshell form 5 to recalculate shear stiffness scale factors when plasticity material models 3, 18, 24, 123, or 165 are included in a composite section. Prior to this change the scale factors were based on elastic properties so after yielding, the stress distribution was not what was expected. This new capability supports the constant stress option, the parabolic option, and the SHRFAC option on *PART_COMPOSITE_TSHELL_LONG.
- Improved tshell 5 when used with mixed materials in the layers. A failure to use the correct Poisson's ratio was causing a less accurate stress tensor.

- Modified the time step calculation for tshell forms 3 and 5. A dependence on volumetric strain rate was removed in order to prevent oscillations in the time step which caused stability problems, particularly for tshell 5.
- Fixed tshell constant shear stress option (TSHEAR = 1 on *SECTION_TSELL or *PART_COMPOSITE). It was producing a not very constant stress distribution.
- Fixed stress and strain output of tshells when the composite material flag CMPFLG is set on *DATABASE_EXTENT_BINARY. The transformation was backwards.
- Fixed mass of parts reported to d3hsp when *ELEMENT_SHELL_-SOURCE_SINK is used. The mass of inactive elements was being included.
- Enabled *MAT_026 and *MAT_126 (HONEYCOMB) to be used with tshell forms 3, 5, and 7. It was initialized incorrectly causing a zero stress.
- Added a missing internal energy calculation for tshell form 6.
- Enabled tshell forms 1, 2, and 6 to work with material types 54, 55, and 56.
- Modified the z-strain distribution in tshell forms 5 and 6 when used in composites with mixed materials that are isotropic. The existing assumed strain scheme was doing a poor job of creating a constant z-stress through the thickness.
- Increased the explicit solution time step for thin shell composite elements. The existing method calculated a sound speed using the stiffness from the stiffest layer and dividing it by the average density of all layers. This could be overly conservative for composites with soft layers of low density. The new method uses the average stiffness divided by average density. This is still conservative, but less so.
- Corrected rotational inertia of thin shells when layers have mixed density and the outer layers are denser than inner layers. The fix will mostly affect elements that are very thick relative to edge length.
- Added support for *ELEMENT_SHELL_SOURCE_SINK to type 2 shells with BWC = 1 on *CONTROL_SHELL.
- Prevent inactive shell elements (from *ELEMENT_SHELL_SOURCE_-SINK) from controlling the solution time step.
- Fixed *LOAD_STEADY_STATE_ROLLING when used with shell form 2 and Belytschko- Wong-Chang warping stiffness (BWC = 1 *CONTROL_-SHELL). The load was not being applied.
- Improved the brick element volume calculation that is used by the option erode elements (ERODE = 1 on *CONTROL_TIMESTEP or PSFAIL.ne.0 on *CONTROL_SOLID). It was not consistent with the element calculation which caused an error termination.
- Fixed all tshell forms to work with anisotropic thermal strains which can be defined by *MAT_ADD_THERMAL. Also, this now works by layer for layered composites.
- Reworked shell output so that we can correctly output stress in triangular shells when triangle sorting is active, that is when ESORT = 1 or 2 on *CONTROL_SHELL.

- *ELEMENT_T/SHELL_COMPOSITE(_LONG) and *PART_COMPOSITE_-T/SHELL_(LONG): Permit the definition of zero thickness layers in the stacking sequence. This allows the number of integration points to remain constant even as the number of physical plies varies and eases post-processing since a particular integration point corresponds to a physical ply. Such a capability is important when plies are not continuous across a composite structure.
  To represent a missing ply, set THK to 0.0 for the corresponding integration point and additionally, either set MID = -1 or set PLYID to any nonzero value. Obviously, the PLYID option applies only to the keywords containing LONG.
- Implemented sum factorization for 27-node quadratic solid that may increase speed by a factor of 2 or 3.
- Support second order solid elements (formulations 23,24,25,26) for *SET_-NODE_GENERAL.
- Invoke consistent mass matrix of 27-node hex element for implicit dynamics and eigenvalues.
- Reorder node numbering when assembling global stiffness matrix for 27-node hex. This fixes a bug in which it was reported than the implicit 27-node element didn't work
- Automatically transfer nodal boundary conditions for newly generated nodes if H8TOH27 option is used in *ELEMENT_SOLID.
- Modify initialization of material directions for solid elements. If there are only zeros for all the 6 values in *INITIAL_STRESS_SOLID, then the values from the other input (e.g. *ELEMENT_SOLID_ORTHO) are kept.
- Enable *PART_STACKED_ELEMENTS to pile up shell element layers. Before, it was necessary that solid element layers were placed between shell element layers. Now, shell element layers can follow each other directly. Contact definitions have to be done separately.
- Allow *PART_STACKED_ELEMENTS to be used in adaptive refinement simulations.
- Add alternative mass calculation for critical time step estimate of cohesive elements. This hopefully resolves rarely occurring instability issues. Option ICOH on *CONTROL_SOLID is used for that.
- Correct the strain calculation for tet formulation 13. This did not affect the stress response, only output of strains. Nodal averaging was not accounted for.
- User defined elements (ELFORM = 101 to 105 on *SECTION_-SHELL/SOLID) can now be used together with *MAT_ADD_EROSION.
- Add option to define a pull-out force vs. time curve in *ELEMENT_BEAM_-SOURCE by defining a negative variable FPULL. |FPULL| can refer to *DEFINE_CURVE or *DEFINE_CURVE_FUNCTION.
- Solid tet form 13 supported for all materials in implicit, including a presumable consistency improvement for the future.

- The Hughes-Liu beam is supported in *INTEGRATION_BEAM such each integration point may refer to a different part ID and thus have a different coef. Of thermal expansion. See *MAT_ADD_THERMAL_EXPANSION.
- Shell types 2 and 16 that combines thermal expansion and thick thermal shells, see *MAT_ADD_THERMAL_EXPANSION and TSHELL on *CONTROL_SHELL, now correctly treat temperature gradient through the thickness to create bending moments. All shell types are to be supported in due time.
- *SECTION_BEAM_AISC now provides predefined length conversion factors for specific unit systems.
- 3D tet r-adaptivity now supports *DEFINE_BOX_ADAPTIVE.
    - For every adaptive part, users can define multiple boxes where different BRMIN & BRMAX (corresponding to RMIN & RMAX in *CONTROL_REMESHING) can be specified for 3D tet remesher to adjust the mesh size.
    - Current implementation does not support LOCAL option.
- Fix bug in 3D adaptivity so that users can now have both non-adaptive tshell parts and 3D adaptive parts in one analysis.
- Fix the bug in 3D adaptivity so that users can now have both dummy nodes and 3D adaptive parts in one analysis.

- ***EM (Electromagnetic Solver)**
    - Randles Circuits for Battery Modeling
        - A Randles circuit is an equivalent electrical circuit that consists of an active electrolyte resistance r0 in series with the parallel combination of the capacitance c10 and an impedance r10. The idea of the distributed Randles model is to use a certain number of Randles circuits between corresponding nodes on the two current collectors of a battery unit cell. These Randles circuits model the electrochemistry that happens in the electrodes and separator between the current collectors. The EM solver can then solve for the EM fields in the current collectors, and the connections between them.
        - Added analysis of distributed Randles circuits to MPP.
        - Added d3plot output for distributed Randles circuits:
          D3PL_RAND_r0_EM,
          D3PL_RAND_r10_EM,
          D3PL_RAND_c10_EM,
          D3PL_RAND_soc_EM,
          D3PL_RAND_i_EM,
          D3PL_RAND_u_EM,
          D3PL_RAND_v_EM,

D3PL_RAND_vc_EM,
D3PL_RAND_temperature_EM,
D3PL_RAND_P_JHR_EM,
D3PL_RAND_P_dudt_EM,
D3PL_RAND_i_vector_EM

This output can be visualized in LS-PrePost versions 4.3 and 4.5 on the separator part of the battery cell using Post/FriComp/Extend/EM node.

- Added tshells for EM analysis for use in battery modeling.
- Added new capability for modeling Randles short, based on *DEFINE_FUNCTION so that the user has a lot of freedom to define where and when the short happens as well as the short resistance.
- Added a new capability for battery exothermal reactions also based on *DEFINE_FUNCTION. The new keyword *RANDLE_EXOTHERMAL_REACTION makes it possible to complement the heating of a short circuit created by a short by exothermal reactions if, for example, the temperature becomes higher than a threshold value.

- **FORMING ANALYSIS**

  - Extend *INCLUDE_AUTO_OFFSET to solid and beam elements (draw beads).
  - Add a new keyword for springback compensation: *INTER-FACE_COMPENSATION_NEW_REFINE_RIGID to refine and break rigid tool mesh along the user supplied trim curves so compensated tool mesh follows exactly the blank mesh (file "disp.tmp"). This needs to be done only once in the beginning of the springback compensation (ITER0).
  - *CONTROL_FORMING_ONESTEP:

    - Change the default element formulation option for onestep method to QUAD2.
    - Add a new option QUAD to allow quadrilateral elements to be considered.
    - Limit the maximum thickening by using a new variable TSCLMAX for the sheet blank.
    - Set the value of OPTION to a negative value to output the file 'onestepresult' in large format (E20.0).
    - Calculate and add the damage factor and output to the 6th history variable in the output file "onestepresult". Add the variable for a curve ID to define the fracture strain vs. triaxility. Add another variable DMGEXP (damage parameter), as used in GISSMO model.
    - Keep the original coordinates for the onestep output "onestepresults".

  - Add a new option VECTOR to *CONTROL_FORMING_BESTFIT to output deviation vector (in the format of: NODEID, xdelta, ydelta, zdelta) for each node to its closest target element. The deviation vectors are output under the keyword *NODE_TO_TARGET_VECTOR.

- *CONTROL_FORMING_OUTPUT:
  - Output will skip any negative abscissa (Y1) value.
  - When CIDT < 0, the positive value defines the time dependent load curve.
- Add a warning in springback compensation *INTERFACE_SPRING-BACK_-COMPENSATION to identify which input file (typically the blank with adaptive mesh not output directly by LS-DYNA) has the wrong adaptive constraints.
- *INTERFACE_COMPENSATION_3D: turn off the output of nikin file.
- *ELEMENT_LANCING:
  - Allow some unused lancing curves to be included in the input.
  - When the gap between the two ends of a lancing curve is not zero, but small enough, then this curve is automatically closed.
  - Allow several parts to be cut during lancing; the parts can be grouped in *SET_PART_LIST, and defined using a negative value IDPT.
  - Specify the distance to bottom dead center as AT and ENDT when the new variable CIVD is defined.
  - Set IREFINE = 1 (default) in lancing, to refine blank mesh automatically along the lancing curves.
  - Re-set the adaptive level to be 1 to prevent those elements along the lancing route to be further refined.
  - When IREFINE = 1, elements along the lancing curve will be refined to make sure that no adapted nodes exist in the neighborhood. This helps get improved lancing boundary.
  - Change of tolerance for lancing to merge the small elements into bigger ones.
- Add a new keyword to perform trimming after lancing (shell elements only): *DEFINE_LANCE_SEED_POINT_COORDINATES. Maximum of two seed nodes can be defined.
- Extend *CONTROL_FORMING_TOLERANC to *MAT_036, *MAT_037, *MAT_125, and *MAT_226. When beta is less than -0.5, there is no necking and no calculation of FI. When beta is greater than 1.0, beta = 1.0/beta. This keyword adds a smoothing method to calculate the strain ratios for a better formability index.
- Sandwiched parts (*CONTROL_ADAPTIVE, *DEFINE_CURVE_TRIM):
  - Disable *CONTROL_ADAPTIVE_CURVEs for sandwich parts, since refinement along the curve is automatically done during trimming.

        - Refine the elements along the trimming curve to make sure no slave nodes are be cut by trimming curves.
        - Allow mesh adaptivity.
        - Allow multi-layers of solids.
        - Add a check to the variable IFSAND in *CONTROL_ADAPTIVE for sandwich part to be refined to exclude solid elements.

    - Solid element trimming (*DEFINE_CURVE_TRIM):

        - Refine those elements along the trimming curve.
        - Improve solid trimmig to allow the trimming of one panel into two panels with two seed nodes.

    - Add a new keyword *CONTROL_FORMING_REMOVE_ADAPTIVE_-CONSTRAINTS to remove adaptive constraints on a formed, adapted blank, and replaced them with triangular elements.
    - *DEFINE_CURVE_TRIM_NEW: Allow trimming of tshells.
    - Add a new keyword: *INTERFACE_WELDLINE_DEVELOPMENT to obtain initial weld line from the final part and the final weld line position.

        - When Ioption = -1, convert weld line from its initial position to the final part.
        - Output the element nodes that intersect the weld line in the final part, and the output file is: affectednd_f.ibo
        - Output the element nodes that intersect the weld line in the initial part, and the output file is: affectednd_i.ibo

    - Add a new variable DT0 to *CONTROL_IMPLICIT_FORMING so there is no need to use *CONTROL_IMPLICIT_GENERAL to specify DT0.
    - *INTERFACE_BLANKSIZE:

        - Add a new feature DEVELOPMENT option. When ORIENT = 2, then a reference mesh file for the formed part should be included. The calculated and compensated boundary will be based on the reference mesh.
        - Add a new option SCALE_FACTOR that allows the target curve to be moved. This is useful when multiple target curves (e.g. holes) and formed curves are far away from each other.

- ***FREQUENCY_DOMAIN**

    - Added new keyword *CONTROL_FREQUENCY_DOMAIN to define global control parameters for frequency domain analysis. Currently two parameters are defined:

- REFGEO: flag for reference geometry in acoustic eigenvalue analysis (either the original geometry at t = 0, or the deformed geometry at the end of transient analysis).
- MPN: large mass added per node, to be used in large mass method for enforced motion.

◦ *FREQUENCY_DOMAIN_ACOUSTIC_BEM:

- Enabled using incident wave (*FREQUENCY_DOMAIN_ACOUSTIC_INCIDENT_WAVE) in Rayleigh method (METHOD = 0).
- Enabled acoustic pressure fringe plot (*FREQUENCY_DOMAIN_ACOUSTIC_FRINGE_PLOT) in Rayleigh method (METHOD = 0).
- Fixed bug in running acoustic analysis with multiple boundary conditions in MPP.
- Fixed running MATV (Modal Acoustic Transfer Vector) approach in MPP (*FREQUENCY_DOMAIN_ACOUSTIC_BEM_MATV).
- Added treatment for triangular elements used in Rayleigh method (METHOD = 0).
- Added output of acoustic intensity to binary database D3ACS (defined by *DATABASE_FREQUENCY_BINARY_D3ACS).
- Fixed bug in acoustic pressure fringe plot for collocation BEM (METHOD = 3) and dual BEM based on Burton-Miller formulation (METHOD = 4).

◦ *FREQUENCY_DOMAIN_ACOUSTIC_FEM:

- Fixed bug in acoustic analysis by FEM, when dimensions of mass and k (stiffness) matrices are mismatched.

◦ *FREQUENCY_DOMAIN_ACOUSTIC_FRINGE_PLOT:

- Implemented acoustic fringe plot for MPP for the options PART, PART_SET, and NODE_SET.

◦ *FREQUENCY_DOMAIN_FRF:

- Added new loading types:
  VAD1 = 5: enforced velocity by large mass method
  = 6: enforced acceleration by large mass method
  = 7: enforced displacement by large mass method
  = 8: torque
  = 9: base angular velocity
  = 10: base angular acceleration
  = 11: base angular displacement
- Added rotational dof output for FRF.

- *FREQUENCY_DOMAIN_MODE:
  - Added option _EXCLUDE to exclude some eigenmodes in modal superposition in frequency domain analysis.
- *FREQUENCY_DOMAIN_RANDOM_VIBRATION:
  - Fixed bug in running random vibration with random pressure wave load (VAFLAG = 2) in MPP.
  - Improved random vibration analysis by allowing using complex variable cross PSD functions. Previously cross PSD was defined as real variables thus the phase difference was ignored.
  - Added PSD and RMS computation for Von Mises stress in beam elements.
- *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM:
  - Added Von Mises stress output for beam elements in database D3SPCM.
  - Corrected computation of response spectrum at an intermediate damping value by interpolating spectra at two adjacent damping values. Now the algorithm is based on ASCE 4-98 standard.
- *FREQUENCY_DOMAIN_SSD:
  - Added new loading types:
VAD = 5: enforced velocity by large mass method
= 6: enforced acceleration by large mass method
= 7: enforced displacement by large mass method
= 8: torque
  - Fix for running SSD fatigue in MPP (affected keyword: *FREQUENCY_DOMAIN_SSD_FATIGUE).
  - Updated ssd computation with local damping, and enabled the restart feature by reading damping matrix.
  - Implemented ERP (Equivalent Radiated Power, keyword *FREQUENCY_DOMAIN_SSD_ERP) for MPP.
- *DATABASE_FREQUENCY_ASCII:
  - Added keyword *DATABASE_FREQUENCY_ASCII_{OPTION} to define the frequency range for writing frequency domain ASCII databases NODOUT_SSD, ELOUT_SSD, NODOUT_PSD and ELOUT_PSD.

- ***ICFD (Incompressible Fluid Solver)**
  - New ICFD features and major modifications

- Simple restart is now supported for ICFD.
- Added wave damping capabilities. See *ICFD_DEFINE_WAVE_-DAMPING.
- Added steady state solver. See *ICFD_CONTROL_GENERAL and *ICFD_CONTROL_STEADY.
- Added steady state potential flow solver. See *ICFD_CONTROL_-GENERAL.
- Weak thermal coupling for conjugate heat transfer is now possible in addition to the classic monolithic approach. See *ICFD_CONTROL_-CONJ.
- Windkessel boundary conditions are now available for blood flow. See *ICFD_BOUNDARY_WINDKESSEL.
- It is now possible to output the heat transfer coefficient as a surface variable in LSPP or in ASCII format on segment sets for a subsequent solid-thermal only analysis. See *ICFD_DATABASE_HTC.
- Two way coupling is now possible with DEM particles. See *ICFD_-CONTROL_DEM_COUPLING.
- Modifications introduced in the SUPG stabilization term used in thermal and conjugate heat transfer problems for improved accuracy and speed.

◦ Additions and modifications to existing ICFD keywords

- *ICFD_BOUNDARY_FSWAVE:
  Added a boundary condition for wave generation of 2nd order stokes waves with free surfaces
- *ICFD_CONTROL_FSI:
  Added a flag which, when turned on will project the nodes of the CFD domain that are at the FSI interface onto the structural mesh. This is recommended for cases with rotation.
- *ICFD_CONTROL_MESH:
  Added a flag to allow the user control over whether there will be re-mesh or not. If there is no re-mesh then we can free space used to backup the mesh and lower memory consumption.
- *ICFD_CONTROL_MESH_MOVE:
  Added option to force the solver to turn off any mesh displacements. This can be useful in cases where the mesh is static to save a little bit of calculation time.
- *ICFD_CONTROL_OUTPUT:
  Added option to support output in Fieldview format, binary and ASCII.
  When output of the fluid volume mesh is requested, the mesh will be divided into ten distinct parts, grouping elements in ten deciles based on the mesh quality (Part 1 has the best quality elements, part 10 the worst).

- *ICFD_CONTROL_POROUS:
  Improvements for RTM problems.
- *ICFD_CONTROL_TIME:
  Added an option to define an initial timestep.
  Added an option to shut off the calculation of Navier Stokes after a certain time leaving only the heat equation. This can be useful to save calculation times in conjugate heat transfer cases where the fluid often reaches steady state before the thermal problem.
- *ICFD_DATABASE_DRAG:
  It is now possible to output the force on segment sets in a FSI run directly in LS-DYNA compatible format. This can be useful for a subsequent linear FSI analysis running only the solid mechanics part.
  Added flag to output drag as a surface variable in LSPP.
- *ICFD_DATABASE_FLUX:
  Added option to change output frequency
- *ICFD_DATABASE_NODOUT:
  The user node IDs are now required rather than the internal node IDs
- *ICFD_CONTROL_IMPOSED_MOVE:
  Added the option to choose between imposing the displacements or the velocity.
- *ICFD_CONTROL_TRANSIENT:
  Choose implicit time integration scheme for NS.
- *ICFD_CONTROL_DEM_COUPLING:
  Added a scale factor for the sphere radius in the computation of the DEM force.
- *ICFD_MODEL_POROUS:
  Added a scale factor option on the permeability for model 1 and 2. A *DEFINE_FUNCTION can also be used.
- *MESH_BL:
  Added option to generate boundary layer mesh using a growth factor.

◦ ICFD bug fixes and minor improvements

- Fixed bug when multiple *DEFINE_FUNCTIONs were used in an ICFD problem. Only the last one was taken into account.
- LES turbulence model: fixed van Driest damping issue in the boundary layer. LES models can use wall functions.
- RANS turbulence models: Standard k-epsilon, realizable k-epsilon, Wilcox k-omega uses HRN laws of the wall by default while SST and Spalart Allmaras use LRN. Improvements on the convergence of all those models.
- The DEM particle volume is now taken into account in free surface problems.
- Average shear is now output as a surface variable in the d3plots.

- *ICFD_CONTROL_MONOLITHIC is obsolete (replaced by *ICFD_-CONTROL_GENERAL).
- Added more output for the mesh generation indicating the stage of the meshing process and the amount of elements that are being generated as a multiple of 10000. Added progress % for the extrusion of the mesh during the BL mesh generation.
- Improvements on the element assemble speed in MPP.
- Fixed synchronization problem for the last timestep in an FSI problem.
- More options have been added to the timer output.
- Correction of the calculation of the flux in *ICFD_DATABASE_FLUX in free surface cases.
- Boundary layer mesh can go through free-surfaces or mesh size interfaces.
- The Center of Gravity of the fluid is output in the icfd_lsvol.dat ASCII file in free surface problems

- **Implicit (Mechanical) Solver**

  - Enhanced termination of MPP eigensolver when non eigenmodes are found.
  - Implicit was enforcing birth and death times on *BOUNDARY_SPC during dynamic relaxation contrary to the User's Manual. These times are now ignored by implicit during dynamic relaxation.
  - Corrected output of eigenvalues and frequencies to file eigout for the asymmetric eigenvalue problem.
  - Enhanced logic that determines when to write out the last state to d3plot for implicit.
  - Improved error message for reading d3eigv file for *PART_MODES for the case when the user inputs a d3eigv file from a different model than intended.
  - Corrected the reporting of kinetic and internal energy in file glstat for implicit.
  - Applied corrections to tied contact in implicit (MPP). This affects slave nodes coming from other processes.
  - Corrected output to file d3iter (implicit nonlinear search vectors) for restart.
  - Enhanced termination process when the implicit solver determined an early termination.
  - When implicit springback was following an explicit transient step, the implicit keywords with the _SPR were not properly handled. This is now corrected.
  - Added a warning about the combined use of *CONSTRAINED_RIGID_-BODY_STOPPERS and the Lagrange multiplier formulation for joints (*CONTROL_RIGID) for explicit. The warning recommends switching to the penalty joint formulation.
  - Applied numerous bug fixes to the implicit solver associated with *CONSTRAINED_INTERPOLATION where there are lots of independent degrees-of-freedom.

- Corrected initialization of MPP tied contact with implicit mechanics when the implicit phase follows explicit dynamic relaxation.
- Fixed an implicit problem where a linear implicit analysis follows inertia relief computation.
- Added gathering of damping terms from discrete elements from implicit especially for FRF computations and matrix dumping.
- Fixed Implicit for the case of multi-step Linear (NSOLVR = 1) with Intermittent Eigenvalue Computation.
- Corrected the output to d3iter when 10-noded tets are present.
- Keypoints specified in *CONTROL_IMPLICIT_AUTO are now enforced at the initial time step and on restart from explicit.
- Skip frequency damping during implicit static dynamic relaxation, i.e. IDRFLAG > 5.
- *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS:
  - The VID of the rotating axis can now be defined by both *DEFINE_VECTOR and *DEFINE_VECTOR_NODES. It enables the movement of the rotating axis. Previously, only *DEFINE_VECTOR could be used to define the VID.
  - The rotational dynamics now work in MPP.
- Shell forms 23 and 24 (high order shells), 1D seatbelts, Hughes-Liu and spotweld beams (types 1 and 9) are now supported with the implicit accuracy option (IACC = 1 in *CONTROL_ACCURACY) to render strong objectivity for large rigid body rotations. Also, shell type 16 is supported with implicit accuracy option, resulting in forms 16 and -16 giving the same solution.
- Translational and generalized stiffness joints are now strongly objective for implicit analysis. See CONSTRAINED_JOINT_STIFFNESS....
- In implicit it may happen that the initial loads are zero, for instance in forming problems. In addition, the goal is to move a tool in contact with a workpiece, and the way line search and convergence works, it is hard to get things going. We now attempt to handle this situation by automatically associating an augmented load to the prescribed motion simply to get off the ground.
- New tolerances on maximum norms are introduced for convergence in implicit: ratio of max displacement/energy/residual, and absolute values of nodal and rigid body translation/rotational residual can be specified. See DNORM.LT.0 on *CONTROL_IMPLICIT_SOLUTION for defining an additional card for these parameters DMTOL, EMTOL and RMTOL. Furthermore, maximum absolute tolerances on individual nodal or rigid body parameters can be set on NTTOL, NRTOL, RTTOL and RRTOL on the same card.
- If ALPHA < 0 on first *CONTROL_IMPLICIT_DYNAMICS card, the HHT implicit time integration scheme is activated.

- ***INITIAL**

- Fix *INITIAL_VELOCITY_GENERATION when used with *INCLUDE_-TRANSFORM, which was broken due to misplaced conditionals in r100504.
- Fix 3 bugs for *INITIAL_VELOCITY_GENERATION involving omega > 0 and icid > 0:
  - When nx = -999. Now the directional cosine defined by node NY to node NZ will be the final direction to rotate about. In other words, the direction from node NY to node NZ will not be projected along icid any more.
  - When nx != -999, (xc,yc,zc) should not be rotated along icid, since (xc,yc,zc) are global coordinates.
  - When *INITIAL_VELOCITY_GENERATION is included by *INCLUDE_TRANSFORM, (xc,yc,zc) is transformed.
- Add the option of ramping time steps, ndtrrg, for *INITIAL_FOAM_REFERENCE_GEOMETRY. The solid elements with reference geometry and ndtrrg > 0 will restore its reference geometry in ndtrrg time steps.
- Fix incorrect initial velocity when ICID.ne.0 in *INITIAL_VELOCITY_GENERATION, and rotational velocity, omega, is not zero and *PART_INERTIA is also present.
- Add variable IZSHEAR in *INITIAL_STRESS_SECTION to initialize shear stress.
- Fix incorrect initial velocity for *INITIAL_VELOCITY if IRIGID = -2 and ICID > 0.
- Fix incorrect NPLANE and NTHICK for *INITIAL_STRESS_SHELL when writing dynain for shell type 9.
- Fix *INITIAL_STRAIN_SHELL output to dynain for shell types 12 to 15 in 2D analysis. Write out strain at only 1 intg point if INTSTRN = 0 in *INTERFACE_SPRINGBACK_LSDYNA and all strains at all 4 intg points if INTSTRN = 1 and nip = 4 in *SECTION_SHELL.
- Skip transformation of the initial velocities if ICID > 0 and *INCLUDE_-TRANSFORM is used to transform the keyword input file with the *INITIAL_VELOCITY.... keyword. Also echo warning message, KEY+1109, that the transformation will be skipped since icid is specified.
- Fix incorrect transformation of *DEFINE_BOX which results in incorrect initial velocities if the box is used in *INITIAL_VELOCITY.
- Fixed *INITIAL_STRESS_DEPTH when used with 2D plane strain and axisymmetric elements. The prestress was being zeroed.
- Improved the precision of the initial deformation calculation for *INITIAL_-FOAM_REFERENCE_GEOMETRY in the single precision version.
- Fixed stress initialization (*INITIAL_STRESS_SECTION) for type 13 tet elements. The pressure smoothing was causing incorrect pressure values in the elements adjacent to the prescribed elements.
- Add _SET option to *INITIAL_STRESS_SOLID for element sets.

- Fix bug in 3D adaptivity so that users can now define *INITIAL_TEMPERATURE for adaptive parts.

- **Isogeometric Elements**

  - The stability of the trimmed NURBS shell patches has been improved.
  - Add *LOAD_NURBS_SHELL to apply traction type loading directly on the surface of NURBS shell.
  - Users can use the PART option of *SET_SEGMENT_GENERAL to define segment set of a NURBS patch. The segment set will contain all segments of interpolated null shell elements.
  - *ELEMENT_SOLID_NURBS_PATCH:

    - Isogeometric solid analysis implemented for MPP.
    - Isogeometric solid analysis implemented for SMP with multiple CPUs, including consistency (ncpu < 0).
    - Activate user-defined materials for isogeometric solid.

  - *ELEMENT_SHELL_NURBS_PATCH:

    - Isogeometric shell analysis now implemented for SMP with multiple CPUs, including consistency (ncpu < 0).
    - Add a power iteration method to get the maximum eigen-frequency for each isogeometric element. This will be used to set a reasonable time step for trimmed elements.

  - *ELEMENT_SHELL_NURBS_PATCH:

    - Changed the way of projecting the results from isogeometric (NURBS) elements to the interpolation elements. Now a background mesh, spanned over the locations of the integration points of the isogeometric (NURBS) elements serves as basis to interpolate results from the integration points to the centroid of the interpolation elements. This change may lead to slightly different post-processing results in the interpolation elements.

  - Add support for trimmed NURBS to work in single precision. Anyway, it is still recommended to use double precision versions for trimmed NURBS patches.
  - Add post-processing of strains and thickness for interpolation shells.

- ***LOAD**

  - Fixed bugs affecting discrete beam elements (ELFORM = 6) when used with staged construction. Here, "dormant" refers to elements that have not yet become active as defined on *DEFINE_STAGED_-CONSTRUCTION_-PART.

  - Dormant discrete beams could still control the timestep and attract mass-scaling, when they should not do so.
  - Dormant discrete beams reaching a failure criterion defined on the *MAT card were deleted, when they should not be.
  - The displacements output (see *DATABASE_DISBOUT) included displacements occurring while the elements were dormant. Now, the output displacements are reset to zero at the moment the element becomes active.

- Fixed bug in Staged Construction: if FACT on *CONTROL_-STAGED_-CONSTRUCTION had been left blank, and Dynamic Relaxation was active, an error termination occurred.
- Fixed bug: *LOAD_GRAVITY_PART (and also gravity loading applied by *DEFINE_STAGED_CONSTRUCTION_PART) was failing to account for non-structural mass when calculating gravity load: NSM on *SECTION_-BEAM and MAREA on *SECTION_SHELL.
- Fixed bug in *LOAD_VOLUME_LOSS: inconsistent results when run in SMP parallel.
- Fix bugs affecting *LOAD_SEGMENT_FILE:

  - Remove LOAD_SEGMENT_FILE file size limit (It used to be 200M).
  - Apply correct pressure on the shared boundary between processors.

- Fix GRAV = 1 in *PART which was not working correctly with *LOAD_-DENSITY_DEPTH. Make *LOAD_DENSITY_DEPTH work for Lagrangian 2D elements.
- Fix insufficient memory error, SOL+659, when using *LOAD_-ERODING_-PART_SET with MPP.
- Fix incorrect loading when using *LOAD_ERODING_PART_SET with BOXID defined.
- Added *LOAD_SUPERPLASTIC_FORMING for implicit analysis.
- *LOAD_SUPERPLASTIC_FORMING box option now works in MPP.

- ***MAT and *EOS**

  - *MAT_197 (*MAT_SEISMIC_ISOLATOR) could become unstable when the parameter DAMP was left at its default value. A workaround was to input DAMP as a small value such as 0.05. The timestep for *MAT_197 is now smaller than previously, irrespective of the DAMP setting, and the behavior is now stable even if DAMP is left at the default.
  - Fixed bug: Timestep calculation was wrong for *MAT_089 solid elements. Response could be unstable especially for higher values of Poisson's ratio, e.g. 0.4.

- Fixed bug: An error trap was wrongly preventing ELFORM = 15 for *MAT_-169 (*MAT_ARUP_ADHESIVE). Wedge elements with ELFORM = 15 are now permitted.
- *MAT_172 (*MAT_CONCRETE_EC2):
  Note that items (1) and (2) below can lead to different results compared to previous versions of LS-DYNA.

  - (1) The number of potential cracks in MAT_172 shell elements has been increased from 2 to 4. MAT_172 uses a fixed crack model: once the first crack forms, it remains at the same fixed angle relative to the element axes. Further cracks can then form only at pre-defined angles to the first crack. Previously, only one further crack could form, at 90 degrees to the first crack. Thus, if the loading direction subsequently changed so that the principal tension is at 45 degrees to the first crack, that stress could exceed the user-defined tensile strength by a considerable margin. Now, further cracks may form at 90, +45 and -45 degrees to the first crack. Although the maximum principal stress can still exceed the user-defined tensile strength, the "error" is much reduced. There is an option to revert to the 2-crack model as in R9 (to do this, add 100 to TYPEC).
  - (2) Add element erosion to MAT_172. This change may lead to different results compared to previous versions, because erosion strain limits are now added by default. Elements are now deleted when crack-opening strain becomes very large, or the material is crushed beyond the spalling limit. Plastic strain in the rebar is considered too. Previously, these elements that have passed the point of being able to generate any stress to resist further deformation would remain in the calculation, and sometimes showed very large non-physical deformations and could even cause error terminations. Such elements would now be deleted automatically. Default values are present for the erosion strains but these can be overridden in the input data, see new input fields ERODET, ERODEC, ERODER.
  - (3) New history variables 10,11,12 (maximum value so far of through-thickness shear stress). This is useful for checking results because MAT_172 cracks only in response to in-plane stress; before cracking occurs, the through-thickness shear capacity is unlimited. The data components are:
    Ex History Variable 10 - maximum out of 11 and 12
    Ex History Variable 11 - maximum absolute value of YZ shear stress
    Ex History Variable 12 - maximum absolute value of ZX shear stress
    These are in the element local axis system. Note that these variables are written only if TYPESC is zero or omitted. TYPESC is a pre-existing capability that requests a different type of shear check.

  - (4) Fixed bug. Elastic stiffness for MAT_172 beams was not as described in the manual, and the axial response could sometimes become unstable. The bug did not affect shell elements, only beams.
  - (5) *MAT_172 can now handle models with temperatures defined in Kelvin (necessary if the model also has heat transfer by radiation). *MAT_172 has thermally-sensitive material properties hard-wired to assume temperatures in Centigrade. A new input TMPOFF in *MAT_-172 offsets the model temperatures before calculating the material properties.
  - (6) When the input parameter AGGSZ is defined, the maximum shear stress that can be transferred across closed cracks is calculated from a formula that has tensile strength and compressive stress as inputs. In MAT_172, the tensile strength of concrete is reduced when compressive damage has occurred (see description of UNLFAC). Up to now, compressive damage was therefore influencing the maximum shear across cracks. However, the Norwegian standard from which the shear formula is taken treats the tensile strength as a constant. Therefore, for the purpose of calculating the maximum shear stress across closed cracks only, the compressive damage effect is now ignored.
  - (7) Added capability for water pressure in cracks, for offshore applications. The water pressure is calculated from the depth of the element below the water surface (calculated from the z-coordinate). The water pressure is applied as a compressive stress perpendicular to the plane of any crack in the element. See new input fields WRO_G and ZSURF.

- *MAT_119 (*MAT__GENERAL_NONLINEAR_6DOF_DISCRETE_BEAM): Fixed bug in UNLOAD option 2. The bug occurs if an unloading curve has been left zero (e.g. LCIDTUR) while the corresponding loading curve was non-zero (e.g. LCIDTR), and UNLOAD = 2. Depending on the computer system, the symptoms could be harmless or the code could crash. Now, if the unloading curve is left blank, it is assumed to be the same as the loading curve i.e. load and unload up and down the same curve. That behavior was already implemented for UNLOAD = 1.

- Added Equation Of State 19 (*EOS_MURNAGHAN). Used extensively for fluid modeling in SPH through Weakly-Compressible formulation, in conjunction with SPH formulations 15 (fluid form) and 16 (normalized fluid form).
- *MAT_ADD_FATIGUE: Added a new form of Basquin equation to define material's SN curve: LCID = -3: S = a*N^b, where a and b are material constants.
- Add the option of A0REF for *MAT_FABRIC. That allows the option of using reference geometry to calculate A0 for the purpose of porosity leakage calculation.

- Add optional parameter DVMIN for *MAT_ADD_PORE_AIR to define the min volume ratio change to trigger pore air flow analysis.
- *DEFINE_HAZ_PROPERTIES:
- Distance of shell from the weld center is treated consistently under MPP and the shell material's yield stress is scaled properly.
- *MAT_168 and *MAT_279: Fixed support for element erosion.
- *MAT_092: Improved of implicit convergence for shells.
- *MAT_224: Fixed bug where wrong shear modulus was used in EOS.
- *MAT_270: Increased stability for thickness strain iterations for shells.
- *MAT_240: Added support for cohesive shell formulation +/-29.
- Scale load curve, LCSRS, of *MAT_ADD_EROSION when used with *INCLUDE_TRANSFORM.
- Fix incorrect results when using *MAT_TABULATED_JOHN-SON_-COOK/*MAT_224 with table LCKT defined and the first abscissa value, temperature, is negative.
- Fix spurious element deletion when using table for LCF in *MAT_TABULATED_JOHNSON_COOK/*MAT_224 and *MAT_TABULA-TED_JOHNSON_COOK_GYS/*MAT_224GYS.
- Error terminate with message, KEY+1142, if *MAT_ADD_EROSION is applied to resultant materials 28,116,117,118,130,139,166,170 and 98(with 1 intg point).
- Increase robustness of *MAT_033/*MAT_BARLAT_ANISOTROPIC_-PLASTICITY for solids.
- Fix input error when using *MAT_ELASTIC_WITH_VISCOSITY_-CURVE/*MAT_060c when LCID = 0.
- Fix seg fault when using shell type 15, axisymmetric volume weighted, with *MAT_ADD_EROSION and also materials with equation-of-states.
- Store computed yield strength as history variable #6 for *MAT_255.
- Fix inconsistency for *MAT_MODIFIED_PIECEWISE_LINEAR_-PLASTICITY/*MAT_123 when ncpu < 0.
- Include original volume output to dynain file for 2D analysis when materials with an equation-of-state are used. This is needed to compute the deformation gradient when initializing a run using the dynain file.
- Fix improper stress initialization using *INITIAL_STRESS_SHELL via dynain for *MAT_018/*MAT_POWER_LAW_PLASTICITY with VP = 1.0.
- Make AOPT < 0 work for *MAT_170/*MAT_RESUL-TANT_ANISOTROPIC, i.e. with material coordinate system using *DEFINE_COORDINATE_(OPTION).
- Fix incorrect operation of TDEL for *MAT_MODIFIED_-PIECEWISE_LINEAR_PLASTICITY/*MAT_124 and *MAT_PLASTI-CITY_WITH_DAMAGE/*MAT_081/*MAT_082.
- Fix incorrect damping when using *DAMPING_PART_STIFFNESS for *MAT_16/*MAT_PSEUDO_TENSOR and *EOS_TABULATED_-COMPACTION.

- Fix incorrect computation of bulk modulus which caused complex sound speed error when using *EOS_TABULATED/EOS_09 with tabulated input.
- Fix moving part with *MAT_220 during dynamic relaxation when velocities are initialized.
- Fix convergence issue for *MAT_065/*MAT_MODIFIED_ZERILLI_-ARM-STRONG for shells when VP = 1.
- Error terminate with message, KEY+1115, if _STOCHASTIC option is in-voked for materials 10,15,24,81,98, or 123 but no *DEFINE_STOCH-ASTIC_-VARIATION or *DEFINE_HAZ_PROPERTIES keyword is present in the input file.
- Fix spurious error termination when using *DEFINE_HAZ_PROPERTIES with adaptivity.
- Fix incorrect results or seg fault for *MAT_FU_-CHANG_FOAM/*MAT_-083 if KCON > 0.0 and TBID.ne.0.
- If SIGY = 0 and S = 0 in *MAT_DAMAGE_2/*MAT_105, set S = EPS1/200, where EPS1 is the first point of yield stress input or the first ordinate point of the LCSS curve.
- Allow *MAT_ENHANCED_COMPOSITE_DAMAGE/*MAT_054 failure mechanism to work together with *MAT_ADD_EROSION for shells.
- Fix incorrect erosion behavior if *MAT_ADD_EROSION is used with failure criteria defined for *MAT_123/*MAT_MODIFIED_PIECEWISE_-LIN-EAR_PLASTICITY.
- Implement *MAT_FHWA_SOIL/*MAT_147 for 2D analysis, shell types 13, 14 and 15.
- Implement scaling of failure strain for *MAT_MODIFIED_-PIECEWISE_-LINEAR_PLASTICITY_STOCHASTIC/*MAT_123_STOCHASTIC for shells.
- Fix incorrect behavior for *MAT_LINEAR_ELASTIC_DISCRETE_-BEAM/*MAT_066 when using damping with implicit (static) to explicit switching.
- Fixed *MAT_FABRIC/*MAT_034 with the negative unloading curve op-tion. When searching for the intersection point of the load and unload curves, and extrapolation of one of the curves was needed to find the inter-section point, the extrapolated stress was calculated incorrectly causing un-predictable behavior.
- Fixed fabric material forms 0 and 1 when used with a reference geometry. There were two problems, both occurring when there are mixed quad and triangular elements in the same block. A flaw in the strain calculation was leading to possible NaN forces in the elements. When a reference geometry was not used, the forces from triangular elements in mixed element blocks were 2 times too high.
- Added a new option for *MAT_SPOTWELD called FMODE. The FMODE option is available for DMGOPT = 10, 11, and 12. When the failure function is reached, and when FMODE > 0.0 and < 1.0, the value of FMODE will de-termine if a weld will fail immediately or will have damage initiated. The

failure function may include axial, shear, bending and torsion terms. If the sum of the squares of the shear and torsion terms divided by the sum of the square of all terms is greater than FMODE, then the weld will fail immediately. Otherwise, damage will be initiated.

- Enabled OPT = -1 on *MAT_SPOTWELD for brick elements which had not worked previously. Also, fixed TRUE_T when used with brick element forms 0, 1, and -1.
- Fixed spot welds with DMGOPT = 12 by removing warning STR+1327 which made it impossible to set a small value of RS without triggering this warning, or without setting EFAIL smaller. Setting EFAIL small however could lead to damage initiation by plastic strain when the user wanted only initiation by the failure function.
- If DMGOPT = 10, 11, or 12 and EFAIL = 0, on *MAT_SPOTWELD, damage will now initiate only by the failure function. If EFAIL > 0, then damage will initiate be either then failure function or when plastic strain exceeds EFAIL. Prior to this version, damage could initiate when plastic strain exceeds zero if the user set EFAIL = 0. This behavior is still true for DMGOPT = 0, 1, or 2, but no longer for DMGOPT = 10, 11, or 12.
- Allow solid spot welds and solid spot weld assemblies to have up to 300 points in the running average that is used to smooth the failure function. In other words, up to NF = 300 is possible.
- Fixed a problem with brick spot weld assemblies when OPT = 0 failure is used without defining any weld resultant values. Welds were being immediately deleted.
- Added new PID option for *DEFINE_SPOTWELD_FAILURE (applies to *MAT_SPOTWELD, OPT = 10). Changes the Card 3 input for static strength values to use part set ID’s rather than material ID’s.
- Modified shell *MAT_214/*MAT_DRY_FABRIC to calculate fiber strains based on the current distance between the points where the fibers intersect with the element edges. Previously, they were calculated from the rate-of-deformation, but this was not as accurate as the new total strain measure.
- Fixed unit scaling for GAMAB1 and GAMAB2 on *MAT_DRY_FABRIC. We were incorrectly transforming them as stress.
- Reworked the plastic stress update in *MAT_225/*MAT_VISCOPLASITC_-MIXED_HARDENING to prevent a divide by zero.
- Enabled *MAT_ADD_EROSION to be used with beams that have user defined integration. Memory allocation was fixed to prevent memory errors.
- Fixed *MAT_106 when used with tshell form 5 or 6. The elastic constants used in the assumed strain field were not reasonable.
- Fix issue that could have led to problems using *MAT_054 (or *MAT_058 or *MAT_158) in combination with TFAIL/TSIZE.gt.0.0 and damping.
- *MAT_054 - *MAT_ENHANCED_COMPOSITE_DAMAGE:
  - Add possibility to use failure criterion in *MAT_054 for solids in a transversal isotropic manner. It is assumed that the material 1-

direction is the main axis and that the behavior in the 2-3 plane is isotropic. This feature is invoked by setting TI = 1 in *MAT_054.

- *MAT_058 - *MAT_LAMINATED_COMPOSITE_FABRIC:

  - Bugfix for shear stiffness behavior in *MAT_058 when using a table definition for GAB and only providing stress-strain-curves for positive shear.
  - Bugfix for strain-rate dependent stiffness behavior in *MAT_058 when using a table definition for EA, EB or GAB under compressive loading.
  - Add default values for strengths (XT,XC,YT,YC,SC) 1.e+16 for *MAT_-058. If no values for the strengths were defined, unpredictable things could have happened.

- *MAT_138 - *MAT_COHESIVE_MIXED_MODE:

  - Store total mixed-mode and normal separation (delta_II & delta_I) on history variables 1&2 for *MAT_COHESIVE_MIXED_MODE (*MAT_-138). This is only for post-processing and should not lead to any changes in the results.

- *MAT_157 - *MAT_ANISOTROPIC_ELASTIC_PLASTIC:

  - Add Tsai-Hill failure criterion (EXTRA = 2).
  - Allow strain-rate dependent strength values (XT,XC,YT,YC,ZT,ZC,SXY,SYZ,SZX) using *DEFINE_CURVE. This is available for Tsai-Wu (EXTRA = 1) and Tsai-Hill.
  - Fixed bug in using *MAT_157 with IHIS.gt.0 for shells. Thickness strain update d3 was not correct and plasticity algorithm may have failed.
  - Add additional option to IHIS in *MAT_157 for SHELLs.
  - Now also the strength values (XT,XC,YT,YC,SXY) may be initialized via *INITIAL_STRESS_SHELL. See variable IHIS and remarks in the User's Manual for details of initializing various blocks of material parameters.

- *MAT_215 - *MAT_4A_MICROMEC:

  - Add new material *MAT_215 that is a micromechanical material model that distinguishes between a fiber/inclusion and a matrix material. The material is intended for anisotropic composite materials, especially for short (SFRT) and long fiber thermoplastics (LFRT). This model is available for shells, tshells and solids.

- *MAT_225 - *MAT_VISCOPLASTIC_MIXED_HARDENING:

        - Fixed bug in *MAT_225 (*MAT_VISCOPLASTIC_MIXED_HARDENING) when using a table for LCSS together with kinematic hardening.

    - *MAT_261 - *MAT_LAMINATED_FRACTURE_DAIMLER_PINHO:
      *MAT_262 - *MAT_LAMINATED_FRACTURE_DAIMLER_CAMANHO:

        - Allow table input for fracture toughness values for mats 261/262.Table represents fracture toughness vs. element length vs. strain rate (shells, tshells, solids)
        - Fixed bug in mats 261/262 when using *DAMPING_PART_STIFFNESS together with RYLEN = 2 in *CONTROL_ENERGY.
        - Correct shear failure behavior in *MAT_262. This will most probably have no effect to any real application, but could be seen in very special 1-element tests.

    - Changed storage of history variables for *MAT_249 (*MAT_REINFORCED_THERMOPLASTIC). A new variable POSTV controls which variables are written and at what history variable location in d3plot.
    - *MAT_254 (*MAT_GENERALIZED_PHASE_CHANGE) can now be used with shell elements and thermal thick shells.
    - Added flag 'EZDEF' to *MAT_249_UDFIBER. In this case the last row of the deformation gradient is replaced by 0-0-1.
    - Add opt. damage limitation curve/table LCDLIM for *MAT_ADD_GENERALIZED_DAMAGE.
    - Add pre-defined damage tensors option PDDT to *MAT_ADD_GENERALIZED_DAMAGE.
    - *MAT_ADD_GENERALIZED_DAMAGE now works for solid elements (only shells in R9).
    - Add optional failure criterion FFCAP to *MAT_100 with OPT = -1 or 0.
    - Enable *MAT_ADD_COHESIVE to be used in implicit analysis.
    - Add alternative version of *MAT_280 invoked by new flag on 1st card. It is a physically based damage model with 4 new parameters.
    - Enable *DEFINE_CONNECTION_PROPERTIES' option PROPRUL>=2 to be used with spotweld clusters, i.e. not only 1 hex element but several (via *DEFINE_HEX_SPOTWELD_ASSEMBLY or RPBHX > 1 on *CONTROL_SPOTWELD_BEAM).
    - Enable *MAT_ADD_EROSION to be safely used with material models that have more than 119 history variables, for now the new limit is 169 (e.g. necessary for *MAT_157 with IHIS = 7).
    - Add Tsai-Wu failure criterion to *MAT_157 for solid and shell elements invoked by EXTRA = 1 on card 6 and corresponding parameters on cards 8 and 9.
    - Add viscoelastic option to *MAT_187 (SAMP-1). Rate dependent Young's modulus and associated settings can be defined on new optional card 5.

- Add new option IRNG for *DEFINE_STOCHASTIC_VARIATION to govern random number generation (deterministic or true random).
- Add option to define element size dependent parameters EN and SN for *MAT_120 and *MAT_120_JC by setting them to negative values (curves).
- Minor improvements for *MAT_252: Optional output of damage initiation information and more post-processing history variables.
- If the first abscissa value of *MAT_224's failure strain curve LCG is negative, it is assumed that all abscissa values are natural logarithms of a strain rate.
- Put *MAT_100_DA's "failure function" value to history variable 18.
- Add optional in-plane failure strain to *MAT_169 (ARUP_ADHESIVE): new input parameter FSIP.
- *MAT_USER_DEFINED_MATERIAL_MODELS now provides a few more variables for cohesive elements, i.e. additional arguments in subroutines umatXXc: temperature, element size, implicit rejection flag, integration point identifier, and total number of integration points.
- A modified version of the 3-parameter Barlat model (*MAT_036) is introduced as *MAT_EXTENDED_3-PARAMETER_BARLAT. In this model, hardening in 00, 45, 90, biaxial and shear can be specified as load curves. Furthermore, r-values in 00, 45, 90, biaxial and shear can be specified in terms of load curves vs plastic strain or constants. This is an extension of hardening law 7 of the original 3-parameter Barlat model.
- Improve implicit version of *MAT_098/*MAT_SIMPLIFIED_JOHNSON_-COOK.
- *MAT_181/*MAT_SIMPLIFIED_RUBBER/FOAM is now supported for 2D implicit simulations.
- Fixed issue in which *MAT_WINFRITH_CONCRETE wrote d3crack data too frequently.
- *EOS_JWL now has an AFTERBURN option. This adds afterburn energy to the EOS, where the energy can be added at a constant or linear rate, or can be added according to Miller's extension.
-
- *MAT_084 (*MAT_WINFRITH_CONCRETE) with predefined units (CONM < 0) is now transformed correctly with *INCLUDE_TRANSFORM.
- User-defined materials for Hughes-Liu beams can now be used with implicit analysis by defining the appropriate tangent modulus in the supplied routine urtanb.
- User-defined cohesive materials can now be used with implicit analysis by defining the appropriate tangent stiffness.
- *MODULE for user-defined materials and other user-defined capabilities:

    - A new command line option "module = filename" is added to load one module file without changing the input deck. It provides back compatibility to input deck without the MODULE keywords.
    - The system paths defined in LD_LIBRARY_PATH are also included for searching module files for those filenames start with "+".

- Add shell implementation to *MAT_277 (*MAT_ADHESIVE_CURING_-VISCOELASTIC).
- Add *MAT_278 for carbon fiber prepreg compression forming simulation. This material model is available for both solid and shell formulations.
- Add *MAT_293 non-orthogonal material model for carbon fiber prepreg forming simulation. This material model is only available for shell formulations.
- *MAT_260A:
  - Extend *MAT_260A to include solid elements.
  - Add a new option XUE for Xue's fracture criteria/theory for *MAT_-M260A (solid elements only).
- *MAT_260B:
  - Set default values for P's and G's in *MAT_260B.
  - Add a length scale to the fracture limit. The fracture limit strongly depends on the length scale in the measurement.
  - Add a new fracture criterion to *MAT_260B (Xue and Wierzbicki, Int. J. solids and Structures 46 (2009) 1423-1435). When the option XUE is activated, an additional card is needed, for example:

```
$     ef0     plim       q     gama       m
     0.70    925.7   0.970    0.296    2.04
```

- *MAT_037:
  - Improve *MAT_037 with negative R value in implicit calculation. The modification will allow the implicit method stress calculation to be more accurate.
  - Add a new option NLP2 to calculate formability index in *MAT_037. The previous method (option NLP_FAILURE) was based on the effective strain method, which assumes that necking happens at one instant. In fact, it might happen over a longer process. The new method calculates the damage accumulation.
- Add *MAT_165B (*MAT_PLASTIC_NONLINEAR_KINEMATIC_B) for shells and solids.

• **MPP**

- Fix the report of decomp balance (shown as "Normalized element costs assigned during decomposition" in the d3hsp file), which was broken in r109760
- MPP decomposition has not been properly balanced since r112652 due to a bug in that revision

- Fix MPP SYNC error due to inconsistent summation in *CONTACT_SLIDING_ONLY_PENALTY.
- Allow real values as the scale multipliers for "memory=" on the command line. For example, "memory = 2.5G memory2 = 1.1G" and the like.
- MPP: fix support for nlq setting in *CONTROL_SOLUTION which was not being honored on processors other than 0.
- Significant improvements in MPP groupable routines for FORMING contact.
- MPP: increase contact release distance for SINGLE_SURFACE contacts in the case of a node coming into contact with a solid element. The previous interpretation was releasing when the contact penetration was 0.5*solid thickness, but now when the node passes below the solid surface by 0.5*solid thickness (which is different by the half thickness of the slave material, in the case of a shell slave node).
- MPP: fix for viscous damping in automatic tiebreak contact.
- Implement new bucket sort based extent testing for MPP single surface contact.
- Added MPP support for *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_LUBRICATION.
- Fixed *CONTROL_MPP_PFILE so that it honors ID offsets from *INCLUDE_TRANSFORM for parts, part sets, and contact IDs referenced in "decomp { region {" specifications.
- Furthermore, such a region can contain a "local" designation, in which case the decomposition of that region will be done in the coordinate system local to the include file, not the global system. For example:
- *CONTROL_MPP_PFILE decomp { region { partset 12 local c2r 30 0 -30 0 1 0 1 0 0 } } would apply the c2r transformation in the coordinate system of the include file, which wasn't previously possible. The local option can be useful even if there are no such transformations, as the "cubes" that the decomposition uses will be oriented in the coordinate system of the include file, not the global system.
- Furthermore, the following decomposition related keywords now have a _LOCAL option, which has the same effect:

  *CONTROL_MPP_DECOMPOSITION_PARTS_DISTRIBUTE_LOCAL
  *CONTROL_MPP_DECOMPOSITION_PARTSET_DISTRIBUTE_LOCAL
  *CONTROL_MPP_DECOMPOSITION_ARRANGE_PARTS_LOCAL
  *CONTROL_MPP_DECOMPOSITION_CONTACT_DISTRIBUTE_LOCAL

- MPP job performance profiles are output to both .csv and .xy files.

- **OUTPUT**

- Fix for writing d3plot file when individual output states exceed 8GB in single precision
- Added new option *INTERFACE_SPRINGBACK_EXCLUDE to exclude selected portions from the generated dynain file.
- Add a new option to *INTERFACE_COMPONENT_FILE to output only 3 degrees of freedom to the file even if the current model has 6.
- Fix missing plastic strain tensors in d3plot when STRFLG in *DATA-BASE_-EXTENT_BINARY is set and INTSTRN = 1 in *INTERFACE_-SPRING-BACK.
- Fix no output to bndout when run with q = remap even though the keyword *DATABASE_BNDOUT was present in the remap run but was not present in the initial run.
- Fix d3plot output frequency which was different from the dt specified in *DATABASE_BINARY_D3PLOT when *CONTACT_AUTO_MOVE is used.
- Fix stress output to elout for solid elements which was in the global coordinates instead of local coordinates when CMPFLG = 1 in *DATABASE_EXTENT_BINARY and OPTION1 > 0 in *DATABASE_-ELOUT.
- Fix incorrect mass properties for solids in ssstat file when using *DATABASE_SSSTAT_MASS_PROPERTY.
- Fix seg fault during writing of dynain file if INSTRN = 1 in *INTERFACE_-SPRINGBACK and STRFLG.ne.0 in *DATABASE_-EXTENT_BINARY and the *DATABASE_EXTENT_BINARY comes after *INTERFACE_SPRING-BACK.
- If HYDRO is nonzero in *DATABASE_EXTENT_BINARY, LS-PrePost will now combine the solid and shell internal energy densities when fringing 'Internal Energy Density' in the Misc menu.
- By putting SIGFLG/EPSFLG = 3 in *DATABASE_EXTENT_BINARY, the stresses and plastic trains are excluded not only for shell elements but also for solids. This applies to d3plot and d3eigv.
- Added new file option *DATABASE_BINARY_INTFOR_FILE to define interface file name.
- Fixed legend in bndout in the case of multiple *BOUNDARY_PRESCRIBED_MOTION_SET_ID.
- Fix corrupt d3part data caused by DECOMP = 4 in *DATABASE_-EXTENT_BINARY.
- Fixed the legend of ssstat in binout.
- Added *DATABASE_EXTENT_SSSTAT_ID. The subsystem id will be included in the ASCII ssstat file.
- Fixed bug in stbout (seatbelt output) if NEWLEG = 0 in *CONTROL_OUTPUT.
- Fixed bug in which DECOMP = 2 corrupted d3part.
- Fixed d3plot bug if dynamic relaxation was activated in the input deck.
- Added another digit for coordinates in *NODE in dynain, e.g., what was written as 0.999266236E+00 is now written as 9.992662368E+00.

- Added *DATABASE_EXTENT_BINARY_COMP for alternative (simpler) control of output to d3plot and d3eigv.

  - Output control flags: 0-no 1-yes
  - IGLB : Global data
  - IXYZ : Current coordinate
  - IVEL : Velocity
  - IACC : Acceleration
  - ISTRS: 6 stress data + plastic strain
  - ISTRA: 6 strain data
  - ISED : Strain energy density

  This command can be used in combination with regular *DATABASE_EXTENT_BINARY but will disable most of the options in the latter, including output of extra history variables.
- Bugfix: *DATABASE_TRACER without the optional NID parameter was read incorrectly when used with *INCLUDE_TRANSFORM, but is now fixed
- Fixed incomplete output from Windows version of LS-DYNA. This affected demtrh (*DATABASE_TRACER_DE) and curvout (*DATABASE_-CURVOUT).

- **Restarts**

  - Enable definition of sensors in full restarts.
  - For a small restart in MPP, the value of "memory=" (M1) needed for each processor is stored in the dump files. This is the minimum requirement to read back the model info. If the value of "memory2=" (M2) is specified on the command line, the code will take the maximum of M1 and M2.
  - Fix input error during structured input when using *INITIAL_VELOCITY_-GENERATION and *CHANGE_VELOCITY_-GENERATION together in a full deck restart.
  - Fix incorrect full deck restart analysis if initial run was implicit and the full deck restart run is explicit. This affects MPP only.
  - Fix insufficient tying of nodes when doing full deck restart and the contact is newly added to the restart involving newly added parts. This applies to SMP contact only.
  - Fix incorrect velocity initialization for SMP full deck restart when using *INITIAL_VELOCITY_GENERATION and *INITIAL_VELOCITY_-GENERATION_START_TIME.
  - Fix incorrect initialization of velocities for SMP full deck restart when using *CHANGE_VELOCITY_OPTION & *INITIAL_VELOCITY_OPTION. Velocities of existing parts defined by *STRESS_INITIALIZATION should not be zeroed.
  - Fix *CHANGE_CURVE_DEFINITION for curve specifying d3plot output.

- Fixed bug in full deck restart if the new mesh has different part numbers.

- ***SENSOR**

  - Fix a bug regarding *SENSOR_JOINT_FORCE for *CONSTRAINED_-JOINT_STIFFNESS, that was triggered when the force refers to a local coordinate system.
  - Add the option of "ELESET" to *SENSOR_CONTROL to erode elements.
  - Add the option of NFAILE to *SENSOR_DEFINE_MISC to track number of eroded elements.
  - Fix a bug that was triggered when using a sensor to control spotwelds. The bug was triggered when the spotweld-connected nodal pairs happen to belong to more than 1 core (MPP only).
  - Add FAIL option to *SENSOR_DEFINE_ELEMENT to track the failure of element(s).
  - Fix a bug related to *SENSOR_DEFINE_FUNCTION when there are more than 10 sensor definitions.
  - Effect of TIMEOFF in *SENSOR_CONTROL is implemented for TYPE = PRESC-ORI.
  - *SENSOR_CONTROL can be used to control *BOUNDARY_PRE-SCRIBED_-ORIENTATION_RIGID.
  - Add optional filter id to SENSORD of *DEFINE_CURVE_FUNCTION.
  - Enable *CONSTRAINED_JOINT_..._LOCAL to be monitored by *SEN-SOR_-DEFINE_FORCE.
  - Allow moments in SPCFORC and BNDOUT to be tracked by *SENSOR_-DEFINE_FORCE.
  - Fix *SENSOR_CONTROL using TYPE="PRESC-MOT" which was not switching at all.

- **SPG (Smooth Particle Galerkin)**

  - MPP is ready in 3D SPG fluid particle stabilization (ITB = 1 & 2 in *SEC-TION_SOLID_SPG).
  - Added one SPG control parameter (itb = 2) for semi-brittle fracture analysis. In comparison to itb = 0 or itb = 1, itb = 2 is more efficient in modeling the fragmentation and debris in semi-brittle fracture analysis such as impact and penetration in concrete materials.
  - Fixed a bug related to E.O.S. in SPG.
  - Removed some temporary memory allocations to improve efficiency.
  - Changed the sequence of SPG initialization so that all state variables are properly initialized.
  - Subroutines were developed for SPG failure analysis with thermal effects. Both explicit and implicit (diagonal scaled conjugate gradient iterative only) SPG thermal solvers are available in SMP version only. However, thermal effect is applied only on material properties, which means thermal induced

deformation (i.e., thermal strain or thermal expansion) is not currently included.

- Modified *MAT_072R3 for SPG method in concrete applications.
- Fixed a bug for SPG method in using continuum damage mechanics. (IDAM = 0).
- Added the "fluid particle algorithm" (itb = 1) to SPG method. This algorithm is implemented in R10.0 as an alternative to the (itb = 0) option in previous version to enhance the numerical stability for SPG method. Users are recommended to use this new option for their ductile failure analysis.

- **SPH (Smooth Particle Hydrodynamics)**

  - Add ITHK flag in *CONTROL_SPH, card 3. If flag is set to 1, the volume of the SPH particles is used to estimate a node thickness to be employed by contacts.

    - Affects *AUTOMATIC_NODES_TO_SURFACE and *CONTACT_-2D_-NODE_TO_SOLID.
    - The thickness calculated by ITHK = 1 is used only if SST or OFFD are set to zero in the contact cards definitions.

  - Add SOFT = 1 option to *CONTACT_2D_NODE_TO_SOLID. This should help obtain reasonable contact forces in axisymmetric simulations. Default penalty PEN is 0.1 when SOFT = 1.
  - Implemented non-reflecting boundary conditions for SPH using a new keyword *BOUNDARY_SPH_NON_REFLECTING.
  - Bug fix for renormalized SPH formulations with symmetry planes. The renormalization was slightly incorrect in the vicinity of symmetry planes.
  - Density smoothing in SPH formulations 15 and 16 is now material sensitive. The smoothing only occurs over neighbors of the same material.
  - Resolved an MPP bug in SPH total Lagrangian formulations (FORM = 7/8) which was causing strain concentrations at the interfaces between CPU zones.
  - SPH total Lagrangian (FORM = 7/8) in SMP was pretty much serial, hence much slower than forms 0 or 1. SPH with FORM 7 and 8 now scales properly.
  - Added support for FORMs 0/1 in axisymmetric. Until now, renormalization was always active (equivalent to FORM = 1) which can be problematic for very large deformations or material fragmentation.
  - Improved tracer particles output for SPH: Use normalized kernel function for interpolation between particles.
  - Implemented enhanced fluid flow formulations (FORMs 15/16) with pressure smoothing.
  - Recode SPH neighborhood search algorithm to reduce the memory requirement and produce consistent results from MPP and HYBRID code.

- *DEFINE_ADAPTIVE_SOLID_TO_SPH now reports both active and inactive adaptive SPH particles in the fragment file sldsph_frag. This file gives a report of nodal mass, coordinates, and velocities.
- MPP now supports:
  - SPH type 3 inflow
  - Multiple *BOUNDARY_SPH_FLOW
  - Bulk viscosity option for SPH
- Sort SPH by part and then node ID to ensure consistent results while changing order of input files.
- *DEFINE_SPH_TO_SPH_COUPLING:
  - Corrected the SPH sphere radius (half of the particles distance) for node to node contact detecting algorithm.
  - Updated masses for SPH node to node coupling with damping contact force option.
  - Added a new option (Soft = 1) for SPH to SPH coupling: contact stiffness comes from particles masses and time step for softer contact.
- *DEFINE_ADAPTIVE_SOLID_TO_SPH:
  - Updated temperature transfer (from solid elements to SPH particles) when converting solid elements into SPH particles with ICPL = 1, IOPT = 0.
  - Bug fixed when part ID for newly generated SPH particles is smaller than the original SPH part ID.
  - Introduced a new pure thermal coupling between SPH part and solid parts with ICPL = 3 and IOPT = 0 option (no structural coupling provided).
  - Added a thermal coupling conductivity parameter CPCD. Applies to ICPL = 3 option.
  - Normalized the nodal temperatures for the corner SPH particles with ICPL = 3 and IOPT = 0 option (MPP only).
  - Extended ICPL = 3 and IOPT = 0 option to Lagrangian formulation (form = 7, 8).
- *BOUNDARY_SPH_SYMMETRY_PLANE:
  - Added in an error message if TAIL and HEAD points are at the same location.
- *CONTACT_2D_NODE_TO_SOLID:
  - Added a variable OFFD to specify contact offset.

  - Added a new option IEROD = 2 in *CONTROL_SPH in which SPH particles that satisfy a failure criterion are totally eactivated and removed from domain interpolation. This is in contrast to IEROD = 1 option in which particles are partially deactivated and only stress states are set to zero.
  - Added *MAT_SPH_VISCOUS (*MAT_SPH_01) for fluid-like material behavior with constant or variable viscosity. Includes a Cross viscosity model.
  - Output strain rates for SPH particles to d3plot, d3thdt, and sphout file.
  - Added support of *MAT_ADD_EROSION, including GISSMO and DIEM damage, for SPH particles.
  - Echo failed SPH particles into d3hsp and messag file.
  - *DEFINE_SPH_INJECTION:
    - Changed the method of generating SPH particles. SPH particles will be generated based on the injection volume (injection area*injection velocity*dt)*density from the material model, resulting in more consistent particle masses and particle distribution.
    - Offset injecting distance inside each cycle so that outlet distance will be consistent for different outlet SPH layers.
    - Corrected mass output in d3hsp.

- **Thermal Solver**
  - Modify the thermal solver routines so they return instead of terminating, so that *CASE works properly.
  - *MAT_THERMAL_USER_DEFINED: Fixed bug in element numbering for IHVE = 1.
  - Accept load curve input for dtmin, dtmax and dtemp in *CONTROL_THERMAL_TIMESTEP. As usual if a negative integer number is given its absolute value refers to the load curve id.
  - The temperature results for the virtual nodes of thermal thick shells are now accounted for in *LOAD_THERMAL_D3PLOT. For the mechanics-only simulation thermal thick shells have to be activated.
  - New contact type for thermal solver that models heat transfer from and to a shell edge onto a surface (*CONTACT_..._THERMAL with ALGO > 1):
    - Shells have to be thermal thick shells.
    - Shells are on the slave side.
    - So far only implemented for SMP.
    - Includes support for quads and triangles.
  - New keyword *BOUNDARY_THERMAL_WELD_TRAJECTORY for welding of solid or shell structures.
    - Keyword defines the movement of a heat source on a nodal path (*SET_NODE).
    - Orientation given either by vector or with a second node set.

- Works for coupled and thermal only analyses.
- Allows for thermal dumping.
- Different equivalent heat source descriptions available.
- Can also be applied to tshells and composite shells.
- Weld torch motion can be defined relative to the weld trajectory.

- Solid element formulation 18 now supports thermal analysis.
- Thermal solver now supports the H8TOH20 option of *ELEMENT_SOLID. This includes support of *INITIAL_TEMPERATURE condition for the extra 12 nodes generated by H8TOH20.
- Thermal solver now supports the H8TOH27 option of *ELEMENT_SOLID.
- Explicit Thermal Solver

  - *CONTROL_EXPLICIT_THERMAL_SOLVER: Implement an explicit thermal solver and adapt it to support multi-material ALE cases.
  - *CONTROL_EXPLICIT_THERMAL_PROPERTIES: Enter thermal properties for the explicit thermal solver.
  - *CONTROL_EXPLICIT_THERMAL_CONTACT: Implement a thermal contact for the explicit thermal solver.
  - *CONTROL_EXPLICIT_THERMAL_ALE_COUPLING: Implement a thermal coupling between ALE and Lagrangian structures for use by the explicit thermal solver.
  - *CONSTRAINED_LAGRANGE_IN_SOLID_EDGE: For the explicit thermal ALE coupling, allow the heat transfer through the shell edges if _EDGE is added to *CONSTRAINED_LAGRANGE_IN_SOLID.
  - *CONSTRAINED_LAGRANGE_IN_SOLID: For the explicit thermal solver, add work due to friction to the enthalpies of ALE and structure elements coupled with *CONSTRAINED_LAGRANGE_IN-_SOLID (CTYPE = 4).
  - *CONTROL_EXPLICIT_THERMAL_INITIAL: Initialize the temperatures for the explicit thermal solver.
  - *CONTROL_EXPLICIT_THERMAL_BOUNDARY: Control boundary temperatures for the explicit thermal solver.
  - *CONTROL_EXPLICIT_THERMAL_OUTPUT: Output the temperatures at element centers for the explicit thermal solver.
  - *DATABASE_PROFILE: For the explicit thermal solver, output temperature profiles.

- **Miscellaneous**

  - *INITIAL_LAG_MAPPING: Implement a 3D to 3D lagrangian mapping and map the nodal temperatures.
  - *CONTROL_REFINE_SHELL and *CONTROL_REFINE_SOLID: Add a parameter MASTERSET to call a set of nodes to flag element edges along which new child nodes are constrained.

- *BOUNDARY_PRESCRIBED_MOTION_SET_SEGMENT: Add DOF = 12 to apply velocities in local coordinate systems attached to segments.
- Fixed bug occurring when a part has non-zero *DAMPING_-PART_STIFFNESS, AND is defined using *PART_COMP-OSITE, AND the MIDs referenced by the different integration points have different material types. Symptoms could include many types of unexpected behavior or error termination, but in other cases it could be harmless.
- *DAMPING_FREQUENCY_RANGE (including _DEFORM option): Improved internal calculation of damping constants such that the level of damping more accurately matches the user-input value across the whole of the frequency range FLOW to FHIGH. As an example, for CDAMP = 0.01, FLOW = 1 Hz and FHIGH = 30 Hz, the actual damping achieved by the previous algorithm varied between 0.008 and 0.012 (different values at different frequencies between FLOW and FHIGH), i.e. there were errors of up to 20% of the target CDAMP. With the new algorithm, the errors are reduced to 1% of the target CDAMP. This change will lead to some small differences in results compared to previous versions of LS-DYNA. Users wishing to retain the old method for compatibility with previous work can do this by setting IFLG (7th field on Card 1) to 1.
- Fixed bug that could cause unpredictable symptoms if Nodal Rigid Bodies were included in the Part Set referenced by *DAMPING_FREQUENCY_-RANGE or *DAMPING_FREQUENCY_RANGE_-DEFORM. Now, the _DEFORM option silently ignores NRBs in the Part Set while *DAMPING_-FREQUENCY_RANGE (non _DEFORM option) damps them.
- Fixed bug in *PART_COMPOSITE: if a layer had a very small thickness defined, such as 1E-9 times the total thickness, that layer would be assigned a weighting factor of 1 (it should be close to zero).
- Fix errors in implementation of *DEFINE_FILTER type CHAIN.
- Fix for *INTERFACE_LINKING_LOCAL when LCID is used. During keyword processing, the LCID value was not properly converted to internal numbering.
- Switch coordinates in keyword reader to double precision.
- Change "Warning" to "Error" for multiply defined materials, boxes, coordinate systems, vectors, and orientation vectors. The check for duplicate section IDs now includes the element type and remains a warning for now, because SPH is still detected as a SOLID. Once that is straightened out, this should be made an error.
- Add "TIMESTEP" as a variable for *DEFINE_CURVE_FUNCTION. This variable holds the current simulation time step.
- Fix a bug for the case of CODE = 5 in *DEFORM-ABLE_TO_RIGID_AUTOMATIC. (Fields 3 to 8 are now ignored.)
- Issue error message and terminate the simulation when illegal ACTION is used for *DEFINE_TRANSFORM.
- Add option of POS6N for *DEFINE_TRANSFORM to define transformation with 3 reference nodes and 3 target nodes.

- Fix a bug that can occur when adaptive elements are defined in a file included by *INCLUDE_TRANSFORM.
- Merge *DEFORMABLE_TO_RIGID_AUTOMATIC cards if they use the same switch time. This dependency of results on the order of the cards and also gives better performance.
- If *SET_PART_OPTION is used, a "group_file" will be created which can be read into LS-Prepost (Model > Groups > Load) for easy visualization of part sets.
- Forces on *RIGIDWALL_GEOMETRIC_CYLINDER can now be subdivided into sections for output to rwforc. This gives a better idea of the force distribution along the length of the cylinder. See the variable NSEGS.
- Added the keywords *DEFINE_PRESSURE_TUBE and *DATABASE_-PRTUBE for simulating pressure tubes in pedestrian crash.
- Fix non-effective OPTIONs DBOX, DVOL, DSOLID, DSHELL, DTSHELL, DSEG for *SET_SEGMENT_GENERAL to delete segments.
- Fix incorrect transformation of valdmp in *DAMPING_GLOBAL with *INCLUDE_TRANSFORM.
- Make *SET_NODE_COLLECT work together with *NODE_SET_MERGE.
- Fixed bug in adaptivity for *INCLUDE_TRANSFORM if jobid is used.
- Bugfix: *INTERFACE_SSI with blank optional card is now read in correctly.

Capabilities added to create LS-DYNA R11:

See release notes (published separately) for further details.

- ***AIRBAG**

  - Fix an airbag bug that can cause MPP to hang when reference geometry is used in a huge model.
  - Limit the EXCP option (excluding application of bag pressure to partition parts) in
    *AIRBAG_INTERACTION to *AIRBAG_WANG_NEFSKE and *AIRBAG_-HYBRID only.
  - Allow no-pressure venting parts for *AIRBAG_WANG_NEFSKE.
  - Output material leakage information, including leakage rate, in abstat for *AIRBAG_WANG_NEFSKE.
  - Allow airbag venting holes of *AIRBAG_HYBRID to be represented by a part set.
  - Enhance the robustness of *AIRBAG_INTERACTION to curtail instability, especially in MPP, due to excessive mass exchange owing to numerical noise.
  - Support UP (uniform pressure) airbag in MPP full deck restart. (We use the equivalent terms UP airbag or CV (control volume) airbag for airbags that are NOT based on ALE or particle (CPM) methods.)

- Treat PPOP for *AIRBAG_WANG_NEFSKE similar to other types of UP airbag - once it pops, it stays open.
- *AIRBAG_CPM_INTERACTION:
    - Support slave bag with chamber definition.
    - Support master/slave bags having multiple gas components.
- New CPM (*AIRBAG_PARTICLE) particle-to-fabric and particle-to-particle contact algorithms. These are about 3x faster than the old scheme and it is the default method without any input modifications.
- Support *SENSOR_SWITCH_SHELL_TO_VENT for SMP.
- New variable TSPLIT in *AIRBAG_PARTICLE to activate particle splitting. Particles that exit by porosity leakage or a vent are removed from the system. If TSPLIT is set, the code keeps track of the number of removed particles (A) and active particles (B) every 200 cycles after time TSPLIT. Once A is greater than B, each active particle will be split into A/B + 1 particles for a better particle density in the volume.
- Enable OpenMP. This allows MPP hybrid to scale better.
- Support the exterior air drag force capability while switching from particle to UP formulation.
- Enable *DEFINE_FUNCTION for user to control the airbag particle deflection angle. Three airbag history variables are passed to the user defined function to control the surface roughness: bag pressure, bag volume, current time.
- Support *MAT_ADD_AIRBAG_POROSITY_LEAKAGE
    - Support FAC in the option can be defined as a factor/define_curve/define_function
    - If defined_function, it only works with vopt 7/8. the porosity leakage velocity is a function of current part pressure and time.
- Support molar-fraction-based inflator gas flow rate curve LCMASS (MOLEFRACTION option) even after particle airbag switches to CV airbag via the TSW switch.
- Enhance dm_out calculation to treat trunction error.
- Support specification of inflator orifice area as function of time to control the distribution of mass flowrate.
- Fix bug affecting particle-to-UP switch for multi-chambered airbag. The bug was introduced in r82352 and showed up if gas contains multi-species.
- *AIRBAG_LINEAR_FLUID:
    - Prior to R11, the tangent stiffness matrix resulting from a pressure-volume relationship in simple airbag models was merely an approximation and sometimes resulted in poor convergence. A justification of the approximation is that a rigorous treatment reveals that coupling

terms between all nodes in bag result in a dense stiffness matrix and slow execution. Exploiting the low rank structure of this matrix contribution and the use of Sherman-Morrison formula, the correct (or at least close to correct) stiffness matrix is now implemented without sacrificing speed and at the same time obtaining much better convergence.

- ***ALE**
  - *ALE_STRUCTURED_MESH_CONTROL_POINTS:
    - If the S-ALE mesh has different mesh sizes and transition zones, suggest new lists of control points to create optimal mesh transitions (the new list is output in the messag file).
  - *ALE_STRUCTURED_MESH:
    - 2D version of the S-ALE code.
  - *ALE_INJECTION:
    - Inject materials in an ALE mesh (i.e. add a group to some elements and prescribe the velocities of the related nodes).
  - *ALE_PRESCRIBED_MOTION:
    - Prescribe or initialize velocities by ALE group.
  - *ALE_FSI_TO_LOAD_NODE:
    - Use the 2nd line for a directory path where to read or write the files with the fsi data (alefsi2ldnd*.tmp*). Implement the restart.
  - *CONTROL_ALE and *SECTION_ALE2D_SMOOTHING
    - mpp version of the AFAC, BFAC, CFAC and DFAC mesh smoothings in the 2D ALE code.
  - *CONTROL_ALE (END < 0, 2nd line, 2nd column):
    - If END < 0 in *CONTROL_ALE, remove the ALE mesh after |END|.
  - *CONTROL_ALE (IALEDR = 1, 4th line, 2nd column) and *CONTROL_DYNAMIC_RELAXATION:
    - Exclude ALE computations if IDRFLG = 1 unless IALEDR = 1.
  - *MAT_ADD_EROSION (IDAM > 0, 1st line, 3rd column):

        - Implement the GISSMO damage model in ALE.

    - *DATABASE_ALE_MAT:

        - Output the volume averaged pressures of each ALE groups

    - *INITIAL_DETONATION and *CONTROL_EXPLOSIVE_SHADOW:

        - Compute the lighting times by ALE explosive groups.

    - *DATABASE_ALE_BINARY:

        - Output ALE data in a binary file that can be post-treated.

    - *CONTROL_MPP_DECOMPOSITION_DISTRIBUTE_ALE_ELEMENTS (ALECOMM = 1, 1st line, 2nd column):

        - Create a MPP communicator for processors involved in ALE applications.

    - Added *ALE_STRUCTURED_MESH_TRIM so initial mesh can be trimmed to better fit the domain of interest.
    - S-ALE mesh starts from a rectangular box and the elements away from the domain of interest are then deleted based on the distance to certain geometry or part/element set. This way S-ALE mesh could be of irregular shape. Fewer elements leads to less running time.
    - Added *ALE_STRUCTURED_MESH_MOTION so S-ALE mesh can be made to follow the motion of the mass center of certain ALE material material(s).
    - Add variable TDEATH in *ALE_STRUCTURED_MESH so S-ALE mesh can be deleted during the run to save running time. All related FSI and nodal coupling cards are automatically deleted too.
    - Fix bugs affecting S-ALE (*ALE_STRUCTURED_MESH):

        - Fix sometimes improper volume fraction initialization by *INITIAL_VOLUME_FRACTION_GEOMETRY when using with ESORT = 1 in *CONTROL_SOLID.
        - Fix sometimes incorrect effect of *INITIAL_HYDROSTATIC_ALE and *ALE_AMBIENT_HYDROSTATIC when using with ESORT = 1 in *CONTROL_SOLID.

- ***BOUNDARY**

    - Speedup the input error checking for *BOUNDARY_PRESCRIBED_MOTION.

- Fix *BOUNDARY_PRESCRIBED_ORIENTATION_RIGID_DIRCOS which did not orientate the rigid body when angles are close to PI when using double precision.
- *BOUNDARY_CYCLIC is supported in selective mass scaling (*CONTROL_TIMESTEP).
- *BOUNDARY_SPC_SYMMETRY_PLANE:
  - Add a new option _SET.
  - Allow more than two definitions of symmetric planes.
  - Fix core dump when *BOUNDARY_SPC_SYMMETRY_PLANE, and *ALE_MULTI-MATERIAL_GROUP are used together.

- **Blast**
  - For *LOAD_BLAST_ENHANCED BLAST = 4, apply blast pressure to segments which reside below the ground plane with new parameter FLOOR. This allows blast pressure to be applied to a deformable floor which deflects below the ground plane, and accommodates floors which are not planar.
  - Implement blast particle-SPH coupling algorithm as invoked by variable NODID in *DEFINE_PARTICLE_BLAST (SMP). The SPH smoothing length is taken into consideration.

- **CESE**
  - Added a new prescribed boundary condition card that allows a normal velocity to be specified in the inlet boundary.
  - Added a positivity-preserving strategy for the spatial derivatives calculations to make the solver more stable (currently only CESE 2D, 3D, and CESE-FSI-IBM 2D solvers are supported).
  - Made it a fatal error to attempt to use *LOAD_BLAST_ENHANCED as a CESE boundary condition with other than a type-2 blast.
  - Improved error tolerances in the conjugate heat transfer method solver used with the immersed-boundary method FSI solver, and corrected other small related errors.
  - Any unused nodes specified in a standalone CESE problem are now treated as a fatal error. This fixes an issue with d3plot output when such orphan nodes were passed on to the d3plot file.
  - Corrected some weightings for the MPP CESE moving mesh solvers. This could have been an issue in cases where an MPP CESE shared node is next to a moving boundary.
  - Adjusted the information output by the implicit ball-vertex mesh motion solver to reflect that the absolute error tolerance is no longer used, only the relative error tolerance. This applies to both the ICFD and CESE solvers that use this mesh motion solver.
  - For the *CESE_DATABASE cards, corrected these issues:

        - Any used segments that are not in the CESE mesh are reported.
        - Element sets look up is now guaranteed to work.
        - Drag calculation output that is attempted for the incorrect FSI method, is now reported. Here are the illegal cases:
            - *CESE_DATABASE_FSIDRAG is only available with the CESE immersed boundary FSI solvers.
            - *CESE_DATABASE_SSETDRAG is not available with the CESE immersed boundary FSI solvers. In other words, it only works with the non-FSI solvers and moving mesh FSI solvers.
        - Failure to find an internal node number in *CESE_DATABASE commands is now reported.
        - Output was not being performed for the initial conditions. Instead, it is now done after the first time step.
        - Warning is given when illegal IDs are given in *CESE_DATABASE... cards. Such IDs are then ignored.

    - Changed behavior for the *CESE_DATABASE cards includes:

        - Output times are now selected in a manner similar to CESE d3plot output, meaning when the CESE simulation time has reached at least the requested output times. Also, output never occurs more than once per CESE time step.
        - Cycle number was reported in some outputs as the 'time step'. The actual solver time step is now used instead.

    - When performing d3plot output for the immersed boundary CESE FSI solver, take into account what structural elements overlay CESE elements for the Schlieren number and chemical species mass fractions.
    - Formation enthalpy concept is added to the CESE Navier-Stokes solver.
    - Many restart-related bugs were fixed in various CESE solvers.

- ***CHEMISTRY**

    - The chemistry solvers are working with the following CESE solvers:

        - 2D, 3D, and axi-symmetric Euler and Navier-Stokes solvers.
        - Immersed boundary method FSI 2D, 3D, and axi-symmetric Euler solvers.

    - Added a restart capability for the CESE chemistry solvers.

- **CONTACT**

    - Fix initialization error related to *CONTACT_GENERAL_INTERIOR that resulted in every subsequent *CONTACT_GENERAL also being treated as INTERIOR.

- Fix for spotwelds improperly deleted due to rigid body conflict when an IPBACK contact interface is in effect. The failure of the constraint-based contact to tie should not cause the weld to be deleted if the penalty-based contact ties.
- Fix reported contact forces and energies for *CONTACT_-TIED_SHELL_-EDGE_TO_SOLID.
- Fix bug for *CONTACT_ERODING_NODES_to_SURFACE. The slave nodes' thickness was ignored, which was problematic for SPH simulations, for example. SST is now taken into account for this contact.
- Fix a problem with SMOOTH contact during implicit springback.
- Enhance implicit processing of tied contact and rotational dofs on slave nodes for special cases with *CONTACT_..._CONSTRAINED_OFFSET.
- Adjust logic for unconstraining rigid body slave nodes for SMOOTH contact to be compatible with deformable to rigid switching.
- Enable the use of *CONTACT_FORMING_..._SMOOTH for implicit.
- Fix the problem of tied contact in MPP having a slave node from another process working with new approach for tied contact.
- Correct initialization of MPP tied contact with implicit mechanics when explicit is running dynamic relaxation followed by implicit.
- Fix a bug in assignment of contact thickness for null beams. This bug was introduced at r89658/R9.
- Add variable 2DBINR (Card F) for surface-to-surface contact involving 2D seatbelt elements initially inside a retractor. This is only available for SOFT = 2 contact.
- Fix a bug for contact involving high order shell elements that occurs when 8-node shell elements are generated by SHL4_TO_SHL8, or when using large part id like 100000001.
- Add an alternative implementation of ICOR (coefficient of restitution) for contact. The algebraic sign of ICOR determines which implementation is used.
- Command line option, soft = 1to2 converts all possible contacts from soft = 1 to soft = 2
- Command line option, soft = 2to1 converts contacts from soft = 2 to soft = 1
- Fix *CONTACT_AUTOMATIC_GENERAL for spotweld beams when using SSID = 0, i.e. all parts included in the contact, and CPARM8 = 2.
- Implement unloading curve, UNLCID, for options FCM = 2/3 in *CONTACT_RIGID_(OPTION).
- Fix zero frictional energy output to glstat and sleout when using *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_ORTHO_FRICTION.
- Implement SMP consistency mode, i.e. ncpu < 0, for:

    - *CONTACT_SURFACE_TO_SURFACE_CONTRACTION_JOINT
    - *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_COMPOSITE

- Fix seg fault when using *CONTACT_AUTOMATIC_NODES_TO_-SURFACE for SPH elements impacting on shell plate.
- Fix *DEFORMABLE_TO_RIGID switching using contact forces, CODE = 2 & 4, when the contact type is *CONTACT_FORCE_TRANSDUCER.
- Fix zero contact stiffness if *EOS_TABULATED is using curve LCC instead of tabulated input cards.
- Allow *CONTACT_TIED_SURFACE_TO_SURFACE to use MAXPAR in Optional Card A. It was previously hardcoded to 1.01.
- Implement edge treatment option SRNDE = 1,2 (Optional Card E) for *CONTACT_AUTOMATIC_NODES_TO_SURFACE.
- Fixed MPP segment based (SOFT = 2) contact when used with 2-surface force transducers. Output data may have been incorrect due to errors in accessing arrays. This error occurred when different processors participate in different contact interfaces.
- Fixed MPP implicit SOFT = 2 contact which was failing to properly reset the segment and node data from a converged state leading to possible "floating invalid" error or possible convergence failure.
- Fixed an MPP SOFT = 2 contact problem that could occur if a contact interface used SBOXID to eliminate slave segments. This could cause a segmentation fault during initialization.
- Fixed a possible memory error during initialization of SOFT = 2 contact, when using DEPTH = 5, 15, 25, or 35 or SBOPT = 4 or 5.
- Fixed an error in MPP SOFT = 2 eroding contact that could cause force spikes and inappropriate contact detection after solid elements erode.
- Fixed an error in SOFT = 2 eroding contact that caused the contact to fail to create new segments on interior elements when there were no active segments at the start of the run. This error was most likely to occur in MPP runs when a decomposition can result in a process that has no active segments at the start of the run.
- Changed the behavior of SOFT = 2 contact when the contact keyword indicates a surface to surface contact, but the master surface has no segments. In older versions, the contact would use the slave segments and do a single surface contact with them. Going forward, the contact will not do anything because the slave surface has nothing to contact.
- Fixed some contact options when using linear 3D solids or linear shell elements in implicit analysis. These element were causing a zero explicit time step, and some contacts use this explicit step in the denominator of the stiffness matrix. These include SOFT = 1, SOFT = 2, and tied, penalty-based contacts.
- Added a gap calculation to SOFT = 2 contact. The gaps are written to the binary interface force file. This is supported for both SMP and MPP. Overlaps are reported as negative gaps.
- Fixed a flaw in the support of SSF on *PART_CONTACT when using SOFT = 2 contact with DEPTH = 5. The consequence was to sometimes

choose the wrong part when looking up the scale factor so the stiffness was not predictable.

- Added variable SSFTYP on card F of *CONTACT which affects how SSF on *PART_CONTACT works when used with SOFT = 2 contact. By default, the contact chooses a master segment for each pair of segments in contact. The SSF factor is then taken from the slave segment. When SSFTYP is set to 1 (or any nonzero value), it changes the behavior so the maximum SSF value of the 2 segments in contact will be used. The contact stiffness is therefore independent of how the contact chooses master and slave segments.
- Fixed MPP *PART_CONTACT when used with SSF. It was possible for SSF to be incorrectly taken as zero, so penetration would occur.
- Improved the accuracy of *CONTACT_2D_AUTOMATIC contact. For problems where penetration depths are small, there was an accuracy issue that could cause contact forces to be set to zero in cycles when they should not have been, which could cause a loss of contact history. Because the penetrations are small, the results look about the same, but the contact pressure history is smoother.
- Fixed *CONTACT_2D_AUTOMATIC so it will correctly add new segments when elements using *MAT_081 or*MAT_082 are used for an eroding part.
- Fixed *CONTACT_2D_AUTOMATIC_TIED which could could fail if more than 1 tied contact was used in a model.
- Fixed the sliding option, ISLIDE, of *CONTACT_2D_AUTOMATIC type contacts in both SMP and MPP.
- Fixed MPP *CONTACT_2D_AUTOMATIC_SINGLE_SURFACE and *CONTACT_2D_AUTOMATIC_SURFACE_TO_SURFACE which had a problem in the bucket sort where the buckets may have been too small if segments passed from other processors were larger than the segments already in the processors. This could cause some contact to be missed. It could also cause the job to terminate with Error SOL+1274 if the passed segments are large enough such as can occur with shooting nodes.
- Added an option to offset the contact surface segments attached to 2D solid elements in *CONTACT_2D_AUTOMATIC. The option is controlled by two new parameters, SLDSOS and SLDSOM, on new optional card 5. If the card is omitted, or SLDSOS or SLDSOM are input as zero, then there is zero offset which has always been assumed for surfaces of solids. If input with a value greater than zero, then the offset is set equal to the input value. This behavior is not consistent with the 3D SLDTHK option, which uses half the input value, but it is consistent with the negative value option of SOS and SOM on card 2 which uses the input value, not half of it.
- Add option NEHIS to *USER_INTERFACE_FRICTION:

    - With NEHIS = 0 (default), special choice of element history variables is provided in subroutine usrfrc, see comments there.
    - With NEHIS > 0, plastic strain and element history variables up to NEHIS-1 are provided in original order in subroutine usrfrc.

- Add new *CONTACT_AUTOMATIC_..._TIEBREAK model OPTIONs 13 and 14, which are based on *MAT_240.
- Allow unlimited number of history variables for user-defined tiebreak, *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_SURFACE_TIEBREAK_USER.
- IGNORE = 2 on *CONTACT_..._MORTAR is set to default to avoid unnecessary bad behavior.
- If ISYM < 0 on *CONTROL_CONTACT, this will point to a node set containing all nodes on symmetry planes in the model, which will be picked up by the mortar contact and treat edge treatment accordingly.
- If MPAR1 < 0 on *CONTACT_..._MORTAR for IGNORE = 3, this will govern the penetration reduction as a function of a curve.
- If SLDTHK < 0 on *CONTACT_..._MORTAR, the contact surface is offset "inwards", previously only a positive value was supported.
- Implicit tied contacts (strong objective) supported for groupable option, which is necessary when running a coupled thermal/mechanical simulation.
- User friction supported in mortar contact, see *USER_INTERFACE_-FRICTION. The subroutine is called mortar_usrfrc and found in dyn21.F.
- FS = 2 on *CONTACT_..._MORTAR supported, allowing friction as function of sliding velocity and contact pressure in mortar contact.
- For large penetrations in mortar contact during implicit analysis, the step will be abandoned and a retried with a smaller time step.
- *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR_TIED_-WELD supported, SMP and MPP, particularly intended for implicit.
- When mortar tied contact is used with shells, offsets are accounted for to induce a moment from tangential tractions in the interface.
- In mortar contact, let PENMAX and SLDTHK take over the meanings SST and TKSLS have in R9 and earlier, although in a different way. Now PENMAX corresponds to the maximum penetration depth for solids. SLDTHK is used to offset the contact surface from the physical surface of the solid element, instead of playing with SST and TKSLS, which was rather awkward. This update also saves the pain of having to treat shells and solids in separate interfaces if these features are desired.
- A "shooting node logic" algorithm is implemented for mortar contact and is always active. This should reduce the presence of negative sliding energies when run in explicit and also improve initial penetration situations in implicit.
- The MPP groupable contacts include all eroding contacts, i.e.

  - *CONTACT_ERODING_NODES_TO_SURFACE
  - *CONTACT_ERODING_SURFACE_TO_SURFACE
  - *CONTACT_ERODING_SINGLE_SURFACE.

- The MPP groupable contacts include beam-to-beam treatment in *CONTACT_AUTOMATIC_GENERAL. It supports rectangular beams, and output of contact forces to rcforc for an accompanying 2-sided force transducer.
- The MPP *CONTACT_AUTOMATIC_SINGLE_SURFACE_TIED is implemented and supports parameter CLOSE.
- MPP's *CONTACT_AUTOMATIC_BEAMS_TO_SURFACE_ID now supports the contact of slave beam nodes from both sides of the master surface.
- Fix incorrect *INITIAL_STRESS data written to dynain in MPP when np > 1.
- *CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURFACE:
  - Improve SOFT = 6 behavior. In guide pin contact, consider the pin's curved edge, therefore overcome the false penetration problem.
- *CONTACT_FORMING_NODES_TO_SURFACE_SMOOTH:
  - Allow segment definition, in addition to part or part set ID.

- ***CONSTRAINED**
  - Fix MPP message passing error that could occur if a node involved in *CONSTRAINED_SHELL_TO_SOLID is shared between more than 2 processors.
  - *CONSTRAINED_GLOBAL: Added user-defined tolerance parameter TOL in length units. If non-zero, the internal mesh-size dependent tolerance gets replaced by this value.
  - Added option to continue a calculation with constrained interpolation after an independent node is deleted using INDSW in *CON-STRAINED_INTERPOLATION.
  - Fix to exactly singular constraint matrix for *CONSTRAINED_-INTERPOLATION running in fast/vector mode.
  - Correct MPP communication during the detection of co-linear nodes in *CONSTRAINED_INTERPOLATION.
  - Add additional error checking on coincident nodes for revolute and screw joints.
  - Promote linear algebra processing for *CONSTRAINED_INTERPOLATION in explicit single precision to real*8 to remove round-off accumulation errors.
  - A more robust solution to the processing of birth and death times for prescribed motion constraints for the LaGrange Multiplier Formulation of joints for explicit.
  - Add implicit capabilities pertaining to UNITCELL:
    - Always unconstrain the last 3 nodes in input and output if they are referenced in *CONSTRAINED_MULTIPLE_GLOBAL constraints. These are control points for UNITCELL.
    - At the end of implicit time step always go through the computation to compute the resultant forces. If there are *CONSTRAINED_-MULTI-

PLE_GLOBAL constraints, output the resultant forces on the last 3 nodes.

- Enhance nodal constraint handling for superelements using explicit to properly handle the implicit/explicit switching case. I also extended the code to recognize *BOUNDARY_SPC definitions instead of just those on the *NODE cards.
- The logic to handle *CONSTRAINED_INTERPOLATION with large number of independent nodes extended for *CONTROL_IMPLICIT_MODES.
- Implicit was enforcing birth and death times on *BOUNDARY_SPC during dynamic relaxation contrary to the manual. So such times are now ignored with dynamic relaxation.
- Added a warning about the combined use of rigid body stoppers and the lagrange multiplier formulation of joints for explicit recommending switching to the penalty formulation.
- Corrections to get implicit to work with *CONSTRAINED_-INTERPOLATION with lots of independent nodes especially in MPP.
- *CONSTRAINED_BEAM_IN_SOLID:

  - Add _PENALTY option to invoke a penalty-based formulation (implicit and explicit).
  - Implicit support for IDIR = 1 (allow sliding along axial direction).
  - Implicit support for AXFOR (user prescribed debonding force between beam and solid).
  - Thermal solver recognizes *CONSTRAINED_BEAM_IN_SOLID by constraining temperature fields between beam and solid nodes.
  - Write LSDA format output file named "cbisfor" containing debonding force (constraint-based) or penalty force (penalty-based).

- Add two keywords *CONSTRAINED_SHELL_IN_SOLID and *CONSTRAINED_SOLID_IN_SOLID, which are similar to *CONSTRAINED_-BEAM_IN_SOLID, but are used to couple shells immersed in solids and solids immersed in solids, respectively.
- Fix a bug affecting *CONSTRAINED_LOCAL whereby z-translation was mistakenly constrained when IRC = 0.
- Fix a bug for PIDCTL of *DEFINE_CURVE_FUNCTION, that occurs when using "0" as sampling rate.
- *CONSTRAINED_SPOTWELD:

  - Enable the normal and shear forces at spotweld failures to be temperature-dependent functions.

- Allow the same part to be used in SPR connections, i.e., MID and SID can now match in *CONSTRAINED_SPR2 or rather PID1 and PID2 can match in *CONSTRAINED_INTERPOLATION_SPOTWELD. The requirement for this is that the SPR node lies in between the shell elements to be self-connected.

- *CONSTRAINED_INTERPOLATION_SPOTWELD, MODEL = 1, can now be used as connection of two beam nodes.
- Several updates for *CONSTRAINED_INTERPOLATION_SPOTWELD (new card 5):
  - Incorporate torsion mode with GAMMA > 0.
  - Allow definition of separate stiffnesses STIFF, STIFF2, STIFF3, and STIFF4 for tension, shear, bending, and torsion.
  - Optional exponential damage law via LCDEXP.
  - Alternative shear kinematics treatment, parameter SROPT.
  - Output of history variables with NEIPB = 7 on *DATA-BASE_EXTENT_BINARY.
- Add GISSMO damage as failure variable for *CONSTRAINED_TIED_-NODES_FAILURE.
- Fixed bug in processing *CONSTRAINED_NODAL_RIGID_BODY and *PART_INERTIA so code does not abort with a bogus "Error 10144".
- *CONSTRAINED_COORDINATE:
  - Fix a machine dependent error when the input coordinates are far away from the part.
  - Extend to rotational constraints. The constraint will be appled to the closest node.
    When IDIR > 10, IR = IDIR-10:
      IR = 1, constrain rx
      IR = 2, constrain ry
      IR = 3, constrain rz
      IR = 4, constrain rx+ry
      IR = 5, constrain ry+rz
      IR = 6, constrain rz+rx
      IR = 7, constrain rx+ry+rz
  - Fix a segmentation fault problem when IDIR is greater than 10.
- *CONSTRAINED_RIGID_BODY_STOPPERS:
  - Extend to have multiple rigid body stoppers for one single rigid body.

- **CONTROL**
  - *CONTROL_REFINE_SOLID and *SECTION_SOLID (ELFORM = 2):
    - Support fully integrated S/R solids.
  - *CONTROL_REFINE_... (IBOX, 1st line, 5th column):

    - Implement *DEFINE_BOX_LOCAL and *DEFINE_BOX_ADAPTIVE to refine elements checking criteria provided by these box keywords.

- *CONTROL_ADAPTIVE (ADPTYP = 8, 1st line, 3rd column) and *SECTION_SHELL (ELFORM = 12):

    - Map the plain stress shell thicknesses for the 2D adaptive remeshing

- Staged construction (*CONTROL_STAGED_CONSTRUCTION, *DEFINE_-STAGED_CONSTRUCTION_PART):
- Change the behaviour when a part is added.
  Previously, the stiffness and strength of the part would ramp up gradually according to ramp time ATR on *DEFINE_CONSTRUCTION_STAGES. Now, the stiffness and strength have their full value immediately at the start of the stage when the part becomes active (STGA). The reason for the change is to prevent unrealistic deformations occurring while the part is only partially stiff. The same change applies if a part is added using STGA on *LOAD_STIFFEN_PART. There is no change to gravity loading associated with *DEFINE_STAGED_CONSTRUCTION_PART: when a part is added, the gravity load still ramps up according to ramp time ATR. Nor is there any change to the behaviour during part removal.
- Other updates to Staged Construction are:

    - Enabled _TITLE option for *DEFINE_CONSTRUCTION_STAGES and *DEFINE_STAGED_CONSTRUCTION_PART. The titles are ignored by LS-DYNA and are used only in pre-processing. Previously, adding _TITLE would have caused an input error.
    - Fixed bugs in energy balance output for staged construction (*DEFINE_STAGED_CONSTRUCTION_PART and also *LOAD_-STIFFEN_PART and *LOAD_GRAVITY_PART). The internal energy was wrongly calculated for dormant parts, and the gravity loading could sometimes be missing from the external work calculation, depending on the other contents of the model. These bugs affected the energy outputs (e.g. in glstat file) but did not affect the other results (stress, displacement, etc).
    - Fix bug in dynain file related to the format of *INITIAL_PWP_-NODAL_DATA card, which is written if the model contains *CONTROL_PORE_FLUID. The resulting dynain file could not be read into LS-DYNA.
    - *DEFINE_CONSTRUCTION_STAGES have option inputs RTS, RTE ("real time" at start and end of stage). This is intended to enable an "accelerated analysis" in which processes that really take days or weeks can be modelled in seconds of analysis time. These inputs previously had no effect, they were present only to help the user understand the

times in the input file. Now, "real times" will be written to the output files in place of analysis time.
- Fixed bug that occurred with *CONTROL_STAGED_CONSTRUCTION if Dynamic Relaxation was also switched on. Staged construction is compatible with D.R. only if the analysis begins at the first stage (i.e. time = 0). If not, D.R. now gets switched off automatically and a warning is issued.
- Enabled *LOAD_GRAVITY_PART for tshells.

- **CONTROL_ADAPTIVE**

  - Fix to avoid seg fault in R-adaptivity if 2*(number of shells) > 3*(number of nodes).
  - Fix problem with 2d adaptivity and boundary merging. Some boundary points between materials weren't merged in some cases, depending on where the program thought the 2d contours started. This only applies for *CONTROL_ADAPTIVE ADPTYP = 8 with mmm2d = 1.
  - Fix excessive memory growth during adaptive problems, and reduce memory requirements overall.
  - Generate input error message if *DEFINE_FILTER is used together with *CONTROL_ADAPTIVE, because these filters just won't work with adaptivity.
  - Increase the size of some statically allocated arrays so that larger 2d adaptive problems can be run.
  - *CONTROL_ADAPTIVE:

    - Added Card 5 with only one input variable called INMEMORY.
      EQ.0: conventional out of core shell adaptivity
      EQ.1: in-core shell adaptivity with load rebalancing.

  - *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS:

    - The VID of the rotating axis can now be defined by both *DEFINE_VECTOR and *DEFINE_VECTOR_NODES. It enables the movement of the rotating axis. Previous verion only use *DEFINE_VECTOR to define the VID.
    - The rotational dynamics can work in MPP now.

  - Added variable ICRFILE in *CONTROL_OUTPUT so that nodes sets and element sets associated with *DATABASE_CROSS_SECTION are written to a file to facilitate checking of the cross-section definition.
  - The adaptive element size range defined by RMIN/RMAX in *DEFINE_BOX_ADAPTIVE can now be out of the range defined in *CONTROL_REMESHING for 3D adaptivity.
  - *CONTROL_ADAPTIVE:

   - Add warning message that IREFLG is not supported in 3D adaptivity (ADPTYP = 7).

- *CONTROL_ADAPTIVE:

   - Add a new feature to the adaptivity of sandwich part: allow multi-layers of solid core to be refined.
   - Support the refinement of 6-node solid elements in the refinement.
   - Fix a bug in the output of 'adapt.msh': an extra blank line was output in *ELEMENT_SHELL_THICKNESS.
   - Fix a duplicate beam error with adaptivity involving beam elements.

- *CONTROL_ADAPTIVE_CURVE:

   - Fix a bug where the refinement width control was not functioning properly.

- **Discrete Element Method**

   - Switch material type from rigid to elastic for DEM.
   - Enhance MPP's particle-DEM coupling algorithm for *PARTICLE_BLAST.
   - Fix bug if de_massflow_plane and interface force file for *DEFINE_DE_TO_-SURFACE_COUPLING are both defined.
   - Fix MPP bug for *PARTICLE_BLAST if some processors contain solid elements while other processors have no solid elements.
   - Fix bug for *PARTICLE_BLAST if solids are used as geometry for HE particles.
   - Implement rebalancing algorithm for *PARTICLE_BLAST, the performance of particle-particle contact and particle-structure contact is increased by ~10~30 times for hundreds of cores----MPP only.
   - Fix bug for *DEFINE_DE_MASSFLOW_PLANE output error if multiple planes are defined.
   - Fix bug for *DEFINE_DE_INJECTION if multiple injection planes are defined.
   - Implement eroding coupling between particle and structure (shell/solid) for *PARTICLE_BLAST. New surfaces of eroded solid parts are taken into consideration.
   - Fix MPP bug when detonation point is defined using a node for *PARTICLE_BLAST.
   - Fix wear depth calculation error for DEM interface force file (MPP only).
   - Skip *PARTICLE_BLAST calculation during dynamic relaxation phase.
   - Fix MPP bug when there are multiple *PARTICLE_BLAST definitions.
   - Add calculation of coordination number for DEM in MPP.
   - Add automatic bucket sort for *DEFINE_DE_TO_SURFACE_COUPLING.
   - Reformulate particle injection algorithm for MPP such that particles with different radii can be injected.

- Add birth time and death time for *CONTROL_DISCRETE_ELEMENT.
- Fix bug in calculation of capillary force for *DEFINE_DE_-TO_SURFACE_-COUPLING.
- Fix bug for capillary force calculation if DEM model has zero potential contact pairs but non-zero capillary force.
- Implement DEM mass output to interface force file for *DEFINE_DE_TO_-SURFACE coupling.
- Add mass output for demrcf.
- Add moment output to demrcf (SMP only).
- Report damping energy for DE non-reflecting boundary conditions.
- Implement explicit thermal modeling of DES (SMP only).
- Implement user defined curve to get mass flow rate for *DEFINE_DE_INJECTION.
- Fix minor bug for *DEFINE_DE_TO_SURFACE_COUPLING when the thickness of shell is much larger than segment size.

- **EFG (Element Free Galerkin)**
  - Variables STRFLG and INTOUT in *DATABASE_EXTENT_BINARY can now be turned on when using 3D adaptivity on EFG solid.
  - Special decomposition is implemented for EFG shell formulation 41 to avoid the memory error in MPP.
  - Automatically set IPS = 0 (no pressure smoothing) for EFG solid formulation 41.
  - *MAT_076 is now supported for EFG shell formulation 44.
  - Fixed incorrect stresses in:
    - Plane strain EFG formulation (shell type 43) in both explicit and implicit,
    - Axisymmetric EFG formulation (shell type 44) in explicit.

- ***ELEMENT**
  - *INTEGRATION_BEAM with different PIDs at each integration point: this was an existing capability, but the time step calculation was overly conservative for cases where the section contained a small proportion of a stiffer material, such as the reinforcement in reinforced concrete. For the same reason, LS-DYNA could add large amounts of mass-scaling unnecessarily. The time step calculation has been improved to remedy this.
  - Fixed bug for beam elements ELFORM = 2: with certain combination of inputs only, the output forces and moments could be wrongly rotated about the beam axis. This affected the output forces only, not the solution inside LS-DYNA. The error could take two forms: (a) if IST on *SECTION_BEAM is non-zero, the output forces and moments are supposed to be rotated into the beam's principal axis system, but this rotation could be applied to the

wrong beam elements; and (b) when no ELFORM = 2 elements have IST, but the model also contains beams with ELFORM = 6 and RRCON = 1 on the SECTION_BEAM card, some of the ELFORM = 2 elements could have their output forces and moments rotated by one radian. These bugs are now fixed.

- Speed up keyword for some models that use lots of *ELE-MENT_SOLID_-TET4TOTET10 due to too many memory allocation calls.
- *SECTION_BEAM: Added a flag ITORM for improved representation of torsional modes which can be activated (ITORM = 1) only if an eigenvalue analysis is performed and which applies only to beam type 13. If ITORM is not active (ITORM = 0) the torsional inertia from structural analysis is used which may result in too large eigenvalues related to torsional modes.
- Added cubic solid element formulations 27,28,29 (*SECTION_SOLID).
- Added section forces for higher order solid elements.
- Added element deletion capability for higher order elements.
- Correct issue with prescribed motion on superelements for explicit mechanics.
- Enhance the application of damping to superelements for explicit.
- Tune up output of consistent elemental mass matrices. Needed to capture the lumped mass terms that had been removed. This code changes output of the elemental mass matrices to include the lumped terms.
- Enhance *ELEMENT_DIRECT_MATRIX_INPUT where the matrices for the same superelement do not match in order.
- SEATBELT:
  - Fix MPP bug that could occur where there is more than one type-9 seatbelt pretensioner.
  - Fix misprinted section number and material number for 2D belt error message.
  - Limit the bending stiffness of *MAT_SEATBELT to implicit analysis only. Warning will be issued when non-zero bending stiffness is defined for explicit analysis.
  - Fix a MPP bug whereby incorrect belt material length information is output to d3hsp.
  - Fix a bug that occurs when time-dependent slipring friction is used for 2D belts.
  - Fix erroneous 1d seatbelt slipring message that has been there since version R8.0.0.
  - Enhance *DATABASE_RECOVER_NODE so that it works for shell form 23 with 3x3 integration.
- *USER_NONLOCAL_SEARCH:
  - Provides an interface for gathering the history data of specified elements that surround a “master” element to average (or smooth) the

history data of the master element. The surrounding elements are determined using a user defined strategy. The type of averaging is also user specified. Currently this keyword only works for solid elements.
- supports solid form 1, 16, 19, 21
- supports _ORTHO to search in material local abc directions
- enables user to define a nonlocal search in ellipsoid shape

- Added cohesive element formulation -29 that defines the cohesive midlayer from the average of surrounding shell normals. This fomulation is better suited for simulating normal shear.
- Fixed absence of part mass in d3hsp for cohesive shell element formulation 29.
- Fix memory error when using *NODE_THICKNESS_SET and also fix node set not found error when using *NODE_THICKNESS_SET with *SET_-NODE_INTERSECT.
- Make *TERMINATION_DELETED_SOLIDS work with hex spot weld failures.
- Fix seg fault when using shell type 15, axisymmeteric volume weighted, with *MAT_EROSION and also materials with equation-of-states.
- Fixed implicit accuracy when using higher order shell form 23. The element was
  forced to use 3x3 integration but now supports both 2x2 and 3x3.
- Modified discrete beam form 6 when it is defined with rotational stiffness, but no rotational damping. This change adds a small amount of damping to prevent beams from excessive oscillations which can cause error in the element strain measure.
- Fixed the NREFUP option (*CONTROL_OUTPUT) when the ORIENTATION option of *ELEMENT_BEAM is not active and some other options are not active. The NREFUP option is available for beam types 1, 2, 11, 12, and 14.
- Fixed the thinning of shell form 24 which was thinning about 33% more than it should under large tensile strain.
- Enabled adaptive tet meshing (ADPTYP = 7 on *CONTROL_ADAPTIVE) to work for any reasonable solid element form at the start of the calculation. Previously, the part could only be tet meshed with form 13 elements, or else EFG solid elements 41 or 42, or the job would crash at the first remap step. Now it should work with any structural element form.
- Improved the accuracy of tetrahedral solid form 13. During rigid body translation, some pressure could develop. Now it can translate pressure free, and therefore stress free.
- Fixed beam elements when some elements in a part use *ELE-MENT_-BEAM_ORIENTATION, and some do not. The affected beam forms are 1, 2, 11, and 14.
- Modified the behavior of isotropic materials that are used in composites that are modeled with tshell form 5. The oscillation check on the thickness stress

was modified to directly check stress rather than a total strain. For nonlinear materials, this is more reliable as it was possible to miss the oscillations when checking strains. This should have little effect on most solutions.

- Fixed tshell forms 5 and 7 when used with total Lagrangian material models such as *MAT_002. Large rotations were leading to incorrect strains and stresses.
- Fixed tshell form 3 when used with the hyperelastic materials. The L matrix was in the global coordinate system instead of the local system.
- Changed the d3hsp output of 3-node triangular shell elements so that zero thickness is reported for node 4. The inputted node 4 value was being output and included in the element thickness calculation, but now it is ignored.
- Fixed output of membrane shell form 5 when used with materials other than 34 and 134. The stress and history data written to the d3plot file was wrong.
- Fixed the behavior of CDL and TDL on *SECTION_DISCRETE when one of the nodes is constrained by an SPC. Also, corrected the deforc output when CDL and TDL limits are reached.
- Fixed the strain output in eloutdet for higher order shell form 23.
- Improvement of error checking of cohesive elements (solid forms 19/20/21/22). In addition to checking for adjacent elements (shells, tshells or solids), we now check for being part of a tied contact definition. This eliminates a lot of unnecessary warning messages.
- Fixed adaptivity (and restart) for tetrahedral solid formulation 13 by porting nodal averaged data between adaptive steps. Before this fix, spurious peaks in cross-section forces could occur and be mistaken for contact pressure peaks.
- Add new option for *PART_STACKED_ELEMENTS to deal with in-plane composed reference mesh parts. Must be used together with *NODE_-MERGE_-SET.
- Add cross section forces (*DATABASE_SECFORC) for 20-node and 27-node hexas, i.e., solid forms 23 and 24.
- Add variable ITOFF in *SECTION_BEAM to control torsion behavior for spotweld beams (beam form 9).
- Improve time step estimate for 10-noded tet form 16 with "curved" edges to improve robustness.
- Add variable ICRQ = 2 on *CONTROL_SHELL to only treat thickness continuously across element edges. This is an alternative to ICRQ = 1 which treats both thickness and plastic strain continuously across element edges.
- For higher order shells, the body loads account for shape functions when assembling nodal forces.
- The kinematics for warped/hourglassed cohesive solid forms 20/22 are now corrected so that rigid body motion should give zero eigenvalues.
- If IACC = 1 on *CONTROL_ACCURACY, 6-node quadratic shell form 24 is treated
- According to Martin & Breiner for alleviating membrane and shear locking.

  - If IBEAM on *CONTROL_IMPLICIT_EIGENVALUE is set to 2 or 13, then beam formulations 1, 4 and 5 undergo section conversion from an integration rule to resultant properties.
  - Fixed error for reading *ELEMENT_SHELL_OFFSET_COMPOSITE.
  - Fix parsing errors in *SECTION_BEAM_AISC.
  - Fix error in reading of user-defined shell elements.
  - A new 4-node tetrahedral solid element (ELFORM = 60) is implemented to support mixed materials. The volumetric locking is eliminated without nodal averaging.

- **EM (Electromagnetic Solver)**

  - EM Analysis developments are divided into several subcategories below.
  - Batteries

    - Added 2 extra variables in d3plot in EM_FEMSTER_NODE for batteries:
      D3PL_RAND_areaCircuit_EM
      D3PL_RAND_areaCell_EM
    - For local area of each Randle circuit and global area of the cell.
    - Added meshless Randle model, see *EM_RANDLES_MESHLESS.
    - Fixed bugs for addition of joule heat rate in meshless Randle.
    - Added battery model with composite tshells.
    - Switched from *EM_CIRCUIT_RANDLE to *EM_BATTERY_-RANDLES for solids and *EM_RANDLE_LAYERED for composite tshells.
    - In *EM_BATTERY_RANDLES, *EM_RANDLE_LAYERED and *EM_-RANDLE_SHORT, Randle Area = 2 is now the default, and the old Randle Area = 0 ("as is") is now Randle Area = 3.
    - Added cylindrical cells.
    - Fix bug in build of the layered (composite tshells) circuits when layers end up by a separator (like in cylindrical cells).
    - Addition of RANDLES instead of RANDLE in *EM_RANDLE_... keywords to be consistent with the manual (the old RANDLE still works).
    - Fix bug in building layer mesh for addition of different composite tshell parts.
    - Addition of Randle Area in randles circuit for composite tshells.
    - Addition of Randles circuit in connection with new LS-PrePost battery packaging for solid elements, in serial and MPP.
    - Addition of optional joule heating from a meshless Randle circuit to a set of parts (uniformly). This is triggered using *EM_ISO-POTENTIAL_CONNECT.

  - II. Electrophysiology

   - Added Monodomain (EPMD) and bidomain (EPBD) solvers for electrophysiology.
   - Addition of bath loading in EP models, both in EPBD, and in EPMD, using an augmented monodomain approach. This should work both in serial and MPP.
   - Added calcium concentration at nodes vector in EP so that it can be used in mechanical models. It can be visualized using d3plot in relative permeability for now.
   - Added PCG in EP: the user can choose between MF2 and PCG using EM_SOLVER_FEM.
   - Added tetrahedrons in EPMD and EPBD.
   - Addition of *EM_EP_TENTUSSCHER_STIMULUS2 to create stimulus on a node set where the amplitude is time dependent given by a load curve (amplitude vs time). Several such stimuli can be created at the same time.
   - Addition of user cards for *EM_EP_TENTUSSCHER and *EM_EP_TENTUSSCHER_STIMULUS for user to input parameters for electrophysiology.
   - Addition of beta and Cm in *EM_MAT_003 and *EM_MAT_005.
   - Added Godunov method using MF2 in MPP for EP bidomain.
   - Addition of *EM_MAT_005 with 2 conductivity tensors for EP bidomain model.
   - Addition of implicit and first order operator split (+ combination of the 2) in monodomain method.
   - Added activation time as a d3plot output as well as an ASCII file (x,y,z,time) at each node. In d3plot, the activation time is at the ohm heating power for now.
   - Added transmembrane potential in d3plot output: it is in the scalar potential for now.

- III. Resistive Spot Welding

   - Addition of resistive spot welding in 2D (rsw2d).
   - Addition of contact Joule Heat Rate for contact resistance in rsw2d.
   - Addition of resistive heating solver in 2D, for rsw in 2D.
   - Added zero out of em_nodeJHrate in em_zeroEMFieldsOut, so that no more JHR from the contact resistance is added after EM is switched off.

- IV. Eddy current

   - Added option to use *DEFINE_FUNCTION in LCID for the imposed scalar potential circuit type in EM_CIRCUIT. Allows users to use their own circuit equation as input.

- V. Inductive heating

    - Added option to define NUMLS, F and A with a Load Curve function of the macro time if a negative integer value is entered.

  - VI. Miscellaneous EM

    - Creation of d3p_bemDecomp file if gmv > 0 in *EM_OUTPUT with BEM face domains.
    - Corrected an MPP issue for the EM solver that could occur in problems where some solid and shell elements have the same element number.

- **Forming Analysis**

  A special form of shell h-adaptivity called "tube adaptivity" can now be invoked using *DEFINE_BOX_NODES_ADAPTIVE. Here, fission and fusion occurs in shells located inside a "tube", that is, a torus-shaped volume, based on the path of a moving tool. This form of adaptivity can help reduce simulation time for incremental forming or roller hemming.

  - *ELEMENT_BLANKING:

    - Fix corner trimming problem of a flat blank, where the corner elements would not follow the trim line when trimmed.

  - *CONTROL_FORMING_ONESTEP:

    - Improve support to-be-unfolded part as a dynain file, by accepting *ELEMENT_SHELL_THICKNESS, *INITIAL_STRESS_SHELL, and *INITIAL_STRAIN_SHELL.
    - Add *MAT_123 to onestep method.
    - Fix friction force calculation error.
    - Improvements: some of the executables were not able to use multiple CPU in SMP before, now it is possible.

  - *CONTROL_FORMING_TRIMMING, *DEFINE_CURVE_TRIM:

    - Fix bug in trimming: some of the history variables were lost during trimming.
    - Fix bug: some SPC nodes were mistakenly removed.
    - Fix bug in trimming of sandwiched part with multi-layerd core: shells were wrongly created for every layer of the sandwiched core.
    - Improve trimming of solid elements: allow the corner to be trimmed exactly as the trim curve.

  - *CONTROL_FORMING_TRIMMING, *DEFINE_CURVE_TRIM_3D:

    - Improve trimming with _3D option: if a vector id is provided (by mistake) with 3D trimming, the code will set the vector to be zero, as 3D trimming does not require a vector.

- *DEFINE_CURVE_TRIM_2D:
  - Change from *DEFINE_CURVE_TRIM_NEW to *DEFINE_CURVE_-TRIM_2D.
- *CONTROL_FORMING_REMOVE_ADAPTIVE_CONSTRAINTS:
  - Fix segmentation fault.
  - Extend to triangular elements, in addition to quadrilateral elements.
- *CONTROL_IMPLICIT_FORMING:
  - For IOPTION = 2, fix a truncation error that prevented completion of the analysis.
  - Fix a bug that prevented application in dynamic implicit analysis.
  - Fix kinetic energy calculation error.
- *DEFINE_FORMING_CLAMP:
  - Add error message: "Vector: xx was not defined" in case vector's direction is defined incorrectly.
  - Check constraints for the rigid clamper and free the constraints from the moving clampers.
- *DEFINE_PART_FROM_LAYER:
  - Add *CONTACT_SURFACE_TO_SURFACE between layers generated by this keyword.
- *CONTROL_FORMING_AUTOPOSITION_PARAMETER:
  - If separation distance cannot be found, for example, when the MPID is not found when calculating the separation distance, or out of position, or the DIR is not input correctly, instead of returning a very large number, the value of PREMOVE will be returned.
  - Support *DEFINE_COORDINATE_VECTOR.
- *CONTROL_FORMING_OUTPUT:
  - Enable the variable NOUT not only in punch drawing but also in binder closing.
  - Allow output of d3plot according to NOUT without using y1,y2,... or a curve id.
  - Fix a bug to prevent excessive intfor output.
- *DEFINE_FORMING_BLANKMESH:

    - Fix X, Y shifting problem (not working) for NPLANE = 2 and 3.

- *INCLUDE_AUTO_OFFSET:

    - Extend to beam and solid elements.
    - Fix a beam offset problem when adaptivity is turned on.
    - Add a new option _USER. User can now control how much the node and shell element are offset. In addition, with this option, the offset can be used for sheet blank part. Without this option, the auto offset can only be used for rigid bodies.
    - Extend _USER option to include solid and beam parts.

- *CONTROL_FORMING_TIPPING:

    - Improve this keyword to allow three source coordinates and the corresponding three target coordinates for the tipping. Use NMOVE = -6 to activate this feature.

- *CONTROL_FORMING_UNFLANGING:

    - Fix a segmentation fault when writing the result file 'unflanginfo.out'.

- *DEFINE_MULTI_DRAWBEADS_IGES:

    - Fix a duplicate node set issue when SMOOTH contact is used with automatic draw beads generation from IGES.

- *CONTROL_FORMING_PRE_BENDING:

    - Add a warning message that the keyword must be placed at the end of the input file.

- *DEFINE_BOX_DRAWBEAD:

    - Fix a bug in calculating the box size.

- *CONTROL_FORMING_STONING:

    - Allow the element set to be defined with *SET_SHELL_GENERAL and *DEFINE_BOX.

- *DEFINE_FIBERS, and *CONTROL_FORMING_ONESTEP:

    - Add a new keyword (*DEFINE_FIBERS) to define carbon fibers and their related properties for one-step inverse forming simulations of carbon fiber reinforced composites.
    - Output initial flat part, which is re-orientated by aligning node N1 and N2 in x-direction.

    - Output element fiber information, including orientation and width.
    - When N1 or N2 is zero or undefined, another line of input is required for the target coordinates for N1 and N2, and the code will find the nearest nodes.
    - Outputs:
      1st history variable: the angle between two fibers.
      2nd history variable: the angle between the first fiber with respect to the element direction.
      3rd history variable: the angle between the second fiber with respect to the element direction.
    - Up to three major fiber reorientations are allowed.

- *CONTROL_FORMING_STRAIN_RATIO_SMOOTH:

    - Change keyword name from *CONTROL_FORMING_TOLERANCE to *CONTROL_FORMING_STRAIN_RATIO_SMOOTH.

- *CONTROL_FORMING_AUTOCHECK:

    - Add a new variable IFSHARP. When IFSHARP = 0, check the sharp edge and delete the elements; when IFSHARP = 1, ignore sharp edge.
    - Check for and fix for triangular elements incorrectly defined with four unique nodes.

- *INTERFACE_COMPENSATION_FLANGE:

    - Added this new keyword to handle flanging die compensation.

- *DEFINE_PART_FROM_LAYER:

    - Add a new keyword to generate multi-layers of shells for a composite structure (carbon fiber material, for example).

- *ELEMENT_LANCING:

    - Add lancing of multi-layered solid elements.

- *CONTROL_FORMING_BESTFIT:

    - Add some bestfit statistics in the "messag" file after best fit. The first column is the percentage of nodes within the range indicated; the second column is the percentage of nodes within the upper limit of the range indicated.
    - Output the maximum gap.

- *DEFINE_CURVE_FLD_FROM_TRIAXIAL_LIMIT, and *DEFINE_CURVE_TRIAXIAL_LIMIT_FROM_FLD:

  - Added these two new keywords.

- *INCLUDE_COMPENSATION_TRIM_NODE:

  - Add a new option which will include a file containing all the nodes near the trimming line. This file should be generated from *INTERFACE_COMPENSATION_NEW_REFINE_RIGID (bndnd0.tmp).

- *INTERFACE_COMPENSATION_3D:

  - Add a new option _REFINE_RIGID to automatically identify the nodes near the trimming curves, so surface compensation later will be smoother.
  - Add a new variable TANGENT. TANGENT = 1 maintains the boundary tangency from the addendum to the binder.
  - Combine rigid refinement along trimming curve into the main compensation in one single run.
  - Change keyword *INTERFACE_COMPENSATION_NEW to *INTERFACE_COMPENSATION_3D.

- ***FREQUENCY_DOMAIN**

  - Added logic for some frequency response computations so that the amount of drilling rotation control is the same as for eigenvalue computations.
  - *DATABASE_FREQUENCY_ASCII_OPTION:

    - Added modal contribution fraction output to NODOUT_SSD and ELOUT_SSD.
    - Added option NODFOR_SSD as an option to *DATABASE_FREQUENCY_ASCII.
    - Added options NODOUT_PSD and ELOUT_PSD for random vibration analysis.

  - *DATABASE_FREQUENCY_BINARY_OPTION:

    - Added D3ZCF binary database to fringe plot zero-crossing frequencies (with positive slope) in random vibration analyis.
    - Added D3ACC binary database to fringe plot acoustic pressure contribution from boundary elements in BEM acoustic computation.

  - *FATIGUE:

    - Implemented this keyword to run time domain fatigue analysis based on stress or strain.
    - Added new mean stress correction methods: Goodman-tension-only and Gerber-tension-only to provide conservative analysis for compression mean stress.

- Added an option EN to *MAT_ADD_FATIGUE to define material's EN curve.
- Improvement to skip fatigue computation if the local strain is less than 1.e-6.
- Improved strain based fatigue analysis when using Maximum Shear Strain and using Signed Von Mises strain.
- Added restart option RESTRT to fatigue analysis, if the stress/strain time history has been precomputed.

◦ *FREQUENCY_DOMAIN_ACOUSTIC_BEM:

- Added half space option to dual BEM based on Burton-Miller formulation.
- Implemented incident wave to Kirchhoff method.
- Implemented incident wave to Burton-Miller BEM.
- Implemented incident wave to Rayleigh method.
- Fixed bug in running acoustic analysis with multiple boundary conditions in MPP.
- Enabled running BEM restart (ibemrest = 6) based on atv matrix computed previously. SMP only.
- Generating D3ACS database for collocation bem (method = 3) and dual collocation bem (method = 4) only.

◦ *FREQUENCY_DOMAIN_PATH:

- Added option _NOJOBID, so that users can run restart based on the same eigenvector database for each CASE (otherwise, LS-DYNA will add different prefix to the file name in each CASE automatically).

◦ *FREQUENCY_DOMAIN_RANDOM_VIBRATION:

- Added the following new load types for random vibration analysis:
  VAFLAG = 9 base velocity
  10 base displacement
  11 enforced acceleration by large mass method
  12 enforced velocity by large mass method
  13 enforced displacement by large mass method
- Fixed a bug in running PSD interpolation when log-log interpolation is used and the PSD includes both magnitude and phase delay (for cross PSD).
- Implemented Lalanne method for frequency domain fatigue analysis.
- Added option LCTYP2 to define phase difference in cross psd by degrees and radians.

◦ *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM:

- Changed the rule of dumping nodal displacement results to D3SPCM. Now the state variable of nodal displacement results is the displacement peak value itself, without adding original nodal coordinates. LS-PrePost has been updated to accomodate this change.
- Added xyplot file spectrum_curve_print. This file saves the intermediate base acceleration spectrum, converted from base motion time history (LCTYP = 10, 11, 12).
- Added the following new load types for response spectrum analysis:
  LCTYP = 5 (base velocity vs natural period)
  6 (base acceleration vs natural period)
  7 (base displacement vs natural period)
  8 (nodal force vs natural period)
  9 (pressure vs natural period)
  10 (base velocity time history)
  11 (base acceleration time history)
  12 base displacement time history)
- For group force computation, now we calculate the group force for each mode first and then run the mode combination on them. Previously we calculate the group force as the sum of the nodal force, after the mode combination.
- Added Von Mises stress output for beams for response spectrum analysis.
- Updated response spectrum analysis so that it can work with intermittent eigenvalue analysis.
- Added nodal force and group force output (NODFOR_SPCM) for response spectrum analysis.

◦ *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM_DDAM:

- Implemented DDAM for navy ship shock response analysis.
- Implemented CSM (Closely Spaced Modes) treatment for DDAM.
- Added mcomb = -14 to run DDAM with user defined CSM.
- Added a new parameter EFFMASS, to define a required minimum percentage for total modal mass, to decide the modes to be used in DDAM analysis. The default value is 80 (80%).
- Added a parameter UNIT, to define the unit system in the input deck, as the NAVSEA constants are only valid with the BIN unit system.

◦ *FREQUENCY_DOMAIN_SSD:

- Implemented mean stress correction for SSD fatigue analysis.
- Implemented option restmd = 2 to restart with old scratch file modeshp to save time.
- Implemented an option _FRF to *FREQUENCY_DOMAIN_SSD, to provide FRF results in SSD.

- Added option _MODAL_CONTRIBUTION to output modal contribution fraction for nodes and elements.
- Enabled combination of modal damping and local damping in SSD computation.
- Added the following new load types with rotational degree-of-freedom.
  VAD = 9 (base angular velocity)
  10 (base angular acceleration)
  11 (base angular displacement)

- **ICFD (Incompressible Fluid Solver)**

  - New ICFD features and major modifications:

    - Major rework of the boundary layer mesh generation capabilities. See MESH_BL keyword.
    - Major rework on the wave generation capabilities. Added 2D and 3D solitary waves as well as a irregular wave model (JONSWAP spectra). See *ICFD_BOUNDARY_FSWAVE keyword.
    - Added an option so that pmin and pmax in MESH_SIZE_SHAPE can be defined using *ICFD_DEFINE_POINT. The big advantage is that *ICFD_DEFINE_POINT can move and therefore *MESH_-SIZE_-SHAPE as well. Also added new Shape name: SPOL as well as birth and death times.
    - Added keyword to define a volumetric heat source. See *ICFD_-DEFINE_HEATSOURCE.
    - Added keyword which allows the user to define an initial plane for level set rather than building the initial interface mesh. See *ICFD_-INITIAL_LEVELSET.
    - Added keyword to control gap size in embed shell cases. See *ICFD_-CONTROL_EMBEDSHELL.

  - Small feature additions in ICFD and modifications to existing keywords:

    - Output frequency of d3plot in steady state in now controlled by sixth flag of *ICFD_CONTROL_OUTPUT.
    - Added output frequency for *ICFD_DATABASE_TEMP.
    - *ICFD_DEFINE_POINT : can now be made to follow a surface part's displacements.
    - Allowing *DEFINE_FUNCTION to be used for R and LCID in *ICFD_-DEFINE_NONINERTIAL.
    - *ICFD_CONTROL_GENERAL : potential flow solver can now be used in transient analysis (which can for example be useful in cases like conjugate heat transfer).

    - *ICFD_MAT : added the option to scale the surface tension by using a load curve or a *DEFINE_FUNCTION.

  - Bug fixes in ICFD:

    - Fix temperature SUPG stabilizing parameter for Conjugate Heat Transfer problem.
    - Fix the viscosity as a function of temperature when using Non-Newtonian fluids with NNID = 6,7,8 in Free-Surface problems.
    - Fix the surface tension force term.
    - Fix issue in multiphase algorithm. Stability and accuracy greatly improved.
    - Improved stability of k-epsilon model in steady state solver.

  - Minor ICFD improvements:

    - Change the nodal assembly by an element integration and assembly in anisotropic porous media solver. This way, forces at porous media interfaces are better described for coarse meshes and thin porous media domains.
    - Split the mesh statistics in icfd_mstats.xxx.dat into bulk mesh and boundary layer.
    - Added 'wetness' variable for free surface cases that shows how much a surface has touched the water.
    - Support of icfd_timeiter.dat for the steady state solver.
    - Added timer for potential flow solver.
    - Added a spatial smoothing for the surface shear stress calculation.
    - Added ICFD endtime in d3hsp initial keyword reading.
    - Accelerated heat transfer calculation.
    - Added a small warning that ICFD does not scale in SMP with ncpu higher than 1.
    - Change in unreference node detection criteria. Before, it was stopping with an error message, now it proceeds with a warning.
    - Added warning when porous media model or non newt model not detected.
    - Added avg pre and avg flux to ICFD_DATABASE_FLUX output.

- **Implicit (Mechanical) Solver**

  - *CONTROL_IMPLICIT_MODAL_DYNAMIC:

    - Support output to elout for modal dynamic analysis.
    - Make performance enhancements for transient modal analysis including the implicit Newmark scheme for time integration and a node set for loads in *CONTROL_IMPLICIT_MODAL_DYNAMIC.

- When using a direct solver for implicit (LSOLVR = 2 or 6 in *CONTROL_-IMPLICIT_SOLVER), the system of linear equations is reordered (permuted) to reduce the solution cost. Up to and including version R10, two options were available: MMD (Multiple Minimum Degree) and METIS (external package from University of Minnesota). These two options are serial algorithms, and they might run out of memory or consume a large fraction of total run time for very large models. As of version R11, a parallel, distributed-memory algorithm, LS-GPart, is introduced. It scales in both memory and time and should only be attempted for very large MPP implicit models and with the guidance of implicit support staff at LSTC (support@lstc.com). LS-GPart can be used by setting variable ORDER to 4 in *CONTROL_IMPLICIT_SOLVER). A new keyword *CON-TROL_IMPLICIT_ORDERING was also introduced to fine-tune ordering options.
- Correct a number of minor issues with memory access for implicit.
- Enhance implicit handling of nodal inertia for special cases.
- Fix divide by zero in power iteration for buckling problems with inertia relief.
- Enhancements for matrix dumping left out the special case of matrix dumping for intermittent eigenvalue problems which is now corrected.
- Adjust output to d3hsp for implicit linear equation solver options to match keyword manual.
- Correct the MPP implementation for the new stiffness control on implicit rotational dynamics.
- Adjust shift logic for lanczos eigensolver for a special case.
- Enhance implicit treatment of sense switch sw1 to avoid issues in SMP.
- Apply correction for a problem that computed way too many eigenmodes in one iteration of the MPP eigensolver.
- Add the new feature for the specification of stiffness types for *CONTROL_-IMPLICIT_ROTATIONAL_DYNAMICS.
- Lower the implicit dynamic memory greed factor from 0.85 to 0.80.
- Put end-of-file tag on d3eigv after writing stresses for MCMS eigensolver. Normalize MCMS computed eigenvectors to have unit norm.
- Correct the dumping of matrices from implicit when MTXDMP > 1.
- Correct a misalignment of statements in flxinit causing SMP to fail with *PART_MODES and the use of the PARTM feature from *CONTROL_-RIGID.
- Implement handling of failure of *CONSTRAINED_TIEBREAK for implicit. Required saving of the failure flag to use when constraint matrix structure has to be the same as the last analyze phase.
- Fix long standing potential memory clobber for single precision SMP implicit. (Nevertheless, single precision is not recommended for implicit.)
- Improve implicit logic for determining which DOFs are active for 2D problems in MPP.
- Enhance the modal stress scaling to be more responsive to model features to make the computation more robust.

- Enhance implicit key point logic during dynamic relaxation phase. We were using the incorrect end time which led to a zero time step.
- Add error test for when dynamic relaxation is using implicit (idrflg = 5 or 6) but DRTERM is not specified.
- Correct the MPP implementation of *PART_MODES so that the part can now be distributed across processes.
- Enable implicit to collect resultant forces for SPC constraints in local coordinates.
- Fix *CONTROL_IMPLICIT_MODES to correctly build superelements in MPP.
- For implicit MPP, correct tagging of the end of each d3eigv* file.
- Reduce memory requirements for MPP Lanczos.
- Enhance implicit to recognize superelement mass in the mass matrix multiplication.
- Enhance *CONTROL_IMPLICIT_MODES to be able to create a superelement for a model that was already using a superelement.
- Enhance the specification of IMFLAG < 0 for *CONTROL_IMPLICIT_GENERAL. The old approach only allowed toggling between 0 and 1 using a curve. This was extended to allow ordinate values of 0, 1, 4, or 5 in the curve controlling implicit/explicit switching.
- Enhance processing of rotational inertias for implicit, especially discrete elements and rigid body inertias.
- Add additional implicit debugging by checking the ends of beam elements and what they are or are not connected to.
- Reset more arrays to get the nonlinear elements to work correctly for implicit linear multi-step (NSOLVR = -1).
- If two independent nodes for *CONSTRAINED_INTERPOLATION have the same coordinates then a divide by zero could happen. That has been corrected.
- Add INTERFACE_SPRINGBACK to the cases where implicit collects and processes damping terms instead of suppressing them as is the case for implicit statics.
- Update implicit's collection of damping terms for discrete elements to account for user specified coordinate system and the case of no second node.
- Patch up reporting of bad pivots in MPP.
- For MPP, implement the checking for rigid body node replication, which was already in SMP. Also added proper error termination for analyzing the constraint matrix in MPP.
- Correct dynamic memory allocation for the implicit case for cases with *CONSTRAINED_INTERPOLATION with a large number of independent nodes but no inertia relief.
- Adjust the output value for IMFLAG to be the user specified value instead of the internally adjusted value. Also make sure the user cannot input IMFLAG = 3 which is an internal value that should not be used as input.
- Enhance linear implicit for prescribed motion data on deformable nodes.

- Correct reading of constraint mode aux vectors used by modal dynamics to impose prescribed motion constraints.
- Change typing of integer back to integer*4 to get proper output to binary format of superelement file generated by *CONTROL_IMPLICIT_STATIC_-CON-DENSATION.
- Finish migration of implicit eigensolver using Power Method for Inertia Relief + Buckling to dynamic memory.
- Enhance the separable component report to include rigid bodies.
- Enhance Implicit Usage Alert message so that for SMP it only outputs the recommended value for memory and not the memory2 setting.
- Correct the memory management for the stiffness matrix in SMP as it was freed too early when using implicit modes.
- Made a number of enhancements and corrections for the feature to apply boundary prescribed motion constraints during implicit modal dynamics.
- Added a user supplied linear equation solver capability for implicit mechanics.
- Extend changes for implicit nodal rotational inertia scaling to include *CONTROL_IMPLICIT_MODES.
- Account for rotational nodal inertia terms in implicit computations for discrete nodal inertia matrices.
- Remove double booking of rotational nodal inertia and discrete inertias that have had the rotational nodal inertia added. This will change the eigenvalues computed by implicit by making them more correct.
- Adjust implicit's treatment of inertia's so that implicit intermittent eigenvalue computations will match an implicit non-intermittent eigenvalue computation.
- Add control for the accuracy output to eigout.
- Apply patches to implicit mechanics from the development code as part of our work on solving very large implicit problems.
- Fix a deep and long hidden bug in implicit inertia relief.
- Correct underallocation of storage for inertia relief workspace for problems with more than 6 rigid body modes.
- Convert all of the implicit linear algebra to F95 dynamic memory. Added additional tracking statistics on the use of dynamic memory used by implicit. Enhance output with an Implicit Usage Alert to tell users how to set memory and memory2 for this model.
- Fix the marking of the end of the d3mode file when writing stresses.
- Fix a problem marking the end of the d3eigv file when modal stresses are written.
- Add detection of bad beam input for implicit.
- Enhance keypoint logic to enforce keypoints at the initial time step and on restart from explicit.
- Fix a problem with the timing for when "dead" nodes are incorporated into implicit.
- Added features to allow tighter coupling between implicit and USA.

- Correct output of eigenvalues and frequencies for the nonsymmetric eigenvalue problem to match that of rotational dynamics.
- Improve error message for reading d3eigv file for *PART_MODES for the case when the user input a d3eigv file from a different model than intended.
- Enhance implicit inertia relief to be optional for any explicit phase.
- Added MCMS (LSTC's name for AMLS) approximate eigensolver.
- Corrected the computation of reduced mass matrix for *PART_MODES when the nodes for the part are distributed in MPP. Also corrected the computation of total kinetic energy for the same situation.
- Earlier enhancements to adjust for poorly scaled implicit mechanical problems negatively affected the direct linear equation solver used in the implicit treatment of joints for explicit in single precision. The pivot tolerance was inappropriately being reduced. Now fixed.
- Migration of shell arrays to dynamic storage broke d3iter. Also had to reset i/o address for d3iter on restart.
- When implicit springback was following an explicit transient step the implicit keywords with the _SPR were not properly handled. Now corrected.
- Added the feature of resetting implicit geometry at the start of each time step to enable implicit linear multiple load analysis.
- Fix an implicit problem where a linear implicit analysis follows inertia relief computation.
- Enabled tshells to use the consistent mass matrix option of *CONTROL_IMPLICIT_CONSISTENT_MASS.
- Sense switch sw4 is now supported in implicit, in both SMP and MPP.
- If IMASS = 0 on *CONTROL_IMPLICIT_DYNAMICS, i.e., the analysis is static, the kinetic energy is output as zero to make physical sense.
- If HGEN = 1 on *CONTROL_ENERGY, energy from drilling constraint is included in hourglass energy and thus accounted for in energy balance.
- A nonzero start time of initial velocities is supported in implicit dynamics, see *INITIAL_VELOCITY_GENERATION_START_TIME.
- If IRATE = -1 on *CONTROL_IMPLICIT_DYNAMICS, rate effects are active even in implicit statics, which is sensible if viscoelastic effects are used in quasi-static analyses.
- Rotational prescribed motion on rigid bodies induce a fictitious residual force to avoid initial zero loads, which is a way to prevent stalling of the analysis. Translational prescribed motion was supported in this way in prior versions.
- Four variables (GJADSTF/GJADVSC/TJADSTF/TJADVSC) in *CONTROL_-RIGID automatically add joint stiffnesses to all joints, primarily for use in implicit in models with many joints (e.g., dummies) to stabilize the overall behavior.
- NLNORM = 4 on *CONTROL_IMPLICIT_SOLUTION mixes rotational and translational degrees of freedom for computing residual forces by weighing the rotational contribution with a length scale internally calculated to avoid

a units problem. The length scale used is reported in the output and can be overwritten by NLNORM < 0.

- Implemented a chained *CASE treatment for implicit, for splitting a "complicated" process into several "simple" simulations allowing for transfer of state between each such simulation. This makes use of writing and reading dynain.lsda (*INTERFACE_SPRINGBACK_LSDYNA/*INCLUDE) between cases, and in addition to element stresses, etc. that are the common state constituents, we support

  - mortar contact friction, tied, tiebreak and tied weld
  - tied contact slave node and master segment pairs
  - stabilization history of elements (hourglass and drilling)

- This is an ongoing project which is intended to be supported in an Implicit GUI in LS-PrePost for facilitating its setup.
- If IACC = 1 on *CONTROL_ACCURACY, "bad" implicit element formulations are automatically switched to type 2 (solid) or type 16 (shells).
- Fixed bug in d3eigv output for model with *CONTROL_IMPLICIT_-INERTIA_RELIEF.

- ***INITIAL**

  - *INITIAL_LAG_MAPPING (NELANGL = -1, 2nd line, 6th column):
    If NELANGL = -1, no mesh are generated or projected; just map the data if the 2nd run mesh geometry matches the 1st run one at its final cycle
  - *INITIAL_SOLID_VOLUME:
    Recalculate and reset initial volume of solid elements using material models with EOS before analysis if the original nodal position has been moved by nodal projections in contact initialization. This option eliminates calculation of non-physical initial hydrostatic pressure due to the nodal repositioning.
  - Fix ineffective *INITIAL_VELOCITY_GENERATION for part defined with *PART_INERTIA when ID = 0, STYPE = 0 and IRIGID = 1.
  - Fix incorrect initial velocity when ICID.ne.0 in *INITIAL_VELOCITY_-GENERATION, and rotational velocity, omega, is not zero and *PART_INERTIA is also present.
  - Fixed *INITIAL_STRESS_SHELL and *INITIAL_STRESS_TSHELL when used with *INTEGRATION_SHELL. The integration rule was getting lost leading to unnecessary interpolation of data.
  - Fixed dynain writing and reading of *INITIAL_STRESS_SHELL for the fully integrated C0 shell (shell form 20).
  - Enable multiple *INITIAL_VELOCITY_GENERATION keywords to be used with *ELEMENT_SHELL_COMPOSITE or *ELEMENT_TSHELL_-COMP-OSITE. Only one velocity generation keyword was supported previously.

  - With the new parameter IVADD, the *INITIAL_VEHCILE_KINEMATICS velocity field can be superimposed on pre-defined nodal velocities.
  - IZSHEAR = 2 on *INITIAL_STRESS_SECTION gets a special treatment for preloading bolts, each bolt is seen as an entity and the constraint is to prescribe the mean stress in the section and not in each element, meaning that the bolt is capable of take bending resistance, currently applies to the common low order elements (type -2,-1,1,2,10,13,15) for explicit and implicit

- Isogeometric Elements

  - Addition of HAZ (heat affected zone) features to IGA shells. No new keywords were added. The capability works identically to standard FE shells.
  - Reduce unit system sensitivity of *CONSTRAINED_NODE_TO_-NURBS_-PATCH_SET.
  - *ELEMENT_SOLID_NURBS_PATCH:

    - Isogeometric solid analysis is now available in both SMP (with consistency flag turned on) and MPP.
    - Activate user defined materials for isogeometric solids.

  - *ELEMENT_SHELL_NURBS_PATCH:

    - Isogeometric shell analysis is now available in SMP with consistency flag turned on. (MPP was already available.)
    - Add a power iteration method to get the maximum eigen-frequency for each isogeometric shell element. This will be used to set a reasonable scale factor for trimmed element.
    - Modify the time step scale factor for IGA trimmed shell element so that the overall time step will not be unreasonably small due to very small trimmed elements.

  - *ELEMENT_SHELL_NURBS_PATCH:

    - Fixed problem in decomposition for heavily trimmed NURBS-patches.
    - The problem may have occurred if most parts of a NURBS-patch are trimmed (not actually part of the geometry). In such cases the MPP decomposition could have gone wrong.

  - Write element information for NURBS shells to d3hsp (Element ID, Part ID, number of nodes and connectivity). This is invoked by setting NPOPT = 1 in *CONTROL_OUTPUT.
  - Fix bug when NURBS shells are present in a model and extra DOFs where assigned (*NODE_SCALAR, or shell forms 24/25/26).
  - Fix for *ELEMENT_SHELL_NURBS_PATCH when applying body force using *LOAD_BODY_GENERALIZED or *LOAD_BODY_PARTS. Apply the body force on the control points.

- Add support for initialization of shell thickness at in-plane int. point for *ELEMENT_SHELL_NURBS_PATCH(_TRIMMED) using *INITIAL_-STRAIN_SHELL_NURBS_PATCH (SMP and MPP).
- Add support for initialization of stresses, plastic strain, history variables and strains for *ELEMENT_SHELL_NURBS_PATCH(_TRIMMED) using *INITIAL_STRESS/STRAIN_SHELL_NURBS_PATCH.
- Add support of writing *ELEMENT_SHELL_NURBS_PATCH_TRIMMED to ASCII dynain file.
- Add support of *PART_COMPOSITE for isogeometric shells.
- *ELEMENT_SHELL_NURBS_PATCH & *CONTROL_ADAPTIVE:
  Allow adaptivity for regular shell elements if NURBS patches are in the model.
- Add possibility to define a negative real value for NISR and NISS to define a desired size of the automatically created interpolation shell elements. This may be especially useful when using NURBS elements for rigid tools in forming applications.
- Allow *CONTROL_FORMING_AUTOPOSITION_PARAMETER with IGA NURBS shells.
- Allow *PART_MOVE with IGA NURBS shells.
- Write stresses of interpolation elements to elout for *ELEMENT_SOLID_-NURBS_PATCH.
- Add support for initialization of stresses, plastic strain, history variables and strains for *ELEMENT_SOLID_NURBS_PATCH using *INITIAL_-STRESS/STRAIN_SOLID_NURBS_PATCH.

- ***LOAD**

  - Fix INCLUDE_TRANSFORM offset for coordinate systems in *LOAD_-NODE_POINT which were using the wrong offset.
  - Added support for *LOAD_THERMAL for higher order (quadratic and cubic) solids. The temperature at the nodes is interpolated from the nodes to the integration points so that the temperature is not the same at all integration points.
  - Correct the issue where use of *LOAD_BODY_ is applied to a model with rigid bodies to avoid null elemental stiffness matrices.
  - Fixed *LOAD_THERMAL_VARIABLE_ELEMENT_TSHELL. It was not working.
  - Fixed *LOAD_THERMAL_VARIABLE when used with shell form 2 that has default warping stiffness (BWC = 2 on *CONTROL_SHELL), and when OSU = 1 on *CONTROL_ACCURACY to activate an objective stress update.
  - *LOAD_SEGMENT_CONTACT_MASK is now supported for mortar contact in both SMP and MPP, in both implicit and explicit.
  - Fix bug in accessing ground motion ID (*DEFINE_GROUND_MOTION) from *LOAD_SEISMIC_SSI.
  - Allow *DEFINE_FUNCTION for *LOAD_THERMAL_LOAD_CURVE.

- *LOAD_SURFACE_STRESS:
  Fix a bug when there are more than 1 contact on one side of the blank, the area calculation was larger and the pressure smaller than it should be.
- *LOAD_BODY_VECTOR:
  Fix a bug in writing binary dynain (MPP).

- ***MAT and *EOS**

  - Fix bug in *MAT_079. The equations giving the influence of pressure on stiffness and strength were not exactly as written in the manual. PREF had been used in place of (PREF-P0). The code has been corrected to match the manual. This change will affect results from existing models, but usually P0 would be given a very small value so in most cases the difference should not be significant.
  - *MAT_089:

    - Now works with Tetrahedron ELFORM = 13. Previously the behavior was the same as for ELFORM = 10 (volumetric locking could occur).
    - Fixed bug affecting solid elements only - timestep calculation was wrong. Response could be unstable especially for higher values of Poisson's ratio, e.g. 0.4. Workaround was to reduce the timestep.

  - *MAT_119 unload option 3 - very small displacements followed by unloading could result in excessive unload stiffness and unexpectedly large mass-scaling, arising from small numerical rounding errors in the interpolated version of the load curves LCIDTR, LCIDTS, etc. Now fixed.
  - *MAT_169 (*MAT_ARUP_ADHESIVE) - enabled for implicit analysis.
  - *MAT_172 (*MAT_CONCRETE_EC2):

    - Fixed bug for combination of MAT_172 with Staged Construction (*CONTROL_STAGED_CONSTRUCTION). While elements were dormant, crushing damage could occur that persisted after the element became active.
    - Through-thickness strain was wrongly calculated when cracks are opening or closing. This strain is only an output parameter and does not affect the other results, but could potentially have led to unexpected element deletion if used with *MAT_ADD_EROSION.
    - Enabled CMPFLG (*DATABASE_EXTENT_BINARY).

  - *MAT_197 (MAT_SEISMIC_ISOLATOR)
  - Added optional rotational stiffness (new optional Card 7). This is useful when multiple isolator elements are stacked on top of each other.
  - Added new TYPE = 3 Lead Rubber Bearing. Includes cavitation in tension, buckling in compression, degradation of shear strength due to heating of the lead core.
  - *MAT_203

        - Added history variables 10 & 11 for post-processing: high-tide tensile strains in the two local reinforcement directions.
        - Enabled CMPFLG (*DATABASE_EXTENT_BINARY).
        - Fix bug in hysteresis behaviour that could occasionally cause error terminations.
        - Elements are now eroded when existing input parameter EPDAM2 is reached. If EPDAM2 and EPDAM1 are not defined, or if DRESID is non-zero, the element never erodes.

    - *MAT_208 (*MAT_BOLT_BEAM):

        - Now the element erodes when it reaches failure criteria. Previously, the forces and moments were set to zero but the element did not get deleted.
        - Added new input field AXSHFL to control whether shear displacements (excluding sliding within the clearance gap) are capable of lengthening the bolt and increasing axial tension. By default (as in R10 and previous versions), shear displacements can increase axial tension. This is reasonable if the shear deformation is associated with rotation or bending of the bolt itself while the plates remain a fixed distance apart. But if the shearing is largely due to deformation of the bearing surfaces then (in real life) the bolt length does not increase and the tension is unaffected. This latter behaviour can now be invoked by setting AXSHFL to 1. This effect will be more significant as shear displacements become large. Note that displacements associated with sliding across the clearance gap are always ignored for purposes of calculating the axial load.

    - *MAT_211 (*MAT_SPR_JLR) fixed bug - load curve IDs 8 or 9 digits long not read correctly from *MAT card in single precision version
    - GISSMO and DIEM damage models now work with higher order solid elements.
    - Instead of using a number or percentage of failed integration points to trigger erosion of higher order solids in *MAT_ADD_EROSION, use volume fraction of failed material. The reason for this approach is that the volume associated with each integration point varies widely within the higher order solids.
    - Fix a bogus error message for *MAT_ADD_PORE_AIR that occurs when PERMX is defined as "0", while PERMY and/or PERMZ are not zero
    - *MODULE feature for user-defined materials is built in with "sharelib" binaries.
    - *MAT_153/*MAT_DAMAGE_3: Extended to up to 10 backstresses that can be determined
    - from stress-strain data for solid/shell elements. This extension also supports implicit dynamics.

- Added support for cohesive shells in *MAT_240.
- Fix the convergence issue with plasticity algorithm in *MAT_260A/*MAT_-STOUGHTON_NON_ASSOCIATED_FLOW.
- Fix incorrect stress initialization when using *MAT_005/*MAT_-SOIL_-AND_FOAM with vol. strain vs pressure defined using load curve LCID, together with *LOAD_DENSITY_DEPTH.
- Fix seg fault or incorrect stresses when initializing stresses using *INITIAL_-STRESS_SOLID for *MAT_107/*MAT_MODIFIED_JOHN-SON_COOK.
- Fix strain rate effects on *MAT_157/*MAT_ANISOTROPIC_ELASTIC_-PLASTIC for implicit static analysis.
- Fix seg fault when using *MAT_157/*MAT_ANISOTROPIC_ELASTIC_-PLASTIC and MAT_ANISOTROPIC_ELASTIC_PLASTIC for 2D analysis.
- Fix incorrect results when using *MAT_TABULATED_JOHN-SON_-COOK/*MAT_224 with table LCKT defined and the first abscissa value set to a negative temperature.
- Increase robustness of *MAT_BARLAT_ANISOTROPIC_PLAS-TICI-TY/*MAT_033 for solids.
- Fix input error when using *MAT_ELASTIC_WITH_VISCOSITY_-CURVE/*MAT_060c when LCID = 0.
- Store computed yield strength as history variable #6 for *MAT_-PIECE-WISE_LINEAR_PLASTIC_THERMAL/*MAT_255.
- Fix to work-energy sums when *MAT_090/*MAT_ACOUSTIC material is used. If kinematics are not requested via *CONTROL_ACOUSTIC (the default), then acoustic element energies are not included in the solution sums. If kinematics are requested, then the energies are included. Previously, acoustic elements contributed to the work, but not the kinetic energy. Note - computing acoustic element kinematics is not required to calculate acoustic pressures and roughly doubles the cost of the acoustic elements. This correction has no affect upon the computed solution.
- Fixed brick material *MAT_089 with hourglass form 6. The hourglass scaling was bad which caused bad results.
- Modified shell *MAT_214 to prevent dividing by zero in some cases.
- Fixed *MAT_091 when used with shell form 16. The material directions were being calculated incorrectly causing the stress to be wrong.
- Fixed conflict between RTCL damage in *MAT_123 and heat affected zones. The TRIAX parameter was being overwritten causing possibly excessive damage.
- Node and connectivity data for elements that use *MAT_FABRIC/*MAT_-034 has been restored to the the dynain file. It had been removed along with stress and strain data which was removed because the fabric material cannot be prestressed by dynain.
- Fixed laminated shell theory with shell *MAT_172. It was calculating wrong shear strain.

- Fixed *MAT_023, *MAT_072, *MAT_083, *MAT_153, *MAT_223, *MAT_-229, and *MAT_259 when used with linear solid form 18. Some material constants were not passed correctly.
- Fixed *MAT_024 plastic strain failure for beam element forms 4 and 5. The elements were not failing.
- Fixed *MAT_077 for tetrahedral solid form 13. Some newer options were not working.
- Fixed *MAT_244 (shells and solids) when LCY1, LCY2, LCY3, LCY4, or LCY5 was left equal to zero. In this case, the code attempted to use nonexistent data, and it's unclear how this affected results. This caused the Windows executable, to error terminate with a "Program Exception - access violation" message.
- Fixed a problem where *MAT_172 was used to model 100% reinforcement and TYPEC was 3 or 6, and Young's Modulus of concrete (ET36) was input as zero. For steel only, this value should not matter, but it caused a divide by zero error during initialization. This is now prevented by having ET36 default to a small positive number for 100% reinforcement.
- Fixed shell form 17 when used with *MAT_077. There was no call to the stress update routine.
- Fixed ESORT > 0 when used with shell form 18 and *MAT_077. The triangular elements are now switched to triangular shell form 17 to be compatible with quad shell form 18.
- Fixed various errors in *MAT_NONLOCAL:

  - When both incremental and non-incremental data are requested for smoothing.
  - When used with shell materials with more than one in-plane point, or brick elements with more than 1 integration point. When coefficients are calculated for nearby points, the sum of coefficients was excluding other points within the same element or layer. The sum is used to normalize the weighted average. Since it was too small, the function that is smoothed would tend to grow.
  - Fixed an MPP only bug in the *MAT_NONLOCAL material averaging. A buffer could be overwritten if there was more than one nonlocal definition in the model. The same error was fixed for *MAT_CODAM2, *MAT_GURSON_RCDC, or *MAT_PLASTICITY_-WITH_DAMAGE with the RCDC option and the characteristic element length or nonlocal radius was defined.
  - The nonlocal search messages to screen and message files were modified to be more informative. During the search, a progress message is printed with each 100 million points added to the lists. This is intended to combat the perception that the code has hung.

- Added new options for *MAT_SPOTWELD_DAMAGE-FAILURE failure by OPT 6. There are 2 new TFLAG options:

- TFLAG = 2 causes the max sheet thickness to be used.
- TFLAG = 3 causes the sum of thicknesses to be used.

- A sheet thickness scale factor was added which scales the sheet thickness calculated by the TFLAG options.
- Fixed *MAT_126 when used with tshell forms 3, 5, or 7. The angle initialization was incorrect leading to bad stress values.
- Enabled *MAT_025, *MAT_173, *MAT_193, and *MAT_198 to be used with tshell forms 3, 5, and 7.
- Enabled *MAT_123 to work with tshell forms 3, 5, and 7 using options EPSTHIN and LCTSRF, and also NUMINT < 0 option. All 3 options were previously only available for thin shells and tshell forms 1, 2, and 6.
- Fixed the stress from *EOS_GASKET model when used with tshell form 2. Also fixed the compressive failure strain (CFS) for *EOS_GASKET with bricks or tshell forms 5 and 7. It was setting CFS = TFS.
- Fixed the behavior of NUMFIP on *MAT_ADD_EROSION when used with shell or tshell composite sections. When counting failed layers, the zeroing of the counter did not happen unless the first layer of the composite used *MAT_ADD_EROSION. This would have caused elements to fail that should not have failed. All 3 options, NUMFIP > 0, -100 < NUMFIP < 0, and NUMFIP < -100 are now working.
- Fixed the eigenvalue calculation in the first cycle when the model has solids or tshell forms 3, 5, or 7, and these elements are used with *MAT_022, *MAT_054, *MAT_055, *MAT_059, or *MAT_213.
- Fixed thermal strains in *MAT_021 tshells. The material directions were not processed correctly causing incorrect thermal strains. Both deformation and output were fixed and are now working for all tshell forms.
- Fixed the material direction when using shell *MAT_091. Also enabled output in the material direction when CMPFLG = 1 on *DATA-BASE_EXTENT_BINARY.
- If EPSR and EPSF are defined in *MAT_054, correct computation of transverse shears strains for solids and tshells.
- Add possibility to use failure criterion in *MAT_054 for solids in a transversal isotropic manner. It is assumed that the material 1-direction is the main axis and that the behavior in the 2-3 plane is isotropic. This feature is invoked by setting TI = 1 (card 2, column 7) in *MAT_054.
- Fix bug for shear stiffness behavior in *MAT_058 when using a table definition for GAB and only providing stress-strain-curves for positive shear.
- Fix bug for strain-rate dependent stiffness behavior in *MAT_058 when using a table definition for EA, EB or GAB under compressive loading.
- Add history variable for ellipsoidal failure surface for *MAT_059 (SC.lt.0), shells only. Now history variable 8 is the failure surface "f", which is similar to the Tsai-Wu criterion.

- *MAT_100_DA(*MAT_SPOTWELD_DAIMLERCHRYSLER),*DEFINE_-CONNECTION_PROPERTIES:Add the possibility to define a yield curve or table for DSIGY, SIGY in case of using PRUL = 1.
- For *MAT_100 (*MAT_SPOTWELD), OPT = 0/-1: Add the possibility to define force/moment resultants as a function of the effective strain rate, by defining an appropriate load curve. This will be indicated by a negative value for NRR, NRS, NRT, MRR, MSS, MTT. This functionality is implemented for beam, hex and hex-assembly spotwelds.
- Extended capability of initializing *MAT_157 solids using input variable IHIS and *INITIAL_STRESS_SOLID.
  Blocks of material parameters that can be initialized in this way are:

  - material direction (q11,...,q33) - 6 values
  - stiffnesses (c11,c12,c13,...,c66) - 21 values
  - anisotropic constants (F,G,H,L,M,N) - 6 values
  - curve/table - ID (LCSS) - 1 value
  - strengths (XC,XT,YC,YT,ZT,ZC,SXY,SYZ,SZX) - 9 values << NEW

  See User's Manual for details.
- Add value of failure criterion (Tsai-Wu or Tsai-Hill) to history variable 10 for postprocessing in *MAT_157. Shell elements only.
- Fix bug in *MAT_261 when table definition is used to define the non-linear in-plane shear behavior (LCSS). This applies to shells/tshells/solids.
- Add default strength limits (XC, ... 1.e+16) if they are not defined for *MAT_-261/*MAT_262.
- Added a criterion to avoid possible snapback behavior in *MAT_262 by only allowing certain "softening" modulus with respect to the elastic stiffness. This might be important when using rate-dependent strength limits (*DEFINE_CURVE) and the values for the fracture toughnesses are not properly set.
- *MAT_4A_MICROMEC /*MAT_215:
  Add strength limit XC for failure in fiber compression.
- *MAT_GENERALIZED_PHASE_CHANGE/*MAT_254:

  - Added latent heat algorithm to *MAT_254. Input follows the same phase transformation matrix as the other transformation related parameters.
  - Added user-defined history variables. Up to 8 history variables can be defined using *DEFINE_FUNCTION. Parameter list for functions includes time, user histories, phase distribution, temperature, peak temperature, temperature rate, stresses, and plastic strains.

- New parameter dtemp in *MAT_CWM/*MAT_270 that can invoke a subcycling in the material formulation, if the temperature jump within a time step exceeds the limit defined by dtemp.

- Activated latent heat algorithm for thermal material *MAT_THERMAL_-CWM/*MAT_T07.
- *MAT_CORUS_VEGTER/*MAT_136:

  - Renamed material from *MAT_CORUS_VEGTER to *MAT_VEGTER, as Corus no longer exists.
  - Tabular input for strain rate dependency implemented.
  - New option *MAT_VEGTER_STANDARD:
    --expects a parameter alpha_ps,theta instead of the second component of the plane strain point
    --input consistent with most literature data
    --material routine itself is unchanged
    --temperature dependent strain rate dependency also possible with the Bergstrom equation
  - New option *MAT_VEGTER_2017:
    --uses a simplified input
    --Input of tensile strengths (Rm0,Rm45,Rm90), uniform elongation (Ag0,Ag45,Ag90) and plastic anisotropy (R0,R45,R90)
    --based on a model provided by TATA steel, the standard input data for MAT_VEGTER is reproduced
    --material routine itself is unchanged
    --temperature dependent strain rate dependency also possible with the Bergstrom equation

- *MAT_REINFORCED_THERMOPLASTIC/MAT_249:
  Changed handling of post-processing data in history values. User can define the post-processing data to be written into histories.
- Add new failure criterion DTMIN (minimum time step) to *MAT_ADD_-EROSION.
- Add new failure criterion MXTMP (maximum temperature) to *MAT_-ADD_EROSION for solid and shell elements.
- Add new option LCFLD < 0 to *MAT_ADD_EROSION. In this case, |LCFLD| refers to a table where FLD curves are shell thickness dependent (in contrast, existing LCFLD > 0 provides rate dependence).
- Fix for combination of *MAT_ADD_EROSION and tshell form 2 or beams. Strain-based criteria (e.g. MXEPS) did not work correctly before.
- Several changes for *MAT_ADD_EROSION with IDAM = 1 (GISSMO):

  - If LCSDG < 0, then |LCSDG| refers to a *DEFINE_FUNCTION with arguments triaxiality and Lode parameter.
  - New option LCREGD < 0, where |LCREGD| refers to a table that contains regularization factor vs. element size curves for different triaxialities.
  - Compute element size for LCREGD not only once at t = 0 (SIZFLG = 0), but also after each adaptive refinement step.

- New option REFSZ < 0 of GISSMO provides a plastic strain value, that corresponds to that reference size, written to history variable ND+17.
- Improve robustness if DMGEXP < 1 is used.

- Fix for GISSMO to be used in adaptive remeshing. Evaluation of damage coupling flag could go wrong due to averaging of history variables in rezone step.
- Support DIEM (*MAT_ADD_EROSION with IDAM < 0) for beam form 1.
- Add new keywords *MAT_ADD_DAMAGE_GISSMO and *MAT_ADD_-DAMAGE_DIEM. The idea is to separate these damage models from *MAT_ADD_EROSION, where only pure element erosion criteria remain. That should simplify understanding the manual. Old input is still supported.
- Add new MIDFAIL options 2 and 3 to *MAT_ADD_DAMAGE_GISSMO.
- Add new keyword option _STOCHASTIC for *MAT_ADD_DAM-AGE_-GISSMO to allow spatially varying failure behavior when used together with *DEFINE_STOCHASTIC_VARIATION.
- Add new option HISVN to *MAT_ADD_DAMAGE_GISSMO: A user-defined history variable (e.g. hardness) can be used to modify the failure. LCS-DG is in this case a TABLE_3D with the arguments triaxiality, Lode parameter, and that history variable.
- Add option to *MAT_ADD_GENERALIZED_DAMAGE that allows defining the number of failed integration points (to trigger element erosion) for each history variable separately. Only applicable for shells.
- Add new MIDFAIL options 2, 3, and 4 to *MAT_ADD_GENERAL-IZED_-DAMAGE.
- Add NUMINT option to *MAT_089 for shell elements.
- Add two nonlocal failure criteria to *MAT_280. The first one works similar to the ENGCRT/RADCRT criterion of *MAT_ADD_EROSION. The second is similar to the SOFT option of *MAT_054, where tensile strength is reduced in next-to-failed (crashfront) elements.
- Add new material model *MAT_BARLAT_YLD2004/*MAT_199 for solid elements in explicit analysis.
- New option ITERS < 0 in *MAT_143 invokes an alternative plasticity algorithm. It also comes with a new logarithmic rate dependence option, IRATE = 2.
- Allow initial temperatures for *MAT_224_GYS to be set via *INITIAL_-STRESS_SOLID. Also, enable *MAT_224_GYS to be used in coupled thermal-mechanical analyses.
- Modified materials *MAT_234 and *MAT_235 so that they work with angles on the *SECTION_SHELL card to define material directions for layers.
- Add option HISOUT = 1 to store principal strains as history variables 25-27 for *MAT_181 (available for solid elements).
- New options for *MAT_240/*MAT_COHESIVE_MIXED_MODE_ELASTO-PLASTIC_RATE, invoked by adding _THERMAL, _3MODES, or _THER-

MAL_3MODES to the keyword. Allows temperature dependent material data and/or inclusion of a third deformation and fracture mode.

- Add new optional hardening rule HR = 10 to *MAT_036 and *MAT_243: table with pre-strain dependence.
- Variable BETA < 0 of *MAT_224 can now refer to a *DEFINE_TABLE_3D, where the dissipation factor can be defined as a function of temperature (TABLE_3D), strain rate (TABLE), and plastic strain (CURVE).
- Add principal strains as new history variables 18-20 in *MAT_083.
- Add new option IHYPO to *MAT_023. IHYPO = 1 switches the material model formulation from hyperelastic to hypoelastic for solids, which allows stress initialization through *INITIAL_STRESS_SOLID.
- Fix for *MAT_190 when used together with *DAMPING_PART_STIFFNESS and RYLEN = 2. Premature failure due to FLD was likely to occur.
- Allow the Material Model Driver (Appendix K) to be used in batch mode: If a file called "mmd.bat" exists in the working directory, then the commands contained therein get executed sequentially. Supported commands are print, cross, time, and quit.
- Algorithmically consistent tangent modulus implemented for *MAT_024 and *MAT_123.
- Add flag for allowing nonsymmetric tangent moduli in user materials. Currently supports only solid forms -1, -2 and 2.
- Fix bug whereby encrypted *MAT_075 data was echoed in d3hsp.
- For the case where an encrypted material model referenced load curve(s), the material type was revealed by d3hsp in the load curve usage summary. That is no longer the case.
- Change the stress update in *MAT_PML_ELASTIC_FLUID to make it more stable. Viscous damping is still needed to achieve stability.
- *MAT_232/*MAT_BIOT_HYSTERETIC is now supported in all solid element formulations.
- Added *MAT_293/*MAT_COMPRF.
  This material model simulates the behavior of pre-impregnated (prepreg) composite fibers during the high temperature preforming process. In addition to providing stress and strain, it also provides warp and weft yarn directions and stretch ratios after the forming process. The major applications of the model are for materials used in light weight automobile parts.
- Added *MAT_296/*MAT_ANAND_VISCOPLASTICITY. This visco-plastic model uses a set of evolution equations instead of loading-unloading criterion to describe dislocation motion and hardening or softening behavior of materials. This model can be applied to simulate solders used in electronic packaging.
- Added 2-way option for tshell formulation 5 in *MAT_054.
- *MAT_260B:
  - Set default value for P12 = -0.5, p22 = 1.0, p33 = 3.0.
  - Set default value for G12 = -0.5, G22 = 1.0, G33 = 3.0.

- Correct shear stress calculation error.
- Set default value for DEPSO to 0.001, so as to avoid division by zero.

◦ *MAT_260A:

- Set default value for R00 = 1.0, R45 = r00, R90 = r00
- Set default value for sig0, sig45, sig90, and sigb to make sure that the non-associated flow will be degenerated to associated flow.
- Add Equation of State.
- If SIGB is zero, then it is assumed that all the SIGS' are equal.
- Add Xue's failure model.

◦ *MAT_123:
When major strain is used as a failure criteria, the equivalent major strain is added to the current step by using the current strain ratio and the previous deformation strain carried over from the previous (forming) stage.
◦ *DEFINE_CURVE_STRESS:
Add new options to *DEFINE_CURVE_STRESS, add ITYPE = 1,2,3,4,5,11.
◦ *MAT_036:

- Output the optimized material parameters when using both R-values and the A, C, H, P parameters as input.
- Add an error message when the material model is attempted for tshells.
- Turn off the output of material parameters if encryption is used.

- **MPP**

◦ Suppress output of pfile information to d3hsp and messag if the *CONTROL_MPP_PFILE line itself is encrypted.
◦ Fix automatic setting of "decomp { bagref }" in the pfile, which was broken in revision 101313.
◦ Add MPP contact timing calls around force transducer initialization and net force calculations, so they are better represented in the contact timing table.
◦ Adjusted initialization of *PART_MODE for MPP so that certain processing only happens on processor 0.
◦ Correct problems with using MPP predecomposition when using jobid specifications. There were issues with initializing the file ids.
◦ Enhance MPP eigensolver to have a kinder, gentler termination when no eigenmodes found.
◦ For MPP, register the part in *PART_MODES to not be split across processes.
◦ Allow parameter expression in keyword *CONTROL_MPP_PFILE.
◦ MPP load balancing profiles are output to both .csv and .xy files.
◦ pfile directive "decomp { defgeo }" in a full deck restart causes decomposition to be re-done using the current deformed geometry. This can keep

elements in contact in the same processor and helps the MPP performance of models with large deformation for example SPH bird strike, car wading, etc.

- **Output**
  - *DATABASE_ELOUT, *DATABASE_HISTORY_BEAM: Fix old typo in beam history collection routine, the effect of which is not obvious but could possibly have led to incorrect output of some beam data in elout.
  - Do a better job deleting scratch LSDA files from the file system, including removing all %XXX extensions. This may help with some customer file system issues during large adaptive problems.
  - Add rotations (moments) for nodes in the bndout file if the problem has 6 degrees of freedom.
  - Implement new force collection routines for bndout data, which should do a better job of reporting just those forces/energies that are due to the boundary conditions applied.
  - Adjust implicit logic for secforc output for arclength method.
  - Fix a problem in reporting tied contact resultant forces in implicit.
  - Enhance logic that determines when to write out the last state to d3plot for implicit.
  - Fix an MPP bug for nodal stress/strain output, which could occur when more than one part share a node where stress/strain output is requested.
  - Fix missing plastic strain tensors in d3plot when STRFLG in *DATA-BASE_-EXTENT_BINARY is set and INTSTRN = 1 in *INTERFACE_-SPRING-BACK.
  - Fixed stress output for shell forms 13, 14 and 15 when NIP = 4 on *SECTION_SHELL and MAXINT < 0 on *DATABASE_EXTENT_BINARY. The stress outputted was incorrect.
  - Fixed issues when reporting eroded hourglass energy to glstat and matsum. For brick elements, the eroded energy was counted twice, and for thick shells, eroded energy was not counted at all. Both issues caused energy to not balance in the glstat data.
  - Enabled secfor output for higher order shell forms 23 and 24.
  - The 'Effective Stress' option is supported in *DEFINE_MATERIAL_HIS-TORIES, meaning that the material dependent equivalent stress is output to the d3plot database. Currently only *MAT_036 and *MAT_133 honors this option, all other materials will output the von Mises stress for now.
  - Variable NTIED of *DATABASE_EXTENT_INTFOR can be used to output the tied status on the slave side of tied mortar contact, including tiebreak and tied weld.
  - Fixed intfor database for SMP if the file size is larger than 7M byte.
  - Echo *DATABASE_EXTENT_COMP flags in d3hsp.
  - Fixed bug for d3plot if both DECOMP = 5 or 6 in *DATABASE_-EXTENT_-BINARY, and PSETID is specified in *DATABASE_BINARY_-D3PLOT.

- Fixed bug for *DATABASE_HISTORY_DISCRETE if BEAM = 1 in *DATABASE_BINARY_D3PLOT.
- Fixed bug in writing *SECTION_*_TITLE to d3prop file.
- Fixed legend of ssstat in binout.
- Fixed wrong cross-section ID in secforc if dyna.str is used.
- Write dynain file in I10 or long format is keyword input includes those formats.
- Fixed bug affecting d3plot when analysis includes dynamic relaxation.
- Added one additional significant digit to *NODE coordinates in dynain.
- Fix bug in d3plot when both higher order and ALE elements are in the model.
- *DATABASE_HISTORY_NODE_LOCAL_ID:
  Fix bad node labels in nodout when long input format is used.
- Fixed bug where HEADING in *DATABASE_HISTORY_NODE_ID was limited to 10 characters in free (comma-delimited) format.

- **Restarts**

  - Correct implicit memory pointers to work correctly on restart.
  - Also corrected dump and restart lengths for implicit restart.
  - Properly start up explicit LaGrange Multiplier treatment of joints at restart.
  - Fix seg fault when using *DELETE_CONTACT for restart when running with SMP.
  - Fix error termination for full deck restart that includes *DEFINE_ELEMENT_DEATH.
  - Fix input error during structured input when using *INITIAL_VELOCITY_-GENERATION and *CHANGE_VELOCITY_GEN-ERATION together in a full deck restart.
  - Fix incorrect full deck restart analysis if initial run was implicit and the full deck restart run is explicit. This affects MPP only.
  - *CHANGE_CURVE_DEFINITION in a restart was not working properly to modify curve LCDT in *DATABASE_BINARY_D3PLOT.
  - Fix corrupt d3part database affect a small restart with *DELETE_PART and *DELETE_FSI.
  - Fix bug in full deck restart of tied contact that resulted in force discontinuities across the restart.

- ***SENSOR**

  - Fix a bug for *SENSOR_DEFINE_FORCE when FTYPE = JOINT, which occurs when joint id > 9999999.
  - Fix a bug for *SENSOR_CONTROL for type = SPC, that was introduced in r115453 & r115457.
  - The bug occurs when a SPC boundary condition was initially off and then turned on later.

   - Fix a bug for *SENSOR_DEFINE_..._SET that can occur if sensor command is defined before the definition of the related *SET commands.
   - Add MTYPE = BNDOUT to *SENSOR_DEFINE_MISC to trace energy reported in bndout.
   - Add MTYPE = MATSUM to *SENSOR_DEFINE_MISC to trace energy reported in matsum.
   - Fix a sensor bug that occurs when *SENSOR_DEFINE_FORCE is used to trace the force associated with prescribed motion.
   - Fix a sensor bug that occurs when *SENSOR_DEFINE_MISC has MTYPE = CURVE that refers to a load curve of *DEFINE_CURVE_FUNCTION. The bug was introduced in r115362.
   - Couple thermal-only analysis with general sensor so that sensor can be used to terminate the analysis.
   - Add TYPE = ELESET to *SENSOR_CONTROL to erode elements.
   - Add MTYPE = NFAILE to *SENSOR_DEFINE_MISC to trace number of failed elements.
   - Fix a bug affecting *SENSOR_DEFINE_FORCE with FTYPE = JOINTSTIF.
   - Fix bug in spotweld-constraint handling in MPP when TYPE = SPOTWELD in *SENSOR_CONTROL.
   - Fix a bug in *SENSOR_CONTROL when TYPE = SPC. The bug occurs when a node is involved in more than one SPC definition.
   - Add MTYPE = CURVE for *SENSOR_DEFINE_MISC so that sensor can trace the value of a time-dependent curve or a *DEFINE_CURVE_-FUNCTION.

- **SPG (Smooth Particle Galerkin)**

   - Added *CONSTRAINED_IMMERSED_IN_SPG for composite analysis.
     This keyword applies to immersion of beam or shell elements in SPG solids. This is a new feature of SPG method for failure analysis of some particular composites such as rebar in concrete and fiber-reinforced composites.
   - Added *CONTACT_SPG for self-contact in SPG method.
     This feature is useful in studying some self-contact for high velocity impact/penetration applications where failed particles may still interact in compression modes.
   - Optimize the SMP performance of SPG solid formulation 47.
   - *CONSTRAINED_IMMERSED_IN_SPG now works in MPP.
   - *MAT_181 is now supported for SPG solid formulation 47.

- **SPH (Smooth Particle Hydrodynamics)**

   - *CONTACT_2D_NODE_TO_SOLID:

- Contact was not robust when using master surface with sharp angles or thin structure, like a needle. It has been revised to handle better this kind of geometry, and take into account the thickness of slave nodes.
- Add a maximum parametric coordinate parameter MAXPAR for segment search (default = 1.05).
- Change format when transferring values from keyword format to structured format and pfile format. This change greatly improves precision.

- *CONTROL_SPH:
  Add IEROD = 3 to enforce zero deviatoric stress to eroded SPH particles, and conserve volumetric response if an EOS is defined.
- *DEFINE_SPH_To_SPH_COUPLING:
- Allow negative values of SRAD. In that case, we compute a contact distance based on volume instead of smoothing length.
- Fix bug for SPH parts using *MAT_147 (*MAT_FHWA_SOIL). History variables were improperly initialized, rendering the whole damage evolution aspect of the material inoperative with SPH.
- Implement enhancement for fluid formulations in SPH so that interaction between multiple SPH fluid parts is more robust.
- Fix bug affecting *DEFINE_BOX_SPH. This feature was broken.
- Fix bug for *DEFINE_ADAPTIVE SOLID_TO_SPH. History variables were not transmitted to the SPH particles properly for some materials.
- Add option to automatically compute the contact thickness of slave SPH particles using ITHK in *CONTROL_SPH. The thickness calculated by ITHK = 1 is used only if SST (*CONTACT_-AUTOMATIC_NODES_TO_-SURFACE) or OFFD (*CONTACT_2D_NODE_-TO_SOLID) are set to zero. All default behaviors remain unchanged.
- Enable SPH in full deck restart.
- SPH part is now identified as element type 4 in d3hsp like this, "element type .... = 4"
- *ELEMENT_SPH_VOLUME specifies volume instead of mass for SPH particle.
- Added in adaptive SPH formulation (ASPH) with anisotropic smoothing tensor (FORM = 9, SMP only) and renormalization approximation for adaptive SPH formulation with anisotropic smoothing tensor (FORM = 10, SMP only). The axes of those forms evolve automatically to follow the mean particles spacing as it varies in time, space and direction based on the strain rate tensors. These forms have better accuracy and stability than the standard SPH formulation.
- Added a new function ISHOW = 1 in *CONTROL_SPH whereby SPH particles generated by *DEFINE_ADAPTIVE_SOLID_TO_SPH will be shown as points instead of spheres before activating them, i.e., before erosion of parent solid.

- Added in a new variable ISPHKERN in *SECTION_SPH for higher order kernel option: with ISPHKERN = 1, a quintic spline kernel function (smoother and more accurate) will be used instead (supported in FORM = 0,1,4,9,10 in *CONTROL_SPH for both SMP and MPP version of executables).
- Added binout support for SMP rcforc file with *CONTACT_2D_NODE_-TO_SOLID. Write user id (instead of internal ID) in rcforc for *CONTACT_-2D_NODE_TO_SOLID. Fixed rcforc output in the case of multiple *CONTACT_2D_NODE_TO_SOLID contacts (bug 13454).
- Added in _SPH_VARIATION option (SMP only) for *DEFINE_STOCHASTIC_ELEMENT with SPH particles.
- Supported *DEFINE_STOCHASTIC_VARIATION option for SPH particles (combined with *MAT....._STOCHASTIC option) with material models *MAT_010, *MAT_015, and *MAT_024.
- Support conventional mass scaling (negative DT2MS in *CONTROL_-TIMESTEP) for SPH.
  This feature works for both SPH 3D and 2D cases.

- **Thermal Solver**

  - *CONTROL_EXPLICIT_THERMAL_... and *CONTROL_ADAPTIVE (ADPTYP = 8, 1st line, 3rd column): For the explicit thermal solver, map the temperatures after each adaptive restart for ADPTYP = 7,8.
  - *CONTROL_EXPLICIT_THERMAL_...: Implement the explicit thermal solver in ALE 2D.
  - *CONTROL_EXPLICIT_THERMAL_CONTACT:
  - If part id < 0 in the *SET_PART called by *CONTROL_-EXPLICIT_THERMAL_CONTACT, the autocontact is activated for that part. Otherwise, by default, the thermal contact is only searched between different parts.
  - Added death time TDEATH and birth time TBIRTH to *BOUNDARY_TEMPERATURE_NODE/SET in order to activate and deactivate temperature constraints.
  - Add iterative solution option for the thermal radiation boundary condition linear equation solver. Correct logic used in determining the acceptable thermal solver options for SMP.
  - Fixed bug where dynain was missing thermal history variables.
  - New keyword *BOUNDARY_TEMPERATURE_RSW:

    - Prescribe nodal temperatures within a (possibly moving) ellipsoidal region of the structure.
    - Temperatures for the center and the boundary of the ellipsoid have to be input. In between there is a quadratic approximation.
    - Outside of the ellipsoid, no temperature values are prescribed.
    - Position and axis of symmetry are defined by to nodes.
    - Applicable to solid and thermal thick shell elements.

- Modification of variable time stepping in thermal solver:

  - Ensure that the step size of the last time step before a breakpoint is not smaller than half the previous step. If necessary, the last two time step sizes are averaged. The breakpoints are still hit exactly. This avoids drastic step size reductions (sometimes by some orders of magnitude) that slow down the further simulation.
  - Accept load curve input for dtmin, dtmax and dtemp in *CONTROL_-THERMAL_TIMESTEP. As usual if a negative integer number is given its absolute value refers to the load curve ID.

- *LOAD_HEAT_CONTROLLER now implemented for MPP.
- *BOUNDARY_THERMAL_WELD_TRAJECTORY:

  - Additional option for heat source definition. With IFORM.eq.5, the energy rate distribution does not have a pre-defined form, but can be given as a function of the local coordinates r,s,t, the time and the current weld velocity. The formula is input with a *DEFINE_-FUNCTION keyword.
  - Fixed d3hsp output. Now external IDs of used load curves instead of internal ids are written and cross reference for the curves' usage is output

- Thermal edge contact:

  - Models heat transfer from and to a shell edge onto a surface (solid facet or shell).
  - Shell edges have to belong to thermal thick shells (THSHEL = 1 in *CONTROL_SHELL).
  - Shell edges are on the slave side.
  - Activated if parameter ALGO is larger than 1 (2 = two_way, 3 = one_way).
  - Available in both MPP and SMP.

- Composite thick shells:

  - Added composite thick shell functionality to the structure heat transfer solver.
  - Up to this point the lay-up defined by *PART_COMPOSITE_TSHELL has not been taken into account by the thermal solver. Now, additional degrees of freedom are generated and the element is split into virtual elements in the element routine. The implementation is very similar to what has been done for thin composite shells defined by *PART_-COMPOSITE.

- Fixed error in reading *SECTION_SHELL_THERMAL

- *DEFINE_CURVE_FUNCTION is now supported in the thermal solver.

- **XFEM (eXtended Finite Element Method)**

  - Added erosion option for XFEM shells.

- **Miscellaneous**

  - *DAMPING_FREQUENCY_RANGE_DEFORM can now be applied to tshells.
    In previous versions it worked only for solids, beams, shells, and discrete elements.
  - Fixed bug in *DAMPING_FREQUENCY_RANGE_DEFORM.
  - The damping's contribution to internal energy was wrongly calculated. This did not affect the solution (stresses, displacements, etc), only the output values of internal energy.
  - Update to external case driver to support LSDA based includes in the keyword input.
  - Fix incorrect handling of symmetric load curves when checking discretization errors, which resulted in incorrect and misleading error messages.
  - Fix initialization problem that could arise if more than one *INTERFACE_-LINKING instance references the same *INTERFACE_-COMPONENT.
  - Fix some memory allocation and initialization related to *INTER-FACE_-LINKING.
  - Improve reporting of seatbelt input errors in nastran reader.
  - Added *DEFINE_DRIFT_REMOVE to provide a correction to curves to compensate for errors in accelerometers.
  - Fix a bug that occurs when *INCLUDE_TRANSFORM is used together with *DEFINE_BOX and/or *PART_INERTIA.
  - Fix a bug that generates duplicate parts even when *PART_DUPLICATE is never defined.
  - Automatically merge DEFORMABLE_TO_RIGID_AUTOMATIC cards with the same options for better performance.
  - *DEFINE_PRESSURE_TUBE: Added 2nd keyword input line for viscosity, step size, and damping. Added support for automatically generated shell/solid tubes.
  - Fix wrong pointer used for section id/properties when generating part for visualization of rigid wall during adaptivity. This caused error message, KEY+137, during adaptivity.
  - Fix input error for duplicate part if multiple *RIGIDWALL_-GEOMETRIC_..._DISPLAY keywords are used, some with part id PID specified and some not.
  - Fix non-effective OPTIONs DBOX, DVOL, DSOLID, DSHELL, DTSHELL, DSEG for for deleting segments in *SET_SEGMENT_GENERAL.

- Add failure function terms (normal, bending, shear) as arguments of functions in *DEFINE_CONNECTION_PROPERTIES with PRUL.ge.2
- Add new option DGTYP = 5 to *DEFINE_CONNECTION_PROPERTIES.
- Fix for *DEFINE_CONNECTION_PROPERTIES with PRUL.ge.2 (*DEFINE_FUNCTION). The function IDs were not working with *INCLUDE_TRANSFORM and large IDs > 2**24 also failed.
- Add warning message for *DEFINE_FUNCTION: if the function name starts with i, j, k, l, m, n, I, J, K, L, M, or N, it will return an integer value.
- Removed the echo of each *CONTACT data and load curve data in d3hsp if that data are encrypted.
- Fixed bug in reading *CONTROL_REQUIRE_REVISION if free format is used.
- Fix a number of issues related to long format input:
  - Corrected legend that is written in nodout if long format is used.
  - Fixed bug in reading long format if *KEYWORD long = yes is used in include file.
  - Fixed bug in reading long format input for *INCLUDE_STAMPED_PART if optional 4 or 5 card doesn't exist
  - Fixed bug for long format bug for the following keywords:
    *MAT_EMMI
    *PERTURBATION_NODE
    *DEFINE_TABLE_MATRIX
    *INTERFACE_SPRINGBACK_LSDYNA
    *EOS_013
    *ALE_FSI_TO_LOAD_NODE
    *INITIAL_VOLUME_FRACTION
    *CONTROL_ADAPTIVE (3d)
    *CONTACT_AUTOMATIC_GENERAL_INTERIOR_MPP
    *CONTACT_ERODING_SINGLE_SURFACE_MPP
    *PERTURBATION_NODE
    *CONSTRAINED_GENERALIZED_WELD_FILLET
    *AIRBAG_HYBRID_ID
    *MAT_ADD_EROSION
    *FREQUENCY_DOMAIN_FRF
    *PARAMETER
    *DEFINE_HEX_SPOTWELD_ASSEMBLY
    *DEFINE_SPH_TO_SPH_COUPLING
    *ELEMENT_BEARING
    *SPH_COUPLING
    *ALE_2D
    *ELEMENT_BEAM_PIPE
    *BOUNDARY_PRESCRIBED_FINAL_GEOMETRY
    *PARAMETER_EXPRESSION
    *NODE_MERGE

*NODE_MERGE_SET
*ELEMENT_NURB_SOLID_PATH
*INITIAL_STRAIN_*
*BOUNDARY_THERMAL_BULKNODE
*CONSTRAINED_SHELL_TO_SOLID
*MAT_RIGID if geometry contact entity is used
*MAT_VISCOELASTIC_HILL_FOAM
*MAT_002

- Fixed bug in reading multiple entries of any of the following commands:
  *CONTROL_REQUIRE_REVISION
  *SECTION_SHELL_EFG
- Fixed bug for *INCLUDE_TRANSFORM in the case where the material ID is in alpha (non-numeric).
- Fixed bug in reading *SECTION_SHELL and *SECTION_SOLID if multiple sections are entered under one keyword.
- Correct the application of reaction forces from *INTERFACE_SSI_AUX_EMBEDDED in *INTERFACE_SSI (MPP only).
- Fix bug in running SSI problems in single precision.
- 3D adaptivity now partially supports *INCLUDE_TRANSFORM: a transformation of model geometry can now be specified using TRANID.
- *PART_MOVE:
  Extend to support *SET_PART_COLLECT and *SET_PART_ADD.
- Enable support of parametric filename in *INCLUDE when adaptivity is used.
- *INTERFACE_SPRINGBACK_LSDYNA, and *INCLUDE_BINARY:
  Fix missing SPCs output in binary dynain files in both SMP and MPP

Capabilities added to create LS-DYNA R12:

See release notes (published separately) for further details.

- ***AIRBAG**

  - Fix a bug for the output of area and leakage information for all parts constituting a control volume airbag. The bug occurred when an airbag was comprised of more than 10 parts.
  - *AIRBAG_HYBRID_CHEMKIN: fix an MPP bug introduced in r117881 that results in incorrect airbag pressure.
  - Remove the interaction between *AIRBAG_REFERENCE_GEOMETRY and *INITIAL_FOAM_REFERENCE_GEOMETRY so that they can only be used to define the reference geometry of shell and solid elements respectively.
  - *AIRBAG_PARTICLE (CPM):

        - New keyword *DEFINE_CPM_NPDATA to support more part-specific input for *AIRBAG_PARTICLE. Invoked by NPDATA > 0 and STYPEH = 2 or 3. Among other things, this new feature allows user to control smoothing algorithm for applying particle to fabric impulse.
        - Add new limit checking and self adjusting algorithm for 4th order polynomial nonmonotonic function of nonlinear CP to avoid incorrect result.
        - Support inflator mass flowrate curve (LCTi) using *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION and *DEFINE_FUNCTION.
        - New feature for *DEFINE_CPM_VENT for pushout vent to allow user to apply the ambient pressure when internal parts extend out from the external vent.
        - Support C23 (discharge coefficient) as function of vent area.
        - Support Autoliv porous leakage model (FVOPT = -1,-2) under CPM and CPM+UP switch capabilities.
        - Add tire inflation capability under CPM method to maintain the target tire pressure during the initial setup.
        - Support inflator volume evaluated from current geometry while using user-defined inflator chamber.
        - Fix bug for airbag with solid parts inside. The volume from those parts are excluded from bag volume.

    - New keyword *CONTROL_AIRBAG for CV (Control Volume) closed volume check.
    - Support multiple airbags when not all airbags have a reference geometry (*AIRBAG_REREFERENCE_GEOMETRY).

- ***ALE**

    - *INITIAL_ALE_MAPPING: Add a parameter SYM to apply specific mapping rules to elements and nodes outside the mesh of the previous run that wrote the mapping file to be used in the current run.
    - *ALE_MAPPING: Add new keyword to map data during a run (not just initially like with *INITIAL_ALE_MAPPING). A particular state from the mapping file is read.
    - *INITIAL_VOLUME_FRACTION_GEOMETRY and CNTTYP = 7: Allow the user to define the geometry using *DEFINE_FUNCTION.
    - *INITIAL_DETONATION: If PID < -1, |PID| is the ID of a *SET_PART.
    - *INITIAL_HYDROSTATIC_ALE, *ALE_AMBIENT_HYDROSTATIC: Account for the compaction to compute the initial and ambient hydrostatic pressure for *EOS_MIE_GRUNEISEN.
    - *ALE_PRESCRIBED_MOTION and *BOUNDARY_AMBIENT: Add a new parameter SIDR in both keywords to control their use during the dynamic relaxation phases (similar to SIDR in *DEFINE_CURVE).

- *BOUNDARY_AMBIENT: Set birth and death times with the first and last abscissa of *DEFINE_CURVE for the internal energy and relative volume time curves (LCID1 and LCID2).
- *CONTROL_ALE: Add a new variable BNDFLX to select only the ALE groups (*SET_MULTI-MATERIAL_GROUP_LIST) that can flux in when these groups are in the ALE elements along the mesh boundaries. Set BNDFLX = -1 to forbid any influx along free mesh boundaries.
- *ALE_BURN_SWITCH_MMG: Implement this keyword to allow the user to implement his own burn models.
- *ALE_MESH_INTERFACE: Implement this new keyword to mesh material interfaces with triangular shells (the material volume can also be meshed with tetrahedra).
- *CONSTRAINED_LAGRANGE_IN_SOLID: Implement a 2D version of the implicit thermal ALE coupling.
- *CONSTRAINED_LAGRANGE_IN_SOLID_EDGE with CTYPE = 5: Renew the coupling interface along the outside shell edges involved in the ALE coupling after shell erosion.
- *CONSTRAINED_LAGRANGE_IN_SOLID with 4 =< CTYPE <= 6 : Write in and read back from d3full the FSI relative displacements and other coupling arrays.
- *CONTROL_SEGMENTS_IN_ALE_COUPLING: New keyword that deactivates segments in the ALE penalty coupling (CTYPE = 4,5,6 in *CONSTRAINED_LAGRANGE_IN_SOLID) if the segments are face to face. If variable SYM = 1, a segment normally constrained is excluded from the coupling (similar to ISYM in *CONTROL_CONTACT).
- *DATABASE_BINARY_FSIFOR and *CONTROL_MPP_DECOMPOSITION_NUMPROC: Prevent fsifor from being corrupted by a MPP job starting right after MPP decomposition.
- *DATABASE_BINARY_FSIFOR and *SECTION_ALE2D: Output the interface force file (fsifor) in 2D (meshfringe in LS-PrePost should be used to see the coupling force and pressure distributions along the segment edges).
- *DATABASE_FSI: Output the center of pressure at fx-lc,fy-lc,fz-lc (or gx,gy,gz) in dbfsi if a node set (NDSETID) is provided in *DATABASE_FSI.
- *SECTION_ALE1D and *DATABASE_EXTENT_BINARY BEAMIP > 0: Access auxiliary and history variables in d3plot for ALE 1D elements (like NEIPH for solids and NEIPS for shells).
- *DATABASE_TRACER and TRACK = 2: The tracer moves with the ALE mesh.
- *DATABASE_TRACER and ALE mapping: Output the tracer final locations in a keyword format (to be included in the next run initialized by a mapping).
- A new ellipsoid geometry option is added in *INITIAL_VOLUME_FRACTION_GEOMETRY.
- *INITIAL_HYDROSTATIC_ALE now supports ALE single material with void formulation, that is, ELFORM 12 in *SECTION_SOLID.

- Structured ALE (S-ALE):
  - Support *EOS_MURNAGHAN to model weakly incompressible water.
  - *ALE_STRUCTURED_MESH_VOLUME_FILLING implementeded to fill ALE fluids into the initial S-ALE mesh and sidestep *INITIAL_VOLUME_FRACTION_GEOMETRY.
  - *ALE_STRUCTURED_FSI implemented to perform ALE fluid-structure interaction with structured ALE mesh. It could effectively stop the leakage often observed when using *CONSTRAINED_LAGRANGE_IN_SOLID.

- ***BATTERY (Electrochemistry Solver)**

(See also "Battery module Release" in the "EM (Electromagnetic Solver)" section.)

The *BATTERY family of keywords invokes the new electrochemistry solver, which is only available in double precision executables. *BATTERY keywords are documented in Volume III of the LS-DYNA Keyword User's Manual.

The keywords starting with *BATTERY refer to and control the problem set up for detailed one-dimensional electrochemistry modeling of battery cells. This is intended to be used for battery-thermal-structure-interaction problems.

Two Lithium Ion Battery (LIB) models are available via two one-dimensional full cell models.

- Single insertion which has this structure: Anode Lithium metal electrode/Separator/Cathode composite electrode
- Dual insertion which has this structure: Anode composite electrode/Separator/Cathode Composite electrode

These two cell models are based on an electrochemical model with various anode, separator, and cathode material properties. Therefore, the user can select a material property for their anode and cathode including an open-circuit potential (OCP). For the separator, the user can also select a material property among the different electrolytes which include all of the transport properties necessitated in battery simulation. Note that all transport properties are uniquely designed for each battery type.

The selection of the single insertion or dual insertion model is done in keyword *BATTERY_ECHEM_CONTROL_SOLVER. Here, users can select the battery run mode such as Galvanostatic or Potentiostatic, the number of cycles, and termination controls. Please see Vol III of the keyword manual for more details.

With the *BATTERY_ECHEM_CELL_GEOMETRY keyword, users can choose the length of each layer in a single cell including both sides of the current collectors, and the corresponding total number of mesh elements can also be input.

After this is done, then users need to set up their own values of all variables in the material cards for all layers: anode, cathode, and separator. The three main keywords are,

*BATTERY_ECHEM_MAT_ANODE

*BATTERY_ECHEM_MAT_CATHODE

*BATTERY_ECHEM_MAT_ELECTROLYTE

Here, first the OCP ID must be carefully selected (please refer to the keyword manual) and then Coulomb capacity, initial state of charge, thermodynamics data, and porous media data, respectively.

In the *BATTERY_ECHEM_THERMAL keyword card, the user can set the initial temperature and other thermal properties. However, please note that this card is ignored when the battery solver is coupled to the thermal and thermal-mechanical solvers.

Please note that the purpose of these 1D solvers is give the user a tool to test their OCP, material properties, and transport battery properties. For coupling the electrochemical battery solver to the thermal-mechanical solver, users must design their own mesh system with a different part number and must assign the individual part numbers for each solver. For example, if the electrochemical battery solver needs to solve within a specific part (all battery regions), then the user must assign that part number in the first variable column in each of these material keywords: *BATTERY_ECHEM_MAT_ANODE, *BATTERY_ECHEM_MAT_CATHODE, and *BATTERY_ECHEM_MAT_ELECTROLYTE. The same part number should be assigned for use by the structural thermal and mechanical solvers. Structural parts outside of the battery would have other part numbers. Please note that the 1D electrochemistry solver solves all elements assigned to the part numbers for the battery, and also note that this battery solver has its own 1D mesh system in each battery structural element, while the thermal-mechanical solver has its own 3D mesh system.

Currently however, only the dual insertion model is coupled with the LS-DYNA thermal-mechanical solvers. This model covers all Lithium-Ion batteries used by the battery manufacturers of cellular phones, and automotive industries batteries.

So, in version R12.0.0, users can simulate how the Lithium-Ion battery responds thermally and/or mechanically when external heat sources or impacting forces are applied to the battery pack. The battery model can be a single battery cell, a cell stack which has multiple cells connected (over 100 cells), and even multiple cell stacks connected in parallel. By controlling the solution scheme of the structural solvers, users can calculate the structural effects for a certain period of time of battery operation such as initial, middle, and end of state of battery discharging or charging.

For more detail about electrochemical battery theory, please refer to the LS-DYNA multiphysics theory manual. For keyword setup, please refer to the multiphysics (Vol III) keyword manual. In addition, for interested users, please request some sample keyword decks from LS-DYNA technical support, or email directly to kyoungsu.im@ansys.com.

- ***BOUNDARY**
  - Fix bug in *BOUNDARY_PRESCRIBED_MOTION which could cause velocity boundary conditions to be incorrectly handled for dof=+/- 4 or 8 if the node has more than one velocity boundary condition, e.g., one during dynamic relaxation and another during transient analysis.
  - Corrected MPP communication error associated with Implicit loading constraints for *BOUNDARY_PRESCRIBED_MOTION_FINAL_GEOMETRY.
  - For convection, flux and radiation boundary conditions, the parameter PSEROD on *BOUNDARY_... specifies a part set for which new segments exposed to the environment due to solid element erosion will inherit the boundary condition data.
  - New option BNDOUT2DYNAIN on *BOUNDARY_PRESCRIBED_MOTION_RIGID, allows for output of reaction force to dynain for use in subsequent simulations.
  - SPC2BND on *CONTROL_OUTPUT which will convert constraints on *MAT_RIGID to prescribed motions, for access to reaction forces in bndout.
  - Fix *BOUNDARY_NON_REFLECTING to have the correct velocity averaging and force redistribution for triangular segments.
  - *CONTROL_ADAPTIVE: Boundary conditions *BOUNDARY_RADIATION_SET and *BOUNDARY_CONVECTION_SET are updated in accordance with mesh changes due to 3D tet adaptivity under the condition that the segment set(s) are defined using *SET_SEGMENT_GENERAL with OPTION = PART.

- ***BLAST**

  - Fix blast wind velocity field for *LOAD_BLAST_ENHANCED BLAST = 4. Previously velocity was always the ground-reflected wave. Now, if a segment is not in the Mach stem region, the blast wind comes from both the incident-only and ground-reflected waves.
  - Fix a bug which could potentially affect results for models which contained both *LOAD_BLAST and *LOAD_BLAST_ENHANCED with TBO.ne.0. The time offsets could get mixed up or ignored.

- ***CESE** (Compressible Fluid Solver)

  - For the solvers not performing chemistry calculations, a switch has been made to a positivity-preserving method. For most problems, this will lead to a smaller time step, depending upon the shape of the smallest mesh element. Note that this method guarantees that density and internal energy remain positive.
  - Several bug fixes were made that make these solvers more stable.
  - For the CESE moving mesh conjugate heat transfer solver, fixed several bugs, including some MPP bugs that depended upon the mesh decomposition.

- ***CHEMISTRY**

  - The inflator model can be extended up to a 5 chamber model. Previously, it was limited to only a 3 chamber model which consists of the combustion chamber, gas chamber, and tank. In this version, we added two more gas plenum chambers to control both the speed of the gas and pressure into the tank.
  - Resolved issues with back flows when the pressure of the downstream gas chamber is higher than the upper chamber by correcting the pressure equation.
  - Updated the procedure for computing/loading an initial blast profile as an initial condition of a detonation or deflagration.
  - Fixed some problems in the axisymmetric combustion solver.
  - Chemically-reacting FSI flow problems using the Euler equation solver now work for any problem up to 60 species in the combustion gas. FSI with this Euler solver is strong and stabilized.

- ***DUALCESE (Dual CESE Compressible Fluid Solver)**

  - The *DUALCESE family of keywords invokes the new dual Conservation Element/Solution Element (dual CESE) compressible fluid solver. This solver is only available in double precision executables.
  - The *UNIT family of keywords is introduced which provides a coherent way to specify units in an LS-DYNA problem. For now, these keywords only

work with the dual CESE solver but may be extended to other solvers in the future.

- Dual CESE Solver Characteristics:

    - Explicit
    - Double precision
    - Dynamic memory handling
    - SMP and MPP
    - 3D solver / special case 2D solver and 2D axisymmetric solver. Note: 2D axisymmetric solver is only done for the CFD solver, not yet for FSI-ibm or FSI-mmm solver cases.
    - Automatic coupling with the LS-DYNA structural solver
    - Eulerian fixed mesh or moving mesh (Either type input with *DUALCESE_ELE2D or *DUALCESE_ELE3D cards, or via *MESH cards for a tetrahedral mesh)

- Dual CESE Solver Main Features:

    - CFD solver is based upon the dual CESE (Conservation Element / Solution Element) method that is a new version of CESE technology with enhanced accuracy and robustness
    - Highly accurate shock wave capturing
    - Embedded (immersed) boundary approach or moving mesh (fitted) approach for FSI problems
    - Mesh can be broken up into regions/parts with a different solver per region, with the intent to minimize the region where a moving mesh FSI technique is used, or where an immersed boundary FSI method is used. This capability is intended to help optimize solver performance.
    - Complex fluid equations of state (EOS) are now available through the REFPROP and COOLPROP EOS libraries. This is only supported with *DUALCESE models.
    - Bi-cubic table look-up systems are now available for the REFPROP and COOLPROP libraries to dramatically speed up the evaluation of thermodynamic quantities.

- Dual CESE Solver Applications (Non-exhaustive):

    - FSI problems
    - Shock wave capturing
    - Shock/acoustic wave interaction

- Planned future features of Dual CESE Solver:

    - FSI with material erosion (as done in the *CESE immersed boundary FSI solver)
    - Coupled stochastic fuel spray solver (See *STOCHASTIC keywords)

- Coupling with chemistry (See *CHEMISTRY keywords) solver
- Multi-species transport
- Conjugate heat transfer
- Cavitation model

- ***CONTACT**
  - Fix old issue for *CONTACT_ENTITY type 9 to properly handle rotations when the entity is not centered on (0,0,0).
  - Fix for spotwelds improperly deleted due to rigid body conflict when an IPBACK *CONTACT interface is in effect. The inability of the constraint-based contact interface to tie should not cause the weld to be deleted if the penalty side ties. (MPP only.)
  - Fix SMP *CONTACT_2D_NODE_TO_SOLID. Friction was being ignored.
  - Added a warning that *CONTACT_TIEBREAK_NODES_TO_SURFACE_-ID is not supported for implicit computations.
  - *CONTACT_2D_..._THERMAL is now available in MPP.
  - Fix an SMP bug for ERODING contact of solid elements, which could result in erroneous contact thickness.
  - Fix seg fault or incorrect frictional behavior when using _THERMAL_FRICTION option for *CONTACT_AUTOMATIC_ONE_WAY_SURFACE_TO_-SURFACE and *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE (SMP only).
  - Fix instability when using *CONTACT_TIEBREAK_NODES_TO_SURFACE (SMP only).
  - Fix incorrect *CONSTRAINED_TIE-BREAK behavior when the master node is the last node in the input after sorting.
  - Fix failure to detect contact between beam and shell edge when using *CONTACT_AUTOMATIC_BEAMS_TO_SURFACE and the beam diameter is large compared to the segment size (SMP only).
  - Fix *CONTACT_AUTOMATIC_GENERAL for spot weld beams when using SSID = 0, i.e. all parts included in the contact, and CPARM8 = 2.
  - Implement unloading curve, UNLCID, for options FCM = 2/3 in *CONTACT_RIGID_(OPTION).
  - FTORQ = 1 and 2 (Opt. Card E in *CONTACT), affecting transmittal of moments, are now implemented in SMP for *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK and *CONTACT_AUTOMATIC_ONE_-WAY_SURFACE_TO_SURFACE_TIEBREAK.
  - FTORQ = 2 is now implemented in SMP for *CONTACT_AUTOMATIC_-SURFACE_TO_SURFACE and *CONTACT_ONE_WAY_SURFACE_TO_-SURFACE.
  - Added contact energy density to the intfor database for segment-based (SOFT = 2) contact. This option is invoked by NENG = 1 on *DATABASE_-EXTENT_INTFOR.

- Added support for ORTHO_FRICTION option for segment-based (SOFT = 2) contact. This option is available for all versions, SMP, Hybrid, and MPP with and without the groupable option, and for all supported CONTACT keywords.
- Added options to limit the scope the DPRFAC option of segment-based (SOFT = 2) contact. When DPRFAC is set to a value greater than or equal to 1.0, then DPRFAC is the ID of either a shell set, a segment set, or a part set. When both segments are in the set, then the DPRFAC option is active for that segment pair. If either segment is not in then set, the DPRFAC is inactive. When searching for the set, the search will proceed in the order of shell set, then segment set, and finally part set. The first set that is found with a matching ID will be used. The set attribute DA1 will be read and used as the DPRFAC value.
- Added support for VSF parameter on Card 3 of *CONTACT when using SOFT = 2 contact.
- Added support for FTORQ = 2 (*CONTACT, Card E) for SMP and MPP segment-based (SOFT = 2) contact. Setting FTORQ = 2 adds torque to master nodes to counteract the moment that is created by frictional contact forces due to the contact surface being offset from the mid-plane of the element. This added torque reduces the net moment from contact to zero.
- Added keyword *DEFINE_FRICTION_SCALING which allows shell segments to have friction coefficients on the inner and outer face (SOFT = 2 contact only). Independent scale factors on the inner and outer face scale the nominal friction coefficient. This was motivated by airbags which have liners on the inside of the bag thus creating a smoother (slicker) surface on the inside as compared to the outside.
- Added a new variable EDGEK on Card C of *CONTACT which scales edge-to-edge contact stiffness for SOFT = 2 contact having DEPTH = 5, 15, 25, or 35.
- Added control of thick segment checking using the SNLOG flag on *CONTACT. A thick segment check has been done quietly throughout the history of segment-based (SOFT = 2) contact. A search along the contact surface is made looking for thick pairs that are too close, and those pairs are removed from contact. This is done to prevent possible unstable behavior that could occur if the offsets of nearby thick segments come into contact at a bend in the mesh. When SNLOG = 0 or 1, the thick segment check will be done quietly as before. Set to 2, the check will be done and a warning 21465 will be written to report the segment pairs that are removed from contact. Set to 3, the thick segment check will be omitted.
- Enabled segment-based contact (SOFT = 2) to work with segments attached to cohesive elements with zero volume.
- Enabled *CONTACT_ENTITY to work with thick shell elements. It was previously error terminating in the input phase.

- Enabled SINGLE_SURFACE segment-based (SOFT = 2) contact to output gap values to the intfor output database. Previously, output of the gap values worked only for the non-SINGLE_SURFACE contact types.
- Enabled the SPOTHIN and SWRADF variables on *CONTROL_CONTACT to work when spot welds are modeled by sharing nodes with the shell parts that are welded. This has not worked until now because the search for affected shells was based on tied contact, and tied contact is not needed when nodes are shared.
- Enabled *CONTACT_2D_AUTOMATIC to work with erosion of 2D solid elements that use *MAT_081 or *MAT_082.
- Enabled multiple instances of *CONTACT_2D_AUTOMATIC_TIED to be reliably used in a model.
- Improved the PSTIFF option of segment-based contact to make the segment mass better match the nodal masses. The main effect is to increase the stiffness for segments on the edge of the mesh. For a regular mesh with no mass scaling or lumped nodal masses, the PSTIFF option now calculates the same segment mass as the default method based on segment volume and density.
- Fixed segment-based contact for adaptive remeshing and full deck restarts.
- Fixed the coefficient of restitution option (ICOR) in segment-based contact when it is used along with DPRFAC (Depth of Penetration Reduction FACtor).
- Fixed *CONTACT_2D_FORCE_TRANSDUCER with both slave and master sides defined when they are used with *CONTACT_2D_AUTOMATIC in MPP. A memory clobber was likely and also output was incorrect.
- Fixed MPP wear calculation in segment-based contact. A mix-up of ID numbers means that the wear was only working when contact interfaces were numbered sequentially from 1.
- Fixed the THERMAL option of *CONTACT in MPP when some MPP partitions do not participate in all the contact definitions.
- Fixed *CONTACT_2D_AUTOMATIC_TIED when used with selective mass scaling, which is made active by setting IMSL = 1 and DT2MS < 0 on *CONTROL_CONTACT. The problem occurred when selective mass scaling was applied to the nodes that are tied. Prior to the change, kinematic constraints were failing.
- Fixed MPP *CONTACT_2D_AUTOMATIC when beams were in contact, or when solids were used with nonzero values of solid surface offset(s) (variables SLDSOS or SLDSOM). Both cases could lead to using invalid memory space and a segmentation fault.
- Add new option ICNEP = 1 to *DEFINE_FRICTION with which those lines with non-existent parts or part sets will be ignored.
- Variable ENGOUT on *CONTROL_OUTPUT added to write minimum (wherever occurring) contact energy densities to d3plot (MORTAR contact only).
- Variable NENG on *DATABASE_EXTENT_INTFOR added to write contact energy densities to intfor (MORTAR contact only).

- Variable PENOUT on *CONTROL_OUTPUT added to write maximum (wherever occurring) penetrations across interfaces to d3plot (MORTAR contact only).
- Variable NPEN on *DATABASE_EXTENT_INTFOR to write absolute and relative penetrations to intfor (MORTAR contact only).
- Variable NTWELD on *DATABASE_EXTENT_INTFOR writes user tied weld history variables to intfor file (MORTAR contact only).
- Variable VC on *CONTACT_... supported for MORTAR contact.
- Added variable TDPEN on *CONTACT_2D_AUTOMATIC_SURFACE_-TO_SURFACE_MORTAR. This is the analogue to interference option MPAR1 for IGNORE = 3 in 3D automatic MORTAR contact.
- Mortar contact supports erosion, that is, any solid segment or shell edge segment belonging to parts involved in the contact definition will become active in the contact when exposed to the environment due to erosion.
- IGNORE = 4 on *CONTACT_..._MORTAR supports a curve of relative interference reduction vs. time. When MPAR1 < -1.0, |MPAR1| is the curve ID.
- *LOAD_THERMAL_CONSTANT is supported in mortar contact, in those situations where the contact properties depend on temperature.
- In eigenvalue analysis, nonzero mortar contact stress will not influence rigid body modes, i.e., you should see the proper rigid body modes among the available output.
- TEMP < 0 on *CONTACT_..._MORTAR_TIED_WELD calls a user tied weld interface, allowing implementation of "arbitrary" tied condition in weld simulations.
- *CONTACT_TIED_..._THERMAL supported for (spotweld) beams.
- The THERMAL option can now be used for the following tied contacts:

  *CONTACT_TIED_NODES_TO_SURFACE_OFFSET_THERMAL

  *CONTACT_TIED_NODES_TO_SURFACE_CONSTRAINED_OFFSET_THERMAL

  *CONTACT_TIED_NODES_TO_SURFACE_THERMAL

  *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_OFFSET_THERMAL

  *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET_THERMAL

  *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_CONSTRAINED_OFFSET_THERMAL

  *CONTACT_TIED_SHELL_EDGE_TO_SURFACE_THERMAL

  BUT the slave set type can NOT be a node set, at least not yet.
- Now implemented for MPP are *CONTACT_AUTOMATIC_SURFACE_-TO_SURFACE_TIEBREAK and *CONTACT_AUTOMATIC_SURFACE_-TO_SURFACE_TIED_WELD.

- Segment-based contact (SOFT = 2) now supports groupable and non-blocking features.
- Option FTORQ for transmittal of moments across the contact interface is implemented in the groupable node-to-surface contacts, surface-to-surface contacts, and tied contacts.
- *CONTACT_AUTOMATIC_GENERAL supports options SOFT = 1 and SRNDE for shell edge treatment.
- *CONTACT_AUTOMATIC_GENERAL supports options SOFT = 1 and SRNDE for the shell edge treatment.
- A new orthogonal friction model is developed for the constraint and penalty *RIGIDWALL_ORTHO. A RIGIDWALL_ORTHO with equal friction coefficients in all directions now behaves as a regular RIGIDWALL with a single friction coefficient.
- *CONTACT_RIGID_TO_RIGID is enhanced to support unloading curve, FCM = 1/2/3.
- *CONTACT_CONSTRAINT_NODES_TO_SURFACE is enhanced to support variable PENSF on MPP Card 2 of *CONTACT and recover contact forces that existed before solid remeshing associated with 3D solid adaptivity.
- *CONTACT_AUTOMATIC_GENERAL is improved to stabilize short beams with small rotational inertias.
- ERODING contacts are now supported in full deck restarts.
- *CONTACT_AUTOMATIC_BEAMS_TO_SURFACE supports beam to shell_edge contact and detects the contact between shell edges and phantom nodes.
- A new built-in variable IDRFLG added to *DEFINE_CURVE_FUNCTION, *DEFINE_FUNCTION. IDRFLG carries a value of 1 during dynamic relaxation and 0 in the transient phase.
- Multiple *DEFINE_FRICTION tables with the same ID are merged.
- Effect of variable I2D3D is correctd in MPP to avoid duplicated contact segments when the attached shell and solid are not assigned to the same processor.
- ERODING contact in MPP is fixed to compute the stiffness of newly-exposed solid nodes and segments and apply the optional thickness SLDTHK to the newly-exposed solid segments.
- The contact thickness in *PART_CONTACT is applied to *CONTACT_AUTOMATIC_GENERAL.
- *CONTACT_AUTOMATIC_GENERAL is corrected by reducing the contact stiffness to half.

• ***CONSTRAINED**

- Fix ordering issue during initialization of *CONSTRAINED_INTERPOLATION in MPP that could have resulted in incorrect "colinear" warnings or possibly deadlock

- Fix MPP message passing error that could occur if a node involved in *CONSTRAINED_SHELL_TO_SOLID is shared between more than 2 processors.
- Enhanced processing of an exactly singular constraint matrix for *CONSTRAINED_INTERPOLATION.
- Corrected error in the checking for massless *CONSTRAINED_INTERPOLATION constraints in single precision.
- Enhance *CONSTRAINED_INTERPOLATION in the explicit solver so the linear algebra of the constraint equation processing always operates in double precision to reduce loss of precision in single precision executables.
- Add implicit support to *CONSTRAINED_SHELL_IN_SOLID.
- *CONSTRAINED_BEAM_IN_SOLID:
  - Add support for IDIR = 1. This means the beam is allowed to freely slip longitudinally inside solid elements.
  - Add implicit support for the AXFOR option. Debonding between beams and solids may now be simulated using the implicit solver.
  - Add implicit support for CONSTRAINED_BEAM_IN_SOLID_PENALTY.
- Fix incorrect *CONSTRAINED_INTERPOLATION motion of the dependent nodes if the number of constrained dof is less then the total dof, i.e., DDOF.ne.123456.
- Fix ineffective PNODE in *CONSTRAINED_NODAL_RIGID_BODY when PNODE < 0.
- Fix kinetic energy dependence on MPP core count when using *CONSTRAINED_JOINT_(OPTION) with LMF = 1 in *CONTROL_RIGID.
- Added swforc file output for welds modeled by *CONSTRAINED_GENERALIZED_WELD. Output values are the brittle failure forces, axial and shear, and the failure function.
- Improved temperature-dependent failure curve option of spot welds defined by * CONSTRAINWED_SPOTWELD. Curve interpolation was using the wrong segment of the curve. Also, changed swforc output to show zero axial and shear forces after a weld has failed.
- Add correct torsion to SPR3 (*CONSTRAINED_INTERPOLATION_SPOTWELD with STIFF4 > 0).
- Small modification for PIDVB < 0 of *CONSTRAINED_SPR2/SPR3: Beams get deleted after failure (they were just separated before).
- Fix for *CONSTRAINED_SPR2/SPR3: part id of internally generated beams has to take Nodal Rigid Bodies into account.
- Ctl. Add error trap if INTPERR = 1 is set on *CONTROL_SHELL and data interpolation would be done with *INITIAL_STRESS_SHELL.
- *CONSTRAINED_JOINT_STIFFNESS_CYLINDRICAL now available to model cylindrical/revolute connections with play
- *CONSTRAINED_COORDINATE now supports part set ID in addition to part ID.

  - *CONSTRAINED_COORDINATE: When IDIR is negative, the constraint is applied at the nearest node, not at the coordinate.
  - For nodes which are not attached to any elements, turn off the automatic global constraint normally applied to such nodes if those nodes are involved in *CONSTRAINED_MULTIPLE_GLOBAL.

- ***CONTROL**

  - *CONTROL_REFINE_... with NTOTRF = -1: Refine solids or shells with elements dynamically added during the run (as opposed to NTOTRF > 0 for which the user must estimate the number of child elements required for the refinement and these elements are added during the initialization).
  - *CONTROL_REFINE_SHELL: Support contact for the case of NTOTRF = -1, i.e., dynamically add contact segments due to mesh refinement.
  - In *CONTROL_REFINE_... keywords:

    - If CRITRF < 0, reverse the refinement conditions
    - If CRITRM < 0, reverse the coarsening conditions

  - *CONTROL_STAGED_CONSTRUCTION:

    - Added support for *ELEMENT_SOLID ELFORMs 4, 13, 16, 17.
    -
    - Added new input field IDYNAIN to suppress output of dynain file at end of each stage. This can be done for all stages using IDYNAIN on *CONTROL_STAGED_CONSTRUCTION, or for individual stages using IDYNAIN on *DEFINE_CONSTRUCTION_STAGES.
    - Fix bug affecting *MAT_021 and *MAT_021 solids with Staged Construction (*DEFINE_STAGED_CONSTRUCTION_PART). When a part became active, the stresses were wrongly initialized. Bug affected solid elements only.

  - *CONTROL_PORE_FLUID: Fixed bug in internal energy calculation for solid element types with more than one integration point. This affected only models containing *CONTROL_PORE_FLUID.
  - *CONTROL_ADAPTIVE: Small change to 2d adaptive remeshing that should improve behavior for meshes with bad boundaries.
  - Fix problem with 2d adaptivity and boundary merging. Some boundary points between materials weren't merged in some cases, depending on where the program thought the 2d contours started. This only applies for *CONTROL_ADAPTIVE adpopt = 8 with mmm2d = 1.
  - Implement idrflg = -3 for *CONTROL_DYNAMIC_RELAXATION which invokes the version R7 (and earlier) implementation of idrflg = 3. Now idrflg = 3 means the parts not included in the part set are eliminated from the computation, and idrflg = -3 means all parts are included in the explicit

computation but only those in the part set are included in the distortional-kinetic-energy-based convergence check in the dynamic relaxation phase.

- Ignore *CONTROL_SPOTWELD between parts that are inactive during implicit dynamic relaxation phase.
- Fix a bug whereby SHLTRW in *CONTROL_SHELL was ignored when ISTUPD > 0.
- Add variable MLKBAG to *CONTROL_OUTPUT to include accumulated airbag gas leakage in abstat.
- Fix wrong pointer used for section id/properties when generating part for visualization of rigidwall during adaptivity. This cause error message KEY+137.
- *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS:
  - Add a stiffness option (ISTFNS) to *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS.
    EQ.1: no stiffness is added.
    EQ.2: only spin softening is added which keeps stiffness symmetric.
    EQ.3: both spin softening and gyroscopic effects are added which makes the matrix skew-symmetric.
    Default is 3.
  - Add the whirling direction to the output eigout.
- DISPLAY option in *RIGIDWALL_PLANAR will be ignored for stationary rigid walls when SKIPRWG = 1 in *CONTROL_CONTACT.
- Added options to the ERODE parameter on *CONTROL_TIMESTEP. The original options were 0 and 1 where 1 causes solid and thick shell elements to be eroded if their time step dropped below TSMIN*DT0 where TSMIN is specified on *CONTROL_TERMINATION, and DT0 is the solution time step that is calculated in the first cycle. The new options are to check and delete shell elements and beam elements having small time steps. The control of shells is in the 'tens' place and the beams is in the 'hundreds' place. The valid options for ERODE are now 1, 10, 11, 100, 101, 110, and 111. For ERODE = 111, beams, shells, and solids are all checked. For ERODE = 11, shells and solids are checked. The original options of 0 and 1 still behave exactly as they did.
- Added support of *CONTROL_SUBCYCLE for segment-based (SOFT = 2) contact. Contact forces are calculated for all nodes of a segment pair only if any node of the pair is due to have a force calculated.
- Fixed some issues so that solid elements and thick shell elements will work as advertised with respect to DTMIN on *CONTROL_TERMINATION, ERODE on *CONTROL_TIMESTEP, and PSFAIL on *CONTROL_SOLID. These variables control whether we terminate the analysis or fail elements when their time step becomes small or their volume becomes negative.

- Add an option TET13V on *CONTROL_SOLID to choose between the efficient and a more accurate version of the tet type 13 implementation.
- Lagrangian multiplier joints, LMF = 1 on *CONTROL_RIGID, are supported for consistent mass scaling (RBSMS = 1 on *CONTROL_RIGID).
- *CONTROL_ADAPTIVE:Fix bug in 3D tet adaptivity when there are rigid parts with a tetrahedral mesh and ELFORM = 10. Before the bug fix, the rigid parts with ELFORM = 10 were not properly written out during adaptive restart, resulting in an error termination.
- *CONTROL_REMESHING: 3D tet adaptivity is able to handle the remeshing of multiple bodies in a single adaptive parts. Preserving of feature lines is controlled by the variable SEGANG.

- **Discrete Element Method**

  - New keyword *DEFINE_DE_FLOW_DRAG to support DES interacting with external flow field.
  - Add a moving system with birth/death times in *DEFINE_DE_ACTIVE_-REGION.
  - Fix bug of generation failure and uneven distribution of injected DES particles (*DEFINE_DE_INJECTION) when the radius of injected DES particles is very small.
  - Fix segmentation fault when using simple and small restarts in DEM.
  - Add the keyword *DEFINE_DE_COHESIVE to enable the capillary force in specified DES node set or parts.
  - Add the keyword *INITIAL_STRESS_DES to initialize stress in discrete element spheres (DES).
  - Output the DES initial stress and geometry information into dynain_ini and dynain_geo files (FSPLIT = 1 in *INTEFACE_SPRINGBACK_LS_DYNA).
  - Fix incorrect behavior when using *DEFINE_DE_ACTIVE_REGION with itype = 1, i.e. BOX, with *DEFINE_TRANSFORMATION. The box was not rotated.

- **Element Free Galerkin (EFG)**

  - *SECTION_SOLID_EFG: Volume of 6-noded solid elements using element formulation 41 (EFG) was calculated incorrectly. That bug is now fixed.

- ***ELEMENT**

  - For triangular shells with thickness stretch (*SECTION_SHELL, ELFORM = 27) the option IDOF = 1 is made available allowing a continuous thickness field between them and quadrilateral shells with thickness stretch, ELFORM = 25,26.
  - For the Belytschko-Tsay shell with thickness stretch (*SECTION_SHELL, ELFORM = 25) an improved representation of stresses over the thickness may be obtained with the new option IDOF = 11 instead of IDOF = 1 (for

continuous thickness fields) or with the new option IDOF = 12 instead of IDOF = 2 (for discontinuous thickness fields).

- Changed *DEFINE_ELEMENT_EROSION to be more compatible with the original FE options and added support for IGA elements via *DEFINE_ELEMENT_EROSION_IGA.
- Added calculation of implicit tangent for shell formulation 46, a 2D cohesive element.
- *ELEMENT_DIRECT_MATRIX_INPUT:
  - Enhanced the reading of superelement files to support connections using nodes, scalar nodes, internal dofs, and the label of "0" for scalar nodes.
  - Added MPP logic for collecting nodes associated with superelements for use in decomposition for the reading of binary formatted files for superelements.
  - Fix an error associated with using superelements in MPP explicit. Two arrays should have been initialized to zero, but the bug did not manifest itself until the user used 12 processes. The arrays are now properly initialized to zero.
  - Corrected the initialization of superelements involving nodes with SPC constraints.
- *ELEMENT_SEATBELT_PRETENSIONER: make type-8 pretensioner available for 2D belt.
- *ELEMENT_SEATBELT_SENSOR: add type-5 sensor for tracing retractor payout.
- Restructure nodal volume data communication for solid tet formulation 13 for better parallel performance.
- Added new more conservative time step calculation (independent of shell thickness) for improved stability in cohesive shell formulation 29.
- Fix incorrect stress output to d3plot and ASCII files when using tetrahedron solid types 10 and 13 with orthotropic materials and when CMPFLG = 1 in *DATABASE_EXTENT_BINARY.
- Fixed a bug that caused axisymmetric elements (shell ELFORMs 14,15) to instantly fail when DTMIN > 0 in *MAT_ADD_EROSION.
- Shell elements with *MAT_NULL can be deleted by TIME in *DEFINE_ELEMENT_DEATH_SHELL.
- Added a check for invalid ELFORM values on *SECTION cards. Invalid values were leading to terminations without clear error termination messages.
-
- Added an optional modification to the time step calculation that is done for solid elements. See variable IHDO in *CONTROL_TIMESTEP. Setting IHDO to 1 causes the time step calculation to be modified such that it does not have a discontinuity between expansion and compression. With the default

time step calculation, the solution time step can jump up or down about 6% when an element controlling the time step switches from compression to expansion. The size of the jump depends on the linear bulk viscosity coefficient. These jumps may not occur if multiple elements are controlling the time step as it is unlikely for all to be expanding at the same time, so the smaller time step is used consistently. Setting IHDO to 1 prevents this discontinuity from occurring.

- Added support for shell form 23 in output file eloutdet (*DATABASE_-ELOUT). When shell form 23 is used, the code can now output stress and strain at 2x2 or 3x3 integration points and extrapolate to all 8 nodes.
- Added a bulk viscosity option for thick shell (TSHELL) elements (See variable TSTYPE on *CONTROL_BULK_VISCOSITY).
- Enabled plane strain and axisymmetric elements (*SECTION_SHELL ELFORM 13, 14, 15) to include stress initialization by *INITIAL_STRESS_-SHELL for materials that use equations of state.
- Enabled solid spot weld assemblies to use 8 node and 6 node cohesive elements with *MAT_240. There is no assembly failure calculation, but the resultant forces and moments are calculated and can be output to the swforc file. This works with solid element formulations 19, 20, 21, and 22.
- Enabled variables CDL and TDL on *SECTION_DISCRETE to be used when one of the discrete beam nodes is constrained by an SPC.
- Improved the accuracy of pressure loading on fully integrated, volume-weighted axisymmetric solid elements (ELFORM = 15/NIP = 4 on *CONTROL_SHELL), particularly along the axis of symmetry.
- Improved the post-failure behavior of integrated beam elements by adding an option (NAUDP on *SECTION_BEAM) to update the neutral axis when one or more integration points fail so that nodal moments are correctly calculated. This option is available for the integrated beam formulations 1, 9, 11, and 14 when used with *MAT_003, *MAT_098, *MAT_100, *MAT_124, and *MAT_158.
- Improved mass scaling of beam element formulations 4 and 5. A dimensional problem was causing erratic behavior which could cause unstable behavior.
- Fixed bug affecting simultaneous use of ERODING contact with cohesive pentahedral solids, which through use of ESORT = 1, are changed to formulations 21/22. Prior to this fix, this combination of keywords could have lead to error termination due to negative volume.
- Add check if two or more parts use solid element formulation 13 (tetrahedron with pressure averaging) and those parts share some nodes. Write a warning in that case because such a condition could lead to instabilities.
- COHOFF on *SECTION_SOLID allows for adjusting the reference layer for cohesive element formulations 20 and 22.
- Fixed bug in plane strain, 8-node shell formulation 55 to show correct stresses when 4 in-plane points are required for d3plot output. Restored writing out stress components at all integration points in elout. Shell

formulation 55 is for implicit analysis only and requires 4, 9 or 16 integration points (default 4) since it is meant to simulate the singular stress field around a crack tip.
  - For thick shell (TSHELL) formulations 1 and 2, fixed incorrect stress output in elout when used for composite materials and CMPFLG = 1.

- ***EM (Electromagnetic Solver)**
  - *EM_SOLVER_BEM: Option PRECON = 4 (line 1 field 4), LLT factorization of the BEM matrix, can now be used in MPP. It could only be used in serial/SMP up to now.
  - Eddy current / inductive heating solver:
    - Addition of monolithic solver for FEM+BEM solve (higher time steps and improved stability for ferromagnetic materials) (*EM_SOLVER_FEMBEM).
    - Optimization of memory/CPU cost in BEM solve.
    - Improved accuracy of solution on prismatic elements.
    - Addition of LLT factorization as a preconditioner for BEM solver in MPP (*EM_SOLVER_BEM).
    - Added 2D and 2D axi capability for RSW and other applications (*EM_CONTROL and *EM_MAT_004).
    - Support of eroding conductors (*EM_CONTROL_EROSION and *EM_MAT_...).
    - Improved accuracy of thermal coupling for tetrahedral elements.
  - EM contact:
    - Optimize contact search and robustness => reduced calculation times and memory cost
    - Added capability in RSW to specify electric contact law (*EM_CONTACT_RESISTANCE).
  - Battery Module Release:
    - Equivalent circuit model (ECM): Implementation of Randles circuits as simplified electro-chemistry model
    - Dynamic EM-mechanical-thermal coupling for external and internal short circuit (battery crush) (see in particular *EM_RANDLES_SHORT).
    - Bath loading, coupling mono-bi and bath in same domain.
    - Purkinje network (*EM_EP_MECACOUPLING).
    - Current defined stimuli (*EM_EP_TENTUSSCHER_STIMULUS).

- ***FATIGUE**

- Added using *DEFINE_TABLE to define SN curves for multiple mean stress.
- Added new mean stress correction method (METHOD = 5): Morrow equation (SN curve).
- Extended thick shell element fatigue analysis (including multiaxial) to MPP.
- Extended solid element fatigue analysis (including multiaxial) to MPP.
- Extended shell element fatigue analysis (including multiaxial) to MPP.
- Fixed a bug in running *FATIGUE_SUMMATION. This function is now working as a standalone module. Previously it was used inside fatigue analysis.
- Added warning message MSG_FTG+5 in case the calculated stress value is higher than the stress of the first point on SN curve, to remind user to check the SN curve or stress calculation. If the stress value is higher than the stress of the first point on SN curve, extrapolation is used and this may not be accurate, especially if the first point is the UTS (Ultimate Tensile Strength).
- Added a parameter (DMGMIN in *FATIGUE) to allow user to define base damage ratio for elements with zero stress or strain.
- Added a new option _D3PLOT to allow running time domain fatigue analysis based on d3plot.
- Implemented *FATIGUE_FAILURE to remove failed elements from model.
- Implemented *FATIGUE_MULTIAXIAL to run multiaxial fatigue analysis.
- Implemented *FATIGUE_LOADSTEP to run fatigue analysis with multiple load steps.
- Added warning message MSG_FTG+4 if fatigue properties are not defined for the current element under fatigue analysis.
- Implemented definition of initial fatigue damage ratio by part or part set (PID/SID in *INITIAL_FATIGUE_DAMAGE_RATIO).
- Added semi-log SN curve definition. This can be used both in time domain and frequency domain fatigue analysis.
- Added exposure time TEXPOS for fatigue analysis, in case it is different from endtim in *CONTROL_TERMINATION.
- Added an option D3PLOT to *INITIAL_FATIGUE_DAMAGE_RATIO to allow using damage variables from transient preload cases as initial fatigue damage ratio.
- Fixed a bug in defining mean stress correction methods. The UTS/yield strength should be defined on material model, not part ID, according to the Manual.

- **Forming Analysis**

  - *DEFINE_BOX_NODES_ADAPTIVE now supports more than 2 levels of mesh refinement for tube adaptivity. Maximum level is specified by variable LEVEL.

- ONESTEP simulations with triangular elements used to produce misleading effective plastic strain results in regions with high curvatures. This is now fixed.
- Fix bug in ONESTEP method that produces false minor and major strain in the output file onestepresult
- Fix bug in *CONTROL_FORMING_ONESTEP, OPTION = 6.
- Fix MPP bug for ONESTEP output files repositioned.k and onestepresult.
- Fixed the problem of lacking temperatures at the lancing line in MPP lancing simulations.
- Fixed a memory access problem which resulted in error termination of simulations of unflanging process.
- *CONTROL_FORMING_MAXID is supported by 3d tet adaptivity (*CONTROL_ADAPTIVE).
- *INCLUDE_STAMP_PART: Improvement to support *DEFINE_TRANSFORMATION.
- *DEFINE_CURVE_TRIM_3D: Projection of curve to blank is now allowed when the trimming curve is far away from the blank. This was previously disallowed.
- *CONTROL_FORMING_ONESTEP:
  - Add a CPU time report for onestep method to record the time spent on initial unfolding.
  - Allow user to output specific part to the file of 'onestepresult'.
  - The fracture curve LCSDG can be automatically found from *MAT_ADD_EROSION.
- *CONTROL_FORMING_PRE_BENDING(_LOCAL): New option allows user to define a vector based on a local coordinate system. If the LOCAL option is used, then coordinate system ID is input on Card 2.
- *ELEMENT_LANCING: Allow multi-curve lancing with different starting times. Previously, only one starting time was allowed. With this new feature, all the lancing curves will be used to cut at the same time, but the nodes will not be able to separate until the lancing time is reached.
- *CONTROL_FORMING_AUTOPOSITION_PARAMETER
  - Support 2D elements.
  - When CID is a negative value, its absolute value refers to a vector ID (*DEFINE_VECTOR) rather than a coordinate system ID.
- *DEFINE_FORMING_ONESTEP_MASTER: This new keyword allows a second blank to be welded to a master blank. The two blanks are connected by using spot welds.
- Add keyword *CONTROL_FORMING_TRIM_SOLID_REFINEMENT to homogeneously refine elements along a trim curve. Supports *DEFINE_CURVE_TRIM_2D and *DEFINE_CURVE_TRIM_3D.

- *DEFINE_CURVE_TRIM_2D allows refinement of a sandwiched part along trimming curves. The sandwich core comprised of multiple layers of solid elements can now be refined. In addition, the trimming curve can be in any direction (2D trimming only).
- *CONTROL_FORMING_ONESTEP_ORTHO uses 2 nodes in the final part to define the rolling direction of the anisotropic material.
- *DEFINE_FIBERS:
  - User can perform carbon fiber simulation without the matrix (fibers only).
  - No need to define material property for the matrix.

- ***FREQUENCY_DOMAIN**
  - Correction to rotational inertia for *FREQUENCY_DOMAIN_SSD.
  - *con Extend logic for implicit constraint handling to tied contact with _CONSTRAINED_OFFSET to skip any SPC constraints on the rotational dofs of the SPC slave nodes. We already have such logic to skip the constraints on the translational dofs for the slave nodes for all tied contacts.
  - *loa: Correct the issue where use of *LOAD_BODY is applied to a model with rigid bodies. If the vector of nodes having the load applied belong to a rigid body, then the vector of elemental stiffness matrices is null. So logic was added to skip the call to the implicit matrix assembly subroutine for that case.
  - *DATABASE_FREQUENCY_BINARY_*{OPTION}*:
    - Added a new option SUMMATION to *DATABASE_FREQUENCY_BINARY_D3PSD and *DATABASE_FREQUENCY_BINARY_D3RMS to get total PSD and RMS response from multiple loading resources.
    - Update to support *INCLUDE_TRANSFORM.
    - Added NFSPACE = 3 to define output frequency as eigenfrequencies.
    - Added an option (BINARY = 2) to dump out individual mode response to d3spcm (*FREQUENCY_DOMAIN_RESPONSE_SPECTRUM).
  - *DATABASE_FREQUENCY_ASCII_*{OPTION}*:
    - Update to support *INCLUDE_TRANSFORM.
    - Added NFSPACE = 3 to define output frequency as eigenfrequencies.
  - *FREQUENCY_DOMAIN_ACOUSTIC_BEM:
    - Update to support *INCLUDE_TRANSFORM.
    - Added radiated power computation to Rayleigh method.
  - *FREQUENCY_DOMAIN_ACOUSTIC_FEM:

- Update to support *INCLUDE_TRANSFORM.
- Update to use mixed elements (tet, hex and pentahedrons).
- Fixed a bug in running with combined boundary conditions (transient vibration and impedance).
- Added a flag (RESTRT) to run restart from transient analysis results. LS-DYNA will read an existing binary database to extract the velocity boundary condition.
- Added defining impedence boundary condition by *DEFINE_CURVE.
- Added a flag "idump" in *FREQUENCY_DOMAIN_ACOUSTIC_-FEM_EIGENVALUE to dump out acoustic stiffness and mass matrices.

◦ *FREQUENCY_DOMAIN_FRF:

- Update to support *INCLUDE_TRANSFORM.
- Enabled using structural damping (*DAMPING_STRUCTURAL).

◦ *FREQUENCY_DOMAIN_LOCAL:

- Implemented this keyword to run frequency domain analysis on part of model.
- Implemented options SOLID_SET, SHELL_SET, BEAM_SET and TSHELL_SET.
- Implemented option EXCLUDE to exclude specified sets of elements or nodes.
- Enabled using this feature on response spectrum analysis (*FREQUENCY_DOMAIN_RESPONSE_SPECTRUM).

◦ *FREQUENCY_DOMAIN_MODE:

- Implemented option LOAD_PROJECTION, to select the normal modes based on load projection ratio.
- Implemented option MODAL_COEFFICIENT to select the participating modes based on relative significance of the modal coefficients.
- Enabled this keyword to FRF (*FREQUENCY_DOMAIN_FRF) and SSD (*FREQUENCY_DOMAIN_SSD).

◦ *FREQUENCY_DOMAIN_RANDOM_VIBRATION:

- Update to support *INCLUDE_TRANSFORM.
- Added parameters ICOARSE and TCOARSE in *FREQUENCY_DOMAIN_RANDOM_VIBRATION to reduce points in PSD curve to save CPU time.
- Added parameter TOL_INTG to allow user to define the relative tolerance for integration. The default is 1.e-5.
- Updated PSD curve input so that loading types 9 (base velocity), 10 (base displacement), 11 (enforced acceleration), 12 (enforced velocity),

and 13 (enforced displacement) can be defined through time history curve.
- Added structural damping (*DAMPING_STRUCTURAL) to random vibration analysis.

- *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM

  - Enabled running response spectrum analysis with CASE.
  - Added beam elements stress/strain components in elout_spcm when applicable. Previously only resultant forces and moments are provided.
  - Fixed a bug in reading d3eigv family files for multipoint response spectrum analysis.
  - Update to support *INCLUDE_TRANSFORM.
  - Fixed a bug in running response spectrum with base excitation defined in time domain.

- *FREQUENCY_DOMAIN_SSD

  - Fixed a bug in reading d3eigv for restart option (RESTMD = 1) when both 4-noded shells and 8-noded shells (like ELFORM 23) are present in the model.
  - Update on SSD to use the current geometry of model if SSD is performed after dynamic analysis. Previously SSD is always performed based on the initial geometry.
  - Implemented radiated acoustic power computation with Rayleigh integral.
  - Added radiation efficiency computation for ERP (*FREQUENCY_DOMAIN_SSD_ERP).
  - Enabled sense switch for direct SSD (ctrl-c).
  - Updated *FREQUENCY_DOMAIN_SSD_FATIGUE by adding scale factor for each load. Added MSG_KEY+1232, MSG_KEY+1233 to remind user about the changes.
  - Added torque and rotational dof excitation for SSD (VAD = 8, 9, 10, 11), with local damping (DMPFLG = 1).
  - Update to support *INCLUDE_TRANSFORM.
  - Extended *FREQUENCY_DOMAIN_MODE_LOAD_PROJECTION to the computation of nodfor_ssd, nodfor_ssd and elout_ssd.
  - Added option _FREQUENCY_DEPENDENT to run direct SSD with frequency dependent material properties.
  - Added option _FRF to run SSD with FRF setting. Changed the output filename to d3frf. With this option, one can get fringe plot for FRF (previously only xyplot results were given by FRF).
  - Added rotational dof output in nodout_ssd, when applicable.

- Added new loading options vad = 12, 13, 14 for direct SSD (*FREQUENCY_DOMAIN_SSD_DIRECT), for prescribed velocity, acceleration, and displacement.

- ***ICFD (Incompressible Fluid Solver)**

  - Navier Stokes solve/free surface flows:

    - *ICFD_BOUNDARY_FSWAVE: Added Stokes wave of fifth order, solitary wave model and irregular waves using JONSWAP spectrum. Add a complete check of wave generation settings and the validity of the parameters within each wave theory. *ICFD_BOUNDARY_PRESCRIBED_VEL can be used jointly to model the interaction of any wave theory and any prescribed velocity at inlets.
    - *ICFD_BOUNDARY_PERIODIC: Addition of periodic, reflective and sliding mesh boundary conditions.
    - *ICFD_CONTROL_TIME: Maximum and minimum time step definitions can now be made functions of time.
    - *ICFD_CONTROL_OUTPUT_SUBDOM: Supports output to d3plot (before only supported VTK and VTU formats).
    - *ICFD_DEFINE_SOURCE: Allows the user to specify a generic volumetric force entering the velocity solve. Can be useful in various applications, for example, when coupling ICFD with the EM solver.
    - *ICFD_INITIAL: *DEFINE_FUNCTION can now be used to define an initial velocity profile.
    - *ICFD_INITIAL_LEVELSET: Extended the number of initial surfaces which can be generated; boxes and spheres are now possible on top of surface plane.
    - *ICFD_MAT: Variable CA specifies the contact angle for the liquid-vapor interface, for use in surface tension problems.
    - *ICFD_MODEL_POROUS: Added new porous models, pmid = 11 and pmid = 10, for parachute fluid flow. To be used with thin shells (See *MESH_EMBEDSHELL).
    - Bug Fix: Removed pressure outlet smoothing technique after noticing it could cause disruption of results. Results of internal flow problems might be especially affected.
    - Bug fix: Added more digits to icfd_fluidmesh.key output to avoid round off errors.

  - Turbulence models:

    - *ICFD_CONTROL_TURBULENCE, SUBMOD = 2: Added LES Dynamic turbulence model originally proposed by DK Lilly (1991) with localization on coefficient CS by Piomelli and Liu (1995).

  - *ICFD_DEFINE_TURBSOURCE: new keyword which allows addition of turbulent source terms for RANS models.
  - *ICFD_CONTROL_TURB_SYNTHESIS: now supports *DEFINE_-FUNCTIONs.
  - Bug fix: Fixed incorrect calculation of stresses in laws of the wall.

- FSI, DEM coupling and remeshing:

  - *ICFD_CONTROL_ADAPT: Variable KIS may be used to diminish the excessive coarsening that can potentially occur when remeshing by keeping the nodal mesh size distribution of the initial volume mesh.
  - *ICFD_CONTROL_ADAPT: Variable VAR gives the user more control on the error calculation triggering a remesh.
  - *ICFD_CONTROL_DEM_COUPLING: Variable MAXVEL is a 'max velocity' term for the DEM force calculation intended to avoid spurious high velocities which could cause the fluid solver to diverge.
  - *ICFD_CONTROL_DEM_COUPLING: Variable DTYPE = 1 invokes option to use Morrison formula for Drag calculation to apply on DEM particles.
  - *ICFD_CONTROL_FSI: Added control on the number of FSI fluid subiterations. This avoids the sometimes unneeded excessive number of FSI subiterations when the fluid and very light structures (like parachutes) develop a resonance-like mode inside the FSI subiterations (coupling iterations). See variable NSUB.

- Thermal and Conjugate heat transfer (CHT):

  - *ICFD_BOUNDARY_FLUX_TEMP: Modified boundary condition so that the prescribed value is equivalent to quantity prescribed in BOUNDARY_HEAT_FLUX.
  - *ICFD_BOUNDARY_CONJ_HEAT: Added option to impose a temperature drop between solid and fluid (see variables VAL and SFLCID). Also added an alternative penalty based (mortar) coupling method (CTYPE = 1). This can be beneficial in cases with large mesh size discrepancies but is typically associated with higher solve times.
  - *ICFD_BOUNDARY_CONVECTION_TEMP: Added keyword to prescribe a convection boundary condition for thermal problems.
  - *ICFD_CONTROL_CONJ: TSF feature added in ICFD solver for thermal and CHT problems.
  - *ICFD_CONTROL_TURBULENCE: Added thermal law of the wall; see variable TWLAW.
  - Bug fix: Previously, all FSI surfaces were taken into account for the thermal and fluid coupling. Now only the boundaries to which the keyword

    - *ICFD_BOUNDARY_CONJ_HEAT is applied will be coupled to the solid.
    - Bug fix: HCLCID and TCLCID in *ICFD_MAT were incorrectly read in MPP.
    - Bug fix: icfd_thermal.dat now keeps getting output in CHT cases after the Navier-Stokes equations have reached steady state.

- **Implicit (Mechanical Solver)**

  - New LSOLVR options in *CONTROL_IMPLICIT_SOLVER:

    - 17 (SMP only): iterative solver with a preconditioner based on a local incomplete factorization.
    - 27 (MPP only): ditto.

  - *CONTROL_IMPLICIT_ORDERING: Made various improvements to LS-GPart (ORDER = 4).

    - Significant improvements in MPP;
    - Added hybrid parallelism (OpenMP);
    - Added a progress report ("heartbeat") for large problems.

  - New option where ILIMIT in *CONTROL_IMPLICIT_SOLUTION can vary with time. If ILIMIT < 0, |ILIMIT| is taken as the curve ID which defines the iteration limit between automatic stiffness reformations (ordinate) vs. time (abscissa).
  - Implemented an approximate constitutive matrix for implicit analysis of *MAT_187/*MAT_SAMP-1.
  - New implicit feature for automatic generation of solution key points through examination of load curves for points of particular interest, e.g., times at which maximum load occurs. Set DTMIN < 0 in *CONTROL_IMPLICIT_AUTO to invoke this option.
  - Enhanced the performance of the output to d3* files for *CONTROL_IMPLICIT_EIGENVALUE and *CONTROL_IMPLICIT_MODES. This saves 15% of the wall clock time for large eigenvalue problems where hundreds of eigenmodes are written to d3eigv.
  - Corrected the use of load curve specifications for the following implicit control variables:

    - IAUTO < 0 in *CONTROL_IMPLICIT_AUTOMATIC_DYN,_SPR
    - IMASS < 0 in *CONTROL_IMPLICIT_DYNAMICS_DYN,_SPR
    - ISTIF < 0 in *CONTROL_IMPLICIT_SOLUTION_DYN,_SPR

  - *CONTROL_IMPLICIT_INERTIA_RELIEF:

- Corrected the processing of some kinds of contact during the implicit inertia relief computation.
- Disabled use of iterative linear equation solvers in *CONTROL_IMPLICIT_SOLVER when *CONSTRAINED_IMPLICIT_INERTIA_RELIEF is active.
- Needed to account of inertia relief modes in MPP for the computation of resulting forces from constraints.
- Enhance explicit's use of *CONTROL_IMPLICIT_INERTIA_RELIEF by adding a second orthogonalization step for the accelerations.

◦ *CONTROL_IMPLICIT_MODES:

- Fixed application of implicit constraint modes in MPP for the case where a constraint mode is also a shared node.
- Added a switch on *CONTROL_IMPLICIT_MODES to not write d3mode file when generating superelements.
- Improved the creation of superelement files in MPP.
- Correct dynamic memory allocation issue that was causing problems with computing the reduced stiffness matrix in MPP for implicit modes.
- Corrected output to d3mode in MPP.
- Added error checking on nodes for *CONTROL_IMPLICIT_MODES to make sure they are deformable.

◦ *CONTROL_IMPLICIT_MODAL_DYNAMICS:

- Allow the use of shell formulation 18 in modal dynamics when reading from d3eigv.
- Correct a typo in the computation of the explicit time step for modal superposition. Also applies to *CONTROL_IMPLICIT_DYNAMICS with imass = 2 or 3.
- Adjusted reading of d3eigv for modal dynamics for the case where the modes are computed and then immediately used -- all with stresses included.
- Correct the implicit modal dynamics stress output for beams using non-default value for BEAMIP.

◦ *CONTROL_IMPLICIT_EIGENVALUE:

- Freed allocated memory that was not free at the end of subroutine im_mode_stress.
- Adjusted the logic for intermittent eigenvalue computation to deal with explicit not quite getting to the end time.
- Properly handle the output of 10 noded tets to the d3eigv database.
- Reduce size of d3eigv by not adding rotational dofs for solid element only models.

- The computation of the i/o address for writing the d3eigv database in MPP for models with 8 noded shell elements was wrong. The d3eigv database would probably fill the i/o system causing an abnormal termiation. The i/o address computation is now correct.
- Enable the computation of Campbell diagrams for Implicit Rotational Dynamics as part of the Intermittent Eigenvalue capabilities.
- Enable the use of intermittent eigenvalues during both dynamic relaxation and regular transient phase.
- For eigenvalue analyses where there are lots and lots of zero eigenmodes where the eigensolver fails we will now output the computed modes to the d3eigv database for model debugging purposes.
- Extend the cost logic for Lanczos Eigensolvr, both SMP and MPP, to be more aggressive about staying with the current shift. Improves performance for large problems as the cost function is not necessarily monotonic.
- Added shift and scaling to the standard eigenvalue problem to make the solution methodology more robust. This is the same shift and scaling as used for MCMS.
- Corrected output to d3eigv for implicit MPP eigensolver for large number of integration points.
- Add logic so that for some frequency response computations the amount of drilling rotation control is the same as for eigenvalue computations.
- Added Sectoral Symmetry to *CONTROL_IMPLICIT_EIGENVALUE as an implicit eigensolver option for SMP DP. Sectoral Symmetry is an eigensolver for models with a large degree of rotational symmetry such as fan blades.
- Adjust zero shift for Lanczos eigensolver.
- Fix up MPP storage issues in MPP implementation of Lanczos. Increased minimum storage of eigenvalues from 500 to 2500. Restricted eigenvector checking due to MPI buffer length.
- Add additional tag at end of d3eigv for MCMS that was missing. Normalize MCMS computed eigenvectors to have unit norm.

- Correct joint constraint processing in implicit mechanics which should improve the reporting of joint forces (*DATABASE_JNTFORC) in implicit.
- Enhance implicit's processing of nodal mass matrices to include the local coordinate transformation.
- *CONTROL_IMPLICIT_SOLUTION with NSOLVR = 1: Fixed problem in which Implicit Linear Analysis used the forces at the end of the linear step to compute resultant forces. This did not match expectations of the users. Computation of resultant forces, such as those in bndout, were changed to use the force at the beginning of the linear step.
- *CONTROL_IMPLICIT_SOLVER:

    - Correct the removal of ordering reuse logic for the MPP implementation of implicit mechanics. This is an important time saving feature for very large models with multiple time steps.
    - For intermittent eigenvalue analysis the direct solvers used by the eigenvalue analysis were forced for the entire execution. We now save and restore the solver option selected by the user to use during the non-eigenvalue execution of the run. This enables the use of iterative solvers iterative solvers.
    - Fix the implicit logic to properly deal with NEGEV = 1 in *CONTROL_IMPLICIT_SOLVER and IAUTO.ne.0 in *CONTROL_IMPLICIT_AUTO.
    - Disabled the use of ORDER = 1 in *CONTROL_IMPLICIT_SOLVER for MPP as this is a really bad choice in MPP.
    - Adjusted output to d3hsp for implicit linear equation solver options to match keyword manual. (*CONTROL_IMPLICIT_SOLVER)
    - Adjust implicit logic for dumping matrices (MTXDMP in *CONTROL_IMPLICIT_SOLVER). Correct the dumping of matrices from implicit when MTXDMP > 1.

- *CONTROL_IMPLICIT_STATIC_CONDENSATION: Correct indexing of array involved in implicit static condensation.
- Echo the input for *CONTROL_IMPLICIT_RESIDUAL_VECTOR to d3hsp.
- Fix an issue of the implicit linear algebra filenames being inconsistent between Linux and Windows.
- Add an implicit-only option for processing *CONSTRAINED_INTERPOLATION which is referred to as force-geometry-mass (fgm). Forces are moved from the dependent node to the independent nodes, geometry of the dependent node is computed from the independent nodes, and the mass on the dependent node is moved to the independent nodes. All without using a constraint matrix or LaGrange Multiplier approach.
- No longer automatically increase nodal DOF from 3 to 6 due to the presence of *CONSTRAINED_INTERPOLATION. This is unnecessary for solid element models and increases the cost of large solid element models for implicit by 15% due to huge increase in constraint processing to constrain out all of the rotational dofs.
- Correct implicit's handling of SPC constraints on shared nodes in MPP that are also involved in tied contact.
- Enhance Implicit constraint processing to accept function definitions for motor joint movement (*CONSTRAINED_JOINT_***_MOTOR).
- *CONTROL_IMPLICIT_SOLUTION: Added NSOLVR = -1 so that the initial geometry is used for each step of a multistep linear analysis.
- Added implicit time stamps and additional performance tracking for MPP Lanczos eigensolver. Extended command Line Option impcon = 1.
- Added a command line directive of impcon = 1 which tracks events during implicit simulation to analyze performance for large implicit jobs. This is

also a useful debugging tool as it helps identify the particular part of implicit is being executed at an abnormal termination.

- Single precision executables are now disallowed for implicit mechanics (both MPP and SMP).
- Restored final memory report for implicit SMP runs. Logic was based on old memory allocation scheme instead of new dynamic memory allocation.
- Turned on Mumps, a new linear solver option for implicit mechanics, to be included for double precision SMP on XEON64 and AMD64 systems.
- Corrected an inefficiency for implicit when there are many many AutoSPC constraints in MPP. A linear search through the AutoSPC constraints was replaced with a binary search.
- Corrected the collection of resultant forces when using implicit mechanics for tied contact using _OFFSET. Other tied contact such as _CONSTRAINED_OFFSET were OK.
- Added SMP implementation of new keyword *CONTROL_IMPLICIT_RESIDUAL_VECTORS.
- Apply logic that was in R9 and R10 but had not made it into R11 and R12 to check for incorrect values in nodal inertia array for implicit dynamics and eigenvalue analysis.
- Fixed divide by zero in power iteration for buckling problems with inertia relief. (*CONTROL_IMPLICIT_BUCKLING, *CONTROL_IMPLICIT_INERTIA_RELIEF)
- Correct problem where solution control was inappropriately returned to implicit after issuing sense switch sw1.
- Added *CONTROL_IMPLICIT_RESIDUAL_VECTOR. Purpose: Activate and control the computations of residual vectors. Residual vectors are the linear response of a model to a specified load which is then orthogonalized to a set of eigenmodes and any previously computed residual vectors. The eigenmodes can either be computed during the execution or read from an input file. The computation of residual vectors is the same as multi-step linear (NSOLVR = -1 on *CONTROL_IMPLICIT_SOLUTION) but has the additional step of orthogonalization.
- Add a stiffness option to *CONTROL_IMPLICIT_ROTATIONAL_DYNAMICS. The stiffness terms are suppressed to keep the stiffness matrix symmetric but the force terms are added.
- *MAT_ELASTIC_FLUID/*MAT_001_FLUID usage with *CONTROL_IMPLICIT_EIGENVALUE: Warning added to highlight the absence of enforcement of irrotationality in solid elements that use this fluid material model. In eigenvalue analyses this absence will often lead to the prediction of numerous spurious rotational fluid modes. This material model should therefore be used with caution in modal analyses.
- Fixed implicit element stiffness of shell elements when used with laminated shell theory, and when also using anisotropic materials *MAT_022, *MAT_034, *MAT_040, *MAT_054, *MAT_055, or *MAT_058. In certain cases, the shear stiffness was too small.

- ◦ Lagrangian multiplier joints, LMF = 1 on *CONTROL_RIGID, are supported for implicit analysis.
- ◦ Fixed bug with d3eigv when using TET10S8 = 1 on *CONTROL_OUTPUT.

- ***INITIAL**

  - ◦ *INITIAL_LAG_MAPPING:

    - ▪ Thick shell data from a previous run can be mapped to a new thick shell mesh.
    - ▪ *INITIAL_LAG_MAPPING with NELANGL = -3: For the Lagrangian mapping, project the boundary nodes of the current mesh onto the boundary faces of the previous mesh.
    - ▪ In a solid-to-solid mapping, map the strain array output by STRFLG = 1.
    - ▪ Map the nodal pressures for ELFORM = 13 tetrahedra.
    - ▪ If NELANGL = -2 in a 3D to 3D mapping, the data from the previous run are rotated around the axis of an angle ANGLE before mapping.
    - ▪ *INITIAL_LAG_MAPPING_WRITE3DAXI: For a 3D axisymmetric Lagrangian model, add the option WRITE3DAXI to *INITIAL_LAG_MAPPING to output the mapping file as if it came from a 2D axisymmetric model.

  - ◦ Fix bug in the *INITIAL_STRESS_TSHELL module of the dynain output file. For Thick Shells with 6-10 integration points through the thickness, the z-coordinates of the integration points were incorrect.
  - ◦ Added new option ISTIFF to *INITIAL_STRESS_SECTION for enhanced stability.
  - ◦ Added new ISTIFF option for artificial stiffness in *INITIAL_STRESS_SECTION.
  - ◦ Fix *INITIAL_STRESS_TSHELL output to dynain for thick shell form 7. Output stresses for all 4 in-plane integration points instead of just 1.
  - ◦ Fix incorrect rotational axis when using *PART_INERTIA and *INITIAL_VELOCITY_GENERATION with IRIGID = 1 and the rotational axis is defined using nodes by setting NX = -999 and defining the nodes with NY and NZ.
  - ◦ Make *MAT_GENERAL_NONLINEAR_1DOF_DISCRETE_BEAM / *MAT_121 work with *INITIAL_STRESS_BEAM.
  - ◦ Disable *INITIAL_AXIAL_FORCE_BEAM and *INITIAL_STRESS_SECTION when IDRFLG = -999 in *CONTROL_DYNAMIC_RELAXATION.
  - ◦ Fix seg fault or incorrect stresses when initializing stresses using *INITIAL_STRESS_SOLID with *MAT_107/*MAT_MODIFIED_JOHNSON_COOK.
  - ◦ Fixed a bug that made *INITIAL_VELOCITY_GENERATION_START_TIME not work for rigid parts.

- STRFLG in *DATABASE_EXTENT_BINARY will be automatically set to unity when using *INITIAL_STRAIN_SOLID.
- Fixed the writing of dynain files that use beam element formulations 7 or 8. They were writing the wrong number of integration points.
- Fixed a problem that occurred when when initializing stress in thick shells using a dynain file created by a previous run that used a different number of integration points per layer than the new run. In this case, when reading the dynain data to initialize the element, the stress tensor was being properly averaged or extrapolated as needed, but the material extra history variables were not, which for some materials caused wrong material behavior.
- Fixed the reading of initial strain data for thick shell formulations 2, 3, 5, 6, and 7. A strain transformation was missing.
- Add new keyword *INITIAL_HISTORY_NODE(_SET) to initialize select history variables at node locations. These nodal values are internally interpolated to the element integration points using the finite element shape functions. A main difference to using *INTIAL_STRESS_SHELL/SOLID, where you would need to initalize ALL history variables up the ones you really need/want to initialize is that *INITIAL_HISTORY_NODE(_SET) allows you to just pick the few variables you would like to initialize without touching the others.
- *INITIAL_AXIAL_FORCE_BEAM now supported for beam formulation 1 (H-L beam) and any material. Previously, only beam formulation 9 with *MAT_SPOTWELD was allowed.
- Selective mass scaling is not applicable to beam formulation 1 (Hughes-Liu beam).
- *CONTROL_ADAPTIVE: Users are able to use *INITIAL_VELOCITY_-GENERATION for the 3D adaptive parts as long as STYP = 3 (node set) and the corresponding node set is defined using *SET_NODE_GENERAL with OPTION = PART. (This would apply only to the unusual situation in which initial velocities were invoked AFTER the mesh is adapted.)

- **Isogeometric Analysis (IGA)**

  - *ELEMENT_SHELL_NURBS_PATCH:

    - Include initial parameterization check for isogeometric shells.
    - Include automatic removal of trimming loops fully contained within a single element.

  - Updated *DEFINE_SPOTWELD_RUPTURE to work with isogeometric shell elements.
  - Added IGA shell and IGA solid element erosion due to the material failure strain, *MAT_ADD_DAMAGE, and *MAT_ADD_EROSION.

- Contact for IGA in implicit is now available using the automatically generated interpolation elements. The IGACTC option in *CONTROL_CONTACT, however, is not currently supported for implicit.
- Added *INITIAL_VELOCITY_GENERATION support for IGA elements using parts and part sets.
- Added support for checking element stiffness matrices for IGA through the LPRINT option in *CONTROL_IMPLICIT_SOLVER.
- Both IGA solids and IGA shells now work with *MAT_RIGID. Previously, only IGA shells were supported for rigid material.
- IGA now works for implicit springback.
- Added the ISTUPD thickness change options (*CONTROL_SHELL) to ELFORM = 3 IGA shell elements.
- Added drilling stiffness type 4 (see DRCM in *CONTROL_IMPLICIT; DRCPSID and DRCPRM in *CONTROL_SHELL) to the IGA shells.
- Improved the time centering for IGA shells for improved convergence rates in implicit analysis.
- *ELEMENT_SHELL_NURBS_PATCH: Enabled keyword *LOAD_-NURBS_SHELL for trimmed elements.
- *ELEMENT_SOLID_NURBS_PATCH:
  - IINT defines the integration rule.
    - EQ.0: reduced Gauss integration rule (default)
    - EQ.1: full Gauss integration rule.
    - EQ.2: patch-wise reduced integration rule.
    - EQ.3: non-uniform Gauss integration rule uses full integration for exterior elements and reduced integration for interior elements.
  - Add conventional mass-scaling for isogeometric solids
- These material models are now supported for shells in isogeometric analysis:
  - *MAT_224/*MAT_TABULATED_JOHNSON_COOK
  - *MAT_054-055/*MAT_ENHANCED_COMPOSITE_DAMAGE
- Laminated shell theory is now supported for shells in isogeometric analysis. This option can be activated for shell formulations that allow for transverse shear deformability by:
  - Setting LAMSHT = 1,3 or 5 in *CONTROL_SHELL, and
  - Using *INTEGRATION_SHELL to specify an integration rule, which is referenced in *SECTION_SHELL.
- *INITIAL_STRAIN_SHELL_NURBS_PATCH: Fixed output for thickness in case ISTUPD = 0 in *CONTROL_SHELL.

  - *INITIAL_(STRESS/STRAIN)_SHELL_NURBS_PATCH: Fixed error in writing of *DEFINE_NURBS_CURVE to dynain file (*INTERFACE_-SPRINGBACK_LSDYNA).
  - Fixed bugs that arose from improper merging of interpolation nodes at patch boundaries. This could have lead to errors such as "Zero Normal Vector".
  - Remove the limitation to have NURBS patches and related trimming curves defined in a normalized parameter space as this is not common in CAD-files.
  - Fix bug when using IGA-shells (*ELEMENT_SHELL_NURBS_PATCH) together with user defined elements (ELFORM = 100-105).

- ***LOAD**

  - There were unexpected error terminations when *LOAD_THERMAL_-LOAD_CURVE was used in an adaptive analysis. This has been fixed.
  - *LOAD_THERMAL_D3PLOT reads temperature data from a file in d3plot format. The file name is specified as an option on the command line. If this file does not exist in the working directory, LS-DYNA now gives a reasonable message and exits gracefully.
  - *LOAD_BODY_GENERALIZED: fix a bug that happens when CID > 0 and the CID doesn't have its origin at (0,0,0).
  - Fix incorrect results when using *DEFINE_CURVE_FUNCTION with AX/AY/AZ for *LOAD_SEGMENT.
  - Fix output to nodfor when using *LOAD_SSA.
  - Fix ineffective *LOAD_THERMAL_CONSTANT when used with *MAT_-ELASTIC_VISCOPLASTIC_THERMAL/*MAT_106.
  - Fix seg fault when using *LOAD_MOVING_PRESSURE due to uninitialized variable.
  - Fixed a bug preventing *LOAD_BLAST_ENHANCED from working for 2D analysis.
  - Added line-of-sight feature (LSFLG) to *LOAD_MOVING_PRESSURE: When LSFLG.EQ.1, the pressure will be applied to the first-hit segments from the nozzle.
  - When IDIR = 3 in *LOAD_MOVING_PRESSURE, the pressure is in the direction from NODE1 to NODE2, but only the normal component of the pressure is applied on the segments.
  - Enabled *LOAD_THERMAL to work for beam nodes when nodal releases are defined at those nodes. The thermal loading was previously incorrect.
  - New keyword *LOAD_THERMAL_RSW: Prescribes nodal temperatures within a (possibly moving) ellipsoidal region of the structure in mechanical-only simulations. So there is no heat transfer available to the surroundings. Input is comparable to thermal boundary condition *BOUNDARY_TEMPERATURE_RSW. Features include:

    - Temperatures for the center and the boundary of the ellipsoid have to be input. Inbetween, there is a quadratic approximation.
    - Outside of the ellipsoid, no temperature values are prescribed.
    - Position and axis of symmetry are defined by two nodes
    - Optional definition of a heat affected zone (HAZ) around the weld nugget. Temperatures at its boundary are to be given and a linear interpolation from the boundary of the nugget to the boundary of the HAZ is used. Outside the HAZ a default temperature is applied.

  - Implement *LOAD_SPCFORC in MPP.
  - Reduce memory usage and computation time for *LOAD_SEISMIC_SSI_-NODE when ground motions are specified at every node on the soil-structure interface.
  - New keyword *LOAD_EXPANSION_PRESSURE applies a uniform pressure (e.g., from explosive detonation) on the internal surfaces of a chamber while one wall of the chamber moves (as a piston) so that the loaded area increases.

- ***MAT and *EOS**

  - Added *MAT_295/*MAT_ANISOTROPIC_HYPERELASTIC.

    - This is a collection of (nearly-in)compressible, (an)isotropic, hyperelastic material models primarily aimed at describing the mechanical behavior of biological soft tissues. Some of the material models may also be used to analyze a wider class of materials including fiber-reinforced elastomers and stretchable fabrics.
    - Modules include isotropic, anisotropic, and active anisotropic.
    - Allows for electrophysiology-fluid-structure interaction (EFSI).
    - Available for solid element formulations 1, 2, and 10.

  - *MAT_079/*MAT_HYSTERETIC_SOIL: New option for cyclic degradation. See input fields SIGTH, SIGR and CHI on Card 5.
  - *MAT_119/*MAT_GENERAL_NONLINEAR_6DOF_DISCRETE_BEAM: New failure option FCRIT (Field 7, Card 6, see also Remark 1). The pre-existing failure algorithm uses two separate criteria: one that considers only negative displacements and rotations of all six degrees of freedom, and a second criterion considering only positive displacements and rotations. Thus, positive x-displacement and positive y-displacement could combine to cause failure while positive x-displacement and negative y-displacement could not. The new option FCRIT = 1 offers a single criterion that considers both positive and negative displacements/rotations.
  - *MAT_169/*MAT_ARUP_ADHESIVE bugfix: The "interfacial failure" option (TMAXE, SMAXE) was not propagating from element to element as it should.

- Fixed bugs in *MAT_172/*MAT_CONCRETE_EC shear capacity calculation (TYPESC > 0):

  - This capability did not work with fully-integrated shell elements such as ELFORM = 16.
  - Incorrect results or error terminations could occur if material types other than 172 were mixed with MAT_172 in PART_COMPOSITE and the MAT_172 definition had TYPESC > 0.

- *MAT_197/*MAT_SEISMIC_ISOLATOR: New option for sliding isolator with rim failure (TYPE = 5).
- *MAT_203/*MAT_HYSTERETIC_REINFORCEMENT: Fixed bug in MAT_203 affecting beam elements with EPDAM1, EPDAM2 non-zero, and DRESID equal to zero. When the beam reached its erosion limit, error termination or unexpected erosion of neighboring elements could occur.
- *MAT_209/*MAT_HYSTERETIC_BEAM: New material model for modelling buildings in earthquakes. A significantly extended version of *MAT_191 (*MAT_SEISMIC_BEAM) with lumped plastic hinges at both ends of the beam and many additional capabilities.
- Add heat generation factor QCURE to *MAT_ADHESIVE_CURING_VISCOELASTIC/*MAT_277 which relates the heat generated in one time step with the increment of the degree of cure in that step.
- *MAT_296/*MAT_ANAND_VISCOPLASTICITY:

  - Correct the deformation resistance S calculation.
  - Fix floating point exception related bug when compiling with FPE option active.

- Add keyword *MAT_ADD_CHEM_SHRINKAGE.

  - The inputs and features are same as those of *MAT_ADD_THERMAL_EXPANSION except for the usage of curve LCID.
  - The physical chemical shrinkage rate is calculated as

    $$\beta = (1.0 - \alpha)^3 - 1.0$$

    where curve LCID defines alpha (ordinate) vs. temperature (abscissa).
  - When defining curve LCID in *DEFINE_CURVE, the variable DATTYP should be set to -100.

- Extension of THERML variable in *MAT_077_H to allow C11, C20, C02, and C30 to also be specified as a function of temperature.
- Add calculation of internal energy to *EOS_019/*EOS_MURNAGHAN.
- Make *EOS_019/*EOS_MURNAGHAN compatible with *INCLUDE_TRANSFORM.

- Add support for optional EOS in *MAT_COMPOSITE_FAILURE_SPH_-MODEL/*MAT_059.
- Variable PHEL in *MAT_JOHNSON_HOLMQUIST_CERAMICS/*MAT_-110 can be specified equal to 0.0 by default or by user input. In this case PHEL is calculated internally. This value is now echoed as "phel" in d3hsp.
- *MAT_ADD_PERMEABILITY: allow the permeability to be a function of volumetric strain, plastic strain rate, or pressure.
- *MAT_ADD_PORE_AIR: fix an MPP bug triggered when input format is long = s, write standard keyword input and write long structured input deck.
- *MAT_FABRIC: make strain restoration factor TSRFAC available for linear liner.
- *MAT_GENERAL_NONLINEAR_6DOF_DISCRETE_BEAM/*MAT_119:
  - Allow unloading option of "2" for crushable beam, IFLAG = 2.
  - Add the stiffness matrix to enable implicit for crushable beam.
- *MAT_SEATBELT:
  - Enhance 2D belt by offering choice of element formulations and material properties along transverse direction for 2D belt.
  - Enable 2D belt material to have loading curve, LLCID, defined as a table.
  - Issue error message when the edge nodal set of 2D belt part is not on the edge of the related 2D belt part
- Added isotropic plasticity to *MAT_030/*MAT_SHAPE_MEMORY.
- Added new material model *MAT_291/*MAT_SHAPE_MEMORY_ALLOY for shape memory alloys. This is a micromechanics-inspired constitutive model for shape-memory alloys that accounts for initiation and saturation of phase. This model is based on Kelly, Stebner, Bhattacharya (2016) and is available for solid elements only.
- Added support for curves defining thermal properties vs. mechanical history variables in *MAT_THERMAL_ISOTROPIC_TD_LC.
- Fixed the problem in *MAT_244/*MAT_UHS_STEEL where redundant table look-up operations in the plasticity algorithm increased the computational cost.
- Added history variable (at the 8th position) in *MAT_133/*MAT_BARLET_YLD2000 to allow user input of a effective plastic strain shift which can be used to account for stored energy in sub-structures after heat treatment. This option is only applicable to the case of HARD < 0 (curve/table input of hardening curves).
- Added 3D table option to hardening curve input of *MAT_233/*MAT_CAZACU_BARLAT so that the material model is applicable to simulations of hot/warm forming of materials such as magnesium.

- Fix incorrect initial strains when using *MAT_MOONEY-RIVLIN_RUBBER/*MAT_027 with *INITIAL_FOAM_REFERENCE_GEOMETRY and running in single precision.
- Fix problem of solution hanging when using *MAT_PIECEWISE_LINEAR_-PLASTICITY_STOCHASTIC and *DEFINE_HAZ_PROPERTIES (MPP only).
- Fix incorrect behavior for *MAT_GENERAL_JOINT_DISCRETE_-BEAM/*MAT_097 when BEGTIM in *CONTROL_START is set > 0.0.
- Fix incorrect stresses for *MAT_ANISOTROPIC_VISCOPLASTIC/*MAT_-103 and *MAT_ANISOTROPIC_PLASTIC/*MAT_103P when using BETA for *ELEMENT_SHELL_BETA or BETA in the *MAT input or material integration point angles input on *SECTION_SHELL.
- Fix seg fault when using *MAT_PIECEWISE_LINEAR_PLASTICITY/*MAT_024 for Hughes-Liu beam with implicit analysis.
- Output fitted data to curveplot for *MAT_HYPERELASTIC_RUBBER/MAT_77_H when N > 1.
- Fix effect of FBRT on tensile strength, XT, for *MAT_ENHANCED_COMPOSITE_DAMAGE/*MAT_055. FBRT should only affect the tensile strength upon compressive maxtrix mode failure.
- If FAILTM < 0.0 in *MAT_ADD_EROSION, skip erosion failure evaluation during dynamic relaxation phase.
- The calculation of Q12 for *MAT_SPECIAL_ORTHOTROPIC/MAT_130 and *MAT_RESULTANT_ANISOTROPIC/MAT_170 is incorrect. It should be Q12=(v21p*E11p)/(1-v12p*v21p).
- Fix ineffective curves LCSRA, LCSRB, LCSRC, LCSRAB, LCSRBC, LCSCA in *MAT_MODIFIED_HONEYCOMB/*MAT_126.
- Error terminate with message, INI+484, if LOG_INTERPOLATION is invoked for *MAT_PIECEWISE_LINEAR_PLASTICITY/*MAT_024 and the strain rate for the table is defined by LCSS is negative.
- Fix ineffective user-define failure subroutine (FAIL < 0) for *MAT_MODIFIED_PIECEWISE_LINEAR_PLASTICITY/*MAT_123.
- Fix output to elout for solids using *MAT_PAPER/*MAT_274 and CMPFLG > 0 in *DATABASE_EXTENT_BINARY.
- Allow *MAT_065/*MAT_MODIFIED_ZERILLI_ARMSTRONG to compute the flow stress using a more general form by using power exponent, m, of the effective plastic strain instead of the square root of the effective plastic strain in the case of an FCC metal.
- Fix seg fault when using *MAT_ADD_EROSION for *PART_COMPOSITE when the first layer of the *PART_COMPOSITE is not using *MAT_ADD_-EROSION.
- Fix incorrect direction of thermal expansion when using *MAT_NONLINEAR_ORTHOTROPIC / *MAT_040 with *MAT_ADD_THERMAL_EXPANSION.
- Fix contact penetration after element erosion when using *MAT_110 / *MAT_JOHNSON_HOLMQUIST_CERAMICS with eroding contact.

- Fix incorrect stress/strain output to d3plot when using *MAT_023 / *MAT_TEMPERATURE_DEPENDENT_ORTHOTROPIC and running 2D analysis with MAXINT > 3 in *DATABASE_EXTENT_BINARY.
- Fix incorrect stress initialization when using *LOAD_DENSITY_DEPTH in combination with *MAT_005/*MAT_SOIL_AND_FOAM where the material's volumetric strain vs. pressure is input using curve LCID.
- Fix strain rate effects on *MAT_157/*MAT_ANISOTROPIC_ELASTIC_-PLASTIC for implicit static analysis.
- Fix seg fault when using *MAT_278/*MAT_CF_MICROMECHANICS.
- Fixed a bug causing 2WAY = 1 in *MAT_ENHANCED_COMPOSITE_-DAMAGE/*MAT_054 not to be recognized.
- Added rate sensitivity to *MAT_S04/*MAT_SPRING_NONLINEAR_-ELASTIC: LCD can point to a table (*DEFINE_TABLE) which gives loading rates (relative velocity values). There is then defined a corresponding curve of force vs. displacement for each loading rate.
- Added PRESTRAIN keyword option for *MAT_MODIFIED_PIECEWISE_-LINEAR_PLASTICITY/*MAT_123. When PRESTRAIN is used, the variable IPS on Card 5 is used to control the prestrain option. If IPS is set to 1, then the strain tensor defined by *INITIAL_STRAIN_SHELL will be used as prestrain when checking major strain failure, EPSMAJ.
- Added a new option called TTOPT to *MAT_SPOTWELD which is used by solid and solid assembly spot welds. The option controls the behavior of TRUE_T making it possible to revert the TRUE_T behavior back to how it worked in R8 and previously. If TTOPT = 0 or if the TTOPT is not input, then the behavior is unchanged with this update. If TTOPT = 1, then R8 and earlier behavior happens. TTOPT = 2 is like TTOPT = 1, but the average top and bottom values of forces and moments are used for failure making failure invariant with respect to element numbering.
- Added support for user-defined spot weld failure in *MAT_100_DAMAGE-FAILURE with individual solid elements. Previously, only beam elements and solid weld assemblies were supported.
- Enabled *MAT_ADD_THERMAL_EXPANSION to work in thick shell elements when the solutions is mechanical only. Some code was missing causing bad behavior.
- Enabled *MAT_128/*MAT_HEART_TISSUE for thick shell forms 3, 5, and 7. Initialization of the F tensor was missing causing a 1st cycle termination.
- Enabled *MAT_145, *MAT_244, and *MAT_254 to work with thick shell formulations 3, 5, and 7.
- Enabled multiple instances of nonlocal material averaging to be used in MPP. This change affects *MAT_NONLOCAL, and also *MAT_CODAM2, *MAT_GURSON_RCDC, or *MAT_PLASTICITY_WITH_DAMAGE with the RCDC option.
- Enabled the variable EC of *MAT_058/*MAT_LAMINATED_COMPOSITE_FABRIC to be used with thick shell formulations 1, 2 and 6. If a positive value is input, then it will be used. f EC is left blank or set to zero or a

negative number, then the minimum of EA and EB will continue to be used. Also, *MAT_058 will now echo to d3hsp EC = 0 when used with thin shells because the EC value is ignored.

- Improved hourglass form 6 when use with *MAT_089/*MAT_PLASTICITY_POLYMER. A poor choice of the material parameter used to scale the hourglass stiffness made it difficult to use.
- Fixed spot weld assemblies that use user defined failure option 12 in *MAT_100. An error was causing OPT = 12 to not work at all.
- Fixed the use of *MAT_ADD_DAMAGE_DIEM with *MAT_024 and tetrahedral element formulation 13. The damage calculation used bad data leading to a wrong result.
- Fixed the initialization of the material direction for orthotropic materials when used with AOPT = 2 in 2D plane strain, plane stress, and axisymmetric elements.
- Fixed implicit beam spot weld elements (*SECTION_BEAM ELFORM = 9) when used with NF > 0 on *MAT_100/*MAT_SPOTWELD. The averaging of the stress was causing the model to converge to the wrong solution with beam forces too low.
- *MAT_296/*MAT_ANAND_VISCOPLASTICITY: Solved a problem that caused discrepancies between material parameters provided in the input keyword and the values reported in the d3hsp file. Analysis results were not impacted.
- *MAT_097/*MAT_GENERAL_JOINT_DISCRETE_BEAM: Solved a problem that may have caused analysis termination due to fatal error when a simulation switched from implicit to explicit.
- *MAT_261/*MAT_LAMINATED_FRACTURE_DAIMLER_PINHO: Solved a problem that may have prevented accumulation of back stress for in-plane shear plasticity model.
- *MAT_LAMINATED_FRACTURE_DAIMLER_CAMANHO/*MAT_262:

  - Fixed bug in in-plane shear behavior for tshells if curves are used to define strain rate dependency of SIGY and ETAN.
  - Corrected warning messages in case of tshell formulations 1/2 and adjusted the computation of element length for fracture toughness to match with thin shells.
  - Added flag MSG in *MAT_262 to control the writing of warning messages.
  - Added transverse shear damage similar to *MAT_054/*MAT_058.
  - Added shell-only option to use nonlinear (elastic) stress-strain curves instead of constant moduli (EA, EB, GAB). This allows asymmetric tension-compression.
  - Added flag DSF that controls the failure of an integration point based on in-plane shear failure. This was added to be able to reinstate the in-plane shear failure behavior prior to revision 117876 (DSF = 1).

    - Fixed incorrect behavior of *MAT_262 when first layer in *PART_-COMPOSITE used a non-262 material model (shells only).
    - Fixed bug when using table (2D) for defining strain rate dependent fracture toughnesses values with respect to characteristic element length.

- *MAT_LAMINATED_COMPOSITE_FABRIC/*MAT_058:

    - Fixed an error in calculation of the initial time step when curves define the stiffness values for EA, EB, GAB (shells only).
    - Add the SOLID option to support *MAT_058 for solid elements.
    - Fixed bug when using curve LCEFS to define ERODS failure strain as a function of strain rate (shells only).
    - Add option to define direction-dependent failure strains. For this, define a curve LCDFAIL that has 5 (for shells) or 9 (for solids) data points and has LCINT set to the number of data points. The ordinate values define the failure strains in the different directions as follows: 1: ef_11t ; 2: ef_11c ; 3: ef_22t ; 4: ef_22c ; 5: ef_12 ; (and additionally for solids) 6: ef_33t ; 7: ef_33c ; 8: ef_23 ; 9: ef_31, The curve's abscissa values must be ascending to meet input requirements but are otherwise ignored.

- *MAT_ANISOTROPIC_ELASTIC_PLASTIC/*MAT_157: Output the history variable that tracks failure (according to the Tsai-Wu or Tsai-Hill criterion) as history variable #10.
- *MAT_ENHANCED_COMPOSITE_DAMAGE/*MAT_054: Correct the computation of transverse shear strains for solids and tshells when EPSR and EPSF are defined.
- *MAT_VISCOPLASTIC_MIXED_HARDENING/*MAT_225: Fix issues when a table to account for strain-rate dependency is used together with kinematic hardening (BETA < 1).
- Add support of IORTHO = 1 for cohesive user-define materials.
- *MAT_106/*MAT_ELASTIC_VISCOPLASTIC_THERMAL: Added possibility to specify up to 8 user-defined history variables for output. The values of these history variables are computed via *DEFINE_FUNCTION. These values can, for example, be used to evaluate the hardness of the material.
- *MAT_254/*MAT_GENERALIZED_PHASE_CHANGE:

    - New transformation laws (PTLAW = 5,6,7,8,9) which are tailored for titanium (Ti-6Al-4V).

        EQ.5: Recovery of alpha martensite based on JMAK, using Koistinen-Marburger equation to estimate equilibrium concentration. Transformation of recovered phase partially to to-phase/

        EQ.6: Associated with 5; transformation of remaining recovered phase to this to-phase.

EQ.7: Parabolic growth rate calculating the step-wise dissolution of two or more phases into one other phase. This law calculates the sum of remaining phase concentrations. The from-phase is the first to be dissolved.

EQ.8: Associated with 7; additional phases to be dissolved according to 7. Phases are dissolved one after the other.

EQ.9: Extensions of Koistinen-Marburger; threshold of fr-phase and incomplete transformation within a certain range of temperature rates.

EQ.12: JMAK transformation law (same as 2), but active within the temperature range from start and end temperature, i.e. it is active in heating and cooling.

- Read PTTAB6 for JMAK transformation law to account for accelerated phase transformation due to plastic strain in the material.
- Account for TSF (*CONTROL_THERMAL_SOLVER) in phase kinetics.
- History data only written for phases that are actually defined.
- Added variable POSTV which allows user to specify additional history variables (accumulated thermal strain, accumulated strain tensor, plastic strain tensor, equivalent strain) for output. When the temperature is in the annealing range, these values are reset.
- New input variable for initial plastic strain to initialize a constant nonzero equivalent plastic strain value in the whole part.
- Added enhanced annealing option that controls the reset of plastic strains above a certain critical temperature. The anneal process can be described by a time and temperature dependent approach (JMAK, tabular input, load curve inputs).
- Add a cut-off temperature for the thermal expansion.

◦ *MAT_270/*MAT_CWM:

- New option ANOPT (annealing option) that allows cutting the thermal expansion above a temperature limit (e.g. annealing temperature).
- Added variable POSTV which allows user to specify additional history variables (accumulated thermal strain, accumulated strain tensor, plastic strain tensor, equivalent strain) for output. When the temperature is in the annealing range, these values are reset.

◦ *MAT_277/*MAT_ADHESIVE_CURING_VISCOELASTIC:

- Implemented Arrhenius shift function as alternative to the WLF shift function.
- Heat generation due to curing can be accounted for.

◦ *MAT_278/*MAT_CF_MICROMECHANICS:

        - Added curing induced heating.
        - Fixed the solid formulation to give comparable results as the shell formulations.
        - For the solid formulation, added input for the elastic properties of the fiber contribution in the direction that is orthogonal to the fiber plane.

- Fix for combination of *MAT_ADD_PORE_AIR and *MAT_ADD_EROSION with solid element formulations 1, 10, 43, 60, 61, and 99. History variables could have been mixed up.
- Increase accuracy for rate curves SIGVM < 0 and MXEPS < 0 of *MAT_ADD_EROSION. The original input curve is used instead of the rediscretized one.
- Add new variable P4 for *MAT_ADD_DAMAGE_DIEM to define if transverse shear stresses are included in the stress invariant computations (e.g triaxiality) or not. Useful for shells with corresponding “plane stress” material models, such as *MAT_024_2D, *MAT_036, ...
- Add new option MIDFAIL = 4 to GISSMO (*MAT_ADD_DAMAGE_GISSMO) and "eGISSMO" (*MAT_ADD_GENERALIZED_DAMAGE).
- Add new option LP2BI to *MAT_ADD_DAMAGE_GISSMO (shells only): failure becomes a function of triaxiality and a new bending indicator (adopted from *MAT_258).
- Fixes for GISSMO damage when used together with an equation-of-state (*EOS).
- Take IRATE flag from *CONTROL_IMPLICIT_DYNAMICS into account for GISSMO.
- Add new keyword *MAT_ADD_SOC_EXPANSION for geometric expansion due to State Of Charge from EM solver (currently only available for isotropic, hypoelastic materials with solid element formulations -2, -1, 1, 2, and 10).
- Add option to *MAT_024 with VP = 3, where yield stress can now be a function of plastic strain, strain rate, and up to five history variables that can be set using *INITIAL_HISTORY_NODE. That means LCSS refers to *DEFINE_TABLE_XD up to a level of 7.
- Add *MAT_024's option VP = 3 (filtered total strain rate) for solid elements.
- Enhancement when using log interpolation of strain rate data with VP = 1 in shell elements in *MAT_024, *MAT_123, or *MAT_251: eliminate numerical noise for very low strain rates by internally adding lower limits for strain rate and yield stress.
- Add effective strain as history variable #28 for *MAT_034/*MAT_FABRIC with FORM = 14, 24, or -14.
- Fix problem with internal computation of E0 for *MAT_036/*MAT_3-PARAMETER_BARLAT if hardening rule HR = 2 or HR = 5 is used AND E0 is zero in the input AND E is lessthan zero (curve for Young's modulus).
- Add new options to *MAT_068/*MAT_NONLINEAR_PLASTIC_DISCRETE_BEAM: translational and rotational stiffnesses TK{R,S,T} and

RK{R,S,T} can now be assigned negative values referring to curve IDs for load/moment vs. displacement/twist to get nonlinear elastic behavior.

- Fix for *MAT_105/*MAT_DAMAGE_2 with large curve IDs for LCSS (only solids).
- Make *MAT_133/*MAT_BARLAT_YLD2000 available for solid elements (currently only for explicit analysis and without C-R kinematic hardening).
- Fix for *MAT_169/*MAT_ARUP_ADHESIVE with variables referring to *DEFINE_FUNCTIONs: function ID offsets from *INCLUDE_TRANSFORM (IDFOFF) did not work.
- *MAT_187/*MAT_SAMP-1: instability criterion now available as history variable #28.
- Modifications for *MAT_187L/*MAT_SAMP_LIGHT:
  - log interpolation for table (optional) and no extrapolation
  - improved tolerance for plane stress iteration procedure
  - add viscoelasticity (defined by LCEMOD and BETA)
  - add flag to control curve treatment, either using discretized (0, default) or original (1).
- Fix thermal expansion in *MAT_188/*MAT_THERMO_ELASTO_VISCOPLASTIC_CREEP for solid elements: double precision version could show no expansion at all.
- Enable *MAT_199/*MAT_BARLAT_YLD2004 to be used with shell element formulations 25,26,27 and tshell element formulations 3,5,7. Set default values (1.0) for the transformation matrix coefficients.
- Add option to *MAT_224/*MAT_TABULATED_JOHNSON_COOK: variable BETA < 0 can now refer to a *DEFINE_TABLE_4D for triaxiality (TABLE_4D), temperature (TABLE_3D), strain rate (TABLE) and plastic strain (CURVE) dependence.
- Allow *MAT_224 to be used with Lode parameter dependent tables even for shell elements, i.e., in cases where LCF refers to *DEFINE_TABLE and/or LCI refers to *DEFINE_TABLE_3D.
- Allow *MAT_224 implicit to be used with EFG and MEFEM (ELFORMs 41, 42, 43, 45; see *SECTION_SOLID).
- Change behavior of NUMINT > 1 for *MAT_224 if used with multi-integration point solid elements. Integration points that reach damage = 1 still update stresses and therefore carry load. Only if NUMINT integration points reach damage = 1, the element gets eroded. The old way (single IPs got zero stresses before the element was deleted) just led to severe deformations and instabilities.
- Fix floating invalid problem in *MAT_224 (solids and LCG > 0): plastic strain rate could have become negative in rare cases.
- Make *MAT_251/*MAT_TAILORED_PROPERTIES available for solid elements.

  - Enable numerical tangent for *MAT_252/*MAT_TOUGHENED_ADHESIVE_POLYMER allowing implicit analysis i^{th} moderate nonlinearities.
  - Fixed problem using *MAT_258/*MAT_NON_QUADRATIC_FAILURE and *DAMPING_PART_STIFFNESS together with RYLEN = 2 in *CONTROL_ENERGY.
  - Modification for *MAT_280/*MAT_GLASS: Parameter NIPF now also applies to the EPSCR failure criteria, i.e., elements get deleted when NIPF integration points failed due to reaching EPSCR.
  - *MAT_ADD_INELASTICITY introduced for low order solids and shells. This keyword can be used to add inelastic behaviors, including plasticity, viscoelasticity, and creep, to any material model whose stress is calculated incrementally.
  - *MAT_ADD_DAMAGE_DIEM now supports Hughes-Liu beams, thick shell type 2, and axisymmetric elements.
  - All parameters on first two cards of *MAT_ADD_EROSION are supported for Instability on *DEFINE_MATERIAL_HISTORIES.
  - *MAT_EXTENDED_3-PARAMETER_BARLAT/*MAT_036E now supports tables to define stress vs. plastic strain and strain rate. Viscoplastic consistency is mandatory.
  - Enchanced message SOL+737. “Strain range of stress-strain input for *MAT_031 has been exceeded.”
  - In checking for fixed nodes for PML boundary (*MAT_PML_...), correctly merge translational constraints from *NODE and *BOUNDARY_SPC, in case both are used at the same node.
  - Calculate internal energy correctly for PML elements. *MAT_PML_NULL may have issues.
  - Implement *DAMPING_FREQUENCY_RANGE_DEFORM for solid PML elements.
  - Introduce new label “shl2d” for user materials in plane stress elements, distinguishing them from shells.
  - Add keyword "*MAT_ADD_CHEM_SHRINKAGE_TITLE", with which the chemical shrinkage effect can be simulated.
  - *MAT_242/*MAT_KINEMATIC_HARDENING_BARLAT2000:

    - Define Young's modulus with a curve. If EA is less than zero, the absolute value of EA is the curve ID defining Young's modulus as a function of plastic strain.
    - Add option for modified Yoshida formulation with two variables SC1, SC2 (same as *MAT_125).

- **MPP**

  - When MPP predecomposition is done for more processors than the calculation, and the model contains airbags, the airbag ownership was not being properly handled leading to problems including possibly segfault. This has been fixed.

- Improvements and fixes for MPP handling of decomposition regions and processor subsets.
- Fix MPP contact initialization issue that was segfaulting in some models if parts were being excluded from the d3plot output.
- Add new pfile option "transform_keyword" to the "decomp" section. Including this new flag will cause the global decomposition transformations to apply to all the decomposition regions created by these keywords:

  *PARTS_DISTRIBUTE

  *PARTSET_DISTRIBUTE

  *ARRANGE_PARTS

  *CONTACT_DISTRIBUTE

  Before r86885, and after that was reverted in r102224, these regions were decomposed without any coordinate transformation at all. Between those versions, the global transformation applied to all these regions. I think this behavior makes sense, but someone complained that r86885 changed behavior, so it was removed in r102224. Then someone ELSE complained that r10224 changed the behavior that THEY wanted, so this option is now added.
- Fix missing initialization that broke d3plot files when MPP predecomposition was used with models having *DEFINE_CURVE_FUNCTIONs that used the PIDCTL function.
- Corrected MPP processing of *PART_MODES option of ANSID .ne. 0. The node list for this option needed to be sorted and logic adjusted so that all processes, even those with no such nodes, are involved in the communication.

- **Output**

  - *DATABASE_TRACER_GENERAL: In this keyword, tracers are nodes that output histories for any solids or beams or shells or thick shells with the tracers in their volumes.
  - *DATABASE_TRACER_GENERAL and *CONTROL_REFINE_SHELL: Implement the tracers in the shell refinement.
  - *DATABASE_TRACER_GENERAL and *CONTROL_REFINE_SOLID: Implement the tracers in the solid refinement.
  - *DATABASE_PROFILE: Implement TYPE = 6 to output the distribution (along a given direction) of thick shell data.
  - *DATABASE_PROFILE: Output strain profiles if STRFLG = 1 in *DATABASE_EXTENT_BINARY.

- MPP fix for frictional work output to intfor (*DATABASE_BINARY_INTFOR) when soft = 2 contact is being used
- "Interface Pressure" output to intfor is now returned to always being positive. There is no clear way to distinguish between compression and tension at this point in the code, so restoring the old behavior seems best.
- Fix issue with reported time step controlling element in glstat.
- Reaction forces for superelements (*ELEMENT_DIRECT_MATRIX_INPUT) using explicit time integration are now available in spcforc.
- SPCs in *CONSTRAINED_NODAL_RIGID_BODY_SPC are now internally converted to equivalent *BOUNDARY_PRESCRIBED_ MOTION_RIGID if SPC2BND is set to 1 on *CONTROL_OUTPUT. This allows output of reactions to bndout.
- New stress extrapolation from gauss points to nodes for shells (similar to the SOLSIG option for solids). Enable by setting the SHLSIG variable to 1 in *CONTROL_OUTPUT. Applies to shell element formulations 16, 20, and 21.
- *DATABASE_RECOVER_NODE:
  - Fix an MPP bug for the nodal stress calculation of 10-node tetrahedron element.
  - Add 3 more nodal stress components to be recovered by using the velocity vector.
  - Add elemental extrapolation method for stress recovery.
- Error terminate with KEY+501 message if different BEAMIP settings are used in duplicate *DATABASE_EXTENT_BINARY.
- *ele: Eliminate order dependency of IDOF in *SECTION_SHELL in which the IDOF in the first shell part became the default for the other shell parts with the same formulation instead of being independent.
- Fix incorrect rigid body velocities output to matsum for axisymmetric analysis with eroding elements.
- Fix incorrect rigidwall forces & stonewall energy reported for *RIGIDWALL_PLANAR_ORTHO.
- Shell reference surface offset NLOC in *SECTION_SHELL will now properly affect moment calculation for output to secforc (*DATABASE_SECFORC).
- Eliminated output of nodes having zero mass to massout (*DATABASE_MASSOUT).
- Fixed a bug that caused inconsistencies in strains output to d3plot and d3part.
- Fixed a bug that caused discrepancies in elout and disbout for discrete beams using *MAT_121/*MAT_GENERAL_NONLINEAR_1DOF_DISCRETE_BEAM.
- Fixed a bug that caused zero forces in nodfor for discrete beams (beam ELFORM 6).

- Fixed a bug that caused incorrect stresses in d3plot and elout for shell ELFORM 23.
- Fixed a bug that caused discrepancy in rigid body velocity output to glstat and matsum.
- Fixed a bug that caused segfault when using LCUR in *DATABASE_DISBOUT.
- Added missing report of triggered failure criterion when solid elements fail due to *MAT_ADD_EROSION.
- Fixed a bug that caused incorrect filtered results in ncforc when using *DATABASE_NCFORC_FILTER in MPP.
- Enabled the change in beam length to be viewed when fringe plotting discrete beams.
- Enabled variable ENGFLG on *DATABASE_EXTENT_BINARY to apply to thick shells as well as thin shells. This allows internal energy density of tshells to be written to d3plot.
- Fixed a thick shell output problem affecting d3plot that occurred when MAXINT on *DATABASE_EXTENT_BINARY was larger than the number of layers in the element. For the nonexistent layers, we were reading data from the next element in memory. Now data for the last existing point is repeated.
- Fixed two issues when reporting eroded hourglass energy to glstat and matsum. For solid elements, the eroded energy was not removed from the regular hourglass energy, so it was counted twice causing a poor energy balance in the glstat data. Also, thick shells were not reporting eroded hourglass energy.
- Fixed bug in which DECOMP = 2 or 4 in *DATABASE_EXTENT_BINARY corrupts d3drlf.
- Fixed bug whereby a simple restart corrupted the d3part database.
- Echo *DATABASE_EXTENT_BINARY_COMP variables to d3hsp.

- **Restarts**

  - *DAMPING_FREQUENCY_RANGE_DEFORM now works smoothly with full deck restarts (*STRESS_INITIALIZATION). Applies to solids, beams, shells, thick shells. Previously, a stress-jump would occur at the beginning of the restart run.
  - Fix conditional so that node rotational masses are properly synchronized when initializing *CONTACT_TIED_..._BEAM_OFFSET during MPP full deck restart.
  - Fix MPP full deck restart issue that caused hanging if the new input file contained encrypted input.
  - Correct the handling of restarting implicit after full restart.
  - Fix bug affecting restarts in implicit.
  - Support the following for full deck restart and consequently, *CONTROL_-MPP_DECOMPOSITION_REDECOMPOSITION.

        - SPH active region with local coordinate system (*DEFINE_SPH_ACTIVE_REGION, ICID).
        - SPH injection (*DEFINE_SPH_INJECTION)
        - Explicit joints (*CONSTRAINED_JOINT).
        - LMF joint formulation (*CONTROL_RIGID, LMF = 1).

    - Fix seg fault when using *DELETE_CONTACT for restart when running with SMP.
    - Fix error termination for full deck restart using *DEFINE_ELEMENT_DEATH.
    - Added new option COMP to *DELETE_PART for simple restart.

        *DELETE_PART_COMP

      This option eliminates data pertaining to deleted parts from d3plot thereby reducing the size of the d3plot database.

- ***SENSOR**

    - Limit the usage of *SENSOR to transient analysis only, not dynamic relaxation.
    - *SENSOR_DEFINE_CALC-MATH: fix a bug triggered when CALC = SUM and any of involved sensors, SENSi, are negative for subtraction.
    - *SENSOR_DEFINE_ELEMENT: fix a bug triggered when trying to trace the stress/strain of a failed element.
    - *SENSOR_DEFINE_FORCE: fix an MPP bug for TYPE = CONTACT2D.
    - *SENSOR_DEFINE_MISC: fix a bug in which incorrect kinetic energy of a rigid body was sensed.
    - *SENSOR_SWITCH_CALC-LOGIC: allow as many switches as needed for logic calculation.
    - *SENSOR_CONTROL:

        - Fix a bug for TYPE = BAGVENTPOP in which venting is not turned off for *AIRBAG_HYBRID.
        - Fix a bug for TYPE = PRESC-MOT triggered when the referenced node or rigid part has a big ID.
        - Fix a bug for TYPE = BELTPRET that fails to turn on the pretensioner.
        - Fix a bug for TYPE = BELTRETRA that fails to lock the retractor when NEWLEG of *CONTROL_OUTPUT > 0 and no SBTOUT output is requested.
        - Add CNRB option to control *CONSTRAINED_NODAL_RIGID_BODY.
        - Add TYPE = LOADTHM to control *LOAD_THERMAL_VARIABLE and *LOAD_THERMAL_VARIABLE_NODE
        - Add optional time delay.

- Fix order dependency problem when using *DEFORMABLE_TO_RIGID activated by *SENSOR_CONTROL TYPE = "DEF2RIG".

- **SPG (Smooth Particle Galerkin)**

  - Volume of 6-noded solid elements using element formulation 42 (SPG) was calculated incorrectly. That bug is now fixed.
  - Fix SPG bug in MPP whereby the time step size varies with different number of MPP processes.
  - *CONSTRAINED_IMMERSED_IN_SPG: Fix fatal memory issue if the number of immersed nodes is much larger than the number of nodes of the SPG part.
  - Enabled additional material models for SPG application, some of which include orthotropic behavior: *MAT_110, *MAT_122, *MAT_123, *MAT_143, *MAT_199, *MAT_260A, *MAT_269.
  - *SECTION_SOLID_SPG now automatically sets element formulation to 47.
  - If FAIL is set greater than 0 in *MAT_003 or *MAT_024, it is used in the SPG bond failure.
  - Added SPG bond failure criteria corresponding to *MAT_ADD_DAMAGE_GISSMO and *MAT_ADD_EROSION (1st principal stress, max shear strain, 3rd principal strain).

- **SPH (Smooth Particle Hydrodynamics)**

  - Incompressible SPH (FORM = 13) is available in beta form.
  - Add two material models for implicit SPH formulation (FORM = 13):

    - *MAT_300/*MAT_SPH_02/*MAT_SPH_IMPLICIT_FLUID: Input includes surface tension coefficient and surface area minimization coefficient.
    - *MAT_301/*MAT_SPH_03/*MAT_SPH_IMPLICIT_STRUCTURE: Used for boundary particles. Input includes surface adhesion coefficient.

  - Add *DATABASE_SPHMASSFLOW and *DEFINE_SPH_MASSFLOW_PLANE to measure SPH mass flow rate across a defined plane. This feature is to SPH as *DATABASE_DEMASSFLOW is to DEM.
  - Add option to move *DEFINE_SPH_ACTIVE_REGION around by following a moving coordinate system.
  - Add option for buffer zone in *DEFINE_SPH_ACTIVE_REGION.
  - Add an error message when a *DEFINE_BOX_SPH is referenced in *CONTROL_SPH boxid. This should point to a regular *DEFINE_BOX.
  - Add option to reactivate particles, and add an optional buffer zone in *DEFINE_BOX_SPH.
  - *DEFINE_BOX_SPH gets a special treatment with FORM = 13. Particles outside of the activation box get frozen in space, and deactivated particles can

become active when entering the box. This allows calculating only particles surrounding the region of interest.

- Reduce d3plot size for incompressible SPH by trimming down irrelevant variables.
- *DEFINE_ADAPTIVE_SOLID_TO_SPH:
  - Zero out increment of internal energy for eroded particles.
  - Bug fix in SMP: Results were inconsistent in SMP when ncpu was greater than one.
  - Bug fix in stress calculation of embedded SPH particles when ICPL = 1. Internal energy of these particles was also inconsistent.
  - When elements erode, distribute their internal energy to their child particles.
  - Reset IOPT = 0 if ICPL = 0.
  - Fix the MPP 3D strain calculation when adaptive with pure thermal analysis is used. Remove contribution of adaptive particles to strain rate calculation.
  - Properly update density of embedded particles based on element current volume. When particles became active, they used to still carry the rest density.
  - Enforce embedded SPH particles to output zero internal energy, for energy conservation.
- *DEFINE_SPH_INJECTION:
  - Speed-up initial smoothing length calculation when *DEFINE_SPH_INJECTION is present. It used to be very slow for models containing a large number of injected particles.
  - Add option to define a variable speed of injection. If the injection speed scale factor is defined as negative, it refers to a *DEFINE_CURVE defining the velocity magnitude vs. time.
  - Use yet-to-be injected particles as ghost particles on the other side of the injection plane.
  - If the coordinate system is updated in time (IFLAG = 1 in *DEFINE_COORDINATE_NODES), we also update the injection direction.
  - If the injection plane follows a node (NID in *DEFINE_SPH_INJECTION), and IFLAG = 1 in *DEFINE_COORDINATE_NODES, we also continously rotate the whole stack of particles to be injected with the plane.
- Added SPH support for *MAT_193/*MAT_DRUCKER_PRAGER.
- Bug fix in psetid option of *DATABASE_BINARY_D3PLOT if psetid contains SPH parts.
- Fix ASCII output of SPH neighbor count, which was always zero.

- Bug fix in SPH neighbor search when variable smoothing length is employed. The neighbor list was created based on minimum smoothing length instead of maximum. As a result, some neighbors were potentially omitted.
- Add option in *CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION to remove dead SPH particles from the model at each redecomposition step.
- Improve MPP load balancing when a *CONTROL_MPP_DECOMPOSITION_REDECOMPOSITION is present and a moving box defines the activation region of SPH.
- Create MPP variants of 2D plane-strain and 2D axisymetric SPH formulations with thermal coupling.
- Add support for *LOAD_THERMAL_LOAD_CURVE in SPH.
- Fix bug for SPH thermal formulation in MPP. Artificial viscosity calculation was incorrect.
- Fix bug in MPP with ERODING contact and SPH. In some MPP decompositions, some processors were ignoring contact between SPH and Lagrangian parts.
- Restored TEROD and CEROD parameters for *MAT_NULL with SPH particles (these two parameters were not applied from R9).
- Material models added for SPH:
  - *MAT_COMPOSITE_DAMAGE/MAT_022 for SPH plane strain.
  - *MAT_MODIFIED_CRUSHABLE_FOAM/*MAT_163 for SPH axisymmetric, plane strain, and 3D.
  - *MAT_ENHANCED_COMPOSITE/*MAT_054,_055 for SPH axisymmetric and 3D.
- *CONTROL_SPH:
  - Improvements in adaptive smoothed particle hydrodynamics formulations (FORMs 9,10) with an anisotropic kernel. Speed up SMP operationsr; optimize bucket cycle; update principle direction and scale factor.
  - Renamed MEMORY variable to NMNEIGH and added error/warning messages to help clarify meaning for users.
- New kernel functions (SPHKERN in *SECTION_SPH) for 3D SPH.
  - SPHKERN = 1 (quintic spline kernel function) now supports FORMs 5,6. (Previously only supported for FORMs 0,1,9,10.
  - Added SPHKERN = 2 for FORMs 0,1,5,6. This quadratic spline kernel function mainly aims for high velocity impact to relieve the problem of compressive instability.

  - Added SPHKERN = 3 for FORMs 0,1,5,6. This quartic smoothing kernel function (G.R.Liu 2003) is similar to the standard cubic spline kernel function but should be more stable and accurate.

- *DEFINE_SPH_INJECTION: (both SMP and MPP):

  - Add injection plane moving with a node NID.
  - Add injection velocity based on a local coordinate system CID.

- Fix inexplicable penetration when using *CONTACT_AUTOMATIC_-NODES_TO_SURFACE with SPH elements. This affects SMP only.

• **Thermal Solver**

- *CONTROL_EXPLICIT_THERMAL_SOLVER and *ELEMENT_TSHELL: Implement heat conduction for thick shells in the explicit thermal solver.
- *CONTROL_EXPLICIT_THERMAL_SOLVER can use *INITIAL_TEMPERATURE to initialize the element temperatures (it averages the initial nodal temperatures at element centers).
- *CONTROL_EXPLICIT_THERMAL_INITIAL with ID < 0 allows initialization of temperature by element instead of element set (ID > 0).
- New SOLVER options in *CONTROL_THERMAL_SOLVER:

  EQ.18: preconditioner based on a local incomplete factorization;

  EQ.19: preconditioner based on a block low-rank factorization of the global problem.
- *CONTROL_THERMAL_SOLVER:

  - Fixed the echo of thermal solver options to d3hsp.
  - Enhanced the setting of default for GMRES parameters when nonsymmetric iterative was internally chosen over the input value for thermal linear equation solver.
  - Migrated the linear algebra data structures in thermal mechanics to Fortran 95 dynamic storage as part of the storage modernization effort. Reduces requirement of user on setting the memory command line parameter.
  - Declare all old SMP linear equation solvers for thermal as obsolete. Automatically switch to the new solvers with an appropriate warning. This is a step towards modernization of linear algebra for thermal.
  - Added nonsymmetric problem support to the modern linear algebra path in thermal. This is mostly the THERMAL_BULK feature.
  - Turned off support for direct linear equation solvers for thermal in single precision. In single precision change direct solver options to appropriate iterative solvers. Symmetric (11) goes to modified Incomplete Choleski PCG (16) which is the strongest symmetric iterative solver. Nonsymmetric (20) goes to GMRES (17).

    - Correct the case where cgtol is specified on *CONTROL_THERMAL_SOLVER but reltol is not.
    - Corrected testing of linear equation solver option for thermal. The addition of GMRES option for conjugate heat transfer problems had to be properly dealt with.

- Turn off all implicit mechanics if a thermal-only solution.
- *CONTROL_THERMAL_SOLVER: Fix a conflict associated with using the nonsymmetric direct solver (SOLVER = 30) for the thermal side of a coupled thermal/mechanical analysis when the mechanical analysis is implicit.
- *BOUNDARY_RADIATION_<option>_VF_READ requires a view factor file. An input trap was added if this view factor file is missing.
- There was a limit on *BOUNDARY_TEMPERATURE_NODE and *INITIAL_TEMPERATURE_NODE keyword appearances. This limit was removed, and the speed of the keyword reader was improved.
- Added new keyword *CONTROL_THERMAL_FORMING for easy setup of hot/warm forming simulations. This keyword sets default values for thermal control keywords so that there is no need to include *CONTROL_SOLUTION, *CONTROL_THERMAL_SOLVER and *CONTROL_THERMAL_TIMESTEP in the input deck unless fine tuning of simulation parameters is needed. This keyword can also be used to set up default thermal parameters for forming contact pairs.
- Added new keyword *BOUNDARY_TEMPERATURE_TRAJECTORY to apply a temperature boundary condition on nodes enclosed in either a cylindrical or rectangular prism volume moving along a trajectory. The center of the volume moves along the trajectory defined by a nodal path at a prescribed velocity. The geometry of the enclosing volume can be time-variant. This keyword applies only to solid elements.
- New keyword *BOUNDARY_FLUX_TRAJECTORY defines a moving flux surface boundary condition along a nodal path. Based on the shape of the heat source and its aiming direction, the projection onto the surface is calculated and used for application of the flux. Additional features include:

    - Propagation to newly exposed surfaces after element erosion.
    - Option to balance the change in projected area by modifying the surface density.
    - Total amount of heat input as resulting from the numerical integration of the surface densities can be enforced to coincide with the input.
    - Various heat distributions are available:
        - Constant distribution within a double elliptic region
        - Gaussian distribution within a double elliptic region
        - User-defined function (*DEFINE_FUNCTION) based on local coordinates, time, and temperature

- *LOAD_HEAT_GENERATION: Added variable REFNODE to define a reference node for the function definition. The current nodal coordinates can be referred to in the function definition as “xref”, “yref”, and “zref”. It allows defining a heat source motion associated with the motion of the reference node.
- Thermal contact for 'edge' contact of composite tshell elements now considers the composite stacking sequence by way of an internally constructed mesh of stacked tshell elements.

- **XFEM (eXtended Finite Element Method)**

  - *BOUNDARY_PRECRACK: Adjusted the location of pre-crack to avoid passing through nodal points.
  - Recoded neighbors list to handle triangular mesh better.
  - Added support of GISSMO damage model for XFEM. Set FAILCR = 0 in *SECTION_SHELL_XFEM to activate GISSMO model.
  - 2D XFEM shell (ELFORM = 52 in *SECTION_SHELL_XFEM) is a fully integrated form and so NIP = 4 is now hardwired.
  - Second order stress update can now be specified for 2D XFEM shell form 52 (OSU = 1 in *CONTROL_ACCURACY).

- **Miscellaneous**

  - *DAMPING_FREQUENCY_RANGE and *DAMPING_FREQUENCY_-RANGE_DEFORM had an error trap related to the time step being too large to compute damping at frequency FHIGH. The trap was unnecessary for *DAMPING_FREQUENCY_RANGE_DEFORM and has been removed. The trap was previously necessary for *DAMPING_FREQUENCY_RANGE to avoid instability, but now the code has been modified to remain unconditionally stable and the error trap has been removed. The code modification will result in some change of results for *DAMPING_FREQUENCY_-RANGE (but not for the _DEFORM option). The difference will be most noticeable when the product of the timestep with FHIGH is large, such as in Implicit calculations.
  - Added new command line option to run only a subset of the cases defined in the input deck via *CASE. As before, if all cases in a model should be run, the single word “CASE” should appear as a command line argument. But if only some cases should be run, those case ids can be listed on the command line like:

    ```
    CASE=17,22,147
    ```

    There should be NO spaces before or after the =, and the list following the = should be a comma delimited list of case ids, with no spaces.
  - LS-DYNA now returns an exit code of 1 to the system in case of non-normal termination. Previously, we always returned 0 (no error) no matter what, which could be misleading if a user's job script made use of the exit code.

- Suppress d3hsp output of any *PARAMETER that was encrypted in the input.
- Fix old bug in *INTERFACE_LINKING that would cause incorrect behavior for displacements scaled via a *DEFINE_FUNCTION if (and only if) the function was defined in terms of 3 variables (spatial displacement dependence only).
- The third variable in *PART_ADAPTIVE_FAILURE now does this:
  EQ.1: disable adaptivity of THIS part after it is split into two pieces
  EQ.2: disable ALL adaptivity after this part is split into two pieces
  The default value is 0, which does nothing.
- Add keyword *DEFINE_DRIFT_REMOVE to restore periodicity to specified curve(s) that physically should exhibit periodicity. For example, accelerometer data collected as a car drives one circuit around a track should exhibit periodicity such that when integrated to get displacement, that displacement is zero. However, noise in the data leads to nonphysical drift. This keyword activates an algorithm that approximately restores periodicity.
- *INTERFACE_LINKING_NODES: Interface nodes that were shared between processes were not properly dealt with for implicit.
- *DEFINE_CURVE_FUNCTION:
  - Fix a bug for SENSOR or SENSORD that may result in erroneous information in CURVOUT.
  - Fix a bug for function PIDCTL that occurred in a thermal-only analysis.
- *PART_DUPLICATE:
  - Fix a bug in which 10-noded tet elements were not duplicated.
  - Add option of BOXID to assure that the transformed configuration falls inside the box
- *SET_PART_TREE: Define a branch in a tree structure. A branch is a part set that can be defined using parts and/or sub-branches. With this keyword, the whole model can be modeled as a hierarchical tree structure.
- *SET_SHELL: allow *SET_SHELL_LIST_GENERATE to generate a set of 2D belts.
- New keyword *DEFINE_QUASAR_COUPLING to enable coupling of LS-DYNA and Cadlm's QUASAR ROM (Reduced Order Model). Supports multiple ROMs attached to LS-DYNA FEM model.
- Madymo coupling:
  - Support TASS Madymo coupling using Intel MPI.
  - Support Madymo 7.7 and above.
- Fixed outdated argument lists for user subroutines. Before this fix user-defined features using *MODULE may not work correctly.

- *DEFINE_PRESSURE_TUBE now supports decomposition of automatically generated solid/shell tubes in MPP.
- Added support for user element history variables in *USER_NONLOCAL_-SEARCH.
- Implement *DEFINE_DRIFT_REMOVE for MPP.
- Fix empty *SET_SEGMENT error, KEY+140, when using *SET_SEGMENT_-GENERAL with OPTION set to SET_SOL-ID/SET_SLDI0/SET_SLDF#/SET_TSHELL/SET_TSHIO.
- Fix seg fault when using *INTERFACE_LINKING_NODE_SET.
- New functions AX2, AY2, and AZ2 for *DEFINE_CURVE_FUNCTION. The difference compared to AX, AY, and AZ is that if n2 = 0, then return component relative to axes fixed in n1.
- New function DIST() in *DEFINE_CURVE_FUNCTION, which is used to calculate the distance traveled by a node
- Add feature to *DEFINE_MATERIAL_HISTORIES, LABEL="History" whereby if the first attribute A1 is negative, its absolute value points to a curve. The first ordinate value of that curve defines an operation to be performed on a list of history variables. The subsequent ordinate values give the history variable numbers in the list. See the User's Manual for details and an example.
- Add keyword *DEFINE_ELEMENT_EROSION_(SHELL/TSHELL) to allow more flexible control of when to actually delete an element.
- Add two new OPTIONs for *DEFINE_TRANSFORMATION.

  - TRANSL2ND: translation given by two nodes and a distance
  - ROTATE3NA: rotation given by three nodes and an angle (two nodes provide rotation axis, going through the third node)

- Add new option DTYPE = 1 to *PERTURBATION_NODE to allow uniform distribution between SCL × [−AMPL, AMPL] for TYPE = 8.
- Add runtime output to "Open include file: ..." instance.
- Removed the echo of encrypted data to d3hsp for the following keywords: *PARAMETER, *DEFINE_FRICTION, *ELEMENT_DISCRETE, *CONTACT, and *DEFINE_CURVE.
- Added three sense switches swb, swc and swd, all of which will create dynain data. *INTERFACE_SPRINGBACK_LSDYNA must be defined in the input deck.

  swb: a dynain is written and the analysis continues.
  swc: a restart and dynain are written and the analysis continues.
  swd: a restart and dynain are written and the analysis terminates.

  The first dynain file created by one of these sense switches is named dynain.1, and the filename is incremented by 1 (dynain.2, dynain.3, etc.) each time one of the sense switches is submitted.
- Replace keyword *SET_SPRING with *SET_DISCRETE.

- Changed the default of plabel to plabel = no for faster input processing. If the input contains non-numeric (character) ID(s), an error message will be issued, in which case the user would need to include "plabel = yes" on the execution line so that the non-numeric ID(s) could be properly read.
- Fix bug in which message announcing loading of an include files appeared twice.
- Fixed bug in madymo coupling in which the restart dump was called twice.
- Improve check for empty or duplicate *SETs and issue appropriate warnings/errors.
- Can now use an environment variable to set the default memory: setenv LSTC_DEFMEM n where n is the memory size
- Can now suppress almost all output by including "benchmark = y" on either the execution line or on the *KEYWORD line in the input deck. It will also remove part, node and element output in d3hsp.
- Fixed bug in reading long format if *KEYWORD long = yes is used in include file.
- A new flag TOL in *INCLUDE_UNITCELL is introduced for users to define the tolerance for identifying the pairs of nodes in the periodic positions.
- New keywords: *COSIM_FMI_CONTROL and *COSIM_FMI_INTERFACE: Adds capability to remotely co-simulate with other software supporting FMI standard.
- Deliver model order reduction (MOR) *CONTROLLER_PLANT for the linear structural and piezo-electric system. The state space matrices are exported based on the modal truncation or Krylov subspace.
- *PART_DUPLICATE: Add a new optional variable ZMIN. Transformed part(s) will have a minimum z-coordinate equal to ZMIN.
- *INCLUDE_STAMPED_PART_SOLID_TO_SOLID: This keyword extends the idea of *INCLUDE_STAMPED_PART to solid elements. Information about a solid part is mapped from one analysis (e.g., stamping) to another analysis (e.g., crash).

  - Maps the stress tensor, effective plastic strain, and strain tensor.
  - Calculates the thickness from the first analysis and modifies the coordinates of the nodes in the second model to conform to that thickness.
  - When PID is a negative number, it means that the top surface of the solid needs to be reversed to match the forming result. Otherwise, the mapping will be wrong.
  - When the element orientation is defined in *ELEMENT_SOLID_ORTHO, the angles are mapped to the new mesh.

- *PART_MOVE:

  - Maximum number of parts moved in one input increased from 400 to 1000.

- When CID is a negative value, it defines a vector id or VID pointing to the direction of the part move and the variable ZMOV is the move distance.

Capabilities added to create LS-DYNA R13:

See release notes located at https://ftp.lstc.com/anonymous/outgoing/support/FAQ/ReleaseNotes/Release_Notes_LS-DYNA_R13_0_0_rev2.pdf.

Capabilities added to create LS-DYNA R14:

See release notes located at https://ftp.lstc.com/anonymous/outgoing/support/FAQ/ReleaseNotes/Release_Notes_LS-DYNA_R14_0_0_rev1.html.

Capabilities added to create LS-DYNA R15:

See release notes located at https://ftp.lstc.com/anonymous/outgoing/support/FAQ/ReleaseNotes/Release_Notes_LS-DYNA_R15_0_0_rev0.html

MATERIAL MODELS

Some of the material models presently implemented are:

- elastic,
- orthotropic elastic,
- kinematic/isotropic plasticity [Krieg and Key 1976],
- thermoelastoplastic [Hallquist 1979],
- soil and crushable/non-crushable foam [Key 1974],
- linear viscoelastic [Key 1974],
- Blatz-Ko rubber [Key 1974],
- high explosive burn,
- hydrodynamic without deviatoric stresses,
- elastoplastic hydrodynamic,
- temperature dependent elastoplastic [Steinberg and Guinan 1978],
- isotropic elastoplastic,
- isotropic elastoplastic with failure,
- soil and crushable foam with failure,
- Johnson/Cook plasticity model [Johnson and Cook 1983],

- pseudo TENSOR geological model [Sackett 1987],
- elastoplastic with fracture,
- power law isotropic plasticity,
- strain rate dependent plasticity,
- rigid,
- thermal orthotropic,
- composite damage model [Chang and Chang 1987a 1987b],
- thermal orthotropic with 12 curves,
- piecewise linear isotropic plasticity,
- inviscid, two invariant geologic cap [Sandler and Rubin 1979, Simo et al, 1988a
- 1988b],
- orthotropic crushable model,
- Mooney-Rivlin rubber,
- resultant plasticity,
- force limited resultant formulation,
- closed form update shell plasticity,
- Frazer-Nash rubber model,
- laminated glass model,
- fabric,
- unified creep plasticity,
- temperature and rate dependent plasticity,
- elastic with viscosity,
- anisotropic plasticity,
- user defined,
- crushable cellular foams [Neilsen, Morgan, and Krieg 1987],
- urethane foam model with hysteresis,

and some more foam and rubber models, as well as many materials models for springs and dampers. The hydrodynamic material models determine only the deviatoric stresses. Pressure is determined by one of ten equations of state including:

- linear polynomial [Woodruff 1973],
- JWL high explosive [Dobratz 1981],
- Sack “Tuesday” high explosive [Woodruff 1973],

- Gruneisen [Woodruff 1973],
- ratio of polynomials [Woodruff 1973],
- linear polynomial with energy deposition,
- ignition and growth of reaction in HE [Lee and Tarver 1980, Cochran and Chan 1979],
- tabulated compaction,
- tabulated,
- TENSOR pore collapse [Burton et al. 1982].

The ignition and growth EOS was adapted from KOVEC [Woodruff 1973]; the other subroutines, programmed by the authors, are based in part on the cited references and are nearly 100 percent vectorized. The forms of the first five equations of state are also given in the KOVEC user's manual and are retained in this manual. The high explosive programmed burn model is described by Giroux [Simo et al. 1988].

The orthotropic elastic and the rubber material subroutines use Green-St. Venant strains to compute second Piola-Kirchhoff stresses, which transform to Cauchy stresses. The Jaumann stress rate formulation is used with all other materials with the exception of one plasticity model which uses the Green-Naghdi rate.

SPATIAL DISCRETIZATION

The elements shown in Figure 1-1 are presently available. Currently springs, dampers, beams, membranes, shells, bricks, thick shells and seatbelt elements are included.

The first shell element in DYNA3D was that of Hughes and Liu [Hughes and Liu 1981a, 1981b, 1981c], implemented as described in [Hallquist et al. 1985, Hallquist and Benson 1986]. This element [designated as HL] was selected from among a substantial body of shell element literature because the element formulation has several desirable qualities:

- It is incrementally objective (rigid body rotations do not generate strains), allowing for the treatment of finite strains that occur in many practical applications.
- It is compatible with brick elements, because the element is based on a degenerated brick element formulation. This compatibility allows many of the efficient and effective techniques developed for the DYNA3D brick elements to be used with this shell element;
- It includes finite transverse shear strains;
- A through-the-thickness thinning option (see [Hughes and Carnoy 1981]) is also available.

All shells in our current LS-DYNA code must satisfy these desirable traits to at least some extent to be useful in metalforming and crash simulations.

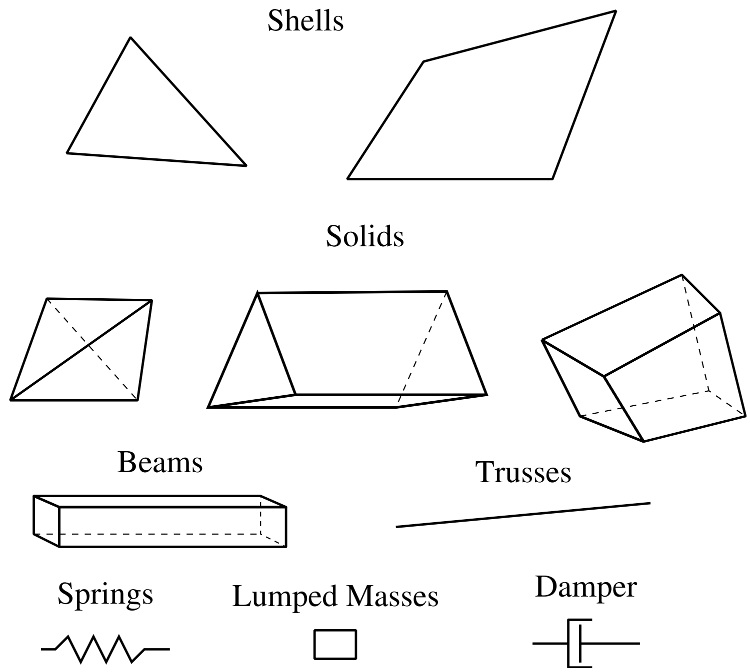

Figure 1-1. Elements in LS-DYNA. Three-dimensional plane stress constitutive subroutines are implemented for the shell elements which iteratively update the stress tensor such that the stress component normal to the shell midsurface is zero. An iterative update is necessary to accurately determine the normal strain component which is necessary to predict thinning. One constitutive evaluation is made for each integration point through the shell thickness.

The major disadvantage of the HL element turned out to be cost related and, for this reason, within a year of its implementation we looked at the Belytschko-Tsay [BT] shell [Belytschko and Tsay 1981, 1983, 1984] as a more cost effective, but possibly less accurate alternative. In the BT shell the geometry of the shell is assumed to be perfectly flat, the local coordinate system originates at the first node of the connectivity, and the co-rotational stress update does not use the costly Jaumann stress rotation. With these and other simplifications, a very cost effective shell was derived which today has become perhaps the most widely used shell elements in both metalforming and crash applications. Results generated by the BT shell usually compare favorably with those of the more costly HL shell. Triangular shell elements are implemented, based on work by Belytschko and co-workers [Belytschko and Marchertas 1974, Bazeley et al. 1965, Belytschko et al. 1984], and are frequently used since collapsed quadrilateral shell elements tend to lock and give very bad results. LS-DYNA automatically treats collapsed quadrilateral shell elements as C^0 triangular elements.

Since the Belytschko-Tsay element is based on a perfectly flat geometry, warpage is not considered. Although this generally poses no major difficulties and provides for an efficient element, incorrect results in the twisted beam problem and similar situations are obtained where the nodal points of the elements used in the discretization are not coplanar. The Hughes-Liu shell element considers non-planar geometries and gives good results on the twisted beam. The effect of neglecting warpage in a typical application cannot be predicted beforehand and may lead to less than accurate results, but the latter is only speculation and is difficult to verify in practice. Obviously, it would be better to use shells that consider warpage if the added costs are reasonable and if this unknown effect is eliminated. Another shell published by Belytschko, Wong, and Chiang [Belytschko, Wong, and Chiang 1989, 1992] proposes inexpensive modifications to include the warping stiffness in the Belytschko-Tsay shell. An improved transverse shear treatment also allows the element to pass the Kirchhoff patch test. This element is now available in LS-DYNA. Also, two fully integrated shell elements, based on the Hughes and Liu formulation, are available in LS-DYNA, but are rather expensive. A much faster fully integrated element which is essentially a fully integrated version of the Belytschko, Wong, and Chiang element, type 16, is a more recent addition and is recommended if fully integrated elements are needed due to its cost effectiveness.

Zero energy modes in the shell and solid elements are controlled by either an hourglass viscosity or stiffness. Eight node thick shell elements are implemented and have been found to perform well in many applications. All elements are nearly 100% vectorized. All element classes can be included as parts of a rigid body. The rigid body formulation is documented in [Benson and Hallquist 1986]. Rigid body point nodes, as well as concentrated masses, springs and dashpots can be added to this rigid body.

Membrane elements can be either defined directly as shell elements with a membrane formulation option or as shell elements with only one point for through thickness integration. The latter choice includes transverse shear stiffness and may be inappropriate. For airbag material a special fully integrated three and four node membrane element is available.

Two different beam types are available: a stress resultant beam and a beam with cross section integration at one point along the axis. The cross section integration allows for a more general definition of arbitrarily shaped cross sections taking into account material nonlinearities.

Spring and damper elements can be translational or rotational. Many behavior options can be defined, e.g., arbitrary nonlinear behavior including locking and separation.

Solid elements in LS-DYNA may be defined using from 4 to 8 nodes. The standard elements are based on linear shape functions and use one point integration and hourglass control. A selective-reduced integrated (called fully integrated) 8 node solid element is available for situations when the hourglass control fails. Also, two additional solid elements, a 4 noded tetrahedron and an 8 noded hexahedron, with nodal rotational degrees of freedom, are implemented based on the idea of Allman [1984] to replace the nodal midside translational degrees of freedom of the elements with quadratic shape functions by corresponding nodal rotations at the corner nodes. The latter elements, which do not need hourglass control, require many numerical operations compared to the hourglass controlled elements and should be used at places where the hourglass elements fail. However, it is well known that the elements using more than one point integration are more sensitive to large distortions than one point integrated elements.

The thick shell element is a shell element with only nodal translations for the eight nodes. The assumptions of shell theory are included in a non-standard fashion. It also uses hourglass control or selective-reduced integration. This element can be used in place of any four node shell element. It is favorably used for shell-brick transitions, as no additional constraint conditions are necessary. However, care has to be taken to know in which direction the shell assumptions are made; therefore, the numbering of the element is important.

Seatbelt elements can be separately defined to model seatbelt actions combined with dummy models. Separate definitions of seatbelts, which are one-dimensional elements, with accelerometers, sensors, pretensioners, retractors, and sliprings are possible. The actions of the various seatbelt definitions can also be arbitrarily combined.

CONTACT-IMPACT INTERFACES

The three-dimensional contact-impact algorithm was originally an extension of the NIKE2D [Hallquist 1979] two-dimensional algorithm. As currently implemented, one surface of the interface is identified as a reference surface and the other as a tracked surface. Each surface is defined by a set of three or four node quadrilateral segments, called reference and tracked segments, on which the nodes of the tracked and reference surfaces, respectively, must slide. In general, an input for the contact-impact algorithm requires that a list of segments be identified for two surfaces. In nonsymmetric contact, these are the tracked and reference surfaces. For symmetric contact, the algorithm is run twice so that each surface acts as both the tracked and reference surfaces. For the single surface algorithm only one surface is defined and each node in the surface is checked each time step to ensure that it does not penetrate through the surface. Internal logic [Hallquist 1977, Hallquist et al. 1985] identifies a reference segment for each tracked node and a tracked segment for each reference node and updates this information every time step as the tracked and reference nodes slide along their respective surfaces. Note that for general automatic definitions only parts/materials or three-dimensional boxes must be given. Then the possible contacting outer surfaces are identified by the internal logic in LS-DYNA. More than 20 types of interfaces can presently be defined including:

- sliding only for fluid/structure or gas/structure interfaces
- tied
- sliding, impact, friction
- single surface contact

- discrete nodes impacting surface
- discrete nodes tied to surface
- shell edge tied to shell surface
- nodes spot welded to surface
- tiebreak interface
- one way treatment of sliding, impact, friction
- box/material limited automatic contact for shells
- automatic contact for shells (no additional input required)
- automatic single surface with beams and arbitrary orientations
- surface to surface eroding contact
- node to surface eroding contact
- single surface eroding contact
- surface to surface symmetric constraint method [Taylor and Flanagan 1989]
- node to surface constraint method [Taylor and Flanagan 1989]
- rigid body to rigid body contact with arbitrary force/deflection curve
- rigid nodes to rigid body contact with arbitrary force/deflection curve
- edge-to-edge
- draw beads

Interface friction can be used with most interface types. The tied and sliding only interface options are similar to the two-dimensional algorithm used in LS-DYNA2D [Hallquist 1976, 1978, 1980]. Unlike the general option, the tied treatments are not symmetric; therefore, the surface which is more coarsely zoned should be chosen as the reference surface. When using the one-way slide surface with rigid materials, the rigid material should be chosen as the reference surface.

For geometric contact entities, contact has to be separately defined. Note that that for the contact of a rigid body with a flexible body, either the sliding interface definitions as explained above or the geometric contact entity contact can be used. Currently, the geometric contact entity definition is recommended for metal forming problems due to high accuracy and computational efficiency.

INTERFACE DEFINITIONS FOR COMPONENT ANALYSIS

Interface definitions for component analyses are used to define surfaces, nodal lines, or nodal points (*INTERFACE_COMPONENTS) for which the time histories of nodal coordinates and if applicable (*INTERFACE_LINKING_NODE/EDGE) and available (beams, shells), nodal rotations, are saved at some user specified frequency (*CONTROL_OUTPUT). This data may then be used to drive interfaces (*INTERFACE_LINKING) in subsequent analyses. This capability is especially useful for studying the detailed response of a small member in a large structure. For the first analysis, the member of interest need only be discretized sufficiently that the displacements and velocities on its boundaries are reasonably accurate. After the first analysis is completed, the member can be finely discretized and interfaces defined to correspond with the first analysis. Finally, the second analysis is performed to obtain highly detailed information in the local region of interest.

When starting the analysis, specify a name for the interface segment file using the Z = parameter on the LS-DYNA command line. When starting the second analysis, the name of the interface segment file (created in the first run) should be specified using the L = parameter on the LS-DYNA command line.

Following the above procedure, multiple levels of sub-modeling are easily accommodated. The interface file may contain a multitude of interface definitions so that a single run of a full model can provide enough interface data for many component analyses. The interface feature represents a powerful extension of LS-DYNA's analysis capability.

PRECISION

The explicit time integration algorithms used in LS-DYNA are in general much less sensitive to machine precision than other finite element solution methods. Consequently, double precision is not generally required. The benefits of this are greatly improved utilization of memory and disk. When problems have been found we have usually been able to overcome them by reorganizing the algorithm or by converting to double precision locally in the subroutine where the problem occurs. Particularly sensitive problems (e.g. some buckling problems, which can be sensitive to small imperfections) may require the fully double precision version, which is available on all platforms. Very large problems requiring more than 2 billion words of memory will also need to be run in double precision, due to the array indexing limitation of single precision integers.