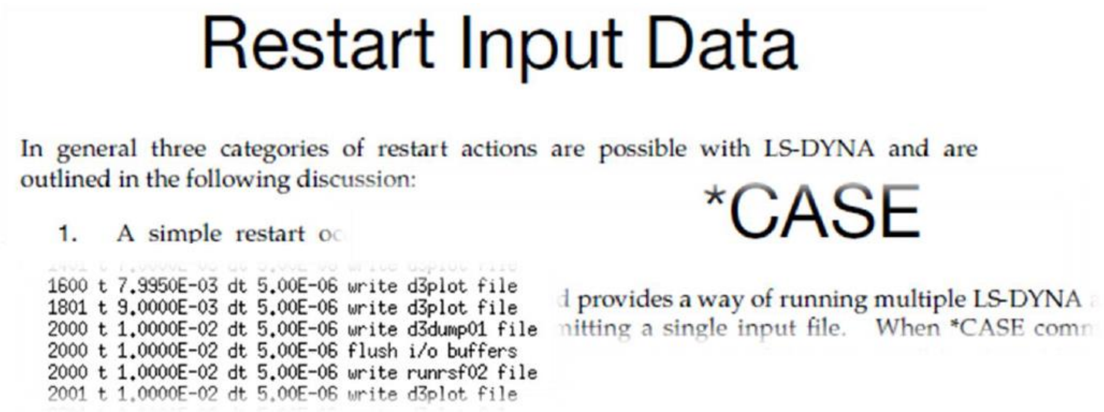
A brief review of the continuation of analyses in LS-DYNA using restart, dynain file and *CASE



Webinar 2020-03-10

Klas Engstrand, DYNAmore Nordic
Anders Jonsson, DYNAmore Nordic

Outline

- About continuation of analyses in LS-DYNA
- About restart
- Restart dump files
- Simple restart
- Small restart
- Full restart
- Sample 1: Apply a load on a prestressed model using full restart (explicit, mpp s R11.1.0)
- Sample 2: Apply a load on a prestressed model using the file dynain.lsda
  - Prestress is done implicit with double precision LS-DYNA (mpp d Dev).
  - Loading is done explicit with single precision LS-DYNA (mpp s Dev).
- Sample 3: The same as above but fully implicit and using `*CASE`.

About continuation of analyses in LS-DYNA

- A simulation can be continued in two ways:
  - Doing a restart from dump files of the initial simulation (d3dump, d3full etc.)
  - Using a dynain file from the initial simulation.
- A principal difference between a restart and a dynain approach is that
  - in a restart, the time continues from the previous analysis, but
  - using the dynain approach, the time starts from zero in all simulations.
- The above difference must be addressed when defining load curves and other time key points.
- Note that this presentation outlines the most common possibilities for continuation of analyses. Many options and possibilities are not covered.

About restart

- There are three types of restarts
  - Simple restart -> Continuation of an interactively stopped simulation with no changes.
  - Small Restart -> Continuation of a simulation where only specific changes are allowed.
  - Full restart -> Continuation of a simulation where more general changes are allowed.

- Prerequisites for restarts
  - Use the same LS-DYNA executable in the restart as in the original run.
  - Use the same number of CPU's in the restart as in the original run.
    (at least it increases the chances of success)

- Restarts are made from binary dump files.

- A dump file contains the complete record (stress, strain, deformation, contact forces, velocity etc) of a simulation at a specific time.

Dump files

- There are four types of dump files:
  - runrsf and d3dump (SMP LS-DYNA).
  - runrsf, d3dump, d3full and runfull (MPP LS-DYNA).

- d3dump/d3full files accumulate whereas runrsf are written over.
  - Note: one d3dump file per thread, whereas there is only one d3full file.

- runfull is equivalent to d3full but only written once, i.e. at termination.

- Dump files are written at cyclical interval according to `*DATABASE_BINARY_RUNRSF` (for runrsf) and `*DATABASE_BINARY_D3DUMP` (for d3dump and d3full).

- The number of runrsf files retained is controlled by the parameter NR in `*DATABASE_BINARY_RUNRSF` (default=1).

Dump files

- Example:

```
*DATABASE_BINARY_RUNRSF
$#     cycl            nr      beam       npltc      psetid
    1000.0              2         0           0           0
*DATABASE_BINARY_D3DUMP
$#     cycl            nr      beam       npltc      psetid
    2000.0              0         0           0           0
```

```
added mass           =    0.0000E+00
percentage increase  =    0.0000E+00

termination time                    = 5.000E-02
    200 t 9.9500E-04 dt 5.00E-06 write d3plot file
    400 t 1.9950E-03 dt 5.00E-06 write d3plot file
    600 t 2.9950E-03 dt 5.00E-06 write d3plot file
   1000 t 5.0000E-03 dt 5.00E-06 write runrsf01 file
   1001 t 5.0000E-03 dt 5.00E-06 write d3plot file
   1201 t 6.0000E-03 dt 5.00E-06 write d3plot file
   1401 t 7.0000E-03 dt 5.00E-06 write d3plot file
   1600 t 7.9950E-03 dt 5.00E-06 write d3plot file
   2000 t 1.0000E-02 dt 5.00E-06 write d3dump01 file
   2000 t 1.0000E-02 dt 5.00E-06 write runrsf02 file
   2201 t 1.1000E-02 dt 5.00E-06 write d3plot file
   2401 t 1.2000E-02 dt 5.00E-06 write d3plot file
   2601 t 1.3000E-02 dt 5.00E-06 write d3plot file
   3000 t 1.5000E-02 dt 5.00E-06 write runrsf01 file
   3001 t 1.5000E-02 dt 5.00E-06 write d3plot file
   3201 t 1.6000E-02 dt 5.00E-06 write d3plot file
   3401 t 1.7000E-02 dt 5.00E-06 write d3plot file
   3601 t 1.8000E-02 dt 5.00E-06 write d3plot file
   4000 t 2.0000E-02 dt 5.00E-06 write d3dump02 file
   4000 t 2.0000E-02 dt 5.00E-06 write runrsf02 file
```

```
adptmp       car.k              d3dump01.0000  d3dump04.0000  d3full01  d3full04  d3hsp     d3plot02  d3plot05  kill_by_pid       mes0000  runrsf01.0000
bg_switch    cont_profile.csv   d3dump02.0000  d3dump05.0000  d3full02  d3full05  d3plot    d3plot03  d3plot06  load_profile.csv  mkascii  runrsf02.0000
binout0000   cont_profile.xy    d3dump03.0000  d3dump06.0000  d3full03  d3full06  d3plot01  d3plot04  glstat    load_profile.xy   runfull  status.out
```

DYNA MORE NORDIC

Dump files

- Dump files written by default (at termination, and after dynamic relaxation)
  - SMP: d3dump
  - MPP: d3dump and d3full

- By adding `*DATABASE_BINARY_RUNRSF` the following files are also written
  - SMP: runrsf
  - MPP: runrsf and runfull

- Extra SMP feature
  - d=nodump on the command line suppresses D3DUMP.

- Extra MPP features
  - `*CONTROL_MPP_IO_NOD3DUMP` -> Suppresses output of d3dump and runrsf files.
  - `*CONTROL_MPP_IO_NOFULL` -> Suppresses output of d3full and runfull.
  - `*CONTROL_MPP_IO_NODUMP` -> Suppresses output of all dump files.

Simple restart

- A simulation has been stopped interactively (sw1, d3kil) and all we want now is to continue the simulation.
- No input changes are allowed.

Sample execution lines:

SMP: ls-dyna r=d3dump03
ls-dyna r=runrsf02

MPP: mppdyna r=d3dump03
mppdyna r=runrsf02

Small restart

- A simulation has been stopped interactively (sw1, d3kil) or by reaching termination time as specified on `*CONTROL_TERMINATION`.
- Now we want to continue the simulation but with some specific changes as explained by the manual, see chapter “Restart Input Data”.

CHANGE_*OPTION
*CONTROL_DYNAMIC_RELAXATION
*CONTROL_SHELL
*CONTROL_TERMINATION
*CONTROL_TIMESTEP
*DAMPING_GLOBAL
DATABASE_*OPTION
DATABASE_BINARY_*OPTION
DELETE_*OPTION
*INTERFACE_SPRINGBACK_LSDYNA
RIGID_DEFORMABLE_*OPTION
STRESS_INITIALIZATION_*{OPTION}
TERMINATION_*OPTION
*TITLE
*KEYWORD
*CONTROL_CPU
DEFINE_*OPTION
SET_*OPTION

Small restart

- A small restart needs a dump file <u>and</u> a small input file with restart input.

Sample execution lines:

SMP: ls-dyna i=restart_input.k r=d3dump06
ls-dyna i=restart_input.k r=runrsf02

MPP: mppdyna i=restart_input.k r=d3dump06
mppdyna i=restart_input.k r=runrsf02

```
*KEYWORD
*CONTROL_TERMINATION
$#  endtim    endcyc     dtmin    endeng    endmas     nosol
      0.06         0       0.0       0.0       0.0         0
*DATABASE_BINARY_D3PLOT
$#      dt      lcdt      beam     npltc    psetid
    0.0001         0         0         0         0
$#   ioopt
         0
*DELETE_PART
$#     id1       id2       id3       id4       id5       id6       id7       id8
         3         0         0         0         0         0         0         0
*END
```

Sample of a restart input file

Full restart

- If many changes are desired a full restart may be the appropriate choice.

- A full restart require a dump file and a full restart input deck, i.e. a complete model.
  - Input for retained nodes, elements, parts, material, contacts, loads etc. is copied directly from the original input deck.
  - The retained input deck, that is copied from the original input deck, can be modified as desired.
  - New parts, materials, contacts, loads, etc. can be added.
  - The command *STRESS_INITIALIZATION must be specified in the full restart deck in order for pre-existing parts to be initialized.
    - Initialized stress, strain, displacement etc. for all parts which are carried over or, optionally, for only a subset of those parts.

- Do not change the element connectivity (mesh topology) of retained elements.

Full restart

- Undeformed coordinates of retained nodes should appear in the `*NODE` data of the restart input deck.
  - I.e. the coordinates will be initialized (updated) according to data saved in dump file.

- `*DELETE` commands are just for small restarts.
  - To eliminate parts and elements in a full restart, omit their *PART and *ELEMENT data, respectively, in the full restart deck.

- Do not use `*INITIAL_VELOCITY` for nodes carried over from the previous run. Use `*CHANGE_VELOCITY_option` to modify velocities of such nodes.

- Preexisting contacts that are to be retained should include the _ID option so that the contact ID number in the original input deck match those in the full restart deck.

- *Note: `*CONTACT_AUTOMATIC_GENERAL` in a full restart deck is an exception in that it is treated as a brand new contact and is not initialized using data from d3full.*

Full restart

Sample execution lines:

SMP: ls-dyna i=model_step2.k r=d3dump06
ls-dyna i=model_step2.k r=runrsf02

Note!

MPP: mppdyna i=model_step2.k n=d3full06
mppdyna i=model_step2.k n=runfull

Note!

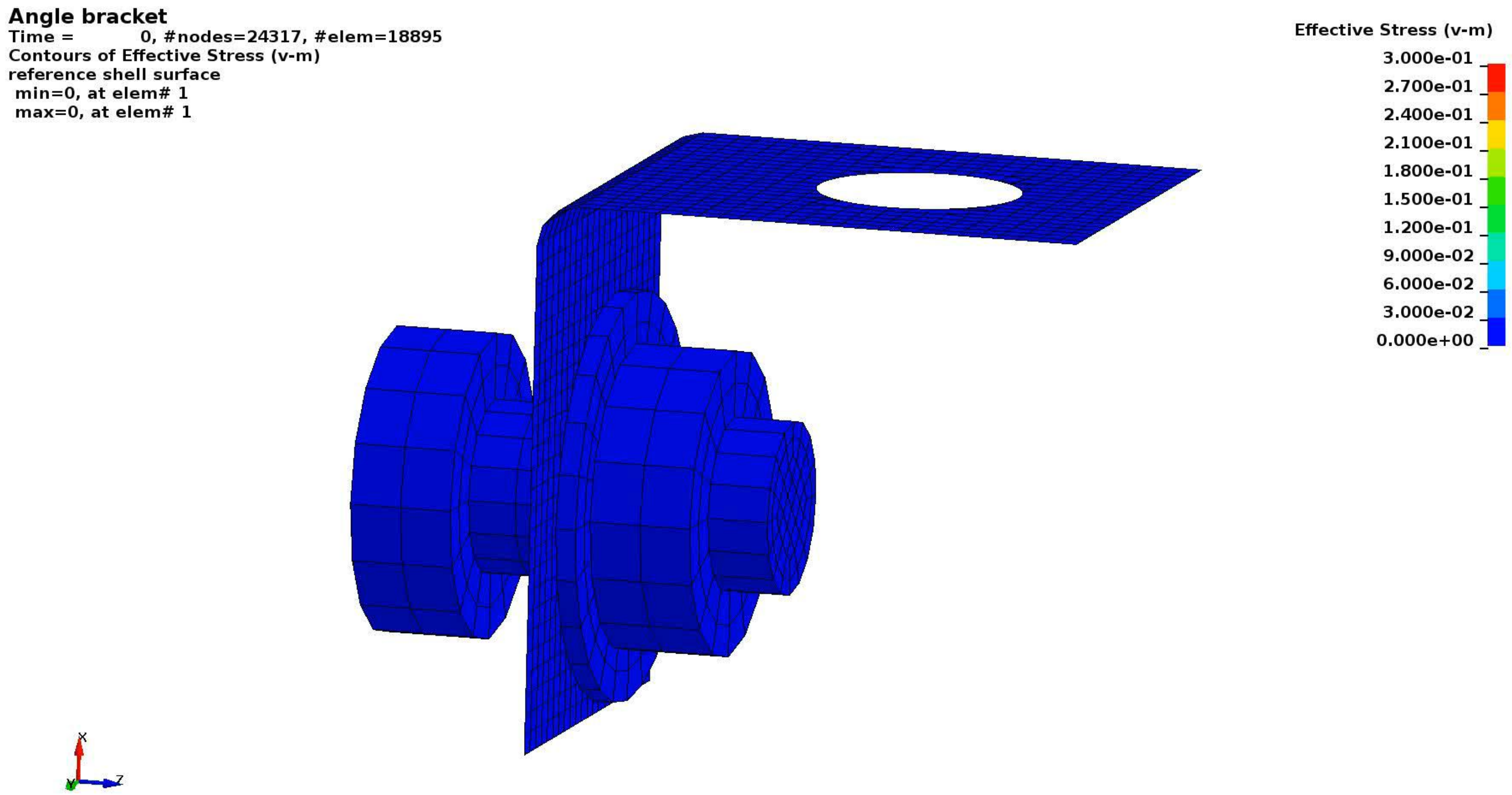
```
*KEYWORD
*STRESS_INITIALIZATION
*PARAMETER
$#   prmr1      val1     prmr2      val2     prmr3      val3     prmr4      val4
rtbumper  3.0       rthood     1.0
*PART
$#                                                                         title
material type # 3 (Kinematic/Isotropic Elastic-Plastic)
$#     pid     secid       mid     eosid      hgid      grav    adpopt      tmid
         1         1         1         0         1         0         0         0
*SECTION_SHELL
$#   secid    elform      shrf       nip     propt   qr/irid     icomp     setyp
         1         2       0.0         0       1.0         0         0         1
$#      t1        t2        t3        t4      nloc     marea      idof    edgset
       2.0       2.0       2.0       2.0       0.0       0.0       0.0         0
```

Sample 1: Apply a load on a prestressed model using full restart

- Procedure
  1. Run the prestress simulation in folder SIM00.
  2. Copy entire folder SIM00 to a new folder SIM01
     (so that the original prestress data is preserved in SIM00).
  3. Go to folder SIM01.
  4. Remove the prestress card from SIM00, i.e. `*INITIAL_STRESS_SECTION`.
     Increase the simulation time (`*CONTROL_TERMINATION`, ENDTIME).
     Add `*STRESS_INITIALIZATION`.
     Add a file where the load is defined using `*INCLUDE_TRANSFORM`.
     Start the restart simulation: mppdyna i=run_step2.key n=d3full01

Note: It was found that using an AUTOMATIC contact with SOFT=1 works fine.
However, currently SOFT=2 is not initiated correctly. The bug is reported to LSTC.

Sample 1: Apply a load on a prestressed model using full restart



Note: The board is here made invisible for demonstration purposes.

Sample 2: Apply a load on a prestressed model using dynain.lsda

- Procedure
  1. Run the prestress simulation in folder SIM00.
     - In this sample the prestress is performed using the implicit solver, but it could have been performed with the explicit solver as well.
     - The contact must be a MORTAR type contact in order to output the contact state for the subsequent simulation.
     - Add `*INTERFACE_SPRINGBACK_LSDYNA`.

For more complex models, with e.g. free nodes, set ndflag = 1 to dump all nodes in dynain file.

```
*INTERFACE_SPRINGBACK_LSDYNA
$#    psid      nshv     ftype         _    ftensr    nthhsv     rflag   intstrn
         3       999         3         0         0         0         1         0
$#    optc      sldo      ncyc    fsplit    ndflag     cflag     hflag
OPTCARD            0         0         0         0         1         1
```

LSDA format

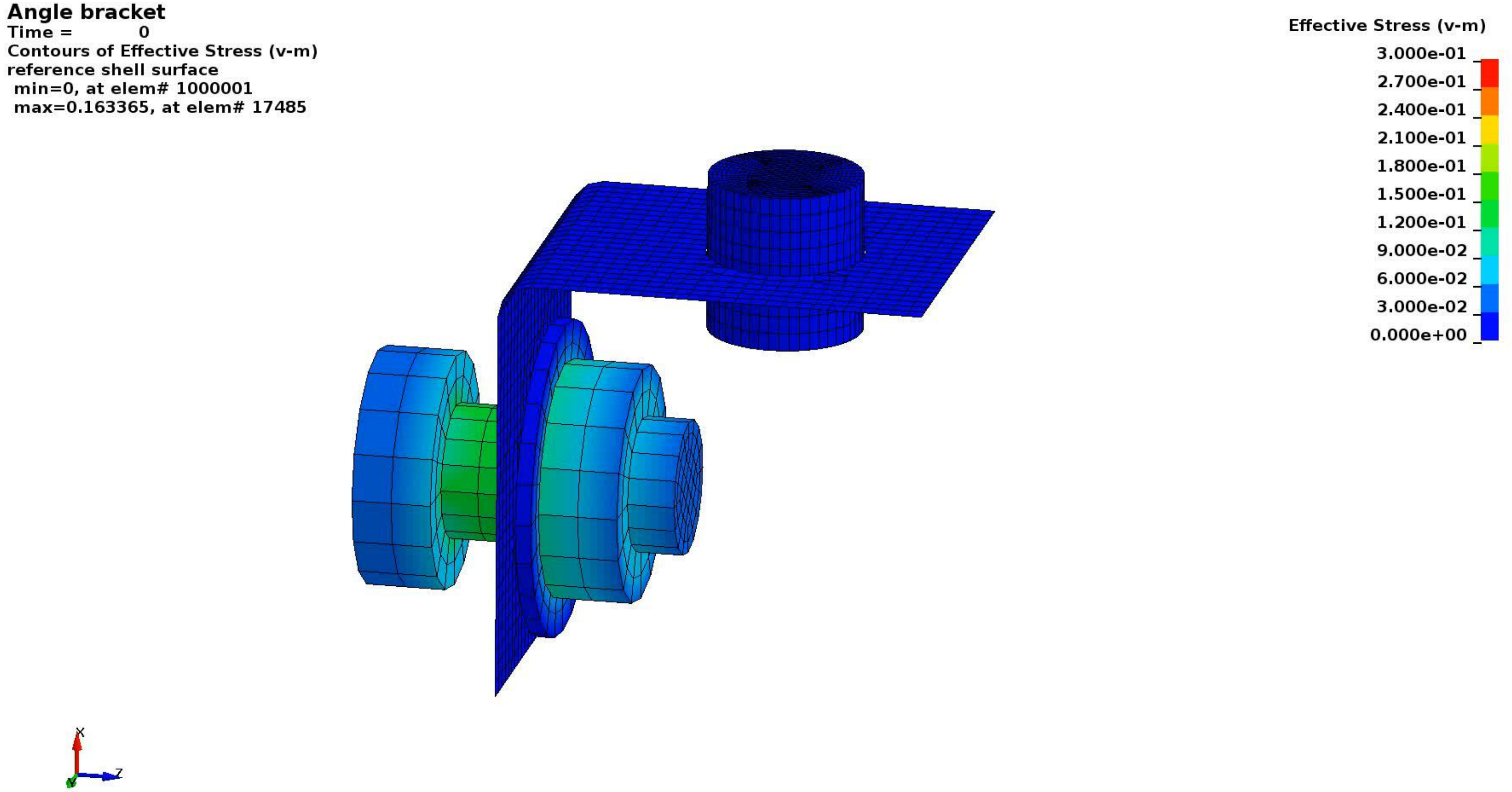
Output contact state

```
*INTERFACE_SPRINGBACK_EXCLUDE
$#                                                                    kwdname
BOUNDARY_SPC_NODE
```

Nodal constraints are removed from dynain.lsda

  2. Run the subsequent loading simulation in folder SIM01.
     - copy dynain.lsda from SIM00 to SIM01 and add it to the model deck, i.e. by `*INCLUDE`.
     - The model deck shall include everything except retained nodes and elements because they are already in dynain.lsda.
     - Note that the time will start at t=0.

Sample 2: Apply a load on a prestressed model using dynain.lsda



Note: The board is here made invisible for demonstration purposes.

Sample 3: Apply a load on a prestressed model using dynain.lsda and *CASE

- By the use of the dynain.lsda combined with the *CASE - feature, analyses of subsequent load steps can be divided into a sequence of LS-DYNA analyses, one per load step.
- By the *CASE feature, the LS-DYNA analyses are run in a sequence automatically.

- Procedure
  - The load step definition for load step *N* begins with the keyword
    CASE_BEGIN_*N
  - Parts that have no previous data from dynain.lsda - files should be included with their complete definition (*NODE, *ELEMENT_..., *PART, etc.) in keyword format
  - For parts analyzed in previous *CASEs, the model deck shall include everything except retained nodes and elements because they are already in the dynain.lsda - file.
  - Information is propagated from *CASE_*N* to *CASE_*N+1* by case*N*.dynain.lsda, by use of *INCLUDE and *INTERFACE_SPRINGBACK_LSDYNA.
  - The load step definition for load step *N* ends with the keyword
    CASE_END_*N

Sample 3: Apply a load on a prestressed model using dynain.lsda and *CASE

- Procedure (cont.)
  - For each *CASE, the time starts at t=0.
  - Output files from *CASE_*N* will have prefix case*N*, for example case*N*.d3hsp, case*N*.d3plot, caseN.binout0000, etc. so all the analyses can be performed in the same directory.
  - Keywords outside *CASE_BEGIN / *CASE_END definitions will be common to all *CASEs.
  - NOTE! The word “case” must be present on the LS-DYNA command line for the *CASE procedure to work.
    `mppsub 4 ls-dyna-version i=run_case.k memory=201m memory2=50m` **`case`**
  - It may also be required to create a “fake” case1.dynain.lsda (depends on the submit script)
  - Note: As from dev. LS-DYNA (SVN Version 142594) it is possible to run a subset of cases by e.g ”case=10,20,47,82”. The simulations will be run in increasing numeric order regardless of the order given on the command line.
  - A keyword template follows:

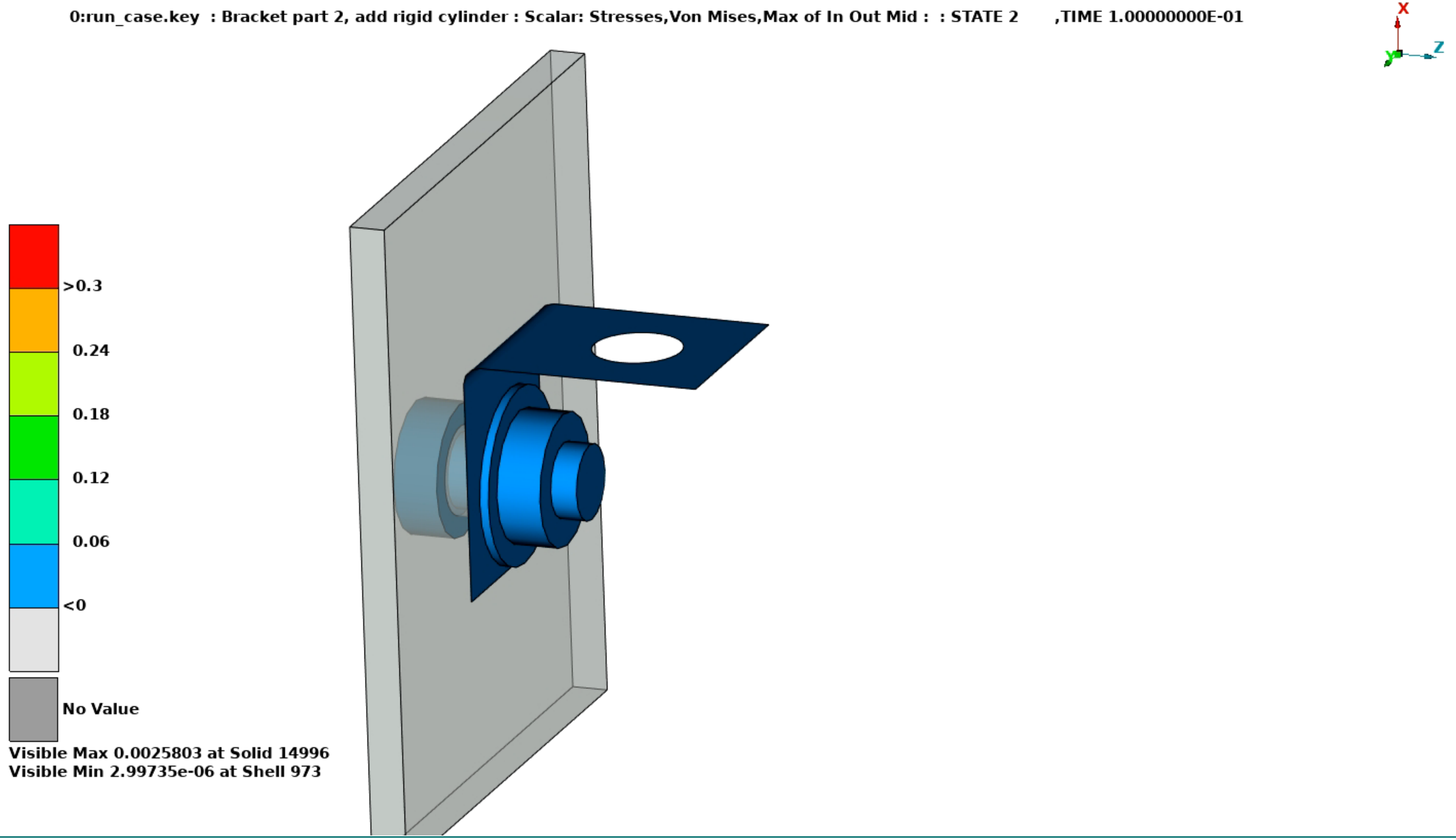
Sample 3: Apply a load on a prestressed model using dynain.lsda and *CASE

Procedure (cont.)

```
*KEYWORD
$===============================================================================
$ Common
$===============================================================================
*INCLUDE
common_control_cards.key
$===============================================================================
$ CASE1: Pre-tens
$===============================================================================
*CASE_BEGIN_1
*INCLUDE
complete_geo.key
$===============================================================================
$ Control and output cards
$===============================================================================
*CONTROL_IMPLICIT_DYNAMICS
$     IMASS     GAMMA      BETA    TDYBIR    TDYDTH    TDYBUR     IRATE
          1       0.6      0.38                 6.0       8.0         2
*CONTROL_TERMINATION
$#   endtim    endcyc     dtmin    endeng    endmas     nosol
       10.0
$===============================================================================
$ Additional output options
$===============================================================================
*INTERFACE_SPRINGBACK_LSDYNA
$#     psid      nshv     ftype         _    ftensr    nthhsv     rflag   intstrn
          3       999         3         0         0         0         1         0
$#     optc      sldo      ncyc    fsplit    ndflag     cflag     hflag
OPTCARD           0         0         0         0         1         1
*INTERFACE_SPRINGBACK_EXCLUDE
$#                                                                        kwdname
BOUNDARY_SPC_NODE
$===============================================================================
$ Simulation title
$===============================================================================
*TITLE
Bracket part 1, pre-tension
*CASE_END_1
```

```
*CASE_BEGIN_2
*INCLUDE
geo_part_mat_sec_mortar_spc.k
*INCLUDE
case1.dynain.lsda
*INCLUDE
additional_geo.k
$===============================================================================
$ Contacts
$===============================================================================
*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR_ID
                .
                .
                .
$===============================================================================
$ Loads and boundary conditions
$===============================================================================
*BOUNDARY_PRESCRIBED_MOTION_RIGID
                .
                .
$===============================================================================
$ Control and output cards
$===============================================================================
*CONTROL_TERMINATION
      10.0
*CONTROL_IMPLICIT_DYNAMICS
$     IMASS     GAMMA      BETA    TDYBIR    TDYDTH    TDYBUR     IRATE
          1       0.6      0.38                                       2
*CONTROL_IMPLICIT_SOLUTION
                .
                .
                .
$===============================================================================
$ Additional output options
$===============================================================================
*INTERFACE_SPRINGBACK_LSDYNA
                .
                .
*INTERFACE_SPRINGBACK_EXCLUDE
BOUNDARY_SPC_NODE
$===============================================================================
$ Simulation title
$===============================================================================
*TITLE
Bracket part 2, add rigid cylinder
*CASE_END_2
*END
```

Sample 3: Apply a load on a prestressed model using dynain.lsda and *CASE



Thank you!

Your LS-DYNA distributor and more