New Options in Frequency Domain Analysis and Fatigue Analysis with LS-DYNA®

Yun Huang, Zhe Cui

Livermore Software Technology Corporation

Abstract

A series of frequency domain analysis and fatigue analysis features have been implemented to LS-DYNA, since version 971 R6 [1]. The frequency domain features include FRF (Frequency Response Function), SSD (Steady State Dynamics), random vibration, response spectrum analysis, and acoustic analysis based on BEM (Boundary Element Method) and FEM (Finite Element Method). The fatigue analysis features include fatigue damage solvers in both time domain and frequency domain (based on random vibration and steady state vibration). The main applications of these features are in NVH and durability analysis of structures and components [2].

A bunch of new options were implemented to the frequency domain analysis and fatigue analysis features since the last European LS-DYNA Conference in Salzburg, Germany, 2017.

They include

- Direct SSD using sparse matrix solvers
- A new _FRF option to SSD, to provide fringe plot for FRF analysis
- Modal contribution output for SSD
- A data reduction method for PSD load curve, to accelerate random vibration computation
- DDAM (Dynamic Design Analysis Method) analysis
- Expanded d3spcm database to include mode contribution
- New mean stress correction methods for fatigue analysis
- Load steps definition in fatigue analysis
- Fatigue damage evolution and fatigue failure simulation in time domain

Some of the new options were implemented according to suggestions from the users. These new options not only enhance the existing frequency domain analysis and fatigue analysis features, but also open the window for new applications of these analysis features.

This paper gives a brief review of these new options for frequency domain analysis and fatigue analysis with LS-DYNA. The plan for future development in this area is also discussed.

1 Direct SSD using sparse matrix solvers

In comparison with mode-based SSD, the direct SSD computes the structural dynamic response in physical space directly. The unknown variables are nodal displacement vectors for each excitation frequency. The governing equation is

$$Ma + Cv + Ku = F \tag{1}$$

Where, *M*, *C* and *K* are global mass, damping and stiffness matrices, *a*, *v* and *u* are nodal acceleration, velocity and displacement vectors and *F* is the external loading vector. Generally speaking everything in equation (1) can be a function of *f*, the excitation frequency, except for the mass matrix M.

Assuming that the loading is a time harmonic function, the resulted equation system can be given as follows in frequency domain.

$$Au = F \tag{2}$$

Where,

$$A = -\omega^2 M + i\omega C + K \tag{3}$$

And *A* is a sparse symmetric complex matrix.

To solve the resulted equation system, a constrained linear system solver in LS-DYNA is adopted. Three options are available: 1) “MF2 direct factor and solve”; 2) Conjugate Gradient method and 3) GMRES method. So far, the “MF2 direct factor and solve” is used and the other two are under study.

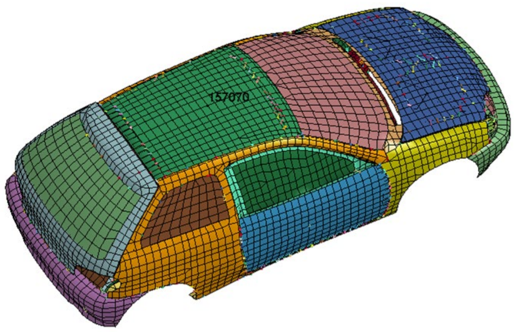


Fig.1: A simplified car model with nodal force excitation

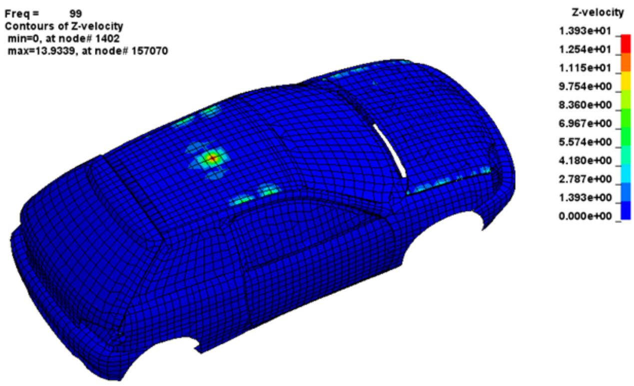


Fig.2: Z velocity response at frequency 99 Hz, by direct SSD

For the simplified car model shown in Figure 1, a harmonic nodal force is applied on the roof (at node 157070), for the frequency range 1-100 Hz. This problem was solved by using mode-based SSD and direct SSD. For mode-based SSD, 4 different sets of participating modes were tried. The more modes are participating in the mode superposition, the more accurate the results are. Table 1 compares the velocity response at the node 157070, for the excitation frequency 99 Hz, given by mode-based SSD and direct SSD.

Method	No. of modes	Highest freq (Hz)	v_z (mm/s)	Difference (%)	CPU time (s)
Mode based SSD	500	125.94	0.123	99.1%	56.73
	1000	194.57	8.533	38.8%	113.56
	2000	298.83	12.233	12.2%	260.05
	5000	658.69	14.091	1.1%	661.14
Direct SSD	N/A	N/A	13.934	0	117.39

Table 1: Z-velocity at node 157070 by mode based SSD and direct SSD

Note: the difference percentage is calculated as

$$\frac{\left|v - v_{ref}\right|}{v_{ref}} \times 100\% \tag{4}$$

Where, the z-velocity given by direct SSD (13.934 mm/s) is taken as the reference value v_{ref}.

The rule of thumb for frequency response analysis is that one should use enough modes to cover 1.2 times the maximum frequency in excitation. When 500 modes are used, the highest natural frequency is 125.94 Hz which is more than 1.2 times the maximum frequency in excitation (100 Hz). But for this problem, the z-velocity given by mode based SSD with 500 modes is only 0.123 mm/s, which is only around 1% of the reference value. To reach 99% accuracy, 5000 modes are needed but it took almost 6 times of CPU time required by direct SSD.

One important reason for CPU cost saving by direct SSD is that with direct SSD, the modal analysis step is skipped, which could be expensive otherwise.

Another advantage of direct SSD is that frequency dependent material properties can be considered, when using both the keywords **`*FREQUENCY_DOMAIN_SSD_DIRECT_FREQUENCY_DEPENDENT`** and **`*MAT_ADD_PROPERTY_DEPENDENCE`**. The keyword **`*MAT_ADD_PROPERTY_DEPENDENCE`** can change a constant property of a material model to be time or frequency dependent (the property is now defined by a time or frequency dependent curve).

With both mode-based SSD and direct SSD available in LS-DYNA, users can have more choices when they need to solve a frequency response analysis problem.

2 Option _FRF for *FREQUENCY_DOMAIN_SSD

FRF (Frequency Response Function) provides transfer function of a structure from load to response for a given range of frequencies. In other words, it is a ratio between response and load. It is a complex variable as both amplitude and phase angle are needed. To calculate FRF, unit load is assumed for the structure for the whole range of frequency. The results of FRF analysis are saved as ASCII xyplot files frf_amplitude and frf_angle, for selected nodes.

Sometimes user may want to see the fringe plot of FRF on whole structure. In this case, the keyword **`*FREQUENCY_DOMAIN_SSD_FRF`** can be used. SSD analysis is performed on the whole model, with input spectrum defined as unit load internally. The results are saved in binary plot database d3frf, which is similar to d3ssd and can be accessed by LS-PrePost.

Figure 3 shows xy-stress FRF (amplitude) fringe plot for a side frame model, subjected to nodal force excitation, at frequency 120 Hz. This plot is given by binary plot database d3frf.

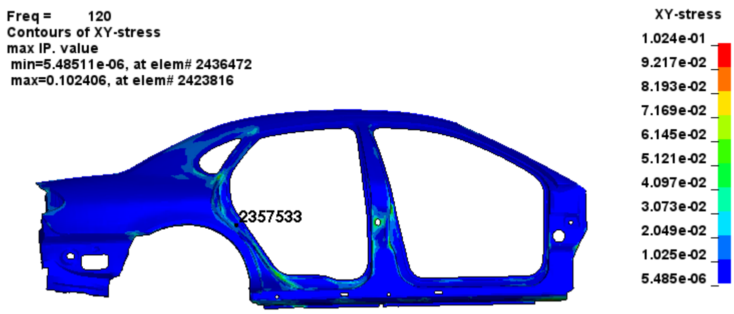


Fig.3: XY stress FRF fringe plot at frequency 120 Hz

With d3frf, user can check fringe plot of FRF results and locate hot-spot on the whole structure (e.g., the stress concentration area), for a given frequency.

The keyword setting for **`*FREQUENCY_DOMAIN_SSD_FRF`** is shown below. One can see, load curves lc1 and lc2 are blank as LS-DYNA assumes that unit load is used for the whole range of frequency, when option **`_FRF`** is used for **`*FREQUENCY_DOMAIN_SSD`**. With the option **`_FRF`**, the result database d3ssd is automatically renamed as d3frf, to remind user that the results in this database should be viewed as transfer function between load and response.

```
*FREQUENCY_DOMAIN_SSD_FRF
$#    mdmin     mdmax     fnmin     fnmax    restmd    restdp    lcflag    relatv
          1       100        0.     2000.         0         0
$#    dampf     lcdam     lctyp    dmpmas    dmpstf
       0.01
$#             istress    memory      nerp    strtyp      nout     notyp      nova

$#      nid      ntyp       dof       vad       lc1       lc2        sf       vid
    2357533         0         2         0
*DATABASE_FREQUENCY_BINARY_D3SSD
$#   binary
          1
$#     fmin      fmax     nfreq    fspace    lcfreq
        10.      140.        14
```

*Fig.4: Keyword setting for *FREQUENCY_DOMAIN_SSD_FRF*

3 Modal contribution output for *FREQUENCY_DOMAIN_SSD

SSD results are obtained by superposition of contribution from a bunch of modes, defined by mdmin, mdmax, or fnmin and fnmax in **`*FREQUENCY_DOMAIN_SSD`**. For example, the displacement *U* is given as

$$U = \sum_{i=1}^{N} c_i \phi_i = \sum_{i=1}^{N} u_i \tag{5}$$

Where, c_i and ϕ_i are the modal coordinate and modal shape for mode *i*. u_i is the contribution to total displacement response from mode *i*. *N* is the number of modes used.

The nodal and element results (nodal displacement, velocity and acceleration, and element stresses and strains) can be saved in ASCII databases nodout_ssd and elout_ssd (using keywords **`*DATABASE_FREQUENCY_ASCII_NODOUT_SSD`**, and **`*DATABASE_FREQUENCY_ASCII_ELOUT_SSD`**). A new option **`_MODAL_CONTRIBUTION`** has been enabled for the two keywords, so that the modal contribution ratio from each participating mode can be added to the two databases.

The modal contribution ratio is the percentage ratio of projected mode response vector on total response vector, as depicted in Figure 5.

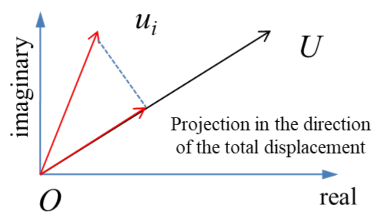


Fig.5: Projecting mode-i response vector to total response vector

A rectangular plate is considered as an example. The plate is subjected to nodal force excitation in normal direction. The nodal force is applied at node 131, as shown in Figure 6. The contribution ratio from each mode on the total z-displacement is requested.

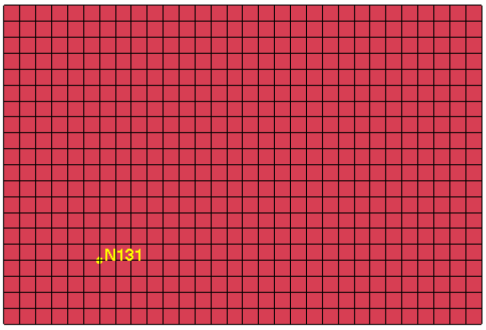


Fig.6: A rectangular plate, subjected to nodal force excitation in normal direction

The keyword setting for running this problem is shown in Figure 7.

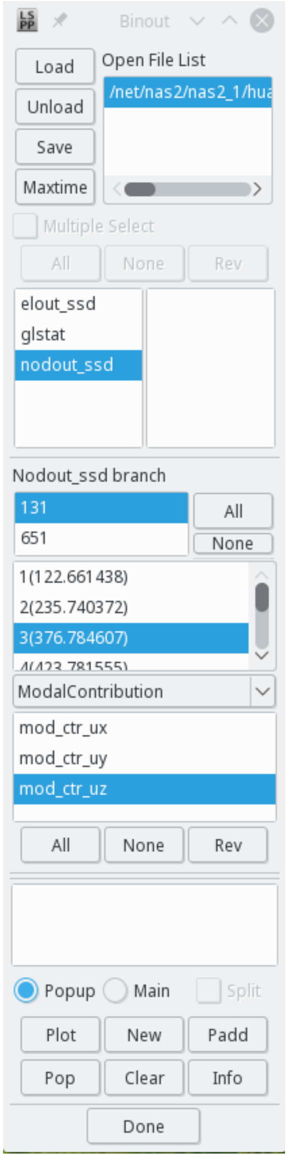
```
*FREQUENCY_DOMAIN_SSD
$#   mdmin     mdmax     fnmin     fnmax    restmd    restdp    lcflag    relatv
         1       100        0.     2000.         0         0
$#   dampf     lcdam     lctyp    dmpmas    dmpstf
      0.01
$#                                                    nout     notyp      nova

$#     nid      ntyp       dof       vad       lc1       lc2    lcflag       vid
       131         0         3         0       100       200
*DATABASE_FREQUENCY_BINARY_D3SSD
$#  binary
         2
$#    fmin      fmax     nfreq    fspace    lcfreq
        1.     1001.       101
*DATABASE_FREQUENCY_ASCII_NODOUT_SSD_MODAL_CONTRIBUTION
$#    fmin      fmax     nfreq    fspace    lcfreq
        1.     1001.      1001
*DATABASE_FREQUENCY_ASCII_ELOUT_SSD_MODAL_CONTRIBUTION
$#    fmin      fmax     nfreq    fspace    lcfreq
        1.     1001.      1001
```

Fig.7: Keyword setting for running SSD with modal contribution output

With option **_MODAL_CONTRIBUTION** added to ***DATABASE_FREQUENCY_ASCII_NODOUT_SSD** and ***DATABASE_FREQUENCY_ASCII_ELOUT_SSD**, modal contribution ratio (percentage) are computed and saved in noudout_ssd and elout_ssd (in binout) for nodal results (displacement, velocity and acceleration) and element results (stress and strain) respectively, and can be extracted and plotted using LS-PrePost.

Figure 8 shows the interface to extract modal contribution ratio data from nodout_ssd or elout_ssd in binout. User can choose from the list of nodes (or elements). When “ModalContribution” is selected, user can choose for which response variable the modal contribution is to be shown, like “mod_ctr_ux”, or “mod_ctr_uy”, etc. for the modal contribution for displacement component in x, or y direction. User also needs to choose for which mode the modal contribution ratio is to be shown. The modes are shown as mode ID with its natural frequency followed in a parenthesis, like 1(122.661438) for mode 1 in Figure 8. After that, user can click “Plot” button and a xyplot curve shows up like the one in Figure 9.



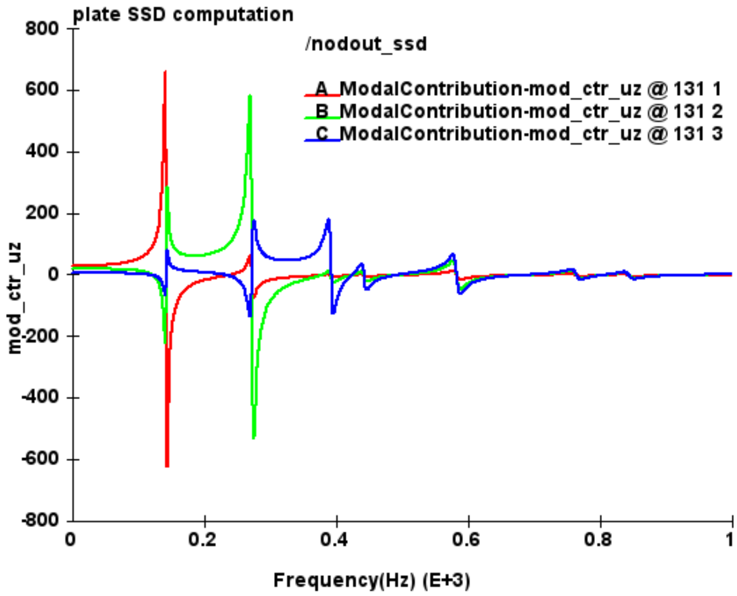


Fig.8: Interface to extract modal contribution ratio data

Fig.9: Modal contribution ratio (%) of the first three modes for z-displacement at node 131.

One can see that sometimes the modal contribution ratio is a negative number. This is because the response vector from that mode has negative projection on the total response vector. In other word, the contribution from that mode on the total response is negative. Similarly the contribution ratio from some modes may be larger than 100%.

4 Data reduction for PSD load in random vibration

For random vibration analysis, RMS (Root Mean Square) value of response is obtained based on integration of PSD (Power Spectral Density) results. If the PSD load curve has a large number of points, the integration can be time costly.

To speed up the integration, the original PSD load curve can be coarsened and the number of points can be reduced without significant loss of accuracy. This is achieved by setting the new parameters icoarse and tcoarse in ***FREQUENCY_DOMAIN_RANDOM_VIBRATION**.

Two options available for the data reduction:

- Coarsening original PSD load curve by keeping only peaks and troughs;

- Coarsening original PSD load curve by removing intermediate points whose slope change percentage is less than a pre-defined tolerance (tcoarse, whose default is 10%)

```
*FREQUENCY_DOMAIN_RANDOM_VIBRATION
$#    mdmin     mdmax     fnmin     fnmax    restrt              restrm
          1       100        0.    90000.
$#    dampf     lcdam     lctyp    dmpmas    dmpstf    dmptyp
       0.03
$#   vaflag    method      unit      umlt     vapsd     varms     napsd     ncpsd
          1         1         0                   1         1
$#    ldtyp   ipanelu   ipanelv    temper              dsflag   icoarse   tcoarse
                                                                       2
$#      sid     stype       dof     ldpsd     ldvel     ldflw     ldspn       cid
                                3      2000
*DATABASE_FREQUENCY_BINARY_D3PSD
$#   binary
          1
$#     fmin      fmax     nfreq    fspace    lcfreq
      0.100     2.000         5
*DATABASE_FREQUENCY_BINARY_D3RMS
$#   binary
          1
```

*Fig.10: Keyword setting of *FREQUENCY_DOMAIN_RANDOM_VIBRATION when PSD load curve coarsening is used*

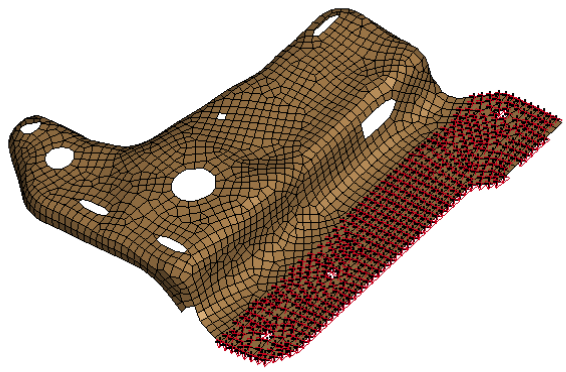


Fig.11: A bracket model constrained to shaker table, subjected to z direction base acceleration PSD

For demonstration, a metal bracket model shown in Figure 11 is used as an example. The model is constrained to a shaker table and is subjected to PSD acceleration from the shaker table. The PSD load curve has 10000 points.

Figures 12 to 14 show the fringe plot of RMS values of Von Mises stress response, based on 1) using original PSD curve; 2) using PSD curve coarsening option 1 - keeping only peaks and troughs; and 3) use PSD curve coarsening option 2 - removing intermediate points whose slope change is less than 10%.

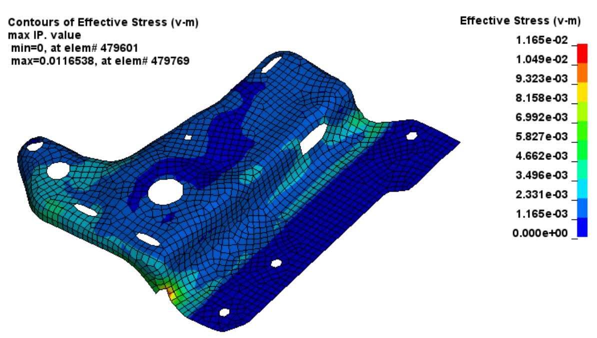


Fig.12: RMS of Von Mises stress, with original PSD curve

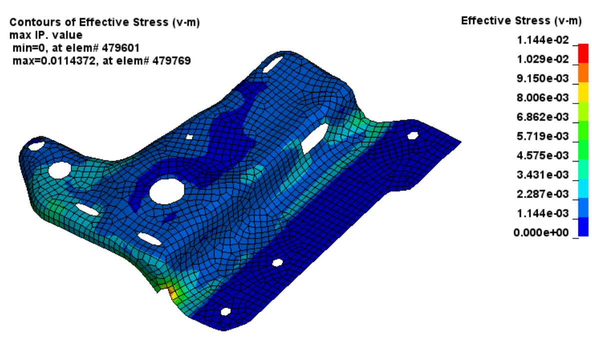


Fig.13: RMS of Von Mises stress, with icoarse = 1

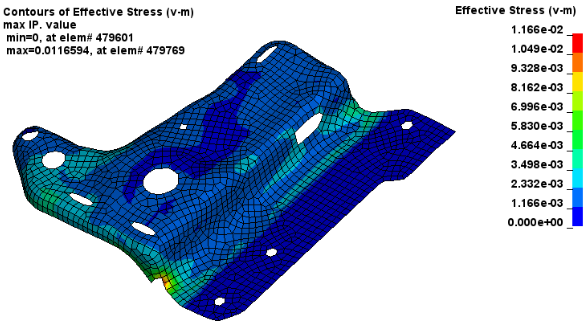


Fig.14: RMS of Von Mises stress, with icoarse = 2

One can see, with icoarse= 1 or 2, less PSD points were used in random vibration computation. Especially for icoarse = 1, around 1/3 of the data in PSD curve were removed. However, with icoarse = 1 or 2, the results are still accurate and are very close to the results given by computation with original PSD curve.

	Original PSD curve	**icoarse = 1**	**icoarse = 2**
No. of points in PSD curve	10000	6579	9821
CPU time (seconds; 1 core)	15134	6790	14211
Max RMS Von Mises stress (MPa)	1.165e-2	1.144e-2	1.166e-2
Element ID of max value	479769	479769	479769

Table 2: Comparison of the results with and without PSD curve coarsening

With the above table, and with the Figures 12-14, one can see that the results given by the three options have a good match. The location for the max value of RMS of Von Mises stress is same. But using PSD data coarsening options (icoarse = 1 or 2) the CPU time is much less than that using the original PSD curve. With icoarse = 1, the CPU time is less than half of the CPU time for running with original PSD curve.

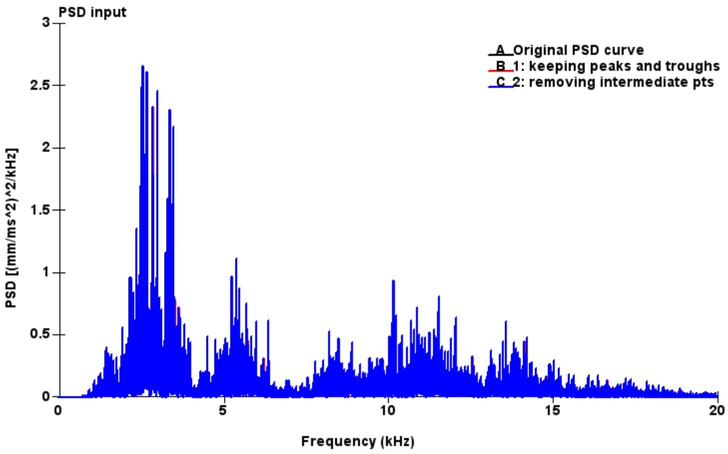


Fig.15: Original and coarsened PSD curves

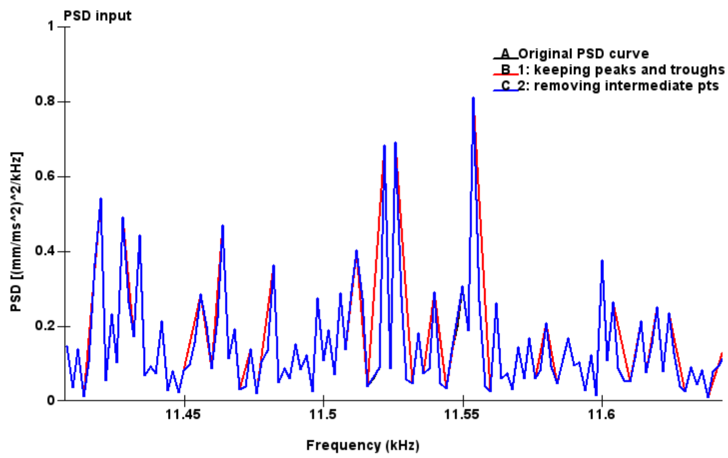


Fig.16: Original and coarsened PSD curves (detailed view for smaller frequency range)

From Figures 15 and 16 one can see that even some points were removed from the original PSD curve, the coarsened PSD curve still has a good and close match to the original one. Particularly the curves A and C are almost overlapping with each other.

5 DDAM analysis

A new option **_DDAM** has been added to the keyword ***FREQUENCY_DOMAIN_RESPONSE_SPECTRUM**, to run DDAM (Dynamic Design Analysis Method). The purpose of this feature is to provide safety evaluation for designing of ship board devices subjected to shock loading, which is caused by UNDEX (underwater explosion). To run DDAM analysis, one can use the NRL-396 design spectrum standard, or use customized spectrum defined by equations with user defined constants. The DDAM analysis results are saved in d3spcm binary plot database, which is accessible to LS-PrePost.

More details about the DDAM analysis is presented in the other paper at this conference [3].

6 Expanded d3spcm database

Response spectrum analysis provides combined response from a bunch of modes for structures subjected to excitation spectrum, like ground acceleration spectrum, in an earthquake event. The results in d3spcm show the possible peak value of response of structures during the excitation. The response variables include nodal displacement, velocity, acceleration and element stress / strain. In other words, d3spcm provides an envelope or upper limit for transient response of structures. This is useful in safety evaluation of important structures (like dams or nuclear power plants) which are under shock spectrum load or ground motion spectrum (e.g. earthquake excitations).

To study the contribution from each mode towards final response, the d3spcm database was expanded to include response from each participating mode. When N modes are used in response spectrum analysis, there are totally N+1 states in d3spcm. The first state shows the final or overall response, given by combining mode responses using SRSS, CQC, NRL-Sum or other mode combination methods. The following N states show individual mode response from mode 1 to mode N.

To activate the output of expanded d3spcm database, one needs to set binary = 2 in keyword ***DATABASE_FREQUENCY_BINARY_D3SPCM**.

For illustration purpose, a multi-story building (shown in Figure 17) which is subjected to x-directional ground acceleration spectrum is used as an example. The response spectrum analysis results (for z-acceleration) are shown in Figure 18. One can notice that the title shows “Freq = 0” which reminds user that this is final or overall response.

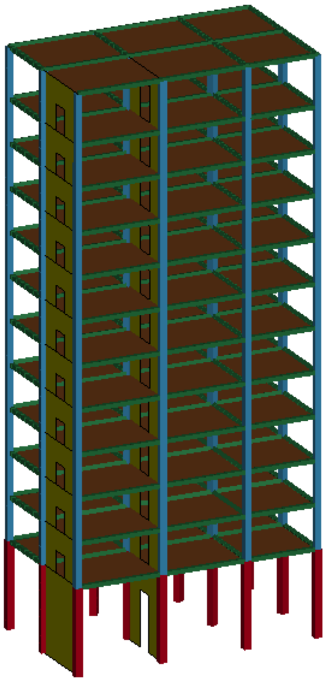


Fig.17: Multi-story building for response spectrum analysis

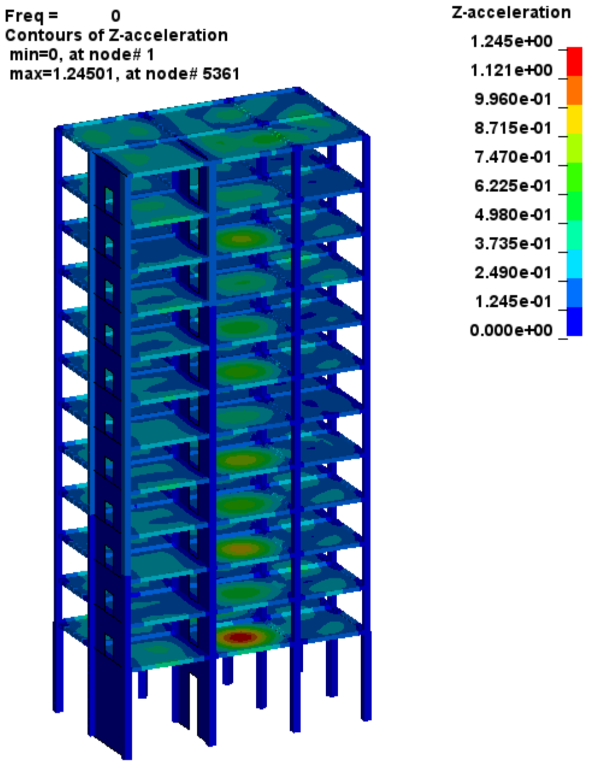


Fig.18: Z-acceleration response

The modal responses from the first 4 modes are given in Figures 19 to 22.

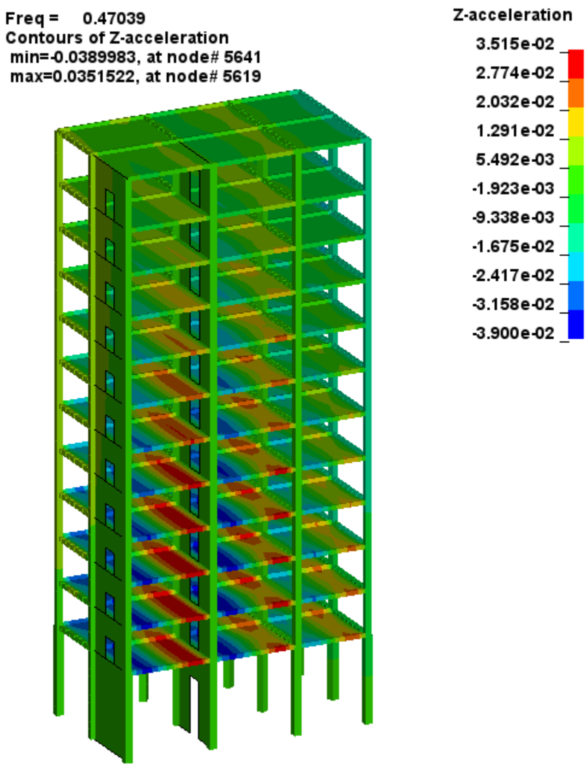


Fig.19: Z-acceleration from mode 1 (0.47039 Hz)

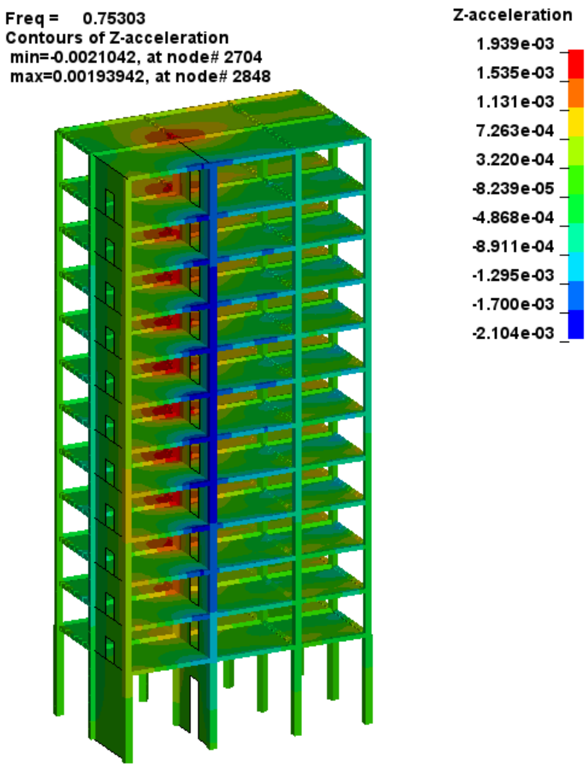


Fig.20: Z-acceleration from mode 2 (0.75303 Hz)

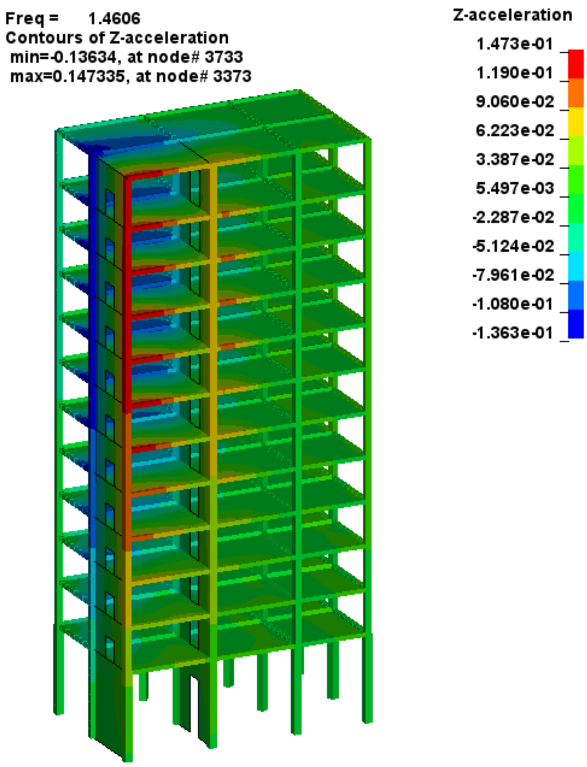


Fig.21: Z-acceleration from mode 3 (1.4606 Hz)

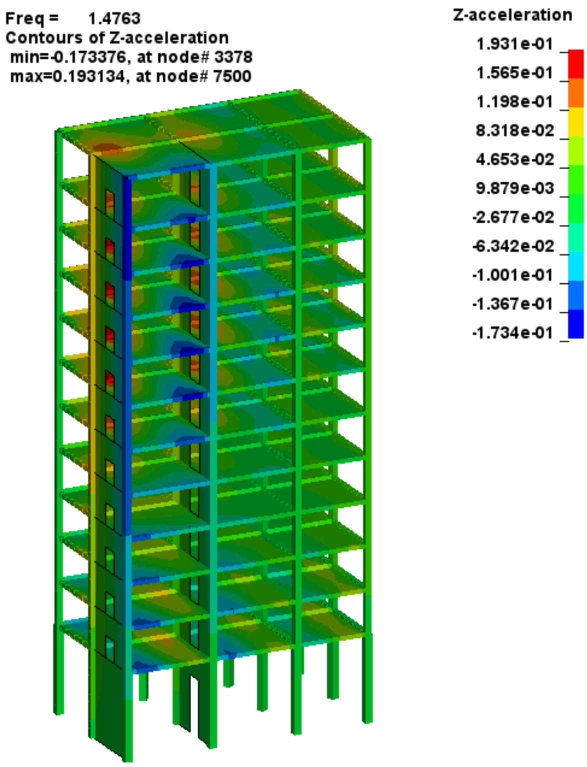


Fig.22: Z-acceleration from mode 4 (1.4763 Hz)

In addition, to study the contribution to overall response at selected nodes or elements, one can use the “History” tool in “Post” in LS-PrePost. For example, to see which mode contributes most to the z-acceleration at node 5361, the following figure can be obtained

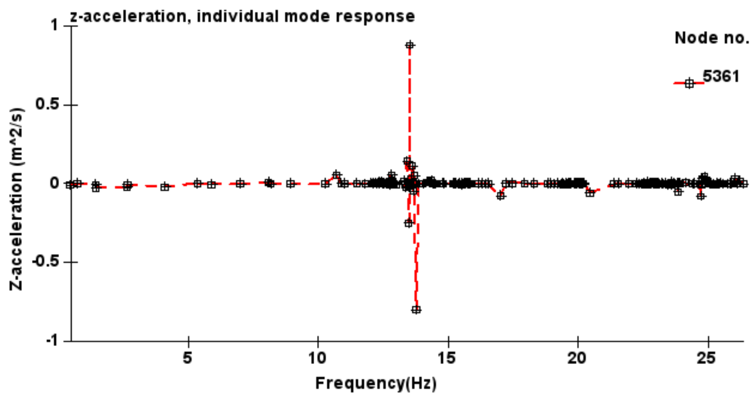


Fig.23: Z-acceleration response at node 5361 from individual modes

From Figure 23, it is clear that two modes with natural frequency between 13.0 Hz and 15.0 Hz contribute most to the total response at node 5361. Clicking on the peak values of the curve, or zooming in on the curve, we know that those modes are mode 48 (frequency = 13.565 Hz) and mode 55 (frequency = 13.804 Hz).

7 New mean stress correction methods for fatigue analysis

Fatigue analysis based on frequency domain approach and time domain approach has been implemented to LS-DYNA [4, 5].

Mean stress has important effect on fatigue behaviour of metal structures. Mean stress correction is necessary for accurate prediction of fatigue life of those metal structures. Under different mean stress, the SN curve of the same material can change quite a lot.

In LS-DYNA, two categories of mean stress correction methods are available.

- Use ***DEFINE_TABLE** to define a family of SN curves. Each curve corresponds to a unique mean stress. In ***MAT_ADD_FATIGUE** keyword, use the table ID for the SN curve. When a mean stress is not represented by the existing SN curves, interpolation is performed to find the corresponding number of cycles for failure *N*, for the given stress range or stress amplitude *S*, under current mean stress;

- Use equations to perform mean stress correction, based on the SN curves obtained by fully reversed testing (R = -1, or mean stress = 0). Following mean stress correction equations are available
  - Goodman equation
  - Soderberg equation
  - Gerber equation
  - Goodman tension only equation
  - Gerber tension only equation
  - Morrow equation (for fatigue analysis based on EN curve)
  - Smith-Watson-Topper equation (for fatigue analysis based on EN curve)

8 Load steps definition in fatigue analysis

A new keyword ***FATIGUE_LOADSTEP** was implemented to define load steps in fatigue analysis.

With this keyword,

- User can choose which segments of loading history are needed in fatigue analysis. Sometimes user may want to skip the starting transient response in fatigue analysis, and use only the steady state cyclic response. With a nonzero **tstart**, user can decide when to start to extract stress / strain time history.

- User can compute fatigue cumulative damage ratio for a long term load, based on representation on a shorter load step. The cumulative damage ratio, computed on the shorter load step, is multiplied by a scale factor (which is the ratio between the duration of real load and the duration of the representative load step), to provide estimation of the cumulative damage ratio for the real load, which could be much longer and be prohibitive to compute otherwise. Of course, it is assumed that stress / strain response in the shorter load step is a good representation of the behaviour in the real load step. And the material properties don't change with the number of load cycles, or with the load sequence. In other words, the fatigue behaviour of the structure is linear.
- User can combine fatigue results from multiple load steps.

The example under consideration is a metal pipe model, subject to temperature oscillation (see Figure 25). For real life application, this pipe can be cannon or gun pipe and the temperature oscillation can be due to continuous gun fires. The thermal stress cycle due to the temperature oscillation induces fatigue damage to the metal. After sufficient number of gun fires, the cumulative damage ratio can reach 1 and indicates the end of the service life of the gun pipe. The pipe is fully constrained at the bottom.

The piple is modelled by ***MAT_ELASTIC_PLASTIC_THERMAL**. The thermal loading is defined by ***LOAD_THERMAL_LOAD_CURVE**. The keyword cards for ***FATIGUE_LOADSTEP** and other keywords for the load can be found in Figure 24.

```
*FATIGUE_ELOUT
$#    ssid    sstype
$
$#      dt
$
$#   stres     index    restrt    texpos
         0         0
*DATABASE_D3FTG
$#  binary
         1
*FATIGUE_LOADSTEP
$#  tstart      tend    texpos
       0.0       50.    10000.
      50.0      100.    20000.
*DEFINE_CURVE
$#    lcid      sidr       sfa       sfo
       888         0       1.0       1.0
$#                a1                  o1
                 0.0                 0.0
                 5.0               200.0
                10.0                 0.0
                15.0               200.0
                20.0                 0.0
                25.0               200.0
                30.0                 0.0
                35.0               200.0
                40.0                 0.0
                45.0               200.0
                50.0                 0.0
                55.0               400.0
                60.0                 0.0
                65.0               400.0
                70.0                 0.0
                75.0               400.0
                80.0                 0.0
                85.0               400.0
                90.0                 0.0
                95.0               400.0
               100.0                 0.0
```

Fig.24: Keyword setting for running fatigue timestep

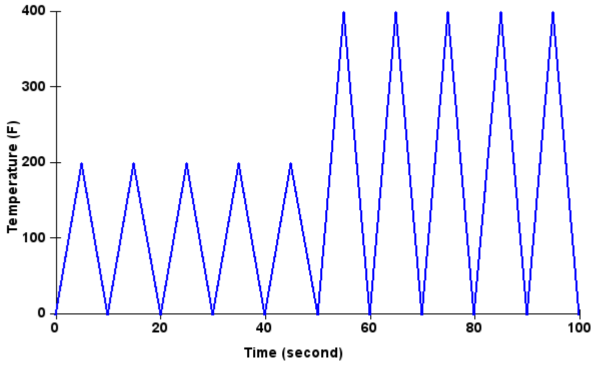


Fig.25: Thermal loading time history

The pipe is subjected to two steps of cyclic thermal loading. For the first load step, the temperature varies between 0°F and 200°F and this lasts for 10000 seconds. For the second load step, the temperature varies between 0°F and 400°F and this lasts for 20000 seconds. It is very time consuming to run finite element simulation for the whole thermal loading history of 30000 seconds. To get a quick estimation of the cumulative damage ratio, we can reduce the duration for each load step to only 50 seconds, and multiply the cumulative damage ratio generated in each step by a scale factor which is the ratio between the real loading period and the reduced loading period.

Figure 26 shows the distribution of effective stress near the end of simulation. Figure 27 shows the cumulative damage ratio of the pipe, after the 30000 seconds thermal loading. One can see that the maximum values of the effective stress and the cumulative damage ratio appear near the bottom of the pipe, probably due to the stress concentration at the constraints.

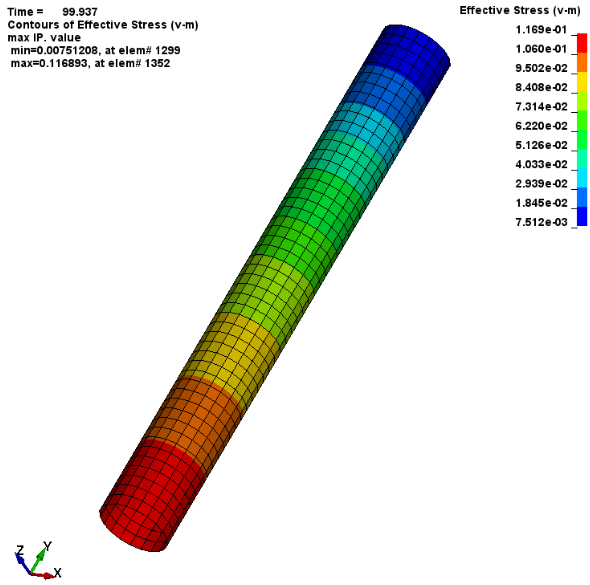


Fig.26: Effective stress at the end of simulation

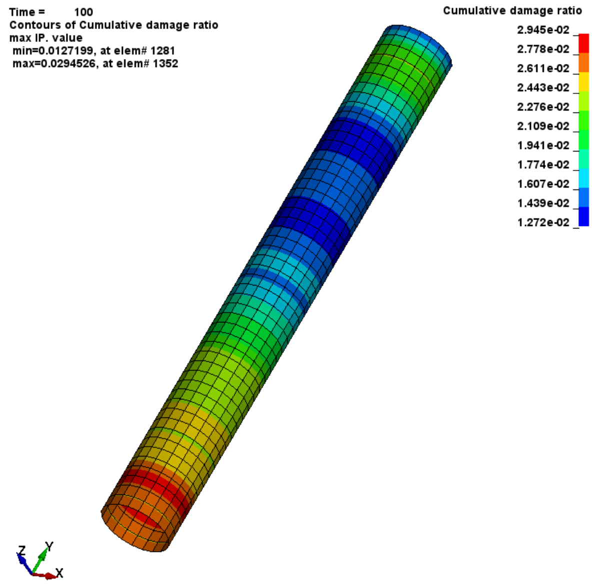


Fig.27: Cumulative damage ratio

Figure 28 shows the xy-shear stress fringe plot at the end of the simulation. The maximum value of the xy-shear stress takes place at element 1250. The xyplot of the same xy-shear stress at element 1250 is given in Figure 29. It is noticed that the stress cycle amplitude is approximately doubled starting from the time 50 seconds, due to the fact that the thermal loading is doubled at the time 50 seconds.

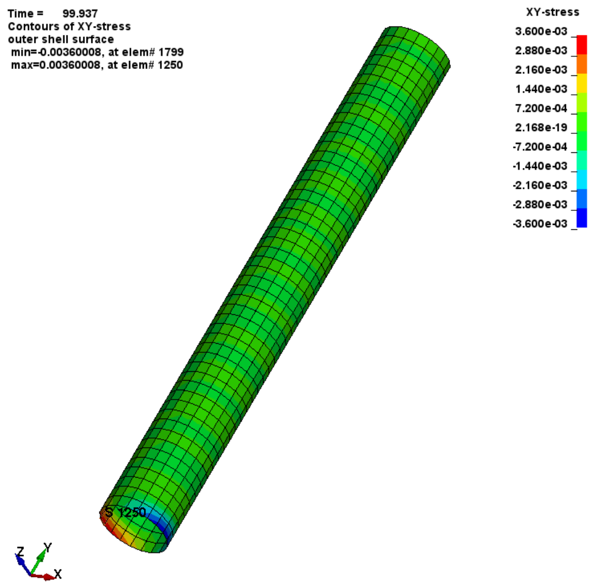


Fig.28: XY-shear stress at the end of simulation

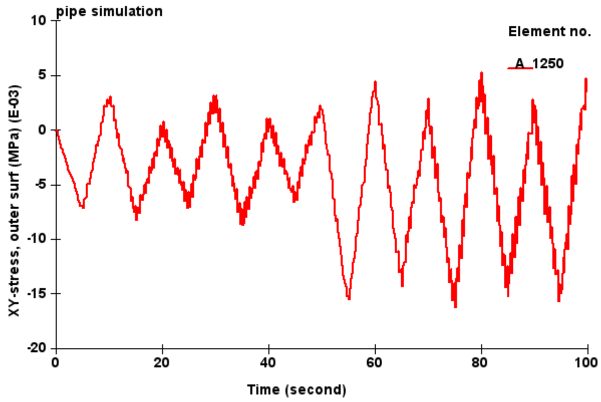


Fig.29: XY-shear stress at element 1250

9 Fatigue damage evolution and fatigue failure simulation

To show fatigue damage evolution, a progressive fatigue analysis is needed. The corresponding keywords are **`*DATABASE_D3FTG`** and **`*FATIGUE_FAILURE`**.

9.1 Fatigue damage evolution

With a nonzero dt in ***DATABASE_D3FTG,** LS-DYNA can perform fatigue analysis and dump out d3ftg database every dt time. Multiple states are saved in d3ftg and can be plotted using LS-PrePost 4.7 or newer versions. Each state saves cumulative damage ratios for the whole structure at one time point. With this database, user can track the fatigue damage ratio evolution for the structure. Particularly, with the "post"->"history"->"scalar" function in LS-PrePost, user can pick a critical element, and plot time history curve for the cumulative damage ratio for that element, and figure out approximately when the cumulative damage ratio of that element reaches a threshold value for failure (usually it is 1.0).

Figure 30 shows an L-beam fixed to a bottom plate by four bolts. The plate is constrained to ground. Prescribed harmonic motion (displacement) is applied on the edge of the hole on the L-beam, in the vertical direction. The prescribed displacement time history is shown in Figure 31.

The contact between the L-beam and the bottom plate is defined by keyword ***CONTACT_AUTOMATIC_SINGLE_SURFACE_MORTAR**.

The evolution of cumulative damage ratios are shown by Figure 32.

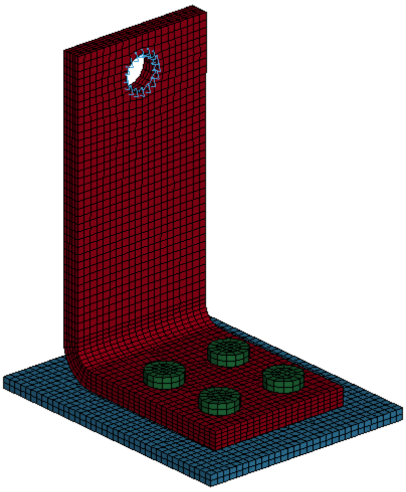


Fig.30: A L-BEAM constrained to a bottom plate

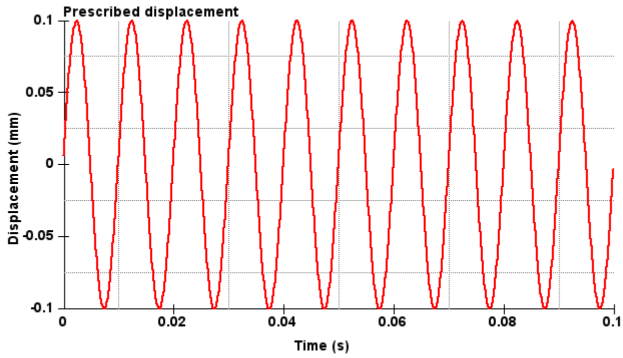
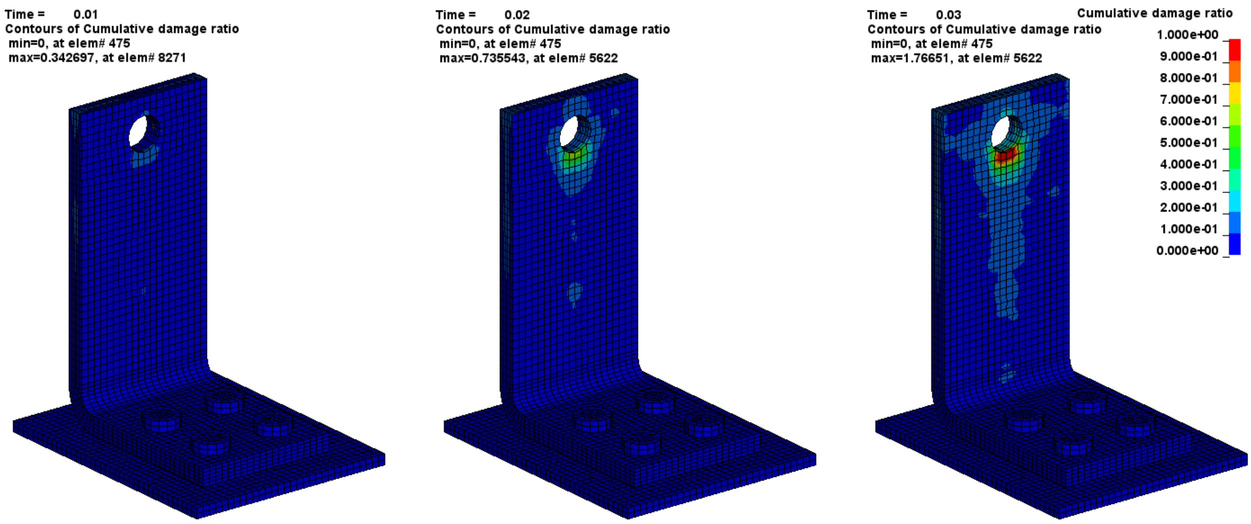


Fig.31: Prescribed harmonic displacement on the hole

The cumulative damage ratio fringe plots at time 0.01s, 0.02s and 0.03s are shown in Figure 32. Constant colour scale from 0 to 1.0 is used for all the plots so that one can easily compare the magnitude of the cumulative damage ratio, and trace the development of the damage. It is clear that the area at the lower edge of the hole experiences higher fatigue damage. The damage ratio increases with time and the damage area expands with time.



Time = 0.01s Time = 0.02s Time = 0.03s

Fig.32: Cumulative damage ratio at different time points

For a selected element (e.g. element 8271, which exhibits the maximum cumulative damage ratio at 0.01s as shown in Figure 32), the time history curve of cumulative damage ratio can be obtained using “history”->”scalar” function and is shown in Figure 33.

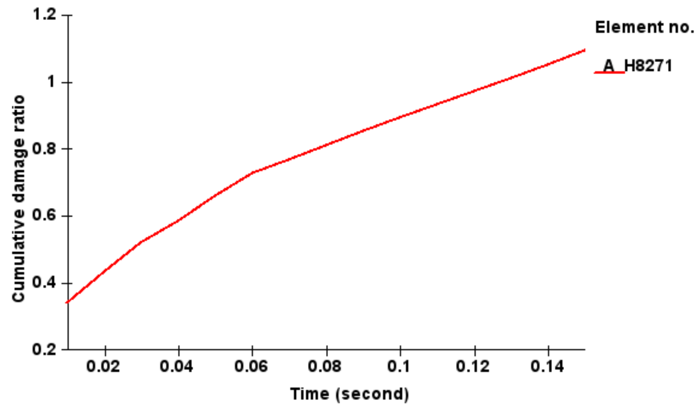


Fig.33: Time history curve of cumulative damage ratio for element 8271

From Figure 33, it is easy to see that the cumulative damage ratio of element 8271 reaches 1.0 at around time 0.13 second. In other words, the element 8271 fails at around time 0.13 second, under the current loading condition.

9.2 Fatigue failure simulation

A new keyword `*FATIGUE_FAILURE` was implemented to introduce a mechanism to model the failure of elements due to fatigue. With this keyword, user can define a threshold cumulative damage ratio (the default value is 1.0) and all the elements with cumulative damage ratio larger or equal to this value can be removed from the structure for subsequent simulation. For increased safety factor, the threshold cumulative damage ratio can be defined as a number smaller than 1.0.

This is a simple way to show the local failure of structures due to fatigue, and it provides an opportunity to study the effect of local fatigue failure on the overall behaviour of structures in a long term. An approximate fatigue crack propagation trajectory can be obtained by this approach.

A more accurate simulation of the fatigue crack propagation can be achieved by using the approach by fracture mechanics, or using the cohesive zone modelling.

The max cumulative damage ratio at time 0.03 second is 1.76651 (see Figure 32). It is obvious that several elements have failed (including element 5622, which exhibits the max cumulative damage ratio 1.76651). With **`*FATIGUE_FAILURE`** and ifailure= 1 and dratio=1.0, LS-DYNA automatically removes those elements whose cumulative damage ratio ≥ 1.0 from the structure. The remaining elements and their cumulative damage ratio fringe plot are shown in Figure 34. Then the cumulative damage ratio of the remaining elements continue to grow with the loading. Figure 35 shows the cumulative damage ratio at 0.04 second. One can see that the cumulative damage ratio of several other elements goes beyond 1.0 at 0.04 second (e.g element 5587), and this results in failure of those elements too. Those failed elements are removed too, as shown in Figure 36. It is expected that with the loading cycles going on, more and more elements will have cumulative damage ratio ≥ 1.0 and will fail and be removed from the structure...

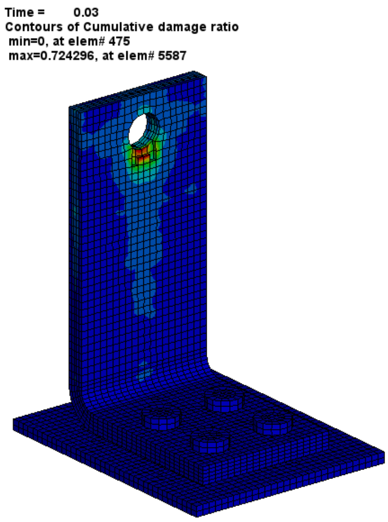


Fig.34: Cumulative damage ratio at 0.03s (failed elements are removed)

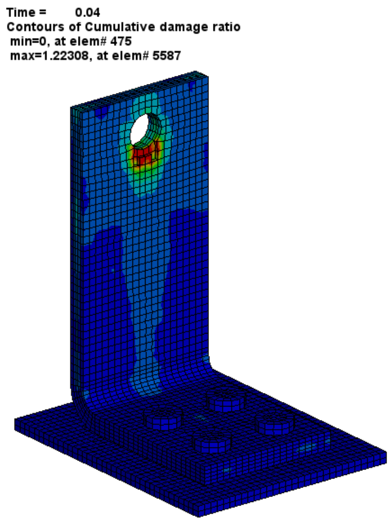


Fig.35: Cumulative damage ratio at 0.04s

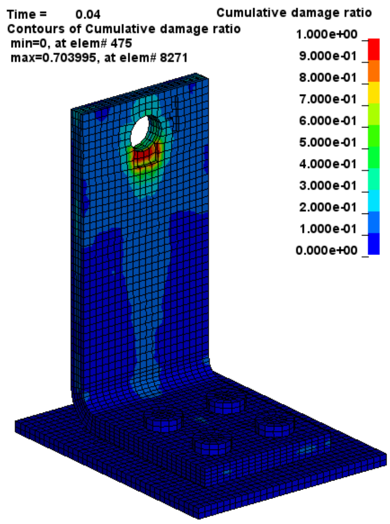


Fig.36: Cumulative damage ratio at 0.04s (failed element are removed)

Figure 37 provides a closer view of the area where several elements were removed from the structure at 0.03 second as their cumulative damage ratios became higher than the threshold cumulative damage ratio (1.0 in this case). The removal of these elements results in redistribution of the stress and actually accelerates the failure of the whole structure.

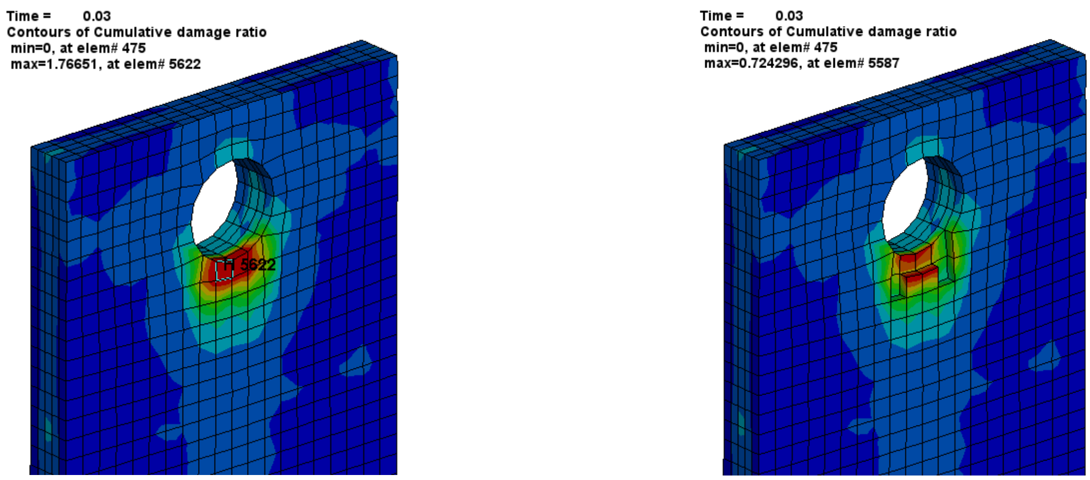


Fig.37: Local elements before and after the failed elements are removed.

The keyword setting for modelling the damage evolution and fatigue failure is shown in Figure 38.

```
*FATIGUE_ELOUT
$#     pid     ptype
$
$#      dt
$
$#   strsn     index    restrt    texpos
         0         0
$
*DATABASE_D3FTG
$#  binary        dt
         1    1.E-02
*FATIGUE_FAILURE
$#ifailure    dratio
         1        1.
*DATABASE_ELOUT
    1.E-05         2
```

Fig.38: Keyword setting for modelling fatigue damage evolution and fatigue failure.

10 Summary

This paper reviews recent updates in frequency domain analysis and fatigue analysis in LS-DYNA, and introduces several new keywords and options for running these features. These new options and enhancements enable users to solve more comprehensive problems in NVH and durability analysis.

For continued development of the frequency domain analysis and fatigue analysis features, we are looking at following topics:

- Further improving the performance and reducing computation time of these features
- Viscoelastic material modelling in direct steady state dynamics
- Coupling vibration, acoustic and fatigue features with other features in LS-DYNA, such as Electromagnetism, CFD, etc., to provide user with a complete software package for modelling multi stage, multi physics problems.
- Developing capabilities for fatigue fracture simulation and composite delamination simulation.

We are also looking forward to hearing comments and suggestions from users, on future development of the frequency domain and fatigue features.

11 Literature

[1] Huang Y, Cui Z: “Recent updates in LS-DYNA frequency domain solvers”, Proceedings of the 10th European LS-DYNA Conference 2015, Wurzburg, Germany.

[2] Kondapalli P, Jankowiak T, Huang Y: “Application of LS-DYNA to NVH Solutions in the Automotive Industry”, Proceedings of the 14th International LS-DYNA Users Conference 2016, June 12-14, 2016, Dearborn, MI, USA.

[3] Huang Y, Cui Z: “DDAM Analysis with LS-DYNA”, Proceedings of the 12th LS-DYNA European Users’ Conference, May 2019, Koblenz, Germany.

[4] Ringeval A, Huang Y: “Random Vibration Fatigue Analysis with LS-DYNA”, Proceedings of the 12th International LS-DYNA Users Conference, June 3-5, 2012, Dearborn, MI, USA.

[5] Huang Y, Cui Z, Ho P, Zhang C: “Application of LS-DYNA in Structural Fatigue Analysis”, Proceedings of the 14th Internatinal LS-DYNA Users Meeting & Conference 2016, June 12-14, 2016, Dearborn, MI, USA.