D E S I G N - A N A L Y S I S T E C H N O L O G I E S

FatDyn-X

Integrated Fatigue & Dynamic Analysis

Where durability meets dynamics

Design-Analysis Technologies LLC · Troy, Michigan

www.dAnalysisTech.com · Vel@dAnalysisTech.com

FatDyn-X — Fatigue and Dynamics Technologies in One Executable

Solvers from Normal Modes

- Inertia-relief / static
- Modal-transient
- Modal-resonance
  - SOL 101 and SOL 112 eliminated
  - Exact from t = 0, no start-up oscillation
  - Events and modes classified as dynamic or quasi-static by Time-PSD screening

Fatigue Engine

- Stress-life and strain-life in one run; Goodman, Gerber, Walker, FKM, Morrow, & SWT mean corrections
- Abs. max. principal, signed von Mises, Tresca and critical plane for shells and solids
- Fatigue at any XYZ point and plane
- Damage sensitivity to loads and modes

Spot and Seam Welds

- From grid-point forces: Rupp, Nakahara, Dong (spot); Dong master curve, Volvo (seam)
- Crack-direction markers, critical load channel per weld, duty-cycle damage maps
- Benchmarked on the SAE FD&E T-tube tests and published spot-weld cases

Brittle and Thermo-Mechanical

- Ceramics, glass, quartz: cyclic S-N plus slow crack growth, temperature-dependent
- Direct-transient fatigue from OptiStruct / Nastran / LS-DYNA thermal runs
- Mixed SN / EN / BR materials in one run; verified against hand solutions

Fracture Mechanics

- Paris, Walker, Forman, NASGRO and tabular da/dN from FE stress histories
- Through, edge, hole, embedded and Newman-Raju surface cracks
- **Additive-manufacturing defects**: Murakami √area and Gumbel extreme value

Frequency Domain

- Modal frequency response H(f) built internally from the normal modes
- PSD fatigue by Dirlik for load input or base-excitation input
- Accelerated-test PSD synthesis: days of field events into one shaker PSD

Dynamics Insight

- Node, element-force and stress time histories; acceleration with rigid-body motion
- Relative motion and contact detection over millions of time steps
- Mass forces on concentrated masses; bolted-joint slip margin of safety

FatDyn-X Studio GUI

- Eleven guided steps from model to results with a live input-deck preview
- Pre-flight checks, material and load plots, open-source material database
- Runs the solver, weld, crack-growth and stand-alone tools; browses results

1-Normal Mode Run
Replaces inertia-relief (SOL 101) and modal-transient (SOL 112) for fatigue and dynamics

≈ 10× faster
Parallel processes on 10 cores, no per-core fees

1-EXE
Stand-alone Windows executable, keyword input deck

Open material DB
S-N, ε-N, brittle, crack-growth and weld cards with provenance

FatDyn-X

Fatigue and dynamic analysis under time-history and frequency-domain loads, from one deck

FE input: unit load cases or normal modes, with grid-point forces, element forces and stresses. Load input: load-channel time histories (csv / rsp) or load / acceleration PSDs.

Inertia-Relief / Static

Normal Modes

Solvers inside FatDyn-X, built on the normal modes

- Inertia-relief / static via normal modes
- Modal-transient via normal modes
- Modal-resonance via normal modes
- Time-PSD tools: event and mode classification
- Modal frequency response H(f), magnitude and phase
- PSD response from the internal FRF (load or base excitation)

Time-history loads

Fatigue

- Stress-life and strain-life in one run; mean stress: Goodman, Gerber, Walker, FKM (S-N); Morrow, SWT (ε-N)
- Stress methods: abs. max. principal, signed von Mises, Tresca, critical plane (shells and solids)
- Fatigue on a user-defined plane at any XYZ; damage sensitivity to loads and modes
- Spot and seam welds from grid-point forces: Rupp, Nakahara, Dong (spot); Dong master curve, Volvo (seam)
- Brittle materials: cyclic S-N + slow crack growth, temperature-dependent
- Fracture mechanics from FE stress histories: Paris, Walker, Forman, NASGRO, tabular da/dN
- Thermo-mechanical direct-transient fatigue from Nastran / OptiStruct / LS-DYNA results

Dynamics

- Node time histories (displacement, velocity, acceleration), acceleration with rigid-body motion
- Relative motion: contact and interference checks, strain between nodes, twist and bending tracking
- Spring / rod / beam / bush and mass-element force histories; shell and solid stress histories
- Inertia relief: acceleration contours and mass forces
- modal transient: modal coordinates and loads, mode amplification

Frequency-domain loads

Frequency response (force excitation)

- Modal frequency response from the normal modes, damping per mode; H(f) magnitude and phase
- Node displacement, velocity, acceleration; element forces and stresses vs frequency, per node / element CSV
- The same transfer functions the PSD path builds internally

PSD response and fatigue (Dirlik)

- Two input paths: load PSD through the internal FRF (no separate FR run), or base-excitation PSD through a two-step punch workflow
- Spectral moments m0, m1, m2, m4; RMS, irregularity factor, peak rate; Dirlik damage with log-log PSD interpolation
- Abs. max. principal critical-plane metric for shells and solids; fatigue element set from the material mapping
- Per-element damage and life to CSV and HWascii; consolidated stress-FRF scan for large element sets

Shared by every route: one master deck, one open material database (S-N, ε-N, brittle, crack-growth and weld cards with provenance), rainflow and Miner, results as CSV and HWascii / UNV contours

PART 1 · Time-History Fatigue and Dynamics

Fatigue & Dynamics using Time-History Loads

FatDyn-X —Time History Topics

Core Capabilities

- Integrated fatigue + dynamic analysis in one workflow
- **Inertia-Relief / Static via normal modes** — eliminates 100s of SOL 101 cases
- **Modal Transient via normal modes** — eliminates SOL 112 and large transient decks
- Dynamic outputs: displacement, relative motion, element forces & stresses, acceleration (incl. rigid-body)
- Mass-force time histories for EV, aerospace, and defense systems

Advanced Fatigue Methods

- Stress-life and strain-life in a single run
- Mean-Stress correction: Goodman/Gerber with plateau correction, Walker, & FKM for Stress-life; Morrow and SWT for strain-life
- Multiple stress methods together: Abs Max Principal, Signed Mises, Tresca, Critical-Plane
- Critical-Plane fatigue for shells and solids
- Location- and plane-specific fatigue at any user XYZ point — vital for 3D-printed and solid parts
- Damage sensitivity ranking by load channel and by mode
- **Brittle materials (ceramic, glass, quartz): cyclic S-N + slow-crack-growth, temperature-dependent**
- **Practical fracture mechanics: Paris, Walker, Forman, NASGRO, tabular da/dN from FE stress histories**
- **Fatigue Analysis of Spot and Seam Welds**
- Thermo-mechanical direct-transient fatigue from OptiStruct / Nastran and LS-DYNA

Automation & Insight

- Automated load-case creation (min–max and time-set based)
- Automated event classification — dynamic vs. quasi-static via Time–PSD screening
- Fatigue damage sensitivity to load channels and modes
- Mode-wise dynamic amplification metrics to guide damping and design changes

Examples Covered in This Deck

Basic Fatigue

- Plate fatigue (shells and solids) — FatDyn-X vs benchmark
- Truck-frame fatigue — FatDyn-X vs benchmark

Advanced Fatigue

- Truck-frame fatigue by four solvers in one run
- Four stress methods in one fatigue run
- Simultaneous stress & strain life in one fatigue run

Critical-Plane and Location-Specific Fatigue

- Critical-plane fatigue (shells and **solids**)
- Critical-plane fatigue (shell vs solid)
- **Fatigue at desired points XYZ & planes**
- Frame fatigue at desired points & planes

Damage Sensitivity

- Fatigue damage sensitivity to loads and modes

Dynamic Calculations

- Uniform constant load (FatDyn-X vs benchmark)
- Modal Transient: displacement, acceleration, force, stress
- Relative motion analysis of plate-structure with masses
- Inertia-Relief vs Inertia-Relief by Modes
- Time history of mass (CONM2) elements

Selecting Analysis Method for Events and Modes

- Choosing analysis methods for events
- Choosing analysis methods for modes

Misc Operations

- IR acceleration contours & mass-forces from a global Mass–Inertia–CG
- Static load cases by min–max loads
- Static load cases for a time set
- IR-static unit load cases
- Transient load cases by min–max loads
- Transient load cases by time set
- Full transient load case
- Normal-mode load case
- Save reduced loads by gate

What’s New — Inertia-Relief / Static now drives dynamics too

Traditional use of Inertia-Relief / Static

Generally limited to stress and fatigue analysis. The full dynamic information sitting inside the IR run goes unused.

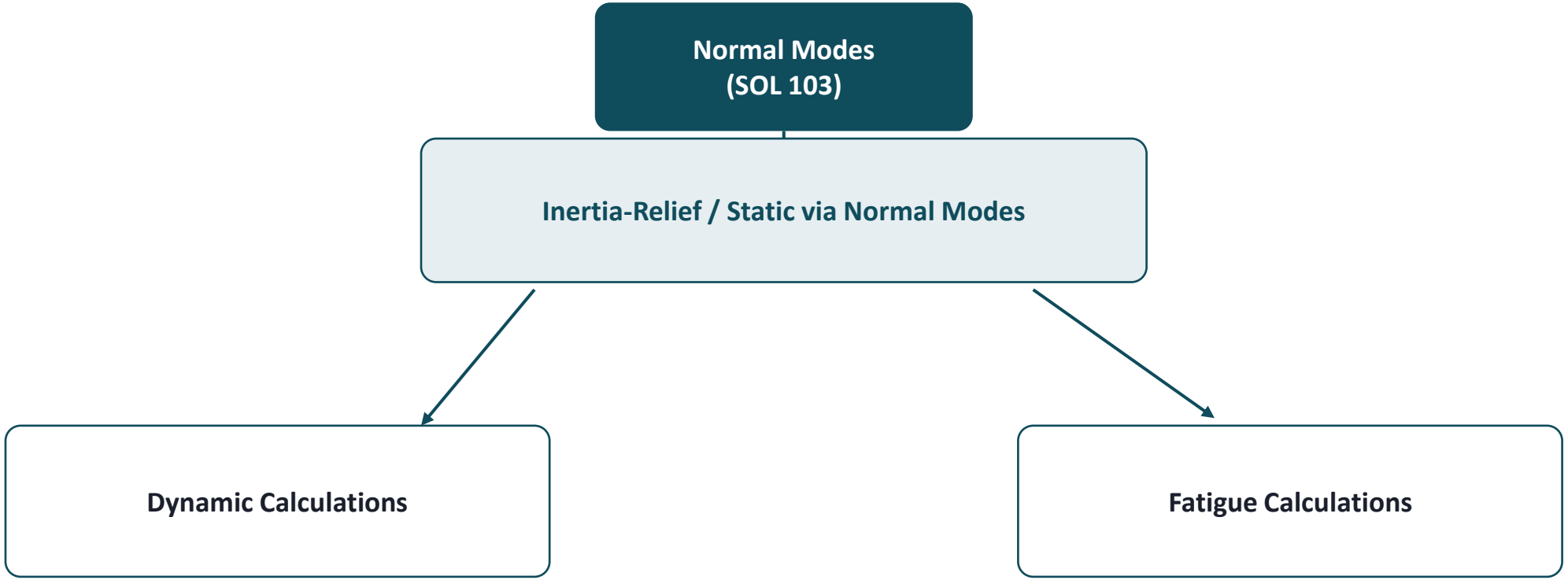
FatDyn-X enhancement

Same IR/Static result + load histories → full dynamic-calculation suite, in addition to stress and fatigue.

Dynamic calculations available from a single IR/Static run

- Node time histories — DIS, VEL, ACC
- Relative-motion / strain between node pairs (contact, interference, twist, bending)
- Spring / rod / beam force time histories
- Shell & solid stress time histories
- Acceleration including rigid-body motion
- Mass-element force histories [unique to FatDyn-X]
- IR acceleration contours & mass-forces [unique to FatDyn-X]

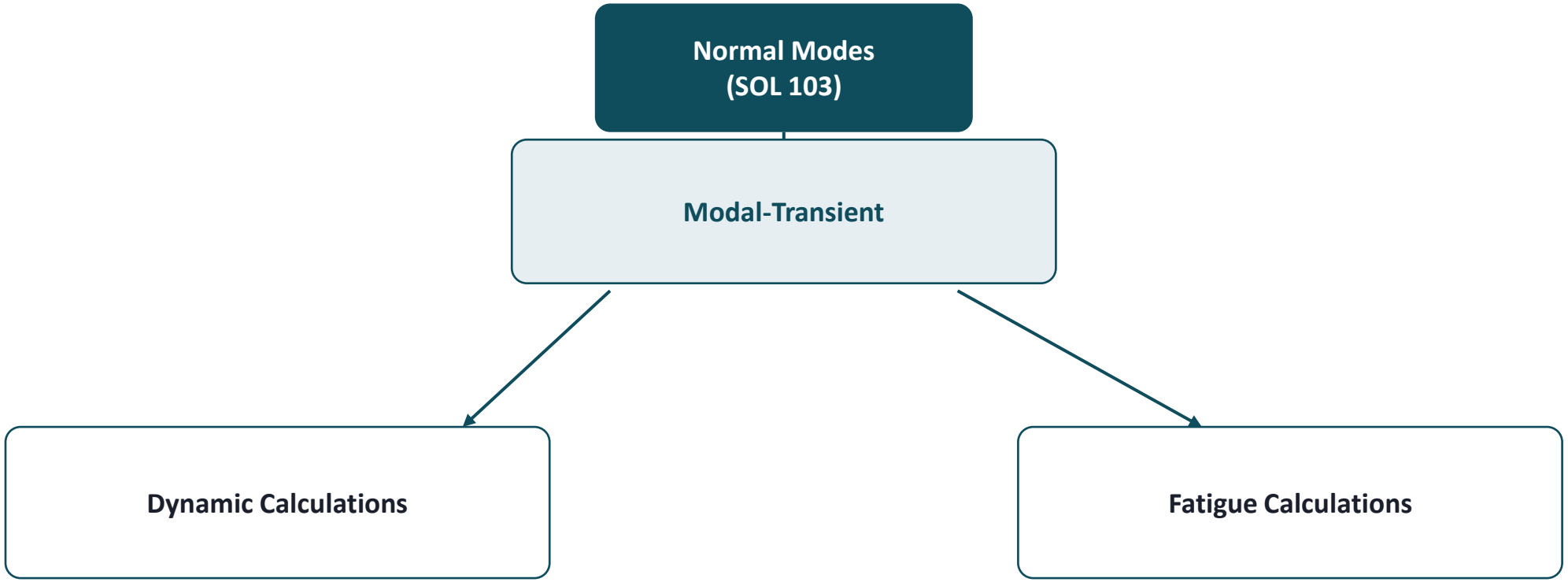
What’s New — Inertia-Relief / Static by Normal Modes



Inertia-Relief (SOL 101) eliminated — hundreds of IR-static load cases are no longer required.

Especially powerful when duty cycles mix quasi-static and dynamic events — one set of modal stresses serves both.

What’s New — Modal Transient by Normal Modes



Modal-Transient (SOL 112) eliminated — hundreds of FORCE / TLOAD / TABLED cards are no longer required.

Loads are projected to modal space and run as modal-transient — much faster, much smaller decks, the same physics.

Exclusive Fatigue Capability

Critical-Plane Method

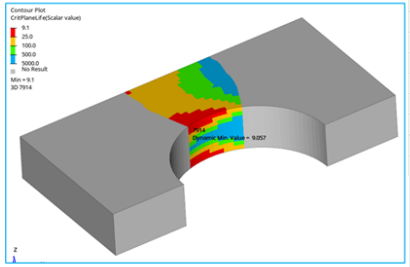
Unique algorithm for both shells and solids.

Location-Specific Analysis

Vital for 3D-printed parts and solid components where location and direction matters.

Industry-Leading

A combination of capabilities not commonly available in other software.



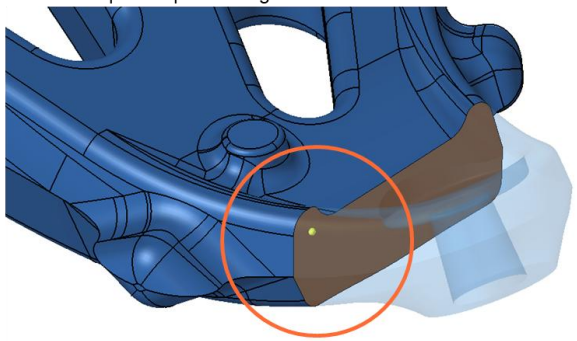
Element_ID	Plane-ID	planeNormalGlobal-i	j	k	FatLife
7914	1	-0.157	0.986	-0.061	9.1
7914	2	-0.086	0.989	-0.124	9.3
7914	3	-0.225	0.974	0	9.5
7914	4	-0.013	0.983	-0.185	10.4
7914	5	-0.291	0.955	0.06	10.6
7914	6	0.064	0.966	-0.252	12.9
7914	7	-0.393	0.906	0.157	15.3
7914	9	0.131	0.942	-0.309	17.2
7914	8	-0.452	0.865	0.216	21.9
7914	10	0.224	0.896	-0.383	31.0
949	1	0.165	0.986	0.009	66.5
949	2	0.252	0.968	0.008	67.1
949	3	0.066	0.998	0.01	74.1
949	5	0.35	0.937	0.006	77.0
949	4	-0.027	1	0.011	92.5
949	6	-0.176	0.984	0.012	175.3
949	7	-0.302	0.953	0.012	440.3

!Location,	Eid,	x,	y,	z,	k1,	k2,	k3,	Fat-Mid
Location,	22441,	0.0,	5.0,	3.0,	0.0,	3.0,	0.0	22
Location,	0,	-51.448,	-1.664,	4.5,	0.0,	0.0,	0.0	22
Location,	0,	-51.448,	-1.664,	4.5,	2.0,	3.0,	0.0	23
Location,	0,	-57.500,	-0.100,	4.5,	2.0,	3.0,	0.0	24
Location,	20266,	5.0,	1.0,	3.0,	0.0,	13.0,	0.0	25
Location,	25497,	5.0,	1.0,	3.0,	1.0,	3.0,	0.0	25

Locations can be **repeated; each repeat can have different plane and material**

3D Part Analysis

Location & plane-specific fatigue detection



SECTION 1

Solver Strategy

How FatDyn-X turns one set of normal modes into both fatigue and dynamic responses

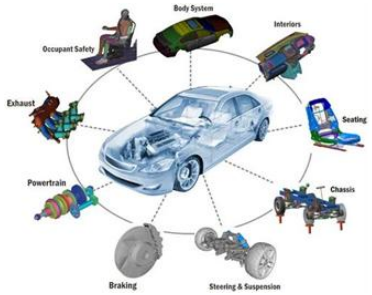
One Dynamic Equation Unifies Auto, Aero, and Space

For structural analysts, the motion and stress of any system reduces to:

$$M\ddot{q} + C\dot{q} + Kq = f(t)$$

FatDyn-X links this single equation to both fatigue and dynamics through modal and inertia-relief methods.

AUTOMOTIVE



AEROSPACE



SPACE



DEFENSE



Structures big and small — same governing equation, same FatDyn-X workflow.

Linear Static and Dynamic Solver Strategy

Static analysis — constrained

$$K\,q = f$$

Constraints make the stiffness matrix non-singular — direct solve.

Inertia-Relief analysis — unconstrained

$$K\,q = f_{applied} - M\,a_r^b$$

No constraints → matrix is singular. Rigid-body components are filtered to solve.

```
Nastran / OptiStruct:
PARAM, INREL, -1 (with support)
PARAM, INREL, -2 (without support)
```

Dynamic analysis — from millions of DOF to a handful via modal transformation

Direct transient (full FE)

- Solve $M\ddot{q} + C\dot{q} + K\,q = f(t)$ on all DOF
- Millions of DOF × long time histories
- Often infeasible for stress time histories

Modal transient (reduced)

- Transform $q = \Phi\eta$ using normal modes
- $\Phi^T M\Phi\,\ddot{\eta} + \Phi^T C\Phi\,\dot{\eta} + \Phi^T K\Phi\,\eta = \Phi^T f(t)$
- Keep a few modes → a few DOF

Solver Techniques — Rigid, Regular, and Residual-Static Modes

$$M\,\ddot{q} + C\,\dot{q} + K\,q = F(t)$$

Millions of DOF

$$\ddot{y}_i + 2\,\zeta_i\,\omega_i\,\dot{y}_i + \omega_i^2\,y_i = p_i(t)$$

A few DOF — modal transformation (y_i = modal coordinate for mode φ_i)

Rigid modes

$\phi_1 \dots \phi_6$

Used only to compute accelerations and mass-forces. With ω = 0, K = 0 and critical damping is not meaningful — FatDyn-X solves an acceleration-only problem with K = C = 0.

Regular elastic modes

$\phi_7 \dots \phi_n$

Mode-wise critical damping. Non-zero initial loads are handled exactly — no start-up oscillation that commercial solvers exhibit (automotive loads always start from 1 g).

Residual-static modes

$\phi_{n+1} \dots \phi_m$

Treated as non-resonant static shapes; resonance is explicitly ignored to avoid artificial peaks.

SOLVER STRATEGY

Stress → Fatigue — A Scalable Workflow

Full-field FE stress time histories over long durations are impractical. FatDyn-X uses an element-wise reconstruction strategy.

1 **Precompute base stress fields**

From inertia-relief static load cases or from modal stress shapes.

2 **Reconstruct local stress(t)**

Superpose bases with force or modal-coordinate time histories — compute each element's local stress history.

3 **Fatigue evaluation**

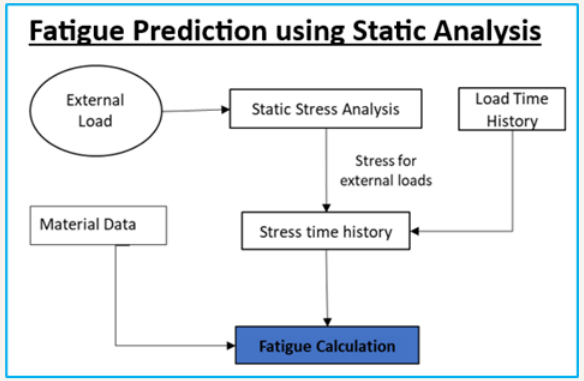
Cycle counting (e.g., rain-flow) and damage accumulation (e.g., Miner's rule) at element level.

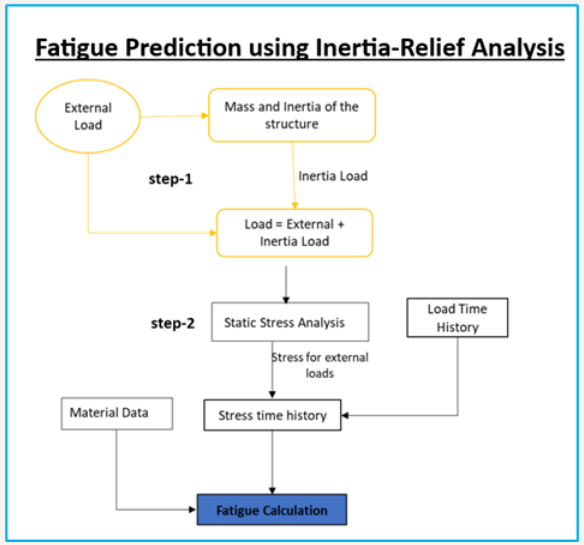
Benefits

- Scales to long time histories — millions of steps
- Preserves spatial accuracy at the element level
- Works with both static (inertia-relief) and modal inputs

Solver Flow Charts

Prefer static / IR cases when the load path is quasi-static





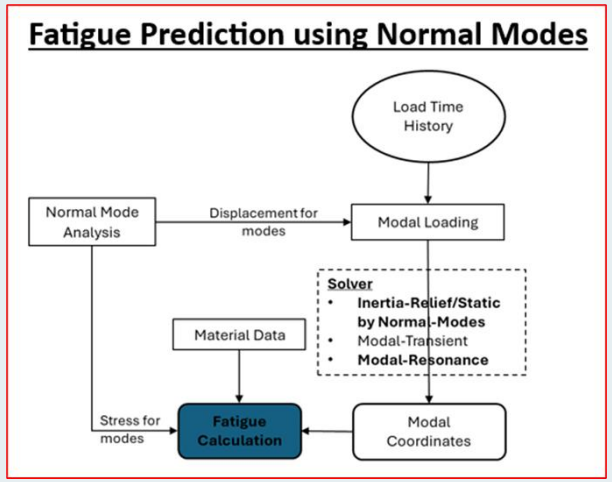
Fatigue using only normal modes

Works well when:

- Dominant dynamics captured by a modest set of modes
- Linear material behavior and small-deflection kinematics
- Loads can be projected into modal space ($\Phi^T f$)
- Modal stress shapes available per element

Why this is powerful

- Orders-of-magnitude reduction in DOF and runtime
- Enables element-level fatigue over long time histories
- No external dynamic solver required

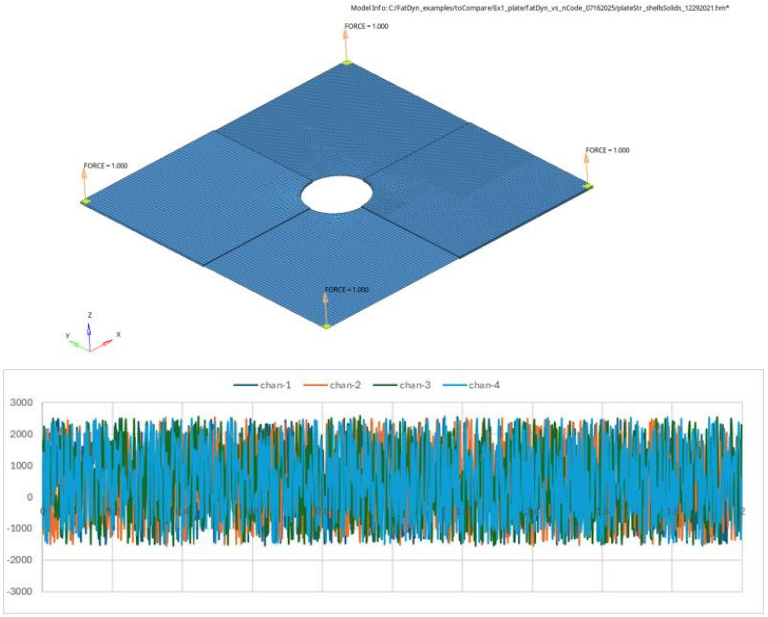


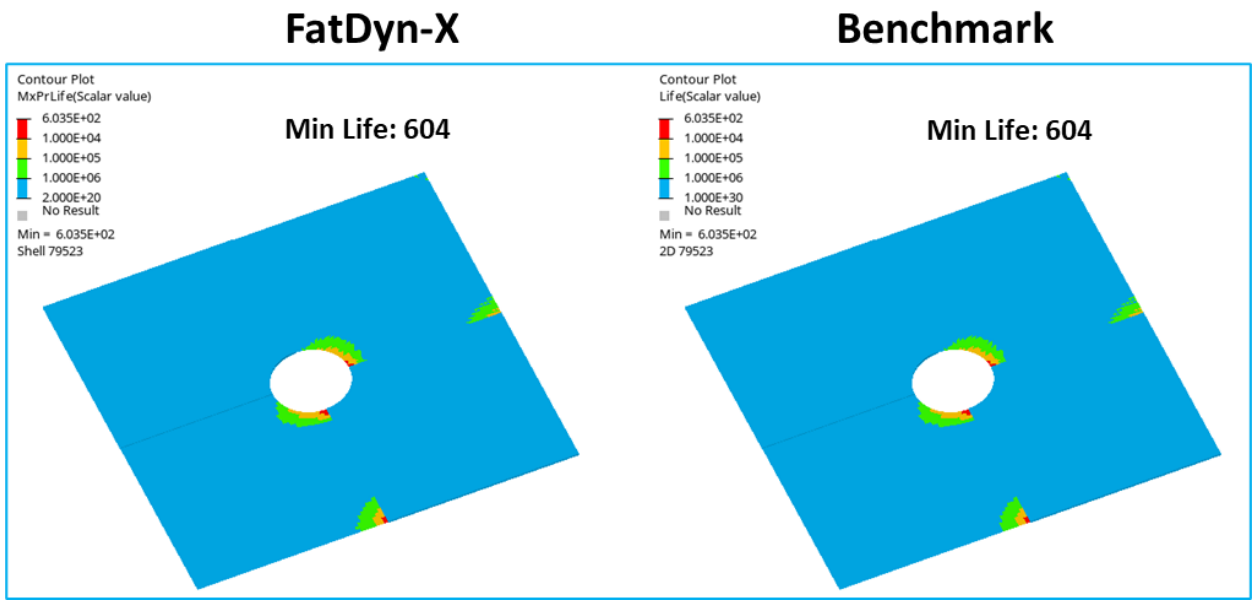
SECTION 2

Fatigue Life — Validation

FatDyn-X versus Benchmark

Ex-1 — Fatigue of Plate (Shells and Solids)



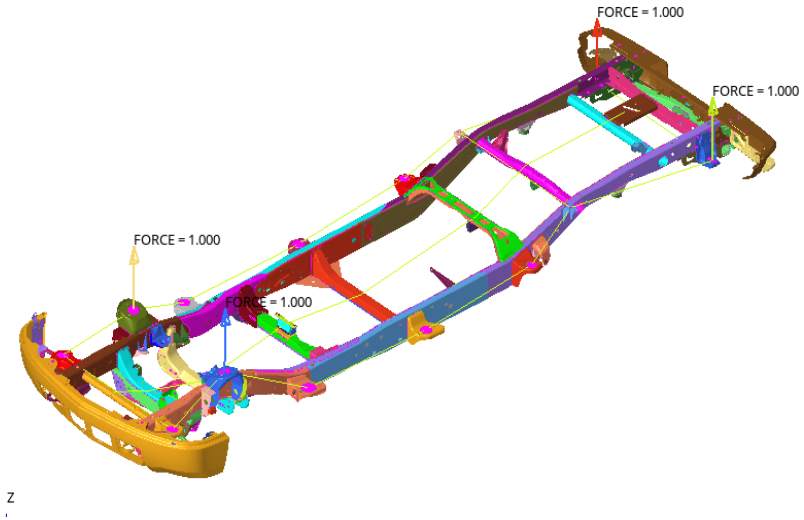


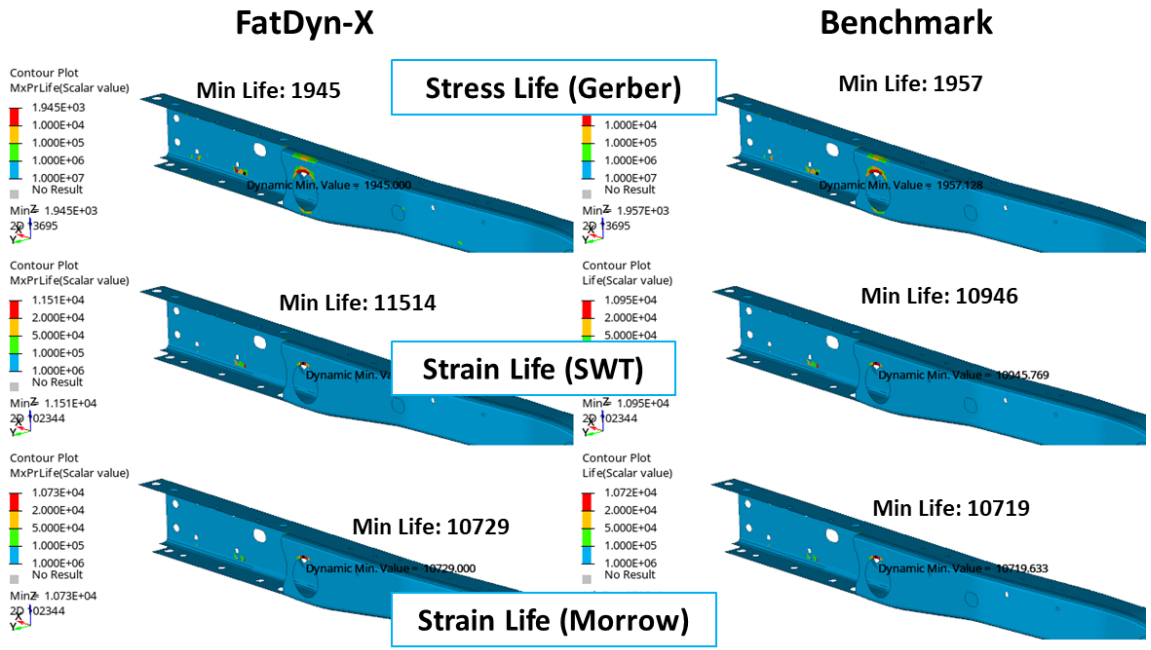
Fatigue Life (AbsMaxPr, SN-Gerber)

Element	FatDyn-X	Benchmark
CQUAD4	1026	1026
CTRIA3	604	604
CHEXA	7.041×10^{9}	1.0×10^{30}
CTETRA (4 nodes)	2.162×10^{6}	2.329×10^{6}

Shell elements (Quad4, Tria3) match exactly. Solid-element results highlight numerical sensitivity in commercial solvers — the CHEXA benchmark life saturates at 1.0×10^{30} while FatDyn-X reports a finite, comparable value.

Ex-2 — Truck-Frame Fatigue (FatDyn-X vs Benchmark)





Stress-life and strain-life results

Method	FatDyn-X (Min Life)	Benchmark (Min Life)
Stress Life (Gerber)	1957	1945
Strain Life (SWT)	10946	11514
Strain Life (Morrow)	10719	10729

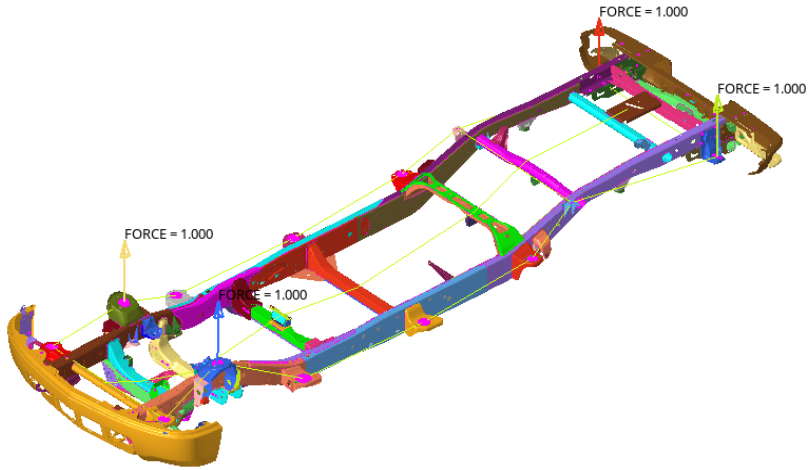
Strong agreement across both stress-life and strain-life methods. Minor differences confirm FatDyn-X accuracy under Gerber and SWT/Morrow corrections.

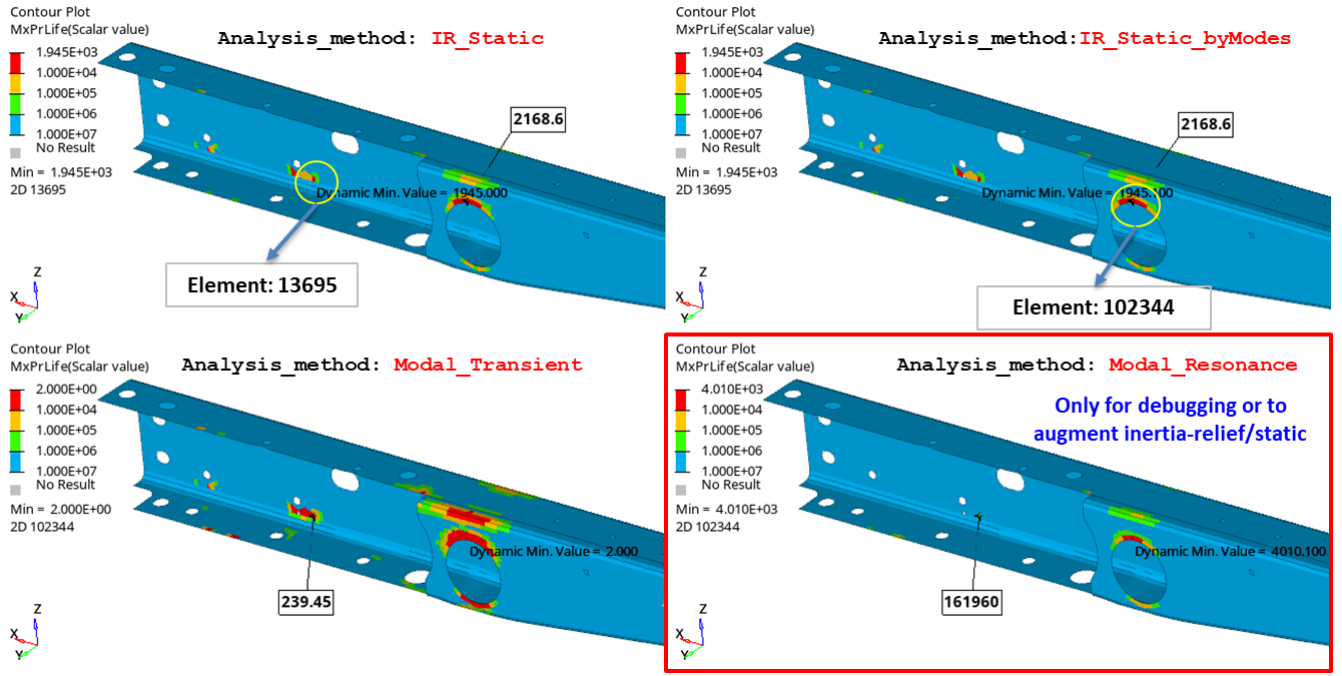
SECTION 3

Advanced Solver and Fatigue Methods

Multiple solvers, multiple stress methods, location-specific outputs

Ex-3 — Truck-Frame Fatigue by Four Solvers (One Run)



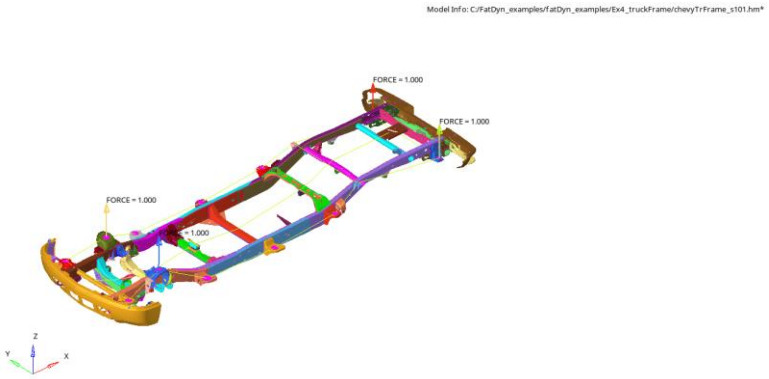


IR_Static	IR_Static_byModes	Modal_Transient	Modal_Resonance

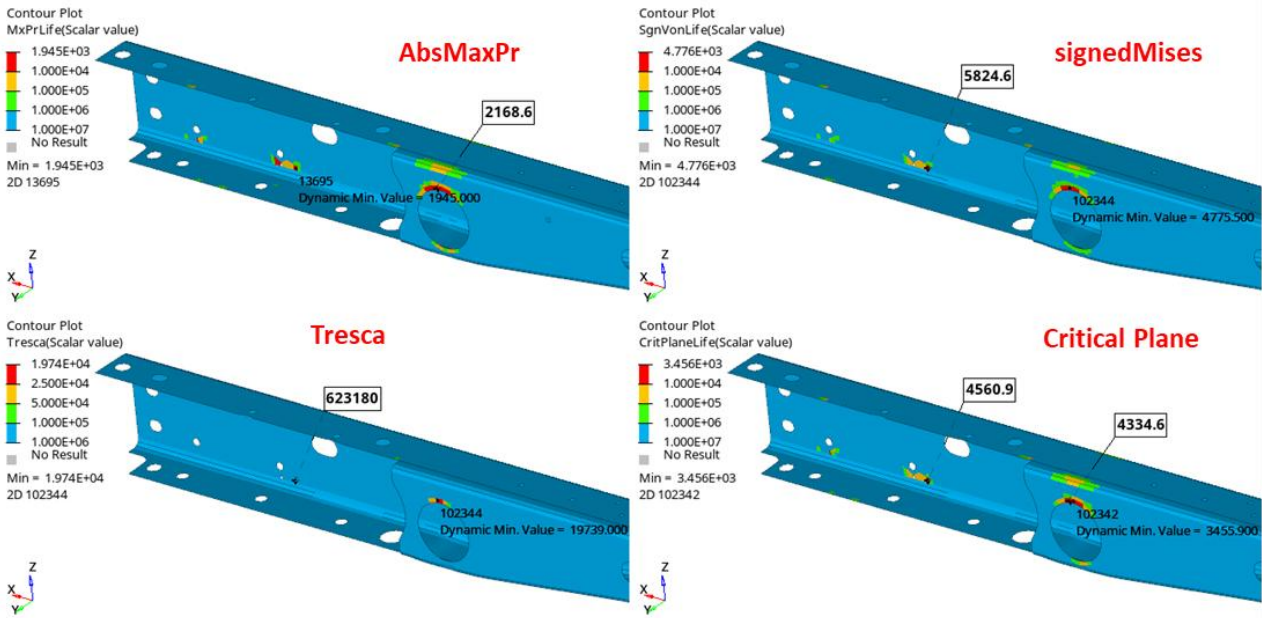
- IR-Static and IR-Static-by-Modes give identical fatigue life — confirming the modal formulation.
- Modal-Transient highlights a different critical element with significantly shorter life — fatigue is sensitive to dynamic loading.
- Modal-Resonance is a diagnostic of dynamic-amplification effects — not a life prediction by itself.

Modal-Dynamics separates dynamic response into static contribution + resonance amplification. Static dominates in most cases; resonance becomes critical when structural modes align with excitation frequencies.

Ex-4 — Four Stress Methods in One Fatigue Run



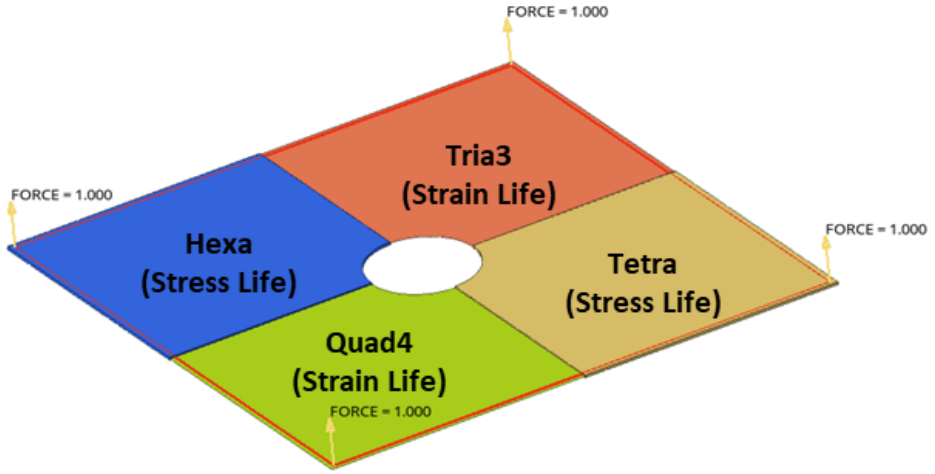

```
*TIMEHISTORY_LOADCASES
   Label                : th1_STRMETH
   ANALYSIS_METHOD      : IR_STATIC
   Fatigue_Analysis     : YES
   FATIGUE_STRESS_METHOD(S): ABSMAXPR SIGNEDMISES TRESCA CRITPLANE
```

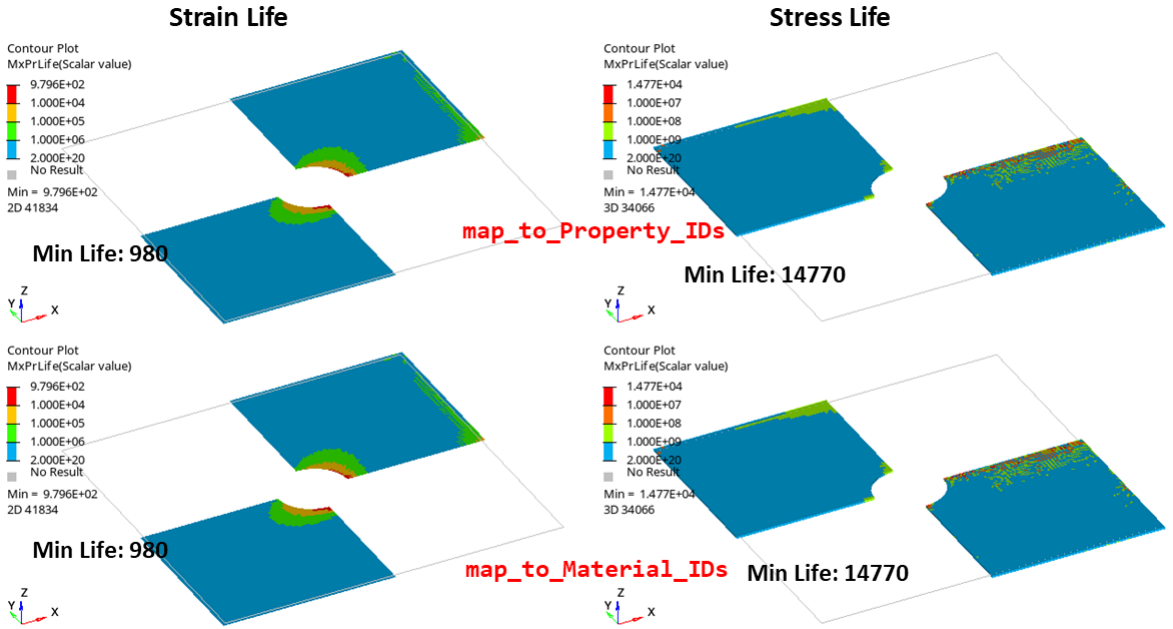


Fatigue life by four stress methods — single run

Stress method	Element 13695	Element 102344
AbsMaxPr	1,945	2,187
Signed Mises	5,825	4,776
Tresca	623,180	19,736
Critical Plane	4,561	4,335

Ex-5 — Simultaneous Stress & Strain Life in One Fatigue Run





Mapping by Property ID

```
*FATIGUE_MATERIAL_MAPPING
   SN_MEAN_CORRECTION       : GER
   EN_MEAN_CORRECTION       : Morrow
   Neuber_Plastic_Correction: Yes
   map_to_Property_IDs   YES
   PID         103       41   ! EN strain life aluminum
   PID_SET     solids    102   ! SN stress life aluminum
   PIDs_REST              41   ! EN strain life aluminum
```

```
*MID_sets
 Label: matsolids
      101
      102

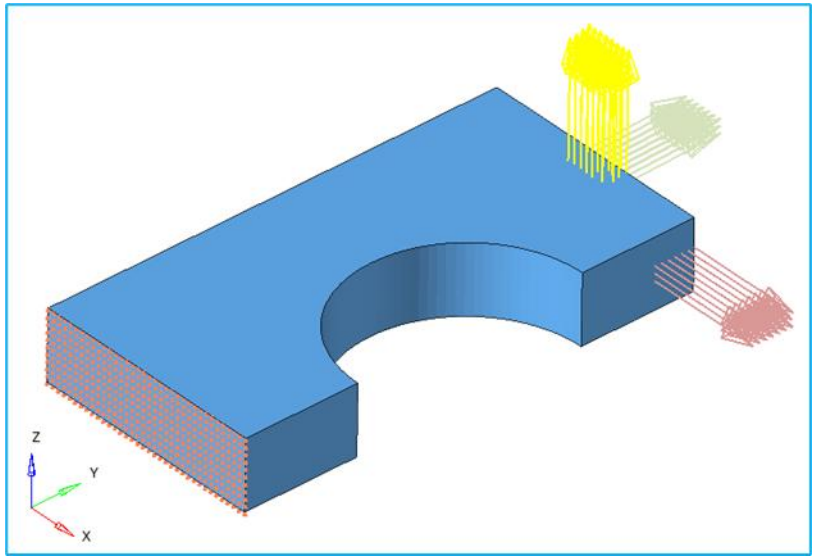
*PID_sets
 Label:solids
   104, 105
```

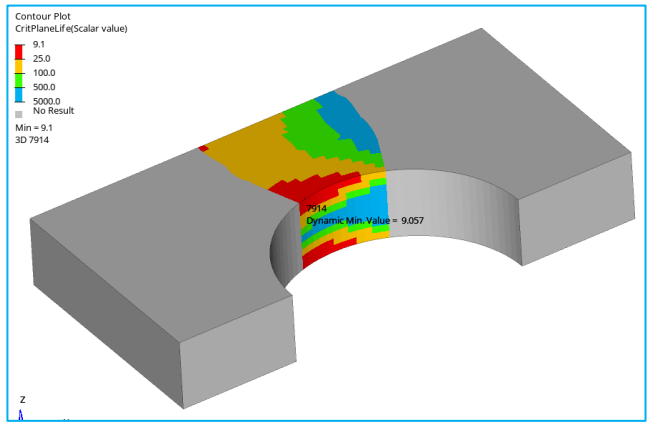
Mapping by Material ID

```
*FATIGUE_MATERIAL_MAPPING
   SN_MEAN_CORRECTION       : GER
   EN_MEAN_CORRECTION       : Morrow
   Neuber_Plastic_Correction: Yes
   map_to_Material_IDs   YES
   MID         103       41   ! EN strain life aluminum
   MID_SET   matSolids   102   ! SN stress life aluminum
   MIDs_REST              41   ! EN strain life aluminum
```

Both runs report Min Life: 980 (strain life) and 14,770 (stress life) — flexible mapping by PID or MID.

Ex-6 — Critical-Plane Fatigue (Solid)



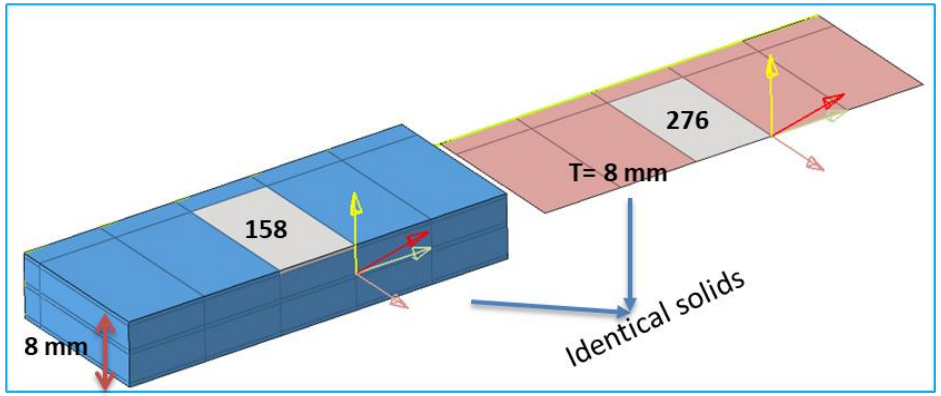


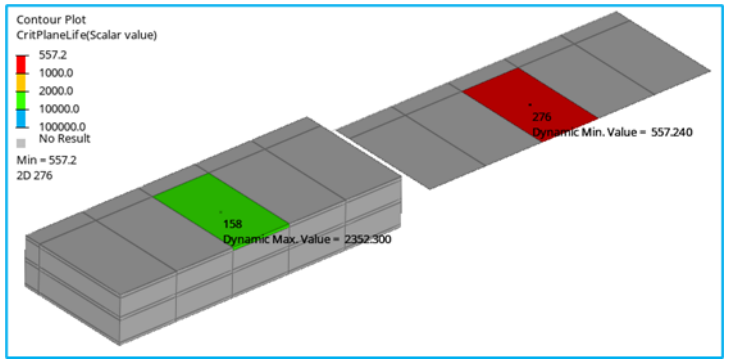
Element_ID	Plane-ID	planeNormalGlobal-i	j	k	FatLife
7914	1	-0.157	0.986	-0.061	9.1
7914	2	-0.086	0.989	-0.124	9.3
7914	3	-0.225	0.974	0	9.5
7914	4	-0.013	0.983	-0.185	10.4
7914	5	-0.291	0.955	0.06	10.6
7914	6	0.064	0.966	-0.252	12.9
7914	7	-0.393	0.906	0.157	15.3
7914	9	0.131	0.942	-0.309	17.2
7914	8	-0.452	0.865	0.216	21.9
7914	10	0.224	0.896	-0.383	31.0
949	1	0.165	0.986	0.009	66.5
949	2	0.252	0.968	0.008	67.1
949	3	0.066	0.998	0.01	74.1
949	5	0.35	0.937	0.006	77.0
949	4	-0.027	1	0.011	92.5
949	6	-0.176	0.984	0.012	175.3
949	7	-0.302	0.953	0.012	440.3

- **Evaluates multiple damaging planes per element**
- **Outputs life and normal vector (i, j, k) for each plane**

A solid component under three independent loading channels is analyzed with the Critical-Plane Method. The contour shows fatigue-life distribution. The result table lists multiple critical planes per element (e.g., Element 7914) — described by global normal components (i, j, k) and associated life. This multi-plane approach captures directional sensitivity for anisotropic materials and 3D-printed components.

Ex-7 — Critical-Plane Fatigue (Shell vs Solid)





Element_ID	Surface	Plane-ID	critPlaneAngle_toElemEdgeG1G2(deg)	FatLife
276	top	1	13.288	557
276	top	3	24.229	980
276	top	2	0.159	1298
276	top	4	55.627	605650
276	bot	1	13.288	542
276	bot	2	21.599	699
276	bot	3	55.627	240210

Shell

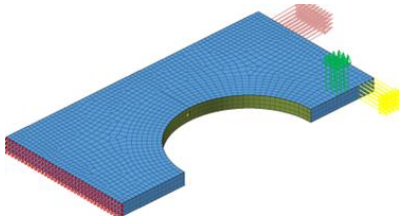
Element_ID	Plane-ID	planeNormalGlobal-i	j	k	FatLife
158	1	0.973	0.226	-0.049	2352
158	3	0.884	0.467	-0.033	5938
158	2	0.998	-0.044	-0.044	7289
158	4	0.653	0.757	-0.014	544380
284	1	0.65	0.589	0.48	32670
284	3	0.456	0.853	0.256	102710
284	2	0.905	0.362	0.224	118890
284	4	0.509	0.4	0.762	129460

Solid

- Shell models are fast but can miss interior peak stresses.
- Solid models capture subsurface critical planes for thick sections.
- FatDyn-X supports both — accurate life estimation regardless of geometry.

Ex-8 — Fatigue at Desired Points & Planes

- *Pick any geometric point — by Element ID (EID) or XYZ — and set a plane via global normal (k1, k2, k3).*
- *Useful where direction matters: 3D-printed parts, welds, and fatigue critical paths for crack-growth assessment.*



```
!Location,    Eid,        x,       y,    z,    k1,   k2,   k3,   Fat-Mid
 Location,  22441,      0.0,     5.0,  3.0,   0.0,  3.0,  0.0      22
 Location,      0,  -51.448,  -1.664,  4.5,   0.0,  0.0,  0.0      22
 Location,      0,  -51.448,  -1.664,  4.5,   2.0,  3.0,  0.0      23
 Location,      0,  -57.500,  -0.100,  4.5,   2.0,  3.0,  0.0      24
 Location,  20266,      5.0,     1.0,  3.0,   0.0, 13.0,  0.0      25
 Location,  25497,      5.0,     1.0,  3.0,   1.0,  3.0,  0.0      25
```

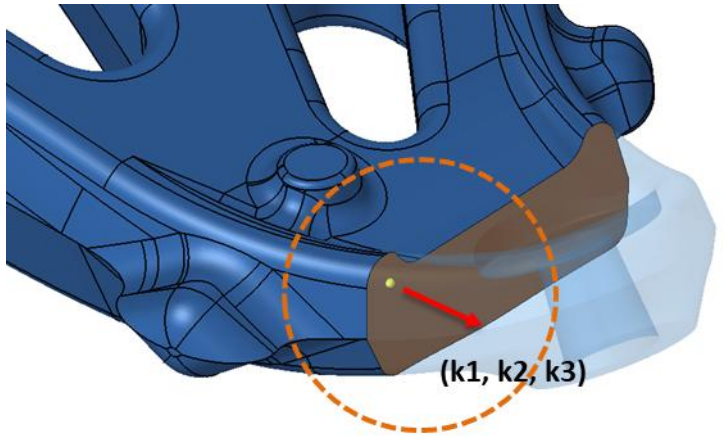
Results

Elem-No	Elem_Dim	Loc_MxPrLife	Loc_givenPlane_Life
22441	3	70.31	62.22
20341	3	5.76E+07	-1.00
20341	3	5.76E+07	2.00E+20
24550	2	2.00E+20	2.00E+20
20266	3	2.00E+20	2.00E+20
25497	2	42.05	66.10

***Locations can be* repeated; each repeat can have different plane and material**

Per requested location, fatigue life is reported by two methods:

- Desired Plane — life on the user-defined plane (normal k1, k2, k3)
- AbsMaxPr — life using the AbsMaxPr stress method
- Ignored-direction handling: if the desired-plane normal is (0, 0, 0), only AbsMaxPr fatigue is included.
- Stress time histories are also output for each chosen location.
- Locations may repeat — each repeat can use a different plane and material.



Ex-9 — Frame Fatigue at Desired Points & Planes

```
*FE_Data
        file_Name_FE_NasModel : trFrame_s101.fem
        file_Name_FE_PCH101   : trFrame_s101.pch
        file_Name_FE_PCH103   : trFrame_s103.pch

        rigid_modes  :  1 to 6
        modes        :  7 to 20  critical_damping = 0.10
        static_modes : 21 to 24
$-------------------------------------------------------------------------------
*Fatigue_Material_Mapping
        SN_Mean_Correction       : GERBER     ! GERBER, GOODMAN, or NO
        Map_to_Locations  YES

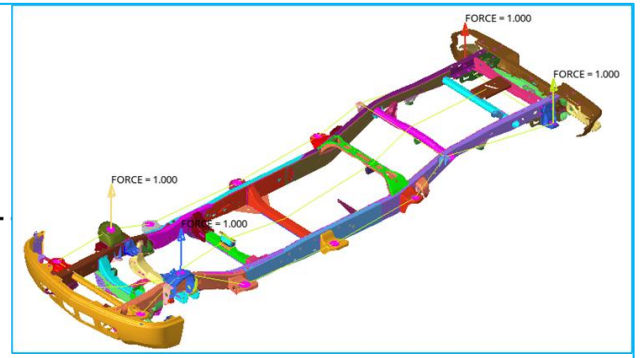
        Search_Radius:30   ! to identify elements if element-id is not given
!Location,      Eid,        x,       y,     z,    n1,   n2,     n3,   Fat-Mid
 Location,    13695,      0.0,     5.0,   3.0,   5.0,  0.0,    0.0,    2
 Location,        0, 5048.000, -457.0, 918.0,   0.0,  0.0,    6.0,    2
$-------------------------------------------------------------------------------
*TimeHistory_LoadCases
        Label: IR   ! load case 1
        Analysis_method  : IR_static
        Fatigue_Analysis : YES

Analysis_Time_Step  : 0.002
$ SerialNo   Label   loadChannel_fileName       noHeadLines   COL_id   divide     multiply     offset     node_id    dir  IRstatic_Loadcase
     1,      Fz_1,        Loading_0p8.csv,          1,          1,     1.00         6.0,        0.0,        2,       3         2
     2,      Fz_2,        Loading_0p8.csv,          1,          2,     1.00         6.0,        0.0,       12,       3         3
     3,      Fz_3,        Loading_0p8.csv,          1,          3,     1.00         6.0,        0.0,        7,       3         4
     4,      Fz_4,        Loading_0p8.csv,          1,          4,     1.00         6.0,        0.0,       17,       3         5

!===============================================================================================================================
        Label: MT   ! load case 2
        Analysis_method  : MODAL_TRANSIENT
        Fatigue_Analysis : YES

Analysis_Time_Step  : 0.002
$ SerialNo   Label   loadChannel_fileName       noHeadLines   COL_id   divide     multiply     offset     node_id    dir  IRstatic_Loadcase
     1,      Fz_1,        Loading_0p8.csv,          1,          1,     1.00         6.0,        0.0,        2,       3         2
     2,      Fz_2,        Loading_0p8.csv,          1,          2,     1.00         6.0,        0.0,       12,       3         3
     3,      Fz_3,        Loading_0p8.csv,          1,          3,     1.00         6.0,        0.0,        7,       3         4
     4,      Fz_4,        Loading_0p8.csv,          1,          4,     1.00         6.0,        0.0,       17,       3         5
$-------------------------------------------------------------------------------
*Fatigue_Material_Data_File
snenLife.txt

*END
```



FatDyn-X input

- FatDyn-X automatically identifies the corresponding element from XYZ when an EID is not provided.
- The global plane normal (n1, n2, n3) is transformed into the element coordinate system to use the element-frame stress tensor.
- Fatigue on the specified plane and via AbsMaxPr stress is computed for all load cases (IR-static and modal-transient).
- Modal-transient fatigue uses normal-mode results only — no SOL 112 results are required.
- Stress time histories at each requested location are also generated.

Frame Fatigue at Desired Points & Planes — Results

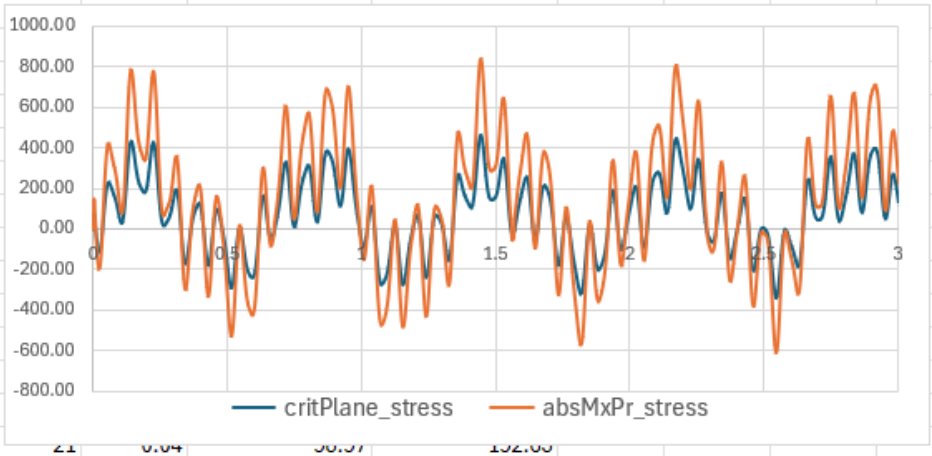
Inertia-Relief Fatigue Analysis					
Elem-No	Elem_Dim	Loc_MxPrDamage	Loc_MxPrLife	Loc_givenPlane_Damage	Loc_givenPlane_Life
13695	2	5.14130E-04	1.94500E+03	6.15360E-06	1.62510E+05
102344	2	4.61120E-04	2.16860E+03	2.69210E-04	3.71460E+03
Modal Transient Fatigue Analysis					
Elem-No	Elem_Dim	Loc_MxPrDamage	Loc_MxPrLife	Loc_givenPlane_Damage	Loc_givenPlane_Life
13695	2	4.16270E-03	2.40230E+02	6.76640E-05	1.47790E+04
102344	2	5.00000E-01	2.00000E+00	5.00000E-01	2.00000E+00

Key observations

- Fatigue is evaluated independently for IR-static and modal-transient load cases — same location, two damage mechanisms.
- Both desired-plane and AbsMaxPr stresses drive the calculation; both histories are plotted.
- Stress-time history at the selected plane provides full traceability for peak-stress validation.
- Results are computed automatically at user-defined points and planes — no predefined element IDs needed.

Stress time history for Element 13695, Surface-1

time-loc	time	critPlane_stress	absMxPr_stress
1	0	0.00	0.00
2	0.002	74.98	144.55
3	0.004	49.54	101.56
4	0.006	22.74	57.76
5	0.008	-4.42	17.74
6	0.01	-30.80	-54.24
7	0.012	-55.25	-94.84
22	0.042	128.44	243.93
23	0.044	155.06	290.29
24	0.046	178.09	330.26
25	0.048	197.04	362.87
26	0.05	211.63	387.67



SECTION 4

Brittle, Fracture and Weld Fatigue

Ceramics, glass and quartz; crack growth from FE stress histories; spot and seam welds from grid-point forces

Fatigue of Ceramics, Glass & Quartz — One Card, Two Mechanisms

Brittle materials break the metal-fatigue rules: there is no endurance limit, only tensile stress does damage, and time-at-stress (static fatigue) can matter more than cycle count. FatDyn-X adds a BR material type that carries both mechanisms on a single card, with every constant temperature-dependent.

Cyclic S-N — ceramic-like, active when k > 0

- $\sigma a(N,T) = KS(T) \cdot SaRef \cdot (10^6/N)^{(1/k)}$ — rainflow on tensile stress, no endurance limit, compression does no damage
- Constant-σmax mean-stress law: $\sigma a,eq = \sigma a + \max(\sigma m, 0)$ — a tensile mean is a full penalty, a compressive mean earns no credit

Slow crack growth (static fatigue) — glass & quartz-like, when n > 0

- $dD/dt = \sigma^n / [\, B \cdot (KS(T) \cdot Si)^{(n-2)} \,]$, failure at D = 1 — from crack velocity $v = A \cdot K^n$; B folds in A, Y and KIC
- Exact integral over every linear stress segment of the raw history — no rainflow, time-step invariant
- Both mechanisms may be active (alumina in humid air): D = D_cyc + D_scg

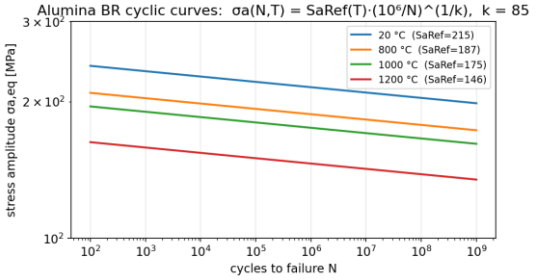
Variables: Si inert (fast-fracture) strength; n, A constants of the crack-velocity law $v = A \cdot K^n$; K stress-intensity factor; KIC fracture toughness; Y crack-geometry factor; $B = 2 / [\, A \cdot Y^2 \cdot (n-2) \cdot KIC^{(n-2)} \,]$ in $MPa^2 \cdot s$, so A, Y and KIC enter only through B; KS(T) strength ratio at temperature T; D damage, failure at D = 1.

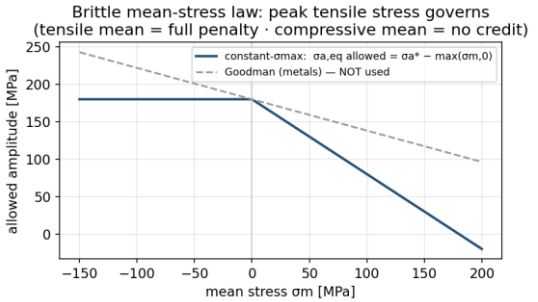
Temperature-dependent everything + overload check

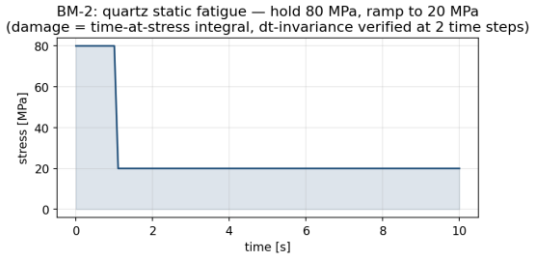
- k(T), SaRef(T), n(T), B(T), Si(T) tables per material — LOG (default) or LIN interpolation, constants at the per-point temperature
- Overload ratio SigMax / Si per element; D_cyc, D_scg, D_total and lives → CSV, HWascii and UNV contours (BRsplit files)

Material card — same *Fatigue_Material_Data block as SN / EN materials

```
*Fatigue_Material_Data
$ id typ label         k     SaRef  nSCG  Bscg   Si     KthR mWbl   E       nu
  7  BR  AL2O3_A480    85.0 215.0  41.0  1.0e4  430.0 0.0  12.0  380.0e3 0.23
  8  BR  SIC_SC1000  132.0 156.0   0.0  0.0    445.0 0.0  10.0  440.0e3 0.17
  9  BR  FQ_HPFS       0.0   0.0  27.0  4.0e5   99.0 0.0  12.0   73.0e3 0.16
*BR_Thermal_Table
$ matID    T      k     SaRef   nSCG  Bscg    Si
  7      20.0   85.0  215.00  41.0  1.0e4  430.00
  7    1200.0   85.0  145.99  25.0  1.0e4  291.97
```



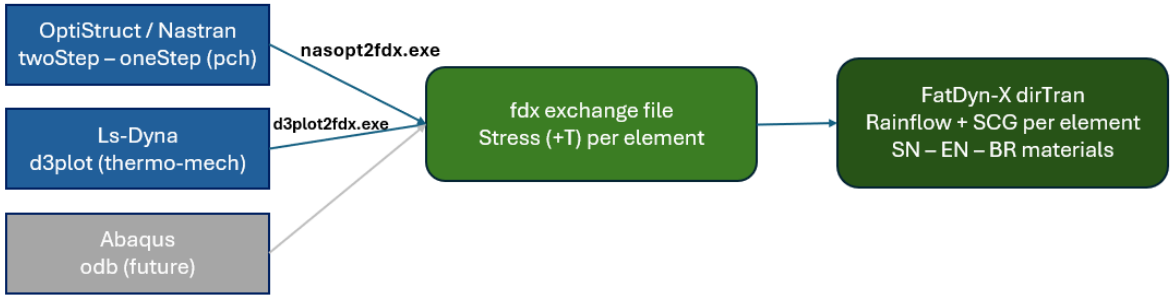




Why it matters

Quartz sensors, ceramic substrates, glass sight-ports and thermal-shock parts fail by time-at-stress, not by cycle count — a metal S-N tool cannot see it.

Direct-Transient Thermo-Mechanical Fatigue — Verified Every Step



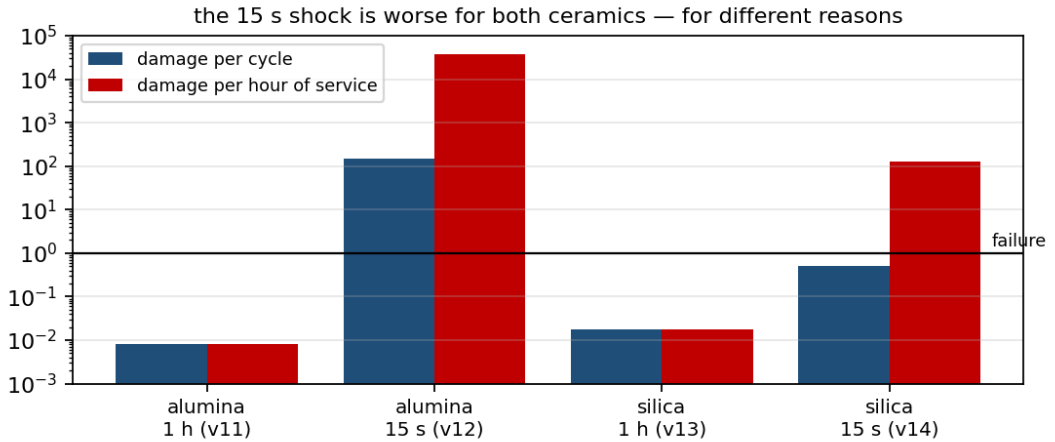
Direct-transient (dirTran): stress + temperature straight from the FE run

- OptiStruct / Nastran two-step and one-step thermal transients (punch) and LS-DYNA d3plot, translated to one FDX exchange file
- Solids and shells, mixed SN / EN / BR materials in one run; per-point temperature drives both damage mechanisms
- RETURN_TO_INITIAL unload ramp closes the cycle; SORT1 / SORT2 punch layouts and element sets handled
- Same rainflow, SCG kernel, gates and outputs as the time-history fatigue path — no superposition needed

Applied so far

- SiC quench tube (NASA TP-2005-212505): 456k elements in 6.6 min; 580 °C peak vs 580 °C measured
- OSTTS manifold: 58,318 CTETRA × 101 steps, 3.5 GB punch in 35 s; LS-DYNA door: 712k elements in 100 s
- Five-material process chamber: steel, SS304, Ti-6Al-4V, alumina and fused silica in one run

Benchmark	Reference	FatDyn-X
BM-1 SiC block cycles, k = 132	0.200267 (hand)	0.200270
BM-2 quartz 80 MPa hold, SCG only	0.778 (hand)	0.7808; dt ×10 → exact ×10
BM-3 alumina ±180 MPa at 900 °C	0.1392 (hand)	0.13915
B4 restrained thermal bar, per-point T	3.02819e-5	3.0282e-5 (438× the RT value)
B5 solid vs skin shell	identical	within 0.005 %
OptiStruct vs LS-DYNA twin models	—	within 0.2 %



Thermal shock vs thermal cycling on the same model: a 15 s shock is worse for both alumina (cyclic) and fused silica (SCG) — for different reasons.

Practical Fracture Mechanics — Crack Growth from FE Stress Histories

FatDyn-X extends its critical-plane and location / plane fatigue results to damage tolerance: the stress-tensor history at each hotspot is projected onto the crack plane and grown with standard LEFM laws. No cracked mesh, no separate fracture solver.

Growth laws

- PARIS · WALKER (R-ratio) · FORMAN · NASGRO (Forman–Newman–de Koning with Newman closure, threshold and Kc-instability terms) · TABLE da/dN–ΔK with R-curves
- Threshold ΔKth(R) = ΔKth0 · (1–R)^Cthp, El Haddad short-crack bridge and Kitagawa non-propagation screen; compression as Kmin → 0 or full

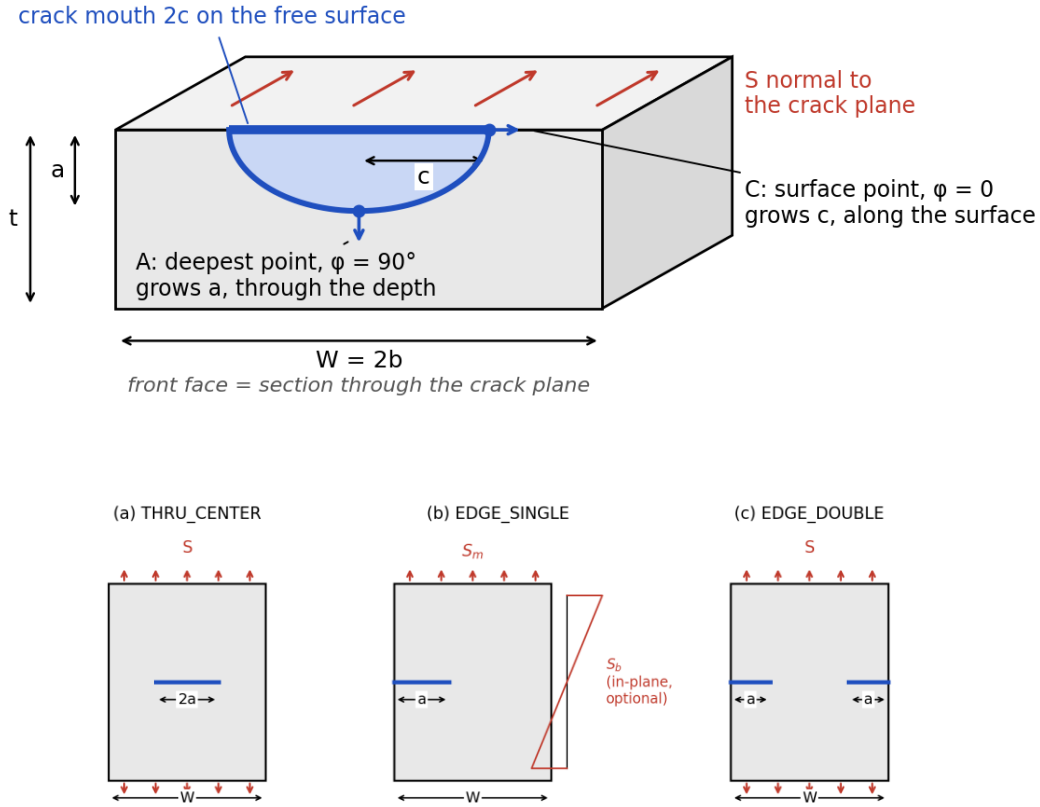
Stress-intensity library (β factors)

- THRU_CENTER (Feddersen) · EDGE_SINGLE (tension + in-plane bending) · EDGE_DOUBLE · THRU_HOLE (1 or 2 cracks, Bowie) · EMBED · USER_TABLE β(a)
- SURF_NR — Newman–Raju surface crack: depth a and half-length c grow independently; membrane + bending split from the two shell surfaces

Integration and stopping

- Rainflow → binned block spectrum → adaptive block integration with midpoint rule; cycle-by-cycle mode as a reference check
- Stops at Kmax ≥ Kc, a ≥ af, net-section yield or Nmax; a surface crack breaking through the thickness becomes a through crack

(a) SURF_NR: semi-elliptical surface crack, crack front seen through the depth



Surface crack: the front is semi-elliptical through the depth (a) and along the surface (c); the two tips grow at their own rates, with membrane and bending stress split from the two shell surfaces. Theory manual v0.5: equations, β-factor derivations, Appendix A K tools, Appendix B input reference.

Crack-Growth Workflow, Additive-Manufacturing Defects and Verification

1

Master deck — one card

Crack_Growth_Export : YES 10 10 RUN in any load case with FATIGUE_STRESS_METHOD CRITPLANE (top-N shell / solid hotspots), or Map_to_Locations for welds, AM build planes and postulated crack paths

2

FatDyn-X exports, crackGrow.exe runs

Per load case: fm_crackGrowth_<LC>\ with tensor histories of the top-N critical-plane elements, an fmSites.csv manifest (normal, thickness, material) and a starter fmRun.inp; RUN launches crackGrow.exe automatically

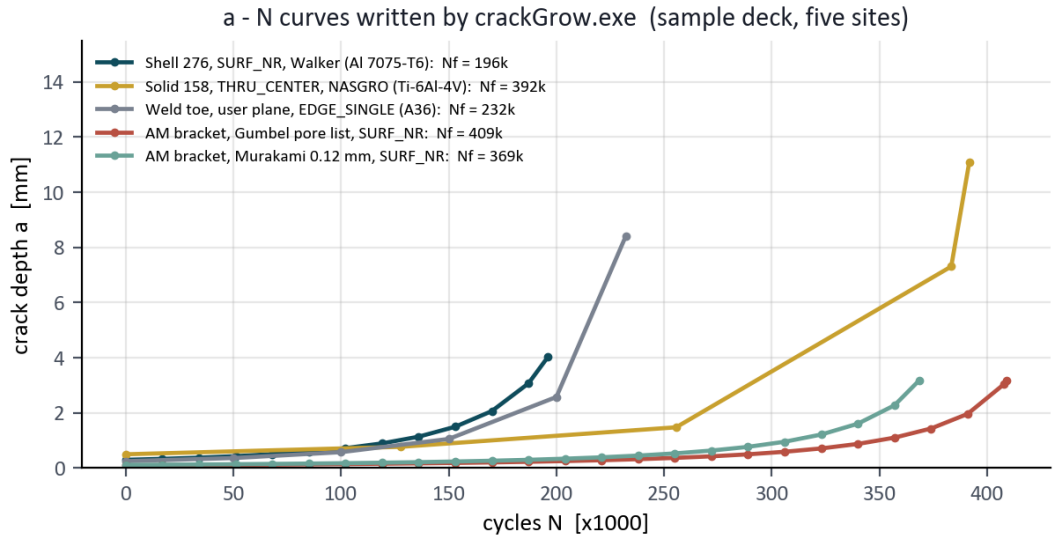
3

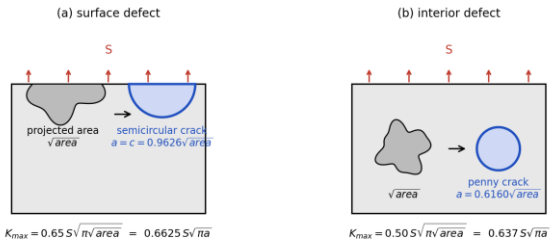
Outputs

Summary CSV — Nf in cycles and hours, stop reason, a_final, Kitagawa flag, inspection interval; a–N curve per site; HWascii / UNV contours of CrackGrowthLife, a_final and ΔKmax/ΔKth; PDF summary

Material row and one site — units, crack assumptions and material map are set once in fmRun.inp

```
*Fatigue_Material_Data
$ id typ label        law      C         m    gamma  Kc    dKth0  Cthp   E        nu
  30 FM  AL7075_T6    WALKER   2.7e-11   3.2  0.65   29.0  2.5    0.5    71.7e3   0.33
  32 FM  TI64_NASGRO NASGRO   1.0e-10   3.2  1.0    80.0  3.5    0.0    110.0e3  0.31
*Site
  id 1  elem 276  surface bot  label shell276_CP
  geom SURF_NR  thick 6.0  width 80.0   a0 0.3  c0 0.3  mat 30
  normal 0.973 0.230 0.0
  hist T2D2  hist_shell276_T2D2.csv  repeats 500
  defect MURAKAMI 0.12      ! or: defect LIST pores_sqrtArea.csv  V0 100  V 25000
```





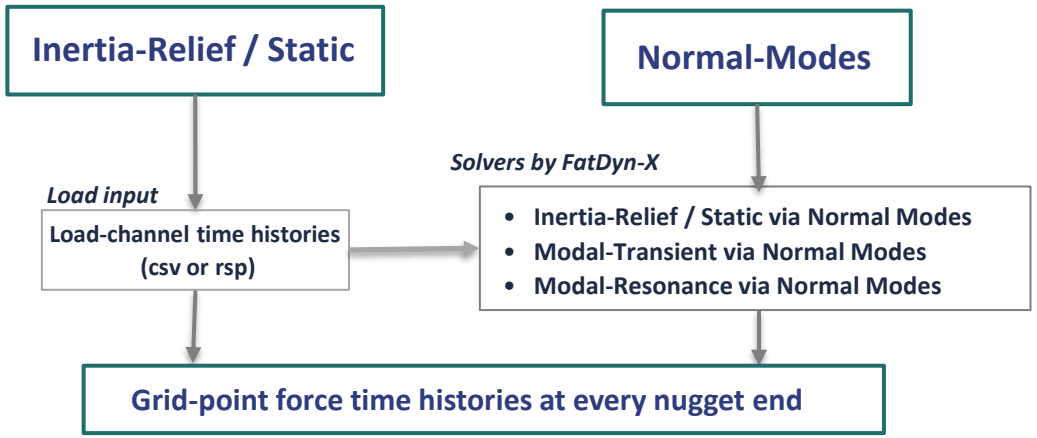
AM defect route: Murakami Varea equivalent crack, or a Gumbel extreme-value design defect fitted to a CT pore list.

Verified

- **fmTest.exe: 38 PASS / 0 FAIL**
- Paris closed form 7e-7; Walker, NASGRO, rainflow and Gumbel exact
- Newman–Raju 0.6625, Bowie 3.36, Feddersen 1.1892
- End-to-end via the FatDyn-X export: 17,589 vs 17,587 cycles (0.01 %)

Spot-Weld Fatigue using FatDyn-X

Force-based fatigue of CBAR, CBEAM, CBUSH, CWELD and ACM-hexa spot welds from grid-point force time histories




```
*Weld_Definition
   Spot_1D_Set    beams          ! CBAR CBEAM CBUSH CWELD (element set)
   Spot_3D_Set    solid          ! ACM hexas: 1 or more per nugget, 2T / 3T stacks
   Spot_Method    RUPP  NAKAHARA  DONG
   Sheet_Mat 50   Nugget_Mat 51  Dong_Spot_Mat 60
   Weld_Diameter    5.0
   Angle_Increment 10.0          ! 36 angles

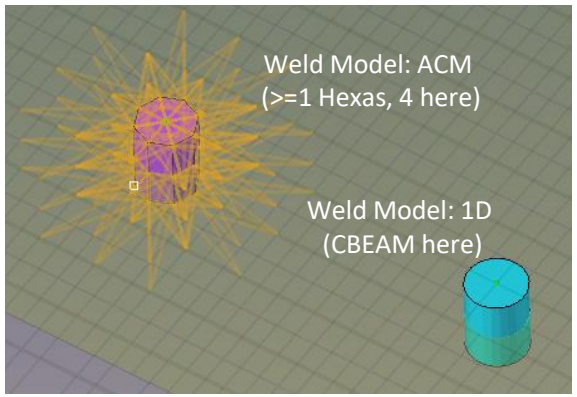
*Element_sets
   SET beams = 3315 THRU 3326
   SET solid = 3327 THRU 3332
```

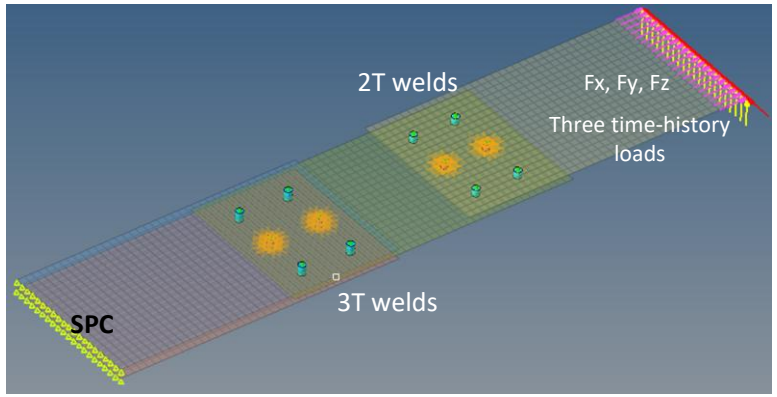
- **Sheet and nugget structural stress at each angle around the nugget**
- Fx, Fy = shear, Fz = axial (tension only), Mx, My = bending at the sheet mid-plane
- d = Weld Diameter, t = sheet thickness from bulk data
- Rainflow per angle → Miner damage, worst angle reported
- Curves: *Fatigue_Material_Data rows SW (sheet, nugget)
  - range-based SRI, b1, b2, Nc1
  - mean correction NONE / FKM

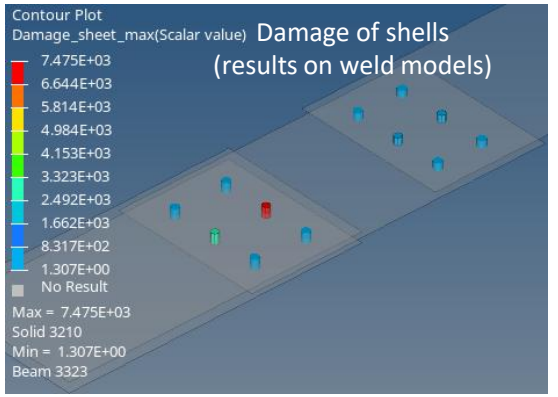
What the weld program does

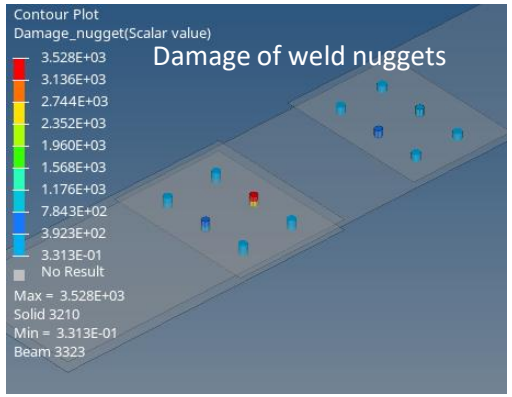
- Reads the bulk data: connected sheets and thicknesses, nugget ends, hexa groups → nuggets and layers (3T stacks found automatically)
- Rotates the grid-point forces of each nugget end into the weld system; sheet 1, sheet 2 (sheet 3 for 3T) and nugget evaluated separately
- Writes CSV summary, stress histories and HWascii / UNV contours: Damage_sheet1/2/3, Damage_nugget, Life_sheet_min, Life_nugget

- Spot-weld fatigue by three methods
- Nearly exact comparison to established benchmarks



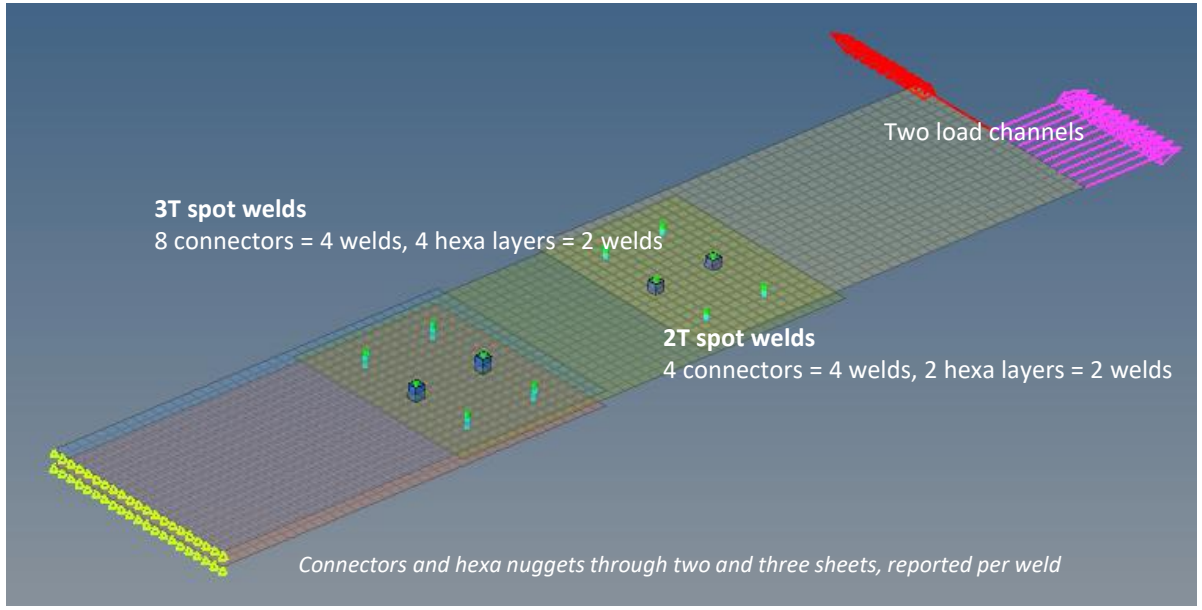


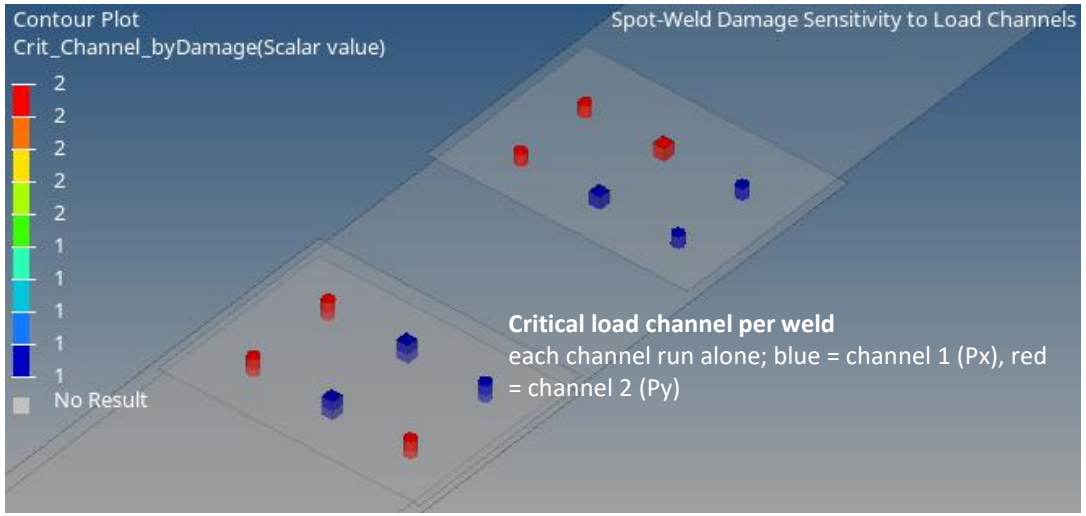


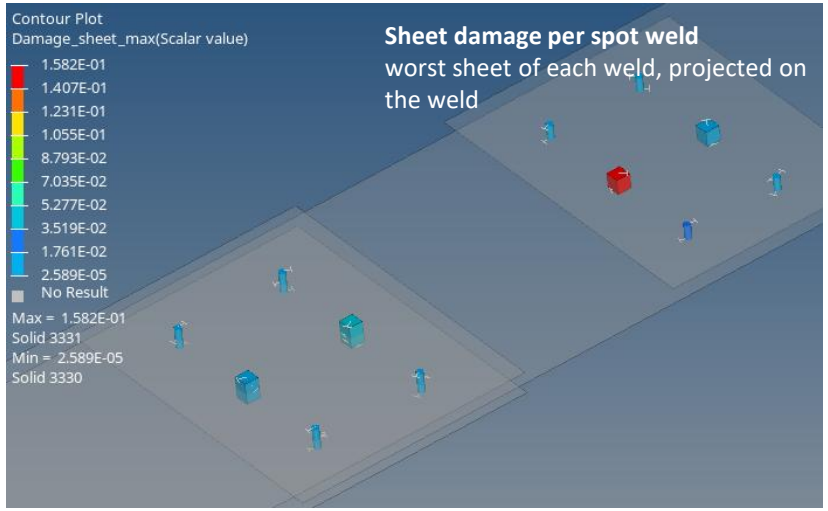


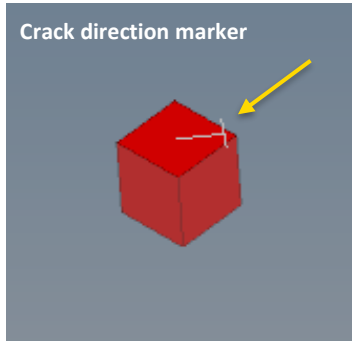
Crack Direction and Damage Sensitivity of Spot Welds in FatDyn-X

Where a spot weld would crack, and which load channel drives it: two design aids from one fatigue run on the same model.









A PLOTEL line at the governing angle of every weld: the site and the direction of initiation, read in any viewer

Weld	Location	Ref damage	Crit ch by range	Stress range	Crit ch by rms	Stress rms	Crit ch by damage	Damage
3315	sheet 2	9.88E-03	1	629.6	1	358.4	1	2.91E-03
3319	sheet 2	1.05E-03	2	390.9	2	228.6	2	1.68E-04
3326	sheet 2	1.97E-02	1	696.6	1	396.6	1	5.34E-03
3327	sheet 2	4.83E-02	1	1001.1	1	569.9	1	4.70E-02
3331	sheet 2	1.58E-01	1	1217.9	1	693.4	1	1.52E-01
3332	sheet 2	1.13E-03	2	375.9	2	219.8	2	1.32E-04

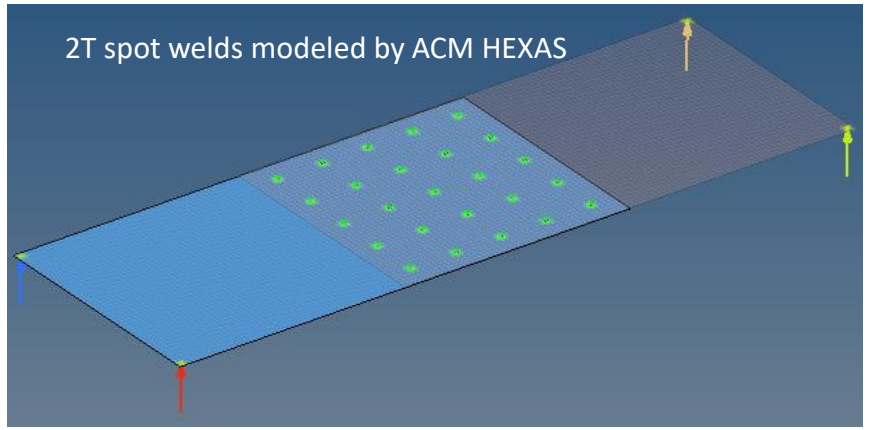
Six of the twelve welds of <run>_weldSens_summary.csv: stress range, rms and damage of the channel that is critical on its own; MPa and damage per history

Welds, not elements. A three-sheet stack of two elements is one weld, with sheet and nugget rows.
Each channel alone. Stress range, rms and damage per channel or mode, ranked against the all-channel reference.
Initiation only. The markers give site and direction; growth depends on the local field, the next FatDyn-X step with crack growth.

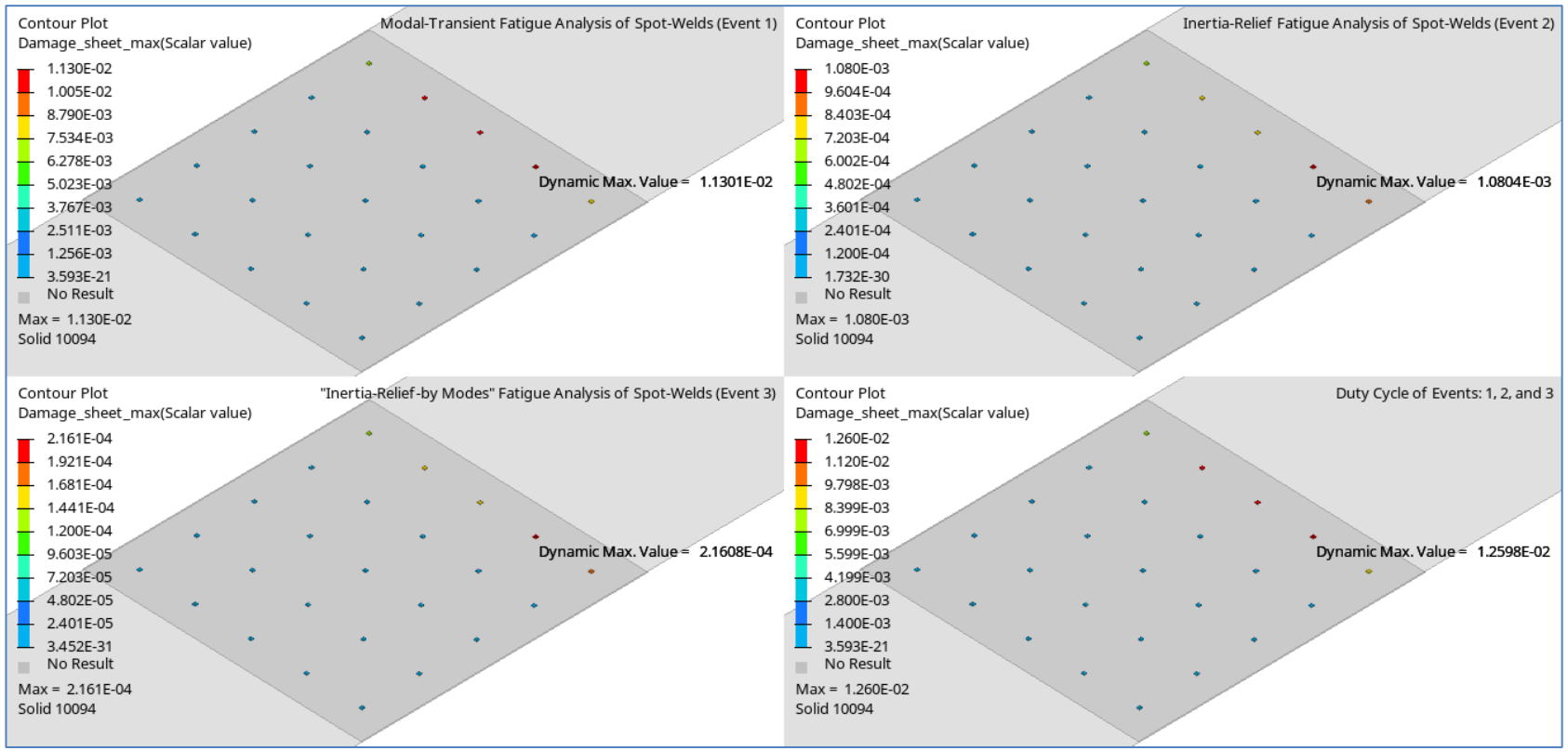
```
SpotWeld_Damage_Sensitivity  YES  swElems  -1.0  RUPP  SHEET      weld_plotel: yes
```

Duty-Cycle Analysis of Spot-Weld Fatigue using FatDyn-X

Modal-transient, inertia-relief and inertia-relief-by-modes events summed into one damage map: two of the three routes from a single normal-modes solve, and the dynamics carrying 90 % of the damage



Identical load history in all three events (three file names, one history), so the columns compare the solves alone	Modal-Transient	Inertia-Relief	Inertia-Relief-by-Modes
Event / repeats per duty block	event 1 x 100	event 2 x 1000	event 3 x 200
Solver result used	normal modes 7-15 (5 % damping) and static modes 16-19: one modes solve	static inertia-relief solve, one subcase per load channel	the same modes file as modal-transient: no static solve at all
Response	dynamic: resonance and damping of the history	quasi-static	quasi-static, assembled from the modes
Max sheet damage, weld 10094	**1.130E-02**	**1.080E-03**	**2.161E-04**
Damage per repeat	**1.13E-04**	**1.08E-06**	**1.08E-06 (= inertia-relief)**
Share of the duty damage	**89.7 %**	**8.6 %**	**1.7 %**



Duty cycle: what the sum says

1.26E-02 sheet damage per duty block

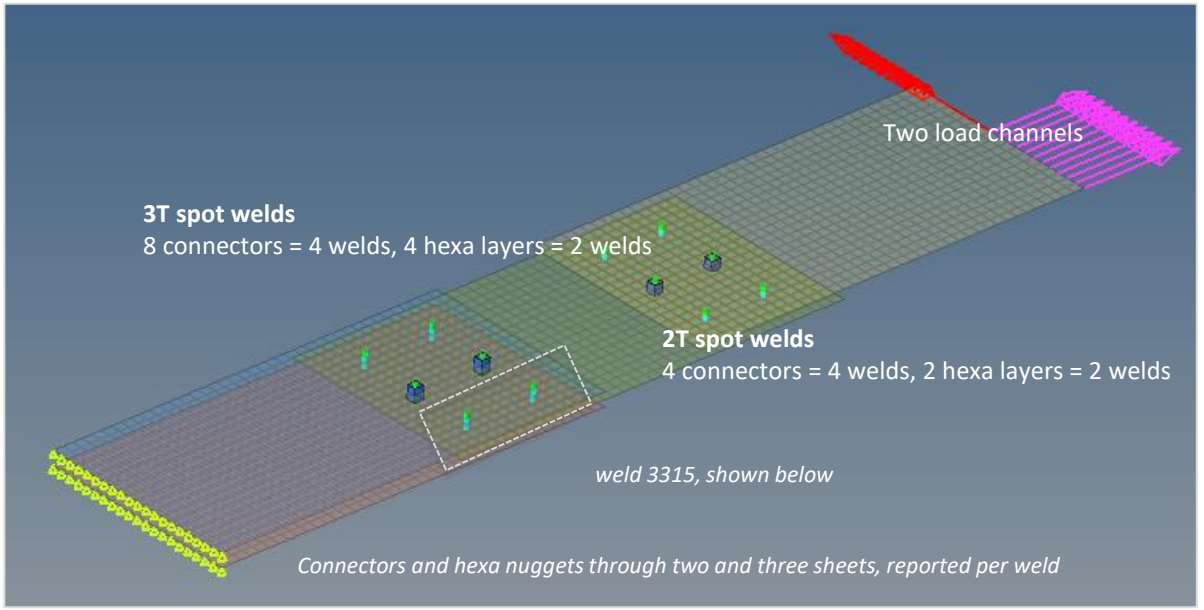
at weld 10094, sheet 1, 90 deg: 79 duty blocks to initiation

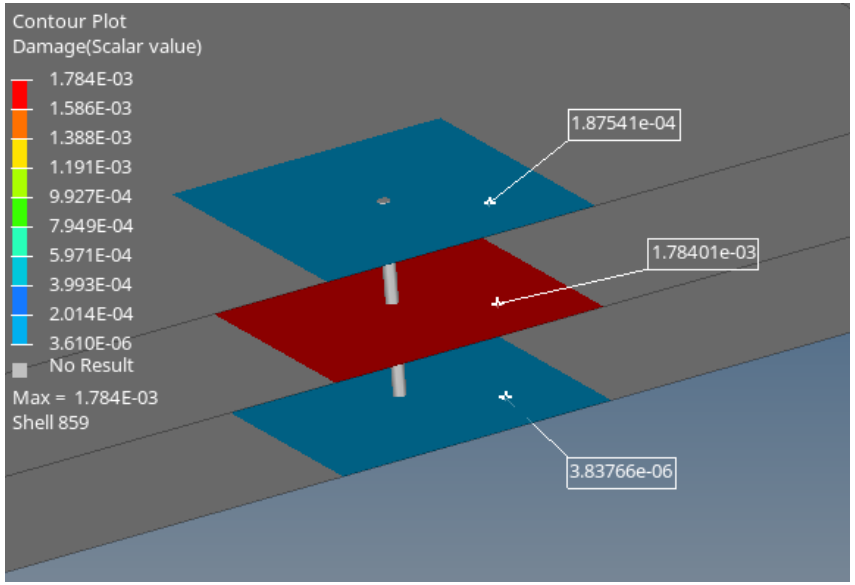
- The three events carry the identical load history under three file names, so the differences are the solves alone: modal-transient matters, and inertia-relief-by-modes equals inertia-relief.
- One duty block = 100 x event 1 + 1000 x event 2 + 200 x event 3. The damage is summed per weld and per angle (36 angles) before the maximum is taken, so the crack direction is that of the summed damage, not of one event.
- At the critical weld the modal-transient event carries 90 % of the duty damage, the two inertia-relief events 8.6 % and 1.7 %.
- Inertia-relief-by-modes reproduces the static inertia-relief solve to four digits per repeat (1.08E-06) from the normal-modes result alone.
- Per repeat, modal-transient gives 105 x the quasi-static damage for the same loads: the dynamics at 5 % damping roughly double the stress range. A quasi-static duty cycle would have missed it.
- Two of the three routes need only one normal-modes solve. We know of no other fatigue tool that runs inertia-relief fatigue from normal modes.

Spot-Weld Fatigue in FatDyn-X: 3T Stacks and Damage Contours on the Sheets Themselves

Sheet 1, sheet 2, inner sheet and nugget each report their own damage, contoured on the shells around the weld and on the crack-direction markers

- **Applied:** on an EV body with more than 5,000 spot welds and load histories of over a million time-steps, 2T and 3T welds in the same run.
- **3T stacks:** the inner sheet is loaded by both nugget elements. Their forces are summed at the shared sheet, the free body of that sheet, and both faying faces are evaluated. The two elements of a stack then report one inner-sheet damage.
- **Checked** on a double-shear coupon with load cells between nugget and sheets: the middle sheet takes the full load with no moment, each outer sheet half. Connector and hexa nuggets agree on the inner sheet; 2T welds compare closely with an established benchmark.
- **Reported by plate,** not by element end: sheet 1 and sheet 2 (the outer sheets), inner sheet and nugget, under the same names in the contours and the CSV files.
- **Two complete contours** from the same run, shown below:
  1. On the shell elements around each weld, found through the weld's own connection (direct, RBE2 or RBE3). No include file.
  2. On the crack-direction PLOTEL markers, with a named set per plate to switch sheet 1, sheet 2, inner sheet and nugget on or off.





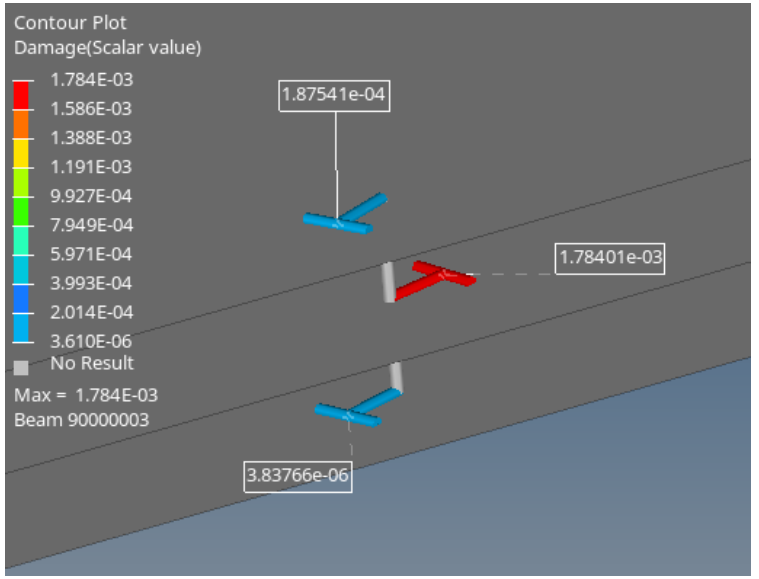
1. *Sheet damage on the shell elements around the weld*

Weld 3315, a 3T stack: what the contours say

1.78E-03 inner-sheet damage (4.0 mm sheet)

sheet 1 (3.0 mm) 1.88E-04 sheet 2 (2.5 mm) 3.84E-06 nugget 3.42E-02

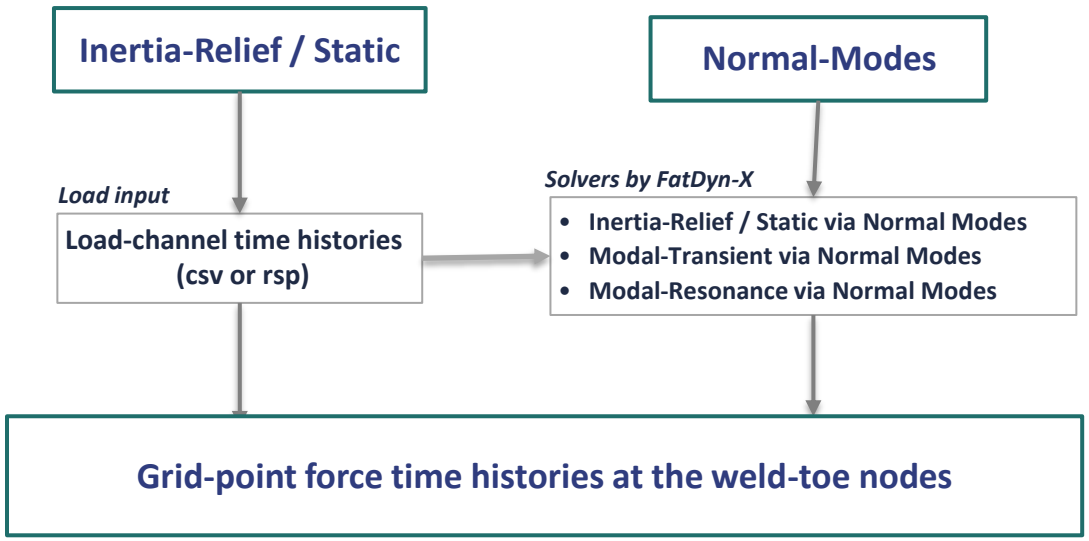
- The inner sheet is the thickest of the three and still has the highest sheet damage, 9.5 times that of sheet 1: it takes the load transfer of both interfaces.
- Left and right show the same three numbers, on the shells of each sheet and on the crack-direction markers. The markers add the angle of initiation.
- The nugget (grey) is left out of both plots to keep the sheets readable. Its own damage, 3.42E-02, governs this weld.
- *Not yet shown: agreement with a 3T fatigue test. The inner-sheet load follows the free body of each sheet; test data is the next step.*



2. *The same damage on the crack-direction markers*

Seam-Weld Fatigue using FatDyn-X

Line-force structural stress at the weld toes from grid-point forces



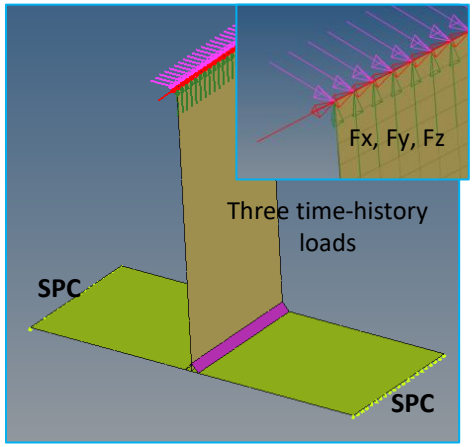
- **Line forces → structural stress at every toe element**
- Membrane and bending stress from f, m and the sheet thickness t; two S-N curves interpolated by the bending ratio, or master-curve equivalent stress
- f, m per unit length from the nodal forces of the toe edge
- Rainflow on the structural stress history
  → Miner damage per toe element
  → both toes of a double fillet, weld ends included
- Curves: *Fatigue_Material_Data rows SM
  - mean stress: NONE or Kim-Dong (1−R) rule

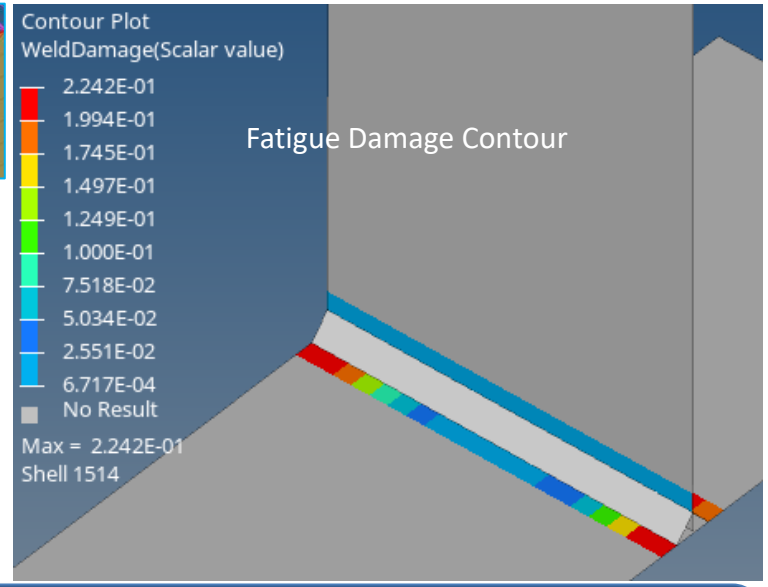
- Seam-weld fatigue by two methods
- Nearly exact comparison to established benchmarks

```
*Weld_Definition
   Weld_PIDs      5              ! PSHELL id(s) of the weld bead elements
   Seam_All       YES            ! every sheet element in front of the bead
   Seam_Method    DONG  VOLVO
   Seam_Mat       60    Seam_Mat_Volvo 61
```

What the weld program does

- Finds the toe elements from the bulk data: sheet elements sharing an edge with the weld-bead, in front of the weld (elements under the bead are listed as root; root damage is a planned option)
- Sums the grid-point forces of the toe nodes, forms the line forces and the membrane / bending stress histories per toe element
- Writes CSV summary (toe element, weld element, side, rbar, damage per method), stress histories and HWascii / UNV contours: damage and life per method (DONG, VOLVO)





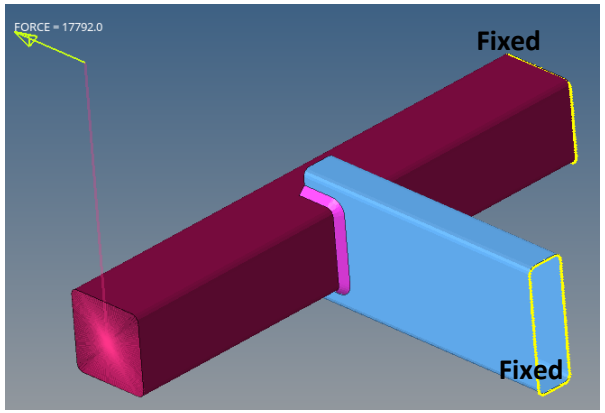
- Benchmark on public test data: SAE FD&E weld-challenge T-tube, 12 specimens, 4000 lbf, R = −1
- FatDyn-X 77,000 cycles at the test crack site; tests 75,000 mean (29,000–190,000); toe stress within 7 % of the published value

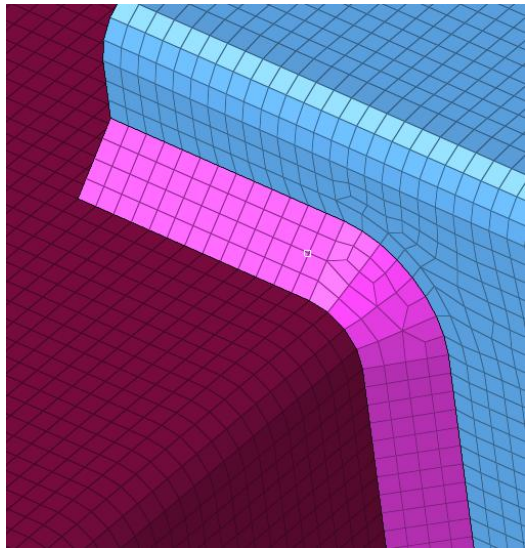
Benchmark: SAE FD&E Weld-Challenge T-Tube with FatDyn-X

Public test data (12 specimens) and the 2003 committee round of blind predictions, all methods and codes

The benchmark

- 4 x 4 in tube (24 in, one end clamped, one end loaded) through a 2 x 6 in stub (base clamped); wall 0.312 in, 5/16 in fillet; A36 steel
- 4000 lbf (17.8 kN) at a 12.5 in lever on the free 4 x 4 end
- FatDyn-X: shell model, toe-to-toe weld shells, stub wall merged to the tube (full penetration); both methods in one run



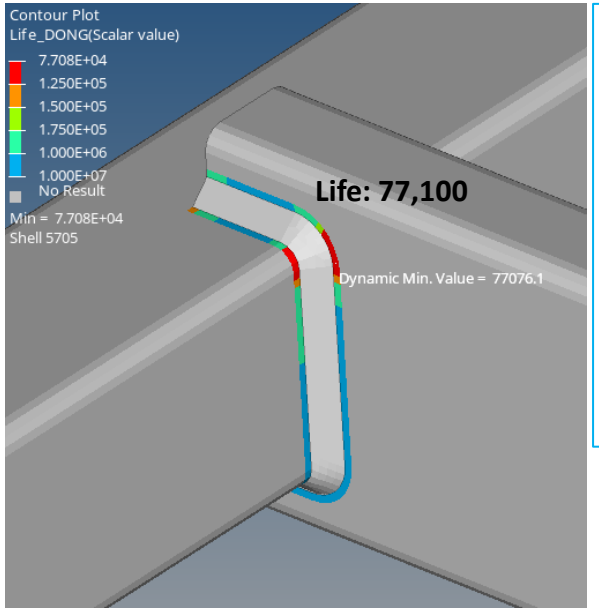


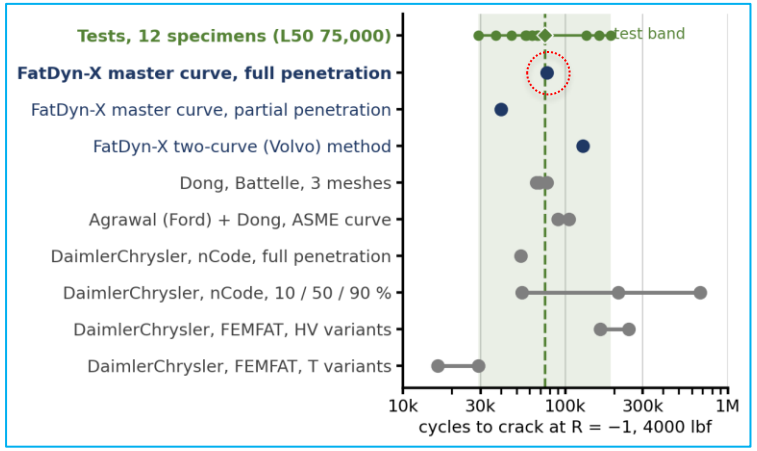
Who / method	N at R = 0	N at R = –1	Remark
Tests, 12 specimens (John Deere)	**—**	**75,000 (29k–190k)**	**crack at the stub-wall corner**
FatDyn-X, master curve (ASME mean), full penetration	**674,700**	**77,100**	**test site; stress +7 %**
FatDyn-X, same model, 8 mm / 16 mm meshes	551,400 / 514,200	63,000 / 58,700	mesh-insensitive (nodal)
FatDyn-X, master curve, partial penetration	349,800	40,000	stub wall not merged
Dong, Battelle: structural stress, 3 meshes	549k–641k	66k–77k	closest in 2003; own curve
Agrawal (Ford) + Dong, ASME mean curve	877k / 750k	105k / 90k	coarse / fine; same curve
FatDyn-X, two-curve (Volvo) method, full penetration	1,043,600	128,000	published Volvo curves
DaimlerChrysler, nCode: 3D model, full penetration	580,000	53,000	
DaimlerChrysler, nCode: 10 / 50 / 90 % penetration	155k/640k/2,100k	54k/211k/674k	penetration: life × 13
DaimlerChrysler, FEMFAT: HV, fixed / not as fixed	9,870k / 7,300k	246k / 164k	
DaimlerChrysler, FEMFAT: T, fixed / not as fixed	244k / 406k	16k / 29k	
FatDyn-X, skidder history (Challenge 2), x19.2 / x27.1	**2,859 / 972 blocks**	**tests 1,750–2,161 / 450**	**first crack, blocks**

Source of the 2003 rows: SAE Fatigue Design and Evaluation Committee, Weld Challenge 1 results presented 16 April 2003 (fde.uwaterloo.ca case studies); lives as published at 4000 lbf

```
*Weld_Definition
   Weld_PIDs       3              ! weld-shell property
   Seam_All        YES
   Seam_Method     DONG  VOLVO    ! master curve, two curves
   Seam_Mat        60    Seam_Mat_Volvo  61
```

- Fatigue from the grid-point forces along the line between weld and toe elements
- Model set-up as shown: shell tubes, one row of weld shells, one deck block
- Predicted life at the test crack site falls within the scatter of the twelve tests





FatDyn-X prediction against the tests

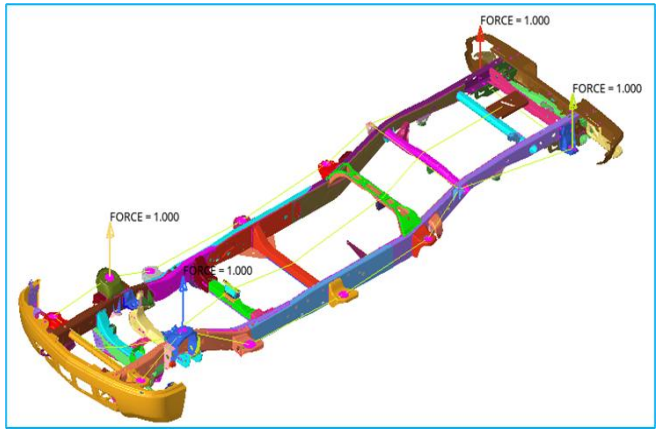
SECTION 5

Ex-10 — Fatigue Damage Sensitivity to Loads and Modes

Which load channel or mode causes maximum damage? FatDyn-X finds it.

Truck-Frame Fatigue — IR-Static and Modal-Transient Setup

FatDyn-X performs both Inertia-Relief / Static and Modal-Transient fatigue in one unified workflow.



- Performs IR-Static and Modal-Transient fatigue analysis in a single run.
- Modal-Transient fatigue uses normal-mode results only — no SOL 112 required.

```
*FE_Data
        file_Name_FE_NasModel : trFrame_s101.fem
        file_Name_FE_PCH101   : trFrame_s101.pch
        file_Name_FE_PCH103   : trFrame_s103.pch

        rigid_modes  :  1 to 6
        modes        :  7 to 20  critical_damping = 0.10
        static_modes : 21 to 24
$-------------------------------------------------------------------------------------------------------------
*Fatigue_Material_Mapping
        SN_Mean_Correction       : GERBER     ! GERBER, GOODMAN, or NO

        EN_Mean_Correction       : Morrow     ! SWT, Morrow, or NO
        NEUBER_PLASTIC_CORRECTION: YES        ! NO

        map_ALL_to_FatMID        : YES   2  ! stress-life steel
$-------------------------------------------------------------------------------------------------------------
*TimeHistory_LoadCases
        Label: IR   ! load case 1
        Analysis_method: IR_static

         FATIGUE_STRESSMETHOD(S) :  ABSMAXPR
         Fatigue_Analysis : YES

Analysis_Time_Step  : 0.002
$ SerialNo   Label   loadChannel_fileName       noHeadLines   COL_id   divide     multiply     offset     node_id     dir  IRstatic_Loadcase
     1,      Fz_1,        Loading_0p8.csv,         1,          1,      1.00        6.0,        0.0,        2,          3         2
     2,      Fz_2,        Loading_0p8.csv,         1,          2,      1.00        6.0,        0.0,        12,         3         3
     3,      Fz_3,        Loading_0p8.csv,         1,          3,      1.00        6.0,        0.0,        7,          3         4
     4,      Fz_4,        Loading_0p8.csv,         1,          4,      1.00        6.0,        0.0,        17,         3         5

!*************************************************************************************************************************************
        Label: MT   ! load case 2
        Analysis_method: MODAL_TRANSIENT

         FATIGUE_STRESSMETHOD(S) :  ABSMAXPR
         Fatigue_Analysis : YES

Analysis_Time_Step  : 0.002
$ SerialNo   Label   loadChannel_fileName       noHeadLines   COL_id   divide     multiply     offset     node_id     dir  IRstatic_Loadcase
     1,      Fz_1,        Loading_0p8.csv,         1,          1,      1.00        6.0,        0.0,        2,          3         2
     2,      Fz_2,        Loading_0p8.csv,         1,          2,      1.00        6.0,        0.0,        12,         3         3
     3,      Fz_3,        Loading_0p8.csv,         1,          3,      1.00        6.0,        0.0,        7,          3         4
     4,      Fz_4,        Loading_0p8.csv,         1,          4,      1.00        6.0,        0.0,        17,         3         5
$-------------------------------------------------------------------------------------------------------------
*Fatigue_Material_Data_File
snenLife.txt

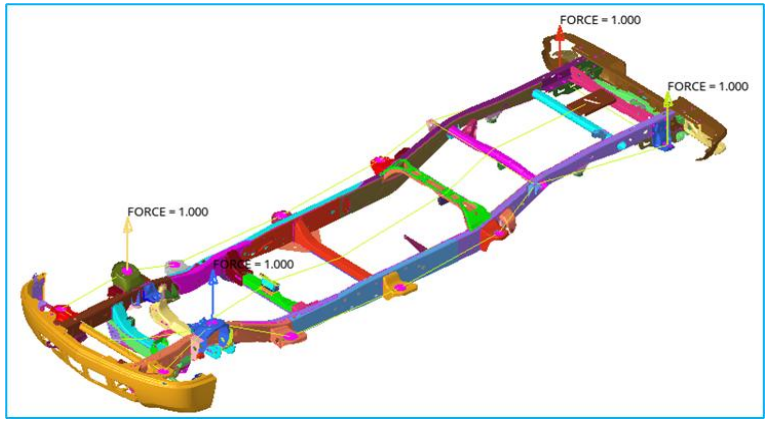
$-------------------------------------------------------------------------------------------------------------
*END
```

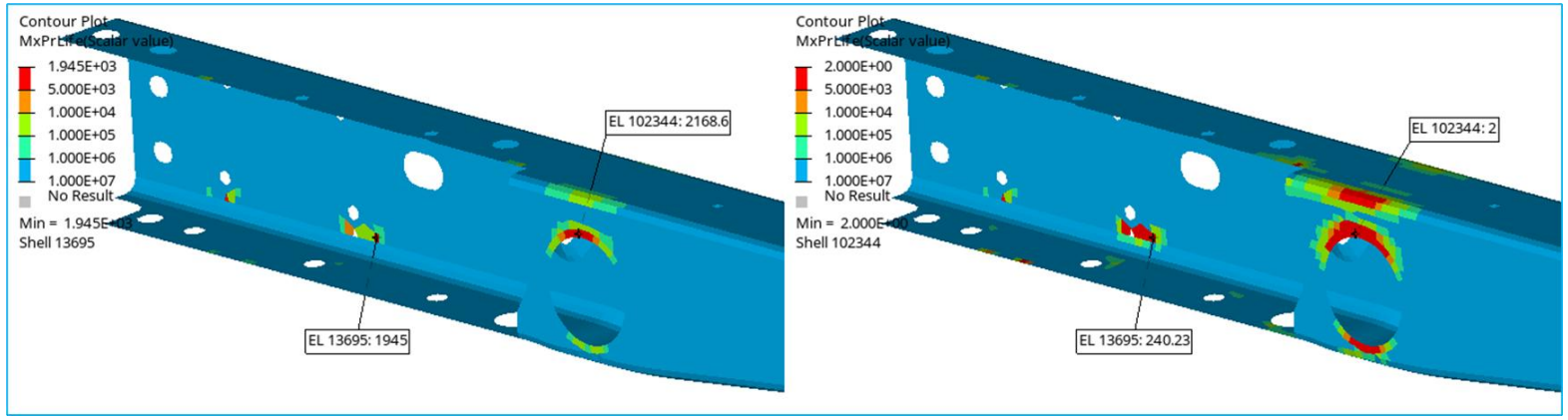
Used for Inertia-Relief/Static fatigue analysis

Used for modal-transient fatigue analysis

Truck-Frame Fatigue — IR-Static vs Modal-Transient

Fatigue life by SN-Life with Gerber mean-stress correction.





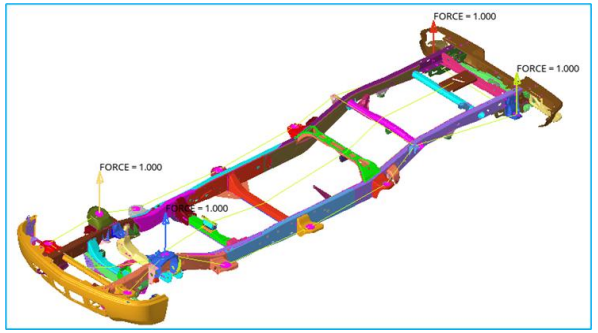
Min Life — element comparison

Element	IR-Static	Modal-Transient
13695	1,945	240
102344	2,169	2

- Which load channel or mode drives fatigue damage? FatDyn-X performs damage sensitivity ranking automatically.
- For modal-transient, FatDyn-X also reports dynamic amplification per mode — useful for damping adjustments.

Fatigue Damage Sensitivity — Setup

Element set → FE data → Load cases → Sensitivity command.



- Specify element set and the load / mode definitions.
- FatDyn-X automatically runs damage sensitivity for both static and modal-transient loads.

```
*ELEMENT_Sets
  Label  fatCrit
  13695, 102344

*FE_Data
        file_Name_FE_NasModel : trFrame_s101.fem
        file_Name_FE_PCH101   : trFrame_s101.pch
        file_Name_FE_PCH103   : trFrame_s103.pch

        rigid_modes  :  1 to 6
        modes        :  7 to 20  critical_damping = 0.10
        static_modes : 21 to 24
$-------------------------------------------------------------------------------------------------------------------------------
*TimeHistory_LoadCases
        Label: IR   ! load case 1
        Analysis_method: IR_static

        Fatigue_Damage_Sensitivity_to_Loads/Modes:  YES  fatCrit  5.0   2   ! Yes/No element-set load-gate  fat-mid

Analysis_Time_Step  : 0.002
$ SerialNo   Label   loadChannel_fileName       noHeadLines   COL_id   divide     multiply     offset     node_id     dir  IRstatic_Loadcase
     1,      Fz_1,        Loading_0p8.csv,         1,          1,     1.00         6.0,        0.0,        2,         3         2
     2,      Fz_2,        Loading_0p8.csv,         1,          2,     1.00         6.0,        0.0,       12,         3         3
     3,      Fz_3,        Loading_0p8.csv,         1,          3,     1.00         6.0,        0.0,        7,         3         4
     4,      Fz_4,        Loading_0p8.csv,         1,          4,     1.00         6.0,        0.0,       17,         3         5

!*******************************************************************************************************************************
        Label: MT   ! load case 2
        Analysis_method: MODAL_TRANSIENT

        Fatigue_Damage_Sensitivity_to_Loads/Modes:  YES  fatCrit  5.0   2   ! Yes/No element-set load-gate  fat-mid

Analysis_Time_Step  : 0.002
$ SerialNo   Label   loadChannel_fileName       noHeadLines   COL_id   divide     multiply     offset     node_id     dir  IRstatic_Loadcase
     1,      Fz_1,        Loading_0p8.csv,         1,          1,     1.00         6.0,        0.0,        2,         3         2
     2,      Fz_2,        Loading_0p8.csv,         1,          2,     1.00         6.0,        0.0,       12,         3         3
     3,      Fz_3,        Loading_0p8.csv,         1,          3,     1.00         6.0,        0.0,        7,         3         4
     4,      Fz_4,        Loading_0p8.csv,         1,          4,     1.00         6.0,        0.0,       17,         3         5

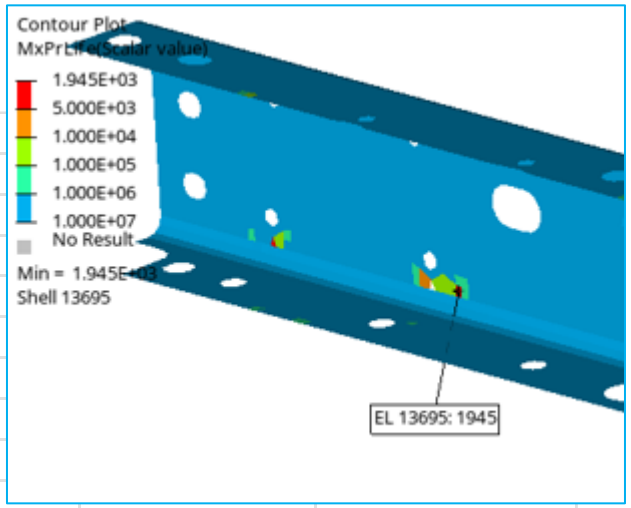
$-------------------------------------------------------------------------------------------------------------------------------
*Fatigue_Material_Data_File
snenLife.txt

$-------------------------------------------------------------------------------------------------------------------------------
*END
```

Element-Set to check for damage sensitivity

Load-Channel Sensitivity — Element 13695

Load channel 3 dominates the fatigue damage — ~30% of isolated damage and ~100% importance for Element 13695.



Fatigue Damage Sensitivity Analysis

==========

Element ID	13695
Reference Stress Range	1.43880E+03
Reference Damage	1.02830E-03

Analysis Method:

- Only Chan_i: Damage from isolated channel i
- w/o Chan_i: Damage with channel i removed from full channel-list

Channel_i	StressRange_Only_i	Damage_Only_i	StressRange_Without_i	Damage_Without_i	Contribution_%	Importance	Delta_Damage_%
1	1.02790E+02	4.21440E-12	1.40250E+03	9.90150E-04	0.000	0.037	3.706
2	9.57580E+01	2.08460E-12	1.43330E+03	8.56600E-04	0.000	0.167	16.694
3	1.15780E+03	3.00990E-04	5.32510E+02	6.26950E-07	29.272	0.999	99.939
4	3.98470E+02	1.31810E-07	1.27070E+03	4.58060E-04	0.013	0.555	55.452

Legend:

- Contribution_% = 100 * Damage_Only_i / Reference_Damage
- Importance = (Ref_Damage - Damage_Without_i) / Ref_Damage
- Delta_Damage_% = 100 * (Ref_Damage - Damage_Without_i) / Ref_Damage
  - (+) = damage decreases without channel i
  - (-) = damage increases without channel i

Mode Sensitivity — Element 13695

FatDyn-X mode sensitivity clearly shows Mode-7 drives the fatigue damage of Element 13695.

Fatigue Damage Sensitivity Analysis

==========

Element ID	13695
Reference Stress Range	1.85850E+03
Reference Damage	8.32540E-03

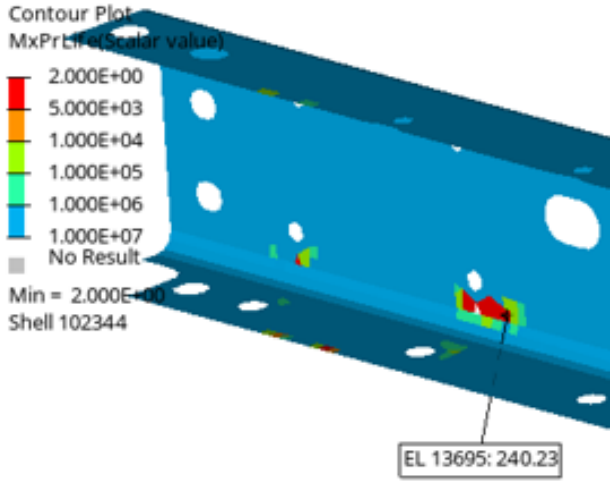
Analysis Method:

- Only Mode_i: Damage from isolated mode-i
- w/o Mode_i: Damage with mode-i removed from full mode-list

Mode_i	StressRange_Only_i	Damage_Only_i	StressRange_Without_i	Damage_Without_i	Contribution_%	Importance	Delta_Damage_%
7	1.19550E+03	3.07580E-04	8.59340E+02	2.66680E-05	3.6940	0.9970	99.6800
8	2.66840E+01	1.59850E-16	1.85890E+03	8.35180E-03	0.0000	-0.0030	-0.3170
9	8.12520E+01	3.79560E-13	1.78630E+03	6.26930E-03	0.0000	0.2470	24.6960
10	6.89990E+00	0.00000E+00	1.85470E+03	8.24110E-03	0.0000	0.0100	1.0120
11	4.08210E+01	3.39190E-15	1.82800E+03	7.32260E-03	0.0000	0.1200	12.0450
12	6.13340E+00	0.00000E+00	1.85330E+03	8.16400E-03	0.0000	0.0190	1.9380
13	1.63480E+01	2.33890E-18	1.85720E+03	8.41590E-03	0.0000	-0.0110	-1.0870
14	7.90400E+00	0.00000E+00	1.85620E+03	8.24660E-03	0.0000	0.0090	0.9470
15	2.08790E+01	1.04590E-17	1.86590E+03	8.59610E-03	0.0000	-0.0330	-3.2510
16	1.80440E+00	0.00000E+00	1.85900E+03	8.34440E-03	0.0000	-0.0020	-0.2280
17	4.94470E+00	0.00000E+00	1.85830E+03	8.28750E-03	0.0000	0.0050	0.4550
18	2.43450E+01	3.85830E-17	1.87590E+03	8.94480E-03	0.0000	-0.0740	-7.4400
19	7.59320E-02	0.00000E+00	1.85850E+03	8.32510E-03	0.0000	0.0000	0.0040
20	4.04580E+01	2.39600E-15	1.86270E+03	8.58850E-03	0.0000	-0.0320	-3.1600
21	5.04140E+02	3.06730E-07	1.47810E+03	1.66640E-03	0.0040	0.8000	79.9840
22	9.28960E+01	7.86390E-13	1.86940E+03	7.69580E-03	0.0000	0.0760	7.5630
23	2.66490E+02	1.83270E-09	1.69560E+03	4.27060E-03	0.0000	0.4870	48.7040
24	5.91270E+02	1.19180E-06	1.80370E+03	6.49570E-03	0.0140	0.2200	21.9770

Legend:

- Contribution_% = 100 * Damage_Only_i / Reference_Damage
- Importance = (Ref_Damage - Damage_Without_i) / Ref_Damage
- Delta_Damage_% = 100 * (Ref_Damage - Damage_Without_i) / Ref_Damage
  - (+) = damage decreases without mode-i
  - (-) = damage increases without mode-i

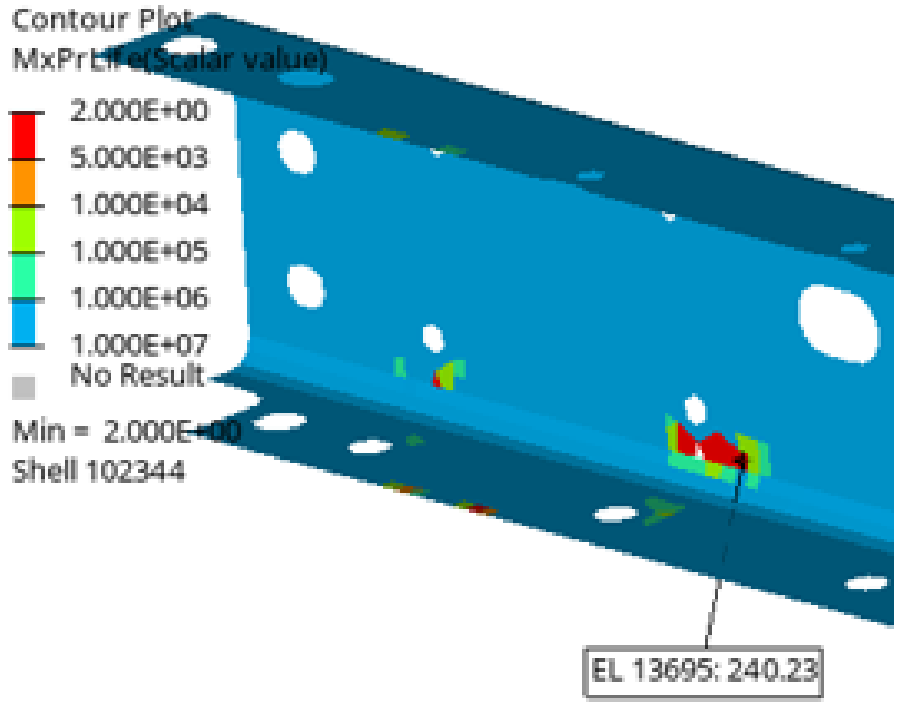


Dynamic Amplification Factors of Modes

Mode 7 shows the highest dynamic amplification (61.4%) at 10% damping — the most dynamically dominant mode.

Dynamic amplification table (FatDyn-X output)

mode_No	modalCoord_dynRange	modalCoord_staRange	DynRange %Diff
7	27.514	17.043	61.44
8	5.368	4.285	25.29
9	1.244	1.162	7.10
10	0.466	0.398	17.21
11	0.974	0.921	5.76
12	0.154	0.152	1.90
13	0.523	0.463	12.92
14	0.442	0.425	4.01
15	0.548	0.531	3.12
16	0.174	0.170	2.72
17	0.441	0.410	7.44
18	0.533	0.521	2.32
19	0.062	0.060	4.10
20	0.444	0.437	1.67
21	0.668	0.668	0.00
22	0.312	0.312	0.00
23	0.368	0.368	0.00
24	0.275	0.275	0.00

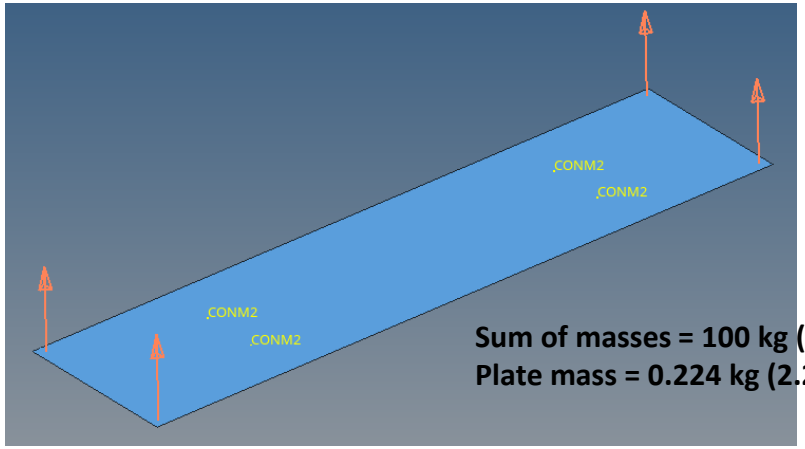


SECTION 6

Dynamic Analysis — Validation

Matches commercial FEA on displacement, acceleration, force, and stress — with extras (rigid-body decomposition; native node-pair relative motion / strain).

Ex-11 — Uniform Constant Load (FatDyn-X vs Benchmark)

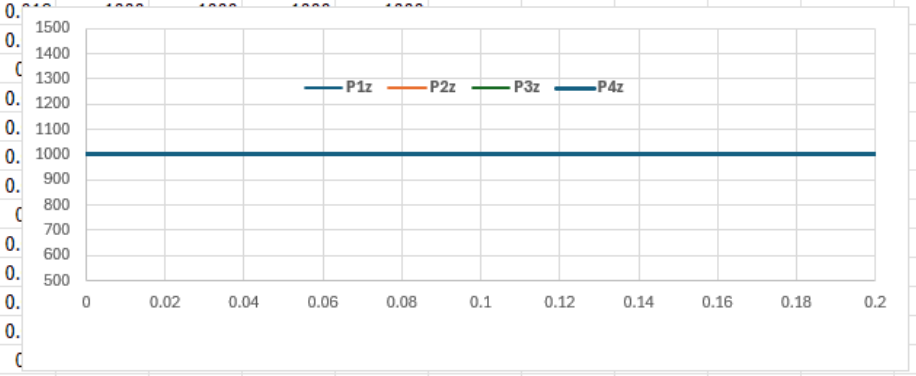


Sum of masses = 100 kg (0.1 ton)
Plate mass = 0.224 kg (2.24e-4 ton)

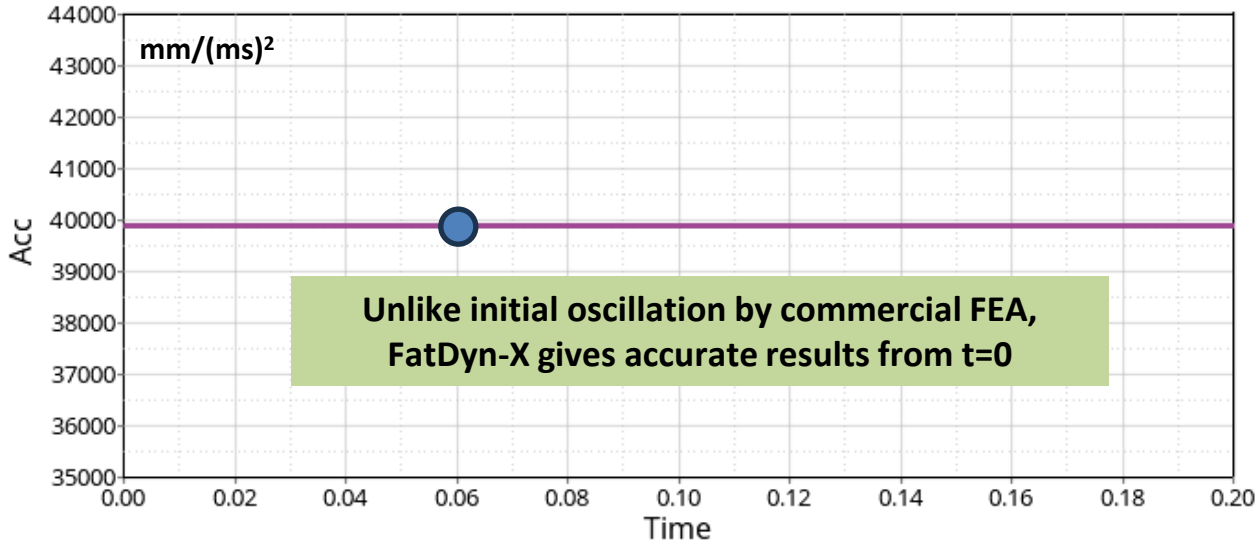
time	P1z	P2z	P3z	P4z
0	1000	1000	1000	1000
0.002	1000	1000	1000	1000
0.004	1000	1000	1000	1000
0.006	1000	1000	1000	1000
0.008	1000	1000	1000	1000
0.01	1000	1000	1000	1000
0.012	1000	1000	1000	1000
0.014	1000	1000	1000	1000
0.042	1000	1000	1000	1000
0.044	1000	1000	1000	1000

Does not exist in real life
(but automotive loads starts from 1G)

Constant Loading

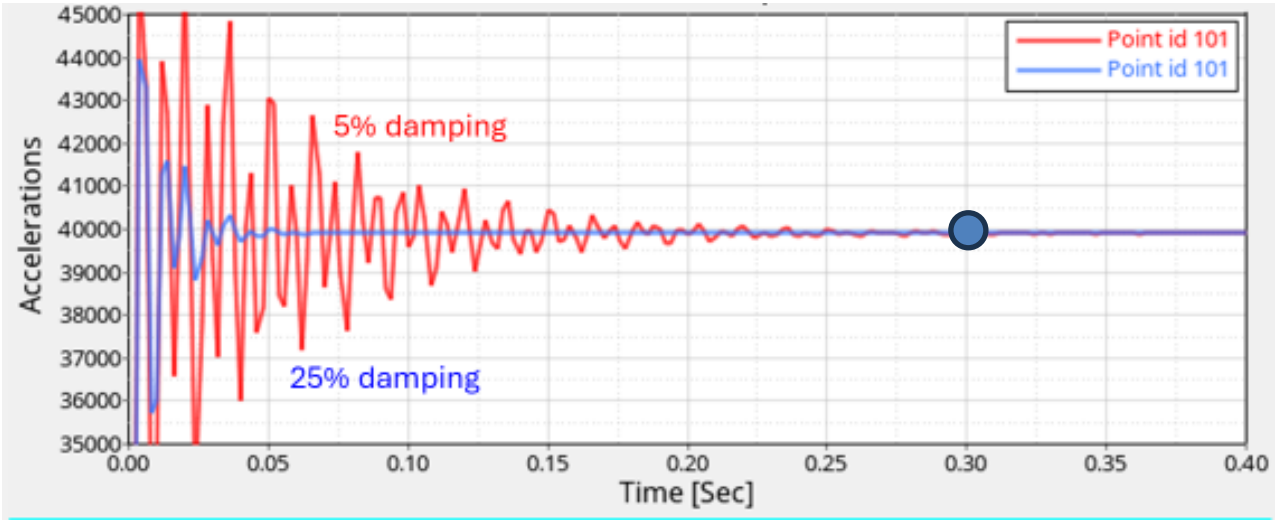


Rigid-Body Acc using Modal-Transient (FatDyn)



Unlike initial oscillation by commercial FEA,
FatDyn-X gives accurate results from t=0

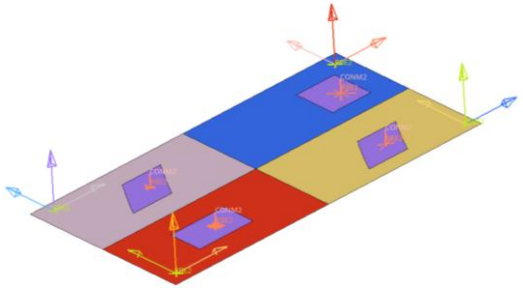
Acceleration calculation by commercial FEA

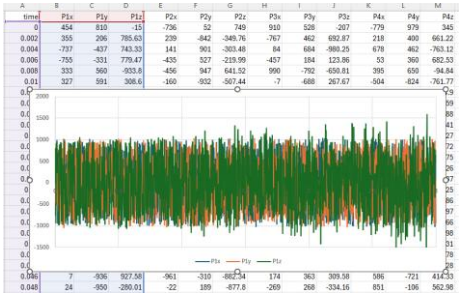


Ex-12 — Modal Transient: DISP, ACC, Force, Stress (Setup)

Time history of a mass-loaded plate — FatDyn-X vs commercial FEA.

- Setup: plate with attached masses; CELAS / CBUSH / CBAR / CBEAM joints.
- Compares FatDyn-X to commercial FEA on DISP / ACC / element forces / stresses.
- FatDyn-X generates time histories directly from the normal-modes solution — no heavy pre/post-processing.





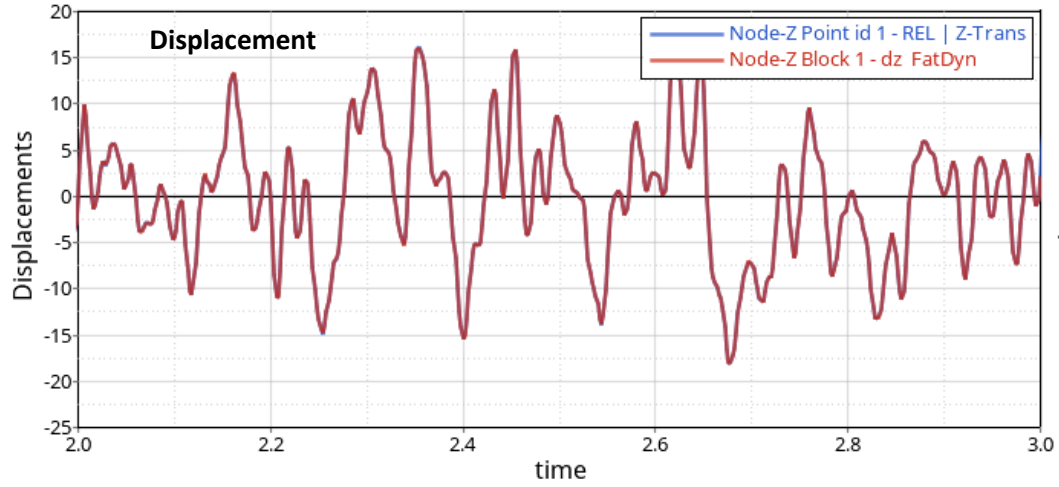
```
*TIMEHISTORY_LOADCASES
    !Load Case 1
    LABEL                : mt
    ANALYSIS_METHOD      : modal_transient

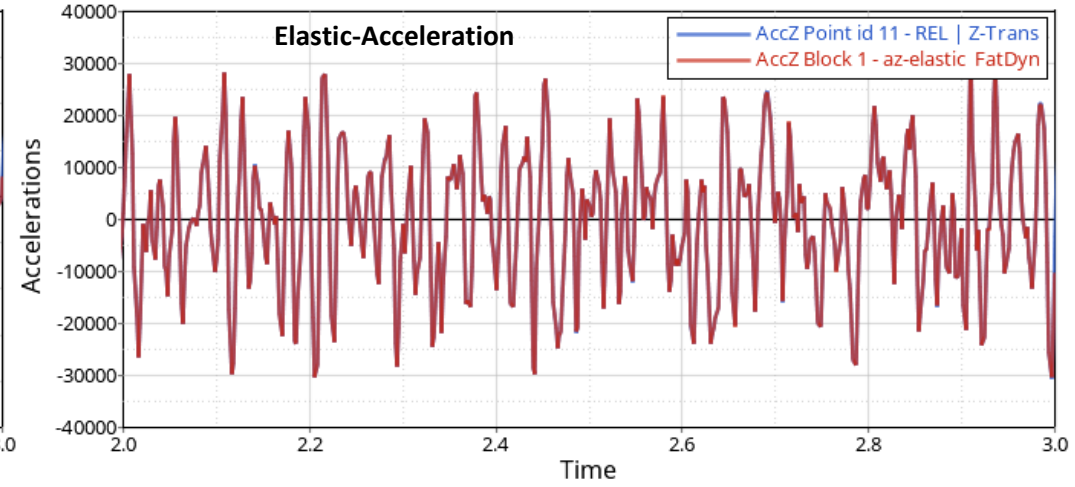
                         ! YES/No  Set-Label
    node_timeHistory     : YES    nodeResp
    nodePair_timeHistory: YES    nodePairs
    Force_timeHistory    : YES    forceElems
    Stress_timeHistory   : YES    stressElems

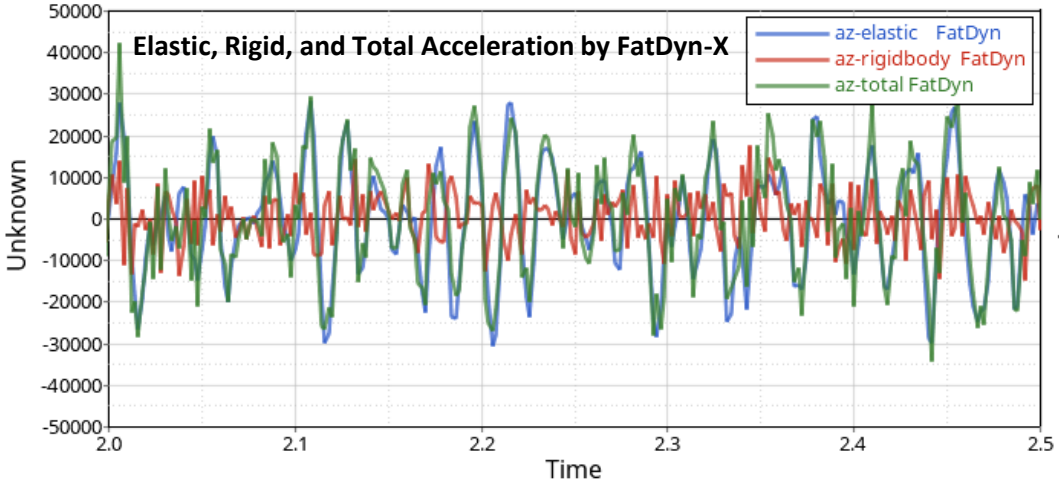
    ANALYSIS_TIME_STEP   :  0.002
$ SerialNo   Label          loadChannel_fileName    noHeadLines   COL_id   divider   scaleFactor   offset        node_id     dir   IRstat_LC
    1,       P1x,     LargePlateLoads_12LCs.csv,         1,         2,     5.00,       1.00,       0.0,          1,        1        1
    2,       P1y,     LargePlateLoads_12LCs.csv,         1,         3,     5.00,       1.00,       0.0,          1,        2        2
    3,       P1z,     LargePlateLoads_12LCs.csv,         1,         4,    10.00,       1.00,       0.0,          1,        3        3
!
    4,       P2x,     LargePlateLoads_12LCs.csv,         1,         5,     5.00,       1.00,       0.0,          2,        1        4
    5,       P2y,     LargePlateLoads_12LCs.csv,         1,         6,     5.00,       1.00,       0.0,          2,        2        5
    6,       P2z,     LargePlateLoads_12LCs.csv,         1,         7,    10.00,       1.00,       0.0,          2,        3        6
!
    7,       P3x,     LargePlateLoads_12LCs.csv,         1,         8,     5.00,       1.00,       0.0,          3,        1        7
    8,       P3y,     LargePlateLoads_12LCs.csv,         1,         9,     5.00,       1.00,       0.0,          3,        2        8
    9,       P3z,     LargePlateLoads_12LCs.csv,         1,        10,    10.00,       1.00,       0.0,          3,        3        9
!
   10,       P4x,     LargePlateLoads_12LCs.csv,         1,        11,     5.00,       1.00,       0.0,          4,        1       10
   11,       P4y,     LargePlateLoads_12LCs.csv,         1,        12,     5.00,       1.00,       0.0,          4,        2       11
   12,       P4z,     LargePlateLoads_12LCs.csv,         1,        13,    10.00,       1.00,       0.0,          4,        3       12
!
```

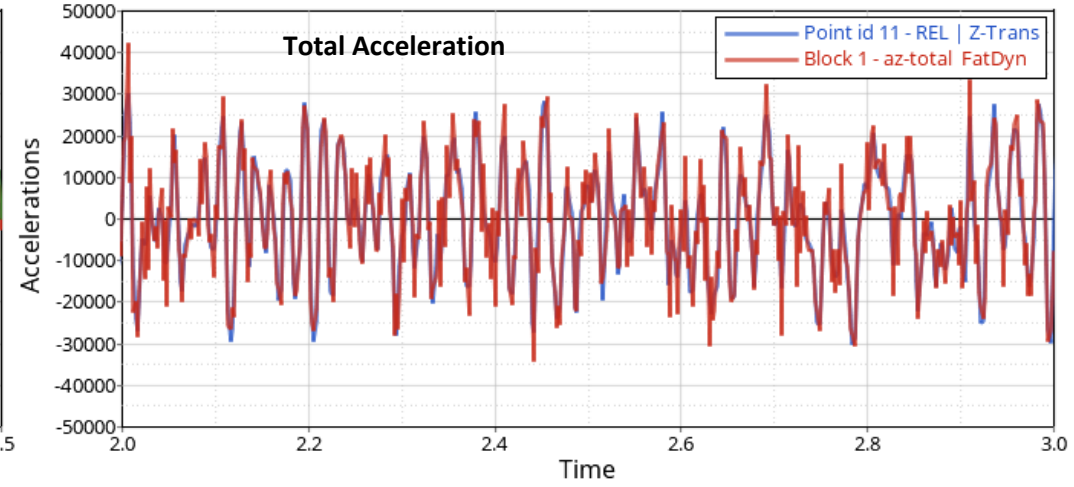
- Specify analysis method
- Calculation requests

Ex-12 Results — Displacement & Acceleration



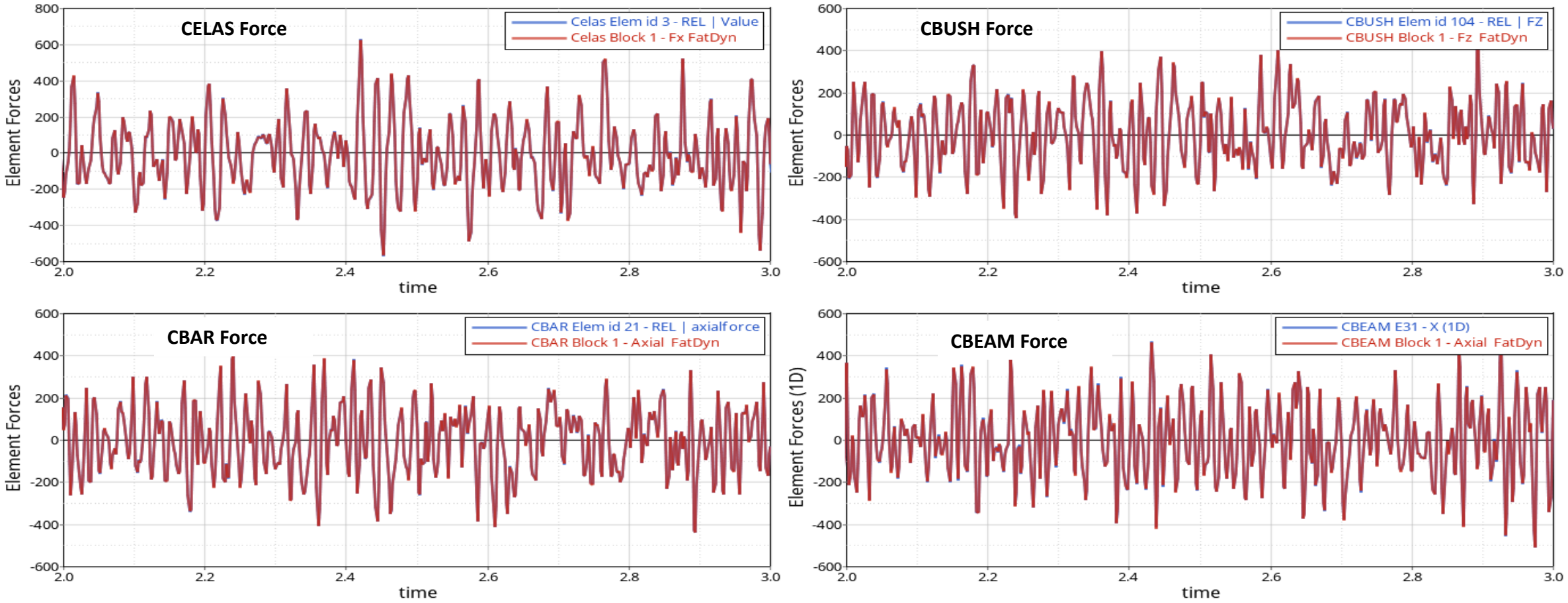






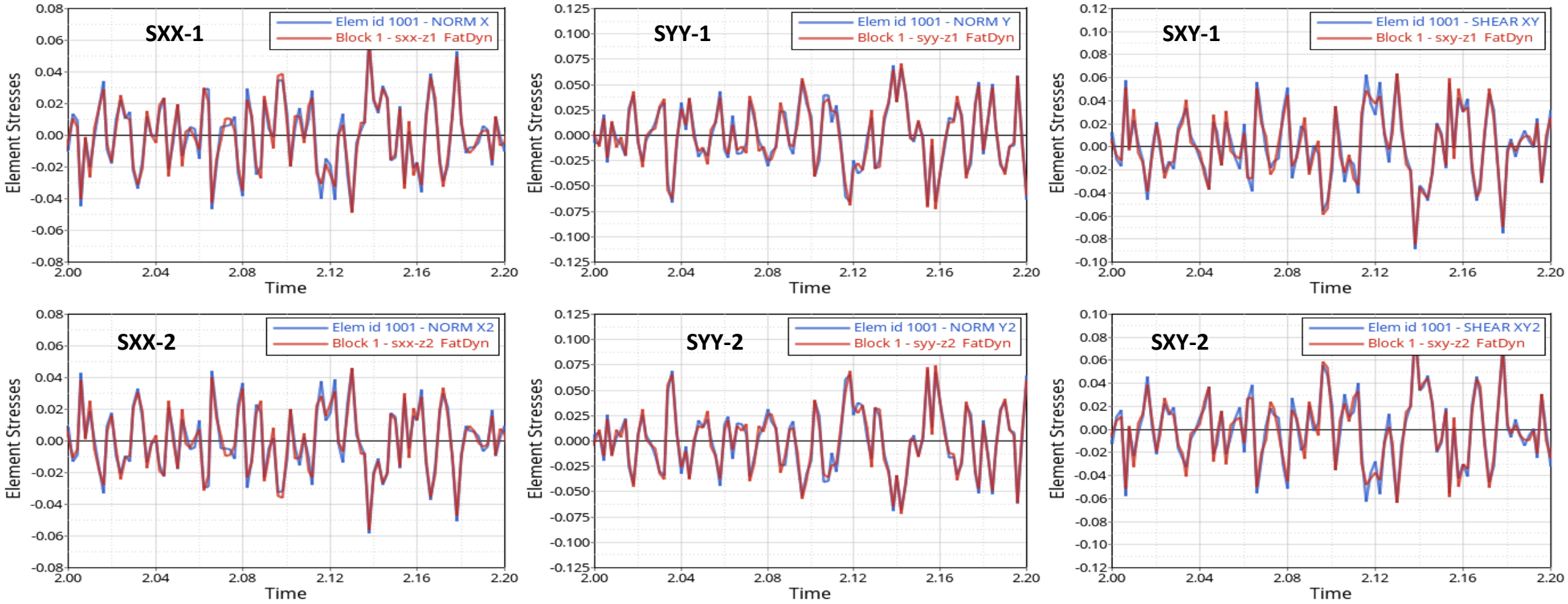
FatDyn-X reproduces displacement and acceleration from commercial FEA and additionally decomposes total acceleration into elastic and rigid-body components — deeper insight without extra post-processing.

Ex-12 Results — Element Forces (CELAS, CBUSH, CBAR, CBEAM)



Element forces match commercial FEA across all element types — close agreement confirms reliable transient force capture.

Ex-12 Results — Shell-Element Stress



SXX, SYY, SXY at top and bottom surfaces — excellent agreement across all in-plane normal and shear components.

SECTION 7

Dynamic Analysis — Advanced Features

Contact & Interference Detection — long-duration durability loads

- Identifies locations of potential contacts and interferences.
- Reports the time instances at which they occur — feed those time sets into HyperView for transient animation.

Why this is hard with traditional methods

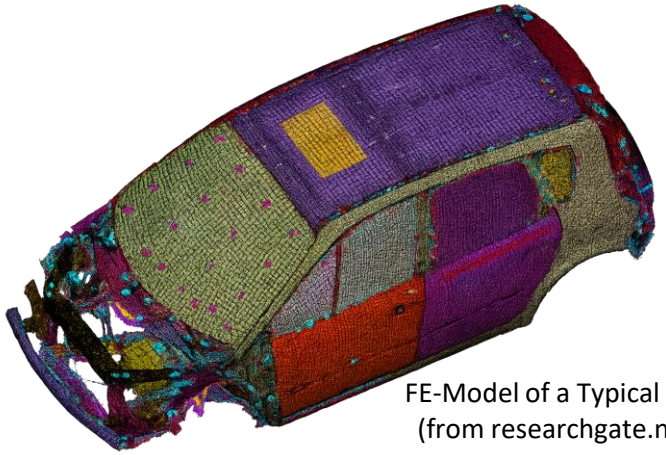
NVH analyses are designed for small-displacement wave propagation — not well suited to detecting contact or interference under real-world durability loading.

Fully nonlinear transient analyses are generally impractical for long time-histories due to excessive computational cost.

How FatDyn-X solves it

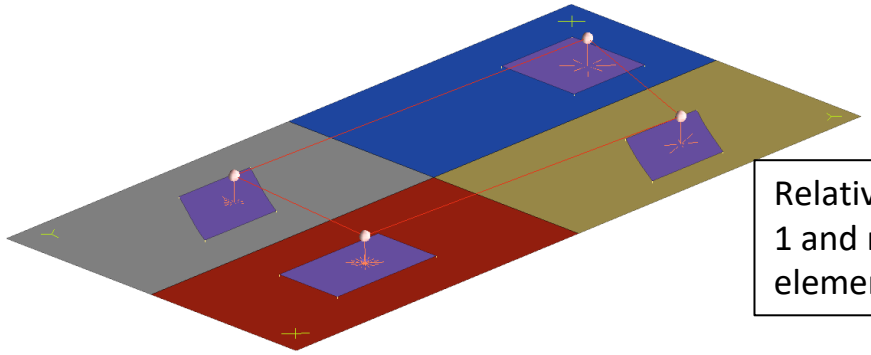
Recently used to detect contact and interference between SUV components under **>2000 s of loading and >1.1M time steps** using a modal-transient approach driven only by Nastran normal modes.

Outputs are written in CSV and HWAscii (HyperView) for efficient post-processing and visualization.

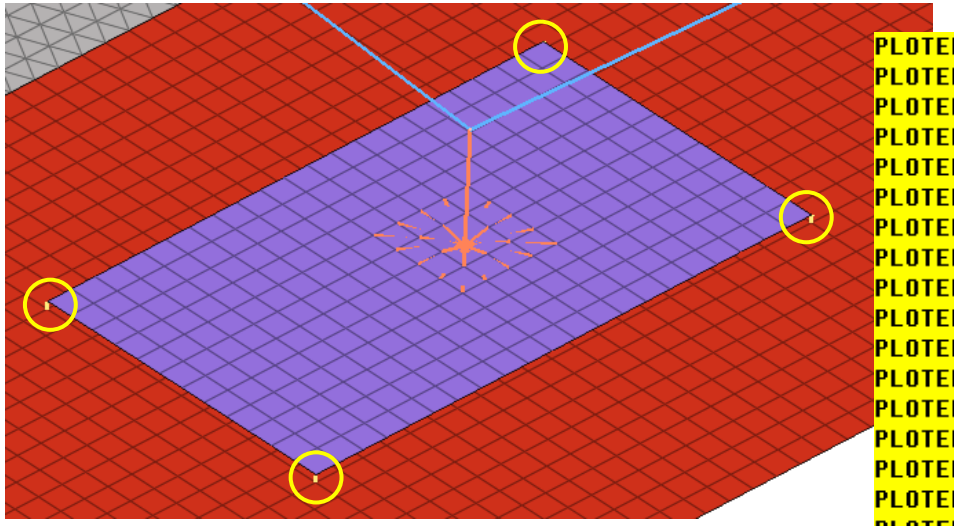


FE-Model of a Typical SUV
(from researchgate.net)

Ex-13 — Relative-Motion Analysis of Plate-Structure with Masses



Relative motion analysis between node-1 and node-2 identified by PLOTEL elements defined in element set



Potential contact locations are modeled using PLOTEL elements

		Node-1	Node-2
PLOTEL	13358	1	2
PLOTEL	13359	2	3
PLOTEL	13360	3	4
PLOTEL	13361	4	1
PLOTEL	16393	11221	9488
PLOTEL	16394	11727	9481
PLOTEL	16395	10236	9502
PLOTEL	16396	10642	9497
PLOTEL	16397	21862	9402
PLOTEL	16398	21863	9407
PLOTEL	16399	21861	9401
PLOTEL	16400	21860	9396
PLOTEL	16401	22346	9432
PLOTEL	16402	22347	9437
PLOTEL	16403	22351	9453
PLOTEL	16404	22355	9467
PLOTEL	16405	13540	9517
PLOTEL	16406	13037	9532
PLOTEL	16407	13229	9523
PLOTEL	16408	13753	9508

Set: plotels

Input for FatDyn-X

```
*Element_sets
 Label: plotels
  13358 to 13361
  16393 to 16408

$============================================================
*FE_DATA
        file_Name_FE_NasModel : LargePlate_modes.fem
        file_Name_FE_PCH103   : LargePlate_modes.pch

          Rigid_Modes :  1 to  6
          Modes       :  7 to 50   critical_damping = 0.05
          Static_modes: 51 to 62

$------------------------------------------------------------
*TIMEHISTORY_LOADCASES
     LABEL            : mt
     ANALYSIS_METHOD : modal_transient
     nodePair_Plotels_timeHistory: YES plotels

ANALYSIS_TIME_STEP   :  0.002
$ SerialNo    Label          loadChannel_fileName     noHeadLines
      1,        P1x,      LargePlateLoads_12LCs.csv,          1,
      2,        P1y,      LargePlateLoads_12LCs.csv,          1,
      3,        P1z,      LargePlateLoads_12LCs.csv,          1,
!
      4,        P2x,      LargePlateLoads_12LCs.csv,          1,
      5,        P2y,      LargePlateLoads_12LCs.csv,          1,
      6,        P2z,      LargePlateLoads_12LCs.csv,          1,
!
      7,        P3x,      LargePlateLoads_12LCs.csv,          1,
      8,        P3y,      LargePlateLoads_12LCs.csv,          1,
      9,        P3z,      LargePlateLoads_12LCs.csv,          1,
!
     10,        P4x,      LargePlateLoads_12LCs.csv,          1,
     11,        P4y,      LargePlateLoads_12LCs.csv,          1,
     12,        P4z,      LargePlateLoads_12LCs.csv,          1,


*END
```

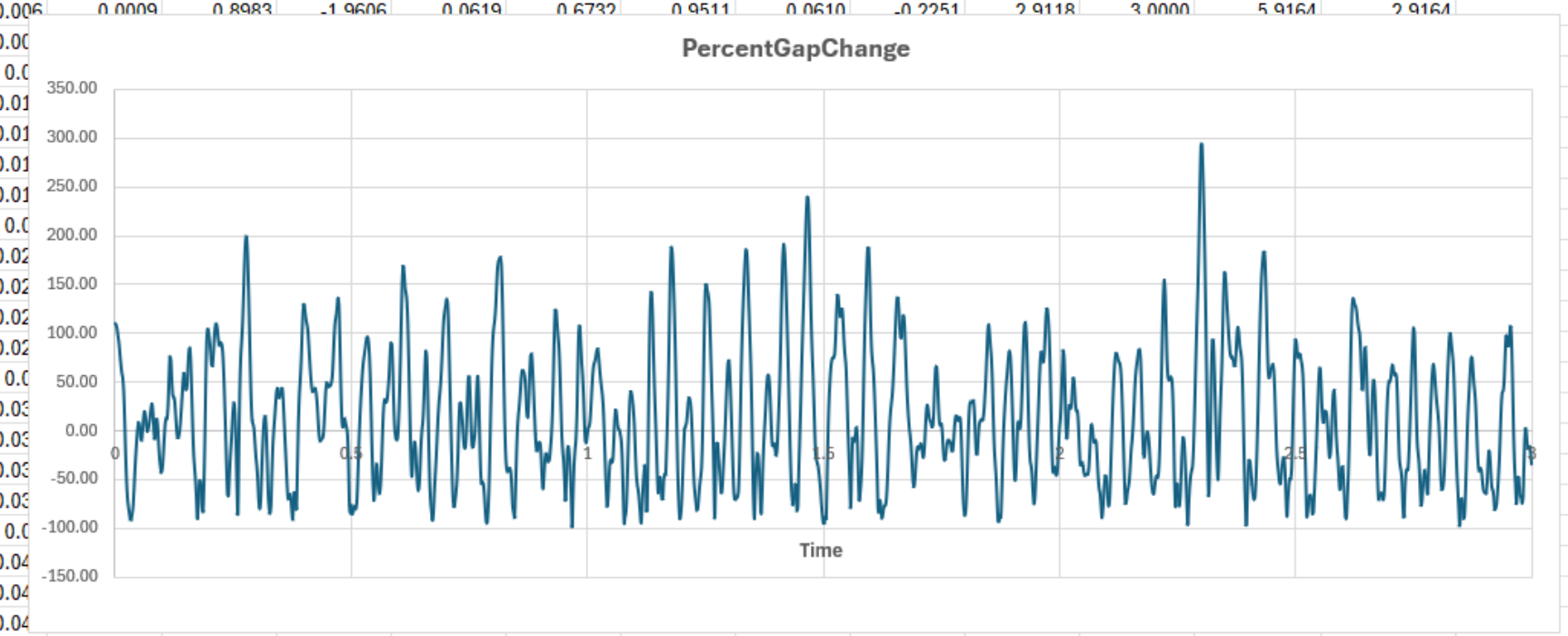
Set-id-label

Relative-Motion Analysis — Time-History Results

Relative-motion time history by PLOTEL element ID. DISP for Node1 and Node2; REL-DISP = Node2 − Node1.

- TH1_MT_nodePairPlotel_13358.csv
- TH1_MT_nodePairPlotel_13359.csv
- TH1_MT_nodePairPlotel_13360.csv
- TH1_MT_nodePairPlotel_13361.csv
- TH1_MT_nodePairPlotel_16393.csv
- TH1_MT_nodePairPlotel_16394.csv
- TH1_MT_nodePairPlotel_16395.csv
- TH1_MT_nodePairPlotel_16396.csv
- TH1_MT_nodePairPlotel_16397.csv
- TH1_MT_nodePairPlotel_16398.csv
- TH1_MT_nodePairPlotel_16399.csv
- TH1_MT_nodePairPlotel_16400.csv
- TH1_MT_nodePairPlotel_16401.csv
- TH1_MT_nodePairPlotel_16402.csv
- TH1_MT_nodePairPlotel_16403.csv
- TH1_MT_nodePairPlotel_16404.csv
- TH1_MT_nodePairPlotel_16405.csv
- TH1_MT_nodePairPlotel_16406.csv
- TH1_MT_nodePairPlotel_16407.csv
- TH1_MT_nodePairPlotel_16408.csv

	DISP: Node1			DISP: Node2			REL-DISP: Node2-Node1						
time	u1	v1	w1	u2	v2	w2	du	dv	dw	Dis_ini	Dis_current	Deformation	PercentGapChange
0	-0.0532	1.2731	-1.8241	-0.0916	1.1690	1.4816	-0.0384	-0.1041	3.3056	3.0000	6.3066	3.3066	110.22
0.002	-0.0400	1.2222	-1.8306	-0.0654	1.1023	1.4269	-0.0255	-0.1199	3.2576	3.0000	6.2588	3.2588	108.63
0.004	-0.0102	1.0993	-1.8645	-0.0013	0.9330	1.2741	0.0090	-0.1662	3.1386	3.0000	6.1409	3.1409	104.70
0.006	0.0009	0.8983	-1.9606	0.0619	0.6732	0.9511	0.0610	-0.2251	2.9118	3.0000	5.9164	2.9164	97.21
0.00													90.22
0.0													79.09
0.01													68.40
0.01													59.65
0.01													54.63
0.01													40.76
0.0													12.78
0.02													-21.85
0.02													-49.69
0.02													-67.70
0.02													-77.36
0.0													-84.28
0.03													-90.24
0.03													-90.54
0.03													-84.46
0.03													-76.52
0.0													-58.90
0.04													-38.60
0.04													-21.62
0.04													-7.32
0.048	-0.2511	0.7136	1.6246	0.1138	0.2254	1.6915	0.3649	-0.4882	0.0669	3.0000	3.1269	0.1269	4.23
0.05	-0.1305	0.4007	1.6919	0.5035	-0.4788	1.7653	0.6340	-0.8795	0.0735	3.0000	3.2592	0.2592	8.64



`TH1_MT_NodePairs_summary.csv`
`TH1_MT_NodePairs_summary.HWascii` *(for HyperView)*
`TH1_MT_mDynEffect.csv` *shows the importance of modal resonance*

Relative-Motion Analysis — Summaries (cont.)

TH1_MT_NodePairs_summary.csv

Plotel_ID	node1	node2	relDisX: max	maxTime	min	minTime	range	average	RMS	relDisY: max	maxTime	min	minTime	range	average	RMS	relDisZ: max	maxTime	min	minTime	range	average	RMS	PercentGapChange: max	maxTime	min	minTime	range	average	RMS
16396	10642	9497	1.1216	0.9120	-1.1833	2.7100	2.3050	-0.0019	0.4155	1.8651	0.6320	-1.6556	0.9120	3.5207	0.0005	0.6282	8.7224	2.3020	-8.4854	2.3240	17.2080	0.0240	2.4763	293.5100	2.3020	-98.2500	0.9680	391.7600	13.9460	69.3860
16395	10236	9502	1.1445	0.6320	-1.1734	0.6200	2.3179	-0.0019	0.4030	1.8650	0.6320	-1.6555	0.9120	3.5206	0.0005	0.6282	13.6870	1.2600	-11.8830	1.2820	25.5700	0.0867	3.3453	456.3100	1.2600	-97.9630	0.1860	554.2800	27.1650	90.5450
16394	11727	9481	1.1216	0.9120	-1.1834	2.7100	2.3050	-0.0019	0.4155	2.3325	0.6200	-2.0626	0.6080	4.3950	0.0006	0.7996	14.2890	1.2820	-18.5580	1.2600	32.8470	-0.1042	4.3547	476.4700	1.2820	-97.7010	1.6140	574.1700	45.0190	110.7900
16397	21862	9402	1.4258	1.1500	-1.3811	1.1400	2.8069	-0.0003	0.3778	2.5623	1.1400	-2.7012	1.1500	5.2635	-0.0011	0.7087	11.9870	0.2480	-13.1240	1.4660	25.1110	-0.0065	4.2840	399.8000	0.2480	-96.8300	1.9220	496.6300	44.8660	109.9600
16400	21860	9396	1.4358	1.1600	-1.4165	1.1500	2.8523	-0.0019	0.3960	0.8236	1.1400	-0.9064	1.1500	1.7299	0.0000	0.2341	7.5198	2.6900	-7.2620	1.2580	14.7820	-0.0522	2.0440	251.8900	2.6920	-96.8000	2.5060	348.6900	5.3376	58.8360
16393	11221	9488	1.1440	0.6320	-1.1743	0.6200	2.3184	-0.0019	0.4031	2.3325	0.6200	-2.0625	0.6080	4.3950	0.0006	0.7996	9.7259	0.2500	-11.0690	2.3020	20.7940	-0.0310	3.5970	324.3000	0.2500	-96.6330	2.7940	420.9400	32.0780	92.9770
16398	21863	9407	1.4408	1.1500	-1.3956	1.1400	2.8365	-0.0003	0.3817	0.5422	1.3200	-0.4931	2.6920	1.0353	0.0007	0.1410	6.2076	1.2600	-6.4066	2.6920	12.6140	0.0438	1.7440	207.2600	1.2600	-96.1250	2.7740	303.3900	5.2956	52.8840
16401	22346	9432	0.1428	2.6900	-0.1502	2.3660	0.2930	-0.0019	0.0504	0.1027	2.8220	-0.1048	2.6060	0.2074	0.0014	0.0374	3.3173	2.6920	-2.9036	2.4640	6.2210	-0.0054	0.8803	110.6300	2.6920	-96.0730	2.4640	206.7000	-0.1517	29.3270
16399	21861	9401	1.4250	1.1600	-1.4043	1.1500	2.8294	-0.0019	0.3928	2.2018	2.3200	-2.1389	1.1400	4.3407	0.0017	0.6070	11.0610	1.4660	-10.1830	0.6340	21.2430	0.0251	3.7058	369.4100	1.4660	-94.0140	1.7520	463.4200	33.6990	96.2650
16405	13540	9517	0.1367	1.2980	-0.1496	2.3000	0.2863	-0.0008	0.0538	0.1049	2.8220	-0.1308	2.6220	0.2357	0.0014	0.0382	1.8164	0.4580	-2.3056	2.9340	4.1220	-0.0213	0.6148	60.5470	0.4580	-76.7820	2.9340	137.3300	-0.6858	20.4840
16404	22355	9467	0.1421	2.6900	-0.1497	2.3660	0.2918	-0.0019	0.0504	0.1027	2.8220	-0.1047	2.6060	0.2074	0.0014	0.0374	2.3076	2.3960	-2.1824	2.1760	4.4900	0.0019	0.7486	76.9220	2.3960	-72.6810	2.1760	149.6000	0.0880	24.9440
16408	13753	9508	0.1370	1.2980	-0.1497	2.3000	0.2866	-0.0008	0.0537	0.1052	2.8220	-0.1308	2.6220	0.2361	0.0014	0.0382	2.1671	2.5300	-2.1318	2.6180	4.2989	0.0024	0.5870	72.2420	2.5300	-70.8390	2.6180	143.0800	0.1048	19.5600
16406	13037	9532	0.1365	1.2980	-0.1498	2.3000	0.2863	-0.0008	0.0538	0.1049	2.8220	-0.1308	2.6220	0.2357	0.0014	0.0382	1.9685	2.2320	-1.5260	2.0300	3.4945	-0.0022	0.5419	65.6590	2.2320	-50.6410	2.0300	116.3000	-0.0476	18.0570
16407	13229	9523	0.1364	1.2980	-0.1497	2.3000	0.2862	-0.0008	0.0537	0.1049	2.8220	-0.1311	2.6220	0.2360	0.0014	0.0382	1.2356	0.0700	-1.1874	2.7020	2.4230	-0.0075	0.3708	41.2420	0.0700	-39.4950	2.7020	80.7370	-0.2267	12.3550
16403	22351	9453	0.1420	2.6900	-0.1500	2.3660	0.2921	-0.0019	0.0504	0.1021	2.8220	-0.1041	2.6060	0.2062	0.0014	0.0373	1.2054	0.4900	-1.0887	0.7180	2.2942	-0.0114	0.3341	40.2210	0.4900	-36.2710	0.7180	76.4930	-0.3582	11.1330
16402	22347	9437	0.1408	2.6900	-0.1494	2.3660	0.2902	-0.0019	0.0503	0.1018	2.8220	-0.1036	2.6060	0.2054	0.0014	0.0372	0.4220	2.9240	-0.3771	2.4540	0.7991	-0.0026	0.1254	14.0710	2.9240	-12.5370	2.4540	26.6070	-0.0646	4.1794
13361	4	1	0.0735	2.6940	-0.0612	0.6340	0.1348	0.0000	0.0215	6.0885	2.6940	-4.9638	2.5920	11.0520	0.0060	1.7663	25.5700	1.1100	-28.3630	2.7060	53.9330	-0.4614	9.0505	0.0316	2.7060	-0.0007	2.4220	0.0323	0.0034	0.0056
13360	3	4	2.3124	2.5920	-2.8353	2.6940	5.1477	-0.0028	0.8225	0.0631	1.4920	-0.0785	2.6940	0.1416	-0.0001	0.0225	25.6800	2.7040	-22.1450	2.7700	47.8240	0.4121	7.8586	0.1201	2.7040	-0.0006	2.1300	0.1207	0.0114	0.0189
13359	2	3	0.0334	2.5900	-0.0441	2.6920	0.0775	0.0000	0.0118	4.8449	2.5920	-5.9387	2.6940	10.7840	-0.0059	1.7235	29.2690	2.6200	-26.5590	0.1280	55.8290	-0.1309	9.0310	0.0356	2.6200	-0.0006	1.7860	0.0361	0.0035	0.0061
13358	1	2	2.7989	2.6920	-2.2850	2.5920	5.0838	0.0028	0.8127	0.0581	1.4900	-0.0714	2.6940	0.1295	-0.0001	0.0204	28.1390	0.6960	-35.2730	2.6200	63.4120	0.1802	9.3780	0.2320	2.6200	-0.0005	2.2200	0.2325	0.0166	0.0292

TH1_MT_NodePairs_summary.HWascii (for HyperView)

```
ALTAIR ASCII FILE
$TITLE    = Load Case
$SUBCASE_ID = 1  subcase1
$BINDING = ELEMENT
$COLUMN_INFO = ENTITY_ID
$RESULT_TYPE=relDisX-max(s),relDisX_min(s),relDisX_range(s),
     13358,  0.27989E+01, -0.22850E+01,  0.50838E+01,  0.5809
     13359,  0.33420E-01, -0.44110E-01,  0.77530E-01,  0.4844
     13360,  0.23124E+01, -0.28353E+01,  0.51477E+01,  0.6311
     13361,  0.73547E-01, -0.61200E-01,  0.13475E+00,  0.6088
     16393,  0.11440E+01, -0.11743E+01,  0.23184E+01,  0.2332
     16394,  0.11216E+01, -0.11834E+01,  0.23050E+01,  0.2332
     16395,  0.11445E+01, -0.11734E+01,  0.23179E+01,  0.1865
     16396,  0.11216E+01, -0.11833E+01,  0.23050E+01,  0.1865
     16397,  0.14258E+01, -0.13811E+01,  0.28069E+01,  0.2562
     16398,  0.14408E+01, -0.13956E+01,  0.28365E+01,  0.5422
     16399,  0.14250E+01, -0.14043E+01,  0.28294E+01,  0.2201
     16400,  0.14358E+01, -0.14165E+01,  0.28523E+01,  0.8235
     16401,  0.14275E+00, -0.15022E+00,  0.29297E+00,  0.1026
     16402,  0.14084E+00, -0.14937E+00,  0.29021E+00,  0.1018
     16403,  0.14204E+00, -0.15001E+00,  0.29205E+00,  0.1021
     16404,  0.14206E+00, -0.14971E+00,  0.29177E+00,  0.1027
     16405,  0.13670E+00, -0.14963E+00,  0.28633E+00,  0.1048
     16406,  0.13650E+00, -0.14984E+00,  0.28633E+00,  0.1048
     16407,  0.13643E+00, -0.14974E+00,  0.28617E+00,  0.1048
     16408,  0.13697E+00, -0.14965E+00,  0.28662E+00,  0.1052
```

HWascii file of max, min, & range for relative DISP-X, Y, & Z, and PercentGapChange

Transient animation models are built around times: 0.9680 and 0.1860 s

Re-arranged min to max

Times at most negative strain

PercentGapChange: max	maxTime	min	minTime	range	average	RMS
293.5100	2.3020	-98.2500	0.9680	391.7600	13.9460	69.3860
456.3100	1.2600	-97.9630	0.1860	554.2800	27.1650	90.5450
476.4700	1.2820	-97.7010	1.6140	574.1700	45.0190	110.7900
399.8000	0.2480	-96.8300	1.9220	496.6300	44.8660	109.9600
251.8900	2.6920	-96.8000	2.5060	348.6900	5.3376	58.8360
324.3000	0.2500	-96.6330	2.7940	420.9400	32.0780	92.9770
207.2600	1.2600	-96.1250	2.7740	303.3900	5.2956	52.8840
110.6300	2.6920	-96.0730	2.4640	206.7000	-0.1517	29.3270
369.4100	1.4660	-94.0140	1.7520	463.4200	33.6990	96.2650
60.5470	0.4580	-76.7820	2.9340	137.3300	-0.6858	20.4840
76.9220	2.3960	-72.6810	2.1760	149.6000	0.0880	24.9440
72.2420	2.5300	-70.8390	2.6180	143.0800	0.1048	19.5600
65.6590	2.2320	-50.6410	2.0300	116.3000	-0.0476	18.0570
41.2420	0.0700	-39.4950	2.7020	80.7370	-0.2267	12.3550
40.2210	0.4900	-36.2710	0.7180	76.4930	-0.3582	11.1330
14.0710	2.9240	-12.5370	2.4540	26.6070	-0.0646	4.1794
0.0316	2.7060	-0.0007	2.4220	0.0323	0.0034	0.0056
0.1201	2.7040	-0.0006	2.1300	0.1207	0.0114	0.0189
0.0356	2.6200	-0.0006	1.7860	0.0361	0.0035	0.0061
0.2320	2.6200	-0.0005	2.2200	0.2325	0.0166	0.0292

Building Transient Analysis Models for Animation

```
*Time_sets
 Label contChk
 0.9680, 0.1860
$=============================================
*FE_DATA
        file_Name_FE_NasModel : LargePlate_modes.fem
        file_Name_FE_PCH103   : LargePlate_modes.pch

          Rigid_Modes :  1 to  6
          Modes       :  7 to 50   critical_damping = 0.05
          Static_modes: 51 to 62

$.....................................................................................................
*TIMEHISTORY_LOADCASES
     LABEL           : mtAnim
   ! ANALYSIS_METHOD : modal_transient
   ! nodePair_Plotels_timeHistory: YES plotels

ANALYSIS_TIME_STEP  :  0.002
$ SerialNo   Label          loadChannel_fileName    noHeadLines   COL_id   divider   scaleFactor    offset        node_id     dir
     1,       P1x,     LargePlateLoads_12LCs.csv,          1,         2,     5.00,       1.00,       0.0,          1,        1
     2,       P1y,     LargePlateLoads_12LCs.csv,          1,         3,     5.00,       1.00,       0.0,          1,        2
     3,       P1z,     LargePlateLoads_12LCs.csv,          1,         4,    10.00,       1.00,       0.0,          1,        3
!
     4,       P2x,     LargePlateLoads_12LCs.csv,          1,         5,     5.00,       1.00,                     2,        1
     5,       P2y,     LargePlateLoads_12LCs.csv,          1,         6,     5.00,       1.00,                     2,        2
     6,       P2z,     LargePlateLoads_12LCs.csv,          1,         7,    10.00,                                 2,        3
!
     7,       P3x,     LargePlateLoads_12LCs.csv,          1,                                                    3,        1
     8,       P3y,     LargePlateLoads_12LCs.csv,          1,                                                    3,        2
     9,       P3z,     LargePlateLoads_12LCs.csv,          1,                                                    3,        3
!
    10,       P4x,     LargePlateLoads_12LCs.csv,          1,                    1.00,       0.0,                4,        1
    11,       P4y,     LargePlateLoads_12LCs.csv,          1,                    1.00,       0.0,                4,        2
    12,       P4z,     LargePlateLoads_12LCs.csv,          1,        13,    10.00,       1.00,       0.0,          4,        3

   Transient_loadCases_byTimeSet : YES  contChk 5000  100   ! YES/NO  setLabel     LC_startID   noPads

*END
```

This creates transient analysis models at times in time-set contChk

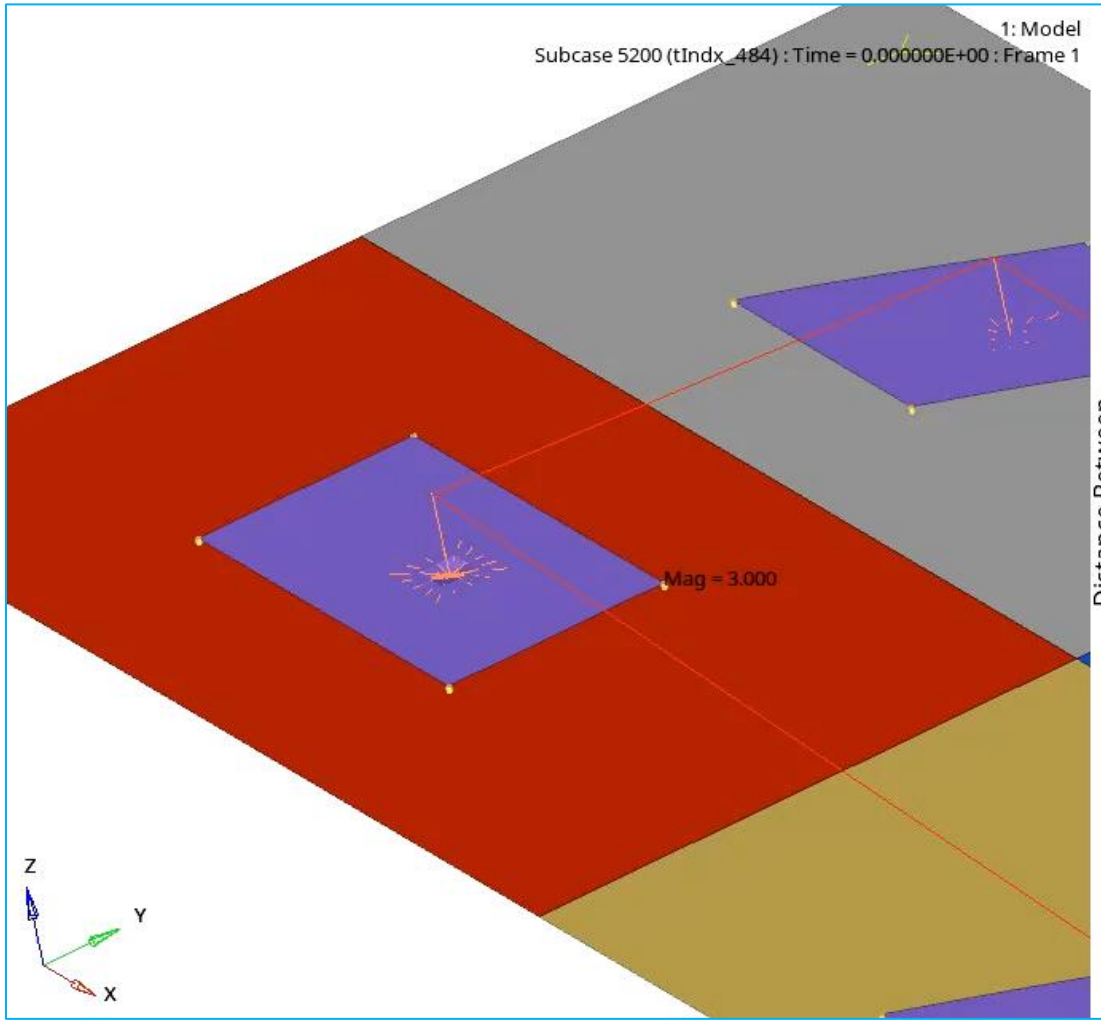
Powerful one-step generator: builds animation models around times of interest in bulk-data format — includes all transient cards (loads in TABLED1 format). The "noPads" parameter inserts time pads before and after each time-set entry.

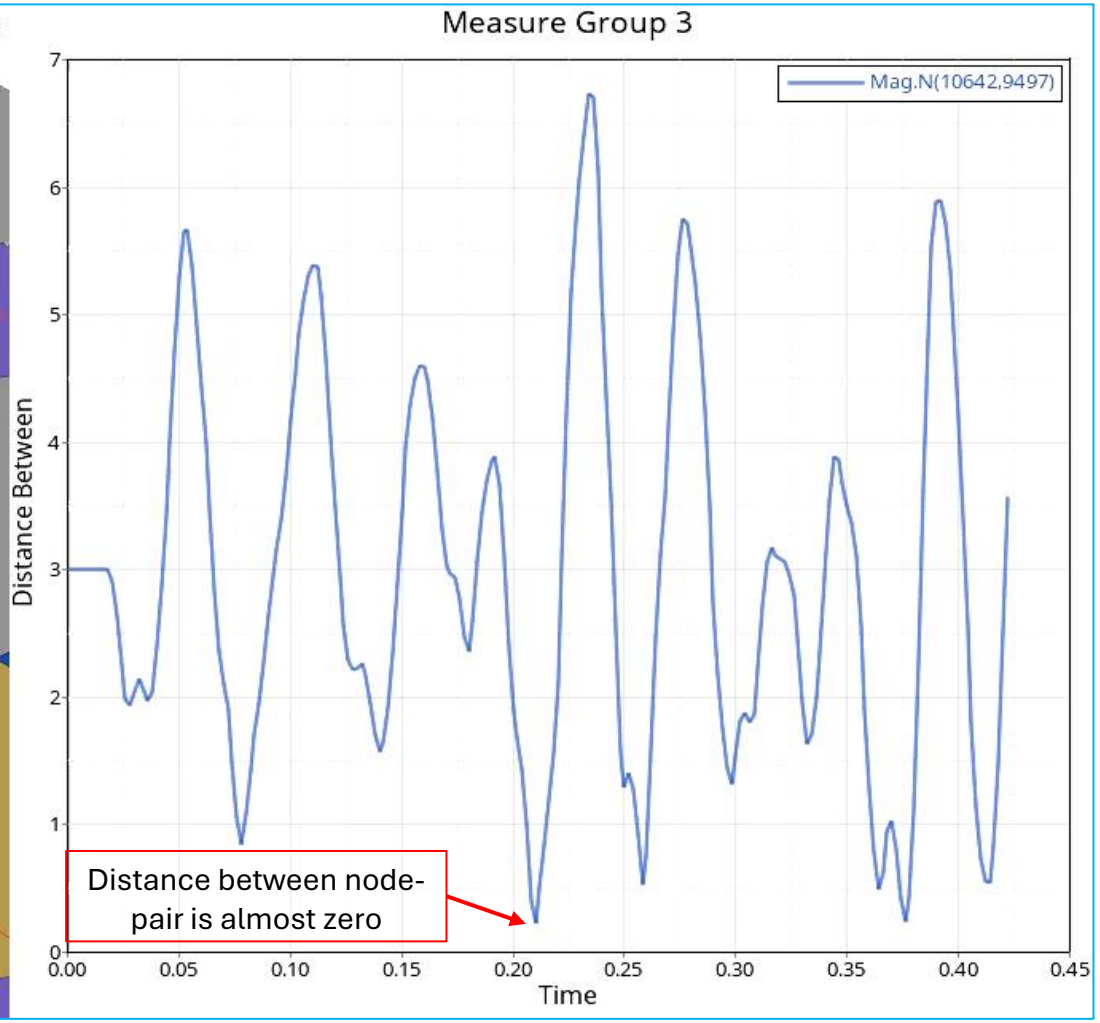
```
DISPLACEMENT(h3d) = all
$-------------------
SUBCASE     5100
  Label       tIndx_94
$ ANALYSIS DTRAN
  ANALYSIS MTRAN
  DLOAD =   5100
  TSTEP =   5100
$---> data for modal transient analysis
  Method  = 200
  SDAMPING =  10000
$-------------------
SUBCASE     5200
  Label      tIndx_484
$ ANALYSIS DTRAN
  ANALYSIS MTRAN
  DLOAD =   5200
  TSTEP =   5200
$---> data for modal transient analysis
  Method  = 200
  SDAMPING =  10000
$=================================================
BEGIN BULK
$ ------------------------------------------------------------------------------
$ Force-Moment time histories include scale-divide-offset transformations
$ Force-Moment = (ScaleFactor/Divider)(TimeHistoryValue - Offset)
$ Transient load cases with time padding and zero ramp-up for stability
$ ------------------------------------------------------------------------------
$ ========================================
$ cards needed if modal-transient is chosen
PARAM, RESVEC,  YES
EIGRL, 200,  ,  ,50
USET1          U6       1       1
USET1          U6       2       1
USET1          U6       3       1
USET1          U6       1       2
USET1          U6       2       2
USET1          U6       3       2
USET1          U6       1       3
USET1          U6       2       3
USET1          U6       3       3
USET1          U6       1       4
USET1          U6       2       4
USET1          U6       3       4
TABDMP1     1
+           2
```

```
$Load case, time:         1      0.1860
FORCE  ,    5101,       1, 0 ,      1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
TLOAD1,    5101,    5101, , ,    5101
TABLED1*             5101            LINEAR          LINEAR
*
*                0.00000     0.00000E+00         0.00200     0.00000E+00
*                0.00400     0.00000E+00         0.00600     0.00000E+00
*                0.00800     0.00000E+00         0.01000     0.00000E+00
*                0.01200     0.00000E+00         0.01400     0.00000E+00
*                0.01600     0.00000E+00         0.01800     0.00000E+00
*                0.02000     0.90800E+02         0.02200     0.71000E+02
*                0.02400    -0.14740E+03         0.02600    -0.15100E+03
*                0.02800     0.66600E+02         0.03000     0.65400E+02
*                0.03200     0.17540E+03         0.03400     0.56000E+02
*                0.03600     0.19520E+03         0.03800    -0.63600E+02
*                0.04000    -0.61000E+02         0.04200    -0.18760E+03
*                0.04400    -0.95000E+02         0.04600     0.15000E+02
*                0.04800     0.14980E+03         0.05000     0.68000E+02
*                0.05200     0.50000E+02         0.05400     0.14460E+03
*                0.05600    -0.42600E+02         0.05800     0.17840E+03
*                0.06000    -0.10480E+03         0.06200     0.13120E+03
*                0.06400    -0.72000E+02         0.06600     0.14000E+01
*                0.06800     0.48000E+01         0.07000    -0.30200E+02
*                0.07200    -0.84600E+02         0.07400    -0.44000E+02
*                0.07600    -0.13200E+02         0.07800    -0.13400E+02
```

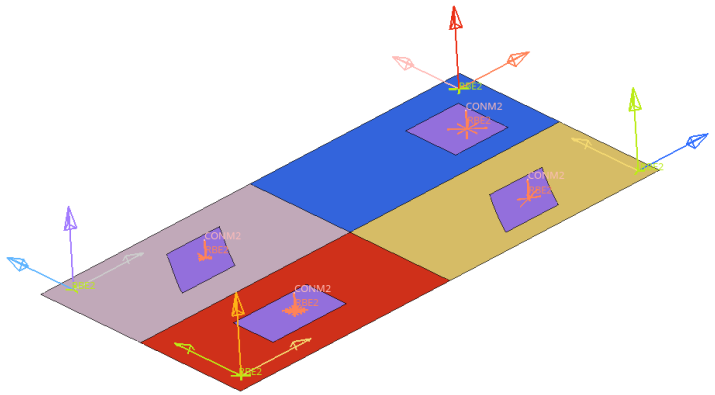
Animation Around Times of Interest

Distance between the node pair drops to almost zero — confirming the predicted contact instant.





Ex-14 — Dynamic Analysis — IR vs IR-by-Modes



- LargePlate_irst.fem
- LargePlate_irst.out
- LargePlate_irst.pch
- LargePlate_modes.fem
- LargePlate_modes.out
- LargePlate_modes.pch
- largePlate_wMs_timeHist.inp
- LargePlateLoads_12LCs.csv

Why is IR-Static still needed? Practical situations — very stiff structures (e.g., control arms) and early-stage designs that are not ready for full dynamic analysis — still call for IR-static. As with fatigue, dynamic responses from IR-static and IR-static-by-modes are expected to correlate exactly.

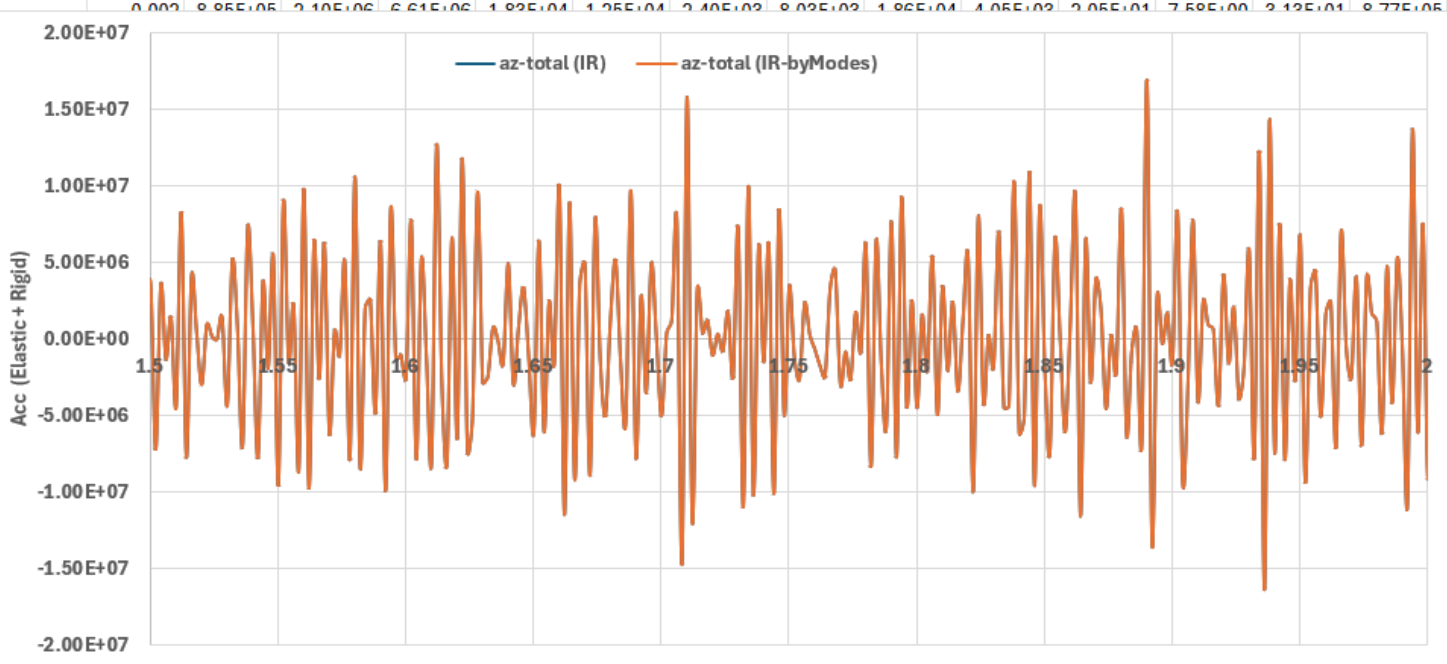
- TH1_IR_cbar_21.csv
- TH1_IR_cbeam_31.csv
- TH1_IR_cbush_11.csv
- TH1_IR_cbush_101.csv
- TH1_IR_cbush_102.csv
- TH1_IR_cbush_103.csv
- TH1_IR_cbush_104.csv
- TH1_IR_celas_1.csv
- TH1_IR_celas_2.csv
- TH1_IR_celas_3.csv
- TH1_IR_celas_4.csv
- TH1_IR_celas_5.csv
- TH1_IR_celas_6.csv
- TH1_IR_crod_41.csv
- TH1_IR_crod_42.csv
- TH1_IR_crod_43.csv
- TH1_IR_crod_44.csv
- TH1_IR_node_1.csv
- TH1_IR_node_2.csv
- TH1_IR_node_3.csv
- TH1_IR_node_4.csv
- TH1_IR_node_11.csv
- TH1_IR_node_12.csv
- TH1_IR_node_13.csv
- TH1_IR_node_14.csv
- TH1_IR_nodeAcc_1.csv
- TH1_IR_nodeAcc_2.csv
- TH1_IR_nodeAcc_3.csv
- TH1_IR_nodeAcc_4.csv
- TH1_IR_nodeAcc_11.csv
- TH1_IR_nodeAcc_12.csv
- TH1_IR_nodeAcc_13.csv
- TH1_IR_nodeAcc_14.csv
- TH1_IR_nodePairPlotel_13358.csv
- TH1_IR_nodePairPlotel_13359.csv
- TH1_IR_nodePairPlotel_13360.csv
- TH1_IR_nodePairPlotel_13361.csv
- TH1_IR_NodePairs_summary.csv
- TH1_IR_NodePairs_summary.HWascii
- TH1_IR_shellElem_1001.csv
- TH1_IR_shellElem_1002.csv
- TH1_IR_shellElem_1003.csv
- TH1_IR_shellElem_1004.csv
- TH1_IR_shellElem_2001.csv
- TH1_IR_shellElem_2002.csv
- TH1_IR_shellElem_2003.csv
- TH1_IR_shellElem_2004.csv
- TH2_IRMODES_cbar_21.csv
- TH2_IRMODES_cbeam_31.csv
- TH2_IRMODES_cbush_11.csv
- TH2_IRMODES_cbush_101.csv
- TH2_IRMODES_cbush_102.csv
- TH2_IRMODES_cbush_103.csv
- TH2_IRMODES_cbush_104.csv
- TH2_IRMODES_celas_1.csv
- TH2_IRMODES_celas_2.csv
- TH2_IRMODES_celas_3.csv
- TH2_IRMODES_celas_4.csv
- TH2_IRMODES_celas_5.csv
- TH2_IRMODES_celas_6.csv
- TH2_IRMODES_crod_41.csv
- TH2_IRMODES_crod_42.csv
- TH2_IRMODES_crod_43.csv
- TH2_IRMODES_crod_44.csv
- TH2_IRMODES_mDynEffect.csv
- TH2_IRMODES_node_1.csv
- TH2_IRMODES_node_2.csv
- TH2_IRMODES_node_3.csv
- TH2_IRMODES_node_4.csv
- TH2_IRMODES_node_11.csv
- TH2_IRMODES_node_12.csv
- TH2_IRMODES_node_13.csv
- TH2_IRMODES_node_14.csv
- TH2_IRMODES_nodeAcc_1.csv
- TH2_IRMODES_nodeAcc_2.csv
- TH2_IRMODES_nodeAcc_3.csv
- TH2_IRMODES_nodeAcc_4.csv
- TH2_IRMODES_nodeAcc_11.csv
- TH2_IRMODES_nodeAcc_12.csv
- TH2_IRMODES_nodeAcc_13.csv
- TH2_IRMODES_nodeAcc_14.csv
- TH2_IRMODES_nodePairPlotel_13358.csv
- TH2_IRMODES_nodePairPlotel_13359.csv
- TH2_IRMODES_nodePairPlotel_13360.csv
- TH2_IRMODES_nodePairPlotel_13361.csv
- TH2_IRMODES_NodePairs_summary.csv
- TH2_IRMODES_NodePairs_summary.HWascii
- TH2_IRMODES_shellElem_1001.csv
- TH2_IRMODES_shellElem_1002.csv
- TH2_IRMODES_shellElem_1003.csv
- TH2_IRMODES_shellElem_1004.csv
- TH2_IRMODES_shellElem_2001.csv
- TH2_IRMODES_shellElem_2002.csv
- TH2_IRMODES_shellElem_2003.csv
- TH2_IRMODES_shellElem_2004.csv

Dynamic Results for Tradional Inertia-Relief/Static and Inertia-Relief/Static by Modes in CSV format

Total Acceleration — IR vs IR-by-Modes

Total acceleration = elastic + rigid-body components.

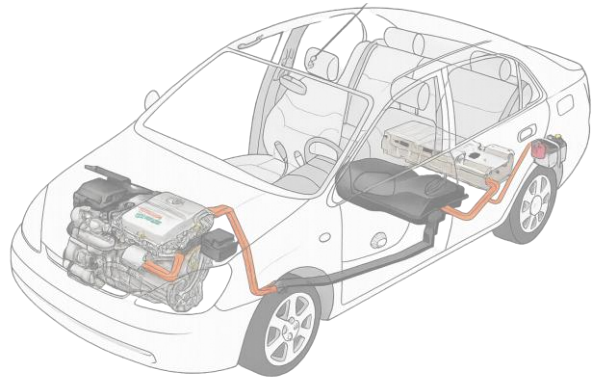
ay-total	az-total (IR)	aRx-total	aRy-total	aRz-total	time	ax-elastic	ay-elastic	az-elastic	Rx-elastic	Ry-elastic	Rz-elastic	rigidbody	rigidbody	rigidbody	rigidbody	rigidbody	rigidbody	ax-total	ay-total	az-total (IR-byModes)	aRx-total
1.06E+06	-3.31E+06	9.17E+03	-6.23E+03	-1.19E+03	0	-4.43E+05	1.05E+06	-3.31E+06	9.15E+03	-6.24E+03	-1.20E+03	1.12E+02	8.04E+03	5.32E+02	2.08E+01	6.31E+00	4.00E+00	-4.42E+05	1.06E+06	-3.31E+06	9.17E+03
2.08E+06	-6.61E+06	1.83E+04	-1.25E+04	-2.37E+0?	0.002	-8.85E+05	2.10E+06	-6.61E+06	1.83E+04	-1.25E+04	-2.40E+03	8.03E+03	1.86E+04	4.05E+03	2.05E+01	7.58E+00	3.13E+01	-8.77E+05	2.08E+06	-6.61E+06	1.83E+04
-1.01E+06	3.12E+06	-2.74E+03	9.56E+03	1.01E+0?															-1.01E+06	3.12E+06	-2.74E+03
1.32E+06	-7.99E+06	1.82E+04	-2.20E+04	-1.57E+0?															1.32E+06	-7.99E+06	1.82E+04
-2.42E+06	1.30E+07	-3.92E+04	2.79E+04	3.08E+0?															-2.42E+06	1.30E+07	-3.92E+04
1.80E+06	-2.81E+06	1.74E+04	1.69E+03	-2.12E+0?															1.80E+06	-2.81E+06	1.74E+04
-9.51E+05	-5.73E+06	4.60E+03	-2.30E+04	9.98E+0?															-9.51E+05	-5.73E+06	4.60E+03
-1.22E+05	4.42E+06	-1.51E+03	2.14E+04	-1.22E+0?															-1.22E+05	4.42E+06	-1.51E+03
9.24E+05	-4.63E+06	3.11E+03	-2.04E+04	-8.04E+0?															9.24E+05	-4.63E+06	3.11E+03
5.04E+04	2.68E+06	-2.91E+01	1.19E+04	-2.44E+0?															5.04E+04	2.68E+06	-2.91E+01
-1.69E+05	9.68E+05	-2.93E+03	3.72E+03	1.47E+0?															-1.69E+05	9.68E+05	-2.93E+03
-1.57E+06	8.24E+05	-1.34E+04	-1.27E+04	1.99E+0?															-1.57E+06	8.24E+05	-1.34E+04
1.99E+06	-6.06E+06	3.05E+04	4.36E+03	-2.85E+0?															1.99E+06	-6.06E+06	3.05E+04
2.98E+04	4.38E+06	-1.83E+04	1.43E+03	5.08E+0?															2.98E+04	4.38E+06	-1.83E+04
-1.36E+06	-1.15E+06	5.35E+03	-2.26E+03	1.27E+0?															-1.36E+06	-1.15E+06	5.35E+03
7.31E+05	3.08E+06	-8.70E+03	9.20E+03	-5.91E+0?															7.31E+05	3.08E+06	-8.70E+03
6.00E+05	-4.33E+06	2.28E+03	-1.79E+04	-5.97E+0?															6.00E+05	-4.33E+06	2.28E+03
-1.09E+06	9.29E+06	-1.23E+04	3.36E+04	1.16E+0?															-1.09E+06	9.29E+06	-1.23E+04
7.48E+05	-8.73E+06	1.87E+04	-2.52E+04	-1.02E+0?															7.48E+05	-8.73E+06	1.87E+04
2.59E+04	1.03E+06	6.23E+03	7.80E+03	-7.85E+0?															2.59E+04	1.03E+06	6.23E+03
-2.12E+06	3.16E+06	-2.35E+04	-5.20E+03	2.57E+0?															-2.12E+06	3.16E+06	-2.35E+04
2.51E+06	-6.25E+06	2.52E+04	-3.60E+03	-3.04E+0?															2.51E+06	-6.25E+06	2.52E+04
-1.56E+06	8.55E+06	-2.64E+04	1.40E+04	2.02E+0?															-1.56E+06	8.55E+06	-2.64E+04
1.31E+06	-9.83E+06	2.58E+04	-2.35E+04	-1.69E+0?															1.31E+06	-9.83E+06	2.58E+04
-2.24E+05	8.55E+06	-1.06E+04	3.34E+04	2.65E+02	0.048	1.96E+05	-2.06E+05	8.55E+06	-1.06E+04	3.34E+04	2.41E+02	8.71E+03	-1.82E+04	-1.59E+03	-3.77E+01	-1.47E+01	2.39E+01	2.05E+05	-2.24E+05	8.55E+06	-1.06E+04
-2.01E+05	-9.68E+06	1.20E+04	-3.91E+04	2.15E+02	0.05	-3.73E+05	-1.91E+05	-9.71E+06	1.21E+04	-3.92E+04	2.09E+02	5.29E+03	-9.77E+03	2.34E+04	-4.09E+01	1.88E+01	6.09E+00	-3.68E+05	-2.01E+05	-9.68E+06	1.20E+04
7.63E+05	1.23E+07	-2.93E+04	3.47E+04	-4.07E+02	0.052	4.46E+05	7.71E+05	1.22E+07	-2.93E+04	3.47E+04	-4.07E+02	-1.06E+04	-8.23E+03	1.48E+03	7.45E+00	6.69E+00	-4.53E-01	4.36E+05	7.63E+05	1.23E+07	-2.93E+04
-2.84E+06	-4.19E+06	8.22E+03	-1.50E+04	2.84E+03	0.054	3.64E+05	-2.86E+06	-4.19E+06	8.22E+03	-1.50E+04	2.86E+03	-5.65E+03	2.29E+04	5.12E+03	-1.99E+00	5.44E+00	-2.70E+01	3.59E+05	-2.84E+06	-4.19E+06	8.22E+03



IR: rigid-body acceleration is computed from unit-load case accelerations in `LargePlate_irst.out`.
IR-by-Modes: rigid-body acceleration is computed from rigid modes (1–6) in `LargePlate_modes.pch`.
These outputs provide physical insight that supports improved designs.

Mass-Force Time Histories

FatDyn-X delivers precise mass-force time histories on every concentrated mass — across all solver modes (Inertia-Relief, IR-by-Modes, Modal-Transient). Most tools cannot do this, leaving mount design dependent on guesswork.

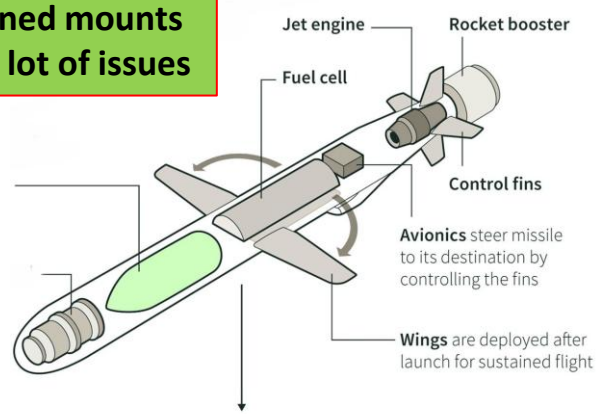


The industry challenge

Modern mass systems — hybrid/EV powertrains, missile payloads, avionics — are highly complex.

- Most software cannot predict the actual forces acting on lumped masses.
- Unpredicted mass-forces lead to unsafe vibrations, unstable trajectories, and premature mount failures.

Well designed mounts can solve a lot of issues

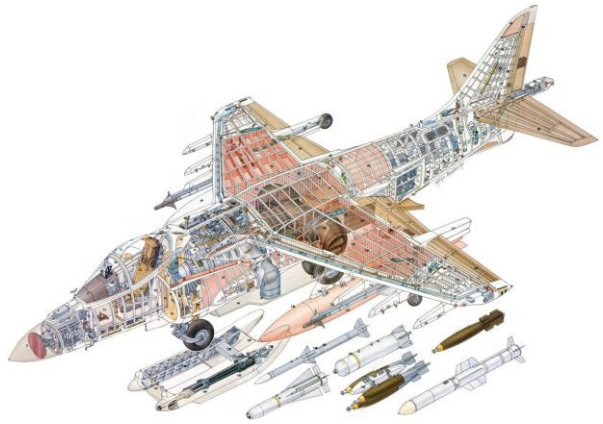


The FatDyn-X advantage

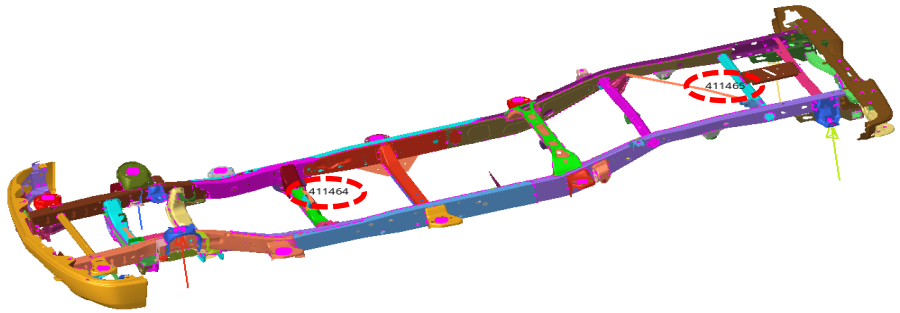
Precise mass-force time histories across all solver types: Inertia-Relief, IR-by-Modes, and Modal Transient.

- Automotive (EV/Hybrid): battery and motor loads under vibration
- Missiles: impulse-driven forces during launch and flight
- Aircraft: structural loads on avionics and heavy equipment

Bottom line: design and optimize the mounts of heavy components based on calculated forces, not assumptions.



Ex-15 — Time History of Mass (CONM2) Elements



- This example used two CONM2 elements as highlighted.
- The frame was analyzed for two scenarios:
  - A Belgian-Block dynamic event
  - A Dirt-Grade quasi-static event
- The complete input file is shown on the right. Only SOL-101 and SOL-103 punch files were required to predict the time history using inertia-relief (static) and modal transient analyses

Complete Input Data

```
$-------------------------------------------------------------
*Element_Sets
Label: conm2set
   411464,  411465

*FE_Data
        file_Name_FE_NasModel : truckFrame_IRstatic.fem
        file_Name_FE_PCH101   : truckFrame_IRstatic.pch

        file_Name_FE_PCH103   : truckFrame_modes.pch
        modes          :  7 to  8  critical_damping = 0.05
        static_modes :  9 to 34
$-------------------------------------------------------------
*TimeHistory_LoadCases
        Label: Belg
        Analysis_method: modal_transient

        massForce_timeHistory: YES  conm2Set

    Analysis_Time_Step  : 0.000977
$ SerialNo   Label   loadChannel_fileName      noHeadLines   COL_id
     1,      Fz_1,   BelgianBlockRd_Run4.csv,      1,         3,
     2,      Fz_2,   BelgianBlockRd_Run4.csv,      1,         4,
     3,      Fz_3,   BelgianBlockRd_Run4.csv,      1,         1,
     4,      Fz_4,   BelgianBlockRd_Run4.csv,      1,         2,

!=============================================
        Label: Dirt
        Analysis_method: INERTIA_RELIEF/STATIC

        massForce_timeHistory: YES  conm2Set

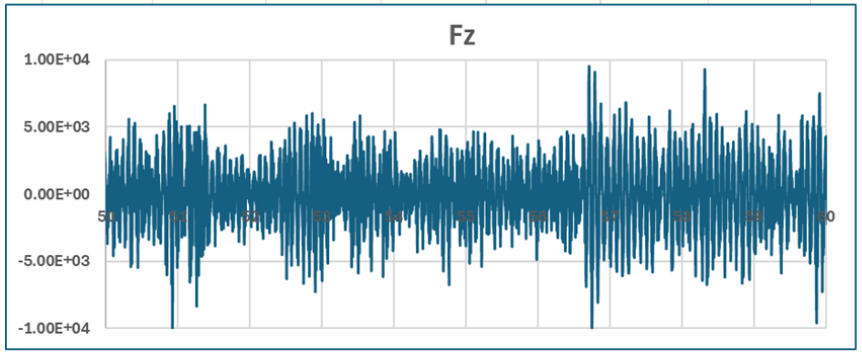
    Analysis_Time_Step  : 0.000977
$ SerialNo   Label   loadChannel_fileName      noHeadLines   COL_id
     1,      Fz_1,   DirtGrade20_Run1.csv,         1,         3,
     2,      Fz_2,   DirtGrade20_Run1.csv,         1,         4,
     3,      Fz_3,   DirtGrade20_Run1.csv,         1,         1,
     4,      Fz_4,   DirtGrade20_Run1.csv,         1,         2,
```

Time History of CONM2 Elements (cont.)

Mass-force comparison — Element 411464 (Belgian-Block, dynamic) vs. Element 411465 (Dirt-Grade, quasi-static).

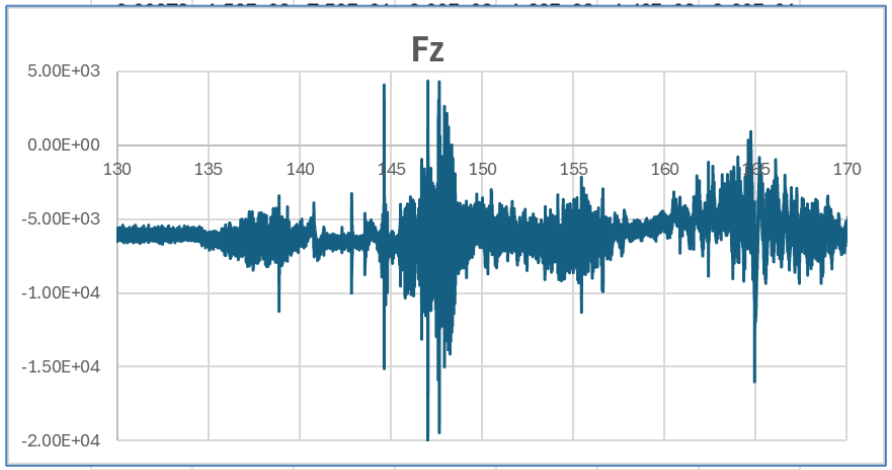
Mass Force of Conm2 Element 411464 for Dynamic Event Belgian Block

time	Fx	Fy	Fz	Mx	My	Mz
0	-2.28E+00	-2.36E+01	-9.64E-02	-1.48E+01	1.74E+01	8.42E+00
0.00098	-4.56E+00	-4.70E+01	-1.93E-01	-2.96E+01	3.48E+01	1.68E+01
0.00195	1.34E+00	1.70E+01	-4.52E+00	7.87E+00	-1.16E+01	-9.98E-01
0.00293	-3.02E+01	3.50E+01	6.72E+01	-4.47E+01	2.46E+02	-8.82E+01
0.00391	-7.14E+00	-4.86E+01	-1.08E+01	4.66E+01	1.65E+01	9.94E+01
0.00489	3.50E+01	-1.83E+01	-5.66E+01	-2.11E+01	-2.43E+02	-2.46E+00
0.00586	-2.28E+01	4.94E+01	5.00E+01	-1.31E+01	1.80E+02	-6.36E+01
0.00684	-8.88E+00	-1.54E+01	2.80E+00	4.13E+01	4.04E+01	5.76E+01
0.00782	6.98E+01	-3.66E+01	-1.13E+02	-4.26E+01	-4.86E+02	-4.94E+00
0.00879	-6.96E+01	2.02E+01	1.12E+02	3.09E+01	4.88E+02	5.60E+00
0.00977	2.38E+01	-1.08E+01	-4.02E+01	4.86E-01	-1.71E+02	1.07E+01



Mass Force of Conm2 Element 411465 for Quasi-static Event Dirt-Grade

time	Fx	Fy	Fz	Mx	My	Mz
0	2.26E+02	1.06E+02	-5.80E+03	2.55E+02	-5.86E+02	1.33E+02
0.00098	2.54E+02	1.94E+02	-5.70E+03	4.53E+02	-7.62E+02	2.53E+02
0.00195	1.11E+02	7.00E+00	-6.10E+03	3.69E+01	-1.61E+02	2.37E+01
0.00293	1.26E+02	-4.59E+01	-6.15E+03	-7.63E+01	-1.55E+02	-5.67E+01
0.00391	3.29E+02	-7.25E+01	-5.95E+03	-1.19E+02	-8.78E+02	-1.83E+02
0.00489	1.22E+02	1.64E+02	-5.75E+03	3.46E+02	2.24E+01	3.03E+02
0.00586	2.28E+02	-3.01E+01	-6.05E+03	-2.88E+01	-6.32E+02	-9.18E+01
0.00684	2.70E+02	1.42E+02	-5.75E+03	3.39E+02	-7.56E+02	1.73E+02
0.00782	1.38E+02	-2.20E+02	-6.30E+03	-4.79E+02	2.36E+01	-3.18E+02



0.02345	1.42E+02	1.70E+02	-5.90E+03	4.10E+02	-3.66E+02	2.68E+02
0.02442	1.67E+02	7.20E+01	-5.85E+03	1.64E+02	-2.02E+02	1.29E+02
0.0254	1.92E+02	-2.07E+02	-6.30E+03	-4.34E+02	-4.10E+02	-3.57E+02

FatDyn-X

(Bolted Joint Design)

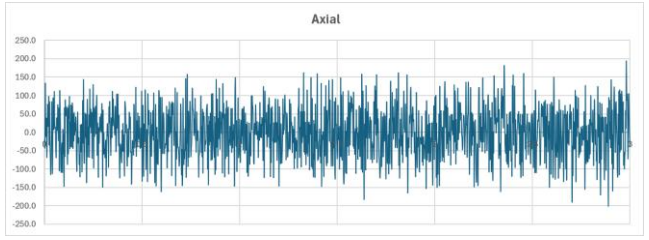
Inertia-Relief Load Cases

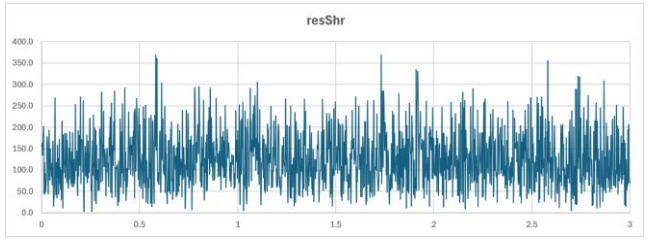
Normal-Modes

Solvers by FatDyn-X

- Inertia-Relief/Static via Normal Modes
- Modal-Transient via Normal Modes
- Modal-Resonance via Normal Modes

Force time-history of joint loads — CBAR / CBEAM elements





Required clamp is calculated as a function of time:

$$C_req(t) = N_a \cdot P^+(t) + N_s \cdot V(t) / (m \cdot \mu)$$

C_req	=	required bolt clamp
m	=	number of friction interfaces
μ	=	friction coefficient
N_a	=	axial-declamp safety factor
N_s	=	shear-slip safety factor
P^+	=	positive axial declamp load
V	=	resultant shear per bolt

Joint Slip Factor & Margin of Safety

Nastran bulk data: CBAR / CBEAM with PBAR, PBARL, PBEAM, PBEAML sections

Bolt	Pitch mm	Proof load kN	Target clamp kN	Dry torque Nm
M5	0.8	8.23	6.17	6
M6	1	11.6	8.7	10
M8	1.25	21.2	15.9	25
M10	1.5	33.7	25.3	51
M12	1.75	48.9	36.7	88
M14	2	66.7	50	140
M16	2	91	68.3	218
M18	2.5	115	86.3	311
M20	2.5	147	110	441
M22	2.5	182	137	601
M24	3	212	159	763
M27	3	275	206	1114
M30	3.5	337	253	1517
M36	4	490	368	2646

```
*JOINT_DESIGNS
Load_cases:  TH1_*, TH2_*
Safety_Margin: YES  LargePlate_irst_longFormat.fem  clampLoadTable.csv  1000.0
# Design     K      d      m     mu      eta     Na     Ns     Elements
  Des1      0.20   12.0    2     0.15    0.85    1.5    1.5    1 to 6, 41 to 44
  Des2      0.18   20.0    2     0.15    0.85    1.5    1.5    101 to 104
  Default   0.20   16.0    2     0.15    0.85    1.0    1.0    *
```

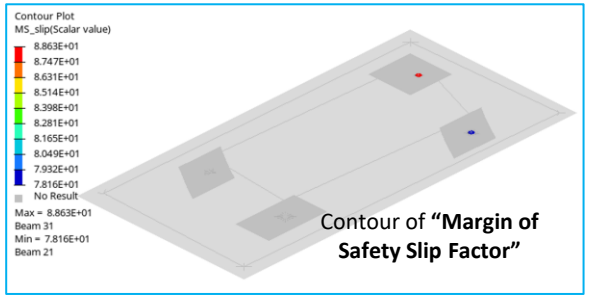
- The joint program reads each bolt diameter from the bulk data and maps it to a target clamp, C_t, from the design table.
- Using the CAE-calculated required clamp, C_{req}, it computes the slip factor and margin of safety, then writes **results in CSV and HWASCII formats**.

Slip Factor: $SF_{\text{slip}} = \frac{C_t}{C_{\text{req}}}$

Margin of safety: $MS_{\text{slip}} = \frac{C_t}{C_{\text{req}}} - 1$

Definitions

- *Creq*: required clamp calculated from CAE bolt axial and shear load
- *Ct* : available target clamp from bolt design-table



Contour of **"Margin of Safety Slip Factor"**

Application

Recently applied on an EV program with ~1,000 bolted joints under long time-history loading. Contour plots of the **"slip-factor margin of safety"** made bolt-size selection fast and visual.

SECTION 8

Choosing Analysis Methods

Use Time + PSD plots embedded with structural frequencies to classify each event and mode.

Which Events Need Dynamic Runs?

Context

Durability programs include many events — from rough road to braking and jumps. Some are quasi-static; others excite structural modes. Use dynamics only when load–structure frequency interaction is significant.

Why not "dynamic everything"?

It is costly and can mislead. Contact changes and other nonlinearities create spurious resonances that disappear under realistic motion.

NVH vs. Durability modes

Normal modes describe small-displacement wave behaviour (ideal for NVH). Durability involves larger motion and boundary effects — mode selection and constraints must be handled carefully.

What FatDyn-X produces

For each event × channel, automated time histories and PSD with natural-frequency overlays (or modal loads in modal space).

Decision output

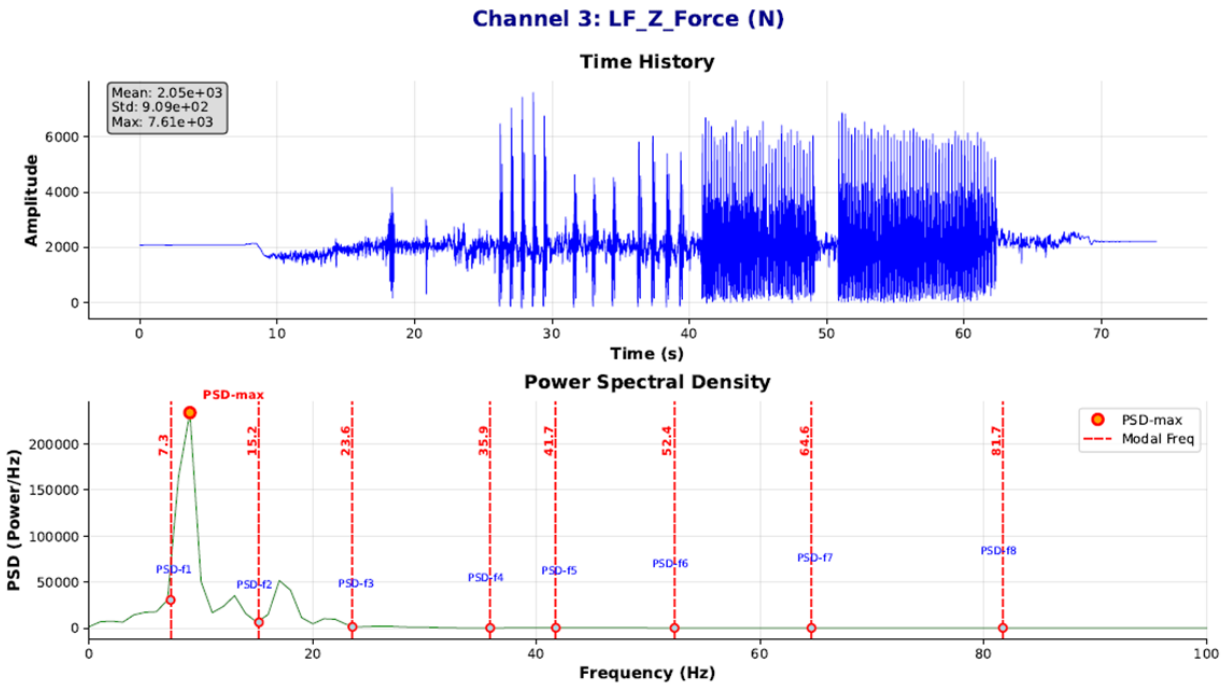
Frequency intersections and simple statistics feed a CSV summary that classifies each event as Dynamic-Transient or Quasi-Static / Inertia-Relief.

How FatDyn-X Flags Dynamic Runs

- For each event × channel, FatDyn-X auto-builds Time and PSD plots; vertical lines mark natural frequencies.
- At each mode f_n, take the PSD value at the intersection and compare to channel PSD_{max} → Ratio = $PSD(f_n) / PSD_{max}$.
- If any ratio crosses the threshold → the event is Dynamic-Transient.
- Outputs: CSV summary + PDF booklet; remaining events are solved Quasi-Static / Inertia-Relief.

Rule of thumb: if any even's $PSD(f_n) / PSD_{max} \geq$ threshold → Dynamic. Else → Quasi-Static / IR.

Channel	Modal_Freq (Hz)	PSD_Magnitude at Intersection	PSD_Max	(Intersection-magnitude) / PSD_max)
RF_X_Force (N)	15.19	4371.06	7786.91	0.56
RF_Y_Force (N)	35.87	1145.69	2416.89	0.47
LF_Y_Force (N)	35.87	1841.01	4633.36	0.40
LF_X_Force (N)	15.19	3315.81	9353.67	0.35
RF_X_Force (N)	7.28	1884.83	7786.91	0.24
LF_X_Force (N)	7.28	1925.24	9353.67	0.21
RF_Y_Force (N)	15.19	485.57	2416.89	0.20
RF_Y_Force (N)	7.28	439.15	2416.89	0.18
LF_Y_Force (N)	7.28	706.61	4633.36	0.15
LF_Z_Force (N)	7.28	30457.78	233714.42	0.13
LF_Y_Force (N)	15.19	580.39	4633.36	0.13
RF_Z_Force (N)	7.28	28868.24	254536.84	0.11
RF_X_Force (N)	23.55	534.10	7786.91	0.07
RF_Y_Force (N)	23.55	125.49	2416.89	0.05
LF_X_Force (N)	23.55	485.49	9353.67	0.05
LF_Y_Force (N)	23.55	208.00	4633.36	0.04
RF_Y_Force (N)	41.74	108.49	2416.89	0.04
RF_X_Force (N)	35.87	315.83	7786.91	0.04
RF_X_Force (N)	41.74	268.06	7786.91	0.03
LF_X_Force (N)	35.87	319.59	9353.67	0.03
LF_Y_Force (N)	41.74	132.24	4633.36	0.03
LF_Z_Force (N)	15.19	6221.58	233714.42	0.03



Which Modes Need Dynamic Runs?

Modal-space view: For each event, project loads to modal space and auto-generate a Time + PSD plot with mode-frequency lines.

Intersection check: For each mode f_n, use $PSD(f_n)/PSD_{max}$ on the modal-load PSD; if a threshold trips → run dynamic transient for that mode.

Sanity filter: High ratios are validated by mode animation / strain-energy to drop spurious or NVH-only modes.

Strategy

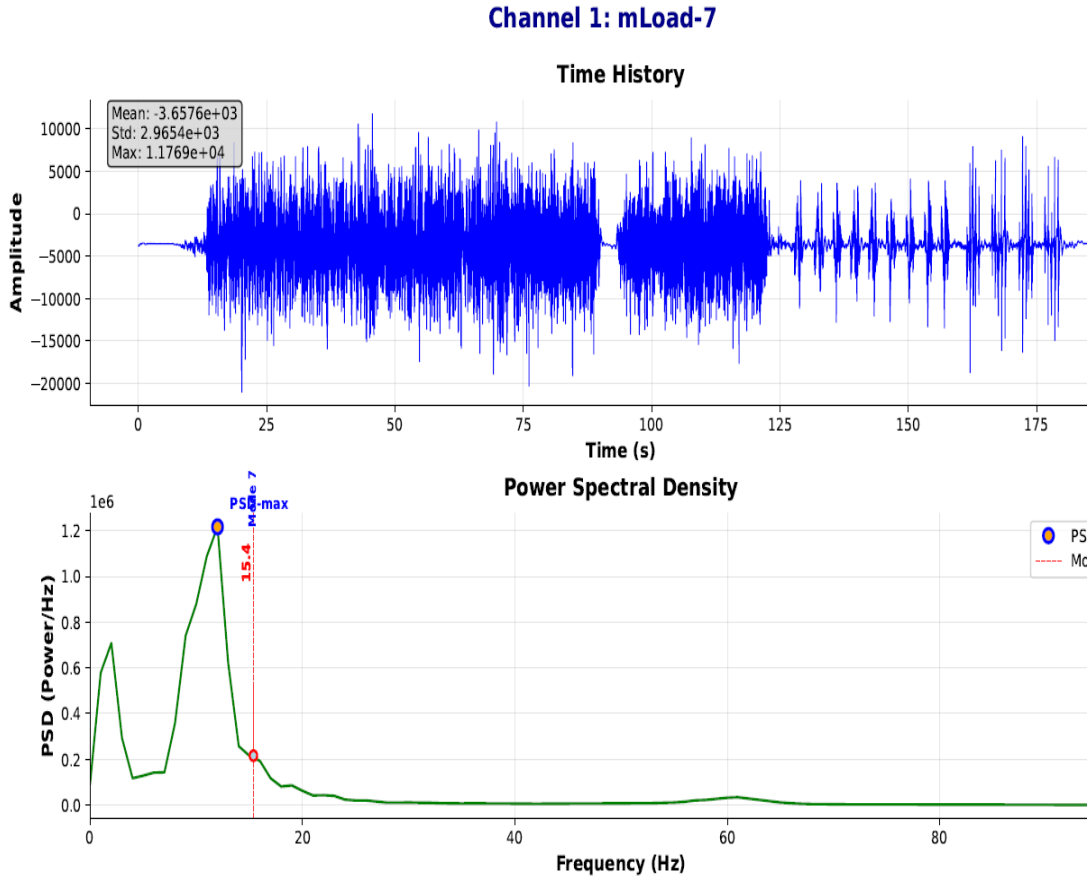
IR-by-Modes — no SOL 101 step required: Output only modal loads for every event

Automatic classification: Each event tagged Modal-Transient (run dynamics) or IR-by-Modes (quasi-static with selected modes) based on modal-space metrics.

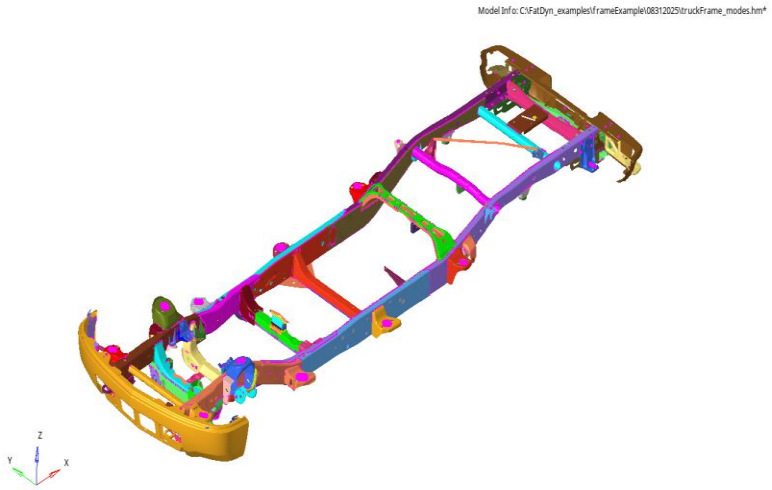
Realistic results: Modes with low intersection magnitude are solved quasi-statically — preventing unrealistic resonance effects.

Loads acting on modes

(only FatDyn-X outputs them)



Ex-16 — Choosing Analysis Methods for Events



Model: open-source FE truck frame.
Loading: four inputs in the Z-direction (production frames often have hundreds of channels).
Events: seven proving-ground cases adapted from a typical vehicle program; used for fatigue loads.
What is next: time histories and PSDs classify each event as dynamic or quasi-static.
Also tracked: which load channel excites structural resonance.

Events analysed

- BelgianBlockRd_Run4
- DirtGrade20_Run1
- Durability2_Run1
- Figure8_Run2
- LatLoads_Run3
- Jump_Run1
- Washboard_Run2

Classify each event

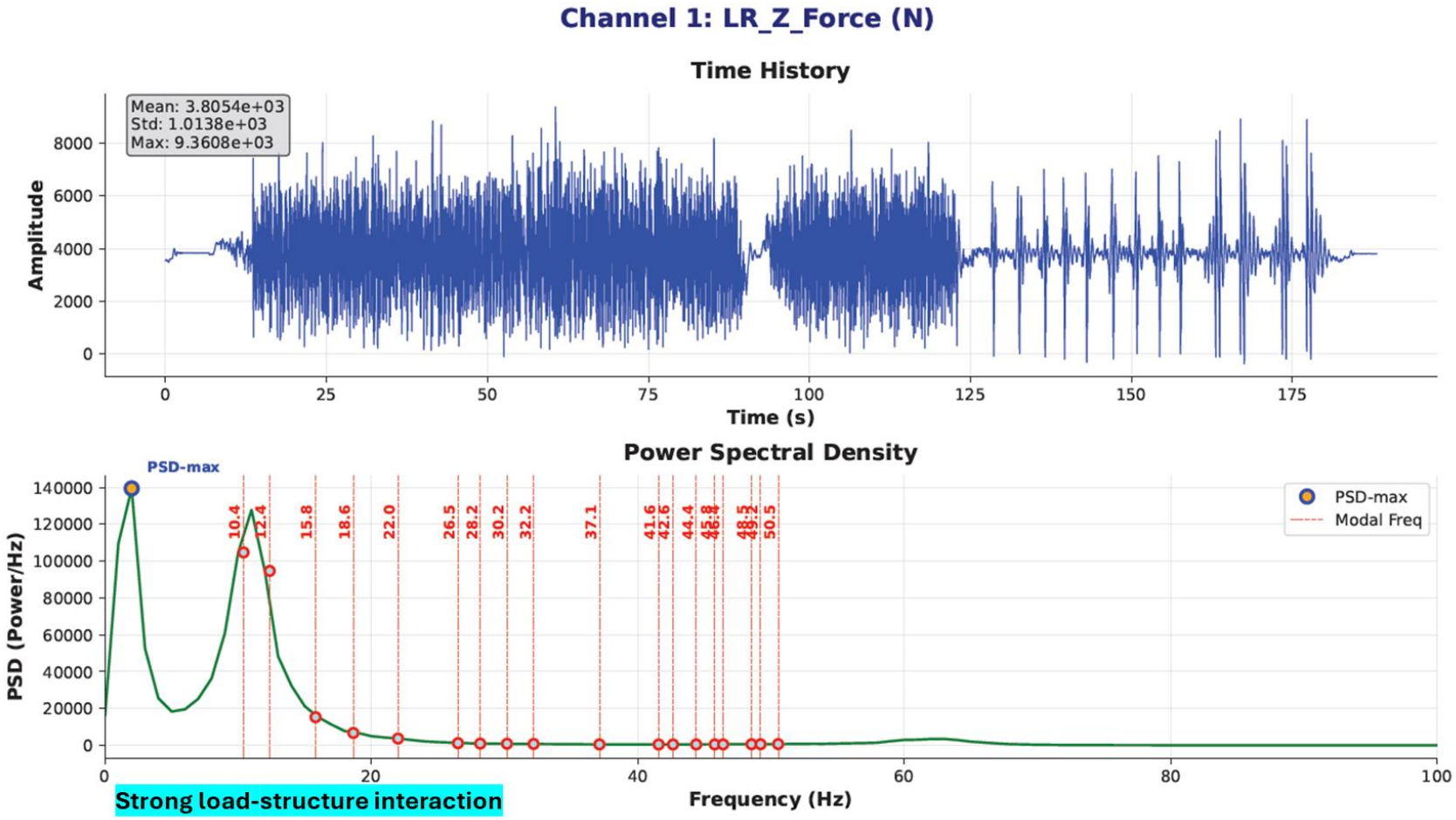
- Dynamic events?
- Quasi-static events?

→

Decided with

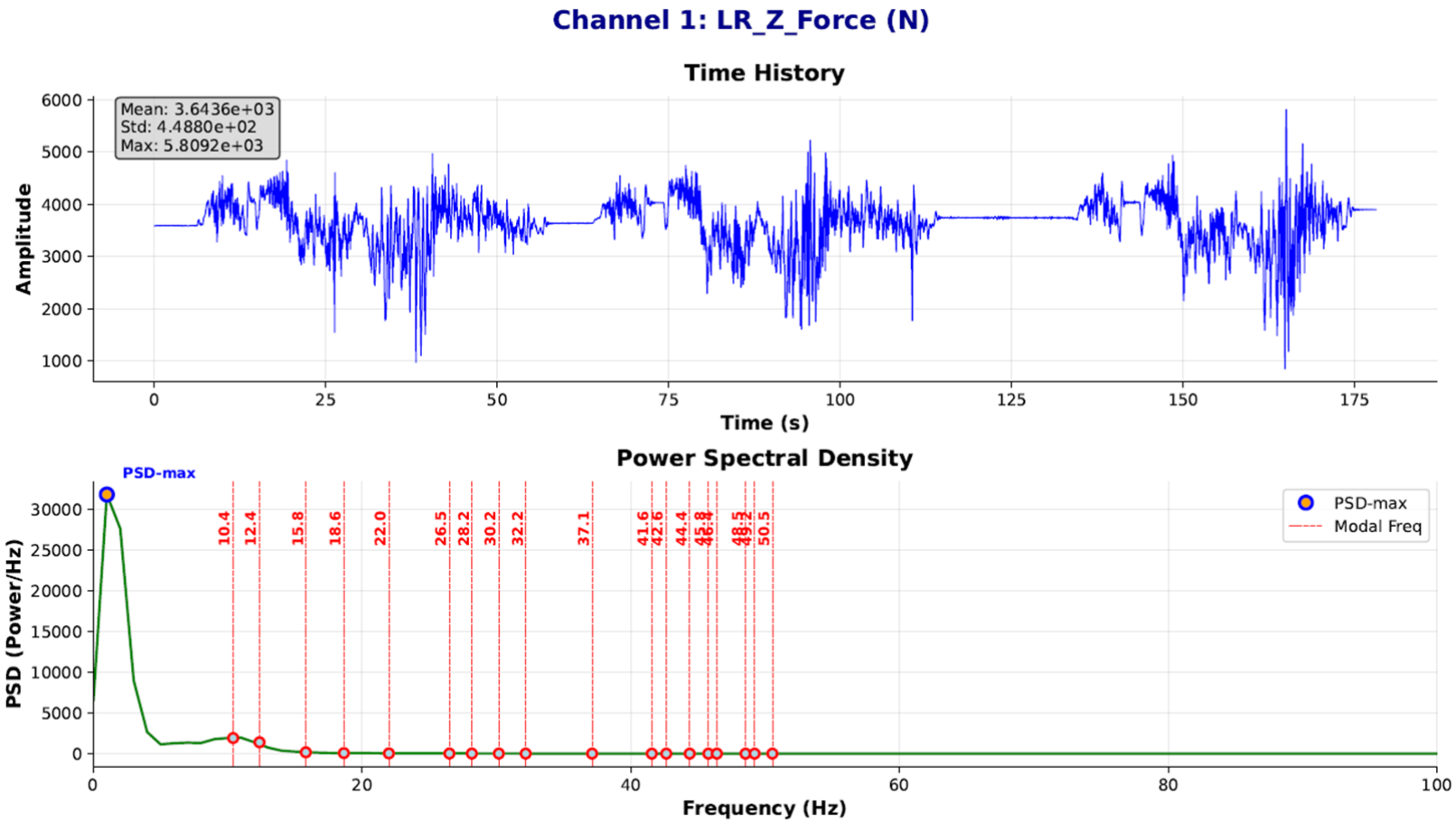
Automatic time / PSD plots embedded with structural frequencies.

Event: Belgian Block — needs transient analysis



PSD shows strong load–structure interaction near several modes → run dynamic transient.

Event: Dirt Road — inertia-relief is sufficient



PSD energy is concentrated below the lowest mode → quasi-static / inertia-relief is sufficient.

Ex-17 — Choosing Analysis Methods for Modes

Earlier time/PSD screening classified the events. The next step identifies which structural modes are excited and must be treated dynamically.

Dynamic events

- Belgian Block
- Durability
- Jump
- Washboard

Quasi-static events

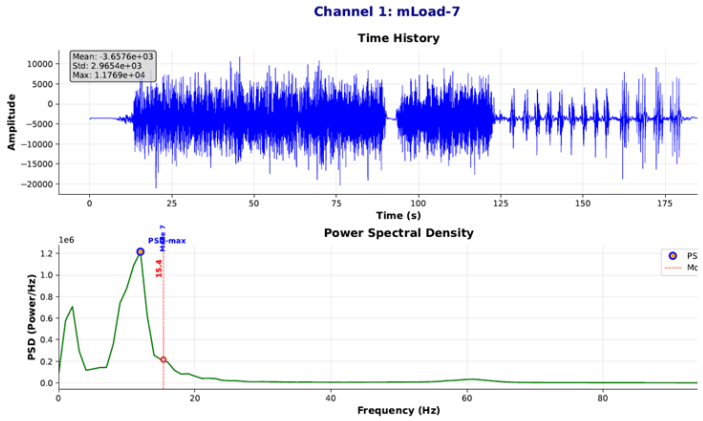
- DirtGrade20_Run
- Figure8_Run2
- LatLoads_Run3

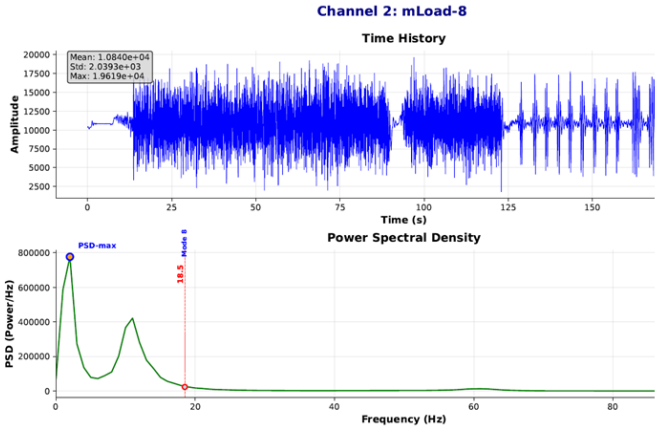
For each dynamic event:

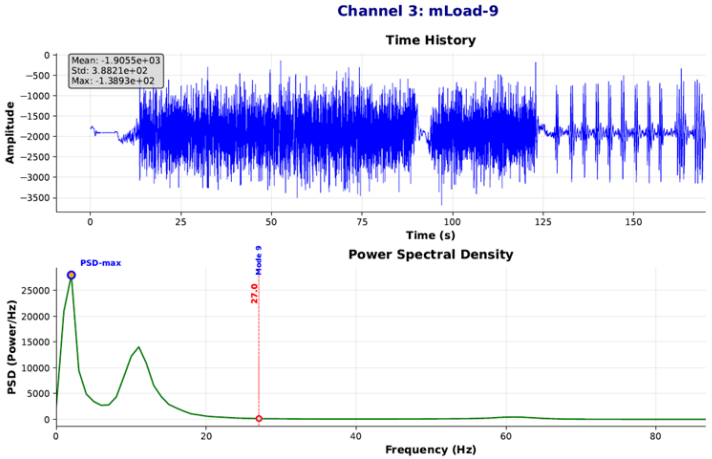
- Dynamic modes?
- Static modes?

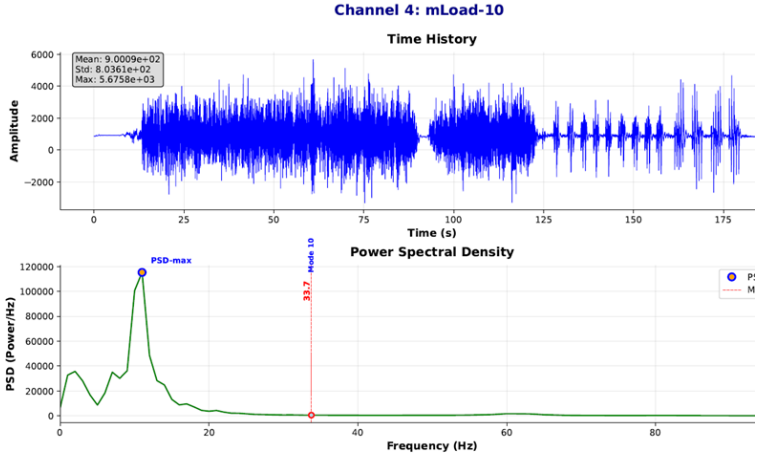
Modal-load PSDs flag which modes need transient solution; the rest stay quasi-static / IR-by-modes.

Belgian Block — modes 7 and 8 are dynamic



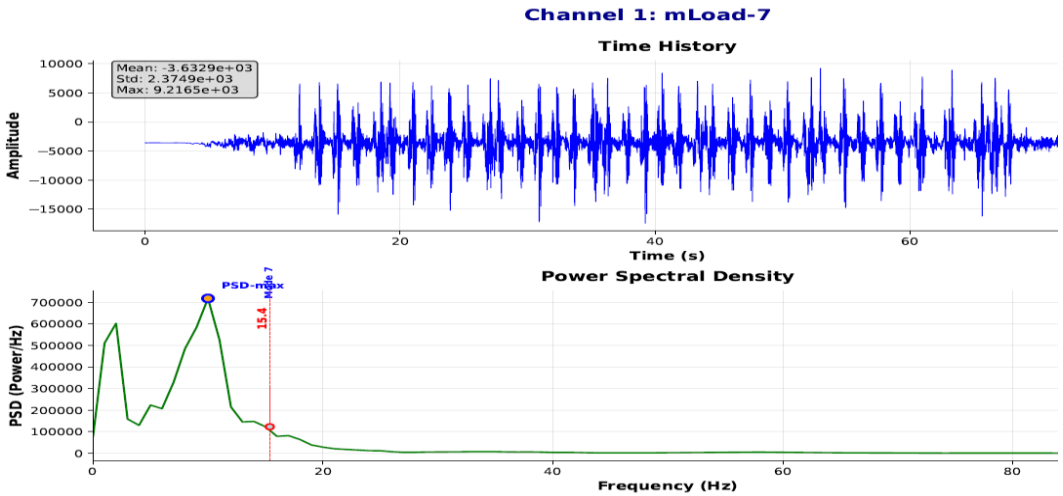


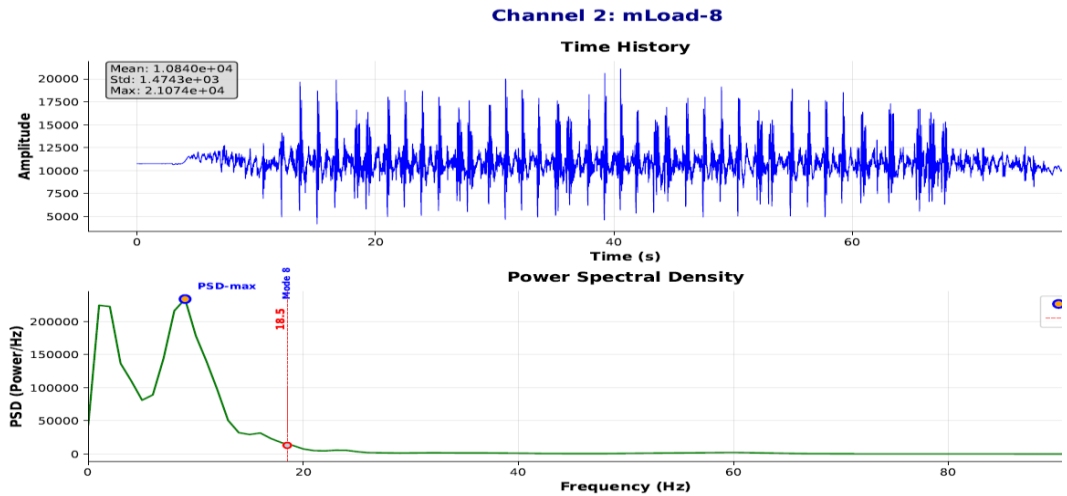


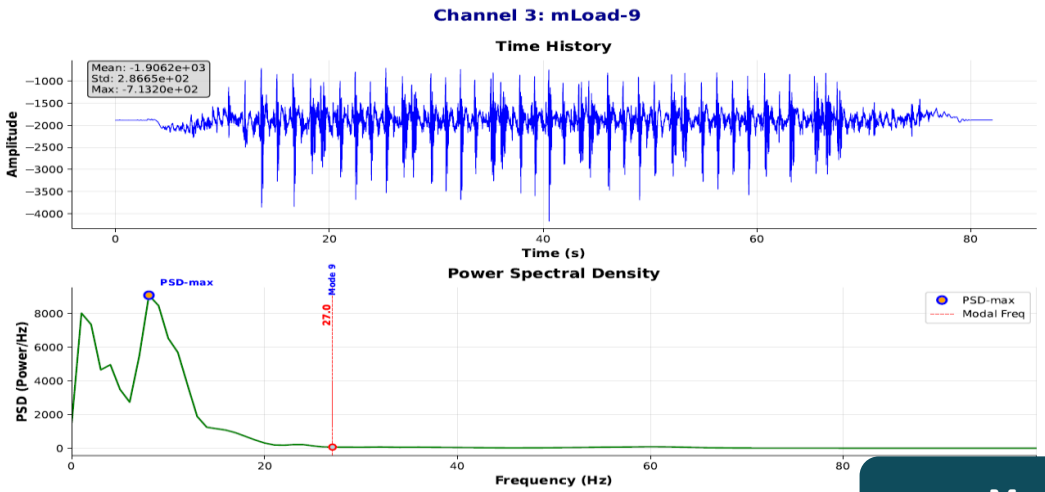


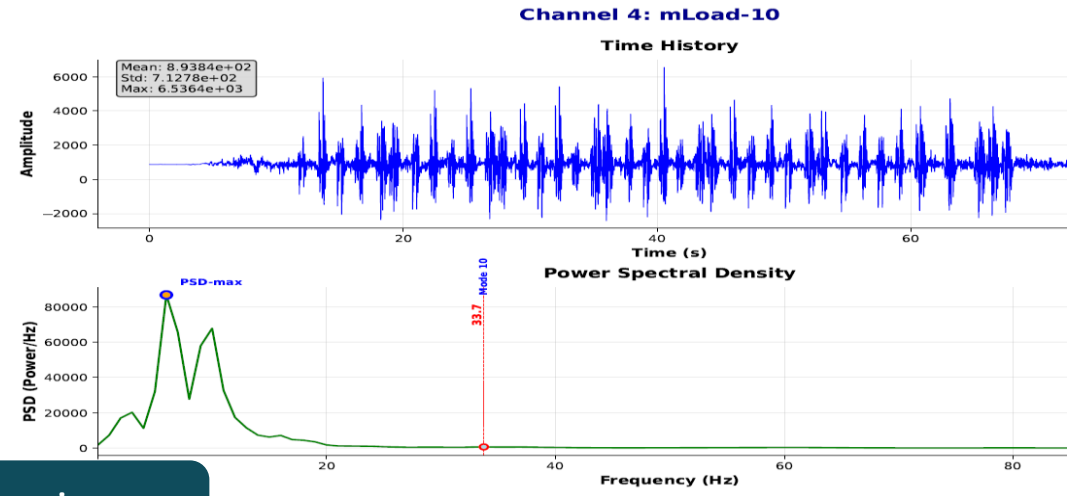
Modes 7 and 8 are dynamic

Durability Run — modes 7 and 8 are dynamic









Modes 7 and 8 are dynamic

SECTION 9

Enhanced Inertia-Relief: Acceleration Contours and Mass Forces

Enhanced Inertia-Relief: Acceleration Contours and Mass Forces

From bulk-data and out / F06 file of the current model — or a trusted accurate global Mass-Model.

1

Convert bulk-data + out / F06 into acceleration contours and forces on heavy masses for faster what-if studies.

2

Fuse inertia-relief accelerations with a trusted Mass-Inertia-CG (from crash model or corner-weights).

3

Replace lumped mass with calculated forces to sharpen IR fidelity.

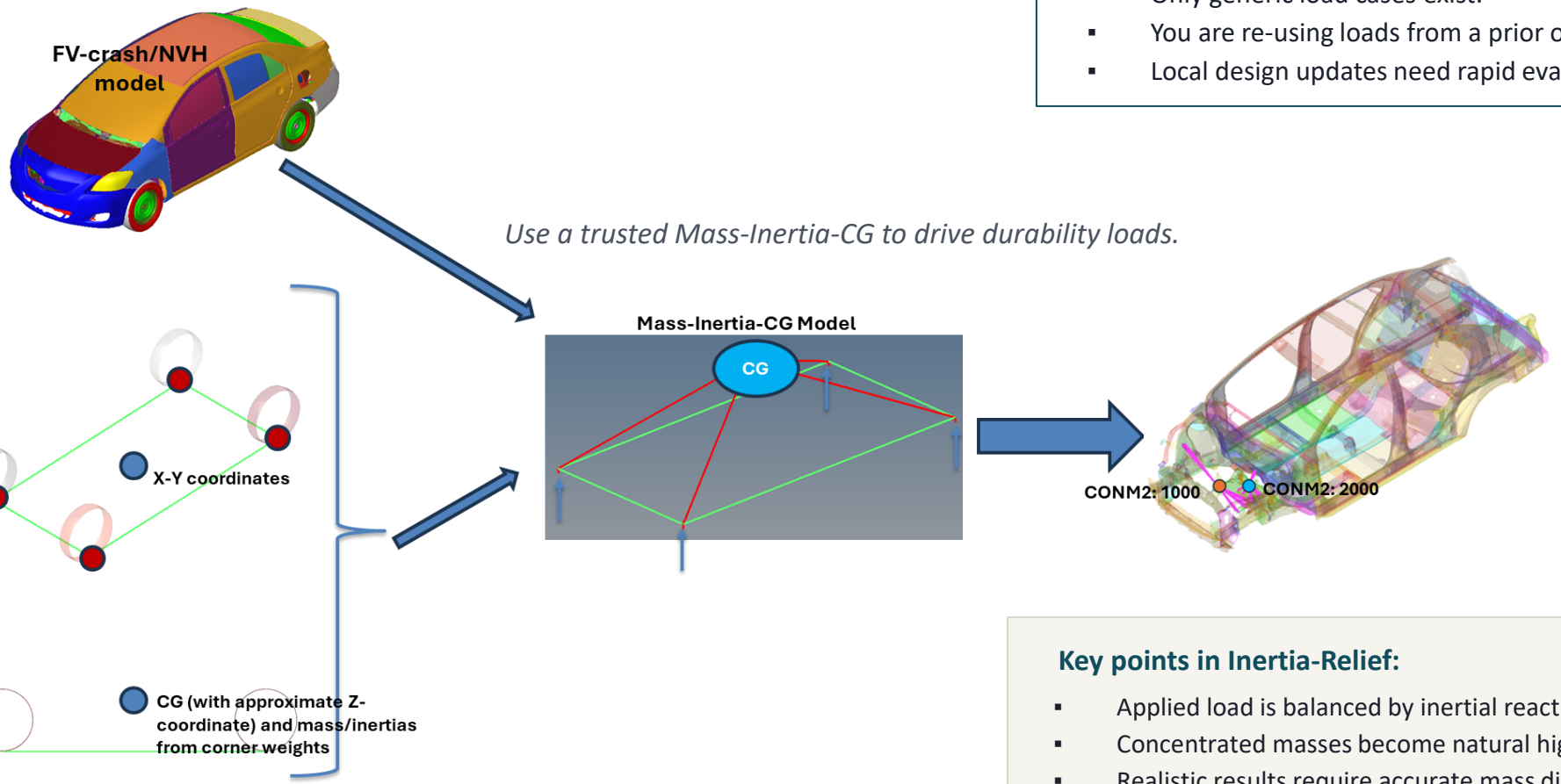
4

Identify critical load paths and optimise the base-structure for durability.

Ex-18 — Acceleration Contours & Mass-Forces from a Global Mass-Inertia-CG

Inertia-Relief is valuable when:

- The durability model lags design (unlike crash).
- Only generic load cases exist.
- You are re-using loads from a prior or similar vehicle.
- Local design updates need rapid evaluation.



Key points in Inertia-Relief:

- Applied load is balanced by inertial reactions × mass.
- Concentrated masses become natural high-stress regions.
- Realistic results require accurate mass distribution and trimming; a Mass-Inertia-CG anchor avoids misleading contours.

Acceleration Contours (g) — Four Load Cases

Inputs: OUT/F06 + element set → Outputs: CSV & HWAscii (contours), bulk-data + CSV (CONM2 forces & moments).

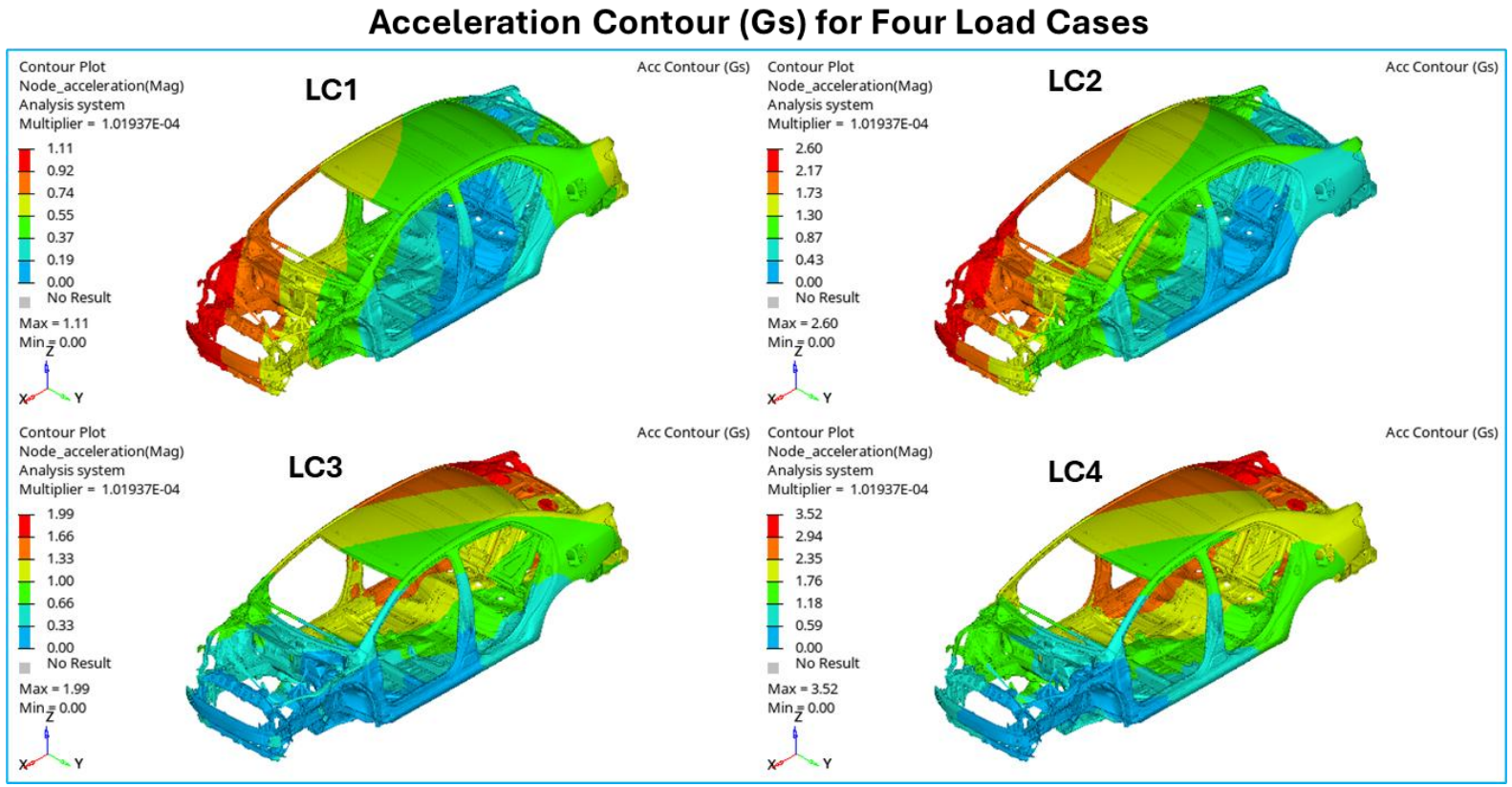
```
*Element_Sets
Label: conm2set
   1000, 2000

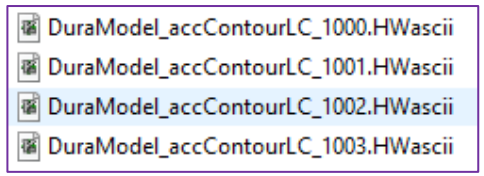
*FE_Data
        file_Name_FE_NasModel : DuraModel.fem
        file_Name_FE_out/F06  : CGmass_IRstatic.out

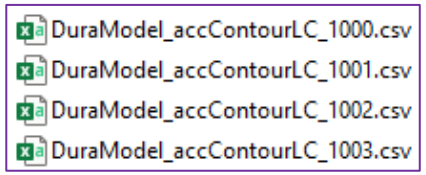
*TimeHistory_LoadCases
      Label: Belg
      Inertia_Relief_Acc_Contour: YES
      Inertia_Relief_Mass_Force : YES    conm2Set
```

Input snippet for FatDyn-X

- Acceleration contours → CSV and HWAscii.
- CONM2 forces & moments → CSV and bulk-data.





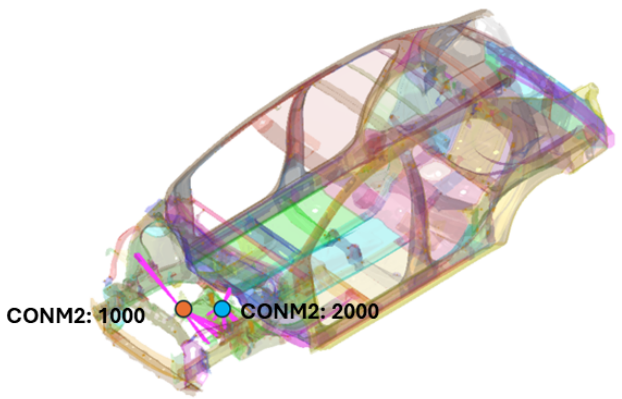


CONM2 Forces & Moments (per Load Case)

Acceleration and Force-Moment calculated for CONM2 elements

Loadcase	LC-id	node-id	mass label	mass	x	y	z	AX	AY	AZ	Rax	Ray	Raz	Fx	Fy	Fz	Mx	My	Mz
LC_1	1	10006	conm2Mass_1000	0.148	-610.710	-99.535	460.180	92.998	-118.920	7613.100	-3.805	-2.997	-0.001	-13790.0	17633.0	-1128900.0	34292.0	11415.0	-9541.6
LC_1	1	10005	conm2Mass_2000	0.016	-636.120	359.050	749.970	-775.050	983.850	5791.900	-3.805	-2.997	-0.001	12579.0	-15968.0	-94002.0	513.4	306.5	-1.1
LC_2	2	10006	conm2Mass_1000	0.148	-610.710	-99.535	460.180	146.540	-331.170	17488.000	-10.597	-4.724	-0.002	-21729.0	49106.0	-2593200.0	98178.0	14589.0	-21490.0
LC_2	2	10005	conm2Mass_2000	0.016	-636.120	359.050	749.970	-1221.600	2739.900	12509.000	-10.597	-4.724	-0.002	19827.0	-44468.0	-203010.0	1430.8	481.8	-2.4
LC_3	3	10006	conm2Mass_1000	0.148	-610.710	-99.535	460.180	-116.220	-209.390	2547.800	-6.700	3.738	-0.001	17233.0	31048.0	-377780.0	67057.0	-22724.0	-4148.0
LC_3	3	10005	conm2Mass_2000	0.016	-636.120	359.050	749.970	967.620	1732.300	-429.870	-6.700	3.738	-0.001	-15705.0	-28115.0	6976.8	906.7	-385.8	-0.4
LC_4	4	10006	conm2Mass_1000	0.148	-610.710	-99.535	460.180	-182.480	-309.600	8346.000	-9.907	5.870	-0.002	27059.0	45907.0	-1237500.0	99404.0	-35227.0	-5652.1
LC_4	4	10005	conm2Mass_2000	0.016	-636.120	359.050	749.970	1519.400	2561.400	3952.000	-9.907	5.870	-0.002	-24660.0	-41571.0	-64141.0	1340.7	-605.6	-0.5

Identified by FatDyn-X from bulk-data



Bulk-data export (BDF) — snippet

```
GRID ,   10006,       0, -610.71,  -99.53,  460.18
GRID ,   10005,       0, -636.12,  359.05,  749.97
PLOTEL,       1,   10006,   10005
PLOTEL,       2,   10006,   10005
$ ================================================
$ Load Case    1
$ ================================================
$ Loads for mass: conm2Mass_1000
FORCE,        1,   10006,       0,  1129.095,    -0.012,     0.016,    -1.000
MOMENT,       1,   10006,       0,  37379.89,     0.92,      0.31,     -0.26
$ Loads for mass: conm2Mass_2000
FORCE,        1,   10005,       0,    96.175,     0.131,    -0.166,    -0.977
MOMENT,       1,   10005,       0,    597.90,     0.86,      0.51,     -0.00
$ ================================================
$ Load Case    2
$ ================================================
$ Loads for mass: conm2Mass_1000
FORCE,        2,   10006,       0,  2593.734,    -0.008,     0.019,    -1.000
MOMENT,       2,   10006,       0, 101556.04,     0.97,      0.14,     -0.21
$ Loads for mass: conm2Mass_2000
FORCE,        2,   10005,       0,   208.771,     0.095,    -0.213,    -0.972
MOMENT,       2,   10005,       0,   1509.71,     0.95,      0.32,     -0.00
$ ================================================
$ Load Case    3
$ ================================================
```

SECTION 10

Misc FE Operations

Automated load-case generation, IR-static unit cases, and bulk-data assembly.

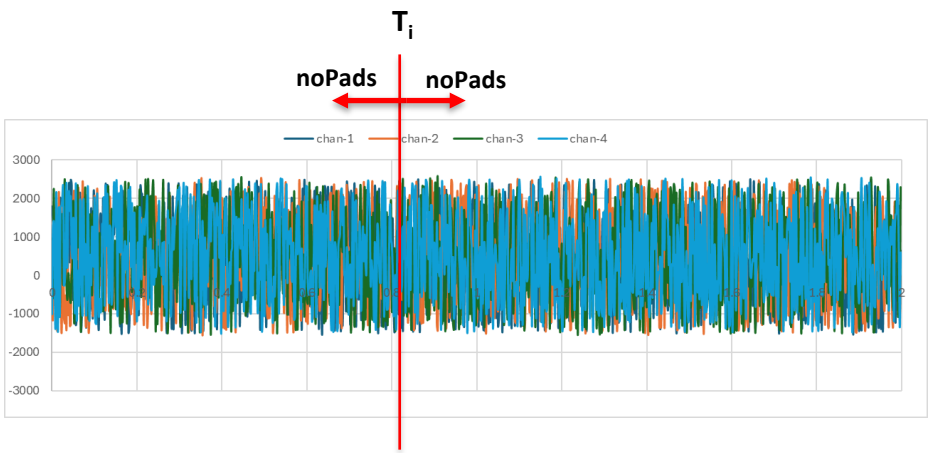
Inertia-Relief/Static and Dynamic Load Case Generation

Automated assembly of subcases and bulk-data for static, IR-static, and transient analyses.

```
Static_loadCases_byMinMax_Loads        : YES   1000   50.0        ! YES/NO  LC_startID   maxPercent
Static_loadCases_byTimeSet             : YES   tset1  5000        ! YES/NO  setLabel     LC_startID
Create_IRstatic_unitLoadCases          : YES                      ! YES/NO

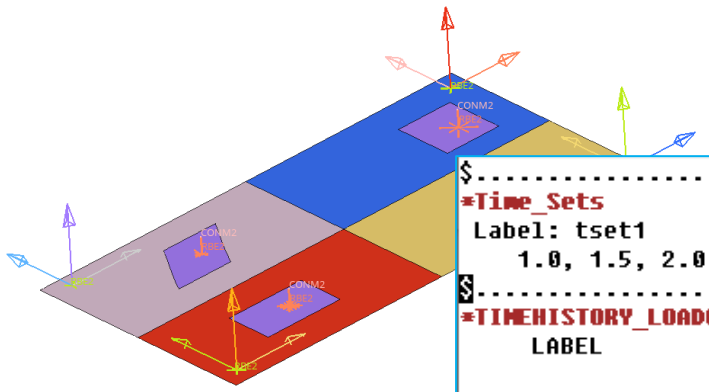
Transient_loadCases_byMinMax_Loads     : YES   1000   70.0  100   ! YES/NO  LC_startID   maxPercent   noPads
Transient_loadCases_byTimeSet          : YES  tset1   5000  100   ! YES/NO  setLabel     LC_startID   noPads
Create_Full_TransientDynamic_Loadcase  : YES                      ! YES/NO
Create_NormalMode_Loadcase             : YES                      ! YES/NO

Inertia_Relief_Acc_Contour             : YES                      ! f06/out file needed with rigid body acceleration output
Inertia_Relief_Mass_Force              : YES   conm2Set           ! f06/out file needed with rigid body acceleration output
```



Element_ID	elemDim	MaxStress_s1	MaxStressTime_s1	damage_s1	Life_s1	MaxStress_s2	MaxStressTime_s2	damage_s2	Life_s2	minLife
15431	2	187.283	1.5800	2.4886E-04	8037	156.5528	0.1800	2.1566E-04	9274	8037
6174	2	165.6149	2.0940	1.9167E-04	10435	166.467	2.9320	1.8307E-04	10925	10435
12083	2	145.8876	1.7940	1.2999E-04	15386	166.3814	1.1860	1.1735E-04	17043	15386
5659	2	174.635	2.6880	8.6861E-05	23025	145.6831	2.9320	8.8387E-05	22628	22628
15484	2	159.8341	2.3800	8.2717E-05	24179	148.5508	1.0900	7.5437E-05	26512	24179
6164	2	141.9651	2.9250	4.9398E-05	40488	145.6611	2.4400	4.7367E-05	42223	40488
15445	2	168.7709	1.5800	4.8200E-05	41493	138.3465	1.4840	4.2823E-05	46704	41493
7773	2	135.9763	2.7720	3.1597E-05	63297	138.2394	2.7240	2.5254E-05	79194	63297
12096	2	145.1344	2.9940	2.7398E-05	72999	131.9625	1.1860	2.5070E-05	79778	72999
15471	2	140.7902	2.4800	2.2143E-05	90322	133.6581	1.0900	2.0414E-05	97971	90322
7761	2	129.0513	1.7320	1.5426E-05	1.30E+05	148.941	2.2000	1.0886E-05	1.84E+05	1.2965E+05
5049	2	139.3726	2.7520	1.3981E-05	1.43E+05	132.1935	1.5800	1.3846E-05	1.44E+05	1.4305E+05
15472	2	122.9139	2.4040	1.2068E-05	1.66E+05	140.6832	2.9380	1.1463E-05	1.74E+05	1.6572E+05
12139	2	119.3989	1.6400	1.1513E-05	1.74E+05	142.8575	1.1860	1.2018E-05	1.66E+05	1.6641E+05
7642	2	134.4362	2.4940	1.1885E-05	1.68E+05	136.8097	2.7040	9.2381E-06	2.17E+05	1.6828E+05
6698	2	115.7272	2.7520	1.0225E-05	1.96E+05	151.6387	2.2000	9.6657E-06	2.07E+05	1.9561E+05
5649	2	133.3186	2.6880	7.6799E-06	2.60E+05	124.8411	2.3400	7.8868E-06	2.54E+05	2.5359E+05
12086	2	130.2954	1.7940	7.7514E-06	2.58E+05	126.9032	1.1860	6.6611E-06	3.00E+05	2.5802E+05
15439	2	121.152	2.2000	6.2295E-06	3.21E+05	128.5728	2.7520	4.5940E-06	4.35E+05	3.2105E+05

Automated Load Case Creation



- **LC_startID** : start ID of new load cases
- Filtering in MinMax load cases:
  - From the list of all time steps identified for min-max, duplicate time steps are eliminated (1st later of filtering)
  - **maxPercent**: In MinMax load cases, F-max (max value of all forces) and M-max (max value of all moments) are calculated. If any load case has its F-max and M-max less than the maxPercent of F-max and M-max, then it is neglected (2nd layer of filtering)
- **setLabel**: time set label
- **noPads**: no of time pads before and after time step at which transient load cases are generated

```
$-----------------------
*Time_Sets
 Label: tset1
    1.0, 1.5, 2.0
$-----------------------
*TIMEHISTORY_LOADCASES
     LABEL

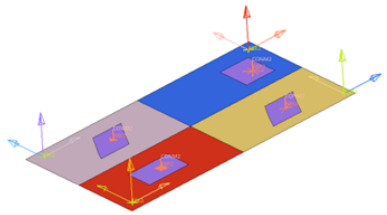
     ANALYSIS_TIME_STEP   :  0.002
$ SerialNo   Label          loadChannel_fileName    noHeadLines   COL_id   divider   scaleFactor   offset      node_id     dir   IRstat_LC
     1,       P1x,     LargePlateLoads_12LCs.csv,          1,        2,     5.00,       1.00,       0.0,        1,        1        1
     2,       P1y,     LargePlateLoads_12LCs.csv,          1,        3,     5.00,       1.00,       0.0,        1,        2        2
     3,       P1z,     LargePlateLoads_12LCs.csv,          1,        4,    10.00,       1.00,       0.0,        1,        3        3
!
     4,       P2x,     LargePlateLoads_12LCs.csv,          1,        5,     5.00,       1.00,       0.0,        2,        1        4
     5,       P2y,     LargePlateLoads_12LCs.csv,          1,        6,     5.00,       1.00,       0.0,        2,        2        5
     6,       P2z,     LargePlateLoads_12LCs.csv,          1,        7,    10.00,       1.00,       0.0,        2,        3        6
!
     7,       P3x,     LargePlateLoads_12LCs.csv,          1,        8,     5.00,       1.00,       0.0,        3,        1        7
     8,       P3y,     LargePlateLoads_12LCs.csv,          1,        9,     5.00,       1.00,       0.0,        3,        2        8
     9,       P3z,     LargePlateLoads_12LCs.csv,          1,       10,    10.00,       1.00,       0.0,        3,        3        9
!
    10,       P4x,     LargePlateLoads_12LCs.csv,          1,       11,     5.00,       1.00,       0.0,        4,        1       10
    11,       P4y,     LargePlateLoads_12LCs.csv,          1,       12,     5.00,       1.00,       0.0,        4,        2       11
    12,       P4z,     LargePlateLoads_12LCs.csv,          1,       13,    10.00,       1.00,       0.0,        4,        3       12
!

  Static_loadCases_byMinMax_Loads   : YES   1000   50.0          ! YES/NO  LC_startID   maxPercent
  Static_loadCases_byTimeSet        : YES   tset1  5000          ! YES/NO  setLabel     LC_startID
  Create_IRstatic_unitLoadCases     : YES                        ! YES/NO

  Transient_loadCases_byMinMax_Loads: YES   1000   70.0  100     ! YES/NO  LC_startID   maxPercent   noPads
  Transient_loadCases_byTimeSet     : YES  tset1   5000  100     ! YES/NO  setLabel     LC_startID   noPads
  Create_TransientDynamic_Loadcase  : YES                        ! YES/NO
  Create_NormalMode_Loadcase        : YES                        ! YES/NO
$-------------------------------------------------------------------------------------------------------------------------------
*END
```

Divide-scale-offset are used in load cases generated

Ex-19 — Static Load Cases by Min-Max Loads



```
Static_loadCases_byMinMax_Loads    : YES    1000    50.0
Static_loadCases_byTimeSet         : YES    tset1   5000
Create_IRstatic_unitLoadCases      : YES

Transient_loadCases_byMinMax_Loads: YES    1000    70.0   100
Transient_loadCases_byTimeSet      : YES   tset1   5000   100
Create_TransientDynamic_Loadcase   : YES
Create_NormalMode_Loadcase         : YES
```

FatDyn-X automatically generates bulk-formatted static load cases using **min/max values** from each time-history channel. It produces two CSV summaries:

- One showing min/max values and their locations per channel
- Another summarizing overall max values for each load case

This allows efficient creation of **hundreds of subcases and load entries**, ready for direct use in commercial solvers via BDF or input decks.

Note: FatDyn-X applies two smart filters:

- **Duplicate removal** to eliminate repeated min/max time steps
- **Magnitude thresholding** to discard insignificant load cases

Load cases generated in bulk format

```
$------------------
SUBCASE      1000
  Label         Loc_46
  ANALYSIS STATICS
  Load =    1000
$------------------
SUBCASE      1001
  Label         Loc_59
  ANALYSIS STATICS
  Load =    1001
$------------------
SUBCASE      1002
  Label         Loc_67
  ANALYSIS STATICS
  Load =    1002
$------------------
SUBCASE      1003
  Label         Loc_111
  ANALYSIS STATICS
```

```
$------------------
SUBCASE      1023
  Label       Loc_1499
  ANALYSIS STATICS
  Load =    1023
$=============================================
BEGIN BULK
PARAM, INREL, -2
$ ---------------------------------------------------------------------
$Force-Moment cards include scale-divide-offset defined in inp-file
$Force-Moment = (ScaleFactor/Divider)(TimeHistoryValue - Offset)
$ ---------------------------------------------------------------------
$Load case, time:         1       0.0900
Force ,     1000,        1, 1 ,       1.0,  0.14800E+02,  0.00000E+00,  0.00000E+00
Force ,     1000,        1, 1 ,       1.0,  0.00000E+00, -0.72000E+02,  0.00000E+00
Force ,     1000,        1, 1 ,       1.0,  0.00000E+00,  0.00000E+00,  0.20268E+02
Force ,     1000,        2, 1 ,       1.0, -0.20000E+03,  0.00000E+00,  0.00000E+00
Force ,     1000,        2, 1 ,       1.0,  0.00000E+00,  0.13160E+03,  0.00000E+00
Force ,     1000,        2, 1 ,       1.0,  0.00000E+00,  0.00000E+00,  0.98432E+02
Force ,     1000,        3, 1 ,       1.0,  0.78000E+01,  0.00000E+00,  0.00000E+00
Force ,     1000,        3, 1 ,       1.0,  0.00000E+00,  0.25800E+02,  0.00000E+00
Force ,     1000,        3, 1 ,       1.0,  0.00000E+00,  0.00000E+00,  0.13504E+02
Force ,     1000,        4, 1 ,       1.0, -0.11600E+02,  0.00000E+00,  0.00000E+00
Force ,     1000,        4, 1 ,       1.0,  0.00000E+00,  0.97600E+02,  0.00000E+00
Force ,     1000,        4, 1 ,       1.0,  0.00000E+00,  0.00000E+00,  0.46433E+02
$-------------------------------------------------
$Load case, time:         2       0.1160
Force ,     1001,        1, 1 ,       1.0, -0.25600E+02,  0.00000E+00,  0.00000E+00
Force ,     1001,        1, 1 ,       1.0,  0.00000E+00,  0.16040E+03,  0.00000E+00
Force ,     1001,        1, 1 ,       1.0,  0.00000E+00,  0.00000E+00,  0.78810E+01
Force ,     1001,        2, 1 ,       1.0, -0.11960E+03,  0.00000E+00,  0.00000E+00
```

Includes both subcases and loads in bulkdata format

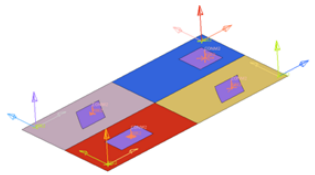
TH1_MISC_minmaxSummary.csv

Chan-id	minLoc	minVal	maxLoc	maxval
1	987	-200.00	810	199.00
2	898	-200.00	926	200.00
3	1464	-156.00	1467	154.00
4	46	-200.00	398	199.00
5	59	-200.00	1337	200.00
6	1499	-132.00	1271	132.00
7	483	-200.00	243	200.00
8	1358	-200.00	286	200.00
9	1492	-157.00	1426	137.00
10	67	-200.00	1374	200.00
11	361	-200.00	111	200.00
12	1203	-152.00	1470	162.00

TH1_MISC_maxSummary.csv

LC-id	TimeIndex	Fmax	Mmax
1	46	200.00	0.00
2	59	200.00	0.00
3	67	199.80	0.00
4	111	200.00	0.00
5	243	200.00	0.00
6	286	199.80	0.00
7	361	199.80	0.00
8	398	199.40	0.00
9	483	200.00	0.00
10	810	199.20	0.00
11	898	199.80	0.00
12	926	200.00	0.00
13	987	199.80	0.00
14	1203	192.20	0.00
15	1271	191.20	0.00
16	1337	199.80	0.00
17	1358	199.60	0.00
18	1374	200.00	0.00
19	1426	188.80	0.00
20	1464	193.80	0.00
21	1467	155.40	0.00
22	1470	162.23	0.00
23	1492	185.80	0.00
24	1499	182.80	0.00
Fmax_all	200.00		
Mmax_all	0.00		

Ex-20 — Static Load Cases for a Time Set



```
*Time_Sets
Label: tset1
   1.0, 1.5, 2.0
```

```
Static_loadCases_byMinMax_Loads    : YES   1000   50.0
Static_loadCases_byTimeSet         : YES   tset1  5000
Create_INstatic_UnitLoadCases      : YES
```

Load cases generated in bulk format

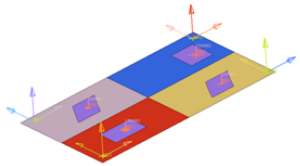
- **FatDyn-X offers a unique ability to generate static load cases directly from user-defined time sets:** For each selected time, FatDyn-X locates the exact time index, extracts corresponding loads, and auto-generates subcases and bulk data entries—eliminating manual effort.
- To support this, FatDyn-X creates a detailed fatigue summary CSV that pinpoints **critical time locations** for each element (e.g., when max stress occurs), enabling users to target meaningful time points for accurate stress evaluation.
- This feature is ideal for building targeted static cases from dynamic events—something not readily available in commercial solvers.

Element_ID	elemDim	MaxStress_s1	MaxStressTime_s1	damage_s1	Life_s1	MaxStress_s2	MaxStressTime_s2	damage_s2	Life_s2	minLife
15431	2	187.283	1.5800	2.4886E-04	8037	156.5528	0.1800	2.1566E-04	9274	8037
6174	2	165.6149	2.0940	1.9167E-04	10435	166.467	2.9320	1.8307E-04	10925	10435
12083	2	145.8876	1.7940	1.2999E-04	15386	166.3814	1.1860	1.1735E-04	17043	15386
5659	2	174.635	2.6880	8.6861E-05	23025	145.6831	2.9320	8.8387E-05	22628	22628
15484	2	159.8341	2.3800	8.2717E-05	24179	148.5508	1.0900	7.5437E-05	26512	24179
6164	2	141.9651	2.9260	4.9398E-05	40488	145.6611	2.4400	4.7367E-05	42223	40488
15445	2	168.7709	1.5800	4.8200E-05	41493	138.3465	1.4840	4.2823E-05	46704	41493
7773	2	135.9763	2.7720	3.1597E-05	63297	138.2394	2.7240	2.5254E-05	79194	63297
12096	2	145.1344	2.9940	2.7398E-05	72999	131.9625	1.1860	2.5070E-05	79778	72999
15471	2	140.7902	2.4800	2.2143E-05	90322	133.6581	1.0900	2.0414E-05	97971	90322
7761	2	129.0513	1.7320	1.5426E-05	1.30E+05	148.941	2.2000	1.0886E-05	1.84E+05	1.2965E+05
5049	2	139.3726	2.7520	1.3981E-05	1.43E+05	132.1935	1.5800	1.3846E-05	1.44E+05	1.4305E+05
15472	2	122.9139	2.4040	1.2068E-05	1.66E+05	140.6832	2.9380	1.1463E-05	1.74E+05	1.6572E+05
12139	2	119.3989	1.6400	1.1513E-05	1.74E+05	142.8575	1.1860	1.2018E-05	1.66E+05	1.6641E+05
7642	2	134.4362	2.4940	1.1885E-05	1.68E+05	136.8097	2.7040	9.2381E-06	2.17E+05	1.6828E+05
6698	2	115.7272	2.7520	1.0225E-05	1.96E+05	151.6387	2.2000	9.6657E-06	2.07E+05	1.9561E+05
5649	2	133.3186	2.6880	7.6799E-06	2.60E+05	124.8411	2.3400	7.8868E-06	2.54E+05	2.5359E+05
12086	2	130.2954	1.7940	7.7514E-06	2.58E+05	126.9032	1.1860	6.6611E-06	3.00E+05	2.5802E+05
15439	2	121.152	2.2000	6.2295E-06	3.21E+05	128.5728	2.7520	4.5940E-06	4.35E+05	3.2105E+05

```
$------------------
SUBCASE      5000
  Label        Loc_501     Index for time 1.0
  ANALYSIS STATICS
  Load =    5000
$------------------
SUBCASE      5001
  Label        Loc_751     Index for time 1.5
  ANALYSIS STATICS
  Load =    5001
$------------------
SUBCASE      5002
  Label        Loc_1001    Index for time 2.0
  ANALYSIS STATICS
  Load =    5002
$=============================================
BEGIN BULK
PARAM, INREL, -2
$ ---------------------------------------------------------------
$Force-Moment cards include scale-divide-offset defined in inp-file
$Force-Moment = (ScaleFactor/Divider)(TimeHistoryValue - Offset)
$ ---------------------------------------------------------------
$Load case, time:        1      1.0000
Force ,     5000,       1, 1 ,      1.0, -0.19080E+03,  0.00000E+00,  0.00000E+00
Force ,     5000,       1, 1 ,      1.0,  0.00000E+00,  0.14660E+03,  0.00000E+00
Force ,     5000,       1, 1 ,      1.0,  0.00000E+00,  0.00000E+00,  0.82575E+02
Force ,     5000,       2, 1 ,      1.0, -0.13900E+03,  0.00000E+00,  0.00000E+00
Force ,     5000,       2, 1 ,      1.0,  0.00000E+00, -0.15080E+03,  0.00000E+00
Force ,     5000,       2, 1 ,      1.0,  0.00000E+00,  0.00000E+00, -0.45143E+02
Force ,     5000,       3, 1 ,      1.0, -0.96600E+02,  0.00000E+00,  0.00000E+00
Force ,     5000,       3, 1 ,      1.0,  0.00000E+00,  0.10060E+03,  0.00000E+00
Force ,     5000,       3, 1 ,      1.0,  0.00000E+00,  0.00000E+00,  0.11850E+01
Force ,     5000,       4, 1 ,      1.0,  0.17320E+03,  0.00000E+00,  0.00000E+00
Force ,     5000,       4, 1 ,      1.0,  0.00000E+00,  0.89800E+02,  0.00000E+00
Force ,     5000,       4, 1 ,      1.0,  0.00000E+00,  0.00000E+00,  0.24500E+02
$---------------------------------------------
$Load case, time:        2      1.5000
Force ,     5001,       1, 1 ,      1.0,  0.47600E+02,  0.00000E+00,  0.00000E+00
Force ,     5001,       1, 1 ,      1.0,  0.00000E+00,  0.66000E+02,  0.00000E+00
Force ,     5001,       1, 1 ,      1.0,  0.00000E+00,  0.00000E+00, -0.42676E+02
Force ,     5001,       2, 1 ,      1.0, -0.61600E+02,  0.00000E+00,  0.00000E+00
Force ,     5001,       2, 1 ,      1.0,  0.00000E+00,  0.55600E+02,  0.00000E+00
```

Includes both subcases and loads in bulkdata format

Ex-21 — IR-static Unit Load Cases



```
Static_loadCases_byMinMax_Loads    : YES    1000    50.0
Static_loadCases_byTimeSet         : YES    tset1   5000
Create_IRstatic_unitLoadCases      : YES

Transient_loadCases_byMinMax_Loads: YES    1000    70.0   100
Transient_loadCases_byTimeSet      : YES   tset1    5000   100
Create_TransientDynamic_Loadcase   : YES
Create_NormalMode_Loadcase         : YES
```

Load cases generated in bulk format

```
*TIMEHISTORY_LOADCASES
    LABEL               : misc

    ANALYSIS_TIME_STEP  :  0.002
$ SerialNo   Label          loadChannel_fileName    noHeadLines   COL_id   divider   scaleFactor   offset        node_id    dir   IRstat_LC
    1,        P1x,     LargePlateLoads_12LCs.csv,          1,        2,     5.00,       1.00,       0.0,        1,      1        1
    2,        P1y,     LargePlateLoads_12LCs.csv,          1,        3,     5.00,       1.00,       0.0,        1,      2        2
    3,        P1z,     LargePlateLoads_12LCs.csv,          1,        4,    10.00,       1.00,       0.0,        1,      3        3
!
    4,        P2x,     LargePlateLoads_12LCs.csv,          1,        5,     5.00,       1.00,       0.0,        2,      1        4
    5,        P2y,     LargePlateLoads_12LCs.csv,          1,        6,     5.00,       1.00,       0.0,        2,      2        5
    6,        P2z,     LargePlateLoads_12LCs.csv,          1,        7,    10.00,       1.00,       0.0,        2,      3        6
!
    7,        P3x,     LargePlateLoads_12LCs.csv,          1,        8,     5.00,       1.00,       0.0,        3,      1        7
    8,        P3y,     LargePlateLoads_12LCs.csv,          1,        9,     5.00,       1.00,       0.0,        3,      2        8
    9,        P3z,     LargePlateLoads_12LCs.csv,          1,       10,    10.00,       1.00,       0.0,        3,      3        9
!
   10,        P4x,     LargePlateLoads_12LCs.csv,          1,       11,     5.00,       1.00,       0.0,        4,      1       10
   11,        P4y,     LargePlateLoads_12LCs.csv,          1,       12,     5.00,       1.00,       0.0,        4,      2       11
   12,        P4z,     LargePlateLoads_12LCs.csv,          1,       13,    10.00,       1.00,       0.0,        4,      3       12
!
```

- Applied unit load or moment at node-id along direction “dir” and a load case is generated with ID given (IRstatic_LC)

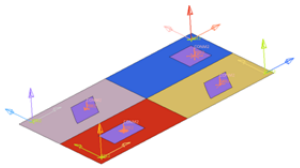
```
$------------------
SUBCASE         1
  Label    unitLoad_1_1
  ANALYSIS STATICS
  Load =        1
$------------------
SUBCASE         2
  Label    unitLoad_1_2
  ANALYSIS STATICS
  Load =        2
$------------------
```

Includes both subcases and loads in bulkdata format

```
$------------------
SUBCASE        11
  Label    unitLoad_4_2
  ANALYSIS STATICS
  Load =      11
$------------------
SUBCASE        12
  Label    unitLoad_4_3
  ANALYSIS STATICS
  Load =      12
$=============================================
BEGIN BULK
PARAM, INREL, -2
Force ,        1,       1, 1 ,      1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
Force ,        2,       1, 1 ,      1.0,  0.00000E+00,  0.10000E+01,  0.00000E+00
Force ,        3,       1, 1 ,      1.0,  0.00000E+00,  0.00000E+00,  0.10000E+01
Force ,        4,       2, 1 ,      1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
Force ,        5,       2, 1 ,      1.0,  0.00000E+00,  0.10000E+01,  0.00000E+00
Force ,        6,       2, 1 ,      1.0,  0.00000E+00,  0.00000E+00,  0.10000E+01
Force ,        7,       3, 1 ,      1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
Force ,        8,       3, 1 ,      1.0,  0.00000E+00,  0.10000E+01,  0.00000E+00
Force ,        9,       3, 1 ,      1.0,  0.00000E+00,  0.00000E+00,  0.10000E+01
Force ,       10,       4, 1 ,      1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
Force ,       11,       4, 1 ,      1.0,  0.00000E+00,  0.10000E+01,  0.00000E+00
Force ,       12,       4, 1 ,      1.0,  0.00000E+00,  0.00000E+00,  0.10000E+01
$=============================================
```

- **FatDyn-X automates the creation of IR-static unit load cases based on user-defined fatigue setup.**
  Using inputs like load label, node ID, direction, and desired load case ID, FatDyn-X generates corresponding subcases and bulk data entries—all in standard format.
- Each unit load or moment is applied at the specified node and direction, with the resulting cases ready for inertia relief analysis or superposition.
- This capability streamlines model setup, ensuring consistency and eliminating manual preprocessing.

Ex-22 — Transient Load Cases by Min-Max Loads



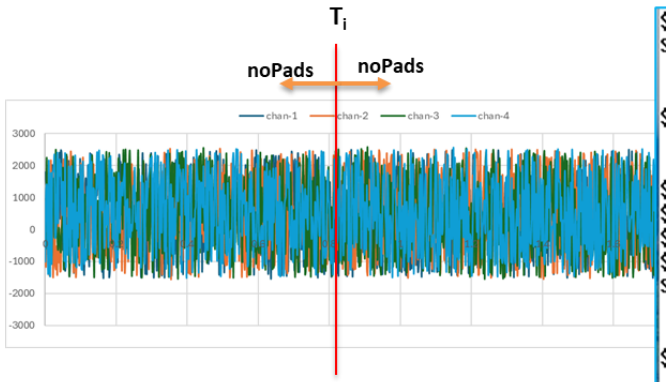
```
Transient_loadCases_byMinMax_Loads: YES   1000   70.0  100 ! YES/NO  LC_startID   maxPercent   noPads
Transient_loadCases_byTimeSet     : YES  tset1   5000  100 ! YES/NO  setLabel     LC_startID   noPads
```

- **FatDyn-X uniquely automates transient load case creation from filtered min-max loads.**
  It pinpoints key time locations, applies optional padding (noPads), and generates all required subcases and bulk data entries automatically.

Each transient load case includes:

- **FORCE, TLOAD1**, and **TABLED1** cards
- **Damping** and **USET1** cards to support either **direct** or **modal-transient** analysis

This workflow enables precise, high-fidelity transient simulations—without manual effort—based on actual fatigue inputs, a capability not offered by standard commercial FEA tools.



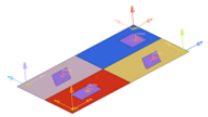

```
$-------------------
SUBCASE      1100
  Label       tIndx_46
  ANALYSIS DTRAN
$ ANALYSIS MTRAN
  DLOAD =    1100
  TSTEP =    1100
$---> data for modal transient analysis
$ Method   = 200
$ SDAMPING =  10000
$-------------------
SUBCASE      1200
  Label       tIndx_59
  ANALYSIS DTRAN
$ ANALYSIS MTRAN
  DLOAD =    1200
  TSTEP =    1200
$---> data for modal transient analysis
$ Method   = 200
$ SDAMPING =  10000
$-------------------
```

FORCE, TLOAD1, and TABLED1 cards for each load

```
$=============================================
BEGIN BULK
$ Damping definitions for direct transient analysis
$ param, G,  0.06: global material damping
$ param,W3, 100.0: to convert global material damping to viscous damping at Freq
$     W3 --- radians/unit-time
$ Param,  G,  0.06
$ Param, W3, 100.0
$ ------------------------------------------------------------------------------
$ Force-Moment time histories include scale-divide-offset transformations
$ Force-Moment = (ScaleFactor/Divider)(TimeHistoryValue - Offset)
$ Transient load cases with time padding and zero ramp-up for stability
$ ------------------------------------------------------------------------------
$ =======================================
$ cards needed if modal-transient is chosen
$PARAM, RESVEC,  YES
$EIGRL, 200,  ,  ,50
$USET1          U6       1       1
$USET1          U6       2       1
$USET1          U6       3       1
$USET1          U6       1       2
$USET1          U6       2       2
$USET1          U6       3       2
$USET1          U6       1       3
$USET1          U6       2       3
$USET1          U6       3       3
$USET1          U6       1       4
$USET1          U6       2       4
$USET1          U6       3       4
$TABDMP1    10000    CRIT
$+            0.0    0.05    10.0    0.05    20.0    0.05   100.0    0.05
$+          200.0    0.05   500.0    0.05  1000.0    0.05ENDT
$ =======================================
$Load case, time:         1      0.0900
FORCE ,    1101,       1, 1 ,      1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
TLOAD1,    1101,    1101, , ,    1101
TABLED1*              1101           LINEAR          LINEAR
*
*                  0.00000     0.00000E+00         0.00200     0.00000E+00
*                  0.00400     0.00000E+00         0.00600     0.00000E+00
*                  0.00800     0.00000E+00         0.01000     0.00000E+00
*                  0.01200     0.00000E+00         0.01400     0.00000E+00
*                  0.01600     0.00000E+00         0.01800     0.00000E+00
*                  0.02000     0.90800E+02         0.02200     0.71000E+02
*                  0.02400    -0.14740E+03         0.02600    -0.15100E+03
*                  0.02800     0.66600E+02         0.03000     0.65400E+02
*                  0.03200     0.17540E+03         0.03400     0.56000E+02
*                  0.03600     0.19520E+03         0.03800    -0.63600E+02
*                  0.04000    -0.61000E+02         0.04200    -0.18760E+03
*                  0.04400    -0.95000E+02         0.04600     0.15000E+02
```

Damping and USET1 cards for direct or modal transient

Ex-23 — Transient Load Cases by Time Set



```
*Time_Sets
Label: tset1
   1.0, 1.5, 2.0
```

```
Transient_loadCases_byMinMax_Loads: YES   1000   70.0   100 ! YES/NO   LC_startID   maxPercent   noPads
Transient_loadCases_byTimeSet     : YES   tset1   5000   100 ! YES/NO   setLabel     LC_startID   noPads
```

- **FatDyn-X enables automated generation of transient load cases using user-defined time sets:** For each specified time in the set, FatDyn-X locates the exact time index and builds the corresponding **subcases** and **bulk data** entries in standard format.
- Each load case includes:
  - **FORCE, TLOAD1, and TABLED1** cards – with loads applied exactly at the defined time, padded before/after as needed using noPads
  - **Damping** and **USET1** cards – fully compatible with both **direct** and **modal-transient** analysis
- This capability offers a powerful and efficient alternative to manual setup, enabling accurate transient simulations directly from meaningful time points—**a feature not available in most commercial solvers**.
- To support this process, FatDyn-X generates a fatigue summary CSV that highlights **critical time locations** (e.g., when max stress occurs), guiding users to select impactful time points for **targeted stress evaluation.**

Element_ID	elemDim	MaxStress_s1	MaxStressTime_s1	damage_s1	Life_s1	MaxStress_s2	MaxStressTime_s2	damage_s2	Life_s2	minLife
15431	2	187.283	1.5800	2.4886E-04	8037	156.5528	0.1800	2.1566E-04	9274	8037
6174	2	166.6149	2.0940	1.9167E-04	10435	166.467	2.9320	1.8307E-04	10925	10435
12083	2	145.8876	1.7940	1.2399E-04	15386	166.3814	1.1860	1.1735E-04	17043	15386
5659	2	174.635	2.6880	8.6861E-05	23025	145.6831	2.9320	8.8387E-05	22628	22628
15484	2	159.8341	2.3800	8.2717E-05	24179	148.5508	1.0900	7.5437E-05	26512	24179
6164	2	141.9651	2.9250	4.9388E-05	40488	145.6611	2.4400	4.7367E-05	42223	40488
15445	2	168.7709	1.5800	4.8200E-05	41493	138.3465	1.4840	4.2823E-05	46704	41493
7773	2	135.9763	2.7720	3.1597E-05	63297	138.2394	2.7240	2.5254E-05	79194	63297
12096	2	145.1344	2.9940	2.7398E-05	72999	131.9625	1.1860	2.5070E-05	79778	72999
15471	2	140.7902	2.4800	2.2143E-05	90322	133.6581	1.0900	2.0414E-05	97971	90322
7761	2	129.0513	1.7320	1.5426E-05	1.30E+05	148.941	2.2000	1.0888E-05	1.84E+05	1.2965E+05
5049	2	139.3726	2.7520	1.3861E-05	1.43E+05	132.1935	1.5800	1.3846E-05	1.44E+05	1.4305E+05
15472	2	122.9139	2.4040	1.2068E-05	1.66E+05	140.6832	2.9380	1.1463E-05	1.74E+05	1.6572E+05
12139	2	119.3989	1.6400	1.1513E-05	1.74E+05	142.8575	1.1860	1.2018E-05	1.66E+05	1.6641E+05
7642	2	134.4362	2.4940	1.1885E-05	1.68E+05	136.8097	2.7040	9.2381E-06	2.17E+05	1.6828E+05
6698	2	115.7272	2.7520	1.0225E-05	1.96E+05	151.6387	2.2000	9.6657E-06	2.07E+05	1.9561E+05
5649	2	133.3186	2.6880	7.6799E-06	2.60E+05	124.8411	2.3400	7.8868E-06	2.54E+05	2.5359E+05
12086	2	130.2954	1.7940	7.7514E-06	2.58E+05	126.9032	1.1860	6.6611E-06	3.00E+05	2.5802E+05
15439	2	121.152	2.2000	6.2295E-06	3.21E+05	128.5728	2.7520	4.5940E-06	4.35E+05	3.2105E+05

```
$-------------------
SUBCASE     5100
  Label      tIndx_501
  ANALYSIS DTRAN
$ ANALYSIS MTRAN
  DLOAD =    5100
  TSTEP =    5100
$---> data for modal transient analysis
$ Method  = 200
$ SDAMPING =  10000
$-------------------
SUBCASE     5200
  Label      tIndx_751
  ANALYSIS DTRAN
$ ANALYSIS MTRAN
  DLOAD =    5200
  TSTEP =    5200
$---> data for modal transient analysis
$ Method  = 200
$ SDAMPING =  10000
```

Index for time 1.0

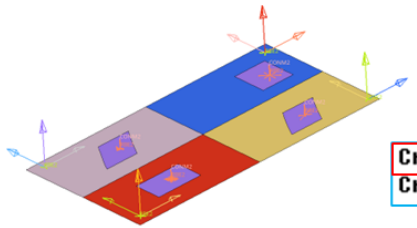
Load cases generated in bulk format

FORCE, TLOAD1, and TABLED1 cards for each load

```
$=============================================
BEGIN BULK
$ Damping definitions for direct transient analysis
$ param, G,  0.06: global material damping
$ param,W3, 100.0: to convert global material damping to viscous damping at Freq
$     W3 --- radians/unit-time
$ Param,  G,  0.06
$ Param, W3, 100.0
$ ------------------------------------------------------------------------
$ Force-Moment time histories include scale-divide-offset transformations
$ Force-Moment = (ScaleFactor/Divider)(TimeHistoryValue - Offset)
$ Transient load cases with time padding and zero ramp-up for stability
$ ------------------------------------------------------------------------
$ ======================================
$ cards needed if modal-transient is chosen
$PARAM, RESVEC,  YES
$EIGRL, 200,  ,  ,50
$USET1          U6       1       1
$USET1          U6       2       1
$USET1          U6       3       1
$USET1          U6       1       2
$USET1          U6       2       2
$USET1          U6       3       2
$USET1          U6       1       3
$USET1          U6       2       3
$USET1          U6       3       3
$USET1          U6       1       4
$USET1          U6       2       4
$USET1          U6       3       4
$TABDMP1    10000    CRIT
$+            0.0     0.05    10.0     0.05    20.0     0.05   100.0     0.05
$+          200.0     0.05   500.0     0.05  1000.0     0.05ENDT
$ ======================================
$Load case,  me:        1      1.0000
                      1, 1 ,      1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
                   5101, , ,    5101
                   5101          LINEAR          LINEAR
*                 0.00000     0.00000E+00         0.00200     0.00000E+00
*                 0.00400     0.00000E+00         0.00600     0.00000E+00
*                 0.00800     0.00000E+00         0.01000     0.00000E+00
*                 0.01200     0.00000E+00         0.01400     0.00000E+00
*                 0.01600     0.00000E+00         0.01800     0.00000E+00
*                 0.02000    -0.73600E+02         0.02200    -0.14720E+03
*                 0.02400    -0.10400E+03         0.02600     0.11640E+03
*                 0.02800    -0.12740E+03         0.03000    -0.12220E+03
*                 0.03200    -0.71000E+02         0.03400     0.16880E+03
*                 0.03600    -0.18200E+02         0.03800    -0.44600E+02
*                 0.04000     0.14180E+03         0.04200     0.38400E+02
*                 0.04400    -0.17700E+03         0.04600     0.68600E+02
*                 0.04800    -0.58600E+02         0.05000    -0.18780E+03
*                 0.05200     0.96200E+02         0.05400    -0.33400E+02
*                 0.05600     0.26000E+02         0.05800    -0.28200E+02
```

Damping and USET1 cards for direct or modal transient

Ex-24 — Full Transient Load Case



```
Create_Full_TransientDynamic_Loadcase   : YES    ! YES/NO
Create_NormalMode_Loadcase              : YES    ! YES/NO
```

- **Transient dynamic load case** is generated for full time history.
- For each load case, **subcase** and **bulkdata** are created.
- **Optional cards** are provided to switch between direct or modal-transient load case.
- **Damping** and **USET1 cards** for either direct or modal transient.
- **FORCE**, **TLOAD1**, and **TABLED1 cards** for each load.
- **TableD1** for complete time history incorporating Divide-Scale-Offset, unit load applied at node-id along “dir” is used in TLOAD1 card.

```
$.....................................................................................................................
*Time_Sets
 Label: tset1
    1.0, 1.5, 2.0
$.....................................................................................................................
*TIMEHISTORY_LOADCASES
     LABEL              : misc

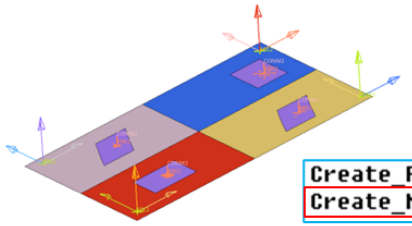
     ANALYSIS_TIME_STEP : 0.002
$ SerialNo   Label        loadChannel_fileName     noHeadLines   COL_id   divider   scaleFactor   offset      node_id    dir   IRstat_LC
     1,        P1x,    LargePlateLoads_12LCs.csv,        1,        2,     5.00,       1.00,       0.0,        1,       1       1
     2,        P1y,    LargePlateLoads_12LCs.csv,        1,        3,     5.00,       1.00,       0.0,        1,       2       2
     3,        P1z,    LargePlateLoads_12LCs.csv,        1,        4,    10.00,       1.00,       0.0,        1,       3       3
!
     4,        P2x,    LargePlateLoads_12LCs.csv,        1,        5,     5.00,       1.00,       0.0,        2,       1       4
     5,        P2y,    LargePlateLoads_12LCs.csv,        1,        6,     5.00,       1.00,       0.0,        2,       2       5
     6,        P2z,    LargePlateLoads_12LCs.csv,        1,        7,    10.00,       1.00,       0.0,        2,       3       6
!
     7,        P3x,    LargePlateLoads_12LCs.csv,        1,        8,     5.00,       1.00,       0.0,        3,       1       7
     8,        P3y,    LargePlateLoads_12LCs.csv,        1,        9,     5.00,       1.00,       0.0,        3,       2       8
     9,        P3z,    LargePlateLoads_12LCs.csv,        1,       10,    10.00,       1.00,       0.0,        3,       3       9
!
    10,        P4x,    LargePlateLoads_12LCs.csv,        1,       11,     5.00,       1.00,       0.0,        4,       1      10
    11,        P4y,    LargePlateLoads_12LCs.csv,        1,       12,     5.00,       1.00,       0.0,        4,       2      11
    12,        P4z,    LargePlateLoads_12LCs.csv,        1,       13,    10.00,       1.00,       0.0,        4,       3      12
!
```

```
$ ------------------------------------------------------------------------
$ Force-Moment time histories include scale-divide-offset transformations
$ Force-Moment = (ScaleFactor/Divider)(TimeHistoryValue - Offset)
$ Transient load cases with time padding and zero ramp-up for stability
$ ------------------------------------------------------------------------
$--------------------
SUBCASE    10000
  Label       fullTran
  ANALYSIS DTRAN
$ ANALYSIS MTRAN
  DLOAD =  10000
  TSTEP =  10000
$---> data for modal transient analysis
$ Method  = 200
$ SDAMPING =  10000
$=============================================
BEGIN BULK
$ Damping definitions for direct transient analysis
$ param, G,  0.06: global material damping
$ param,W3, 100.0: to convert global material damping to viscous damping at Freq W3
$     W3 --- radians/unit-time
$ Param,  G,  0.06
$ Param, W3, 100.0
$ ======================================
$ cards needed if modal-transient is chosen
$PARAM, RESVEC,  YES
$EIGRL, 200,  ,  ,50
$USET1          U6       1       1
$USET1          U6       2       1
$USET1          U6       3       1
$USET1          U6       1       2
$USET1          U6       2       2
$USET1          U6       3       2
$USET1          U6       1       3
$USET1          U6       2       3
$USET1          U6       3       3
$USET1          U6       1       4
$USET1          U6       2       4
$USET1          U6       3       4
$TABDMP1    10000    CRIT
$+            0.0    0.05    10.0    0.05    20.0    0.05   100.0    0.05
$+          200.0    0.05   500.0    0.05  1000.0    0.05ENDT
$ ======================================
FORCE ,    10001,       1,                1.0,  0.10000E+01,  0.00000E+00,  0.00000E+00
TLOAD1,    10001,      10                0001
TABLED1*                                LINEAR          LINEAR
*
*                                    0.90800E+02         0.00200     0.71000E+02
*                                   -0.14740E+03         0.00600    -0.15100E+03
*                                    0.66600E+02         0.01000     0.65400E+02
*                                    0.17540E+03         0.01400     0.56000E+02
*                                    0.19520E+03         0.01800    -0.63600E+02
*                                   -0.61000E+02         0.02200    -0.18760E+03
*                                   -0.95000E+02         0.02600     0.15000E+02
*                                    0.14980E+03         0.03000     0.68000E+02
*                                    0.50000E+02         0.03400     0.14460E+03
*                                   -0.42600E+02         0.03800     0.17840E+03
```

Damping and USET1 cards for direct or modal transient

FORCE, TLOAD1, and TABLED1 cards for each load

Ex-25 — Normal Mode Load Case



```
Create_Full_TransientDynamic_Loadcase  : YES   ! YES/NO
Create_NormalMode_Loadcase             : YES   ! YES/NO
```

- **FatDyn-X automatically generates a normal mode load case**, along with the required **USET1 cards**.
- **USET1 cards** are created for each **node-direction pair** where a load is applied.
- These cards generate **residual static modes**, which are essential for accurate fatigue and dynamic analysis—especially in modal superposition methods.

Load cases generated in bulk format

```
$-------------------------------------------------------------------------------------------------------------------------------
*Time_Sets
 Label: tset1
    1.0, 1.5, 2.0
$-------------------------------------------------------------------------------------------------------------------------------
*TIMEHISTORY_LOADCASES
     LABEL              : misc

     ANALYSIS_TIME_STEP  :  0.002
$ SerialNo   Label          loadChannel_fileName    noHeadLines   COL_id   divider   scaleFactor   offset        node_id    dir   IRstat_LC
     1,        P1x,     LargePlateLoads_12LCs.csv,          1,        2,     5.00,       1.00,        0.0,        1,      1       1
     2,        P1y,     LargePlateLoads_12LCs.csv,          1,        3,     5.00,       1.00,        0.0,        1,      2       2
     3,        P1z,     LargePlateLoads_12LCs.csv,          1,        4,    10.00,       1.00,        0.0,        1,      3       3
!
     4,        P2x,     LargePlateLoads_12LCs.csv,          1,        5,     5.00,       1.00,        0.0,        2,      1       4
     5,        P2y,     LargePlateLoads_12LCs.csv,          1,        6,     5.00,       1.00,        0.0,        2,      2       5
     6,        P2z,     LargePlateLoads_12LCs.csv,          1,        7,    10.00,       1.00,        0.0,        2,      3       6
!
     7,        P3x,     LargePlateLoads_12LCs.csv,          1,        8,     5.00,       1.00,        0.0,        3,      1       7
     8,        P3y,     LargePlateLoads_12LCs.csv,          1,        9,     5.00,       1.00,        0.0,        3,      2       8
     9,        P3z,     LargePlateLoads_12LCs.csv,          1,       10,    10.00,       1.00,        0.0,        3,      3       9
!
    10,        P4x,     LargePlateLoads_12LCs.csv,          1,       11,     5.00,       1.00,        0.0,        4,      1      10
    11,        P4y,     LargePlateLoads_12LCs.csv,          1,       12,     5.00,       1.00,        0.0,        4,      2      11
    12,        P4z,     LargePlateLoads_12LCs.csv,          1,       13,    10.00,       1.00,        0.0,        4,      3      12
!

     Static_loadCases_byMinMax_Loads   : YES   1000   50.0            ! YES/NO  LC_startID   maxPercent
     Static_loadCases_byTimeSet        : YES   tset1  5000            ! YES/NO  setLabel     LC_startID
     Create_IRstatic_unitLoadCases     : YES                          ! YES/NO

     Transient_loadCases_byMinMax_Loads: YES   1000   70.0  100       ! YES/NO  LC_startID   maxPercent   noPads
     Transient_loadCases_byTimeSet     : YES  tset1   5000  100       ! YES/NO  setLabel     LC_startID   noPads
     Create_TransientDynamic_Loadcase  : YES                          ! YES/NO
     Create_NormalMode_Loadcase        : YES                          ! YES/NO
$-------------------------------------------------------------------------------------------------------------------------------
*END
```

```
$=================================================
Subcase 999
  Analysis Modes
  Method(Structure) = 888
  Resvec(uniload)   = Yes
$=================================================
BEGIN BULK
EIGRL, 888,    ,    ,50
$ USET1 cards to generate residual vectors
USET1, U6,   1,         1
USET1, U6,   2,         1
USET1, U6,   3,         1
USET1, U6,   1,         2
USET1, U6,   2,         2
USET1, U6,   3,         2
USET1, U6,   1,         3
USET1, U6,   2,         3
USET1, U6,   3,         3
USET1, U6,   1,         4
USET1, U6,   2,         4
USET1, U6,   3,         4
$=================================================
```

USET1 cards to create residual static modes which are important for fatigue and dynamics

For each Node-id/direction pair, residual vectors are generated using USET1 cards

SECTION 11

Fatigue Prediction — Other Software vs FatDyn-X

Fatigue Prediction — Other Software vs FatDyn-X

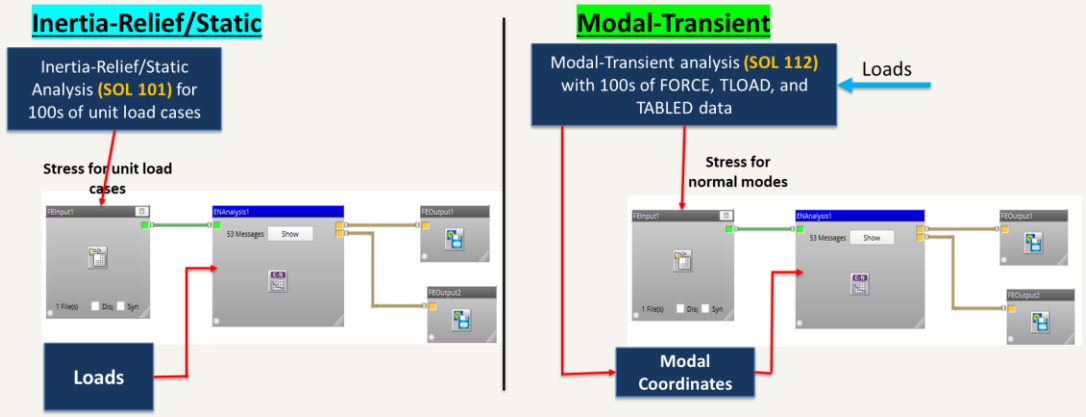
Other software — today

Inertia-Relief / Static

- SOL 101 with 100s of unit load cases
- Stress recovered for unit load cases
- Load time histories combined externally

Modal Transient

- SOL 112 with 100s of FORCE / TLOAD / TABLED cards
- Modal stress for normal modes
- Modal coordinates produced separately

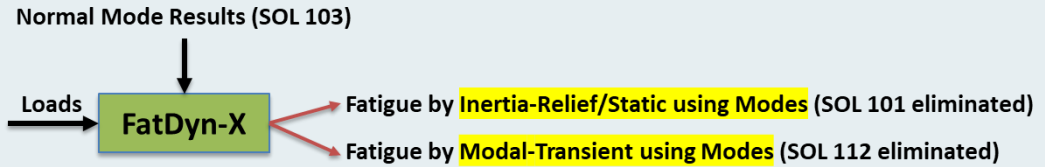


FatDyn-X

Single normal-modes analysis (SOL 103)

- Fatigue by IR-Static using modes — SOL 101 eliminated
- Fatigue by Modal Transient using modes — SOL 112 eliminated
- Loads applied directly inside FatDyn-X
- Modal stress and modal coordinates computed in one tool

Same modal stress is reused across both quasi-static and dynamic events.



FatDyn-X Workflow — At a Glance

Inputs

IR-Static (SOL 101)

Outputs

Fatigue and Dynamics

FatDyn-X

Inputs

Normal Modes (SOL 103)

Solvers

IR-Static
IR-Static via modes
Modal Transient
Modal Resonance

Outputs

Fatigue and Dynamics

Can Eliminate

SOL 101 Inertia-Relief/Static and
SOL 112 Modal-Transient analyses

A single normal-modes analysis covers fatigue and dynamics for both quasi-static and dynamic events.

SECTION 12

FatDyn-X to nCode & Other Fatigue Software

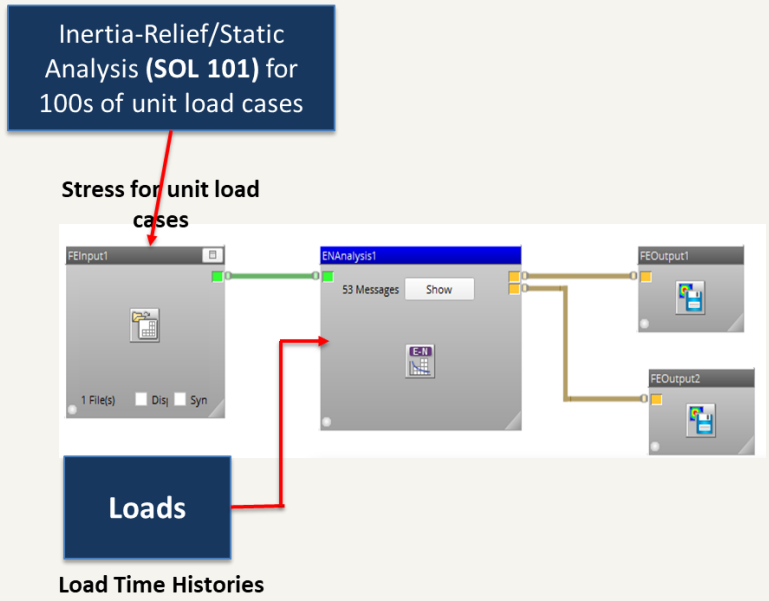
Replace SOL 101 Inertia-Relief/Static and SOL 112 modal-transient with modal coordinates from a single normal-mode run.

TODAY

Fatigue Prediction by nCode & Other Software — Current Methods

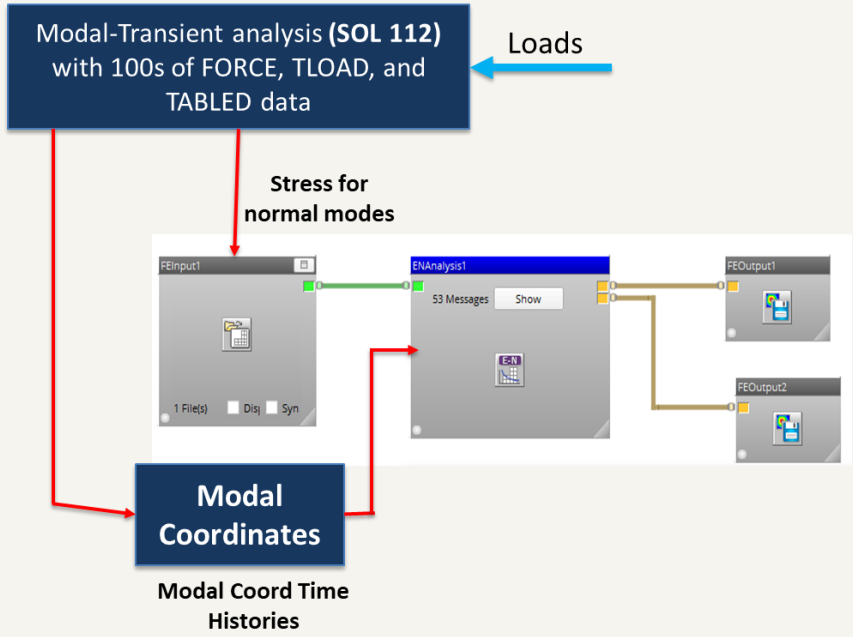
Inertia-Relief / Static

- Inertia-Relief / Static Analysis (SOL 101) for hundreds of unit load cases.
- Stress for unit load cases is exported.
- Load time histories drive duty-cycle stress reconstruction in the fatigue solver.



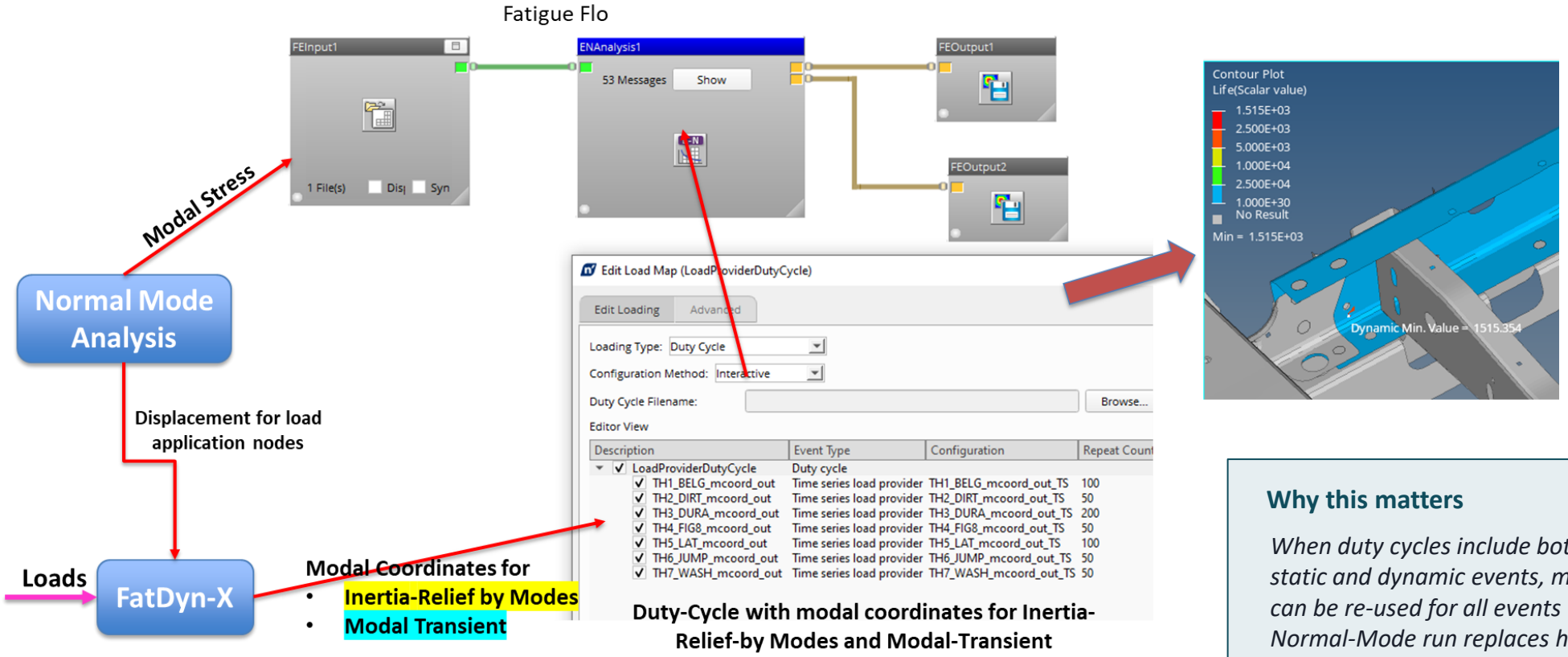
Modal-Transient

- Modal-Transient analysis (SOL 112) with hundreds of FORCE, TLOAD, and TABLED entries.
- Stress for normal modes is exported.
- Modal coordinate time histories drive the fatigue solver.



FatDyn-X for nCode & Other Fatigue Software

- FatDyn-X generates modal coordinates from a single Normal Mode run, in PCH and CSV formats.
- Inertia-Relief Analysis (hundreds of load cases, SOL 101) → **eliminated**
- Modal-Transient Analysis (SOL 112 with TABLED1 and TLOAD entries) → **eliminated**



Why this matters

When duty cycles include both quasi-static and dynamic events, modal-stress can be re-used for all events — a single Normal-Mode run replaces hundreds of solver runs.

Frequency Response Analysis and PSD-Fatigue

FatDyn-X

(Fatigue & Dynamic Analysis for Frequency Domain Loads)

Frequency Response Analysis

Load (force) input

- **Modal frequency response from normal modes (SOL 103)**
- Force / load excitation — the only FR input path
- Modal damping applied per mode
- Complex transfer functions H(f): magnitude and phase
- Node response: displacement, velocity, acceleration
- Spring / rod / beam / bush element forces vs frequency
- Shell and solid element stress vs frequency
- Per-node and per-element results written to CSV
- Consolidated stress-FRF text scan for large element sets

Same transfer functions the PSD load-path builds internally — frequency response is the foundation for random-vibration fatigue.

PSD Response and Fatigue Analysis

Two input paths — each with its own strategy

Load input — via Normal Modes

- FRF built internally from normal modes — no separate FR run
- FR frequency grid auto-derived from the input-PSD band
- PSD fatigue auto-launches when the FR completes

Acceleration (base-excitation) input

- Two-step workflow driven by the result PCH:
- Step 1 — no PCH: writes a runnable Nastran / OptiStruct deck, then stops
- Step 2 — with PCH: unpacks PCH to text, then computes PSD fatigue

PSD Fatigue — Dirlik spectral method (shared by both paths)

- Spectral moments m0, m1, m2, m4 by composite Simpson integration; RMS, irregularity factor, peak rate
- Dirlik damage with log-log PSD interpolation; AbsMaxPr critical-plane metric for shells and solids
- Fatigue elements selected by material mapping — independent of the stress-response output set
- Per-element damage and life → CSV and HWascii, plus optional consolidated stress-FRF scan

FatDyn-X

(Fatigue & Dynamic Analysis for Frequency Domain Loads — continued)

NVH analysis using only normal modes

- **Frequency response analysis** delivers precise results at every frequency. Node, force, and stress frequency responses match commercial FEA results exactly.
- The keyword-driven input structure is straightforward and user-friendly. **Leveraging normal modes alone makes NVH analysis effortless**—removing the need for the complicated input cards required by commercial FEA software.

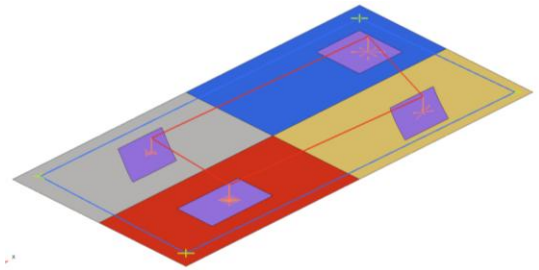
```
*FE_DATA
            file_modes_Res: LargePlate_modes.pch
               Rigid_Modes:  1  to   6
                     Modes:  7  to  50   critical_damping = 0.05
              Static_modes: 51  to  62

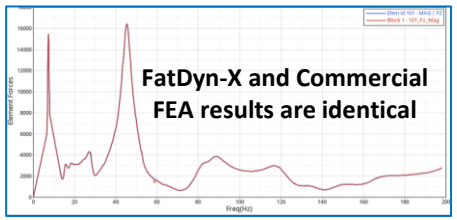
*FREQUENCY_LOADCASES
     node_FreqResp    : YES  nodeResp
     Force_FreqResp   : YES  forceElems
     Stress_FreqResp  : YES  stressElems

     FREQ1  1.0,   1.0, 100
     FREQ4  0.0, 200.0,  0.1,  11

!=========================================================
! Individual load cases  --- minimum 1 load case needed ---
!=========================================================
Individual_Loads: bodyResp
!  LC_id       Lab      loadFileName    noHeadRows   Col-id    Scale-Factor    node-id      dir
     1        nd1_Z     freqLoad.csv        1          2         1000.0          1          3
     2        nd2_Z     freqLoad.csv        1          3         1000.0          2          3
     3        nd3_Z     freqLoad.csv        1          4         1000.0          3          3
     4        nd4_Z     freqLoad.csv        1          5         1000.0          4          3

!=============================================================================================
!  ---- OPTIONAL ----   Combination Load cases: Linear combination of individual load cases
!=============================================================================================
Combination_Load : twist
!  LC_id       scale
    1           1.0
    2          -1.0
    3           1.0
    4          -1.0
```





PSD fatigue using only normal modes

- **Normal modes + PSD → fatigue life.** No frequency response analysis in commercial FEA needed — FatDyn-X forms the transfer functions internally.
- **Matches full time-history fatigue** — same hotspots and ***closer to time-history than the commercial benchmark***
- Simple keyword-based input, uses only normal mode results for load-PSD; creates runnable Nastran / OptiStruct for acceleration input
- **FatDyn-X simplifies a complex process**

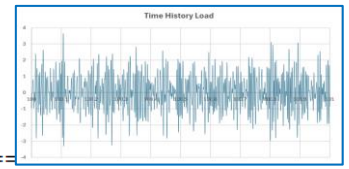
```
*FE_DATA
        file_modes_Res: timeHist_Psd_fat_0p5mm.pch
                 Modes:  1  to  50  critical_damping = 0.03
          Static_modes: 51  to  51
$=================================================================
*PSD_LoadCases
     Fatigue_psd: YES

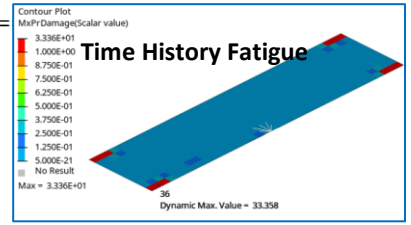
individual_psdLoads: loadPsd
!  LC_id    Lab      psdFileName   noHeadRows   Col-id   Scale-Factor   psd-type   node-id   dir
     1      ev1      psd_ev1.csv       1           2         1.0           Load       30       3
     2      ev2      psd_ev2.csv       1           2         1.0           Load       30       3

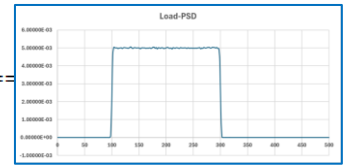
!=============================================================
!  ---- OPTIONAL ----  Duty-Cycle for individual psd loads
!=============================================================
psd_dutyCycle: shk
!  LC_id        Repeats
    1             100.0
    2             300.0

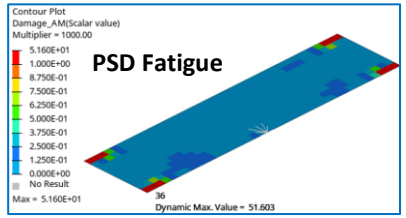
$================================
*Fatigue_Material_Mapping
   SN_MEAN_CORRECTION: NON
   map_ALL_to_FatMID : YES  1
$================================
*Fatigue_Material_Data_File
   snenLife.txt

*END
```









Standalone tools of FatDyn-X

Utilities that build test PSDs, convert any signal to a PSD, and verify results — usable on their own.

FatDyn-X

(Accelerated Fatigue Test using PSD Loads)

Why accelerated PSD testing

- **Time-history fatigue** is precise for many-channel loads — but only at low-to-mid frequency, where cycle counting is meaningful.
- **Mid-to-high-frequency parts** (electronics, battery, motor, engine) are proven on shaker tests: field time histories become **PSD energy**, and many events combine into **one PSD**. Days of duty cycle compress to a **few-hour, single-PSD test** — with a method that is easy to run and easy to prove.

Standalone tools — build the test PSD

(1) csv_to_psd.exe → PSD from any acceleration / load signal

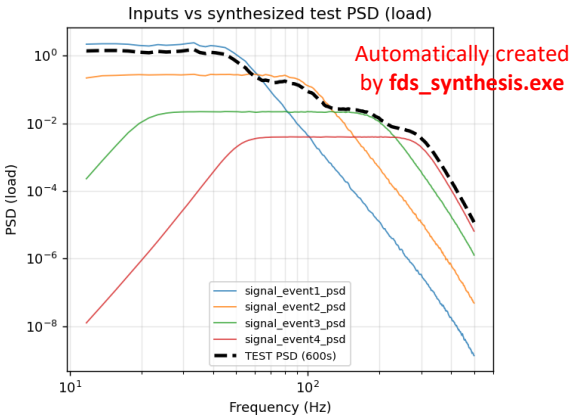
(2) fds_synthesis.exe → combine event PSDs into one test PSD

eventPSDs.inp

```
type   load

psd-file                    event-time
signal_event1_psd.csv         120.0
signal_event2_psd.csv         150.0
signal_event3_psd.csv        1200.0
signal_event4_psd.csv        5000.0
```

Single test PSD



Acceleration & mass-force time histories using FatDyn-X

- Field acceleration is usually measured with **accelerometers**. When it is not, **FatDyn-X computes accurate acceleration and mass-force time histories** from normal modes plus the input load time histories.

```
*FE_DATA
       file_bulkData        : LargePlate_modes.fem
       file_Modes_res       : LargePlate_modes.pch  ! pch, op2, or h3d format
              Rigid_Modes   :  1  to   6  !====>  used only for acceleration and mass forces
                    Modes   :  7  to  50   critical_damping = 0.05
             Static_modes   : 51  to  62

$.....................................................................................................
*TIMEHISTORY_LOADCASES
!==================== Event 1  ===================================================================
     LABEL          : mt_ev1
     ANALYSIS_METHOD: modal_transient

     node_timeHistory     : YES  nodeResp  ! disp-velo-acce time history for a node-set
     massForce_timeHistory: YES  conm2Set  ! force time history for a mass element set


ANALYSIS_TIME_STEP:  0.002
$ SerialNo   Label     loadChannel_fileName     noHeadRows     COL_id         divider    scaleFactor    offset        node_id     dir    IRstat_LC
     1,       P1x,     LoadDate_event1.csv,          1,          2,          1.00,       1.00,        0.0,          1,        1        1
     2,       P1y,     LoadDate_event1.csv,          1,          3,          1.00,       1.00,        0.0,          1,        2        2
     3,       P1z,     LoadDate_event1.csv,          1,          4,          1.00,       1.00,        0.0,          1,        3        3
     .......
```

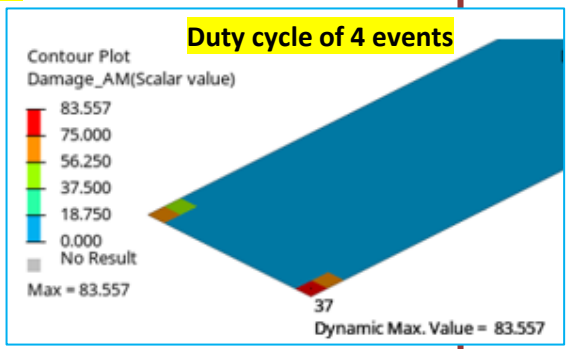
```
*PSD_LoadCases
     Label: Force4Evs
     Fatigue_psd: YES

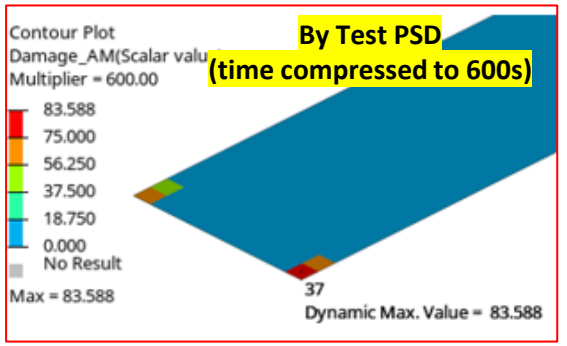
individual_psdLoads: loadPsd
!  LC_id   Lab     psdFileName            noHeadRows   Col-id
     1     ev1     signal_event1_psd.csv      1          2
     2     ev2     signal_event2_psd.csv      1          2
     3     ev3     signal_event3_psd.csv      1          2
     4     ev4     signal_event4_psd.csv      1          2

     5     tst     test_psd.csv               1          2

!  ---- OPTIONAL ----  Duty-Cycle for individual psd loads
psd_dutyCycle: shk
!  LC_id   Duration(s)   Repeats
     1        120.0         1.0
     2        150.0         1.0
     3       1200.0         1.0
     4       5000.0         1.0
```

Verification by FatDyn-X





Test-PSD has exact comparison to duty-cycle of multiple PSDs

SECTION 13

FatDyn-X Workflow

A single keyword-based input file drives every analysis.

Workflow — Input

Input keyword sections

Node_Sets
Element_Sets
MID_sets
PID_sets
Time_Sets

FE_DATA

TIMEHISTORY_LOADCASES
FREQUENCY_LOADCASES
PSD_LOADCASES

Fatigue_Material_Mapping
Fatigue_Material_Data

END

```
*TIMEHISTORY_LOADCASES
!=================  load case 1   ==========  Fatigue and Dynamics ============================================================
     LABEL            : mt
     ANALYSIS_METHOD  : modal_transient
   ! ANALYSIS_METHOD  : INERTIA_RELIEF/STATIC
   ! ANALYSIS_METHOD  : INERTIA_RELIEF/STATIC_byModes
   ! ANALYSIS_METHOD  : Modal_Resonance

     Fatigue_Analysis         : YES   -10   gate1LifeThreshold  -10         ! YES/NO  gate1(%)   gate1LifeThreshold    gate2(%)
     FATIGUE_STRESS_METHOD(S) : ABSMAXPR  ! CRITPLANE, SIGNEDMISES, TRESCA
     NO_REPEATS               : 100

     modal_coord_pch: YES
     modal_coord_csv: YES
     modal_load_csv : YES

     Fatigue_Damage_Sensitivity_to_Loads/Modes:  YES  fatElemSet gate  fatMatId   ! to Loads for Inertia-Relief/Static
                                                                                 ! to Modes for Normal-Mode Method

                                ! YES/No  SET-Label
     node_timeHistory           : YES     nodeResp
     nodePair_timeHistory       : YES     nodePairs
     nodePair_Plotels_timeHistory : YES   plotelSet
     Force_timeHistory          : YES     forceElems
     Stress_timeHistory         : YES     stressElems

     massForce_timeHistory      : YES     conm2Set


     ANALYSIS_TIME_STEP    :  0.002
$ SerialNo   Label          loadChannel_fileName    noHeadRows     COL_id   divider   scaleFactor   offset     node_id     dir   IRstat_LC
     1,       P1x,     LargePlateLoads_12LCs.csv,          1,          2,     5.00,       1.00,       0.0,          1,        1        1
     2,       P1y,     LargePlateLoads_12LCs.csv,          1,          3,     5.00,       1.00,       0.0,          1,        2        2
     3,       P1z,     LargePlateLoads_12LCs.csv,          1,          4,    10.00,       1.00,       0.0,          1,        3        3
!
     4,       P2x,     LargePlateLoads_12LCs.csv,          1,          5,     5.00,       1.00,       0.0,          2,        1        4
     5,       P2y,     LargePlateLoads_12LCs.csv,          1,          6,     5.00,       1.00,       0.0,          2,        2        5
     6,       P2z,     LargePlateLoads_12LCs.csv,          1,          7,    10.00,       1.00,       0.0,          2,        3        6

      Save_reduced_loads_byGates: YES   -1.0  -1.0   95.0   ! max 3 gates, min 1.0, -ve no-gate

      Static_loadCases_byMinMax_Loads        : YES   1000   50.0        ! YES/NO  LC_startID   maxPercent
      Static_loadCases_byTimeSet             : YES   tset1  5000        ! YES/NO  setLabel     LC_startID
      IRstatic_unitLoadCases                 : YES                      ! YES/NO

      Transient_loadCases_byMinMax_Loads     : YES   1000   70.0  100   ! YES/NO  LC_startID   maxPercent   noPads
      Transient_loadCases_byTimeSet          : YES  tset1   5000  100   ! YES/NO  setLabel     LC_startID   noPads
      Full_TransientDynamic_Loadcase         : YES                      ! YES/NO
      NormalMode_Loadcase                    : YES                      ! YES/NO
-
```

Workflow — Run

Step 1

Prepare a simple keyword-based text input file (e.g., test-Input.inp).

Step 2

Run from a command window:

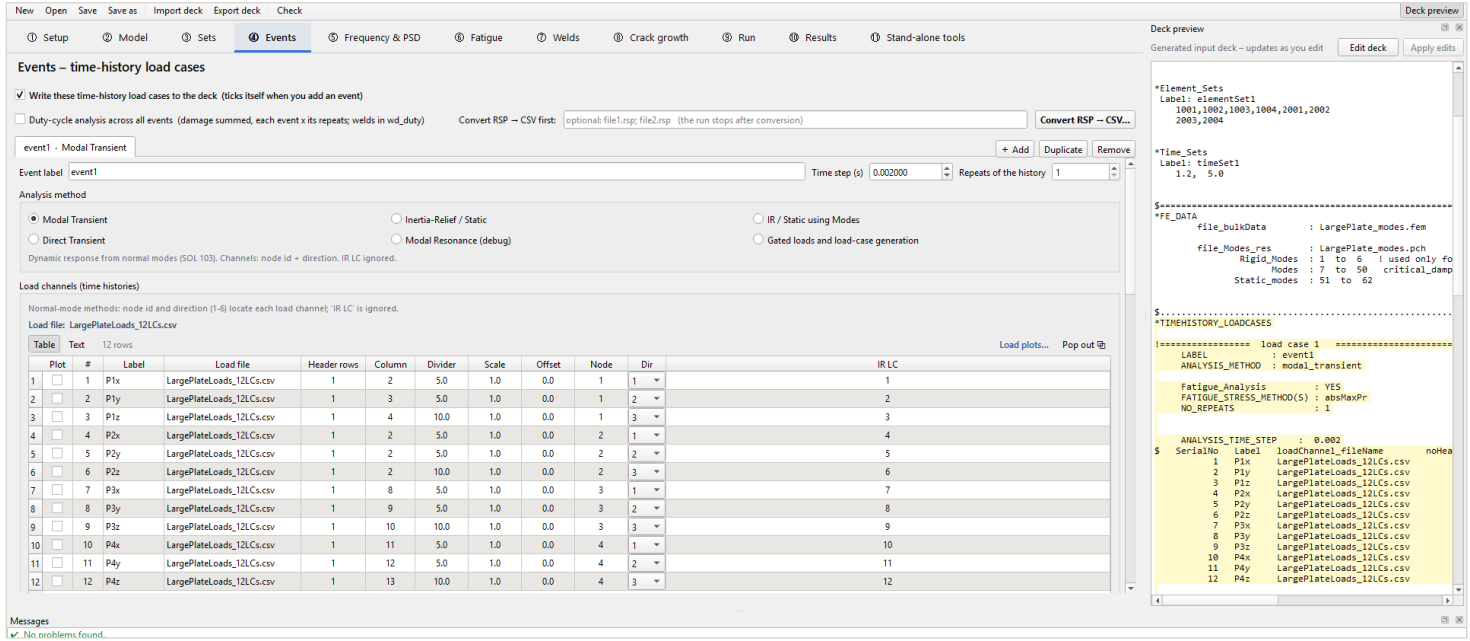
```
FatDyn-X.exe   test-Input.inp
```

Stand-alone Windows EXE without any external dependencies

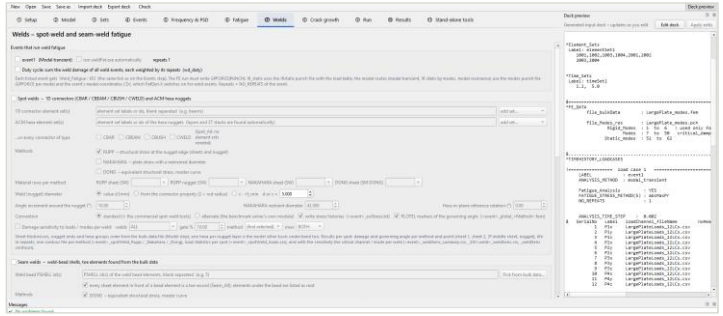
One EXE for everything

A single executable handles all solution types, dynamic calculations, and FE-operations.

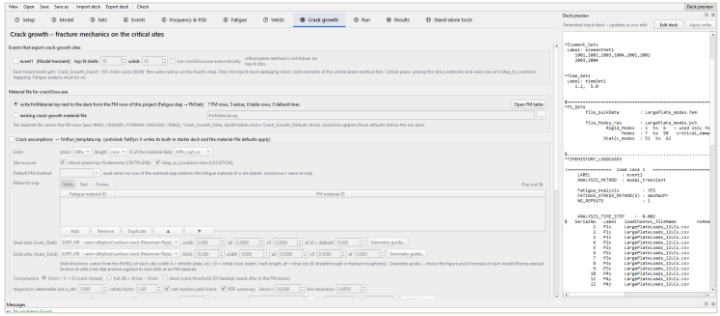
FatDyn-X Studio — the Graphical Front End to Every Analysis



Events step — one tab per load case: method, load channels, fatigue options; the generated deck updates on the right



Welds step — spot and seam weld definitions, per-event flags, results viewer



Crack-growth step — sites, crack-growth material file, crack assumptions

Guided workflow, one deck, every tool

- Eleven guided steps: Setup, Model, Sets, Events, Frequency & PSD, Fatigue, Welds, Crack growth, Run, Results and Stand-alone tools
- Live deck preview: the master input file is generated as you edit; edit the deck text and apply the edits back to the panels
- Pre-flight checks before every run; double-click a message to jump to the step and event that needs fixing
- Reads the bulk data for MIDs, PIDs, sets and mode frequencies; imports load channels from CSV and converts RSP files
- Material tables for S-N, ε-N, brittle, crack-growth, spot-weld and seam-weld rows with curve plots; open-source material database browser with provenance
- Load plots with Welch PSD and mode-frequency overlays; the Results step plots the CSV outputs
- Runs FatDyn-X, weldFat, crackGrow and every stand-alone tool from one window; single-file Windows executable, project saved as .fdxproj

Summary

A consolidated view of FatDyn-X capabilities.

FatDyn-X Capabilities at a Glance

Solvers

- Inertia-Relief / Static
- Inertia-Relief / Static via Normal Modes — unique
- Modal-Transient
- Modal-Resonance — unique
- Frequency response and PSD fatigue from normal modes — unique
- Parallel processing — no per-core fees

Fatigue Calculations

- Stress-life and strain-life
- Stress Methods: absMaxPr, signed Mises, Tresca, critical-plane (shells & solids)
- Spot and Seam weld fatigue
- Location- and plane-specific fatigue — unique
- Brittle (ceramic, glass, quartz) and thermo-mechanical fatigue
- Fracture mechanics: Paris, Walker, Forman, NASGRO, tabular da/dN — unique

Dynamic Calculations

- Node time histories: DIS, VEL, ACC
- Relative motion / strain for node-pairs
- Spring/rod/beam force time histories
- Shell & solid stress time histories
- Acceleration including rigid-body motion
- Mass-element force time histories — unique
- IR acceleration contours & mass forces — unique

Static & Dynamic Load Cases

- Static load cases by min/max loads
- Static load cases by time set
- IR-static unit load cases
- Transient load cases by min/max loads
- Transient load cases by time set
- Full transient dynamic load case
- Normal-mode load case
- Time-PSD plots with embedded structural frequencies

FatDyn-X uses IR / Static (SOL 101) or Normal Mode (SOL 103) — no full transient required. *Items marked unique: not commonly available in standard workflows.*

THE BOTTOM LINE

FatDyn-X unifies what currently requires

multiple tools and compromises.

Faster

More accurate

Integrated

For automotive · aerospace · space · machinery — fatigue and dynamics, integrated.