

# **Workshop: LS-DYNA Implicit**

**Dr. Tobias Erhart  
Bamberg, 8 October 2014**

- Motivation for implicit analyses
- Application examples
- Differences between explicit and implicit → consequences
- Implicit analysis with LS-DYNA: main keywords
- Guidelines: best practice
- **Exercise:**  
Going through different settings by means of a practical example

## Motivation: Why implicit ?

prestressed, quasi statically loaded structures

long duration analysis > 500 ms

different time scales in process

e.g. static loading followed by transient loading  
or transient loading followed by static loading

applications

e.g. metalforming, roof crush, door sag, dummy seating, strength analysis, ...



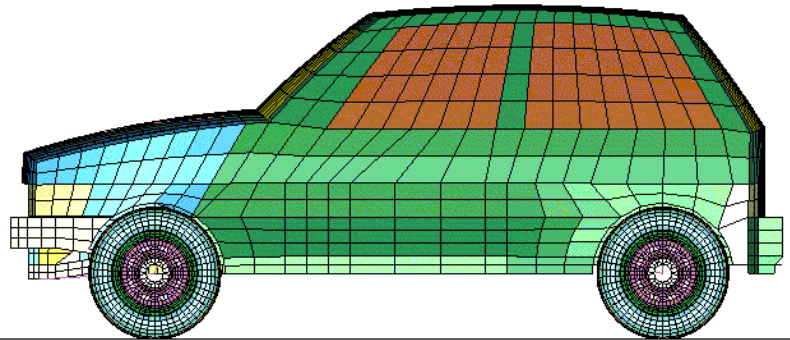
**LS-DYNA provides explicit and implicit solution schemes**

one code – one license - one data structure

one input / output

## Examples

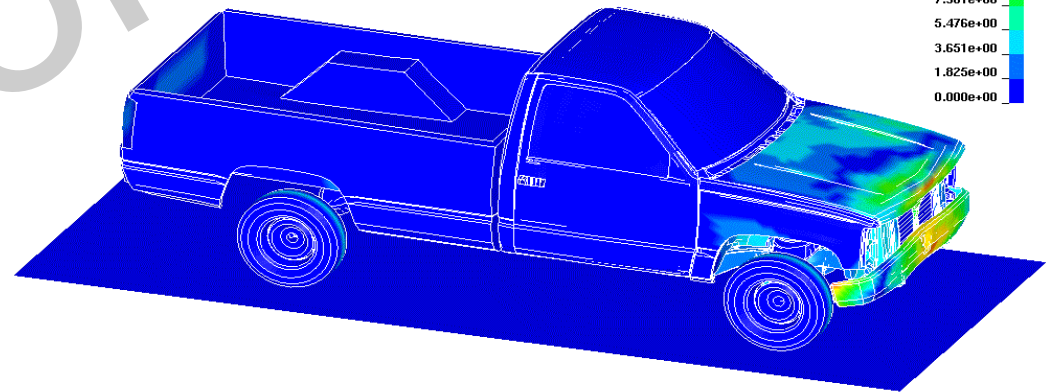
Gravity loading using Implicit Dynamics  
Time = 0



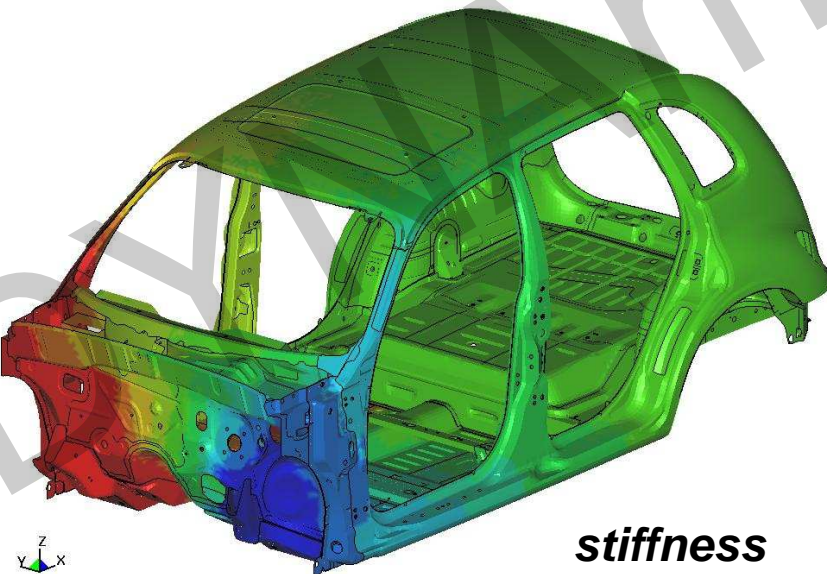
*gravity*



Implicit Dynamics SpringBack  
Time = 0.02  
Contours of Resultant Displacement  
min=0, at node# 105070  
max=17.2191, at node# 4



*springback*



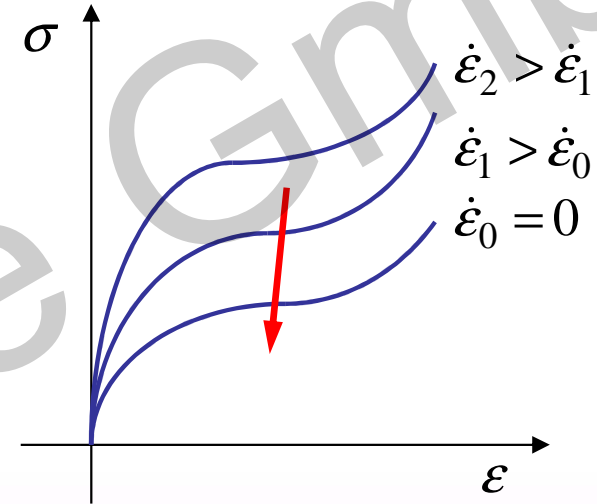
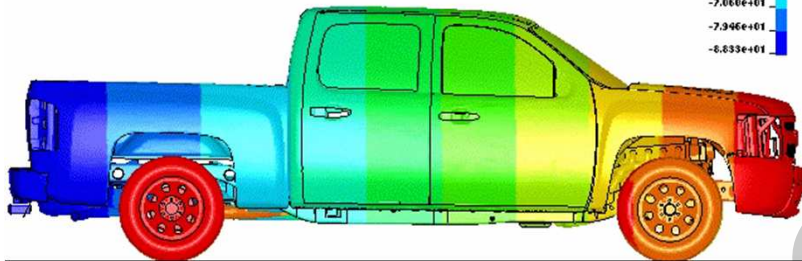
*stiffness*



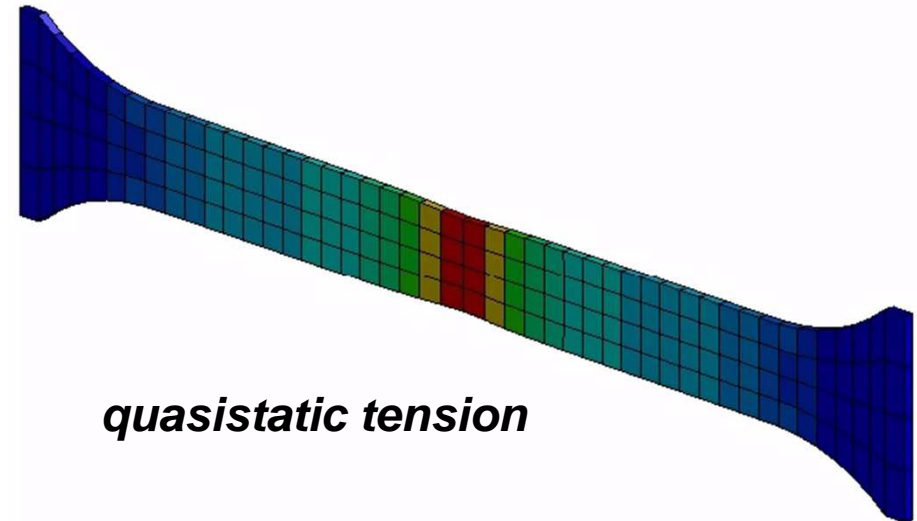
## Examples

IMPLICIT LS-DYNA iteration plot databas  
Freq = 0.001022  
Contours of Z-displacement  
min=-68.3257, at node# 2913163  
max=6.319337, at node# 2495746

*gravity*



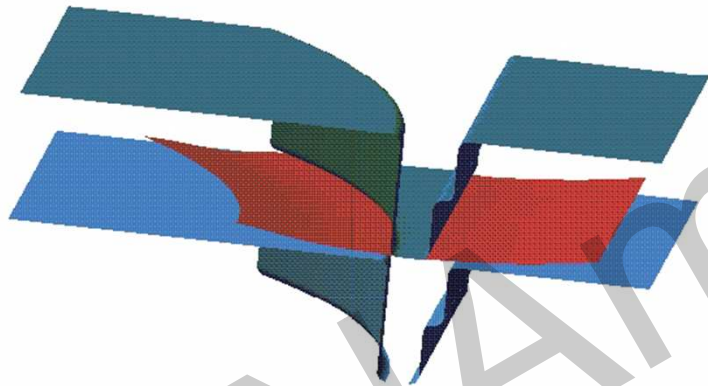
*roof crush*



*quasistatic tension*

## Examples

PNGV-SPP LAB VALIDATION: COMPLEX CHANNE  
Time = 0.003, #nodes=6607, #shells=6280

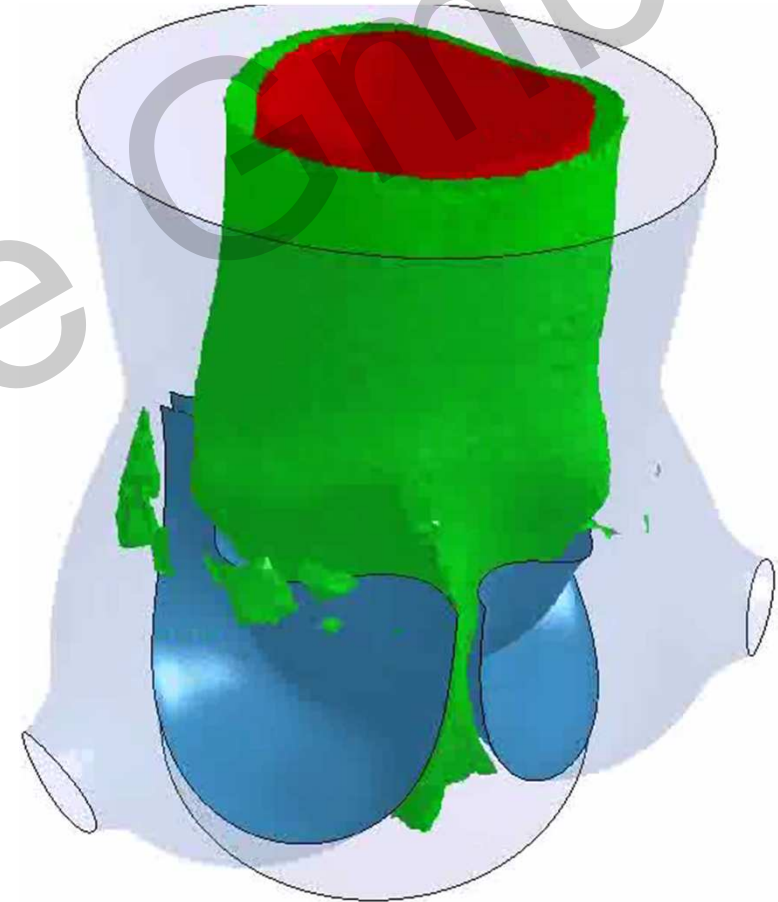
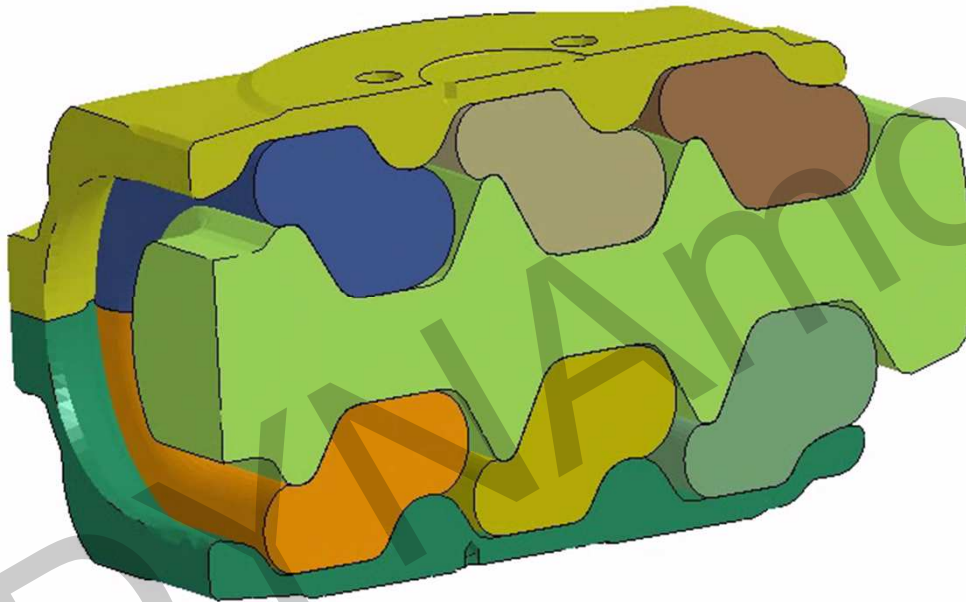


*metal forming*



## Examples

*rubber bushing*



*heart valve with FSI*

## Explicit vs. Implicit

**Explicit:**  $\mathbf{x}_{n+1} = \mathbf{f}(\mathbf{x}_n, \dots)$

$$\mathbf{M}\mathbf{a}_n = \mathbf{f}_n^{ext} - \mathbf{f}_n^{int}$$

- + solution: directly
  - + decoupled: efficient, fast
  - many small time steps
  - conditionally stable (Courant)
- equilibrium? energy balance!

**short time dynamics:**

high frequency response,  
wave propagation

➔ impact, crash, ...

**Implicit:**  $\mathbf{f}(\mathbf{x}_{n+1}, \mathbf{x}_n, \dots) = \mathbf{0}$

$$\mathbf{M}\mathbf{a}_{n+1} + \mathbf{K}\mathbf{u}_{n+1} = \mathbf{f}_{n+1}^{ext} - \mathbf{f}_n^{int} - \mathbf{M}\mathbf{a}_n$$

- solution: iteratively
  - linearization necessary
  - + few large time/load steps
  - + unconditionally stable
- equilibrium! convergence?

**structural dynamics:**

low frequency response,  
vibration, oscillation

➔ earthquake, machines, ...



## Explicit vs. Implicit

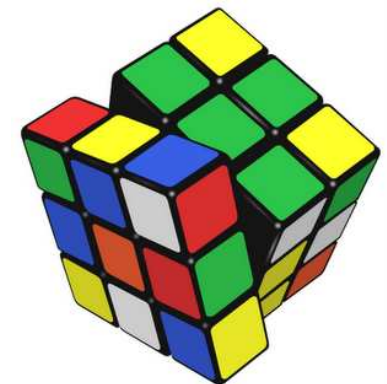
- Explicit inevitably includes inertial effects and resolves high frequencies whether you want it or not
- Implicit can neglect inertial effects and the selected time step size determines the resolved frequency spectrum

### Consequences for FE models

- "cleaner" models in implicit for the sake of convergence, e.g. no initial penetrations, smooth material curves, contact, ...
- expensive features are not so expensive anymore
- no restriction on element size (time step size) in implicit
- often more work to get "normal termination" in implicit



**"Explicit is handcraft - implicit is skill"**



## Types of Implicit Analyses

### Linear Analysis

- static or dynamic
- single, multi-step

### Eigenvalue Analysis

- frequencies and mode shapes
- linear buckling loads and modes
- modal analysis: extraction and superposition
- dynamic analysis by modal superposition

### Nonlinear Analysis

- Newton, Quasi-Newton, Arclength solution
- static or dynamic
- default LS-DYNA: **static and nonlinear!**

## Activating implicit analysis

### Use **\*CONTROL\_IMPLICIT\_GENERAL** to activate implicit

- specify time step size
- all other **\*CONTROL\_IMPLICIT** keywords are optional
- default is nonlinear, static analysis

### Use a double precision executable for implicit analysis

- better convergence for nonlinear
- mandatory for linear, eigenvalue accuracy
- mandatory for MPP

### Stiffness matrix requires lots of memory

huge speed penalty for out-of-core jobs

```
lsdyna i=inp.k memory=500m
```

500,000,000 words:

2000 Mbytes in single precision

4000 Mbytes in double precision

### Most keywords apply to explicit and implicit

easy to run a model with either method, but: carefully inspect input deck

## Activating implicit analysis

### Three types of analyses can be performed

fully explicit (default), fully implicit, or switching (explicit - implicit, implicit - explicit)

### All keywords for implicit

*CONTROL_IMPLICIT_GENERAL	*CONTROL_IMPLICIT_SOLVER
*CONTROL_IMPLICIT_SOLUTION	*CONTROL_IMPLICIT_AUTO
*CONTROL_IMPLICIT_STABILIZATION	*CONTROL_IMPLICIT_DYNAMICS
*CONTROL_IMPLICIT_MODES	*CONTROL_IMPLICIT_EIGENVALUE
*CONTROL_IMPLICIT_BUCKLE	*CONTROL_IMPLICIT_INERTIA_RELIEF
*CONTROL_IMPLICIT_JOINTS	*CONTROL_IMPLICIT_TERMINATION
*CONTROL_IMPLICIT_FORMING	*CONTROL_IMPLICIT_CONSISTENT_MASS
*CONTROL_IMPLICIT_STATIC_CONDENSATION	

### Proper selection of LS-DYNA features

- only few features are not available in implicit mode
- warning & error messages, feature substitution

## Main Implicit Keywords

### \***CONTROL\_IMPLICIT\_GENERAL** (required for implicit)

- activates implicit mode, explicit-implicit switching
- defines implicit time step size (standard LS-DYNA termination time is used too)

### \***CONTROL\_IMPLICIT\_SOLUTION** (optional)

- parameters for nonlinear equation solver (Newton-based methods)
- controls iterative equilibrium search, convergence
- "linear" analysis selected here (a special case where no iterations are performed)

### \***CONTROL\_IMPLICIT\_AUTO** (optional)

- activates automatic time step control
- default is fixed time step size, error termination if any steps fail to converge

## Main Implicit Keywords

### \*CONTROL\_IMPLICIT\_DYNAMICS (optional)

- include inertia terms
- problem “time” must now be real, physical time
- can improve convergence, especially when rigid body modes are present

### \*CONTROL\_IMPLICIT\_EIGENVALUE (optional)

- signals LS-DYNA to perform eigenvalue analysis, then stop
- number of eigenvalues/vectors, optional frequency shift
- great for debugging/model checking

## Day 1

- Introduction
- Linear static analysis
- Dynamic implicit analysis
- Nonlinear implicit analysis
- Eigenvalue analysis
- Modal analysis
- Buckling analysis
- Frequency response function

+ Workshops ...



## Day 2

- **Implicit/Explicit switching**
- **Element types**
- **Material types**
- **Contact types**
- **Nonlinear analysis II - arclength method**
- **Troubleshooting convergence problems**
- **Miscellaneous**
- **Final guidelines & references**

+ Workshops ...





## General

In general an explicit input deck can easily be transformed into an implicit input deck. However, in practice the implicit technique can give (convergence) problems since it is more sensitive to e.g. boundary conditions and non-linear behavior. Some general remarks and tips are given in the following in order to get started using the implicit solver in LS-DYNA

**The following card is added to the deck**

\*CONTROL\_IMPLICIT\_GENERAL: set IMFLAG=1 and DTO=STEPPSIZE

**Default is a static analysis but that can be changed to dynamic**

\*CONTROL\_IMPLICIT\_DYNAMICS: set IMASS=1

**Default is a non-linear analysis but that can be changed to linear**

\*CONTROL\_IMPLICIT\_SOLUTION: set NSOLVR=1

## Recommendations

### Use double precision of the code ( `_d_` in the name)

- required for accurate linear analysis
- improved convergence behavior in nonlinear analysis

### Use the most recent LS-DYNA version possible

implicit functionality is rapidly improving

### Use command line option "memory=" to run job in-core

Verify using `LPRINT=1` on `*CONTROL_IMPLICIT_SOLVER` or "`<ctrl-c> lprint`". The CPU penalty for out-of-core can be as high as 100 times the in-core simulation!!

### Read Appendix P in the User's manual

Nice summary about LS-DYNA's Implicit Solver

## Recommendations

### Element types

- for solids use type 1, -1, 13, or 16 elements for non-linear analysis
- for solids use type 18 elements for linear analysis
- for shells use type 6 or 16 elements for non-linear analysis
- for shells use type 18-21 elements for linear analysis

### Contact

- try to avoid initial penetrations or try IGNORE=1
- for tied connections use penalty based tied contact (\_offset option)
- try IGAP=2 on Additional card C or try the new Mortar contact
- contact often requires small time steps in implicit, too
- use soft part as slave

## Recommendations

### General

- apply second order stress update by setting OSU=1 at \*CONTROL\_ACCURACY
- try to model displacement driven simulation instead of force driven simulation
- try to use IGS=1 (not default) on \*CONTROL\_IMPLICIT\_GENERAL in case of convergence problems
- set DNORM=1 on \*CONTROL\_IMPLICIT\_SOLUTION, displacement tolerance can often be increased in that case, e.g. DCTOL=0.005
- (try ABSTOL=1.e-20 on \*CONTROL\_SOLUTION to improve accuracy)
- often dynamic solution more robust than static solution  
→ if static implicit fails to converge, try dynamic implicit
- try to avoid discontinuities, e.g. in material curves, geometry, ...

## Recommendations

### General

- in case of convergence problems, dump iteration states via "<ctrl-c> iter" (residual forces in d3iter via RESPLT=1 on \*DATABASE\_EXTENT\_BINARY)
- in general, if problems occur when running an implicit model, then try to check the model using \*CONTROL\_IMPLICIT\_EIGENVALUE
- in problems where there is much rigid body motion the displacement tolerance DCTOL may be insufficient, and it may be advisable, in some problems, to tighten the energy tolerance to 0.001.
- element size in implicit is not as important as in explicit (time step)
- CPU cost implicit is roughly proportional to  $\text{ndof}^2$
- CPU cost explicit is roughly proportional to  $\text{ndof}$

## Recommendations

For “typical” implicit analysis, start with the following keyword settings:

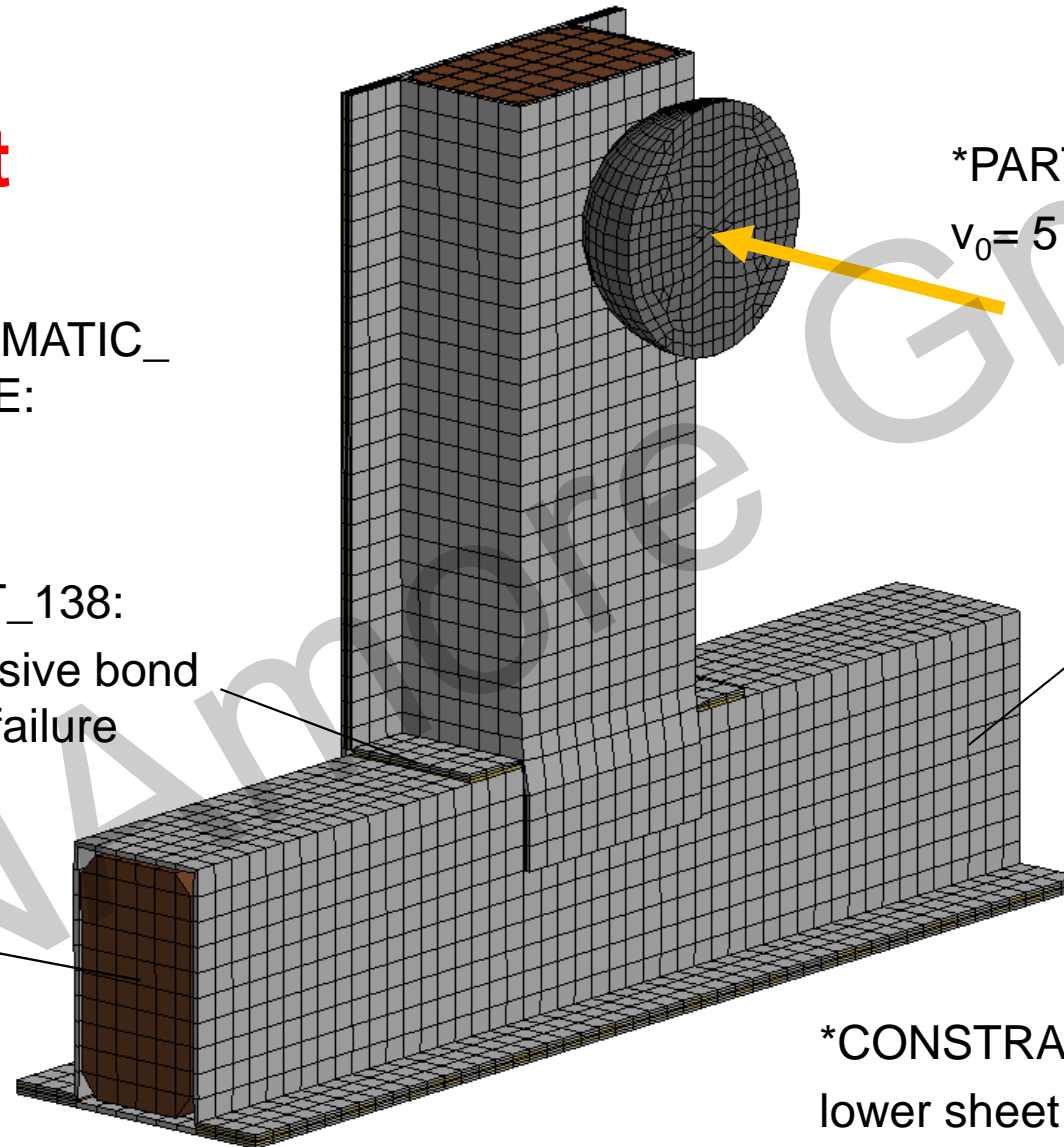
```
*CONTROL_ACCURACY
$      osu      inn
      1        2
*CONTROL_IMPLICIT_GENERAL
$  imflag      dt0
      1        ...
*CONTROL_IMPLICIT_SOLUTION
$  nsolvr      ilimit      maxref      dctl      ectol      rctl      lstol      abstol
  (2/12)              6      0.005
$  dnrm      diverg      istif      nlprint      nlrm      d3itctl
      1              1
$
$
$  lsmtd
  (3/5)
*CONTROL_IMPLICIT_AUTO
$  iauto      iteopt      itewin      dtmin      dtmax
      1        30        10              (term/20)
*CONTROL_IMPLICIT_DYNAMICS
$  imass
      (1)
```

## T-joint component

\*CONTACT\_AUTOMATIC\_SINGLE\_SURFACE:  
overall contact

\*MAT\_138:  
adhesive bond  
with failure

\*MAT\_024:  
wooden blocks



\*PART\_INERTIA:  
 $v_0 = 5 \text{ m/s}$

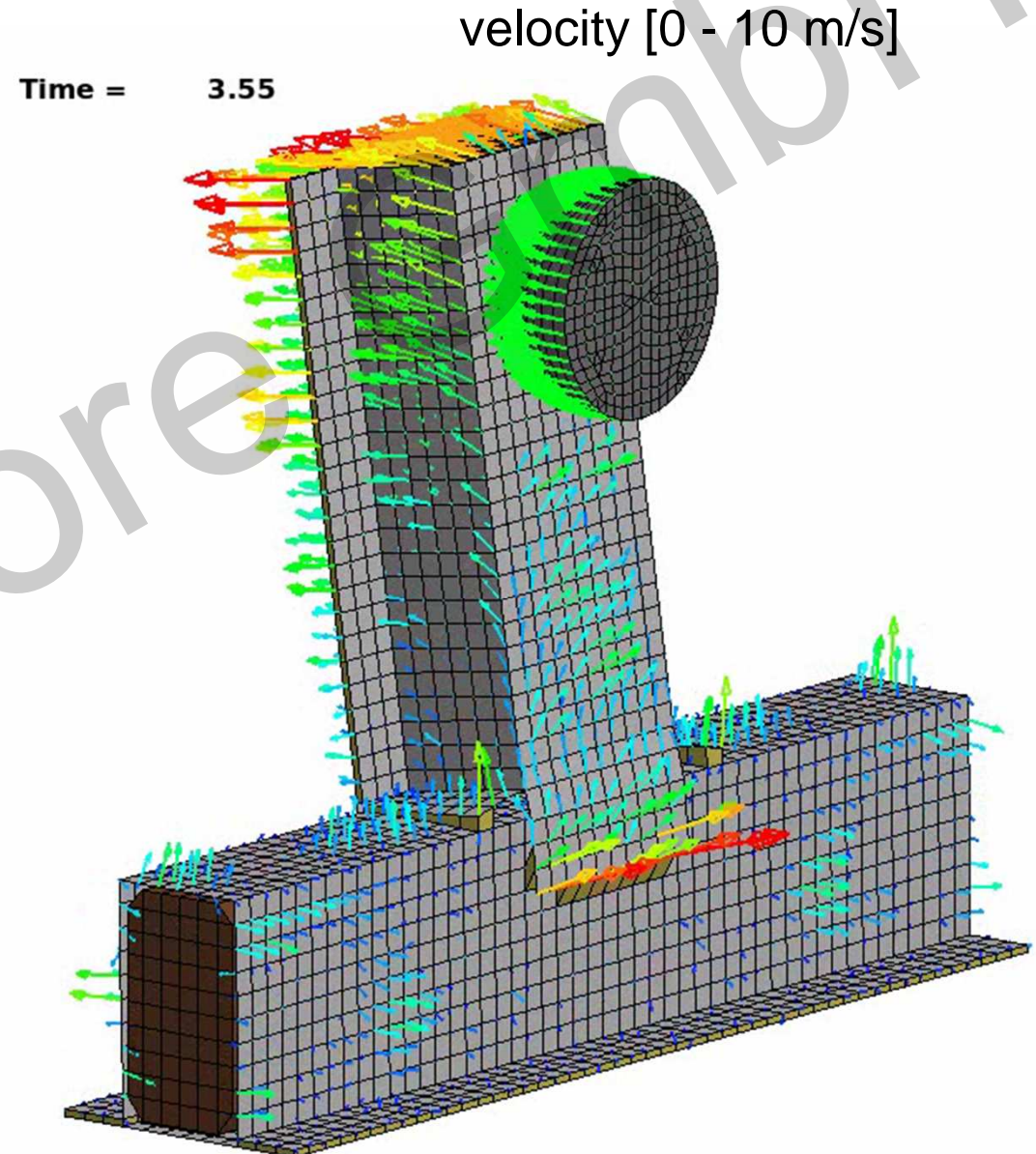
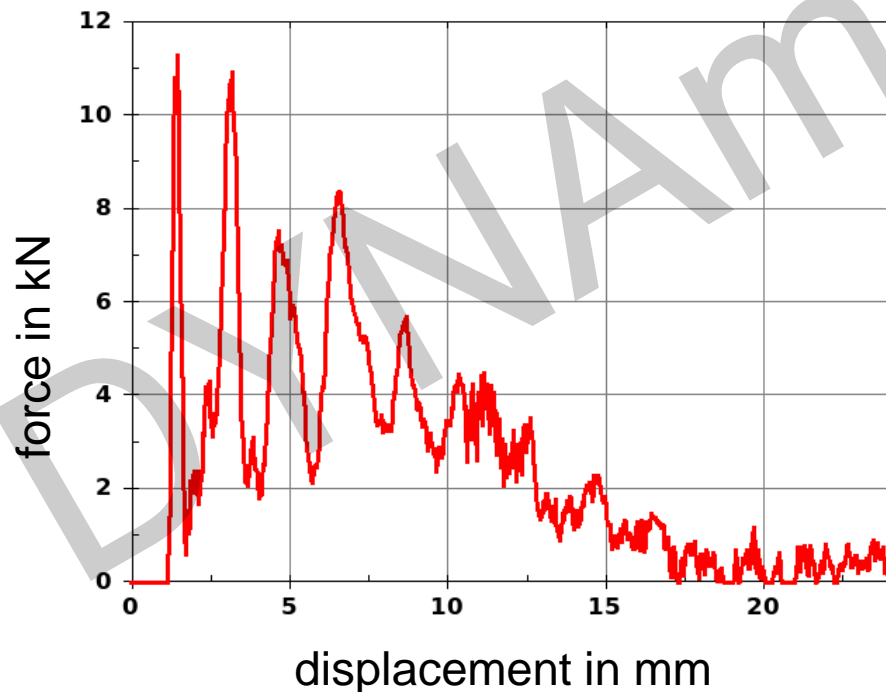
\*MAT\_024:  
DP 800

5 mm mesh  
for steel parts

\*CONSTRAINED\_RIGID\_BODY:  
lower sheet and wooden block

## Dynamic explicit

- Process time = 5 ms
- ~10,000 time steps
- 52 cohesive elements fail
- Low-frequency vibration and high-frequency response (wave propagation)



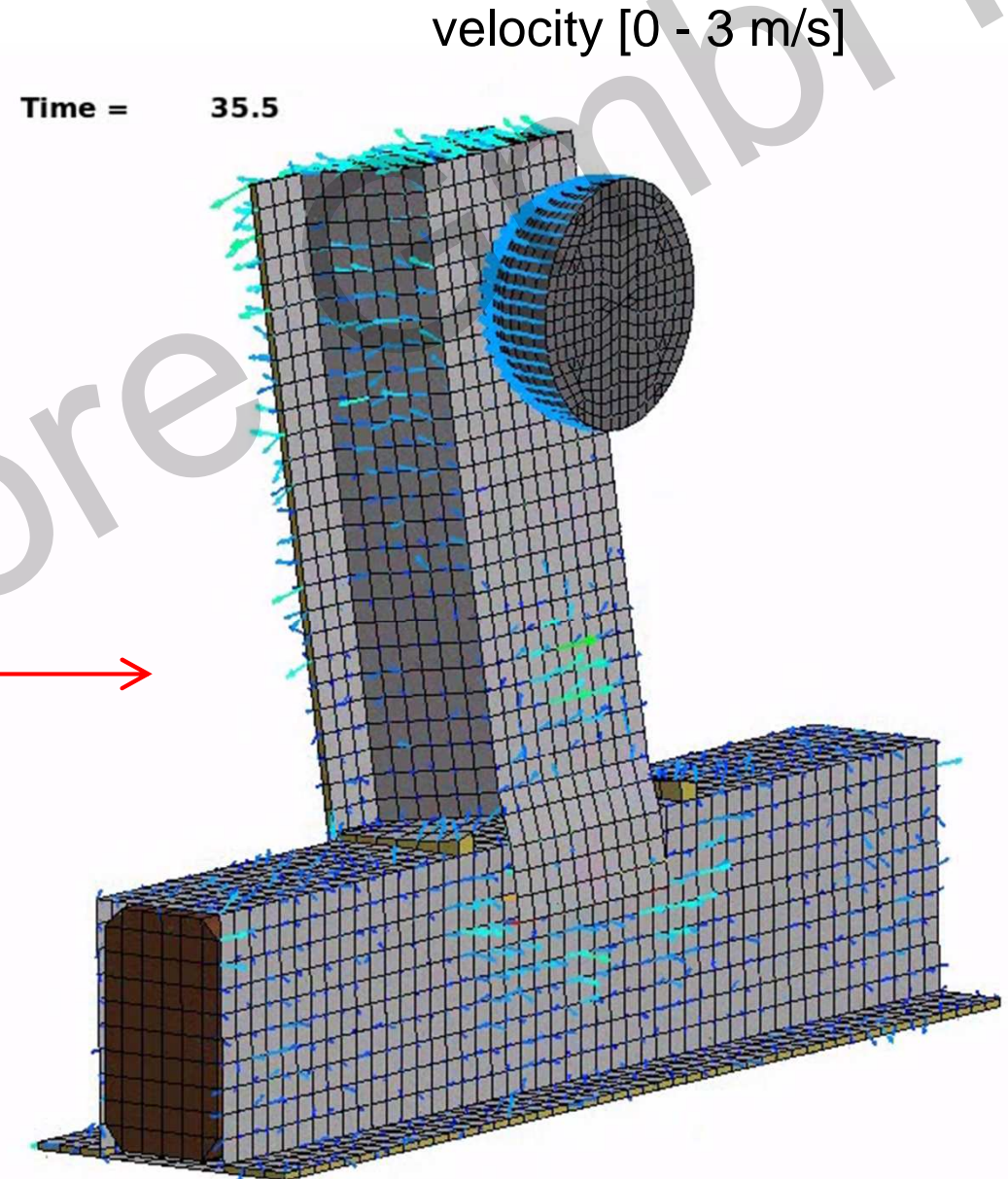
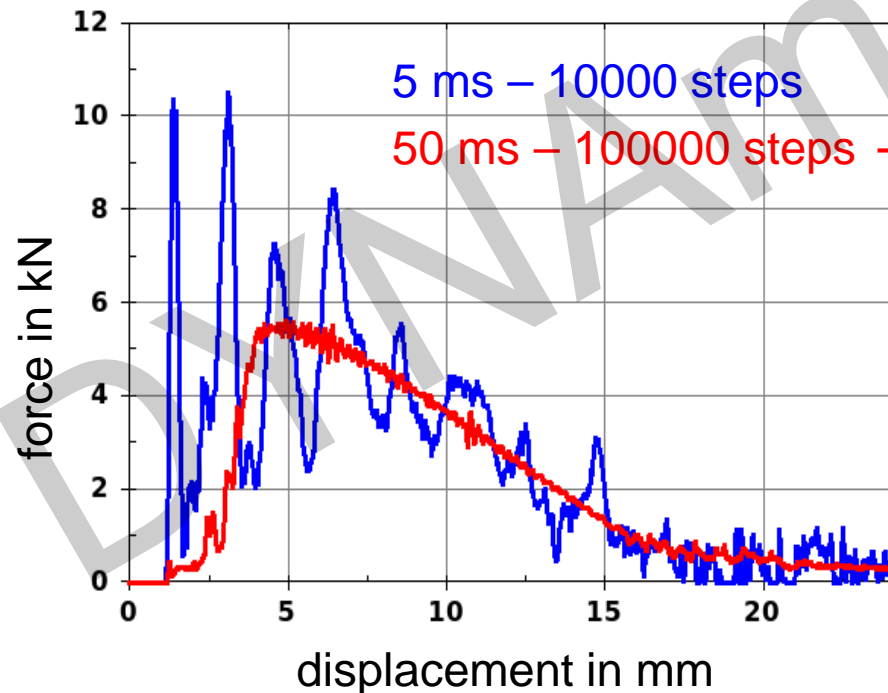


**Now, we want to do a static analysis of that process:**

1. Start with explicit using a larger time period (“slow” loading)
2. Add implicit cards needed for dynamic implicit analysis (“fast” and “slow” loading)
3. Remove dynamics and perform pure static analysis

## Static (??) explicit

- Process time = 5 / 50 ms
- ~ 10,000 / 100,000 time steps
- No initial velocity, but prescr. motion
- 52 cohesive elements fail
- Response still dynamic
- Damping... ??



## Dynamic implicit (default)

- Process time = 5 ms (“fast”)
- \*CONTROL\_IMPLICIT\_GENERAL:  
DT0 = 0.05 (100 steps)
- \*CONTROL\_IMPLICIT\_DYNAMICS:  
IMASS = 1

- 100 steps
- 2779 problem cycles
- 58 failed cohesives

## + Recommendations

- \*CONTROL\_ACCURACY: OSU = 1
- \*CONTROL\_IMPLICIT\_SOLUTION:  
NSOLVR = 12, ILIMIT = 6,  
DNORM = 1 (DCTOL=0.005)
- \*CONTROL\_IMPLICIT\_AUTO:  
ITEOPT = 30, ITEWIN = 10, DTMAX = 0.1

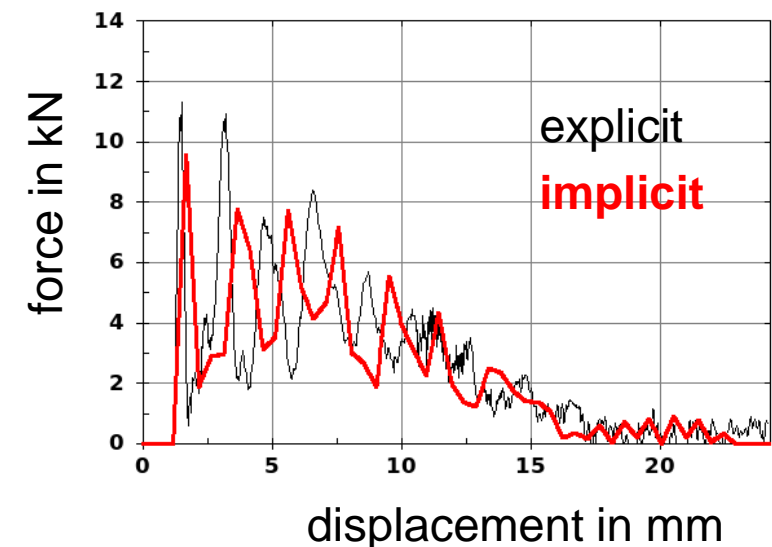
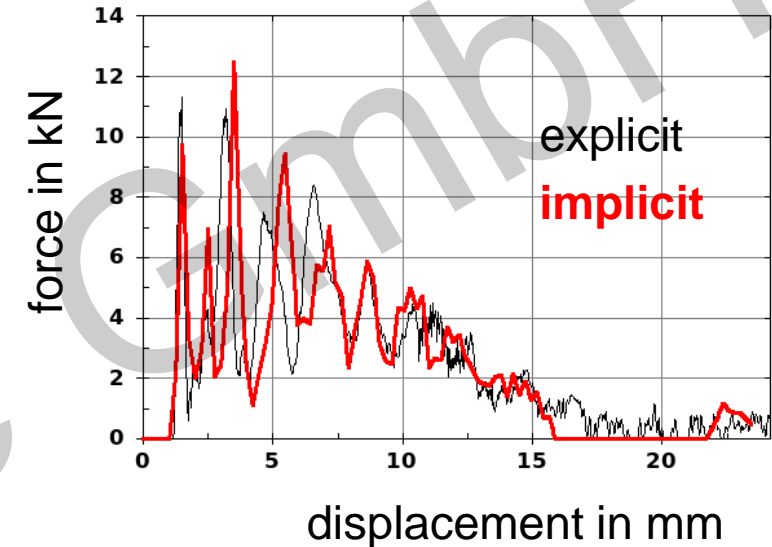
- 51 steps
- 1063 problem cycles
- 52 failed cohesives

## Dynamic implicit (default)

- Process time = 5 ms (“fast”)
- \*CONTROL\_IMPLICIT\_GENERAL:  
DT0 = 0.05 (100 steps)
- \*CONTROL\_IMPLICIT\_DYNAMICS:  
IMASS = 1

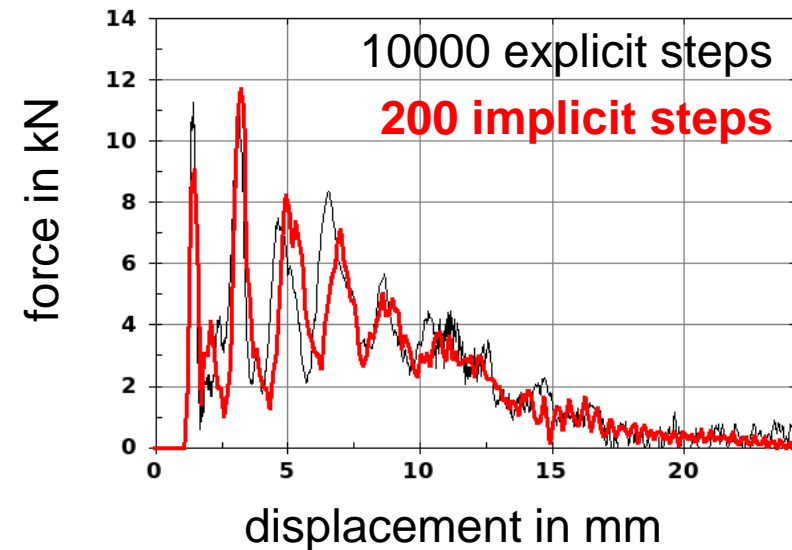
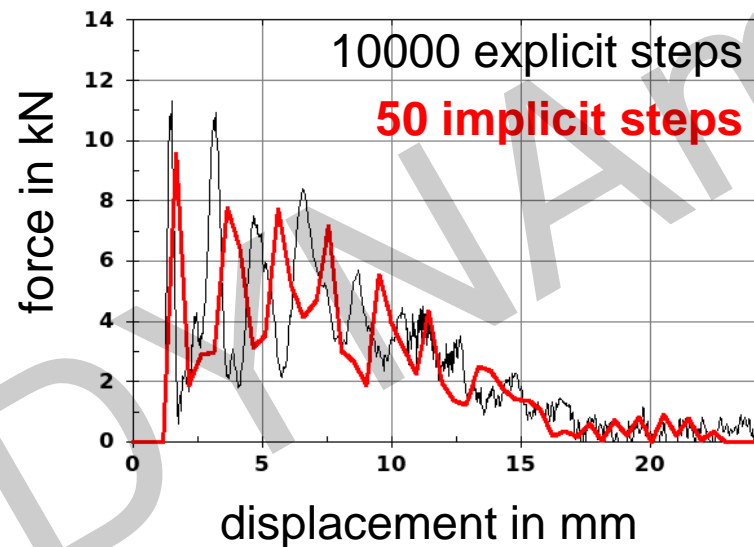
## + Recommendations

- \*CONTROL\_ACCURACY: OSU = 1
- \*CONTROL\_IMPLICIT\_SOLUTION:  
NSOLVR = 12, ILIMIT = 6,  
DNORM = 1 (DCTOL=0.005)
- \*CONTROL\_IMPLICIT\_AUTO:  
ITEOPT = 30, ITEWIN = 10, DTMAX = 0.1



## Dynamic implicit

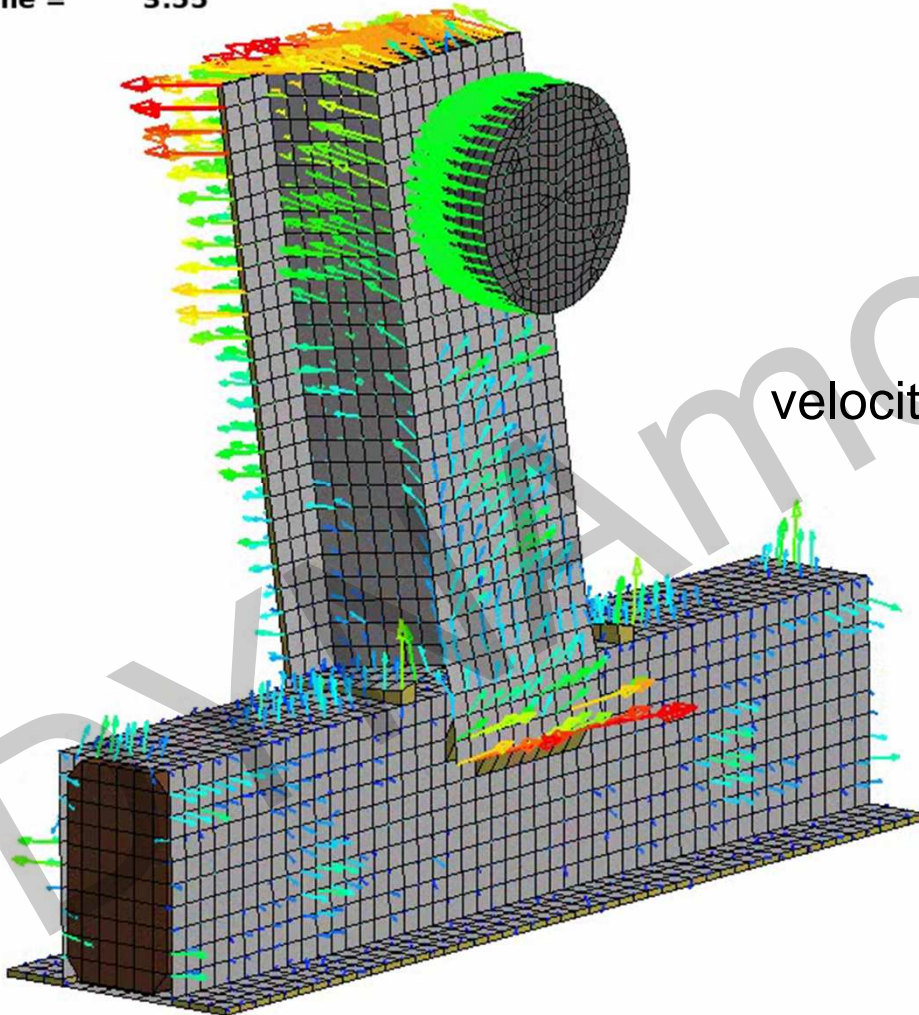
- What time step size is necessary to resolve the dynamic process?
- User needs good knowledge about the problem at hand
- User has to decide about the solution frequency
- Contact dominated problems need small time steps



## Dynamic explicit

- Low- and high-frequency response

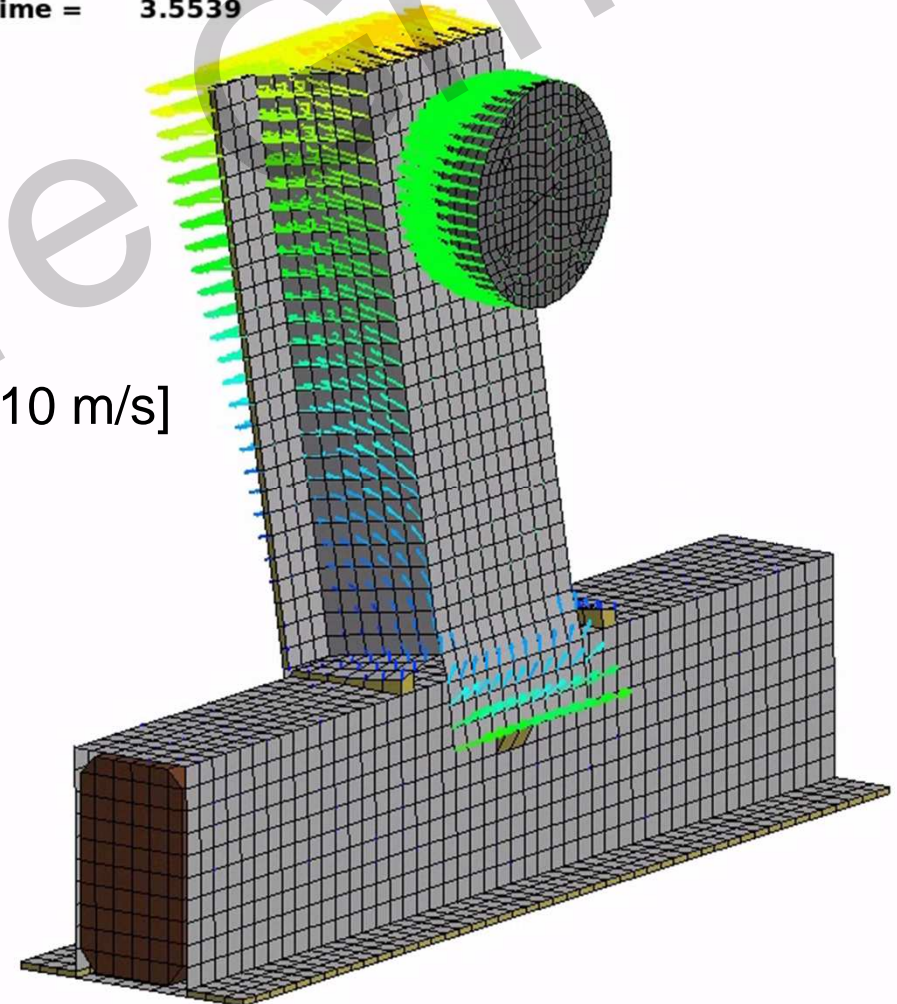
Time = 3.55



## Dynamic implicit

- Low-frequency response

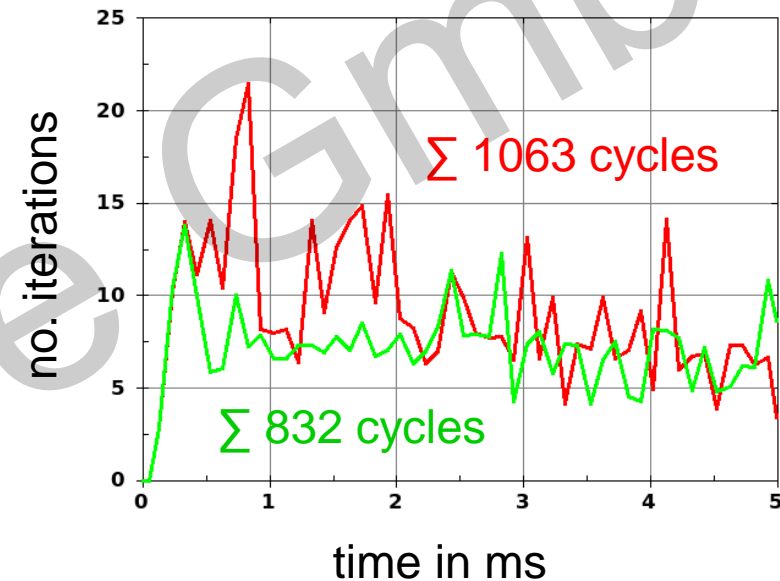
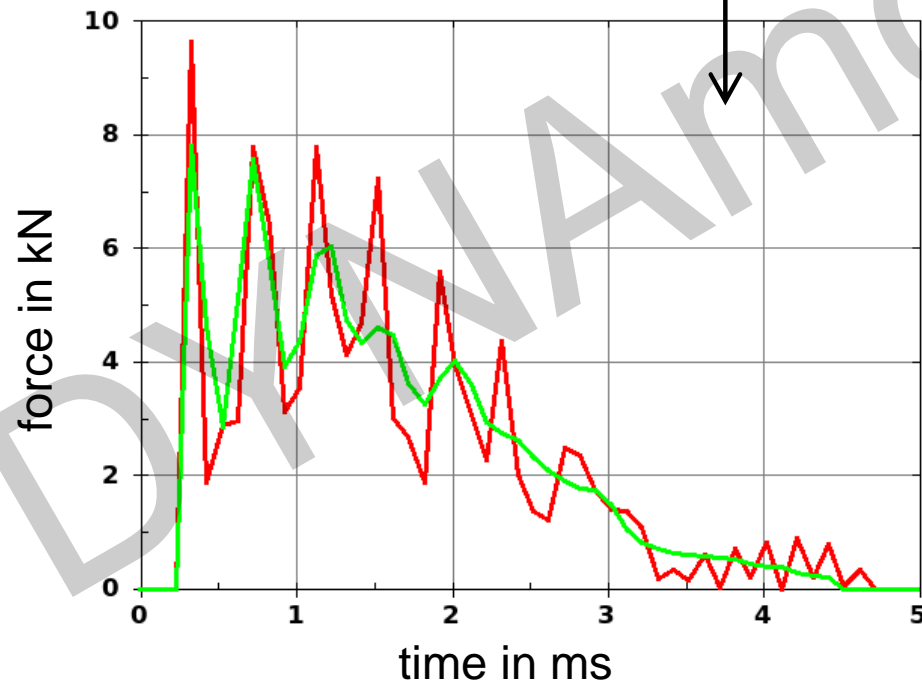
Time = 3.5539



velocity [0 - 10 m/s]

## Dynamic implicit

- Check influence of Newmark parameters GAMMA and BETA
- Default: GAMMA=0.5, BETA=0.25
- Larger GAMMA and BETA values introduce numerical damping
- Often helps convergence →
- But: affects solution! ↓



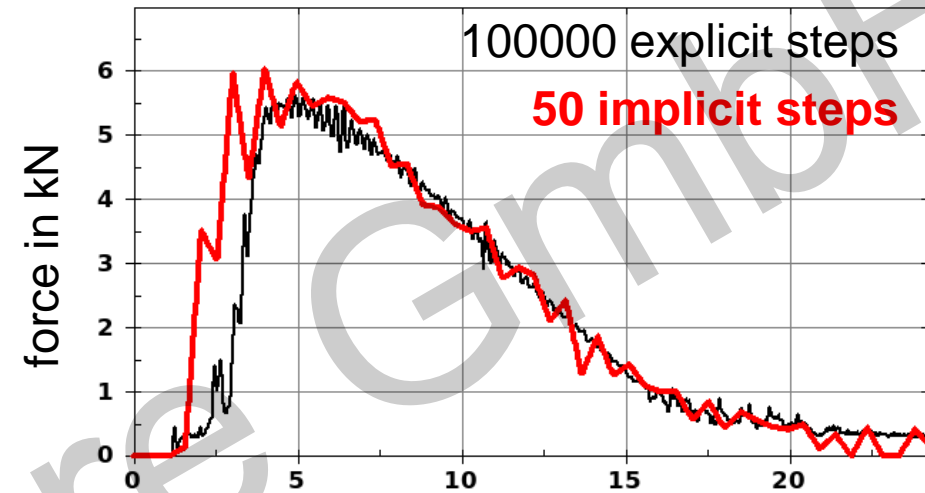
**GAMMA=0.5, BETA=0.25**

**GAMMA=0.6, BETA=0.38**

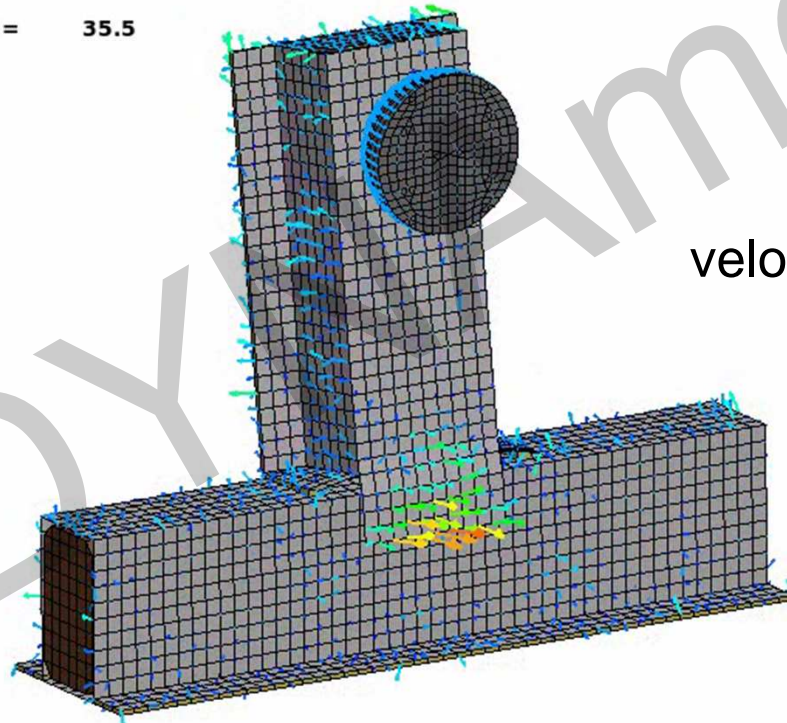
$$\beta > \frac{1}{4} \left( \frac{1}{2} + \gamma \right)^2$$

## Dynamic implicit

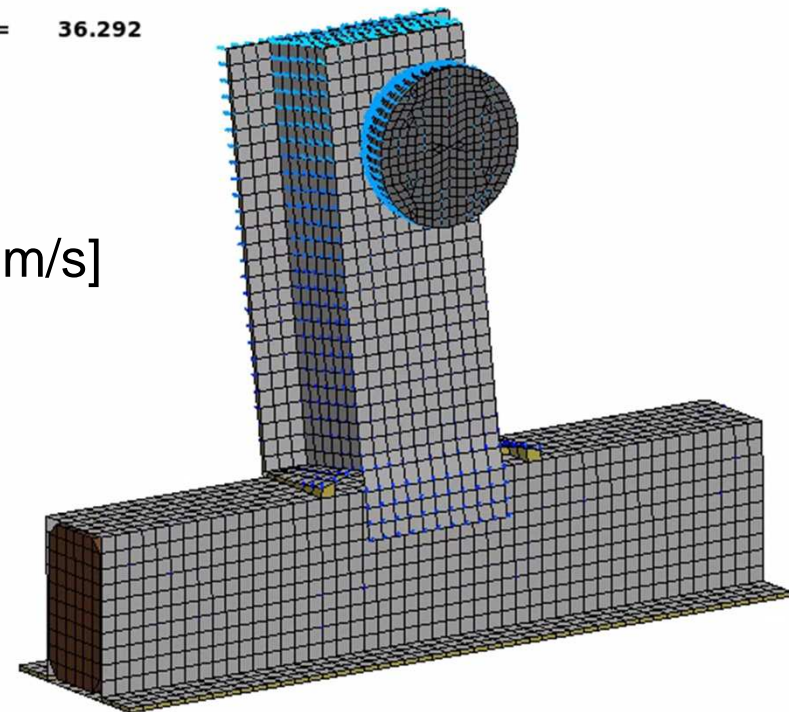
- Process time = 50 ms (“slow“)
- Compare to “slow“ explicit run



Time = 35.5



Time = 36.292

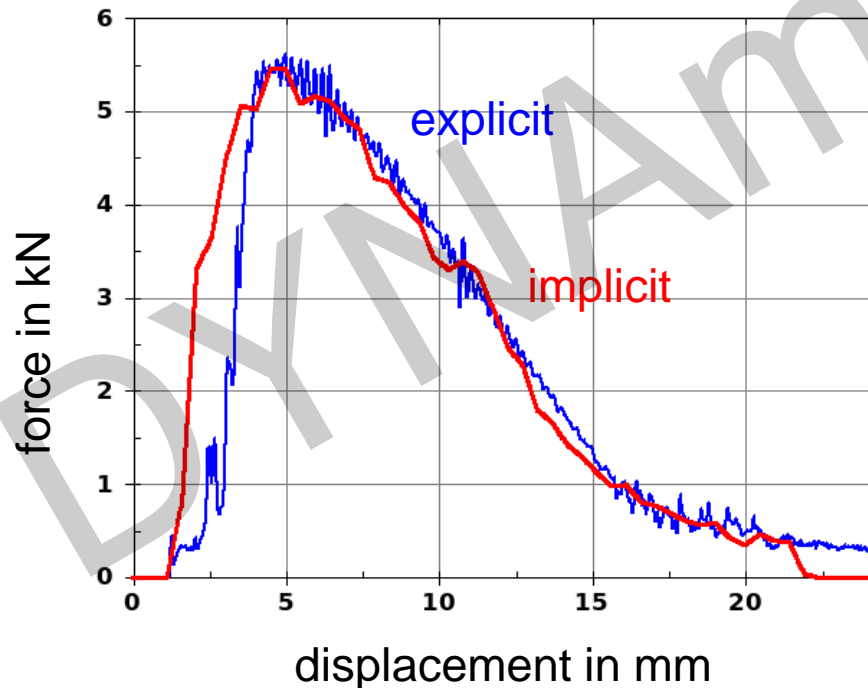


velocity [0 - 3 m/s]



## Static implicit

- Remove \*CONTROL\_IMPLICIT\_DYNAMICS
- No initial velocity, but prescr. motion
- “time“ not physical anymore
- Real static response
- statically defined !?!



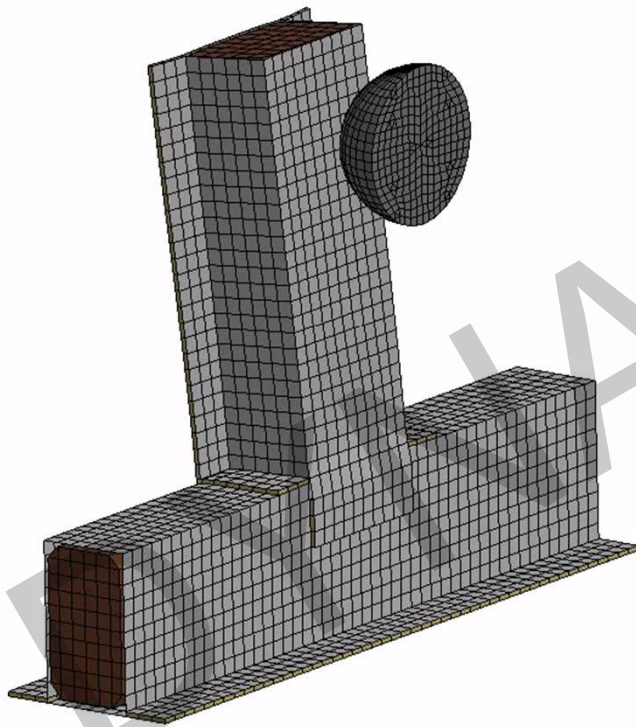
Time = 3.5292



## Eigenvalue analysis

- \*CONTROL\_IMPLICIT\_EIGENVALUE
- Reveals possible rigid body modes
- Superelevated deformations in d3eigv database

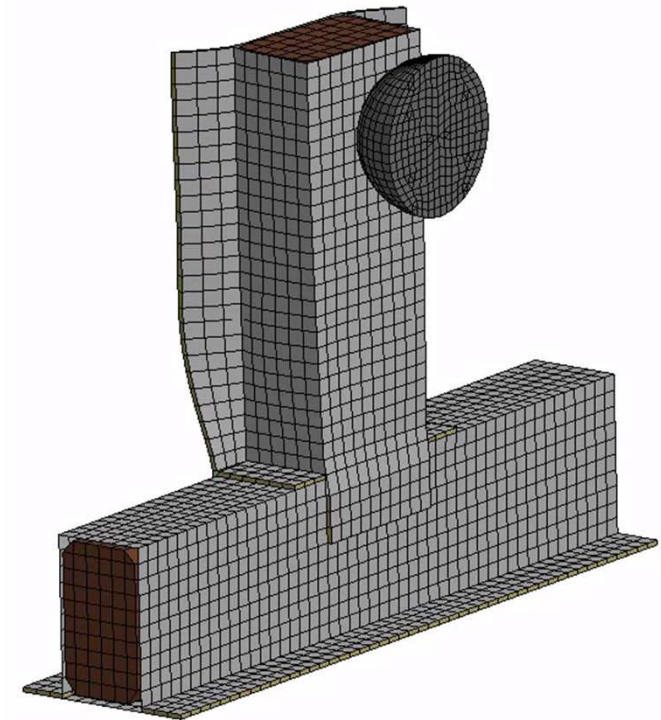
Freq = 1.0361



Freq = 1.744

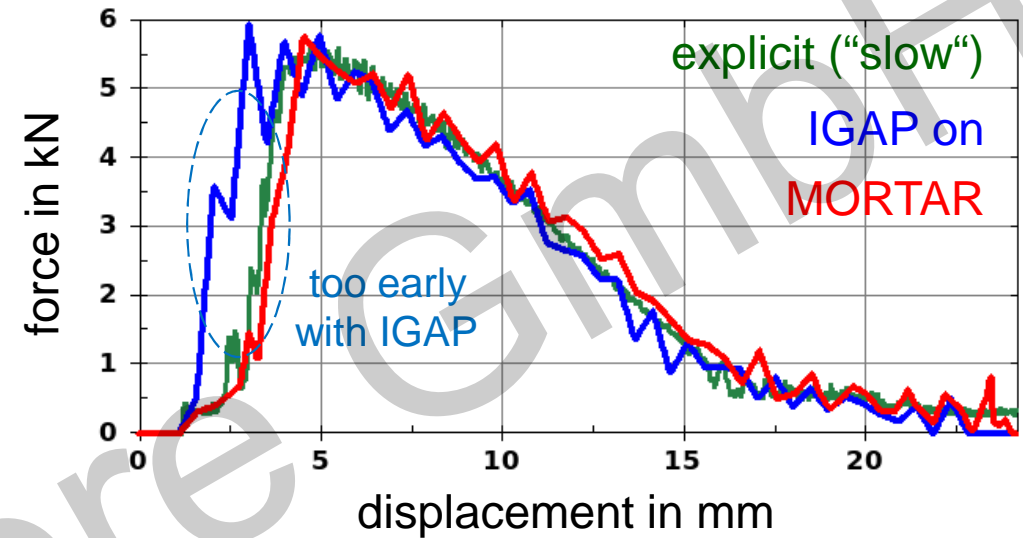


Freq = 2.3331



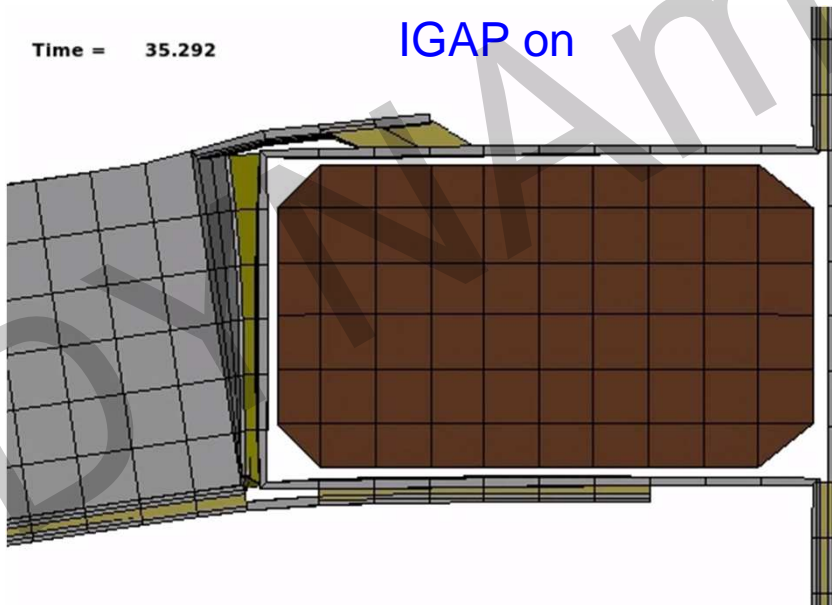
## Implicit contact

- Contact is very important issue (especially) in implicit analysis
- User should know about IGAP options (“sticky contact”) and Mortar contact (continuous tangent)
- Dynamic implicit shown here



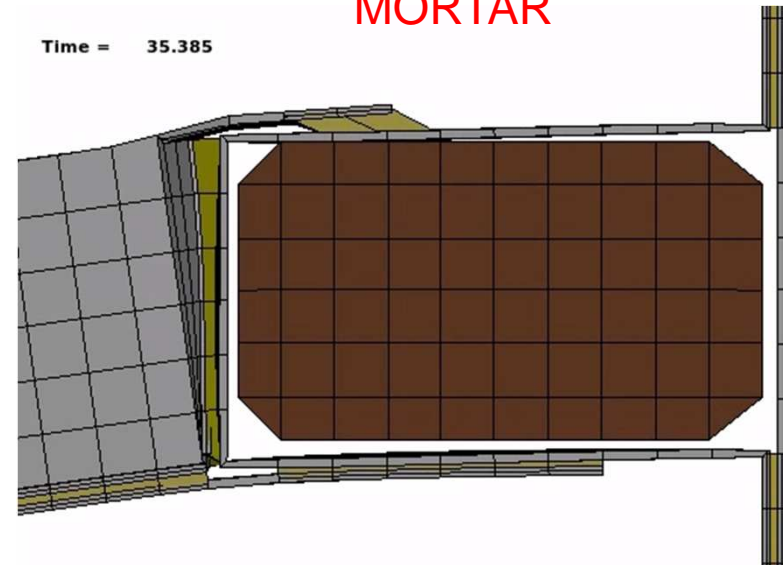
Time = 35.292

IGAP on



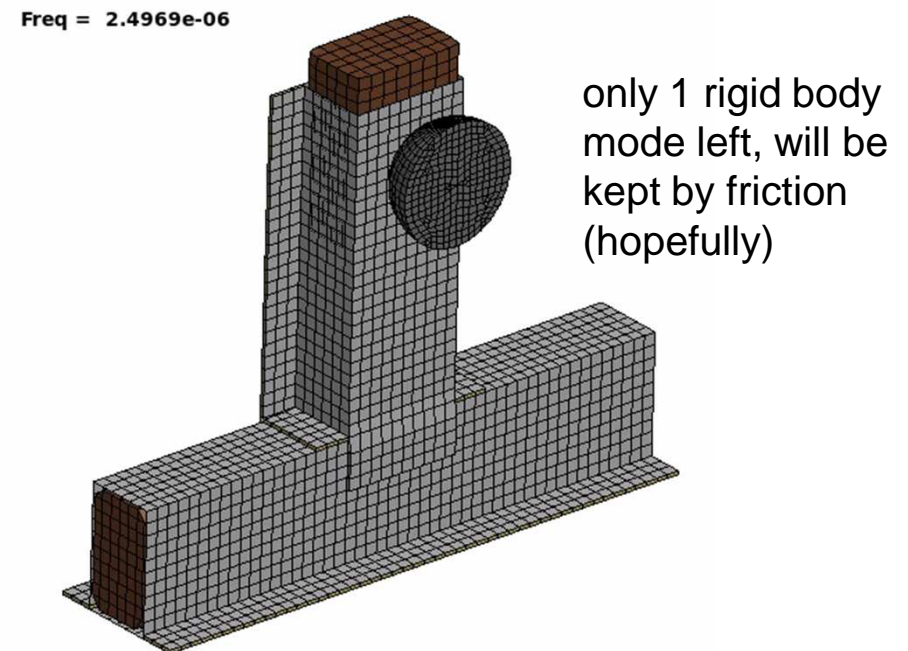
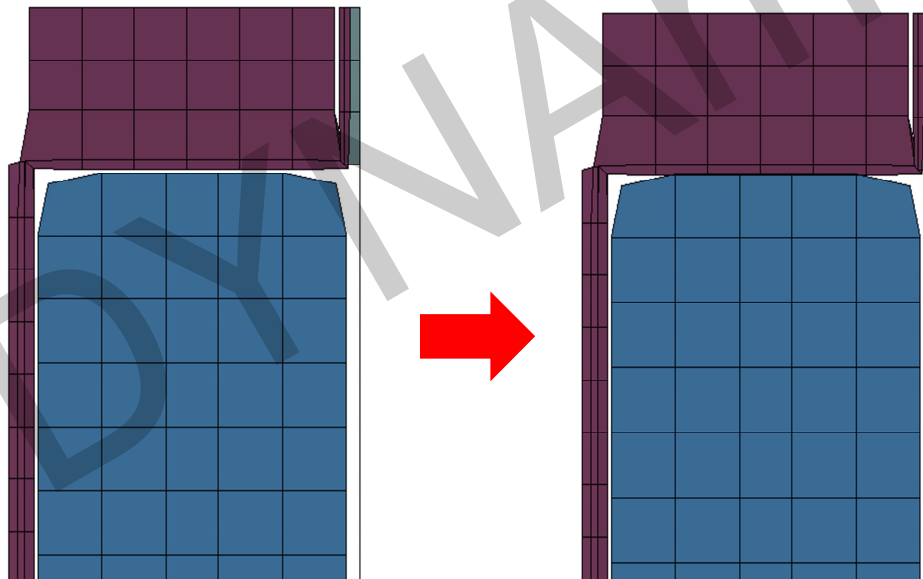
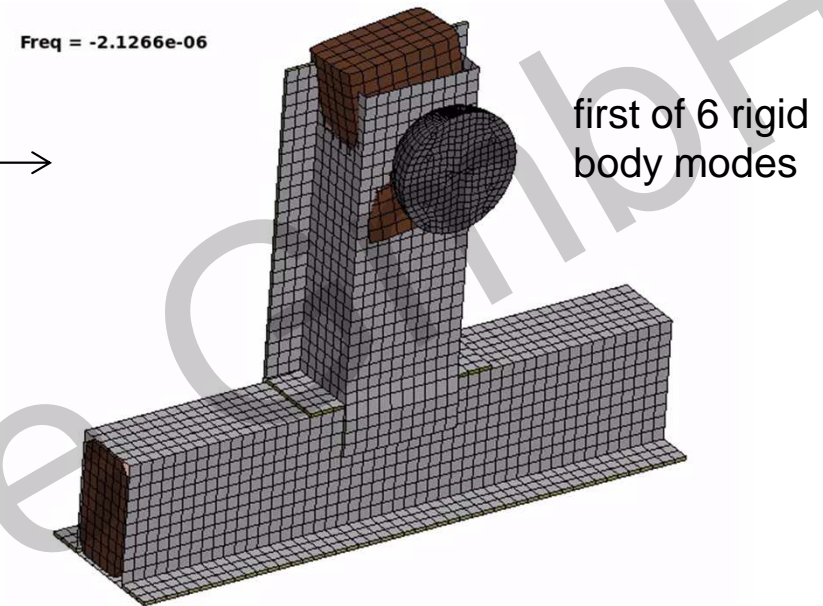
Time = 35.385

MORTAR



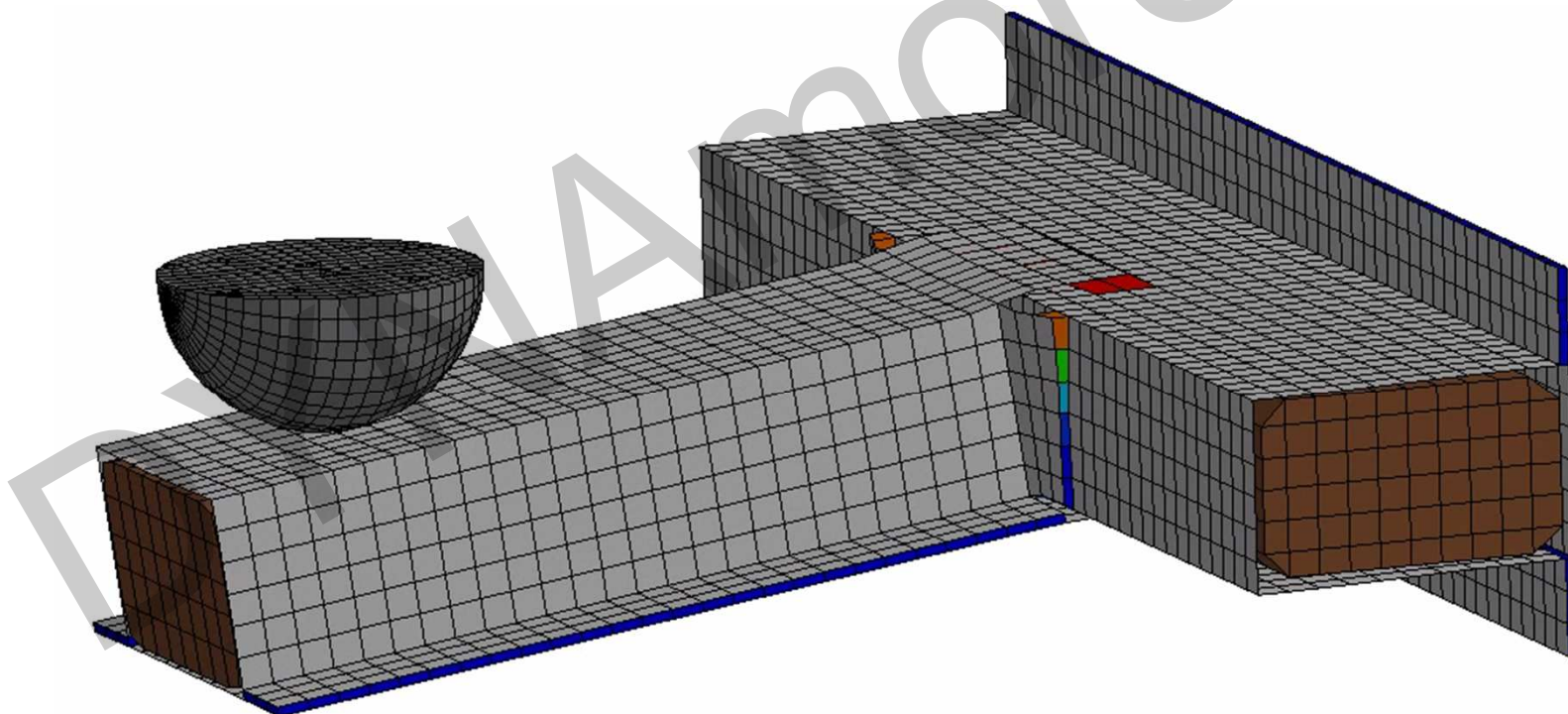
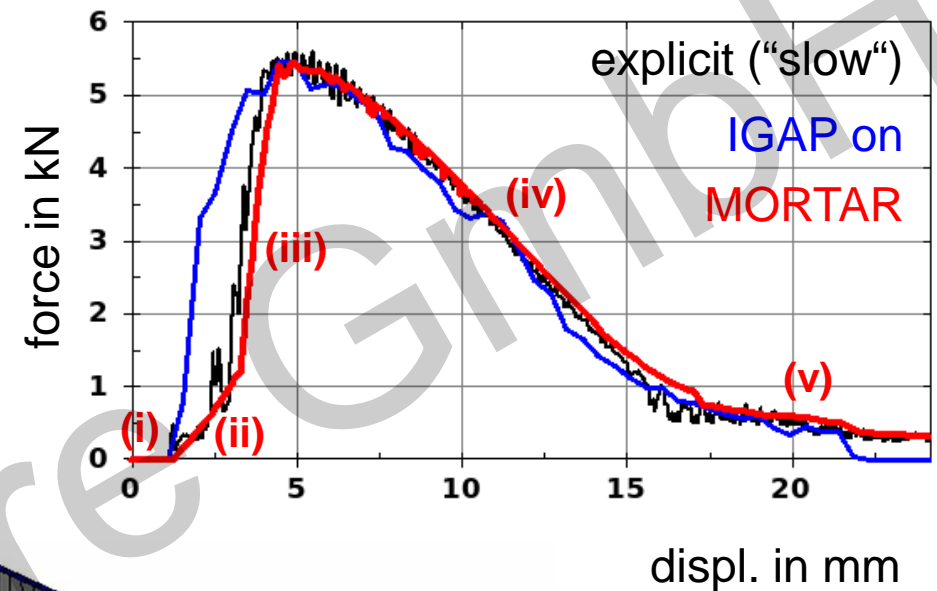
## Static implicit with Mortar contact

- “Missing“ contact gap now reveals 6 rigid body modes (wooden block) →
- Additional action(s) needed to allow for static analysis
- Slight **scaling** of wooden block's size causes initial contact penetration to get statically determined system
- +IGNORE=1 to avoid initial contact forces



## Static implicit with Mortar contact

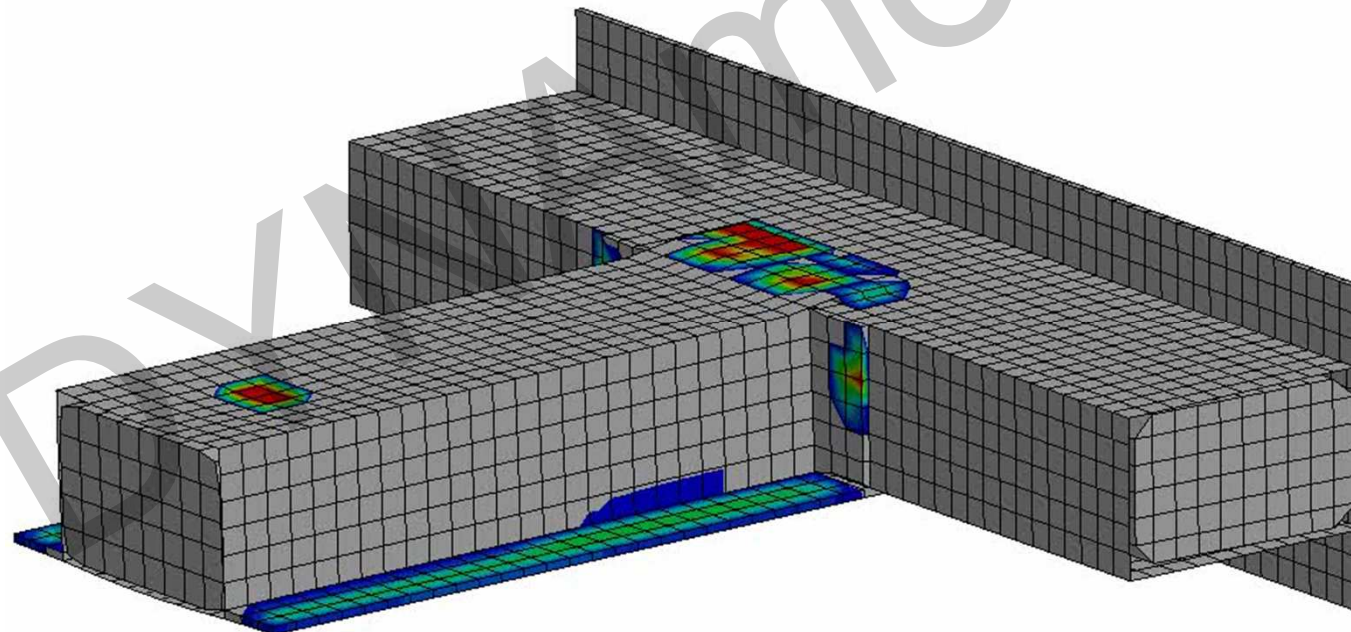
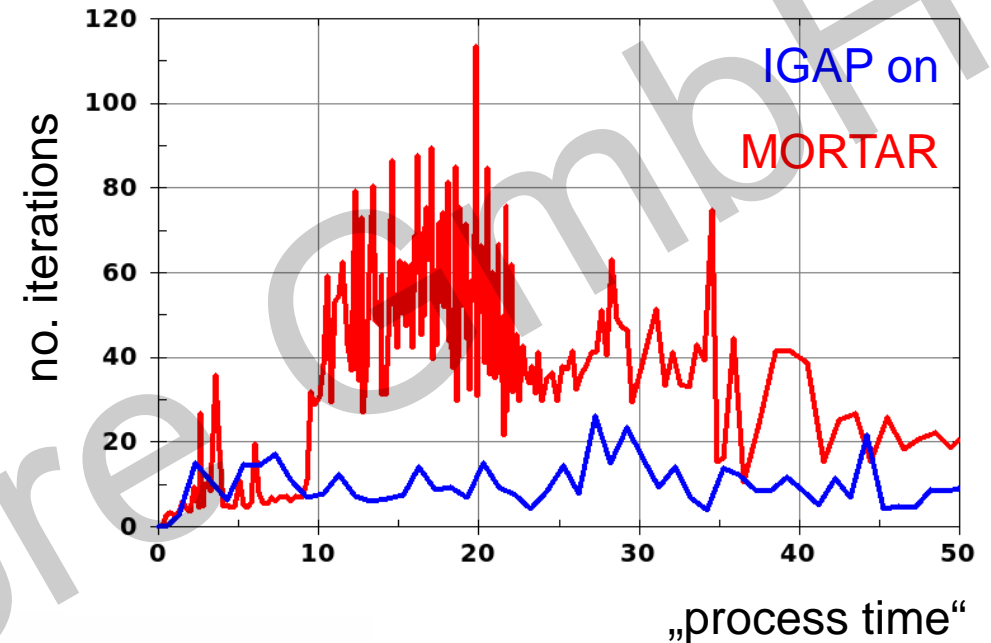
- More realistic results with Mortar contact
- 5 different phases can be observed: no contact (i), tipping (ii), elastic bending (iii), adhesive softening (iv), and glue failure (v)



damage  
(0 - 1)

## Static implicit with Mortar contact

- Convergence becomes more difficult
- Reason(s) for difficulties can be detected with special “iteration plot database” d3iter
- Evolution of out-of-balance forces during iteration process shows critical areas



residual forces

Troubles from damage evolution in cohesive material and contact to impactor

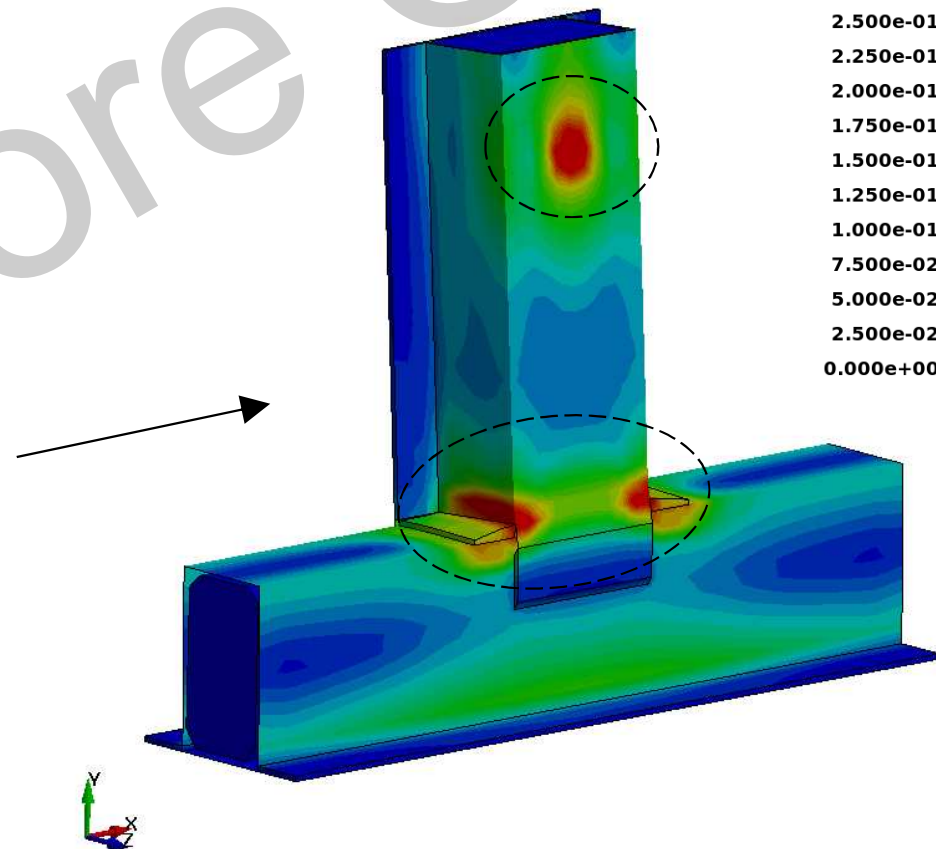
## Ideas for improvement

- Perhaps Full Newton better suited for this problem (ILIMIT=1)
- Modify other implicit settings (timestep size, tolerances, ...) or contact parameters (IGAP, )
- But maybe better to improve the model itself:
- Replacement for cohesive material (MAT\_186 with smooth curve?)
- Mesh refinement in critical areas?
- Dynamic implicit – very slow
- ...

**T-component**  
Time = 22.54  
Contours of Effective Stress (v-m)  
max IP. value  
min=0, at elem# 350000  
max=0.816933, at elem# 200419

Fringe Levels

5.000e-01
4.750e-01
4.500e-01
4.250e-01
4.000e-01
3.750e-01
3.500e-01
3.250e-01
3.000e-01
2.750e-01
2.500e-01
2.250e-01
2.000e-01
1.750e-01
1.500e-01
1.250e-01
1.000e-01
7.500e-02
5.000e-02
2.500e-02
0.000e+00



- Explicit analysis runs into its limits for long duration processes or even real static load cases.
- Therefore, implicit analysis is often preferable. Actually, computation time can be decreased in many cases.
- But: more demanding to get a solution, especially if large deformations, contact, and nonlinear material behavior is involved.
- Users must be aware of crucial differences between explicit (e.g. time step size) and implicit (e.g. “smooth” model)
- Often greater effort is needed to obtain a functional model in implicit, but also the feeling of success is greater in the end