Recent developments in *DEFINE_PRESSURE_TUBE for simulating pressure tube sensors in pedestrian crash

Jesper Karlsson, DYNAmore Nordic AB

Special thanks to Ulf Westberg at Volvo Car Corp.

The pressure tube sensor

- Designed to detect collisions with pedestrians
- Air filled silicone tube embedded in bumper
- Pressure sensors at ends detect collision
- Reveals extent/location of impact

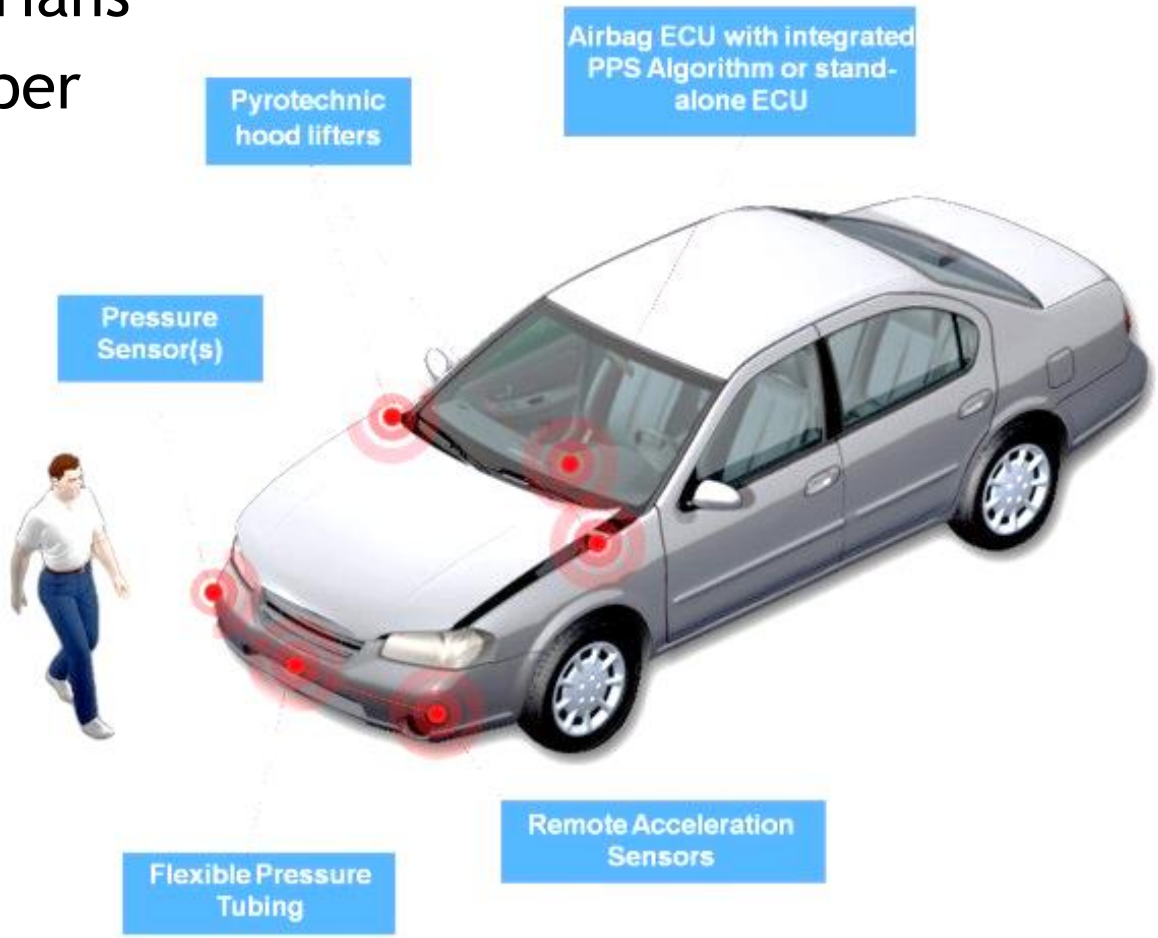


Photo: ZF TRW.

The basics

- Goal: simulate acoustic pressure waves in a thin long tube
  - Outer diameter ~8 mm
  - Inner diameter ~4 mm
  - Length ~2 m
  - 1D model seems appropriate

- Method: acoustic approximation of 1D compressible Euler equations
  - Pressure: $(Ap)_t + p_0(Au)_x = 0$
  - Velocity: $(Au)_t + Ac^2 p_x / p_0 = 0$
  - Density given by sound speed $c = \sqrt{p/\rho}$
  - Area depends on time and space
  - Constant area gives regular wave-equation: $p_{tt} = c^2 p_{xx}$

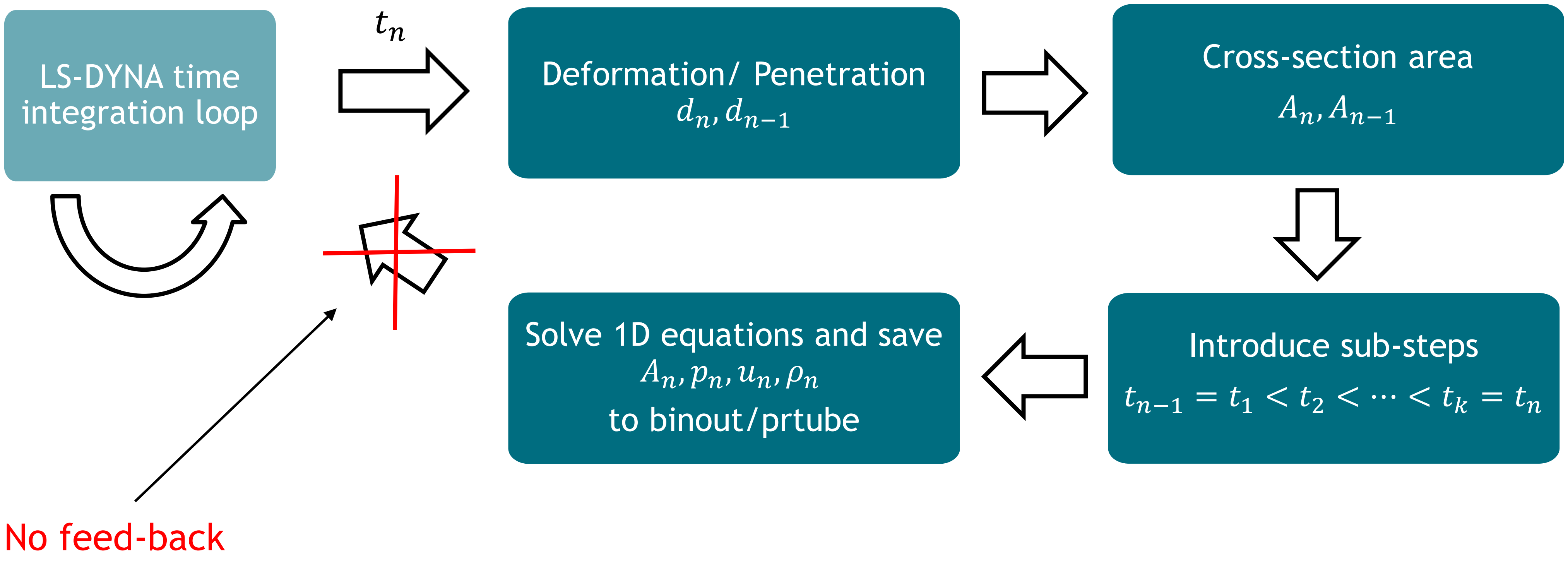
Implementation

- Tube defined by tubular beam elements
  - Variation in tube cross-section area drives pressure evolution
  - Cross-section area given by either
    - tube contact penetration, or
    - deformation of automatically generated solid/shell tube
  - One-way coupling between tube compression and air pressure
  - Output data saved in beam nodes

- *DEFINE_PRESSURE_TUBE
  - PID: Beam element part
  - WS: Wave propagation speed
  - PR: Initial gas pressure

Card 1	1	2	3
Variable	PID	WS	PR
Type	I	F	F
Default	0	0.0	0.0

Solver schematics



Numerics

- Continuous Galerkin in space (artificial diffusion and linear damping)
- Heun's method in time (2nd order Runge-Kutta)
- CFL condition for stability

$$\Delta t < \min_i \frac{\Delta x_i}{\Delta x_i |A_t/A| + 3c}$$

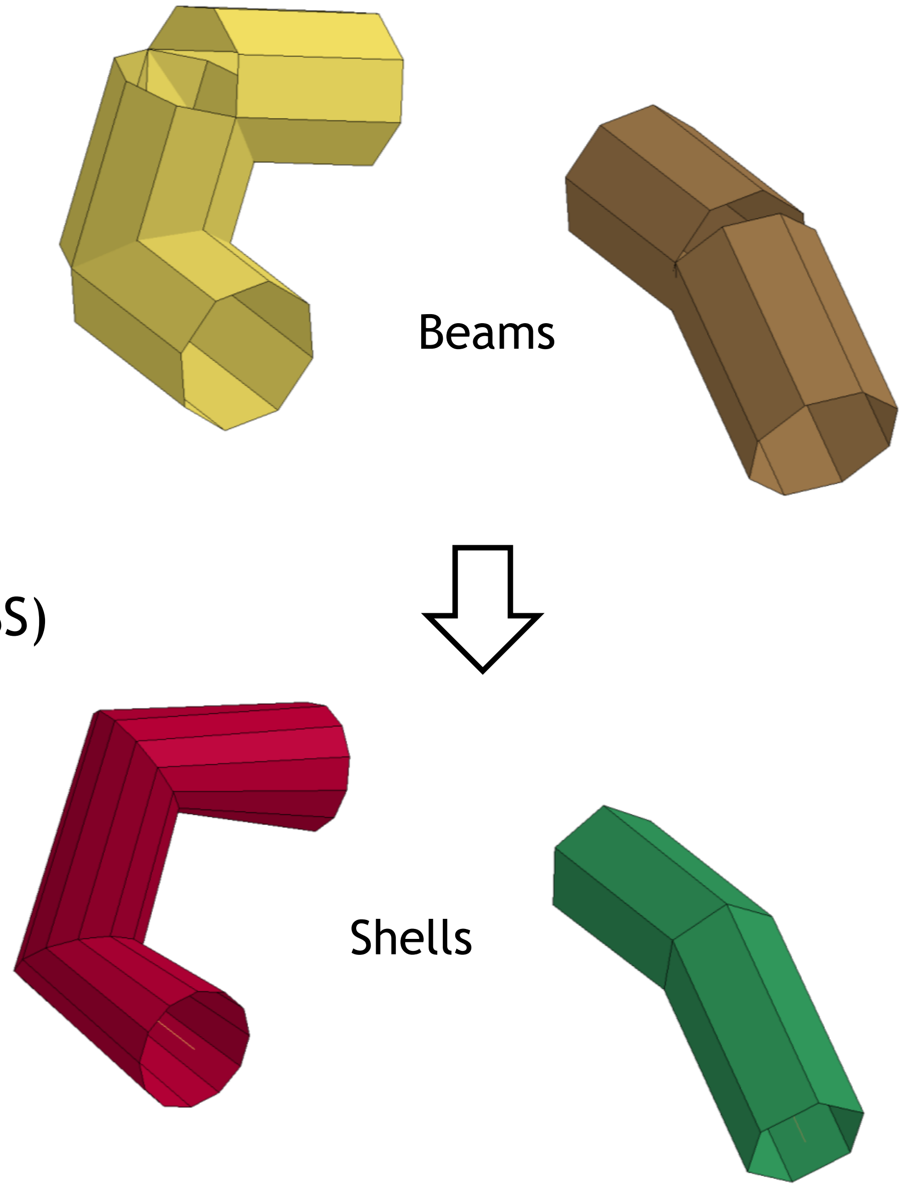
- CFL-condition fulfilled by substepping
  - Does not affect global step
  - Substep changes in time depending on A_t/A
- Tube algorithm uses initial beam element length Δx_i

Pros and cons

- Pros
  - Simple and extremely efficient
- Cons
  - No feedback to mechanical solver
  - Pressure solver only uses radial tube compression
  - Complex geometries like sharp bends, bifurcations, etc, not possible

New features

- New features
  - Automatic generation of shell/solid element tube
    - Better radial response
    - Cross-section area given by nodal displacements instead of contact penetration
  - Varying thickness over length (*ELEMENT_BEAM_THICKNESS)
  - New boundary conditions: open/closed ends



- Future development
  - More boundary conditions
  - Handling cavities at ends
  - Pressure feedback to mechanical solver

Keywords - *DEFINE_PRESSURE_TUBE

- Compulsory
  - PID: Beam element part
  - WS: Wave propagation speed
  - PR: Initial gas pressure
- Optional
  - MTD: Solution methods
  - TYPE:
    - 0 = beam elements
    - 1 = convert to shell elements
    - 2 = convert to solid elements
  - VISC: Artificial viscosity factor
  - CFL: Time step factor
  - DAMP: Linear damping
  - BNDL/BNDR: Boundary conditions
    - 0 = closed end
    - 1 = open end

Card 1	1	2	3	4	5
Variable	PID	WS	PR	MTD	TYPE
Type	I	F	F	I	I
Default	0	0.0	0.0	0	0

Optional card

Card 2	1	2	3	4	5
Variable	VISC	CFL	DAMP	BNDL	BNDR
Type	F	F	F	F	F
Default	1.0	0.9	0.0	0.0	0.0

Keywords - *DEFINE_PRESSURE_TUBE

- Optional shell card
  - NSHL: No. shells on circumference
  - ELFORM: shell element type
  - NIP: int. pts. through thickness
  - SHRF: shear correction factor
  - BPID: new PID for beams
- Optional solid card
  - NSLD: No. solids on circumference
  - ELFORM: solid element type
  - NTHK: no. solids through thickness
  - BPID: new PID for beams

Optional shell card if TYPE=1

Card 3a	1	2	3	4	5
Variable	NSHL	ELFORM	NIP	SHRF	BPID
Type	F	F	F	F	I
Default	12.0	16.0	3.0	1.0	optional

Optional solid card TYPE=2

Card 3b	1	2	3	4	5
Variable	NSLD	ELFORM	NTHK		BPID
Type	F	F	F		I
Default	12.0	1.0	3.0		optional

Keywords - considerations

- *SECTION_BEAM
  - Only ELFORM=1,4,5,11 with CST=1, i.e. tubular beams
  - Initial tube area using inner beam radius if >0, otherwise outer beam radius
- Geometric constraints
  - Each set of joint beam elements in a part will model a separate closed tube
  - Different parts used in *DEFINE_PRESSURE_TUBE cards may not share beam nodes
  - No junctions allowed
- MPP
  - All beam elements in one part will be on same processor

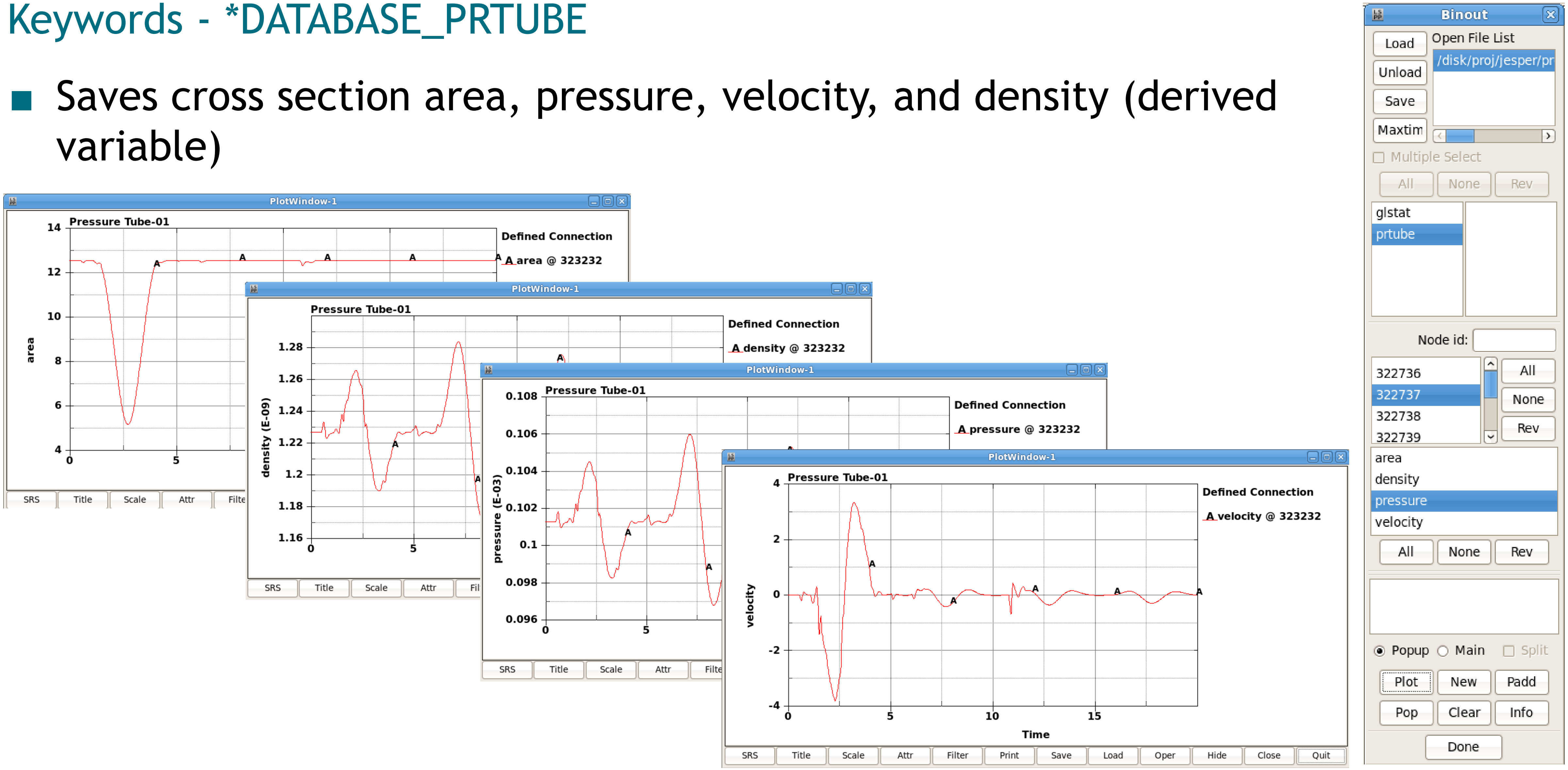
Keywords – beams vs. shells

- Beam tube (TYPE=0)
  - Only works with Mortar contacts
  - Uses contact penetration to calculate area
  - Unphysical radial response - depending on contact stiffness only

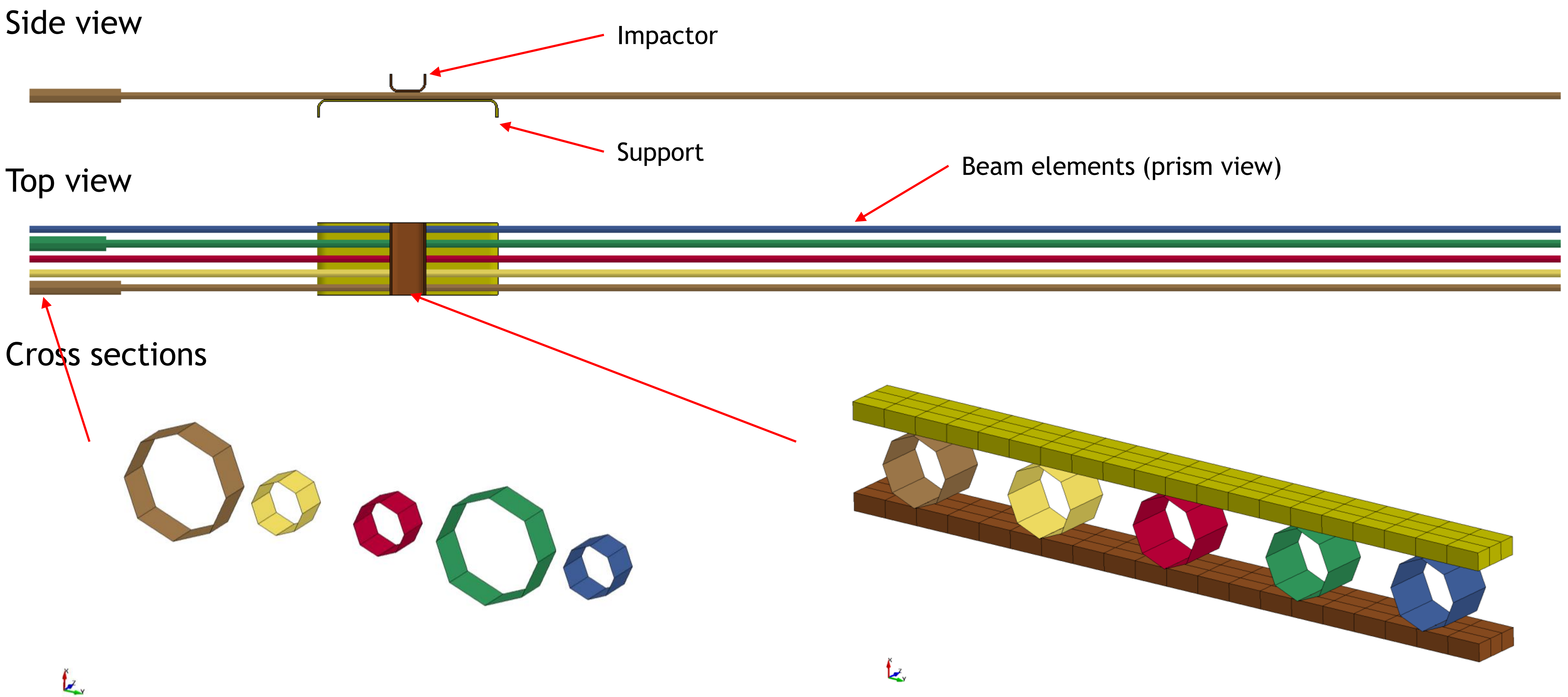
- Shell/solid tube (TYPE=1/2)
  - Shell/solid tube (new *PART, *SECTION, *ELEMENT) is created from beam geometry
  - Shells/solids get beam PID and beams are moved to new PID
  - Contacts and boundary conditions now applies to shells/solids instead of beams
  - Beam part only exists to store pressure solver data
  - Works with all contacts
  - Uses nodal postions to calculate area
  - Better radial response

Keywords - *DATABASE_PRTUBE

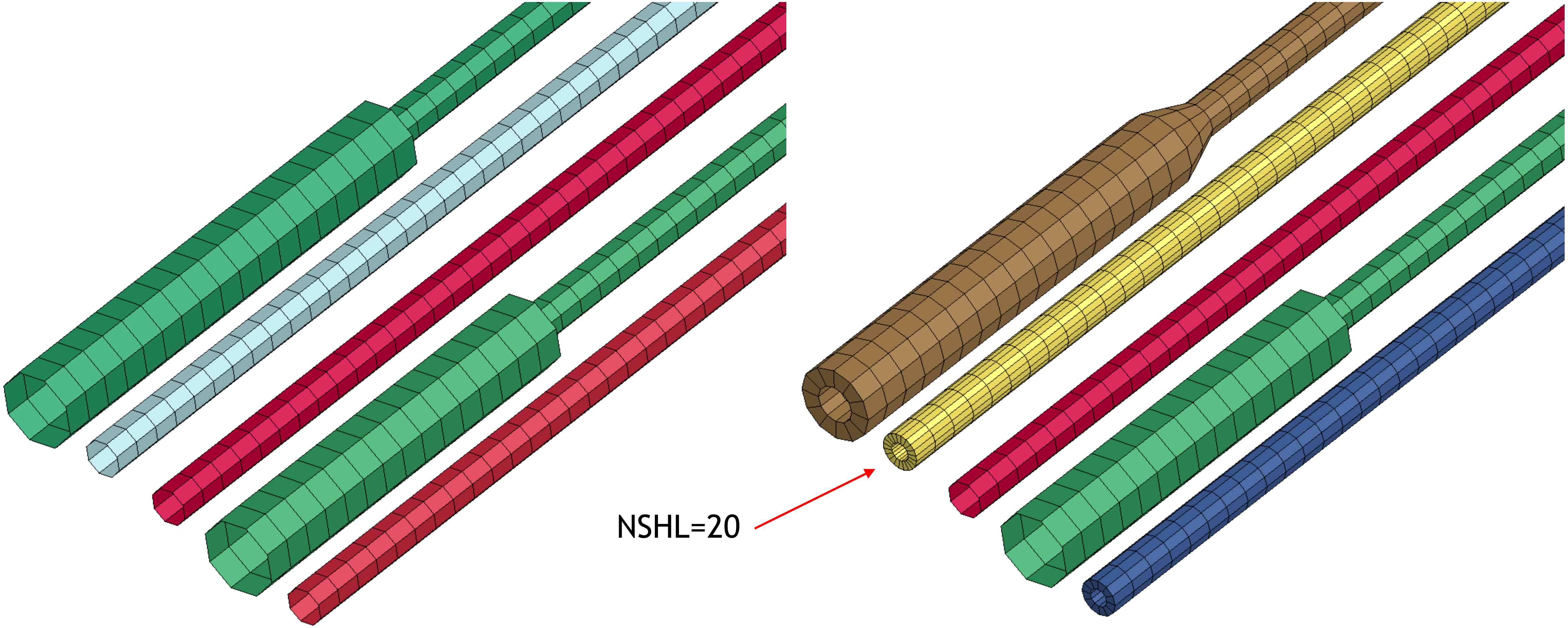
- Saves cross section area, pressure, velocity, and density (derived variable)



Example model – beam elements of varying thickness



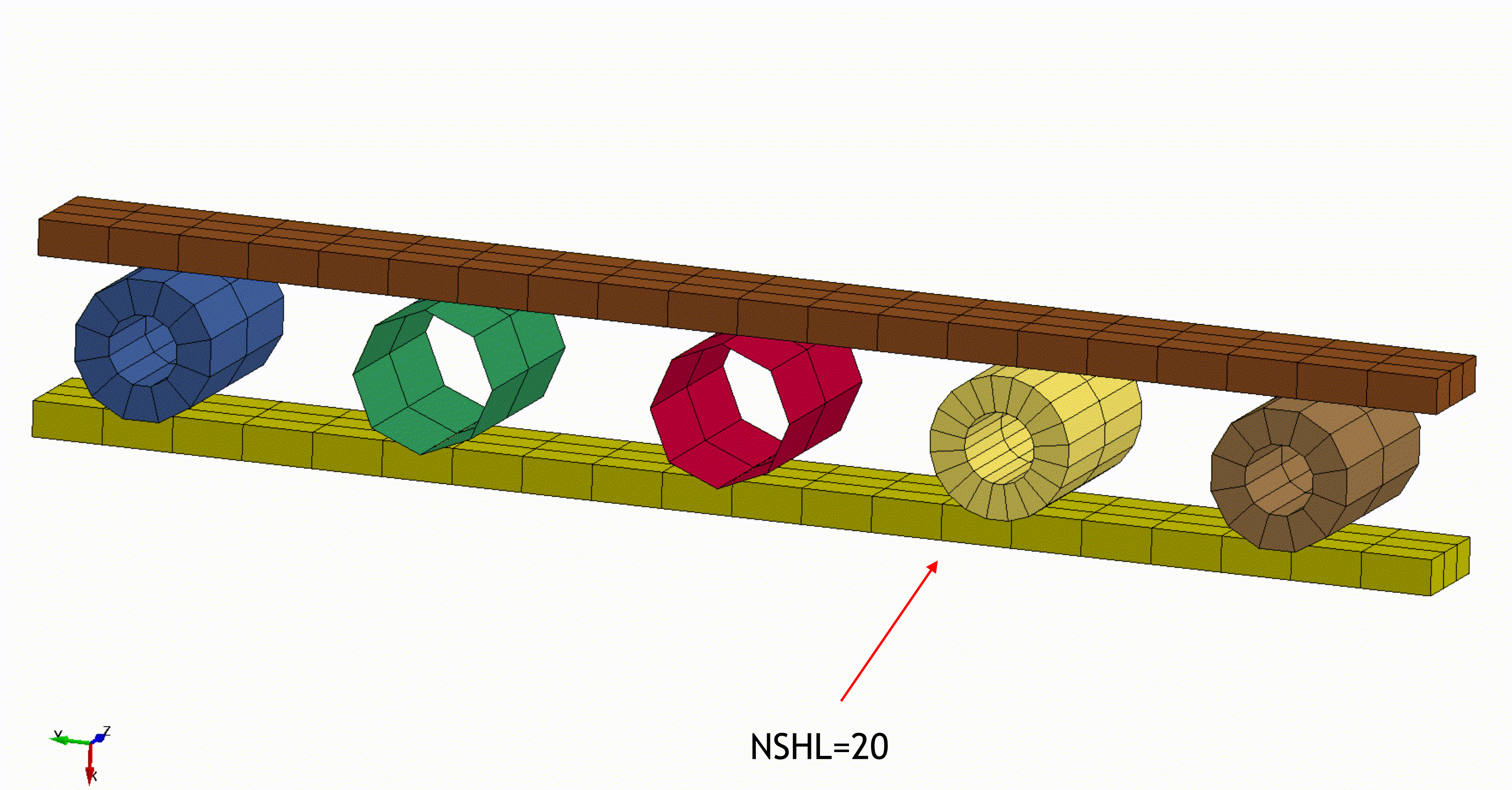
Example model – automatic conversion to shell elements



Beam elements

Beam and shell elements

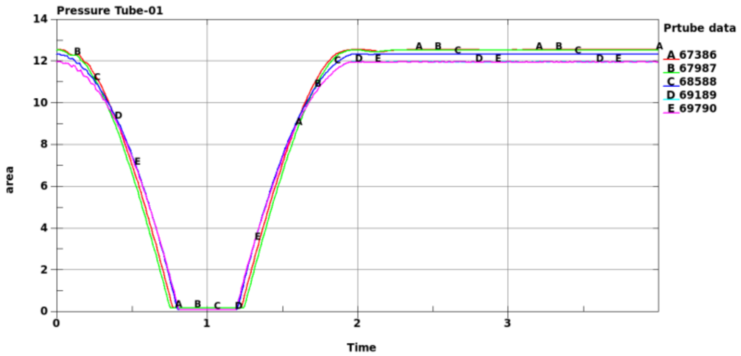
Example model – thickness compression



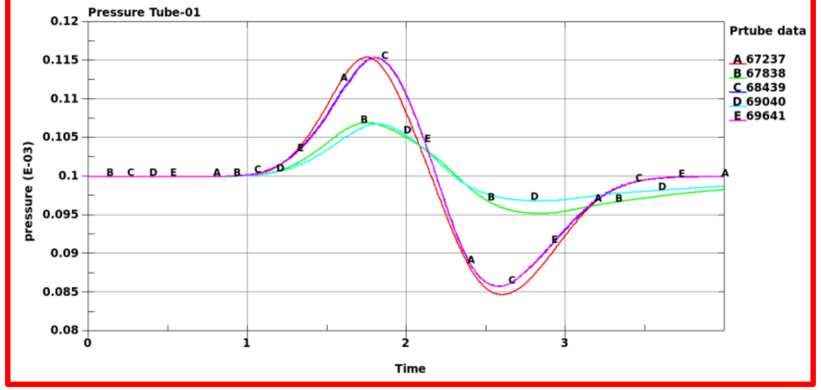
NSHL=20

Example model – cross-section area and pressure

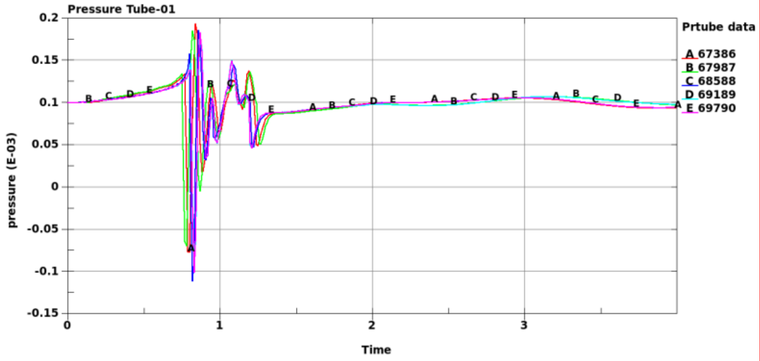
Mid cross-section area



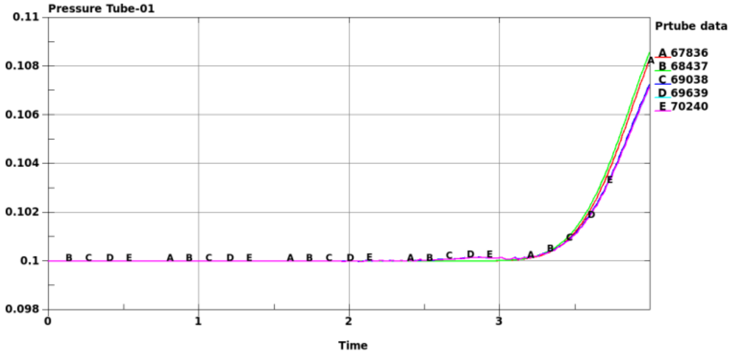
Left pressure



Mid pressure

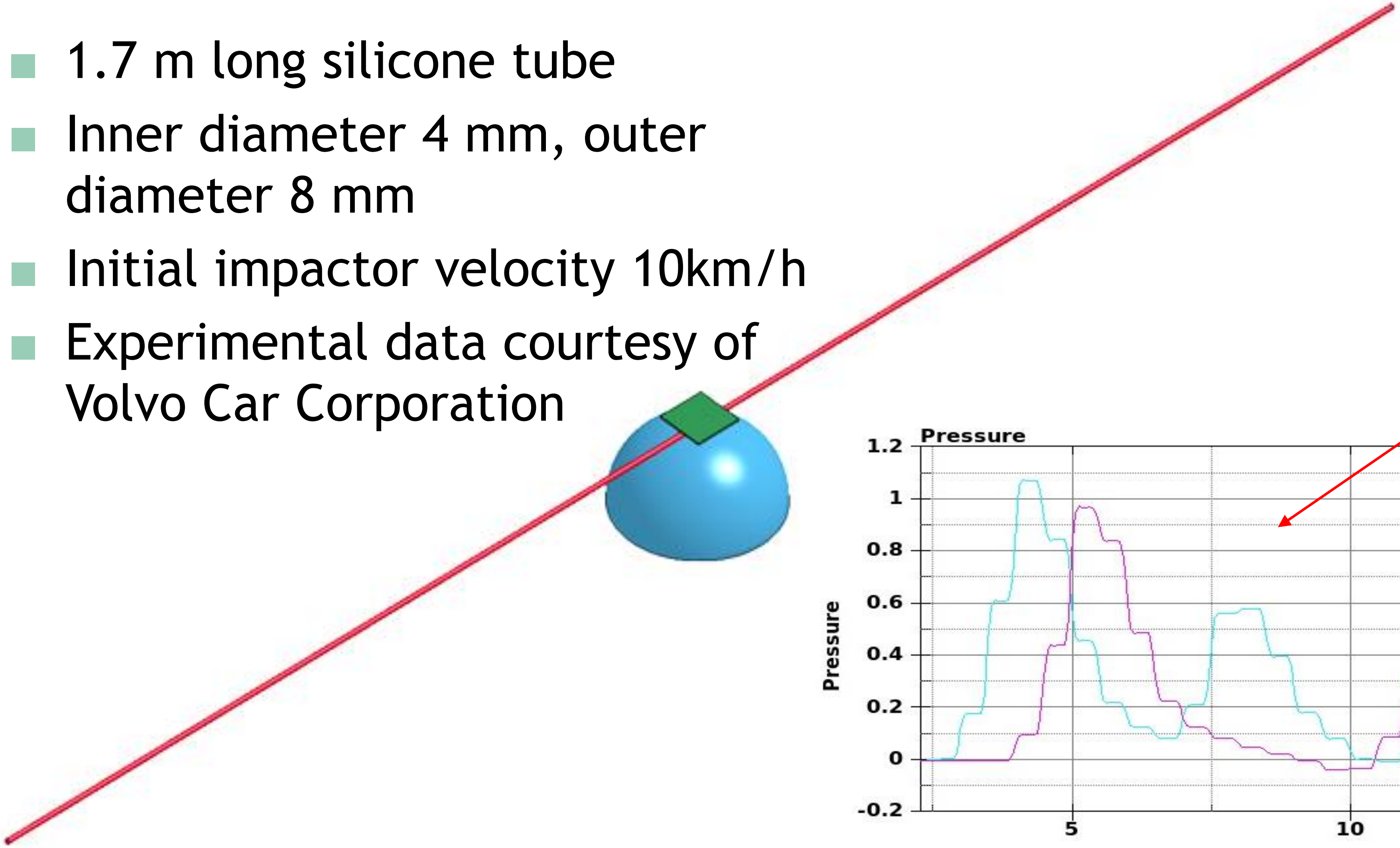


Right pressure

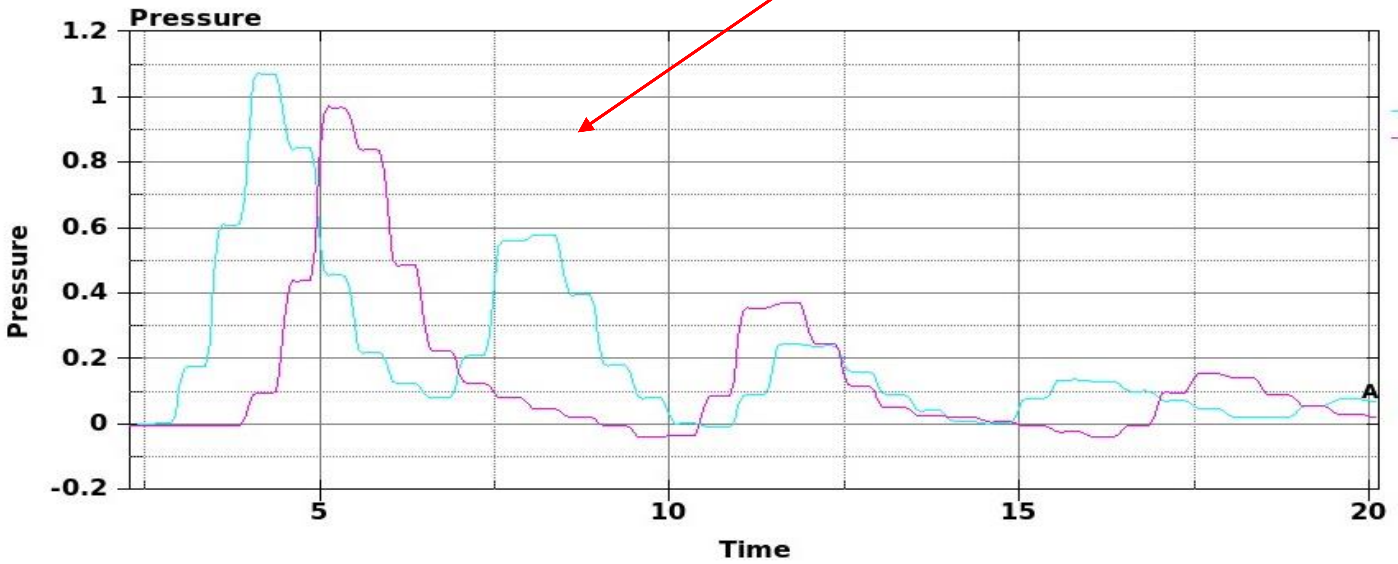


Example 1 – drop test

- 1.7 m long silicone tube
- Inner diameter 4 mm, outer diameter 8 mm
- Initial impactor velocity 10km/h
- Experimental data courtesy of Volvo Car Corporation

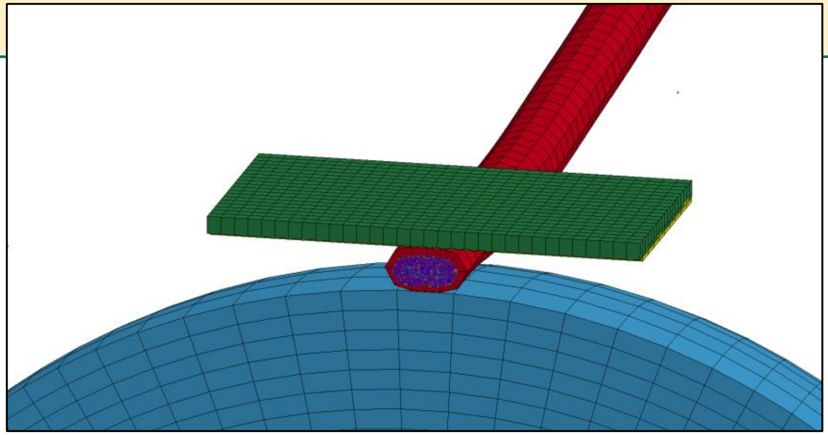


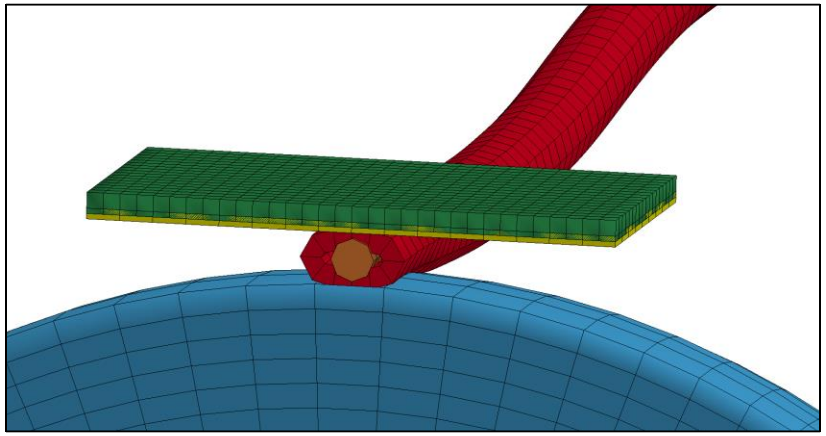
Indicates DAMP>0

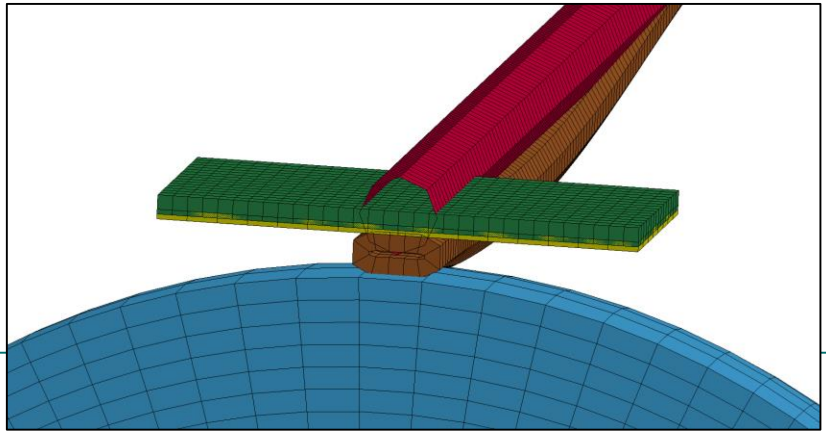


Example 1 - approaches

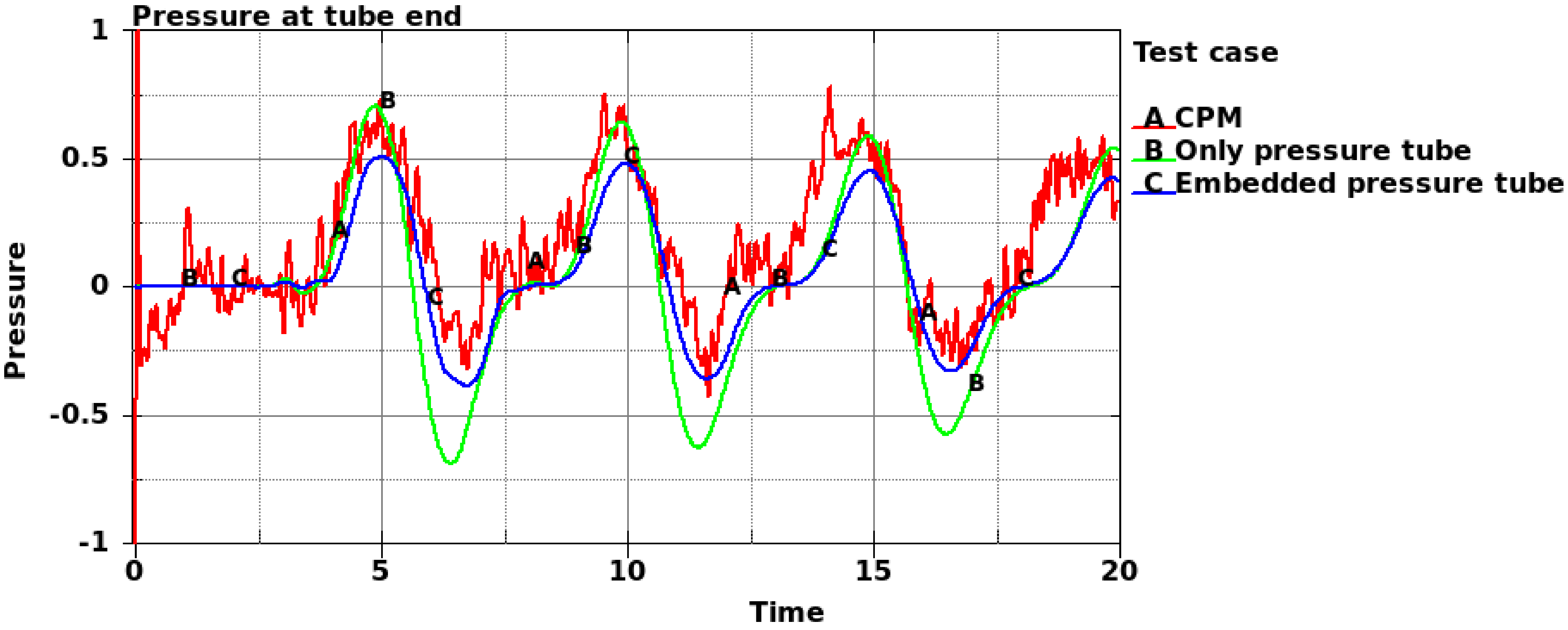
- Corpuscular Particle Method
  - Closed shell tube with gas domain inside
  - 2 million particles needed
  - 170 hours total CPU time
- Beam elements only
  - Tube and air modeled by beam elements
  - 10 minutes total CPU time
- Shell tube with embedded beam elements
  - Tube modeled with shell elements
  - Air modeled with beam elements
  - 4 hours total CPU time
- Automatically generated shell tube
  - 1 hour total CPU time



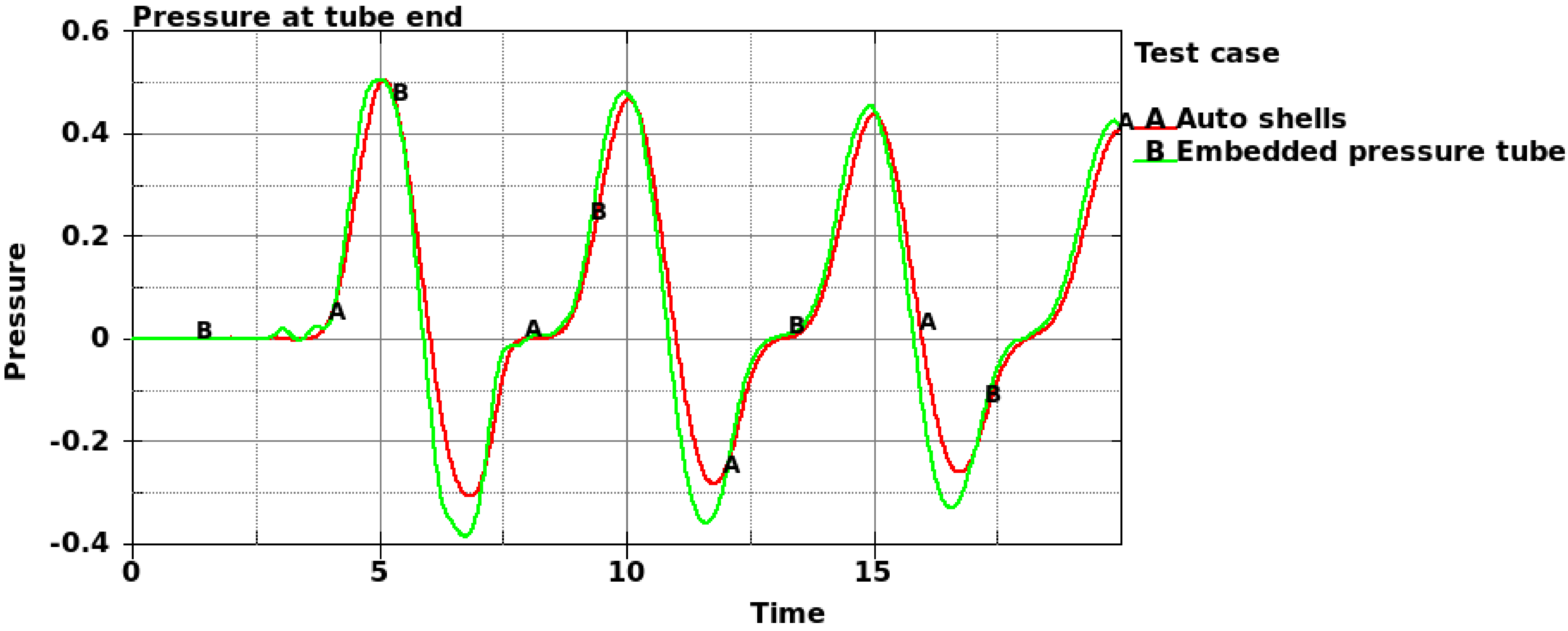




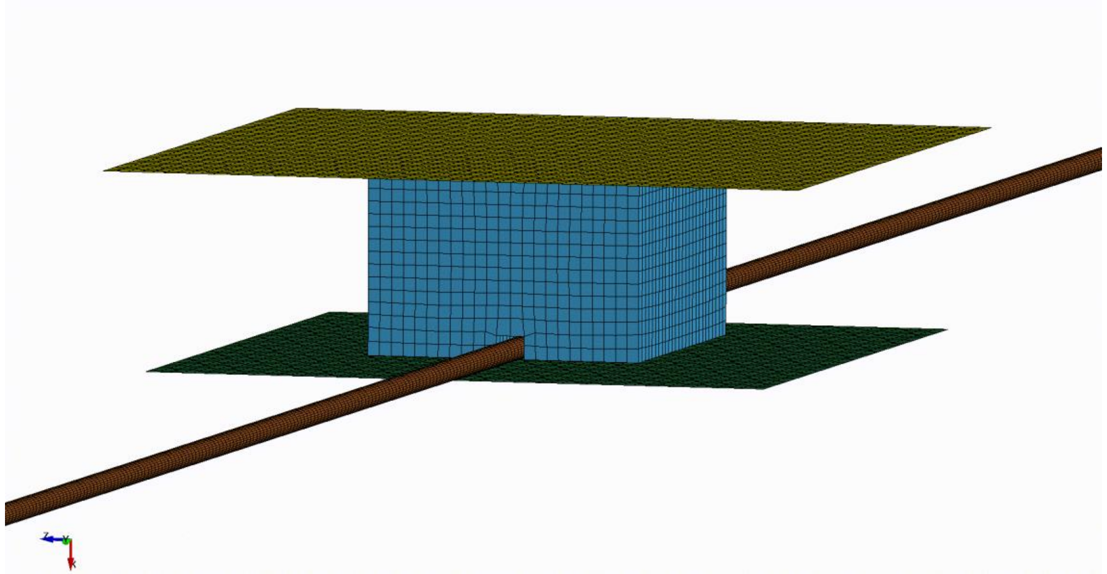
Example 1 – pressures

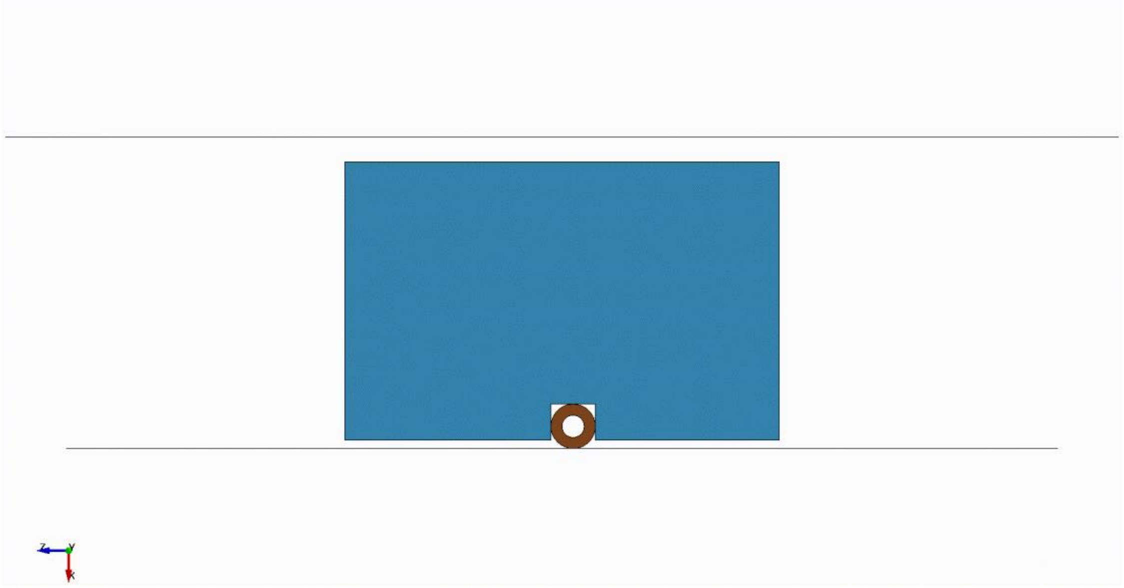


Example 1 – automatic shell tube

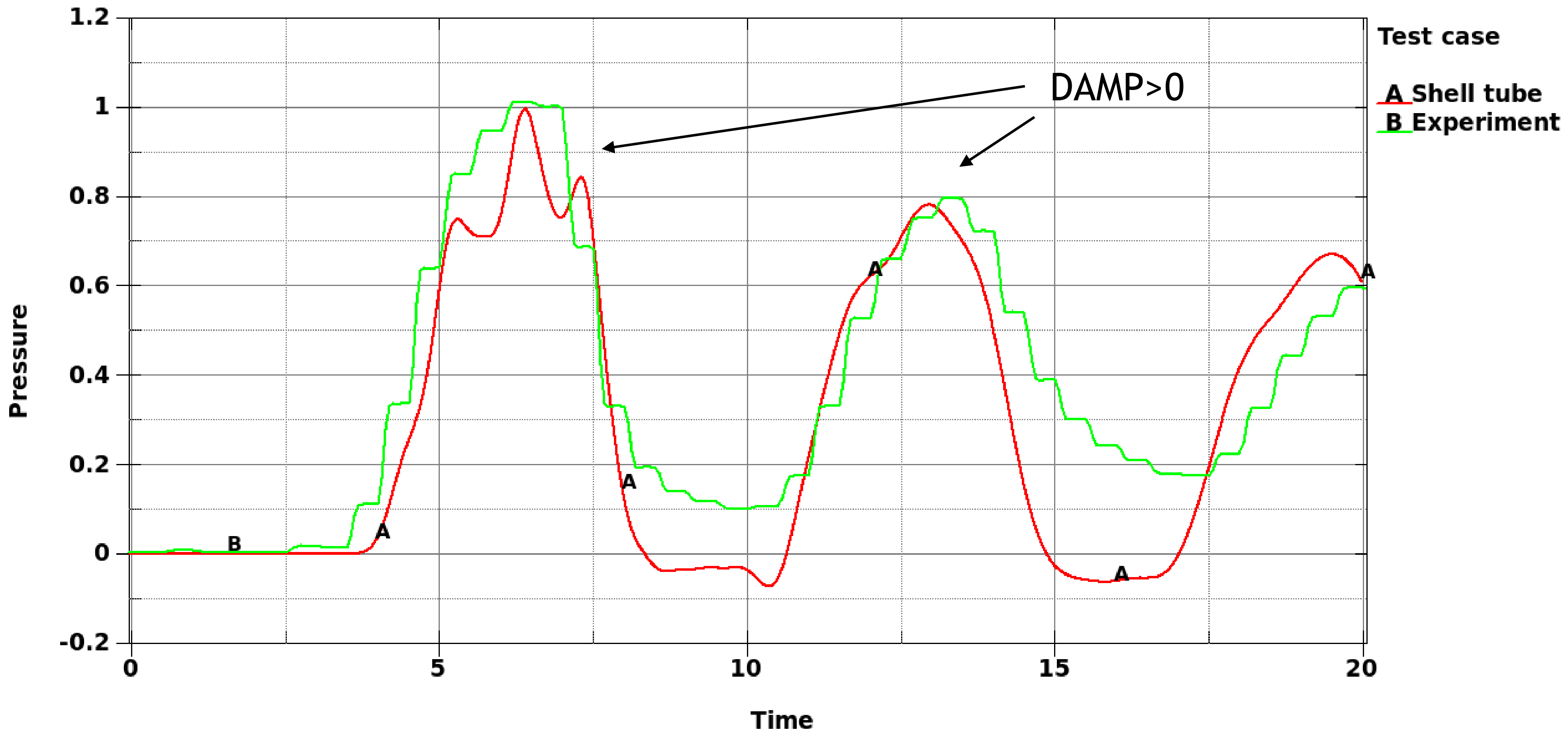


Example 2 - foam





Example 2 - foam



Summary

- Pros
  - Very simple to use
  - Very efficient (ex. 15 min for beam tube, 1 hour for shell tube, 170 hours for CPM)
- Cons
  - Only one-way coupling to mechanical solver
  - Only 1D model
- Open issues
  - Boundary conditions
  - Cavities at ends

Thank you!

Your LS-DYNA distributor and more