Recent Airbag CPM Enhancement

10/15/2018

Jason Wang

jason@lstc.com

LSTC
Livermore Software Technology Corp.

Outline

- **Enhancement**
  - **MAT_ADD_AIRBAG_POROSITY_LEAKAGE*
  - *Inflator orifice area*
  - *Deflection coefficient defined by a FPRIC vs pressure curve*
  - *Vent options (*DEFINE_CPM_VENT)*
    - *Internal vent with cone angle*
  - *R9 scalability and repeatability*
  - *CPM performance improvement*
  - *Special decomposition*
- **Conclusions**

LSTC
Livermore Software Technology Corp.

*MAT_ADD_AIRBAG_POROSITY_LEAKAGE

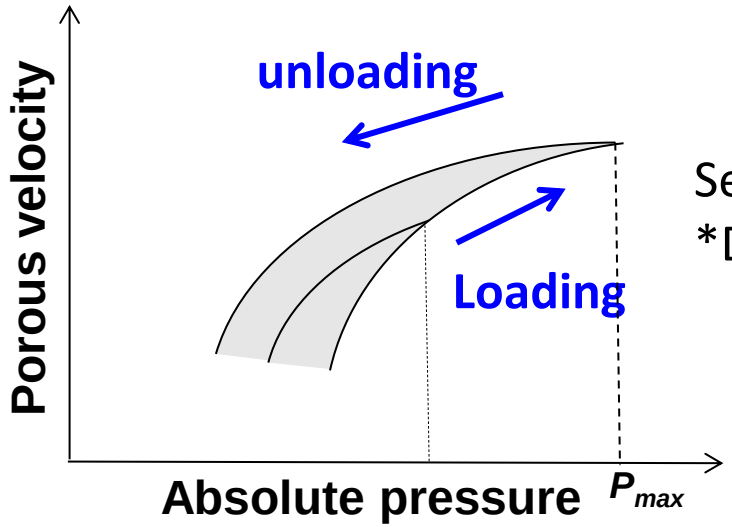
```
*MAT_ADD_AIRBAG_POROSITY_LEAKAGE
$     mid      flc      fac      ela    fvopt       x0       x1
```

- Similar options as the 3rd card of *MAT_34
- If this is used with *MAT_34, the options in this card has higher priority than *MAT_34
- It can be used with any *MAT to define leakage

- *fvopt=7 or 8 for CPM (pressure vs. velocity)*

LSTC
Livermore Software Technology Corp.

*MAT_ADD_AIRBAG_POROSITY_LEAKAGE



Seam leakage, fac for
*DEFINE_FUNCTION

- "fac" can be defined as a factor, a load curve under *MAT_34 and *MAT_ADD
- For CPM, "fac" can be defined as a *DEFINE_FUNCTION under *MAT_ADD card
  - User can control Loading/unloading with different porosity velocity curves
  - Input: part absolute pressure, time
    Output: velocity
  - Extended values, upon request

LSTC
Livermore Software Technology Corp.

*MAT_ADD_AIRBAG_POROSITY_LEAKAGE

```
*MAT_FABRIC
$#     mid        ro        ea        eb        ec      prba      prca      prcb
        19 8.7600E-7  0.300000  0.200000  0.300000  0.200000  0.200000  0.200000
$#     gab       gbc       gca       cse        el       prl    lratio      damp
  0.040000  0.040000  0.040000  1.000000  0.060000  0.350000  0.100000  0.200000
$3.0000000     1.000    -17420
$#    aopt       flc       fac       ela      lnrc      form     fvopt    tsrfac
  3.000000
------
*MAT_ADD_AIRBAG_POROSITY_LEAKAGE
$      mid       flc       fac       ela     fvopt        x0        x1
        19       1.0       -99                 7       1.0
```

```
*DEFINE_FUNCTION
99
float fac(float p, float time)
{  float x, y, pa=101325.15, pmax, frcf;
  pmax=3.0*pa; frcf=471;
  x = (p*1e9 - pa)/1e6;
  y=0.013*x*x+0.47*x+1.3;
  return frcf*y;     }
```

LSTC
Livermore Software
Technology Corp.

*AIRBAG_PARTICLE

*AIRBAG_PARTICLE

.....

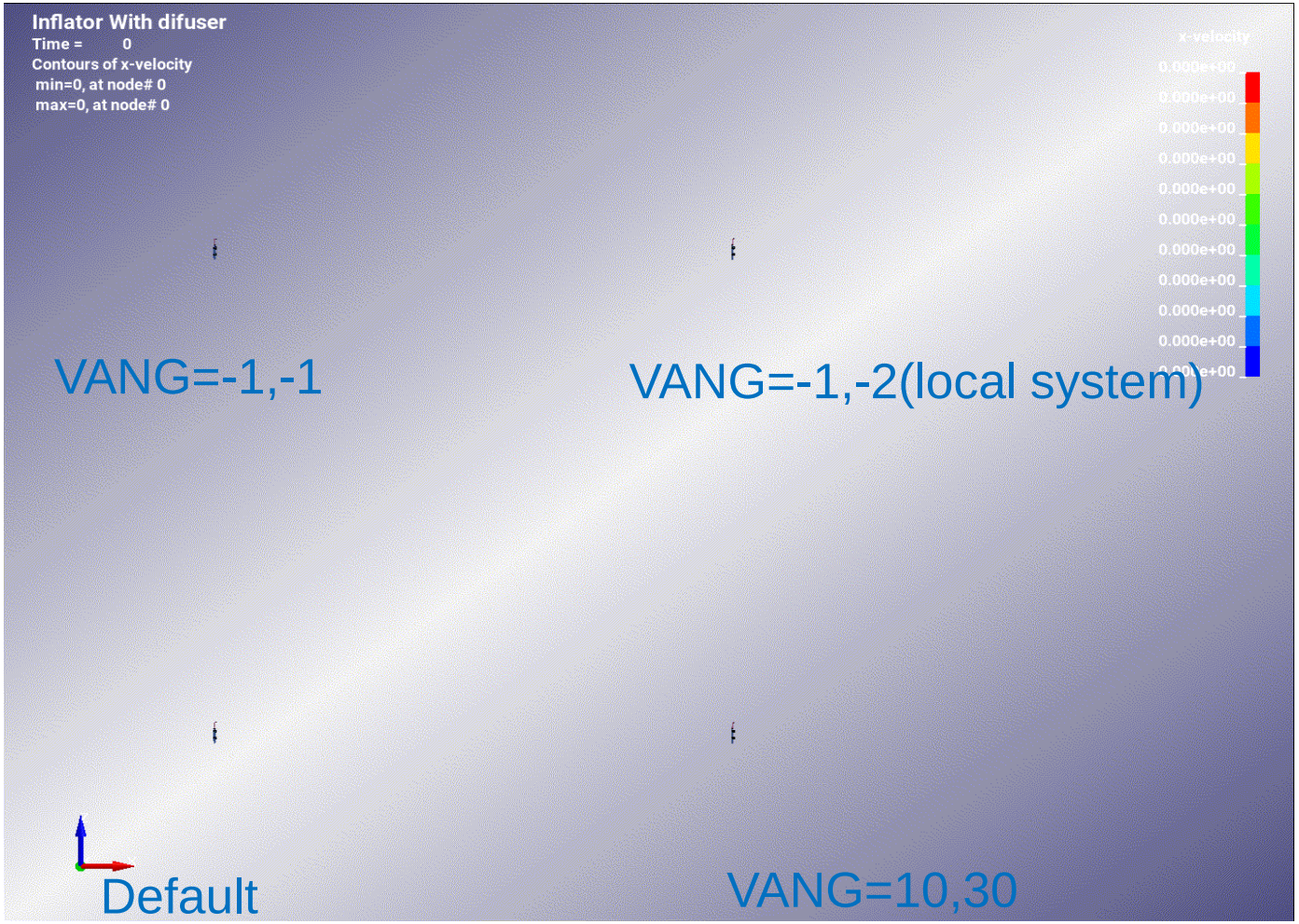
$Orifice cards

$ NID ANi Vdi CAi INFOi

- $AN_{info} = \sum ANi$
- $AN_i \ / AN_{info}$ is used to distribute the mass among orifice
- AN*i* > 0, the value is the orifice area
- AN*i* < 0, the abs(AN*i)* is load curve ID for orifice area vs. time. The mass will be adjusted based on $AN_i(t)/ANinfo(t)$ during run time.

LSTC
Livermore Software Technology Corp.

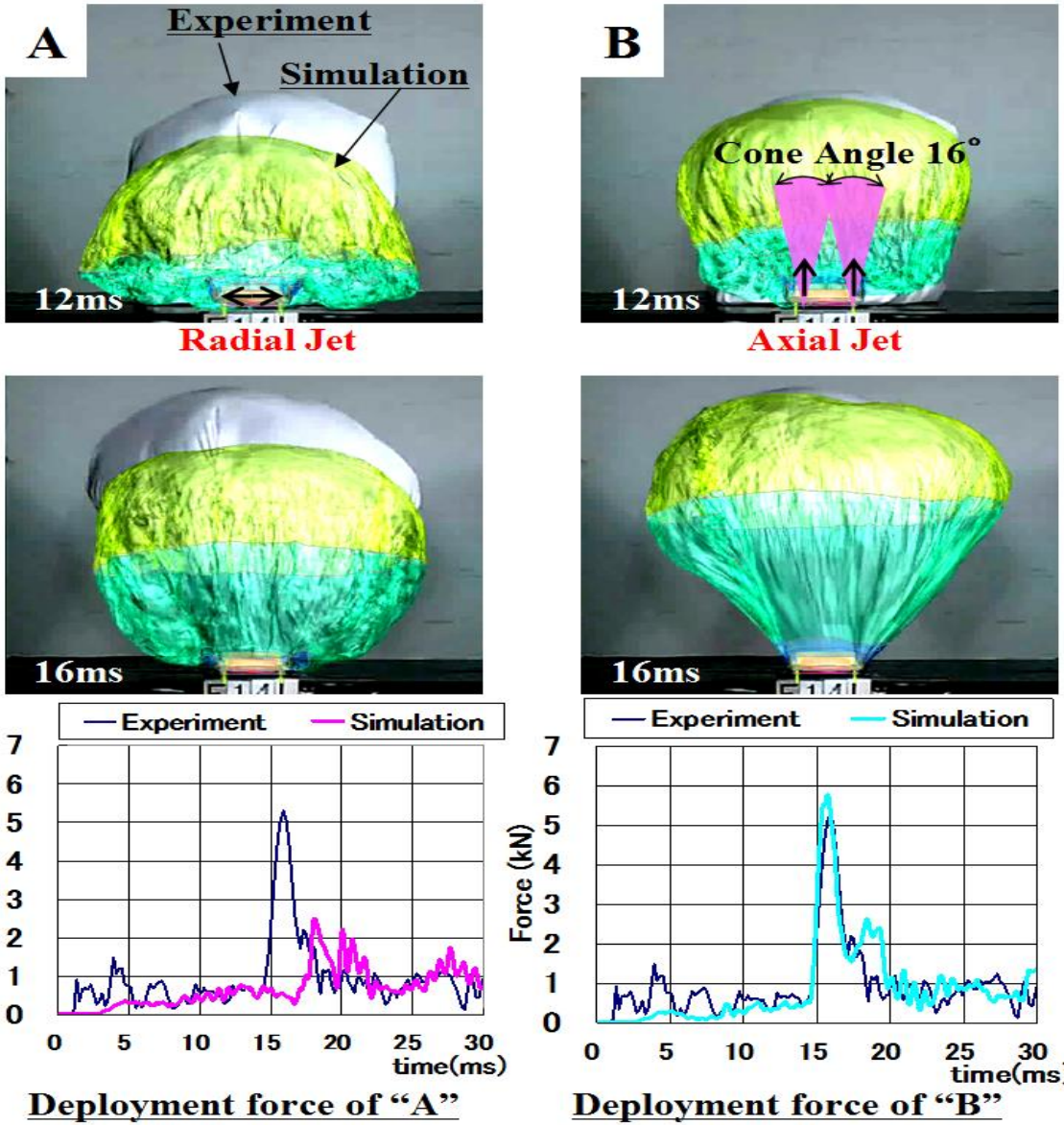
*DEFINE_CPM_VENT
Internal vent with cone angle



Particles tends to have very uniform space distribution passing the internal vent which loss the "jetting" behavior observed in the tests

LSTC
Livermore Software Technology Corp.

*DEFINE_CPM_VENT
Internal vent with cone angle



This new option greatly helps to improve the correlation between tests and simulations

1. Cone angle is defined by using above keyword card.
2. Additional option VANG=-1 will allow code to adjust the release based on the vent condition

H. Ida, M. Aoki, M. Asaoka, K. Ohtani,"A Study of gas flow behavior in airbag deployment simulation",24th International Technical Conference on the Enhanced Safety of Vehicles(ESV). No. 15-0081, 2015.

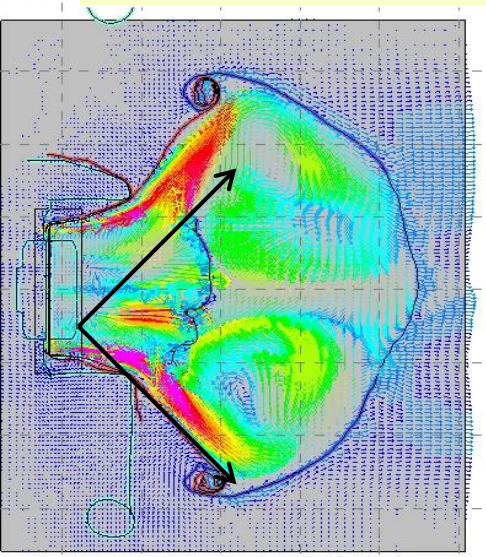
LSTC
Livermore Software Technology Corp.

Benchmark DAB Models

ALE

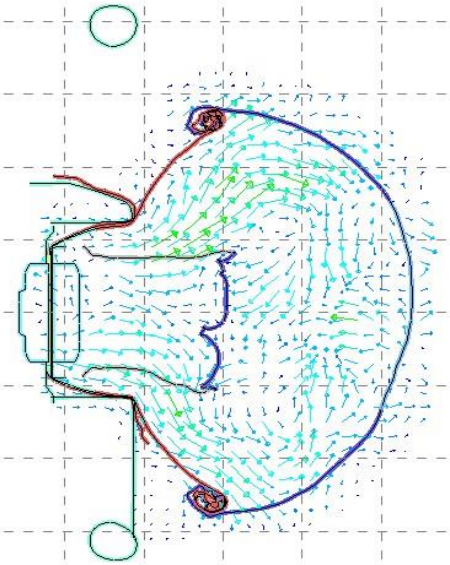
D3PLOT: M1: DAB ALE 06

Estimated gas flow redirection angles (45^0)



CPM: NO VANG

D3PLOT: M4: DAB CPM09

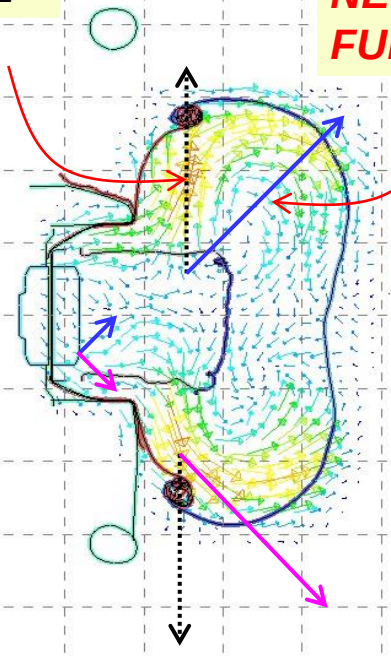


CPM: VANG=-1

D3PLOT: M3: DAB CPM10

VANG=-1

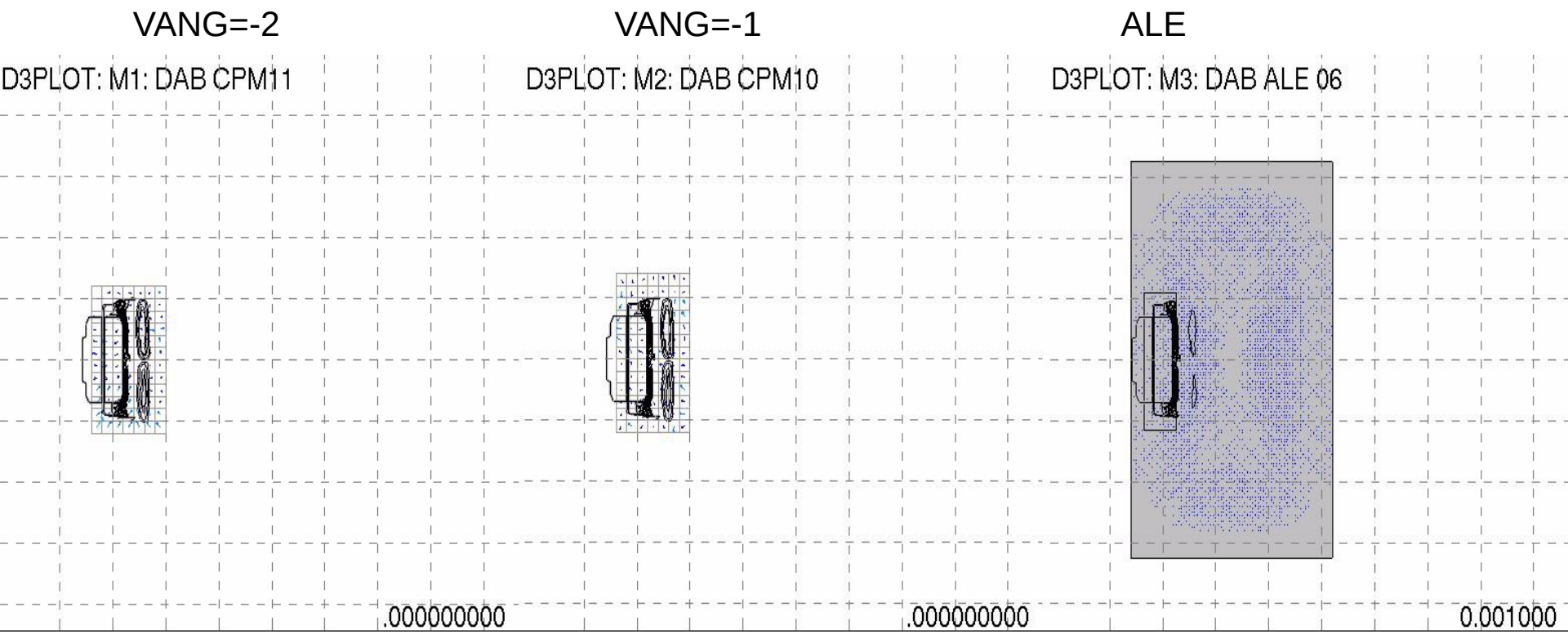
NEW VANG FUNCTION



In some airbags an inner fabric structure is used to redirect the gas flow to inflate the airbag in a certain way.

LSTC
Livermore Software Technology Corp.

*DEFINE_CPM_VENT
VANG=-2

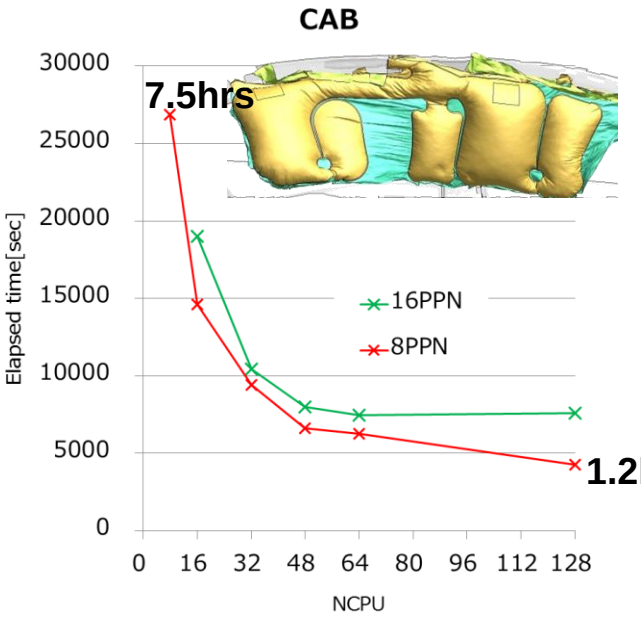


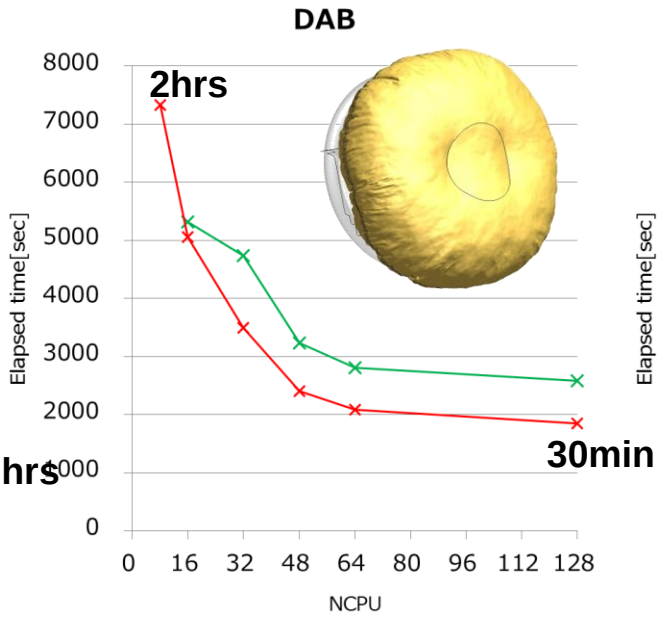
VANG=-2: user defines a local coordinate system for ‘jet’ to follow.

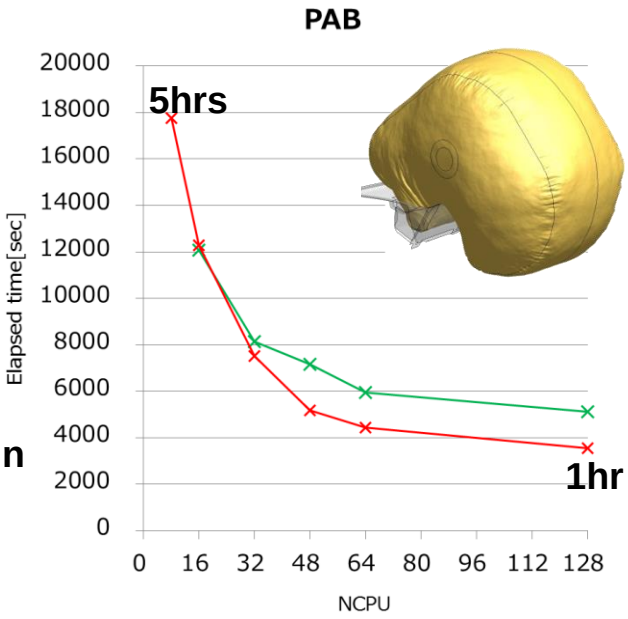
LSTC
Livermore Software
Technology Corp.

Scalability and repeatability R9

- Baseline airbag models created by JSOL/Arup for demo/research purposes.
  - CAB = curtain airbag, DAB = driver's airbag, PAB = passenger airbag
- All models have typical size, shape, inflator & fabric.
- All have been developed to be robust (insensitive, repeatable, not prone to error) and inflate with no issues.
- All models are tested with different number of cores.





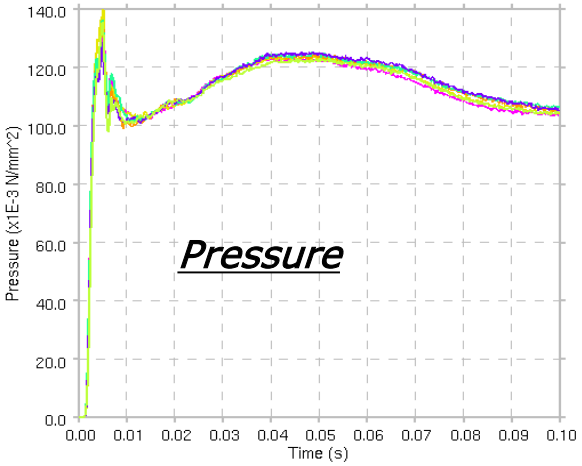


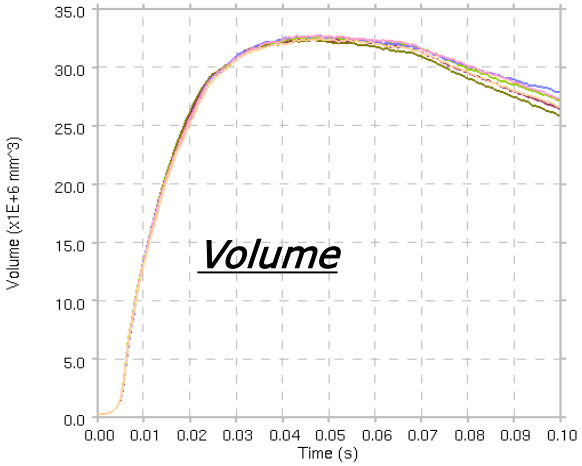
Courtesy of: Richard Taylor, Arup

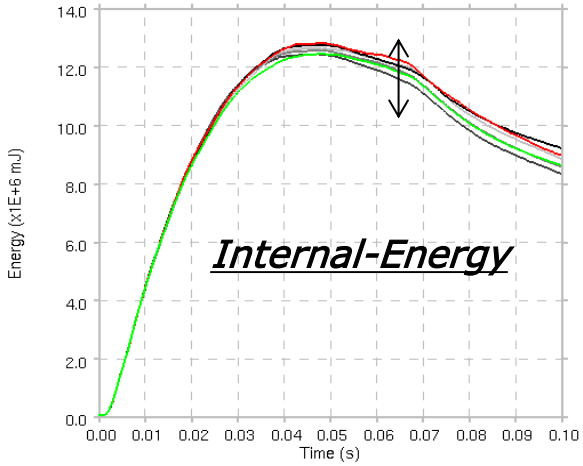
LSTC
Livermore Software Technology Corp.

Scalability and repeatability R9

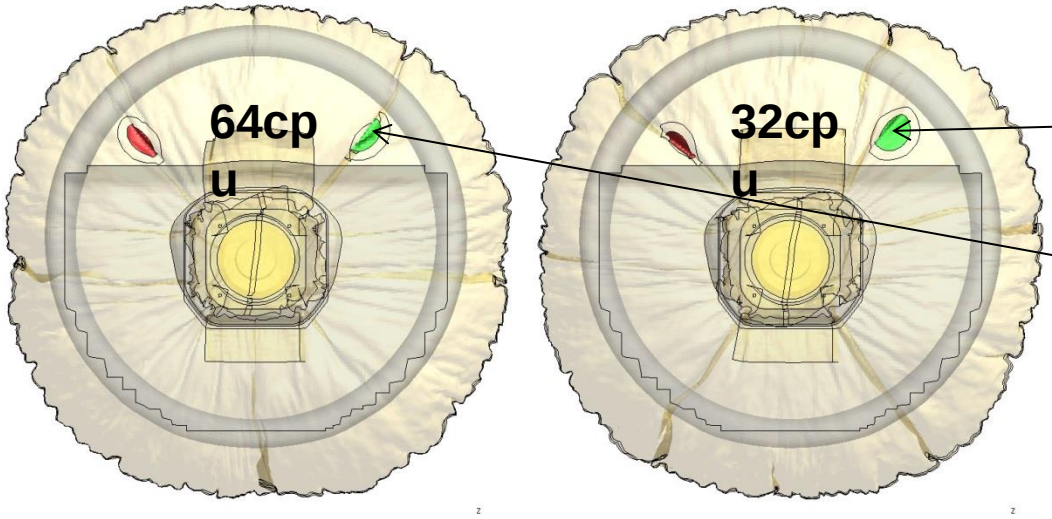
- The DAB model has two external vents, fabric and seam line porosity, all affected by contact blocking. Despite this results are very similar for all analyses.

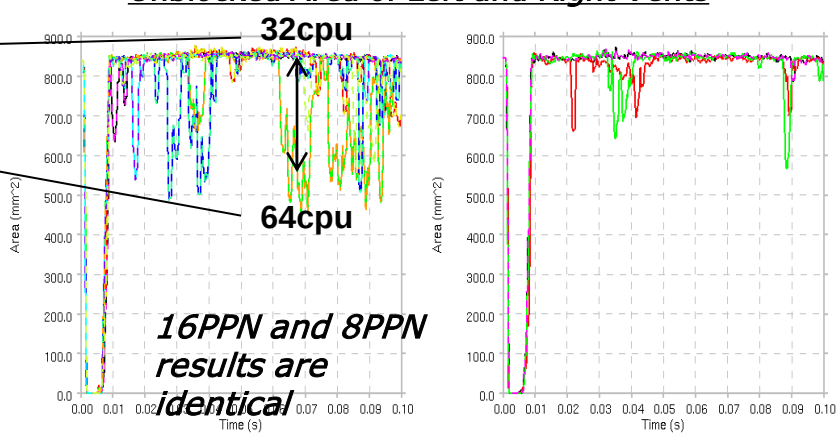






The slight difference in internal energy is due to different levels of vent contact blocking by different crease patterns.



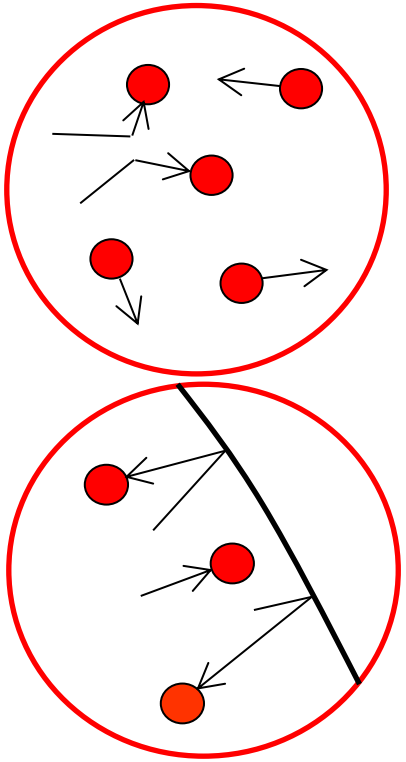


Improvement of execution speed

- Major cost for CPM airbag simulation
  - airbag self contact
  - particle to particle contact (p2p)
  - particle to fabric contact (p2f)

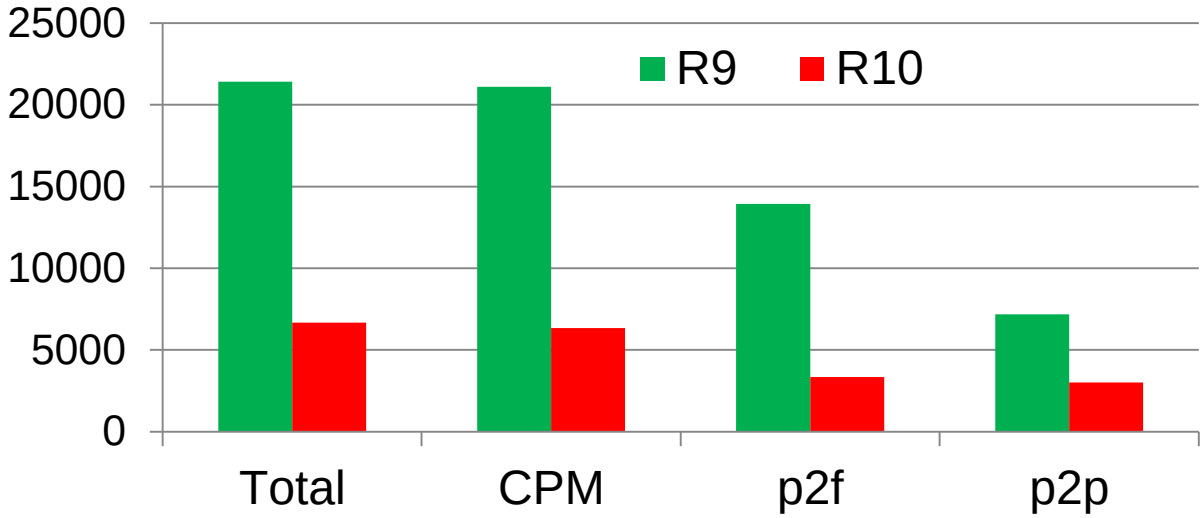
p2p
- nbody collision
- equal space to equal nbody

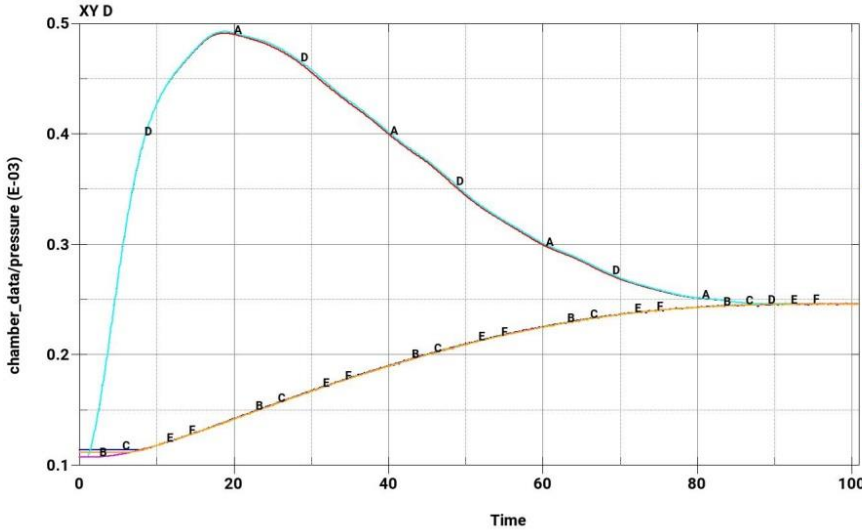
p2f
- node to surface bucket sort
- More efficient neighborhood search
- Improvement on communications



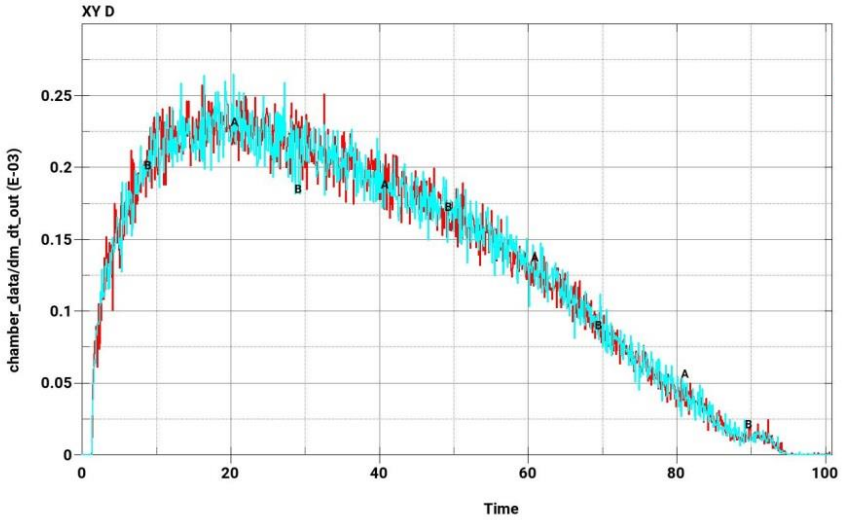
LSTC
Livermore Software Technology Corp.

Performance – Tank test (64 processors)





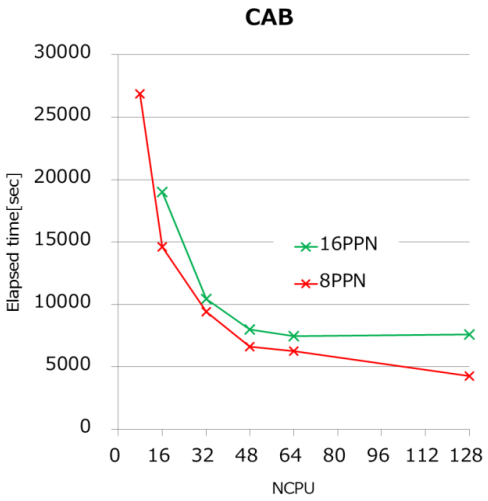
Chamber Pressure

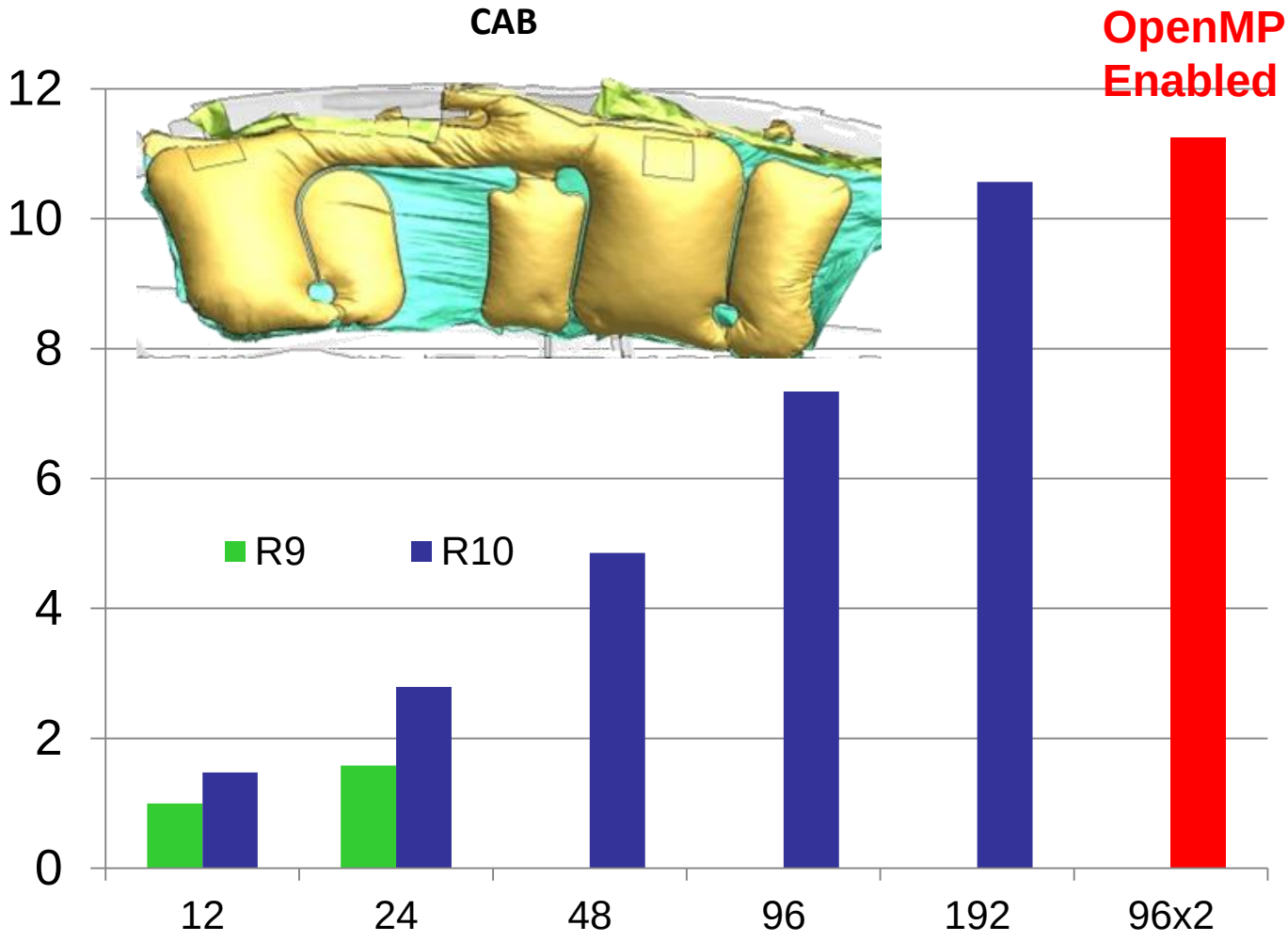


Vent rate

LSTC
Livermore Software
Technology Corp.

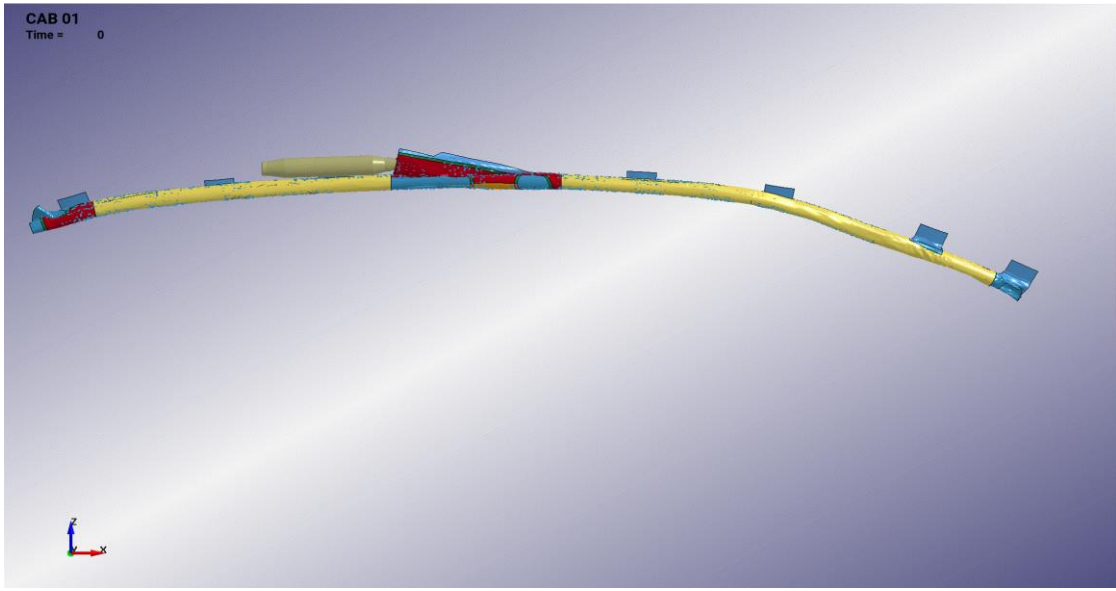
Scalability - CAB





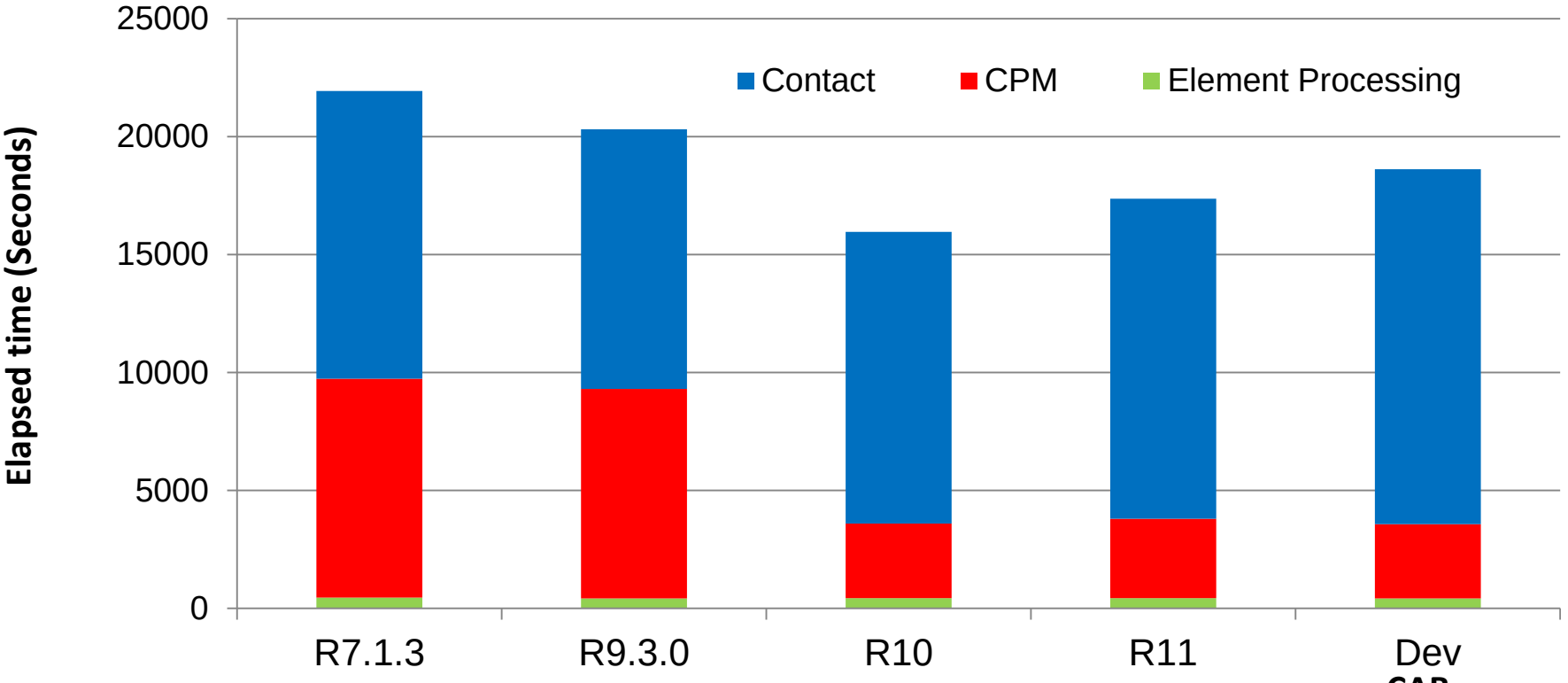
LSTC
Livermore Software Technology Corp.

CPU time improvement on CAB

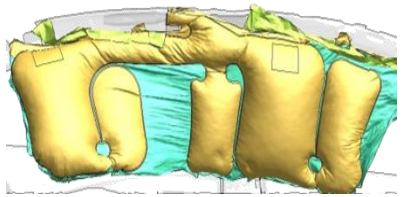


LSTC
Livermore Software Technology Corp.

CPU time of CAB by features

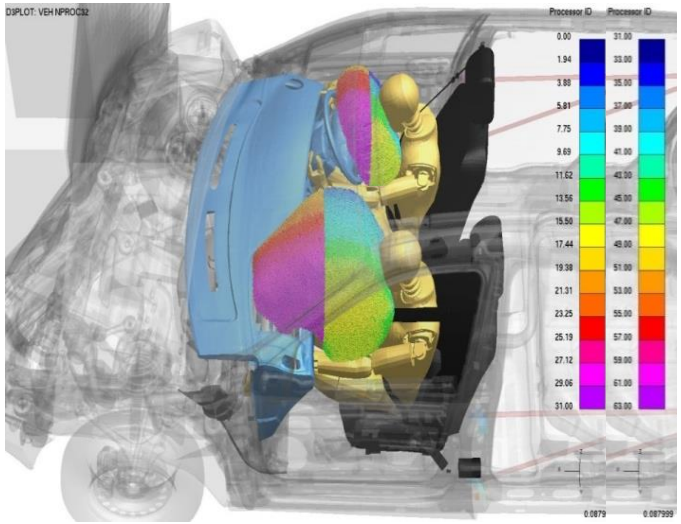


- Performance was measured with 96 processors
- CPM is about 3x faster from R7 to R10
- Self contact about the same
- The overall speed up is about ~20% for bag, ~5% for full car



LSTC
Livermore Software Technology Corp.

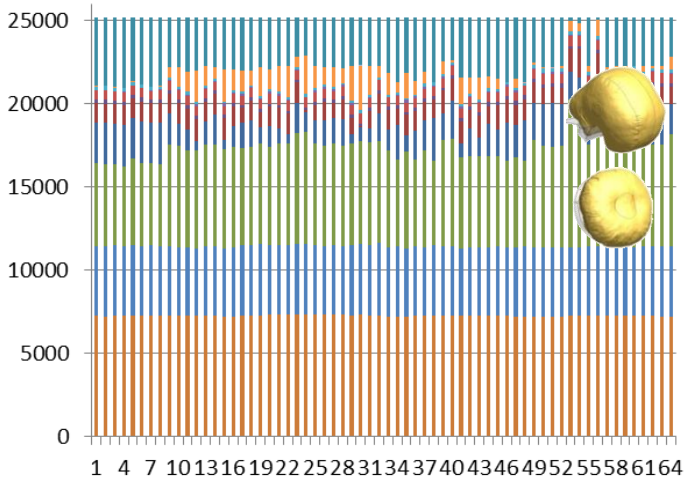
*CONTROL_MPP_DECOMPOSITION_ARRANGE_PARTS



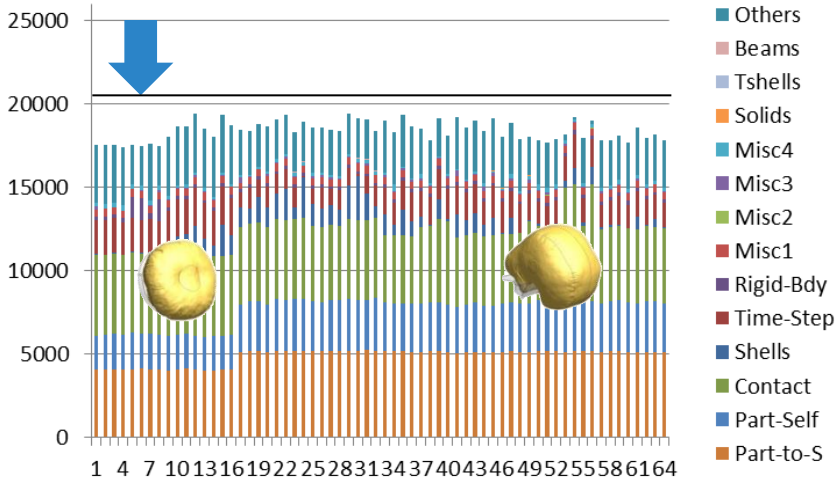
Runtime reduced from 7hrs 5min to 5hrs 53min. 20% faster!

- ***Bags in parallel***
- ***Bag self contacts in parallel***
- ***Contacts between bag and dummy in parallel***

- **set 16 for the DAB**
- **Set 48 for the PAB**



DAB & PAB 64cpu, Default



DAB & PAB 16/48 Distributed

Courtesy of: Richard Taylor, Arup

LSTC
Livermore Software Technology Corp.

Improvement of execution speed

- New algorithm is set as default method for R10.2(later), R11 and Dev binaries
- There is NO input change needed
- If one would like to test the old scheme, input needs to be modified
- New algorithm is about 3x faster for moderate processor counts (16 -64 cores). It will be even better with large core counts (>64 cores) from improved message passing.
- OpenMP enabled
- Due to other features in the model, the overall speed up is about 20%.

LSTC
Livermore Software Technology Corp.

Conclusions

- For simulation with more than 1 bag, please consider distribute each bag, its casing, dummy into particular group of processors
- HYBRID enabled - performance improvement on general and OpenMP features
- Heat sink/source effects are developed last week
- All new features are developed closely with customers and validated with tests. If you have any idea in mind, please share with me.
- jason@lstc.com

Thank you

LSTC
Livermore Software Technology Corp.