Recent MPP Development to Improve Consistency

J. Wang

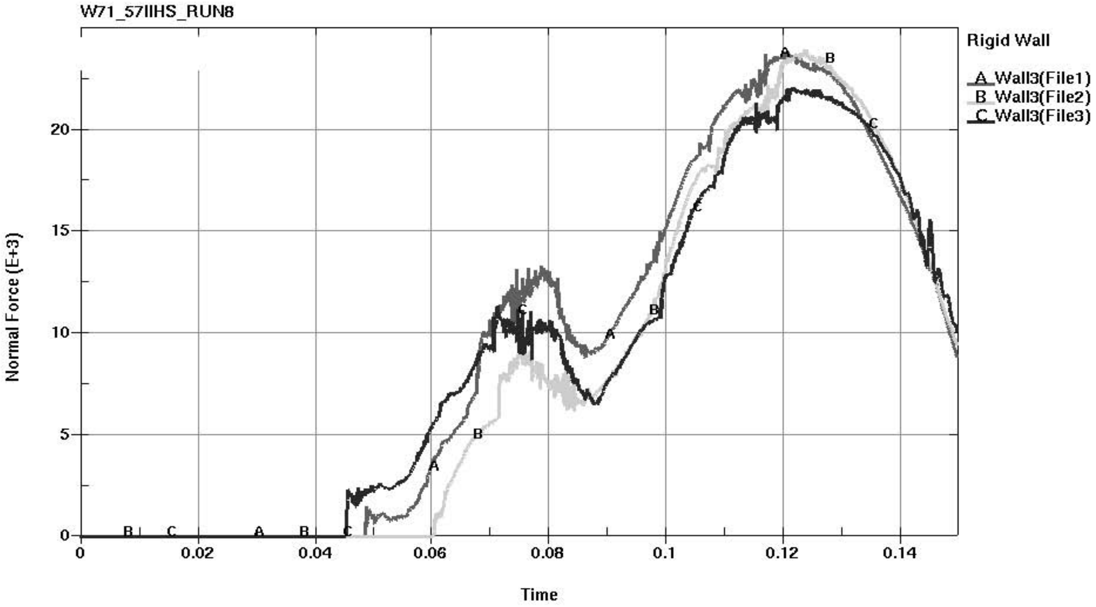
Livermore Software Technology Corporation

Recent MPP Development to Improve Consistency

Jason Wang

June 9, 2010

Numerical Variations



- CPU, MPI, model refinement, etc
- Change core counts

1. Consistency with fix number of cores
2. MPP HYBRID with multiple of core counts and performance
    1. Explicit
    2. Implicit
3. Conclusions

Consistency
Constant of cores

Problem:

Few MPI environments will use different algorithms to sum up data between within node and across nodes. This changing summation order will cause different numerical truncation error even using same number of MPP processors but changing from dual core to quad core system while

LSTC_REDUCE

keyword:

*CONTROL_MPP_IO_LSTC_REDUCE

pfile:

general { lstc_reduce }

LS-DYNA will use fix order to get consistent answer

Consistency
Constant of cores

Problem:

MPP Decomposition is based on averaging computational cost. If model has been modified or refined. The cost profile will change and model will decompose in different way. This may change numerical results.

RCBLOG

keyword:

```
    *CONTROL_MPP_DECOMPOSITION_RCBLOG
```

pfile:

```
    decomposition { rcblog file_rcblog}
```

In the first run of LS-DYNA, it will store all the cut information and also keep all other options in pfile into “file_rcblog”.

In the subsequent runs, replace p=pfile to p=file_rcblog and LS-DYNA will decompose the model base on the preserved cut lines.

Consistency

Problem:

Data on share nodes across domains may operate in different summation order. Changing of this order may cause bifurcation.

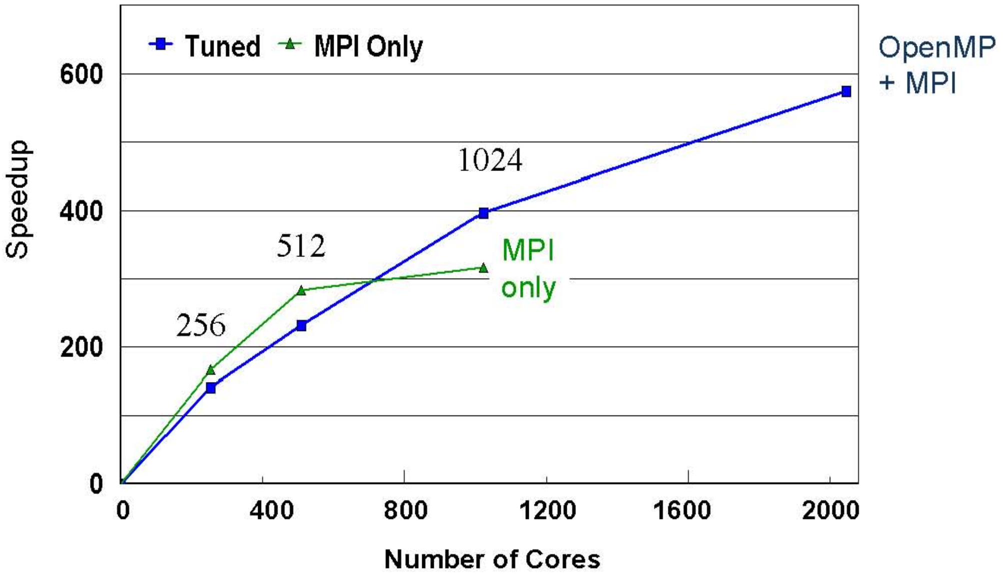
Solution:

Fix number of cores per job (?)

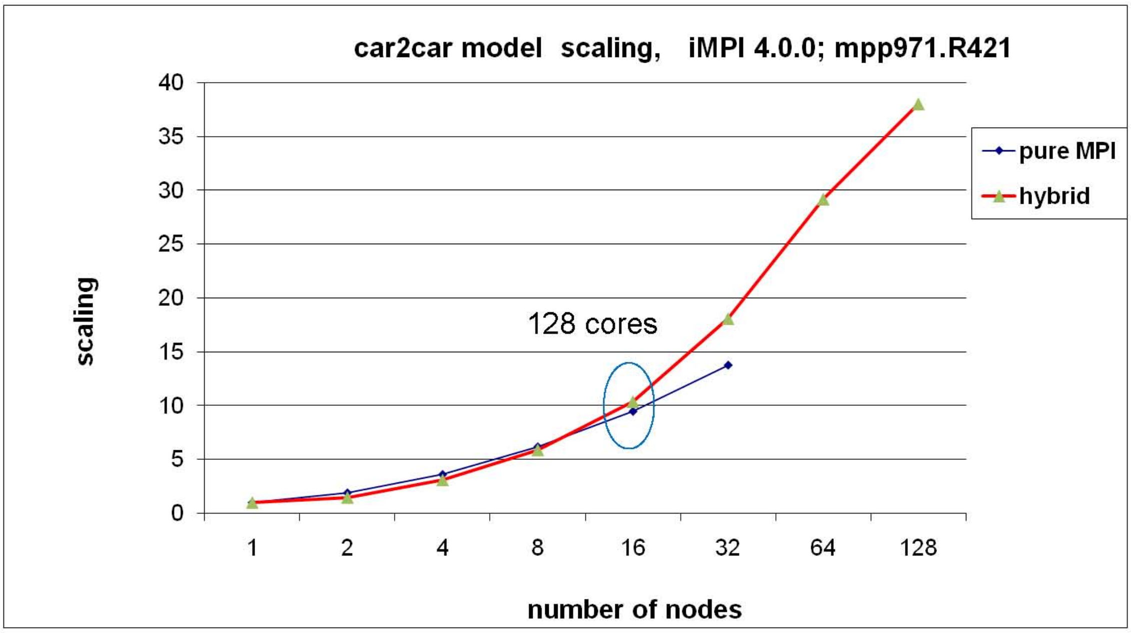
Intel: dual, quad, 6 cores, 8 cores, ….
AMD: dual, quad, 6 cores, 12 cores, ….

Core counts ↑ *, clock rates* ↓ *, model size* ↑

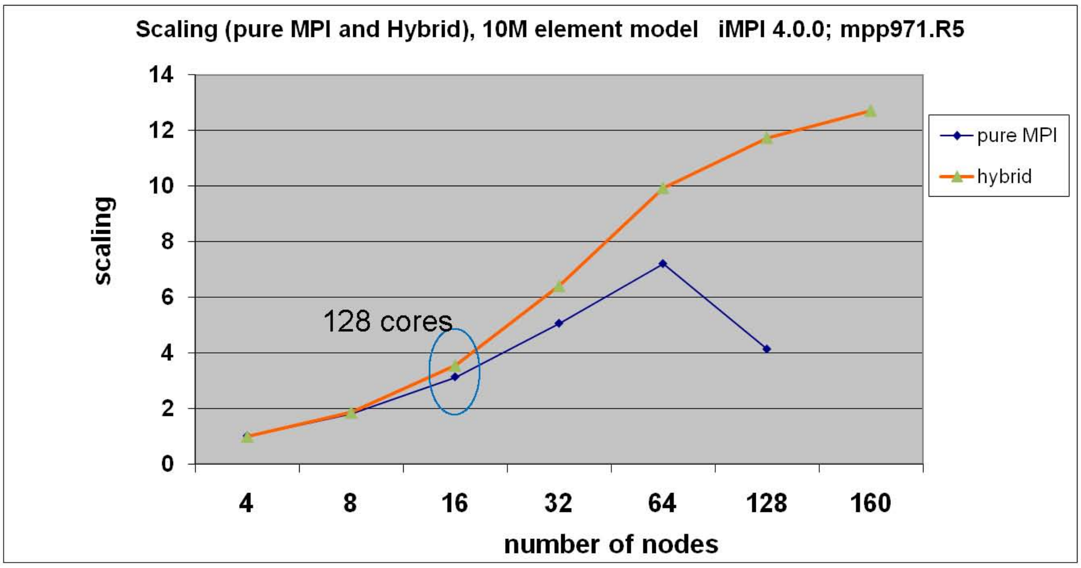
MPP Hybrid
Weather Forecast Software



How about LS-DYNA MPP Hybrid?



How about LS-DYNA MPP Hybrid?



MPP Hybrid

There is a consistent option (ncpu=-N) in LS-DYNA SMP version. Many customers used to run their jobs with the option in SMP era, even though there is about 10-15% performance penalty with the option.

LSTC added the option into LS-DYNA Hybrid version. So customers can use the option for getting consistent numerical result. However, there is a condition here. The condition is you need to fix the number of MPI processes at first.

For example, you select 12 MPI processes, then you can run your job in this way.

```
mpirun –np 12 –perhost M mpp971hy   i=input memory=xxxm memory2=xxm ncpu=-N p=pfile
```

12 cores:	12 MPI processes x 1 OMP thread	(1 nodes x 12 cores)	M=12, N=1
24 cores:	12 MPI processes x 2 OMP threads	(2 nodes x 12 cores)	M= 6, N=2
36 cores:	12 MPI processes x 3 OMP threads	(3 nodes x 12 cores)	M= 4, N=3
48 cores:	12 MPI processes x 4 OMP threads	(4 nodes x 12 cores)	M= 3, N=4
……			
72 cores:	12 MPI processes x 6 OMP threads	(6 nodes x 12 cores)	M= 2, N=6

Then you can get consistent results with 12c, 24c,36c,48c, 60c, and 72c.

MPP Hybrid - explicit

Initial benchmark results with consistent option

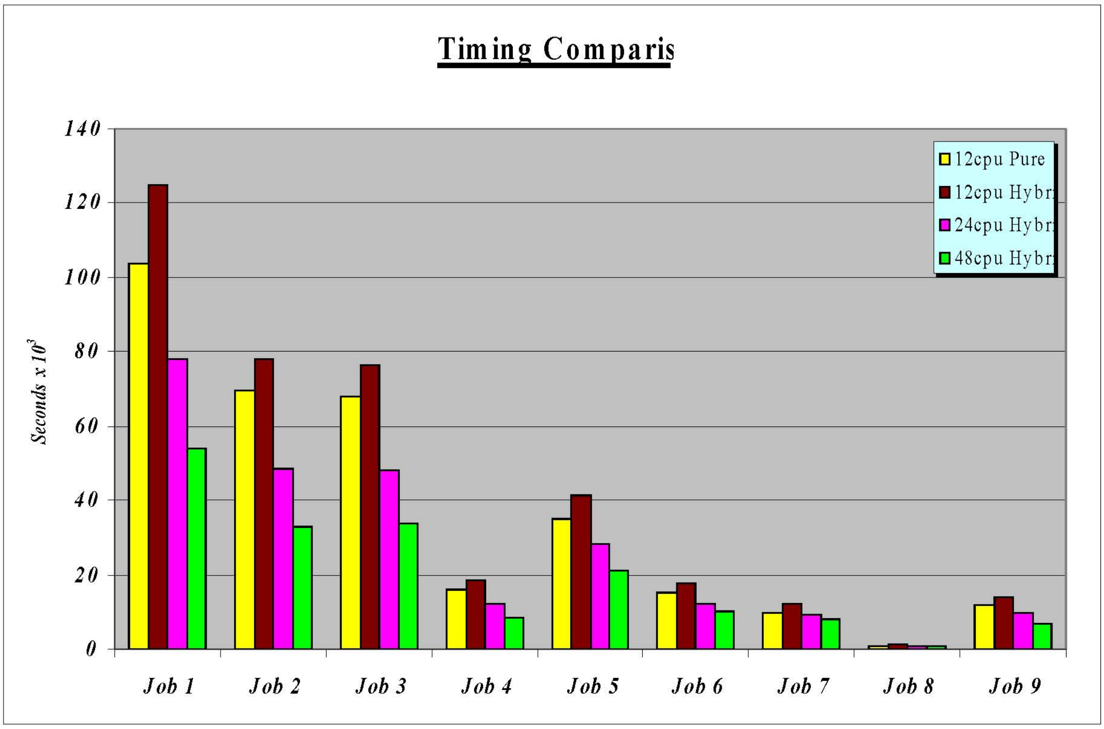
	12p pure	12x-1p	Ratio
Job1	108118s	124035s	**14.9%**
Job2	75028s	85367s	**13.7%**
Job3	68047s	87924s	**29.2%**
Job4	16610s	22677s	**36.5%**
Job5	36522s	44622s	**22.1%**
Job6	14253s	18898s	**32.5%**
Job7	9485s	12753s	**34.5%**
Job8	937s	1260s	**34.5%**
Job9	12640s	16012s	**26.7%**

Final benchmark results with consistent option (tuned)

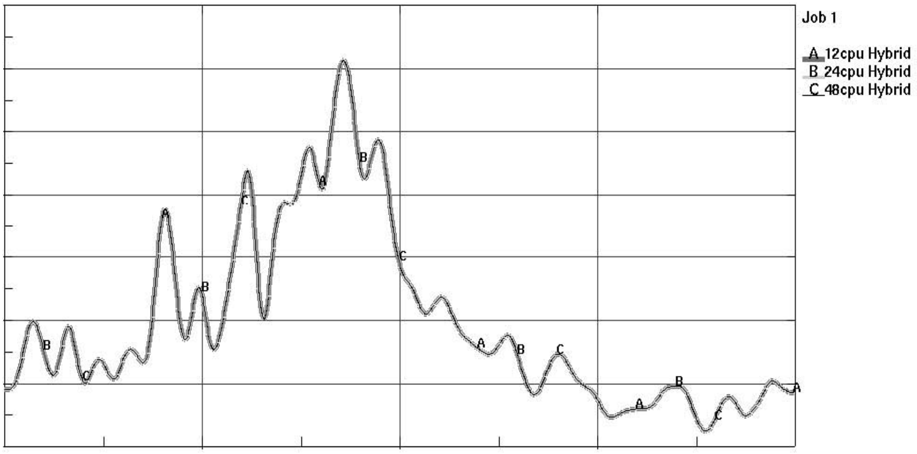
12 cores	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	avg
Hybrid	77862s	50983s	48452s	11031s	24345s	10450s	6871s	628s	7898s	
Pure MPP	69763s	47172s	43822s	11172s	22076s	9734s	6237s	596s	7002s	
Ratio	**11.6%**	**8.1%**	**10.5%**	**-1.2%**	**10.3%**	**7.4%**	**10.2%**	**5.37%**	**12.8%**	**8.4%**

MPP Hybrid - explicit

Performance comparison between pure MPP and HYBRID

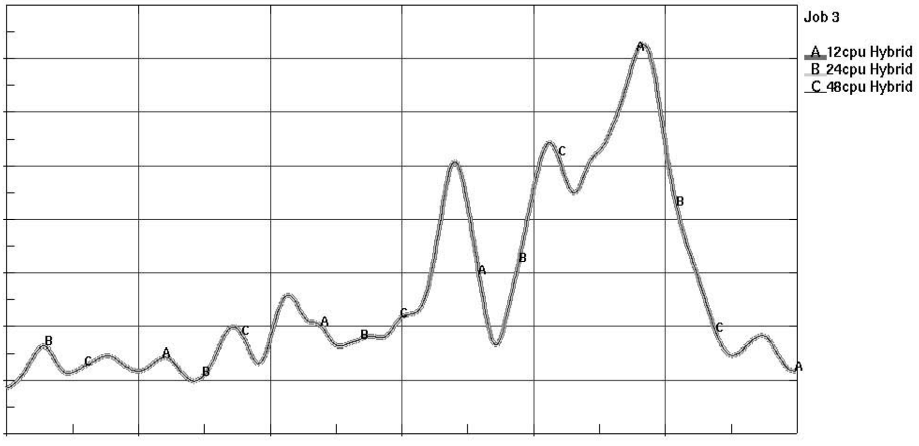


MPP Hybrid - explicit

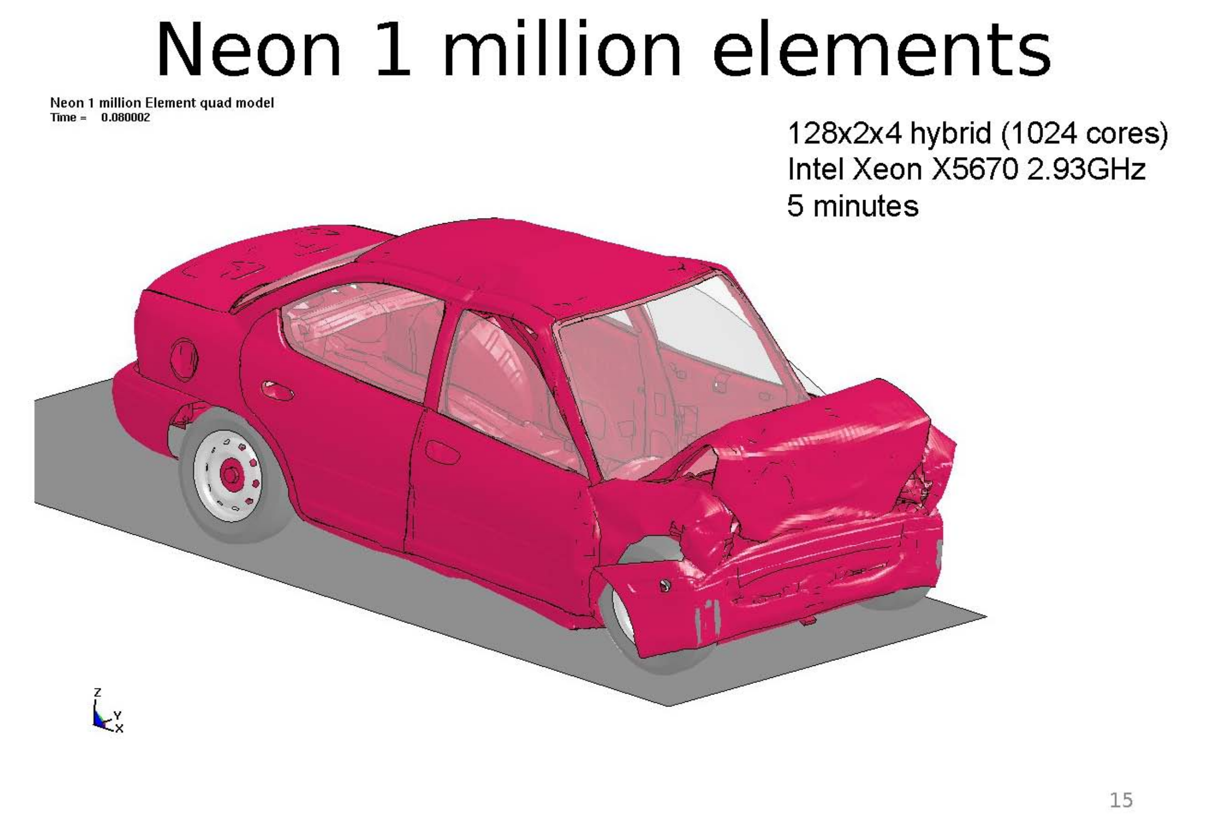


Job1 - vehicle deceleration response as measured at left rear sill

MPP Hybrid - explicit



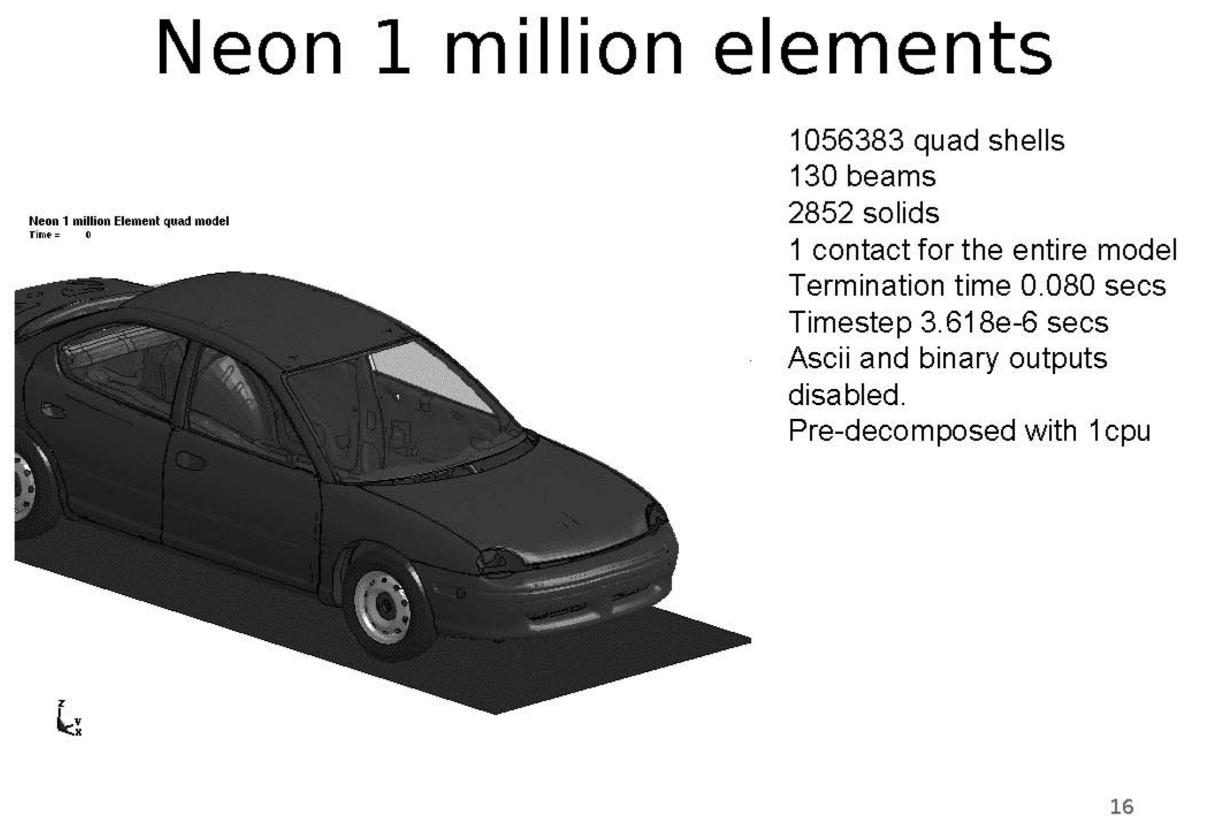
Job3 - vehicle deceleration response as measured at left rear sill



Neon 1 million elements

128x2x4 hybrid (1024 cores)
Intel Xeon X5670 2.93GHz
5 minutes

15

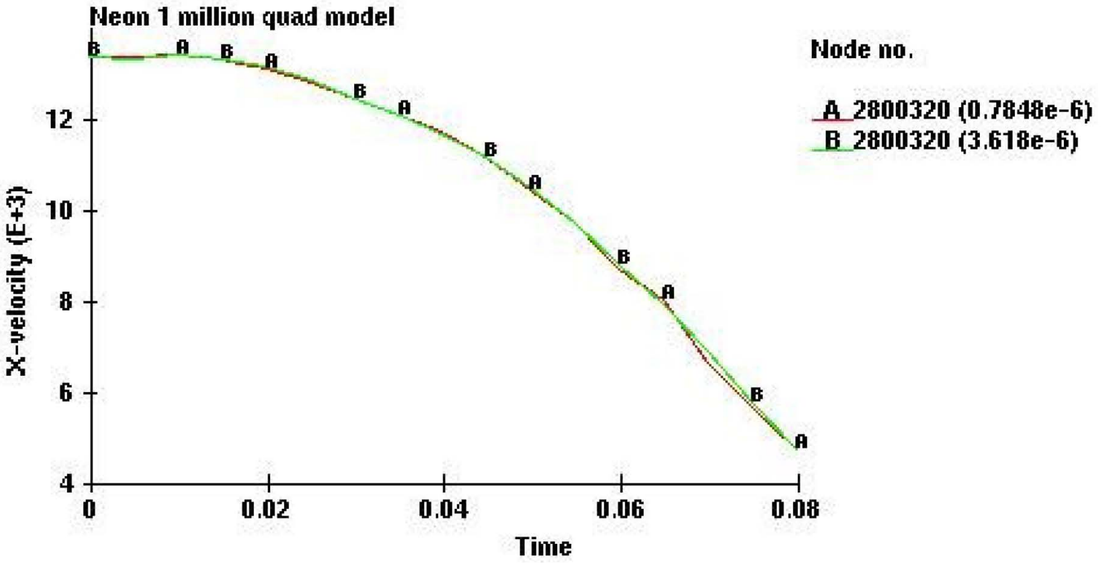


Neon 1 million elements

1056383 quad shells
130 beams
2852 solids
1 contact for the entire model
Termination time 0.080 secs
Timestep 3.618e-6 secs
Ascii and binary outputs disabled.
Pre-decomposed with 1cpu

16

Neon 1 million elements



Neon 1 million elements

128x2x4 dt=7.85e-7 8% mass increase Conventional mass scaling	6 minutes 18 seconds
128x2x4 dt=3.618e-6 894% mass increase Selective mass scaling Ongoing development to support more features for selective mass scaling	5 minutes

MPP Hybrid - implicit

Currently, the typical node configuration of installed crash simulation clusters includes:

- High-end or popular processors (4 to 8 cores)
- Low memory space (2-3GB per core, 8-24GB)
- Two Hard Disks for OS and /tmp

Due to poor file system, it’s impossible to run LS-DYNA/Implicit jobs with out-of-core solver in installed cluster, so in-core solver must be used. However, users has to run 1 MPI process per node due to limited memory space in many cases.

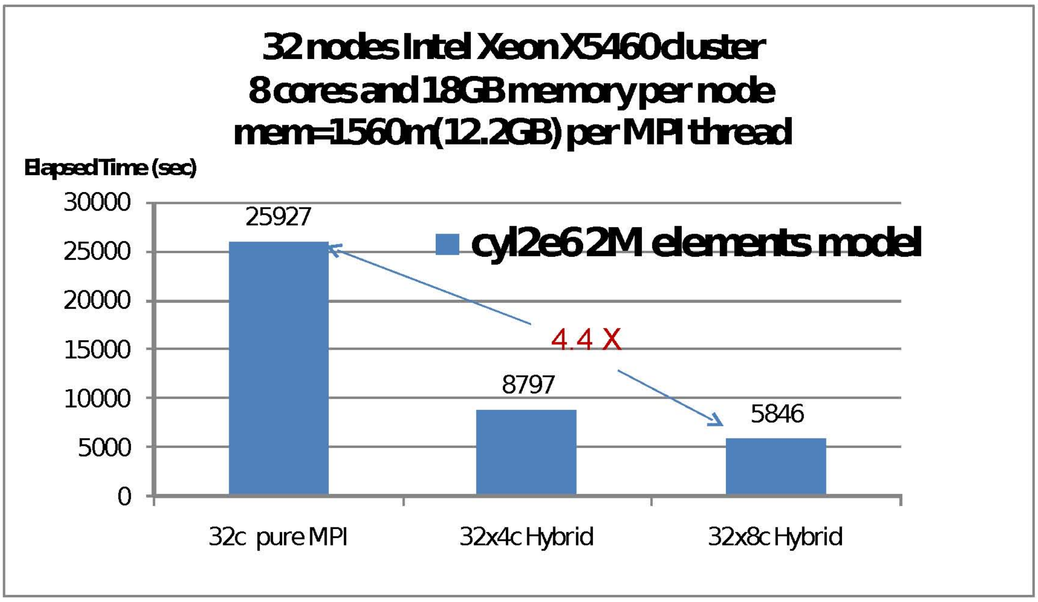
LS-DYNA Hybrid version solved the problem on current installed cluster for crash analysis.

MPP Hybrid - implicit

CYL1E6 Model **921K elements** **3.27 M DOFs**	Intel® Xeon® X7560 system **32** cores @ 2.26Ghz **128**GB memory		Intel® Xeon® X5560 cluster 8 nodes with 8 cores @2.8Ghz 24GB memory	
Cores nodes	4 cores 1 node Pure MPP Best choice	32 cores 1 node Hybrid Best choice	8 cores 8 nodes Pure MPP Best choice	64 cores 8 nodes Hybrid Best choice
Memory requirement	31.2GB per MPI process	31.2GB per MPI process	15.6GB per MPI process	15.6GB per MPI process
Elapsed Time	44013s	7047s	18521s	5541s
Speedup	1.00	6.25	1.00	3.34*

*: HT=ON

MPP Hybrid - implicit



Conclusions

- Use LSTC_REDUCE if several MPI are using
- Use RCBLOG after decided the decomposition
- Keep constant MPP proc
- Migrate to hybrid and use –ncpu to get parallel performance and get identical numerical results

Thank you