Feedback on the Utilization of MUMPS in ESI's Implicit Structural Mechanics FEM Solver

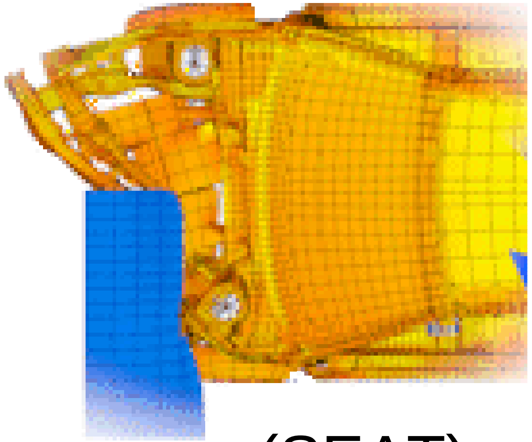
Antoine Petitet, Benoit Lodej

MUMPS Users' Group Meeting

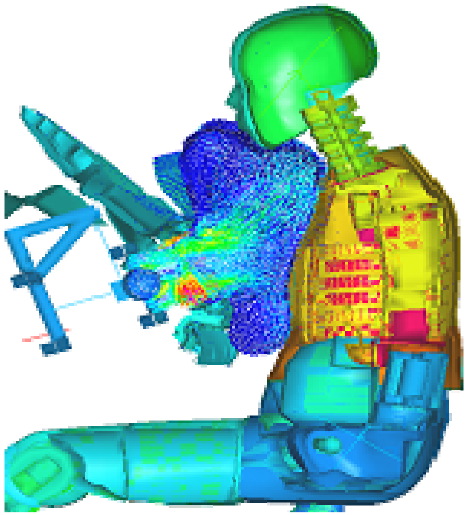
April 16, 2010

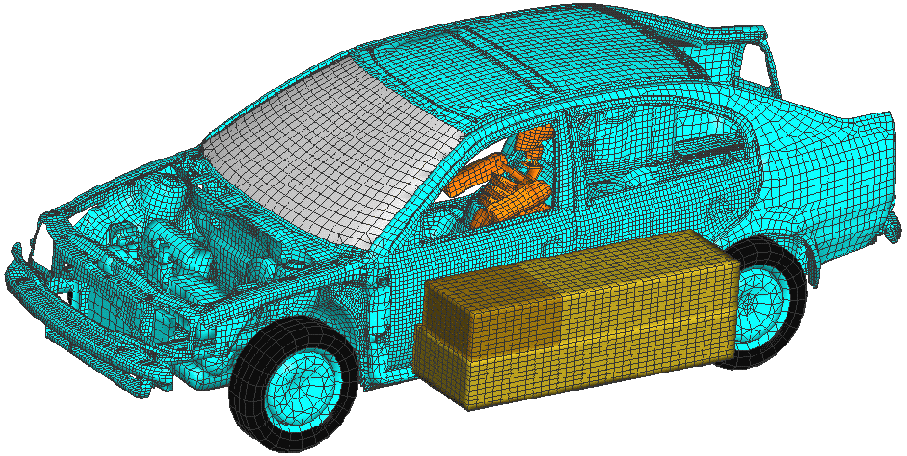
PAM-CRASH/SAFE: Main Applications

- Frontal full and offset Impact
- Side impact
- Airbag deployment



(SEAT)

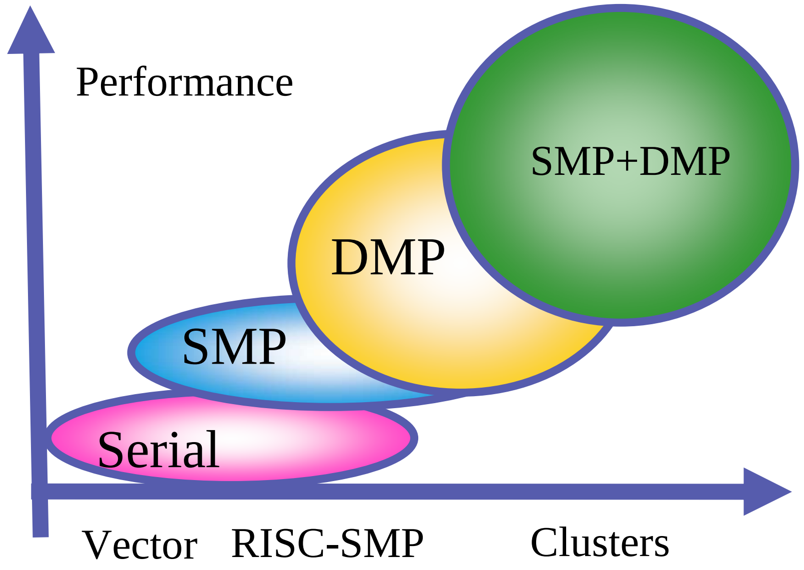




(Skoda)

Hardware/Software Evolution in CRASH Simulation

- Vectorization (Cray X-MP)
- SMP OpenMP (SGI Origin)
- DMP MPI (IBM SP, 2000)
- SMP + DMP (Multi-cores, 2009)



- Large scale multi-core clusters (mainly Linux based)
- Support various interconnects (Ethernet, Infiniband, Myrinet, ...)
- Hybrid parallelism reduce inter-node communications and take advantage of shared-memory architecture of the nodes
- SMP allows for partial load balancing which is difficult with static domain decomposition.

Towards Implicit Methods

- Historically, explicit solver where the matrices are not assembled.
- Since 2008, addition of implicit methods to the solver to tackle static linear, quasi static non linear problems and frequency analysis.
- With implicit methods, large and sparse matrices are effectively assembled and linear systems need to be solved.
- Since 2008, PAMCRASH implicit relies on MUMPS to solve these systems.
- In 2009, implicit solver for DMP released.

get it right™

MUMPS Features Current Utilization

- Single and double precision, real and complex.
- Symmetric and non-symmetric matrix operands.
- Distributed assembled COO input format for analysis and factorization.
- Default orderings (+ Metis) and scalings:
  - ParMetis under evaluation (fill-in, degenerated cases)
  - PT-Scotch evaluation to be done.
- Error analysis (condition number estimation) used internally during development.
- Transposed system solve not used.
- Schur complement not used yet ...

MUMPS Features Current Utilization (2)

- SMP and DMP versions.
- Integer*8 storage for factors.
- OOC enabled and certainly in use by our customers especially in feasability testing.
- Centralized dense right-hand sides.
- Sparse right-hand sides could be used, but not done yet.
- Distributed solution is not used.
- Memory information is always used to monitor its use.
- Matrix inertia used by eigenvalue solvers.
- Null pivot rows detection enabled with defaults.
- Iterative refinement (enabled but not necessary so far.)

MUMPS Features From ESI Point of View

+ Ease of build and use, portability (in use on a dozen of platforms by PAMCRASH), robustness.

+ Very few problems and very good support. Most of the time, the problems are in our poor reading of the MUMPS User guide.

+ Comprehensive set of features.

+ DMP enabled, scalability.

- Source code is hard to read.

- Relatively poor SMP speedup, opportunity to save memory (increasingly important on current clusters).

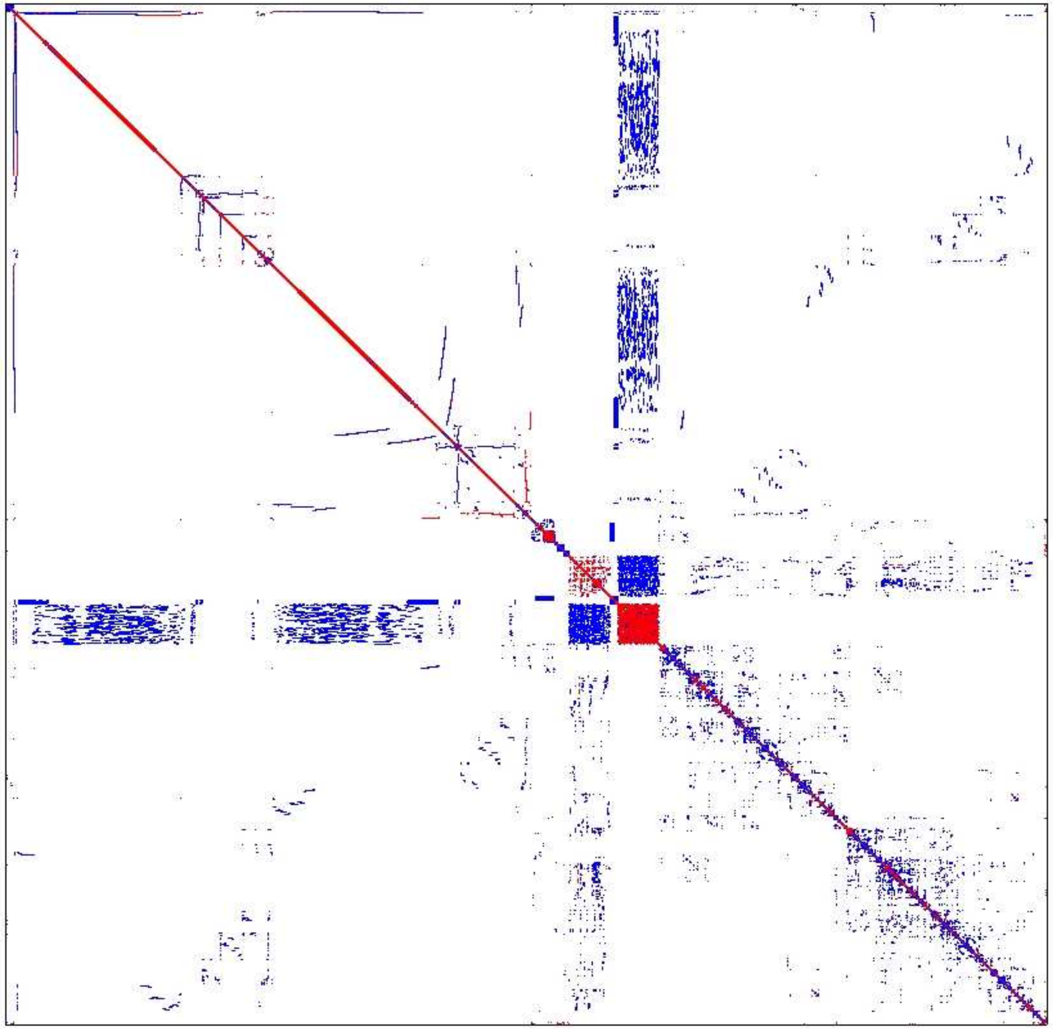
- Error analysis, inertia in DMP

Performance

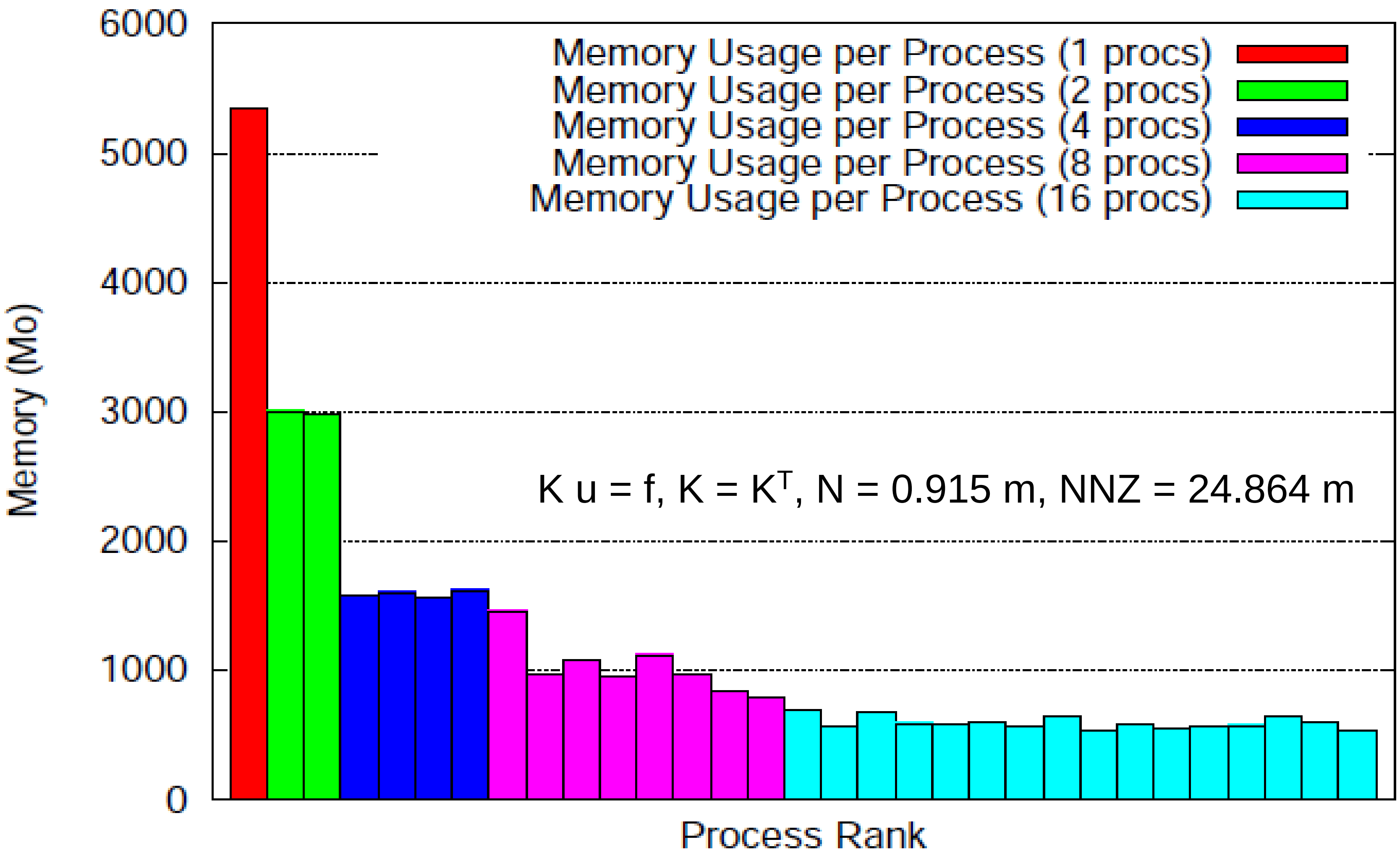
- Nehalem cluster, 4 cores per socket, 2 socket per node, Xeon E5530, 2.4 Ghz, 3 Gb per core, (Linux 5.3)
- Infiniband, HP-MPI 2.2.5.1
- Intel compiler 11.1.046, MKL 10.1
- MUMPS 4.9.2, ARPACK, ScaLAPACK, BLACS
- PAMCRASH 2010.0
- 4 representative test cases, one static linear case (hood), 2 eigenvalue problems (car1 and car2), and one quasi static non linear case with contact.
- Scalability in terms of memory footprint per core at the application level and speedups.

Hood: Matrix graph

- $K u = f$
- $K = K^T$
- N = 0.915 m
- NNZ = 24.864 m



Hood: Memory Footprint



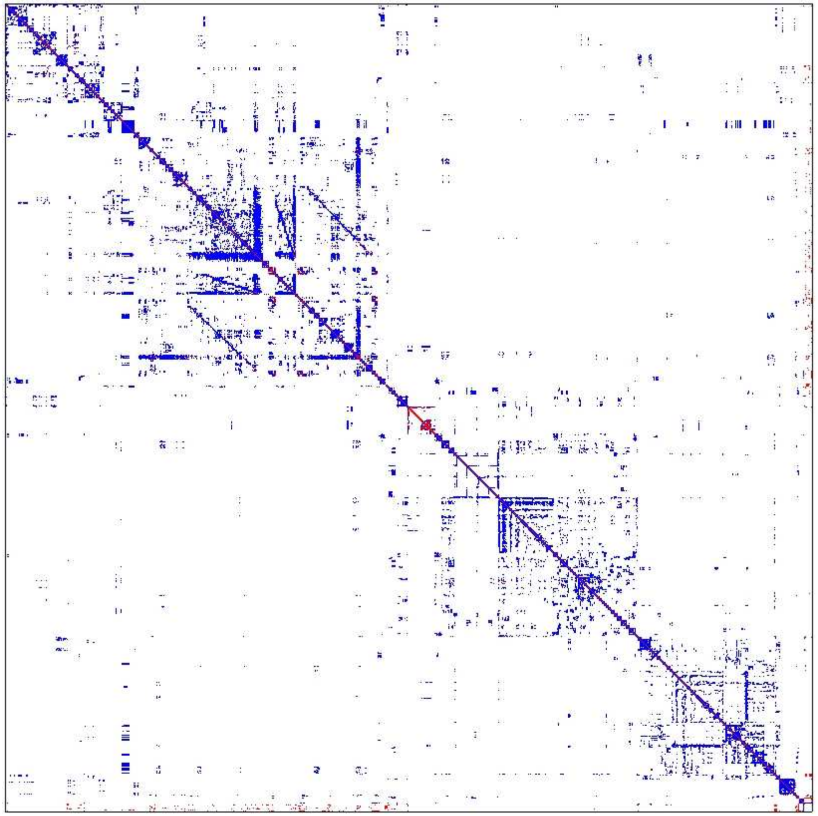
Hood: Speedup

Nb Proc	Solves	Analysis (s)	Fact. (s)	Fact. Speedup	Solve (s)	Solve Speedup
1	1	13.5	93.8	1.00	0.80	1.00
2	1	12.0	59.8	1.56	0.75	1.06
4	1	13.7	31.1	3.01	0.43	1.86
8	1	12.0	19.4	4.83	0.27	2.96
16	1	12.2	11.9	7.88	0.21	3.80

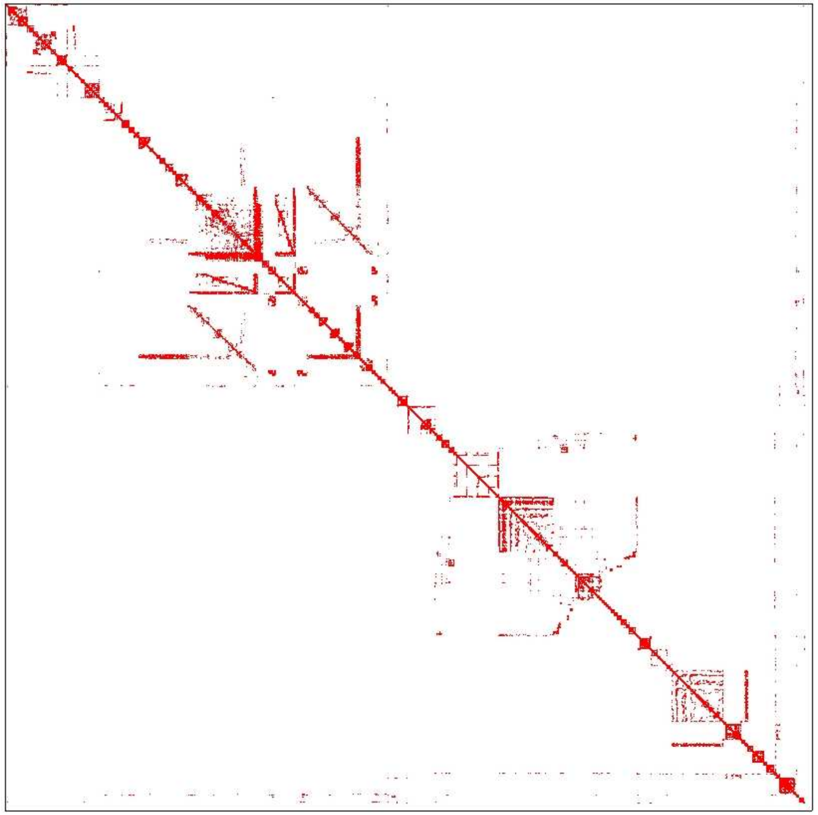
- 2 nodes, sequential Metis.
- Factorization: very good speedup up to 4 processes.
- Solves does not scale as well as factorization, but it does not matter so much here.

Car1 (BiW): Stiffness and Mass Matrices

$K - \sigma M = 0$, $N = 3.417$ m

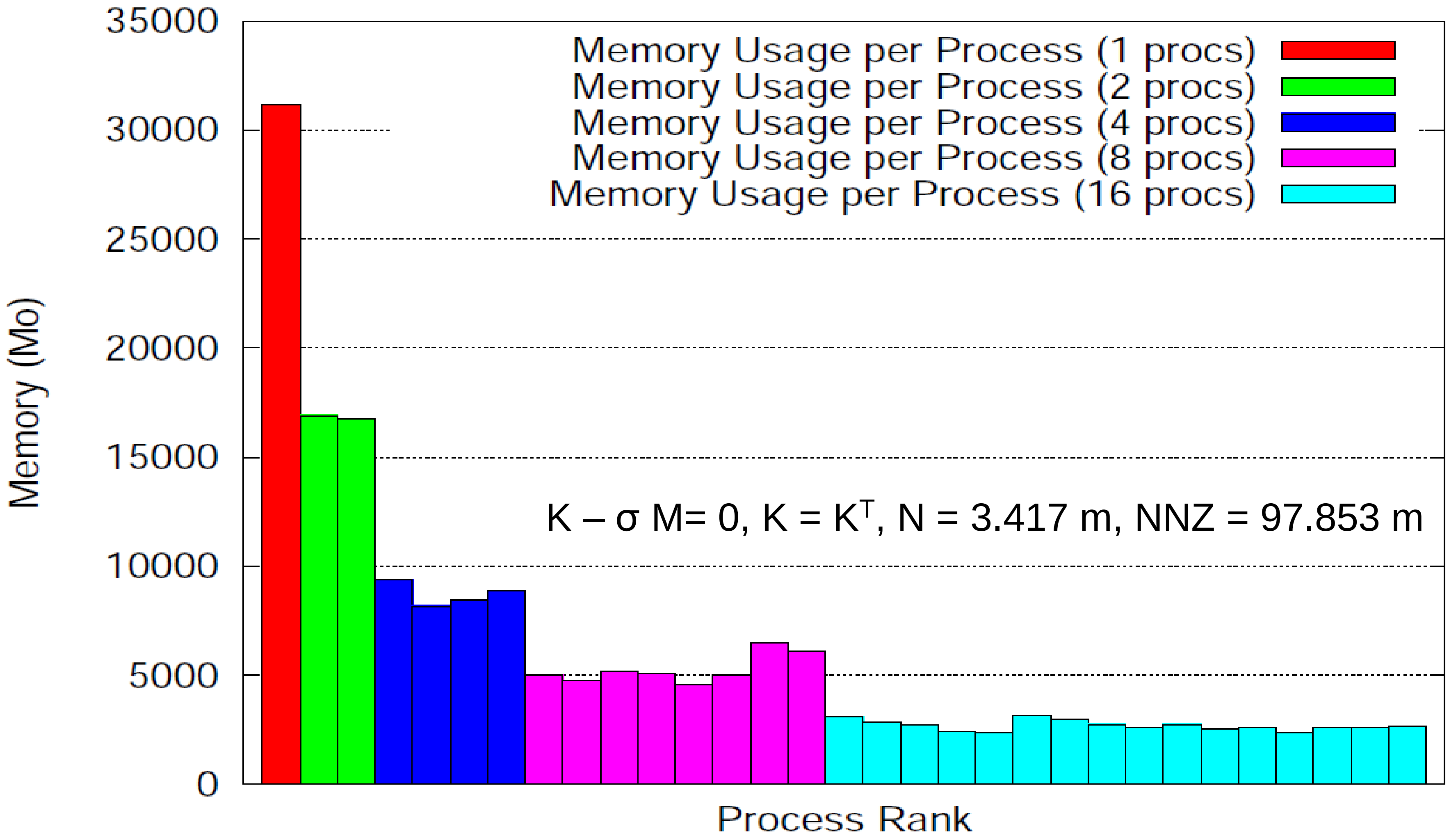


$K = K^T$, NNZ = 97.653 m



$M = M^T$, NNZ = 30.465 m

Car (BiW) 1: Memory Footprint

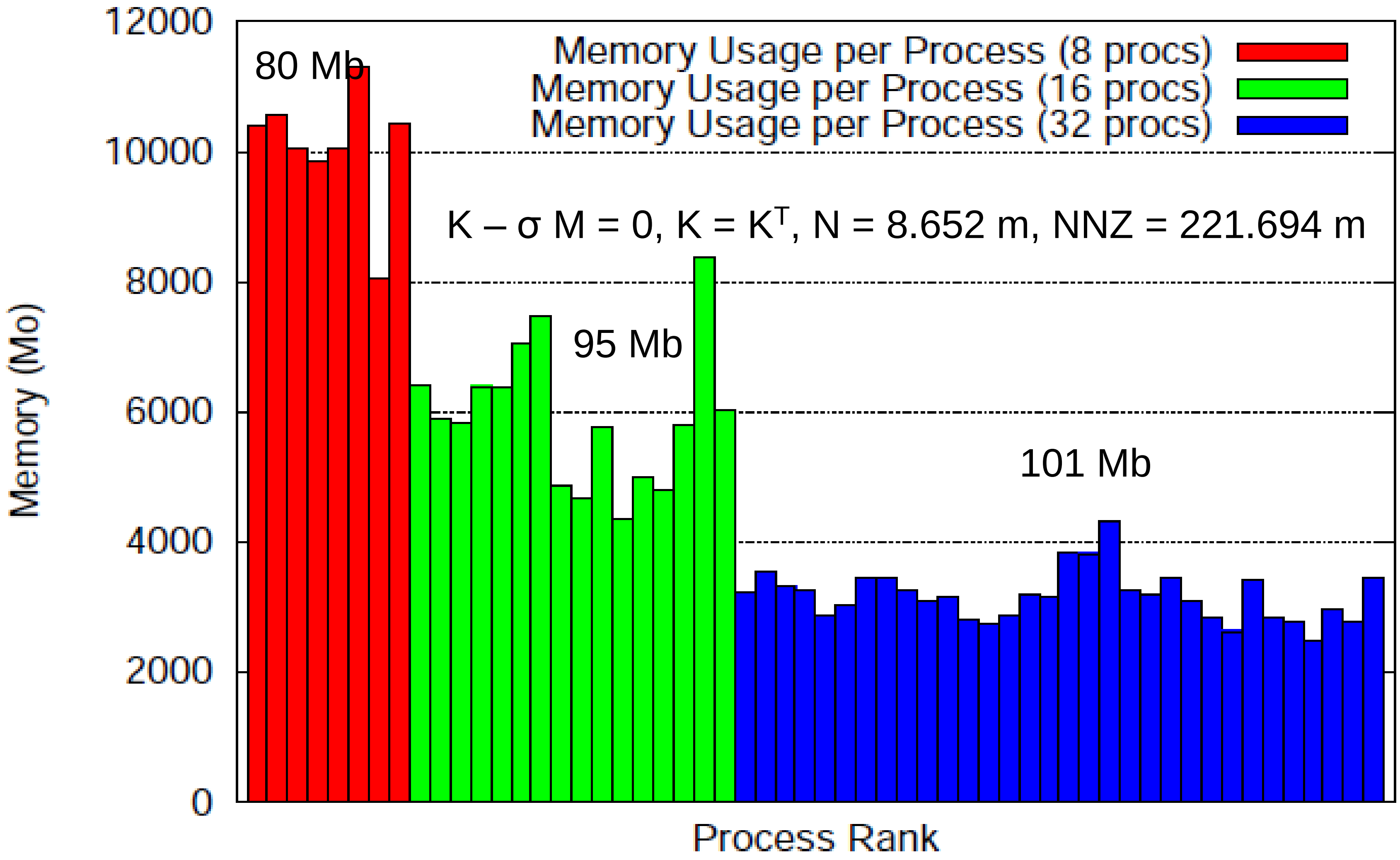


Car (BiW) 1: Speedup

Nb Proc	Solves	Analysis (s)	Fact. (s)	Fact. Speedup	Solve (s)	Solve Speedup
1	46	42.8	580.0	1.00	242.0	1.00
2	46	28.4	460.0	1.26	141.0	1.71
4	46	32.3	191.0	3.03	59.0	4.1
8	46	27.8	128.0	4.53	45.6	5.3
16	46	28.7	67.0	8.65	36.7	6.6

- 2 nodes, 10 eigenvalues, sequential Metis.
- Factorization and solves : very good speedup up to 4 processes.

Car (BiW) 2: Memory Footprint



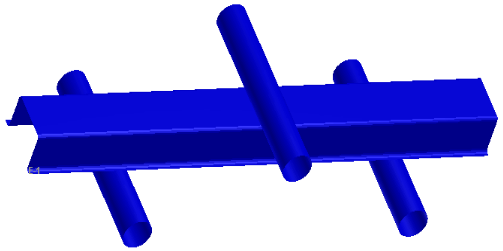
Car (BiW) 2: Speedup

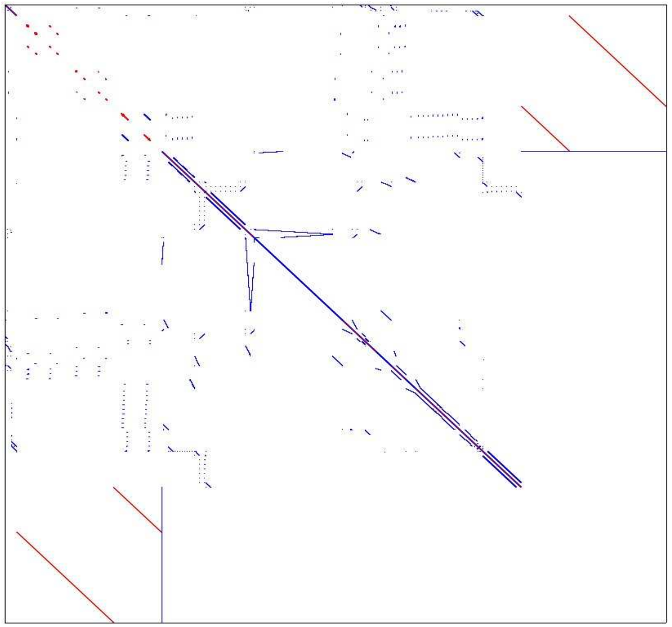
Nb Proc	Solves	Analysis (s)	Fact. (s)	Fact. Speedup	Solve (s)	Solve Speedup
8	91	94.6	315.0	1.00	206.0	1.00
16	91	95.0	194.0	1.62	186.0	1.10
32	91	97.5	108.0	2.91	133.0	1.54

- 4 nodes, 20 eigenvalues, sequential Metis.
- Factorization: good speedup from 8 to 32 processes.
- Solves does not scale as well and becomes the dominant cost of the computation.

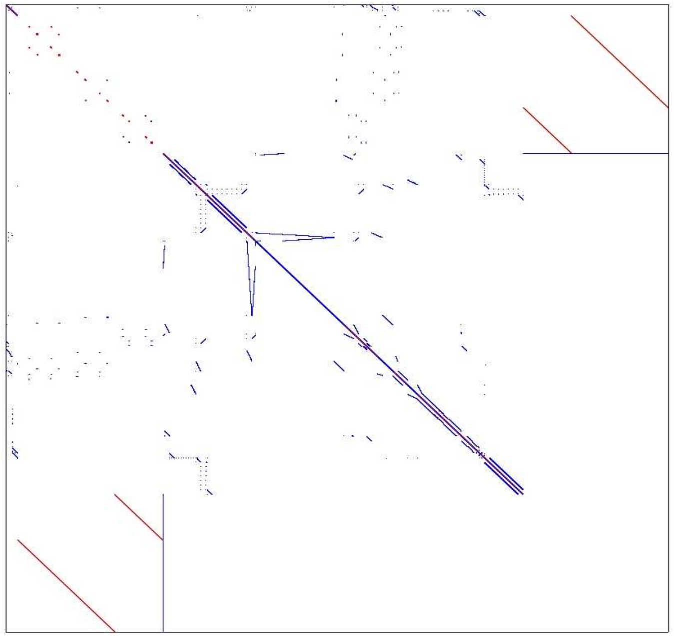
Contact case

- Newton Method, non symmetric matrix
- N = 0.215 m, NNZ = 6.826 m



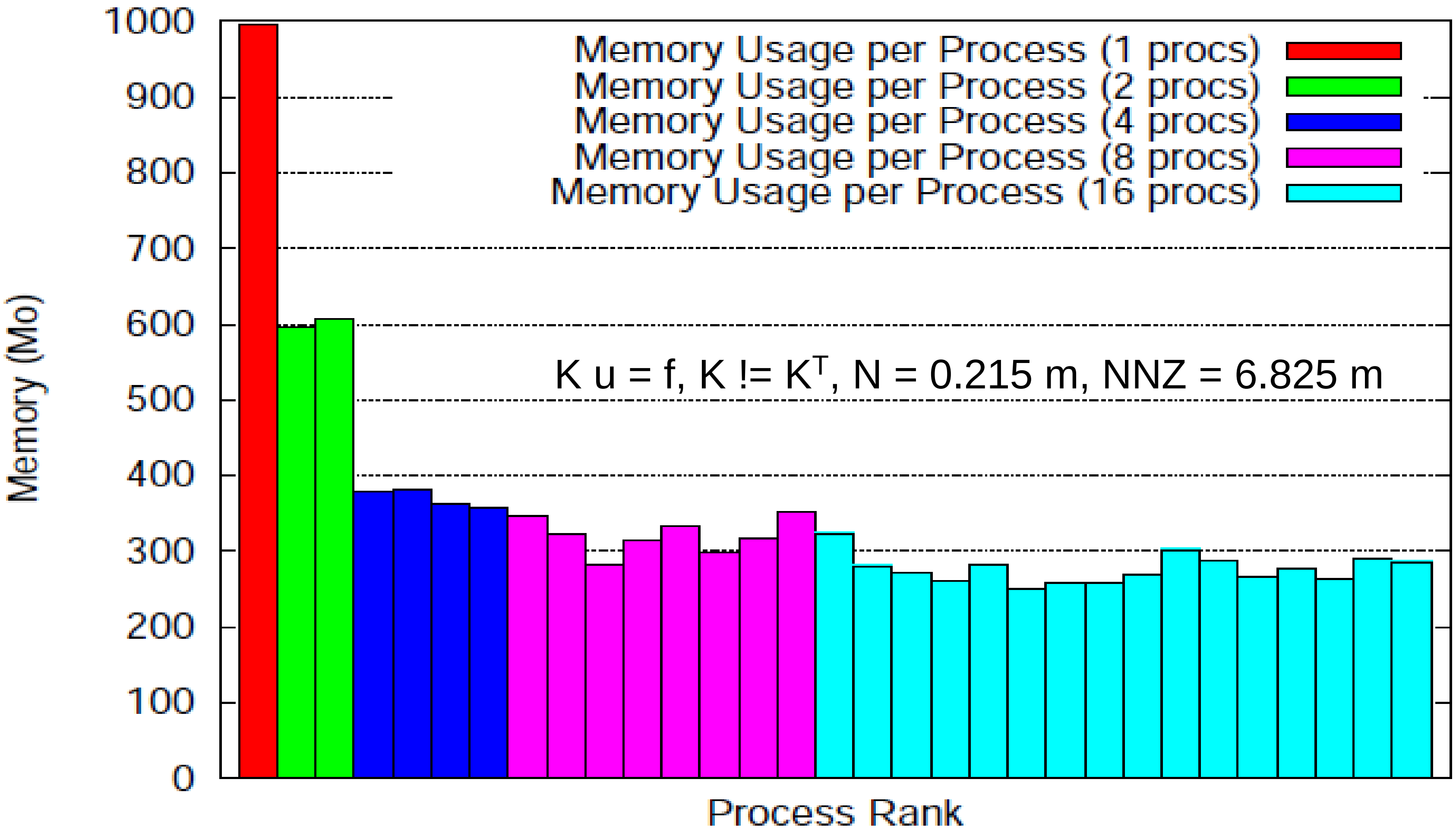


Increment 1



Increment 17

Contact case: Memory Footprint



Contact Case: Performance

Nb Proc	Analysis	Fact.	Analysis (s)	Fact. (s)	Solve (s)
1	259	575	611.6	2349.4	85.6
2	248	562	1150.2	1739.4	73.1
4	266	574	1272.1	1315.5	74.3
8	247	550	1092.6	1501.0	52.8
16	269	566	1234.6	2620.1	68.2

- Results are difficult to compare as the matrices vary with the number of processes and the number of iterations.

Conclusions

- Factorization scales reasonably well. Memory footprint always scales very well.
- In eigenvalue solvers, solves may rapidly become the dominant cost of the computation.
- It is sometime rather difficult to figure out what causes performance degradations.
- Portability, robustness, performance and support are the major MUMPS features that have convinced us to use it in one of our major product.
- MUMPS is currently a fundamental component of the ESI implicit structural mechanics FEM solver.

get it right™

www.esi-group.com