Distributed Memory Parallel MSC.NASTRAN on an IBM Workstation Cluster at Ford Cologne

Authors:

Ford Werke Aktiengesellschaft, Spessartstrasse, 50725 Cologne, Germany:
Ulrich Viersbach, Tel. +49-221-9034825, email: uviersba@ford.com
Dr. Matthias Weinert, Tel. +49-221-9035840, email: mweinert@ford.com
Christian Wilmers, Tel. +49-221-9037015, email: cwilmers@ford.com
MacNeal-Schwendler GmbH, Innsbrucker Ring 15, 81612 Munich, Germany
Dr. Stefan Mayer, Tel. +49- 89 43198761, email: stefan.mayer@mscsoftware.com

Abstract:

MSC.NASTRAN is the main structural FE code and the second most used CAE code on compute servers at Ford worldwide. To reduce costs, alternative ways of computing are being investigated such as using idle cycles on the large number of available Ford workstations. While smaller analyses run efficiently on single workstations, more complex calculations require larger computer resources as potentially available with several workstations clustered together using a distributing memory parallel code.

The presented paper outlines results from evaluating and benchmarking a V70.7 development version of the distributed memory parallel MSC.NASTRAN code on an IBM RS/6000 model 590 workstation cluster at Ford Cologne. It covers linear static (solution 101), normal modes (solution 103) and direct frequency response (solution 108) analyses which were carried out for the first time ever on a workstation cluster. Cluster turnaround times are compared with those on IBM compute servers and Cray C90.

The evaluation showed that the development version has great potential for typical jobs used for the analysis of Ford body structures. Good speed-up with an increasing number of processors is achieved on the workstation cluster. The cluster with 8 workstations showed better turnaround times than the Cray C90 for SOL101 benchmark cases. When using latest technology workstations the cluster is expected to show even better turnaround times with also superior performance to the C90 for SOL103 and SOL108. The turnaround target of overnight completion could be achieved with 8 cluster workstations for all benchmark cases. The IBM SP compute server at MSC.Software's office in Los Angeles which is equipped with similar processors as the cluster workstations showed similar performance as the cluster. The IBM SP compute server at IBM's benchmark center at Poughkeepsie/USA using 8 latest technology processors and a superior I/O subsystem was performing better than the workstation cluster with 8 processors and the Cray for all benchmark cases.

As next steps Ford is planning to carry out runs with the distributed memory parallel version 70.7 on a Ford compute server once this version is officially released and consider production implementation depending on results.
Ford is also planning to evaluate the code with our largest CAE models on the above workstation cluster and in a production environment. Ford will encourage MSC.Software to implement several improvements to MSC.NASTRAN such as the integration of the code with a workload management package to enhance cluster robustness.

Keywords: automotive industry; computer aided engineering; parallel workstation cluster; MSC.NASTRAN, IBM workstations.

1. Introduction

The automotive industry is under pressure to become more efficient and to produce cars with ever increasing quality while reducing the cost per vehicle.

A significant part of Ford's information technology budget is spent on CAE analyses using compute servers. Ford is therefore looking at alternative and more (cost) effective ways of carrying out numerically intensive calculations. Workstations for CAD/CAM are becoming much more powerful due to requirements for e.g. solid modelling and are approaching the performance of CAE workstations. These workstations are idle during absence of the user from his/her desk, during the night, over the weekends and during vacation time. The combined capacity of all workstations at Ford in Cologne-Merkenich over night is potentially higher than the combined capacity of Ford's compute servers world-wide. If these idle cycles could be used efficiently for CAE analyses formerly done on compute servers large cost savings could be realized.

Nastran is the main structural FE code and the second most used CAE code on compute servers at Ford worldwide as shown in Figure 1**.**

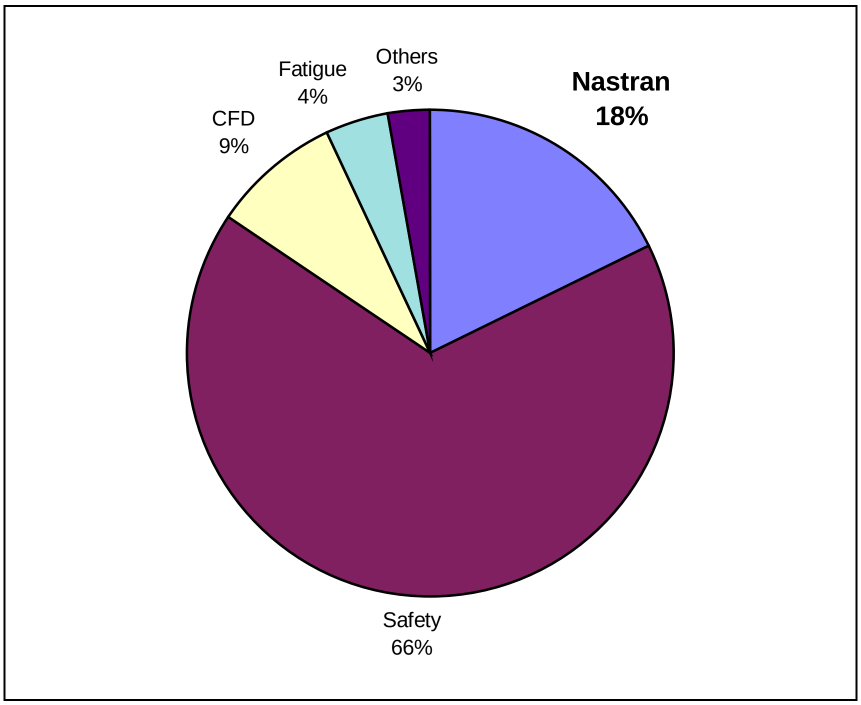


Figure 1: Worldwide CAE Compute Server Usage Share for European Users in 1st half 1999

The traditional shared memory version of MSC.NASTRAN scales only very moderately with multiple CPUs thus requiring fast single CPU performance as available in conventional expensive supercomputers. MSC.Software's future strategy for parallelization is based on the development of a distributed memory parallel version of their code. This version can be used on both compute servers and workstation clusters. If the version ran successfully on less expensive compute servers or even on workstation clusters for the MSC.NASTRAN solutions used at Ford high cost savings could be achieved.

Figure 2 shows the MSC.NASTRAN usage split by solution in the first half of 1999.

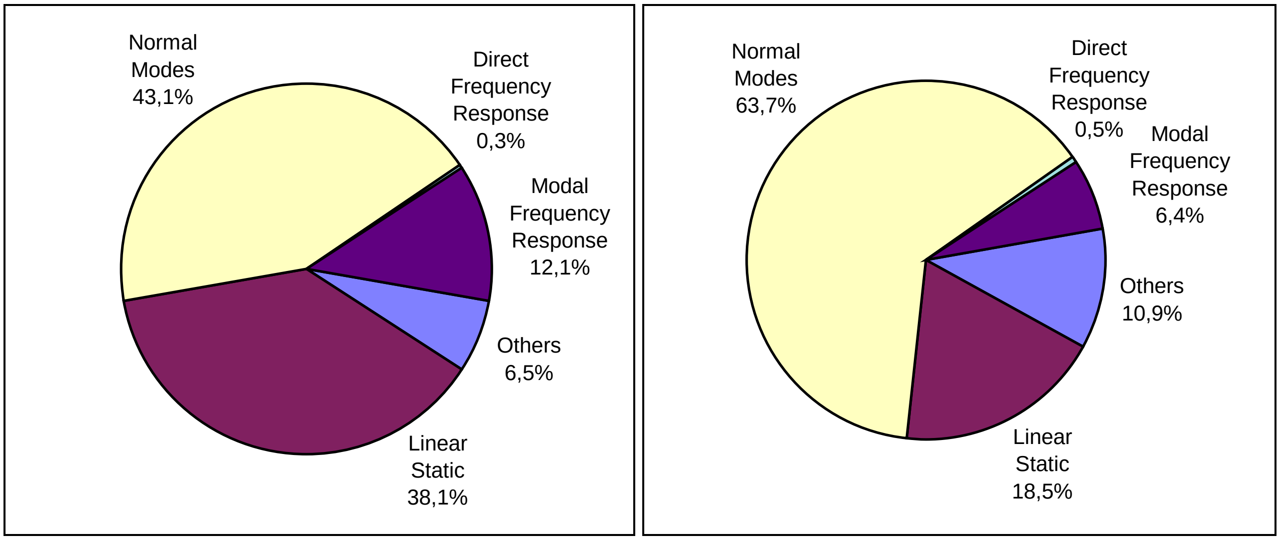


Figure 2a: split by number of jobs **Figure 2b: split by CPU seconds**

Figure 2: MSC.NASTRAN Usage Split by Solution on European Compute Servers in 1st Half 1999

The main solutions applied at Ford are normal modes (solution 103), static analysis (solution 101), modal frequency response (solution 111), non linear static (solution 106) and design optimisation (solution 200). Normal modes analysis accounting for 64% of all CPU seconds used is the most significant area.

2. Problem Definition/Objectives

Objective of the effort was to evaluate the distributed memory parallel version of MSC.NASTRAN (MPI version) on a workstation cluster at Ford Cologne. As the code was most advanced on MSCs development platform IBM, and IBM workstations were readily available at Ford, these were selected for the evaluation. Only a development version of the cluster code was available at the time of this effort. The evaluation included both, the robustness of the code, the available functionality and benchmark runs to identify turnaround times compared to compute servers.

The project covered for solution 101, 103, and 108:

- runs with the distributed memory parallel (MPI) version on an IBM RS/6000 workstation cluster (model 590) at Ford Cologne with 1, 2, 4 and 8 processors
- runs with the MPI version on an IBM RS/6000 SP compute server based on latest technology POWER3 nodes at IBM’s benchmark center Poughkeepsie in the USA with 1, 2, 4 and 8 processors (referred to in the text as ‘IBM SP Pouk’)
- runs with the current production version 70.5.2 on the Cray C90 at Ford Cologne (shared memory version) using one processor
- runs with the MPI version on an IBM RS/6000 SP compute server at MSCs office in Los Angeles with 1, 2, 4 and 8 processors; equipped with processors equivalent to workstation model 390H processors which are similar to the processors used in the cluster workstations (runs for solution 101 only; referred to in the text as ‘IBM SP LA’).

Ford’s focus was on shell type structures.

3. Analysis

3.1 Code Background

MSC.Software offered the first parallel production system of MSC.NASTRAN in 1987, based on the shared memory paradigm. At that time and during the following years the parallelization efforts concentrated on parallelizing the computations in several computationally intensive modules, for example the matrix decomposition. However, due to significant improvements in the numerical methods of MSC.NASTRAN, the fraction of time spent in these formerly dominating modules has been reduced, so that now a more extensive parallelization is necessary to achieve good parallel performance. Moreover, it has turned out that not only computations, but also the I/O traffic must be parallelized in order to obtain highly efficient parallel analysis solutions.
For these reasons, MSC.Software started to work on new parallelization approaches earlier this decade, this time based on the distributed memory paradigm to be able to address parallel I/O issues as well. First successes were obtained in the European Community funded EUROPORT project [1], which resulted in the distributed parallel production version MSC.NASTRAN V69.2 available only on the IBM SP compute server in 1996. Encouraged by the results of this project, the efforts on distributed parallel MSC.NASTRAN have been intensified during the past two years, partially funded by DARPA in the USA. This resulted in versions 70.5.3 and 70.5.4 available on the IBM AIX Parallel Environment in 1997 and 1998, respectively.
MSC.Software is now ready to deliver MSC.NASTRAN V70.7, which will contain further distributed parallel analysis capabilities. V70.7 will offer distributed parallel solutions for linear static (SOL101), normal modes (SOL103), direct frequency response (SOL108) and modal frequency response (SOL111) analysis [2]. MSC.NASTRAN V70.7 is intended for parallel compute servers, for example IBM RS/6000 SP, SGI Origin 2000, HP N- and V-class, SUN HPC 6000, NEC SX-4/5, etc.. The evaluation project on the IBM workstation cluster at Ford Cologne, which is the topic of this paper, was the first time ever that a V70.7 (development) version was tested on a workstation cluster.
Encouraged by the success of this project, a production version of distributed parallel MSC.NASTRAN for workstation clusters is planned for V71, which is scheduled for next year. In this version, for example, parallel speed up for SOL101 is planned to be improved beyond the results reported here by tuning the automatic model domain decomposition in the SEQP module as well as the PRESOL module. Moreover it is planned to extend the parallelization to further solution sequences and to add support for workload management software which is essential for the production usage on workstation clusters.

3.2 . Infrastructure

As the IBM version of the cluster code was the most advanced at the time of the evaluation, a cluster of 8 IBM workstations model 590 was established, which were phased out at Ford Cologne-Merkenich. Each workstation had 512 MB of main memory (see table 1) and 8 GB of disk with the exception of the master workstation which had 17.5 GB of disk to provide sufficient disk space for serial runs. Previous experience with running other parallel applications on workstation clusters had shown that an efficient environment required each workstation to be connected within the network with at least switched 10baseT Ethernet (10Mbit/second dedicated to each workstation) or better 100basedT switched [3, 4, 5, 6]. For the benchmarks a 100baseT switched network was established.

Machine Type	Processor		Memory Size	Disk Type	Communication between Processors	
	Type	Clock Speed	(per pro-cessor)		Type	Sustained MPI Bandwidth
IBM RS/6000 model 590 workstation cluster	POWER2	66 MHz	512 MB	local wide SCSI disks	100BaseT switched Ethernet	7 MB/s
IBM RS/6000 SP (Thin2 nodes)	POWER2	66 MHz	512 MB	local wide SCSI disks	HPS switch	34 MB/s
IBM RS/6000 SP (2-way SMP)	POWER3	200 MHz	4096 MB	local SSA disks	SP switch	125 MB/s
Cray C90	Cray C90 Vector Processor	240 MHz	4096 MB	DA302 striped disk array	Shared memory	10 GB/s

Table 1: Benchmark Machine Details

Figure 3 shows the benchmark infrastructure for the workstation cluster which was dedicated for this effort.

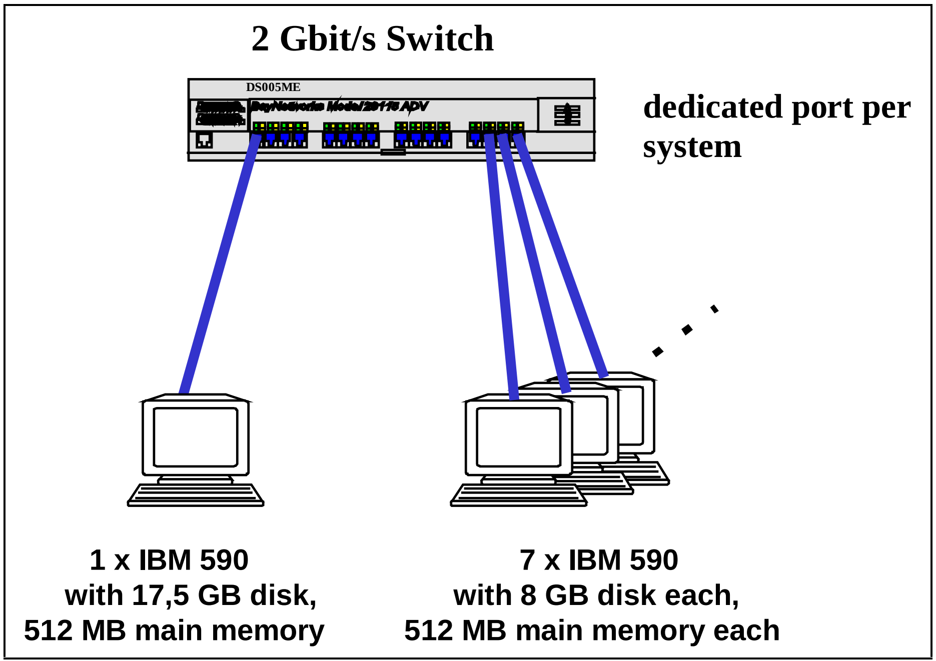


Figure 3: Workstation Cluster Infrastructure with 100BaseTX (Full Duplex) Switched Ethernet

The POWER3 nodes are 2-way SMP nodes. For this benchmark just one CPU per node has been used (one MPI task per node). In terms of the widely used SPECfp95 metric a POWER3 CPU is about 3 times faster than the POWER2 CPU. The integer performance is about 4 times higher.

The I/O performance has a significant impact on the overall MSC.NASTRAN performance. The I/O rate of the evaluated POWER2 machines is limited to few MB/s due to rather old and slow disks. The more recent striped disks attached to the POWER3 system should be able to drive the SSA disk adapter close to its limits (35 MB/s for a read or write operation). I/O performance could be further increased by using the latest SSA adapter technology (90 MB/s peak) and by using more than one SSA adapter per node.

Since the virtual memory management VMM manages virtual memory on behalf of the I/O subsystem, the VMM plays a major part in the operation of an AIX file system. No tuning of the VMM has been applied for the benchmark. For I/O intensive applications like MSC.NASTRAN typically significant I/O performance enhancements are possible by tuning the file memory usage through the AIX vmtune interface [7].

The MPI bandwidth given in table 1 refers to a simple ping-pong communication test for large messages. On the Cray C90 the sustained memory bandwidth e.g. for memory intensive operations is about 10 GB/s per CPU whereas a much lower MPI bandwidth is expected.

3.3 Benchmarking

The following section describes parallel benchmark studies with a V70.7 MSC.NASTRAN development version for solution 101, solution 103 and solution 108.

3.3.1 Linear Static (solution 101)

Introduction

This section covers solution type SOL101 applied to real world Ford problem cases from the body durability area. As shown in Figure 2, SOL101 is the Nastran solution with the second largest compute server usage at Ford. It is the main solution type for the linear durability assessment of body structures at Ford.

It is mostly used in two applications:

Standard loadcase analysis

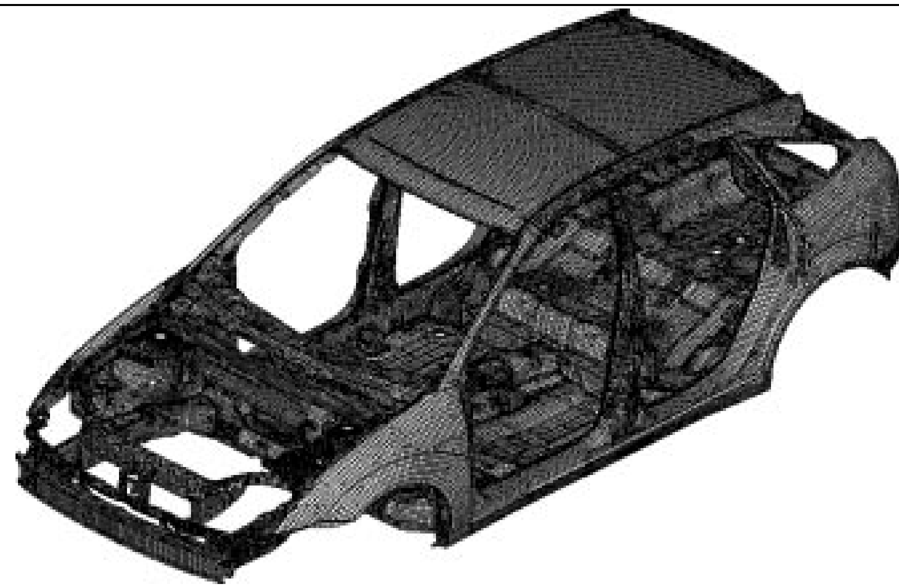
Standard Loadcases are isolated loading conditions which occur during Ford's vehicle durability test (PASCAR). Each Standard Loadcase requires a SOL101 run with one matrix decomposition and a subsequent forward-backward solution. In the past, Ford used half body models for those analyses since most loadcases could be separated into a symmetric and an anti-symmetric load set. Nowadays, those simplifications are avoided and full body models including non-symmetric design features are analysed.

Unit loadcase analysis for fatigue analysis

This analysis is the first step in a fatigue life analysis process. Stresses are calculated subject to unit loads at all body attachment points. These stresses are scaled by the real loads that act at the attachment points and will be superposed**.** The resulting linear stress time histories for all elements are then fed into a damage model that returns a damage or life estimation. Besides the fact that an Inertia Relief analysis is carried out since the body structure is not constrained, the main difference from the numerical point of view lies in the number of forward-backward solutions. The unit loadcase analysis requires still one decomposition of the stiffness matrix but *n* forward-backward solutions where *n* denotes the number of load sets.

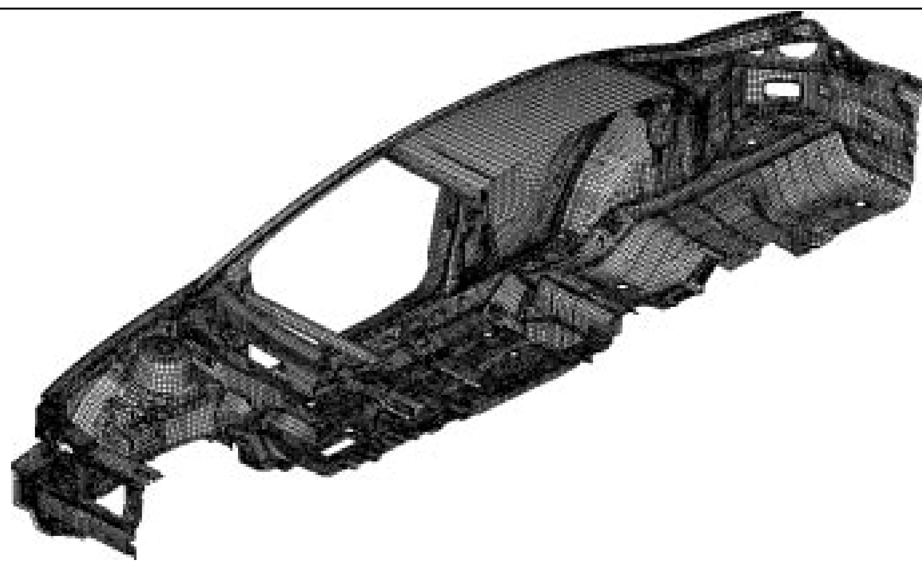
Model Description

All models investigated in this study are recently used real world applications from the body structure area. They are dominated by shell-type elements but also include other element types in a smaller number. The first model (Figure 4) used in the benchmarks is a full body structure model of the Focus without closures. It is a common baseline model shared between the CAE disciplines NVH and Durability. The second model (Figure5) is a half body structure model of the Cougar.



GRIDS =	225,885	CQUAD =	205,848
CTRIA =	12,955	CBEAM =	252
CBAR =	364	CROD =	516

Figure 4: Model 1 - Focus Full Body Model



GRIDS =	102,325	CQUAD =	89,684
CTRIA =	6,717	CHEXA =	80

Figure 5: Model 2 - Cougar Half Body Model

Half body models as shown in Figure 5 were used in the past to reduce turnaround times for CAE assessments neglecting all non-symmetric details. Although the trend is pointing away from those models it is included in our considerations to assess the influence of the model size on speed-up and efficiency results.

Benchmark Runs

According to the above mentioned scope of Ford's linear applications three input decks were generated which were run on a different number of processors and with different output options (ascii-formatted punch, or binary OP2-output). The one processor runs were carried out with the serial MSC.NASTRAN version whereas for the multiple processor runs a development version of the distributed memory parallel code was used.

Run 1 - Focus Model under 1 loading condition
This run is to simulate Ford's regular standard loadcase analysis task. The full body model used here is a state-of-the-art model as used in recent development programs.

Run 2 - Focus Model under 20 loading conditions
This model was generated to investigate the parallel performance for Ford's unit load analysis which is the first analysis step of our fatigue life assessment. Since the inertia relief which is normally required has not been tested in the parallel MSC.NASTRAN version yet, we used the model from run 1 and repeated the same load condition 20 times. This means we did 1 matrix decomposition with 20 subsequent forward-backward solutions.

Run 3 - Cougar Model under 1 loading condition
This model was only used to investigate how the parallel performance changes with the model (matrix) size. Scalability is one of the performance criteria of parallel algorithms and architectures.
The following table summarises the elapsed computer time used for the test cases on the IBM workstation cluster at Ford Cologne. Also the CPU times for a serial C90 run are included. Whereas the IBM systems were dedicated for the runs, the Cray was loaded with normal production work at the time of the benchmark. Therefore for the C90 the CPU time was included in the table rather than the elapsed time.

Type	Processors	Focus				Cougar	
		1 load set		20 load sets		1 load set	
		punch	op2	punch	op2	punch	op2
Workstation Cluster (elapsed seconds)	1	8458	8405	16695	12913	2627	2557
	2	5133	4966	10622	7085	1506	1457
	4	3607	3335	7817	4326	935	798
	8	2450	2235	6748	3182	828	762
Cray C90 (CPU seconds)	1	2738	2539	7167	3354	1112	1019

Table 2: Benchmark Results on IBM Workstation Cluster for Solution 101

Figure 6 shows the comparison of elapsed times on the IBM workstation cluster, the compute servers IBM SP LA and IBM SP Pouk and the Cray C90 located at Ford Cologne for the Focus with one load set and punch output. Runs on the IBM SP LA were only carried out for SOL101 as worst case due to the largest inter-processor communication compared to other parallel MSC.NASTRAN solution types.
Figure 6 also shows speed-up figures for the workstation cluster and the IBM SP systems. Speed-up was defined as the ratio of elapsed time for a serial run divided by the elapsed time on the same platform for 2, 4 or 8 processors.
.

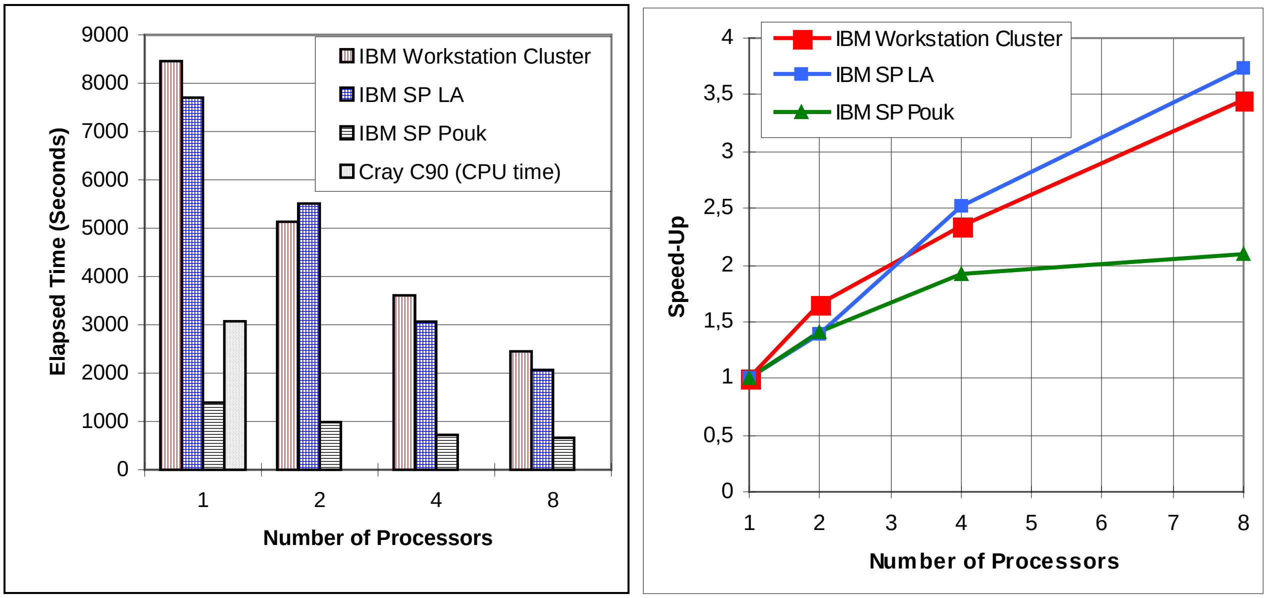


Figure 6: Elapsed Time and Speed-Up, Full Focus Model, Solution 101, 1 Loadcase, Punch Output

The IBM POWER2 platforms at Ford (workstation cluster) and MSC (IBM SP LA) show a very similar performance with respect to the sequential SOL101 benchmark case. Slight differences are caused by a different disk speed and a different memory bandwidth. Both platforms show good speed-up with 2, 4 and 8 processors.
The latest technology POWER3 compute server at IBM (IBM SP Pouk) shows a significantly shorter elapsed time for the single processor run. This is caused by the higher CPU performance and the significantly faster SSA disk subsystem. The speed-up for the benchmark case on this system with multiple processors is not as good as for the POWER2 systems. The reason is that the software modules SEQP and PRESOL, which perform the domain decomposition and calculate some additional data needed for the parallel runs, run in sequential mode, i.e. on a single processor even in multiprocessor runs. These modules take together 325 of 665 seconds total on POWER3 for the 8 processor case. On the slower POWER2 hardware this ratio is not as bad (810 of 2063 seconds total). In addition both the workstation cluster and the IBM SP LA compute server benefit more from I/O caching with a higher number of processors. On the IBM SP Pouk the I/O of the serial run is already close to optimal. MSC.Software is currently tuning above modules for the V70.7 production release.
For all benchmark cases, the turnaround times on the workstation cluster with 8 processors were shorter than on the Cray C90. As the workstations currently used at Ford for production work are significantly more powerful than the evaluated workstations even better results are expected when using the software in our production environment.

When comparing the speed-up values between the punch output option and the OP2 output option it can be seen that the punch option leads to much slower values especially for the model with 20 load sets and one decomposition. The additional effort for each executed forward-backward solution (done in parallel) is only minor in comparison to the additional effort to create the punch output which increases the serial portion of the analysis. The recommendation is to select either OP2 output or to make use of the capability of PATRAN to look at the individual xdb databases on each contributing workstation without a need to consolidate them.

Figure 7 shows a speed-up comparison for a different number of load cases and model sizes.

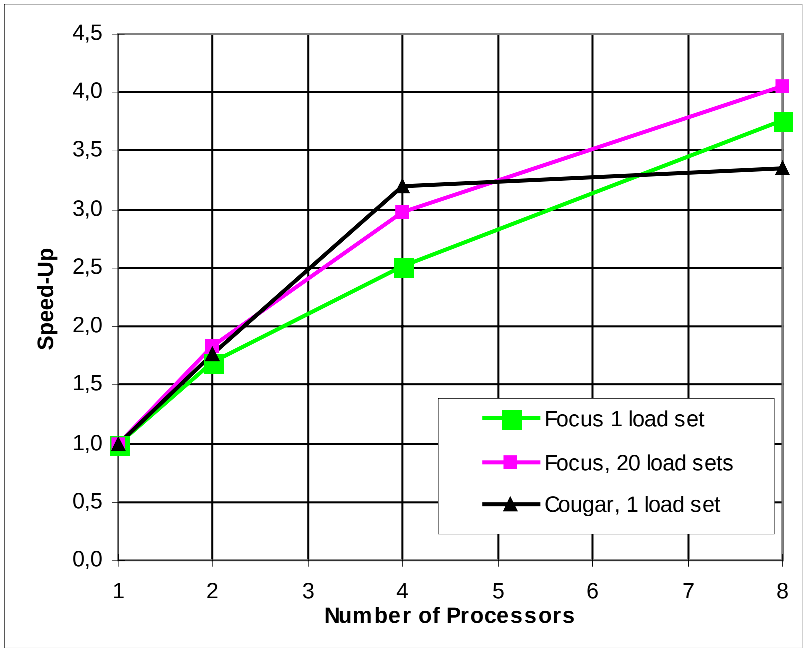


Figure 7: Speed-up Comparison for Different Number of Load Cases and Model Sizes, Full Focus Model, Solution 101, OP2 Output

The Focus model demonstrates that a better speed-up can be reached with a larger number of load sets. The comparison between the larger Focus model and the smaller Cougar model shows that the code scales up to a larger number of processors with larger model sizes.

Figure 8 shows the maximum local disk space and total local I/O traffic. It can be seen that both scale nicely. The reason is the geometric domain decomposition which splits the model across processors. Thus even on workstations with limited disk space larger type models can be analysed. Also the influence of the I/O subsystem to access the disks is decreasing with the number of processors. This is one significant reason why the turnaround times for the workstation cluster and the IBM SP Pouk, which has a much faster I/O than the cluster, are converging with increasing number of processors.

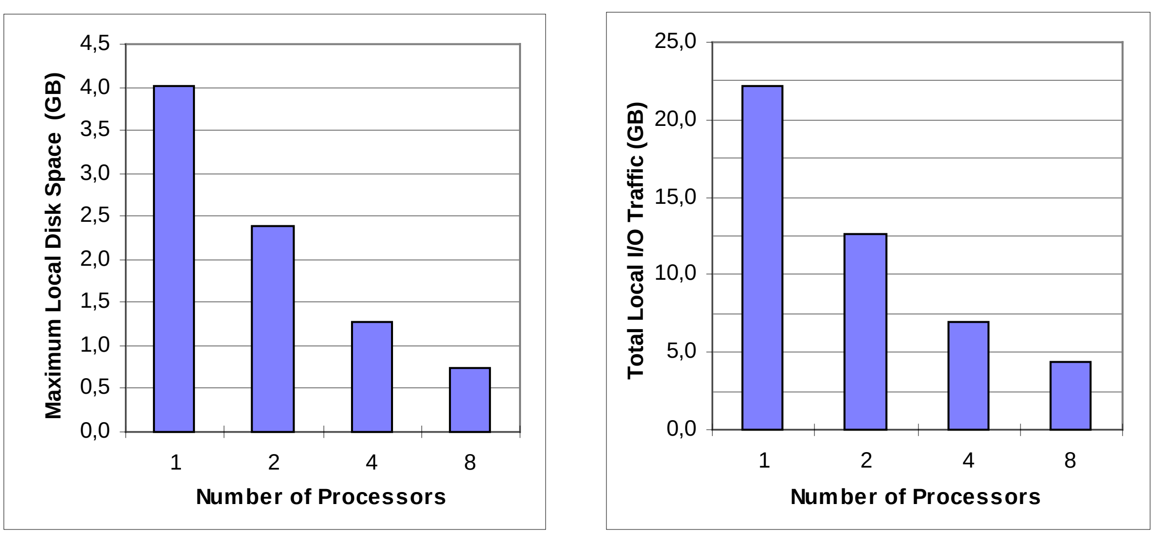


Figure 8: Maximum Local Disk Space Requirement and Total Local I/O Traffic, Full Focus, SOL101, 1 Load Set, Punch Output

3.3.2 Normal Modes (solution 103)

Introduction

This section covers the analysis of the dynamics of a body-in-white structure. As shown in Figure 2, modal analysis accounts for the majority of Nastran jobs at Ford. For an NVH assessment, the alignment of global modes, frequency response functions and forced responses are key parameters.
As a typical case, the modal analysis (SOL103) of a half BIW is computed. Not only global modes can be identified, but these computed modes can be later used in a restart run to compute forced response functions with SOL111. At Ford modal analysis of a Body Structure covers the frequency range up to 600 Hz with frequency response functions computed up to 500 Hz. Residual flexibility vectors are computed as well to reduce the error of the modal synthesis used to compute the frequency response.
In order to investigate the advantages of executing MSC.NASTRAN on a distributed memory architecture, modal analysis for two frequency ranges, 0-300 Hz and 0-600 Hz were computed.
We selected a half body model since the modal behaviour can be separated into a symmetric and an anti-symmetric case.

Model Description

The model used for this pilot is a typical real world application from Body NVH at Ford. For the modal and forced response analysis, a half body structure model of the Focus without closures is used. It is basically the same mesh as above mentioned model 1, Figure 4, with both subframes removed and only the right hand side used. This model consists of 86,235 GRIDs, 78266 QUAD and 4,643 TRIA elements.

Benchmark Runs

Run 4a - Modal analysis 0-300 Hz
The modal analysis run computes 173 eigenvalues in the frequency range 0-300 Hz. For the parallel modal analysis job, the frequency range is automatically split into separate ranges for each processor. For the 8 processor run, the split is shown in Table 3.

Processor	1	2	3	4	5	6	7	8
0-300 Hz	18	15	18	21	24	23	24	30
0-600 Hz	72	101	97	106	100	108	90	98

Table 3: Number of Modes computed on each Processor during an 8 Processor Run

Although the processors are not equally loaded, the default split of the frequency range works well for this development. Manual adjustments by the user are not necessary. Figure 9 shows elapsed times. The small number of eigenvalues computed per processor leads only to a speed-up of 2.7 for an 8 processor job on the cluster as shown in Figure 10.

Run 4b - Modal analysis 0-600 Hz
This run for a frequency range of 0-600 Hz was carried out to show the influence of the numbers of eigenvalues. A total of 772 modes were computed. Table 3 shows the achieved distribution of computed modes using the automatic frequency range split. Again the processors are rather evenly loaded.
Figure 10 shows that the speed-up for this job is better than for run 4a. The speed-up of 3.3 using 8 processors on the cluster for the frequency range 0-600Hz demonstrates that a parallel execution improves the turnaround time for large jobs (significant improvement on the IBM SP Pouk). Turnaround time and speed-up figures for the 2 processor 0-600Hz cluster run are not included in Figure 9 and 10 as there was insufficient disk space on all of the slave workstations. However the disk space requirement decreases with increasing number of processors. Whereas the one processor run requires 10.8 GB, the 8 processor run needs 8.4 GB on the master workstation and a maximum of 4.9 GB on the slave workstations (Figure 11). Using the parallel version of MSC.NASTRAN, it is possible to compute the modal analysis up to 600 Hz of a half BIW structure within 11hours on the workstation cluster, thus an over night turnaround can be achieved without using a compute server.
Nastran kernel tuning for AIX is underway which is expected to reduce turnaround time for serial and parallel runs.

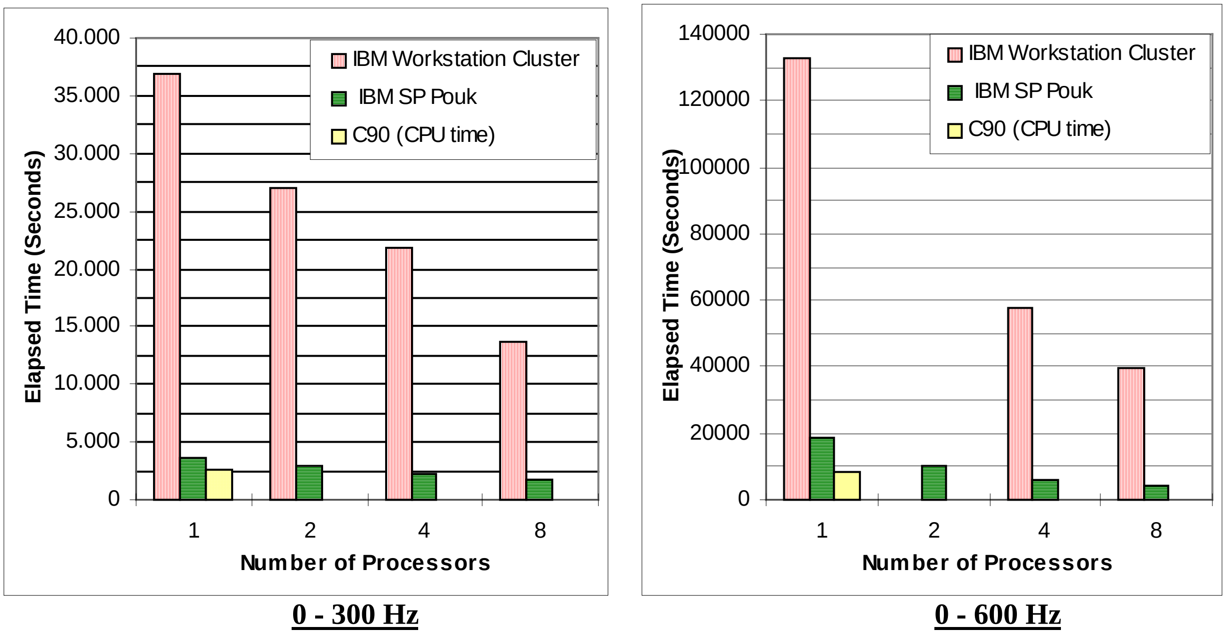


Figure 9: Elapsed Times for Half Focus Model, Solution 103

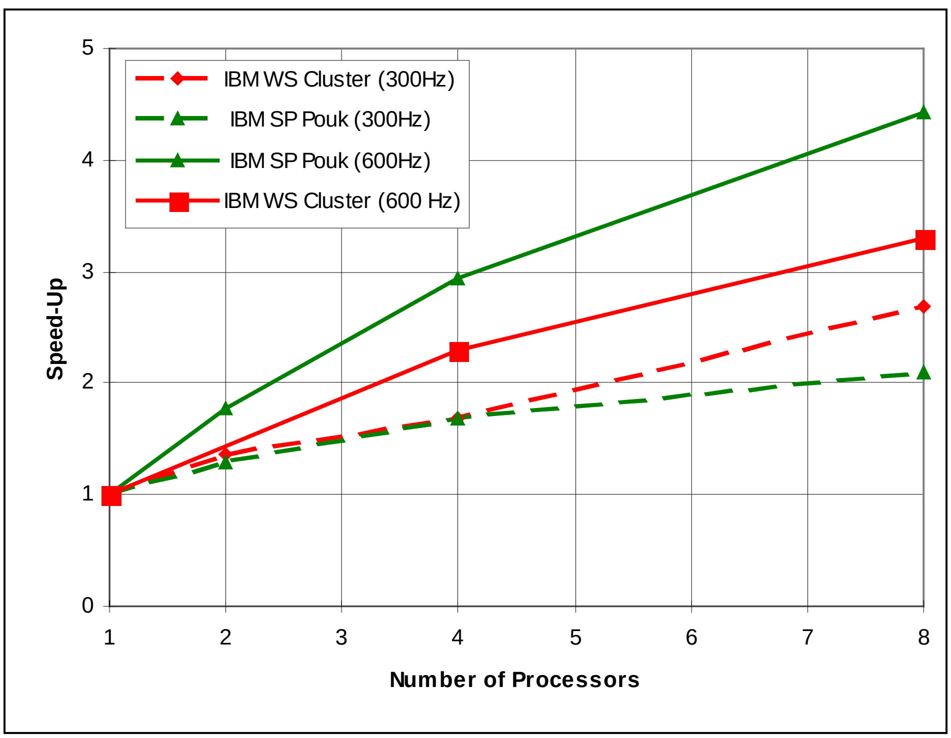


Figure 10: Speed-Up for Half Focus Model, Solution 103

Figure 11 shows that the local disk space requirement is significantly reduced for the slave processors with increasing number of processors. The disk I/0 also decreases with increasing number of processors.

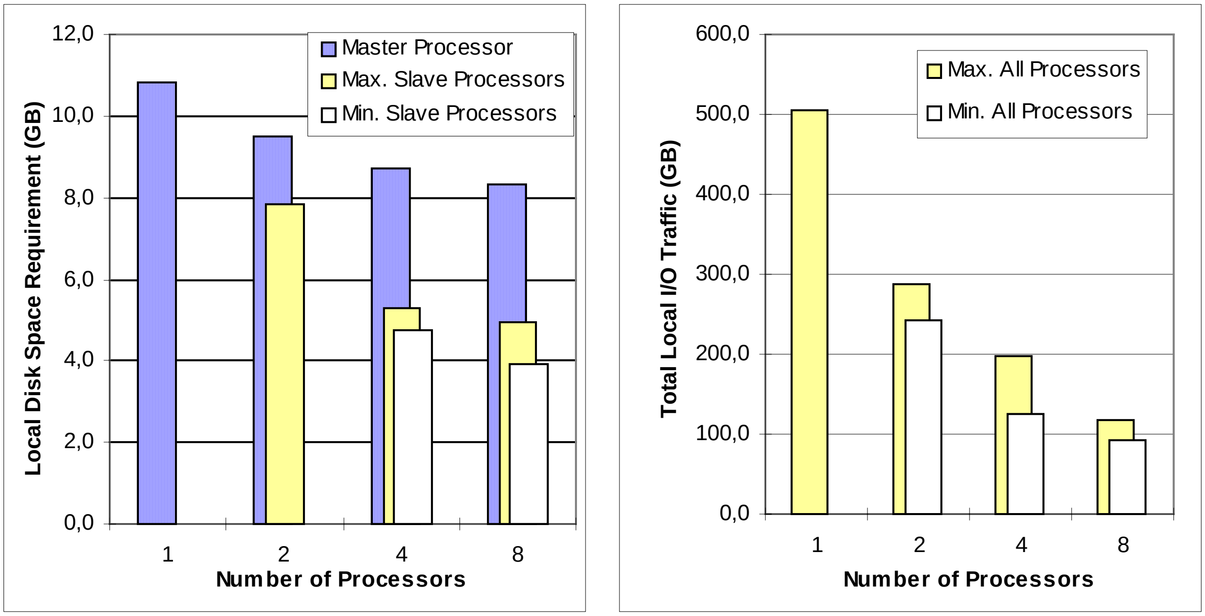


Figure 11: Local Disk Space Requirement and Total Local I/O Traffic, Half Focus, SOL103, 0 - 600 Hz

3.3.3 Direct Frequency Response (solution 108)

Introduction

Although frequency response functions are usually computed at Ford with the modal frequency method (SOL111), the direct frequency response method was used in this pilot. SOL108 is an exact solution whereas the accuracy of SOL111 depends on the number of modes in the normal modes analysis. Also SOL108 does not require large data bases. The question was whether a parallel execution of the direct method can overcome its huge CPU time requirements.

Model Description

The same model as for the modal analysis was used.

Benchmark Runs

Run 5 - Direct Frequency Response
The frequency response functions for 24 excitation points and 32 frequencies were computed. The parallel direct frequency solution splits the frequency range across each processor, i.e. when using 8 processors each processor will compute 4 of the 32 frequencies.
The inherent parallelism of the solution results in an excellent speed-up of approximately 6 on 8 processors, Figure 12. An even better speed-up is expected when more frequencies are computed. Compared with results of the benchmark runs for SOL101 and SOL103 this solution delivers the best speed-up. The low speed-up for the 2 processor run on the workstation cluster needs further investigation. Omitting the 2 processor run, the speed-up curves for the workstation cluster and IBM SP Pouk run in parallel, i.e. the scalability for this job does not depend on the architecture.

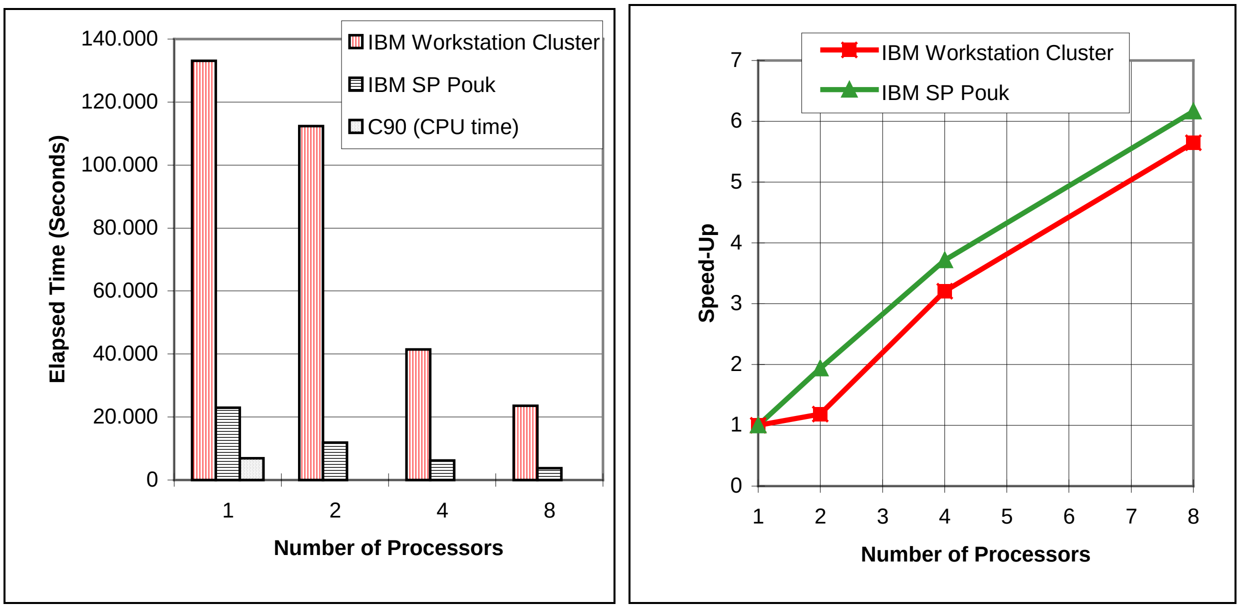


Figure 12: Elapsed Time and Speed-Up for Half Focus Model, SOL108, 24 Load Cases, 32 Freq.

Even when using multiple processors with SOL108, the time to compute the frequency response for a large frequency range is higher than with SOL111. However, SOL108 may be more effective when computing the response for a narrow band of high frequencies that would otherwise require a modal analysis with a huge number of modes. As an example, the frequency response analysis for 400 - 432 Hz with SOL108 takes 23562 seconds on a workstation cluster whereas SOL103 alone requires 39000 seconds.

SOL108 is dominated by the complex sparse decomposition module, which can be tuned to run very efficiently on RISC and Vector machines. When properly tuned, the memory bandwidth requirement is low, as in the case of the Cray version. A tuned AIX version will be available with MSC.NASTRAN V70.7. IBM expects that this will bring the turnaround time on the IBM SP Pouk much closer to the C90 time.

Figure 13 shows for the SOL108 benchmark case that the local disk space requirement and the total local I/O traffic scale well with an increasing number of processors. The requirements for both, disk space and total local I/O were almost identical between processors used.

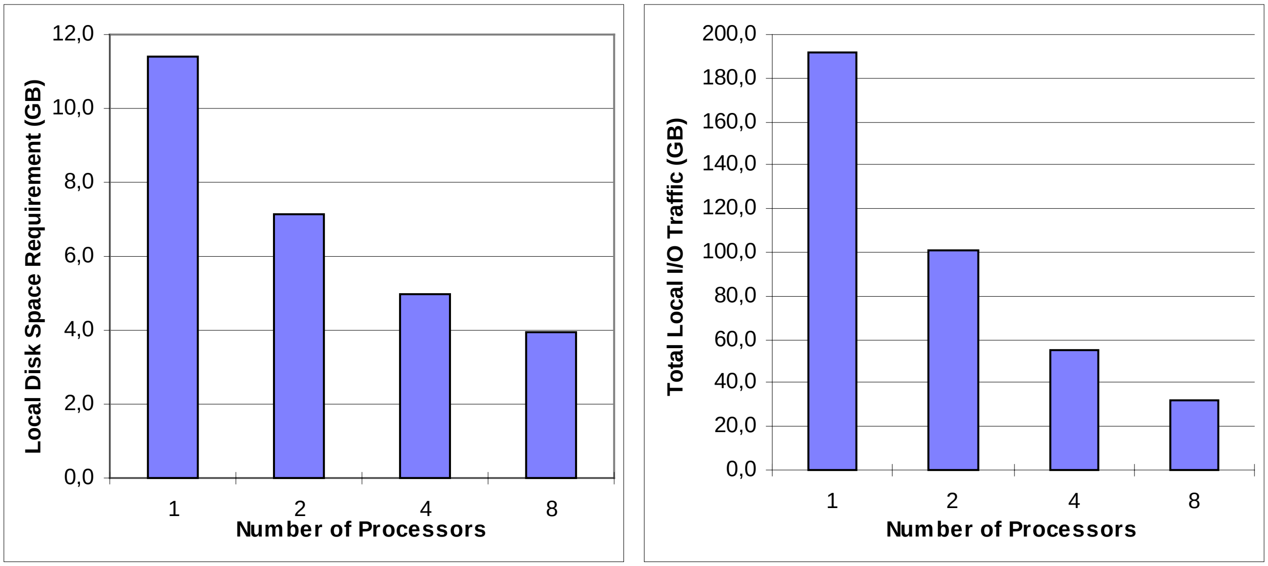


Figure 13: Local Disk Space Requirement Requirement and Total Local I/O Traffic, Half Focus, SOL108, 0 - 600 Hz

4. Conclusions

- The development version 70.7 of parallel MSC.NASTRAN for distributed memory architectures shows great potential for typical jobs used for the analysis of Ford body structures. Good speed-up with an increasing number of processors is achieved not only on the distributed memory compute server IBM SP but also on the cluster of 8 IBM 590 workstations connected with a 100Mbit switched Ethernet.
- The cluster of IBM workstations showed better turnaround times than the Cray C90 for 8 processors for SOL101. When using latest technology workstations the cluster is expected to show even better turnaround times with also superior performance to the C90 for SOL103 and 108. The turnaround target of overnight completion could be achieved with 8 processors for all benchmark cases.
- The compute server IBM SP LA with similar processors to the cluster workstations showed similar performance to the cluster. The IBM SP Pouk with latest technology processors and I/O subsystem was performing better than the Cray with 8 processors for all benchmark cases.
- Small changes to the bulk data deck were required for parallel execution such as the removal of superelement definitions (this will be fixed in version 70.7).
- The SOL101 benchmarks showed that the punch output not only required significant computing resources but also had a negative effect on speed-up in parallel runs especially for models with many load cases. It is therefore recommended to use binary output options such as OP2 wherever possible.
- For all solutions the local disk space requirement was reduced with an increasing number of processors. For SOL101 and SOL108 the largest improvement could be seen. This allows large jobs to be run on a cluster of workstations which are equipped with only limited disk space.
- For SOL103 a good parallel speed-up could only be achieved for a larger number of modes (e.g. 800 modes and above).
- The robustness of the latest version of the code proved to be good during the benchmark period.
- The influence of communications between processors was less significant than expected by Ford even for SOL101 which has the largest inter-processor communication requirement.

5. Next Steps

- Ford will carry out benchmark runs with the distributed memory parallel version 70.7 of MSC.NASTRAN on a Ford compute server once this version is officially released and consider production implementation depending on results.
- As the evaluation of the distributed memory code proved to be successful, Ford will extend the pilot to the largest analyses (SOL103 with 3 mil degrees of freedom).
- Ford is planning to evaluate the code in a Ford production workstation cluster environment. Prerequisites are job scheduling and load balancing tools as well as dedicated disk space on all workstations to enable a robust environment. Special care has to be taken for restart jobs that use existing databases from previous runs.
- Ford will encourage MSC.Software to provide an official quality assured version of the workstation cluster code not too long after the compute server version.
- Ford will encourage MSC.Software to integrate their distributed code with workload management software to improve robustness of the total workstation cluster solution in a production environment
- If the domain decomposition approach of SOL101 proves to be successful, MSC.Software plans to also parallelize SOL103 and SOL108 in a similar way which should lead to further turnaround improvements.
- Ford will encourage MSC.Software to deliver a fully operational version of SOL111 (planned for production version 70.7) and for all other structured MSC.NASTRAN solutions.
- Ford will test SOL101 with inertia relief.
- Ford participants will share the experiences made during the evaluation with Ford CAE engineers and Systems personnel world-wide.

6. Acknowledgements

We would like to thank the following people for their contributions to the above effort:

- Dr. Holger Holthoff, IBM Technical Support, for his contributions to the workstation cluster set-up and the benchmark runs at IBM Poughkeepsie/USA.
- Dr. Gerhard Zellermann, Ford Vehicle CAE Systems, for sponsoring the project from the Ford side and for providing guidance and advise throughout the project.
- Dr. Louis Komzsik, director numerical methods development from MSC.Software Corporation, USA for providing sponsorship from MSC side and supporting availability of the latest code versions.
- Peter Klur, IBM, cluster hardware and software upgrades and sales support.
- Horst Joerres, Ford network support group, for providing networking infrastructure
- Norbert Anselm, Ford workstation Systems group, for his support in providing IBM workstations.

7. References

[1] Europort-1 Management Group, Europort-1: Survey and Results - 14 Parallel Codes Ready for Industrial Use in Fluid Dynamics and Structural Mechanics, Summary Brochure, 1996

[2] Dr. Louis Komzsik, Dr. Stefan Mayer, Petra Poschmann, Paul Vanderwalt, Reza Sadeghi, MSC.Software Corp., USA, A breakthrough in parallel solutions of MSC.Software, 1st MSC Worldwide Automotive Conference, September 1999

[3] Ulrich Viersbach, Mike Hesse, Gerhard Zellermann, Juergen Schlanke: Parallelization - Readiness of Workstation Clustering for Computer Aided Engineering, High Performance Computing on Hewlett-Packard Systems (HiPer97), November 1997

[4] U. Viersbach, Z. Tastan, M. Hesse, Workload Management at Ford Europe, Fluent/UNS Spring '98 Newsletter, WEB Address: http://www.fluent.com/news/spring98/ford.htm

[5] Z.Tastan, U. Viersbach, M. Hesse, Parallel Workstation Clustering of Fluent/UNS using LSF, German Fluent User Group Conference, September 1998

[6] Z. Tastan, Dr. M. Matthes, U. Viersbach, Parallel Workstation Clustering of Fluent/UNS using LSF, 1999 Fluent Automotive User Group Meeting, Michigan, May 1999

[7] Rudy Chukran, Accelerating AIX, Performance Tuning for Programmers and System Administrators, Addison Wesley, ISBN 0-201-63382-5, 1998