Automated Component Modal Synthesis with Parallel Processing

Paresh Murthy, Petra Poschmann, Mike Reymond, Peter Schartz, Charles T. Wilson,
MSC Software Corporation, 2975 Redhill Ave, Costa Mesa, CA 92626

Abstract

Component Modal Synthesis (CMS) has been a powerful tool for dynamicists for several decades. In this paper we discuss the combination of CMS with automatic domain decomposition and parallel processing. This can greatly reduce the time and cost of vehicle NVH analysis. Large finite element models are automatically divided into hundreds of sub-models or superelements, which are then analyzed in parallel. Methods of improving the accuracy of the CMS are discussed. The method called Parallel Automated Component Modal Synthesis (PACMS) is implemented in the MSC.Nastran commercial finite element program. Performance and accuracy of the method are compared to the direct frequency response solution and the modal frequency response solutions using complete vehicle modes. Results are presented for a large automotive analysis with one million degrees of freedom.

Introduction

For the past several years, computer manufacturers have continued to develop computer systems that perform at an increasing rate of speed. Unfortunately, the improvements in hardware and system performance are by and large not able to keep pace with the demands of the Computer Aided Engineering (CAE) industry in general, and with Finite Element Analysis (FEA) in particular. Implicit in recent computer systems innovation is the use of *parallel processing* by FEA applications. Parallel processing simply means that an application, such as MSC.Nastran, executes a single task on multiple processors in parallel.

Historically, CAE applications fall into one of two categories: those which are amenable to parallelization, and those which are not. For various reasons, MSC.Nastran has proven to be relatively difficult to adapt to parallel processing with any effectiveness, although it has remained a persistent research topic. A major stepping stone has been established, however, with the advent and maturation of various *nested dissection* domain decomposition algorithms. These algorithms, based on graph theory, were first introduced in MSC.Nastran to perform sparse matrix reordering for matrix factorization. More recently, they have been adapted to more general use as domain decomposition algorithms within MSC.Nastran. Two such algorithms, known as Extreme [1] and Metis [2] are employed in MSC.Nastran today. A third has been developed internally at MSC, which is better suited for MSC.Nastran.

The purpose of domain decomposition in MSC.Nastran is to divide the model into discrete sub-models, or *superelements*. These superelements may naturally be analyzed independently in parallel with a conventional Component Modal Synthesis (CMS) technique. A key objective of the domain decomposition algorithm is to minimize the amount of intersection among the superelements, which in turn minimizes the global boundary of the structure.

Thus, the combination of an automatic domain decomposition capability, coupled with a CMS solution, gives rise to an Automated CMS capability (ACMS). When multiple processors are employed to capitalize on the independent nature of the automatically generated superelements, we call it Parallel ACMS, or PACMS. The combination of an older, more conventional approach (CMS), with a relatively novel capability (domain decomposition) has resulted in a powerful solution capability with very promising performance indications.

Target problems for PACMS include any large dynamic analysis. This includes automotive analyses such as NVH, fatigue and durability studies.

Theory

To understand the theory of how PACMS operates, one might study independently the distinct subjects of CMS and graph partitioning. The only essential difference between a PACMS solution and a multi-superelement conventional CMS, is that in PACMS, the superelements have been automatically generated without intervention from, or indeed any knowledge of, the user.

CMS is a well-established technique for calculating the modes of structures, which are often described by separate sub-models or *components*. Component based modeling may be a necessity in engineering environments where portions of the structure are designed separately from one another. For each component, interior degrees of freedom (i.e., degrees of freedom (dof) which to not interact with those of other components) are approximated by a reduced set of basis vectors. The eigenvectors of the structure's interior points, excited in a given frequency range, are the most commonly used basis vectors. The component approximations are then combined to produce a global approximation of the entire structure.

The number of eigenvectors required to represent the motion of the interior of a given component is typically much smaller than the number of physical interior dof. This *reducing out*, or omitting, of interior dof is usually responsible for a sizeable reduction in compute resources required to analyze the structure.

With any approximation comes a loss of accuracy. When employing the modal dynamic approach, an accepted guideline is to compute the modes of the structure up to two to three times higher than the highest desired dynamic frequency. Just as the modal approach approximates the direct approach in dynamics, CMS approximates the eigensolution of the modal approach. It should also be noted that ACMS is, in essence, a modified modal approach; that is, it is not a new approach per se, but only a different way of formulating the modal approach.

A commonly accepted criterion for component eigensolution frequency ranges is to keep all the modes in a range 1.5 to two times larger than the global requirement. The ACMS solution provides a real bulk data parameter, UPFAC, allowing the user to select the factor applied to the desired frequency range, which is used to compute the frequency range used for upstream components. The default value for UPFAC is 2.0, meaning that the component mode shapes corresponding to a frequency range two times higher than the desired range are computed. For example, if the EIGR frequency range is 200 hertz, then a value of 2.0 would provide for modes being retained upstream, up to 400 hertz.

The PACMS solution augments the component eigensolutions, and thus improves the approximation, by employing *residual vectors*. Both inertia relief effects (low frequency, below the first modal frequency) and static shapes (high frequency, above the highest modal frequency) are computed and retained. Inertia relief modes are calculated by activating the RESVINER bulk data parameter. Static shape augmentation is achieve by using the RESVEC bulk data parameter. By including these additional shapes, the loss of accuracy, which might otherwise occur, is mitigated.

A thorough treatment of CMS is contained in [3].

Graph partitioning has been an active research area for many years. Efforts at the University of Minnesota and at Silicon Graphics, Inc., in the mid 1990's, resulted in the implementation of efficient and effective nested dissection algorithms. These algorithms approach an optimal solution to the traditional graph partitioning problem, namely,

Given a graph G=(V,E), partition it into k parts, such that

- Each part has roughly the same number of vertices (*constraint*)
- The number of edges that straddle the partitions is minimized (*objective*)

A good introductory treatment of this topic is available at the University of Minnesota website at http://www-users.cs.umn.edu/~karypis/metis/. For a more thorough study, see [4].

Implementation

PACMS is available in MSC.Nastran 2001, on all platforms which support MSC.Nastran Distributed Memory Parallel (DMP) processing. PACMS is supported in MSC.Nastran solutions 103, 108, 109, 111, and 112. In order to execute PACMS, the following steps are necessary:

1. Ensure you are running on a MSC.Nastran DMP-supported platform.
2. Include the Bulk Data card PARAM,ACMS,YES.
3. Specify “dmp=” on the MSC.Nastran command line.

The YES value for the ACMS parameter will direct MSC.Nastran to automatically divide the model into an appropriate number of superelements. A CMS solution is then automatically applied to the model, and results are produced as if you did not use ACMS. Specifying values greater than 1 for the dmp command line keyword will result in the problem being solved on multiple processors in parallel.

Residual vectors are employed by default. Thus, PARAM,ACMS,YES implied that a YES value is also given to bulk data parameters RESVINER and RESVEC.

The ACMS solution triggered by PARAM,ACMS,YES invokes a domain decomposition algorithm from within the SEQP module. The global collection of grid points is divided into NTIPS domains, where NTIPS is selected by the program. Typical values for NTIPS are 64 or 128. The goals of the domain decomposition algorithms employed are twofold. The common boundary is minimized, which reduces the size and complexity of residual superelement (SE) processing. In addition, the size of each domain is kept roughly equal, in terms of number of grid points. This will aid in load balancing when each domain is processed separately on parallel processors.

From the domain information, tip SEs and an SE tree are generated. By default, a binary multilevel tree is generated. On user option, a single level tree can be used. If multiple processors are available (dmp>1), then the first NTIPS/dmp tip SEs (and their subtrees if multilevel) are processed on processor 1; the next NTIPS/dmp tips are processed on processor 2, and so on. Data blocks are communicated back to the master processor for residual SE processing, which represents the global boundary determined during domain decomposition. For dynamic analysis, loads are generated and the dynamic analysis is performed. Subsequently, the SE tree is traversed to recover data back out at the tip superelements. Merging of data recovery is a two step process--vertically and then horizontally. First, the DISOFPM module (with family inputs) merges the results on each processor (vertically down the superelement tree). Then, DISOFPM/S module calls merge the results from the first step, from the slave processors to the master processor (horizontally across processors and the superelement tree).

Case Studies

To demonstrate the ACMS solution, presented here are PACMS results. Frequency response analyses of automotive models of significant size and complexity are used as examples. First, the accuracy of PACMS is compared to equivalent direct and modal approach results. The effect of using residual vectors, as well as that of increasing the upstream component frequency range, is examined.

Serial and parallel ACMS performance is then demonstrated. The computer resources required for PACMS are shown compared to conventional solution techniques.

Model Descriptions

Models are described generically, in terms of vital statistics such as number of grid points, elements, and forcing frequencies. For the modal and ACMS approaches, a description of the eigensolution is also presented.

No.	Name	NGRIDs	Nelem	Naset	NLOAD	Fr.range (hz)	Nfreq	EIGR range(hz)	No. modes
1	VLAW	44314	43949	250913	75	255	251	400	303
2	XLBD	170004	164046	975782	150	400	396	600	1293

Each of these models originated in the automotive user community. They may be described as Body-in-white models.

Analysis Descriptions

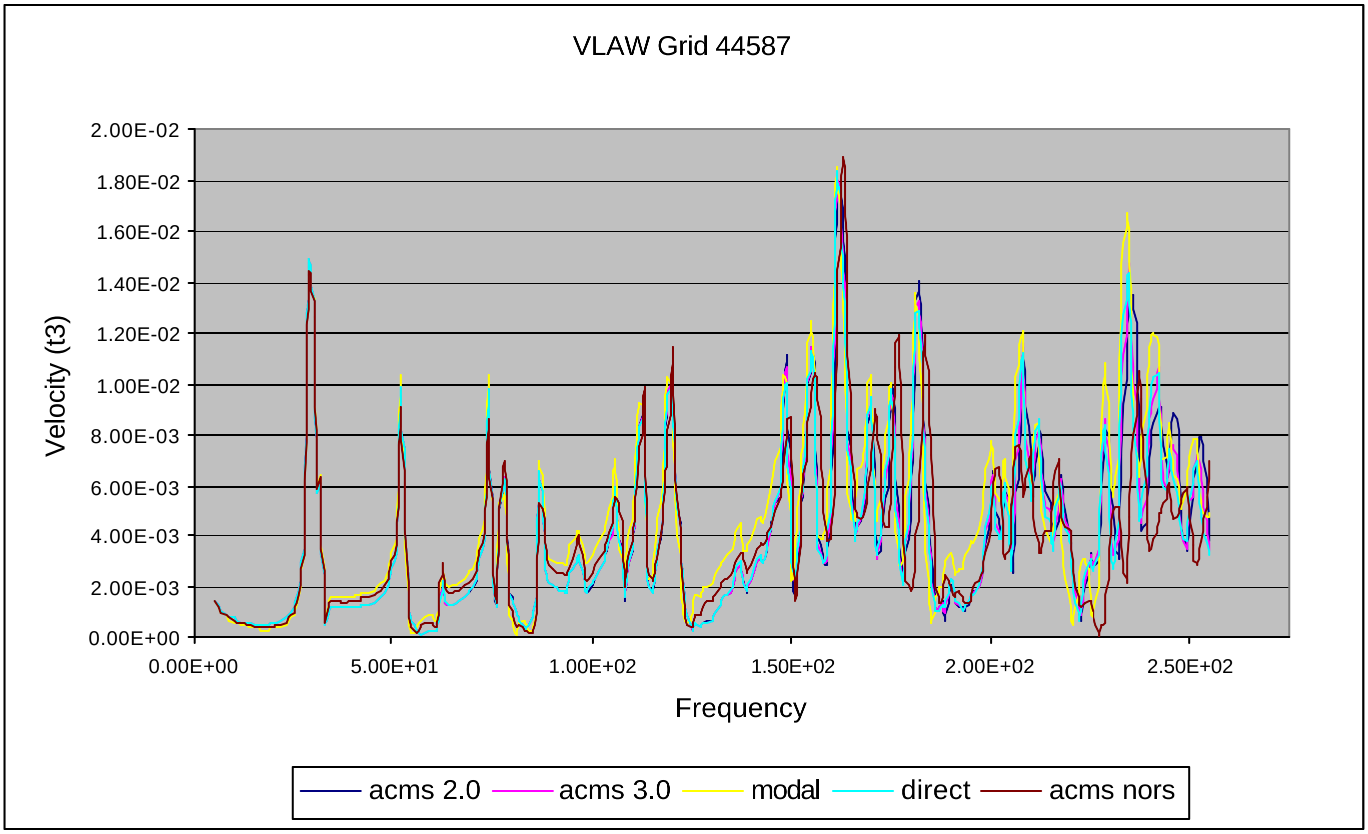
A frequency response analysis was performed for each of the above models. The type of analysis done is prevalent in NVH, also known as "point-mobility" analysis. A small set of grid points is monitored throughout the desired frequency range, for a variety of load cases. This analysis methodology is not the most general, but was chosen to demonstrate PACMS due to its prevalent use in the automotive industry.

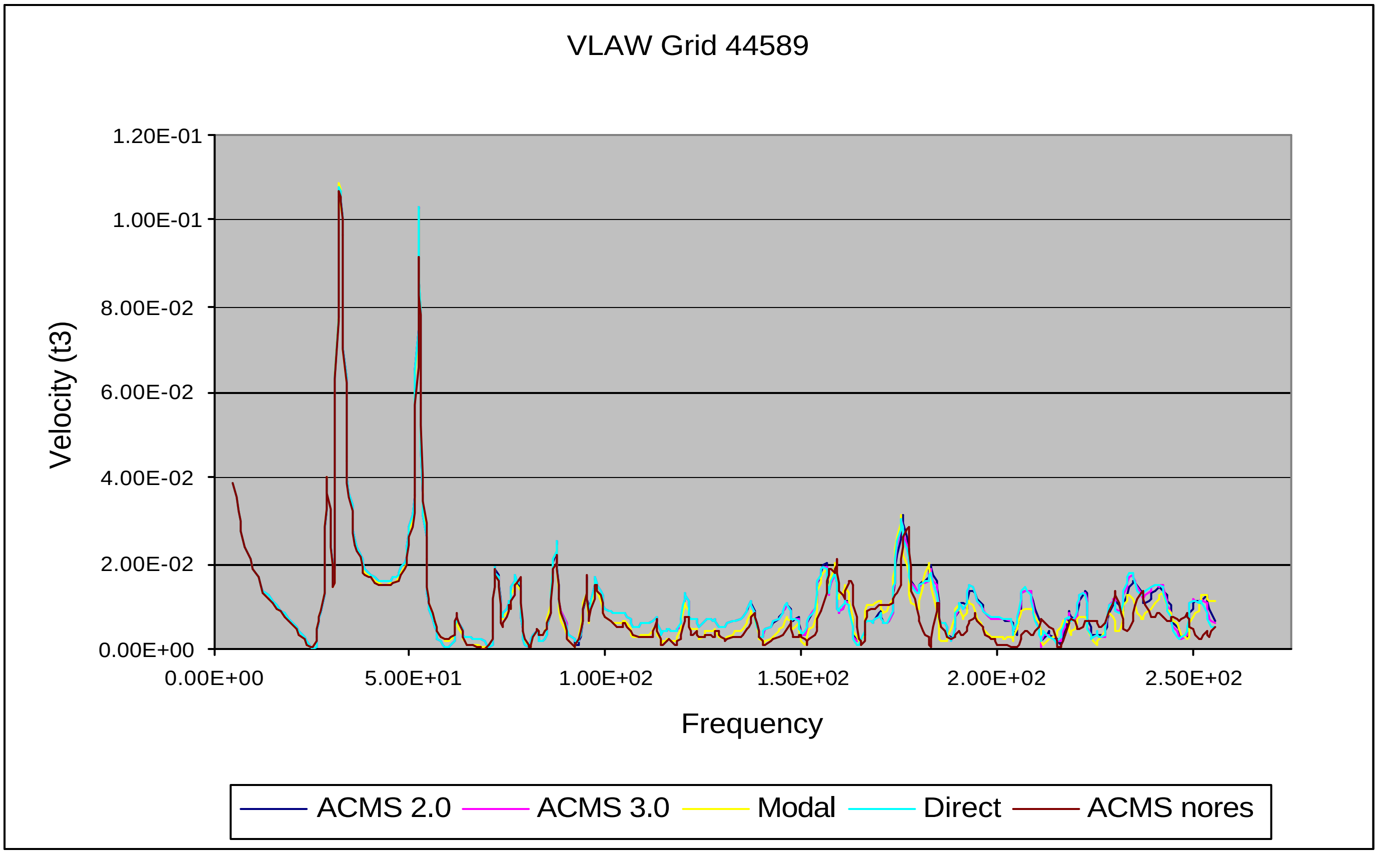
Computer System Description

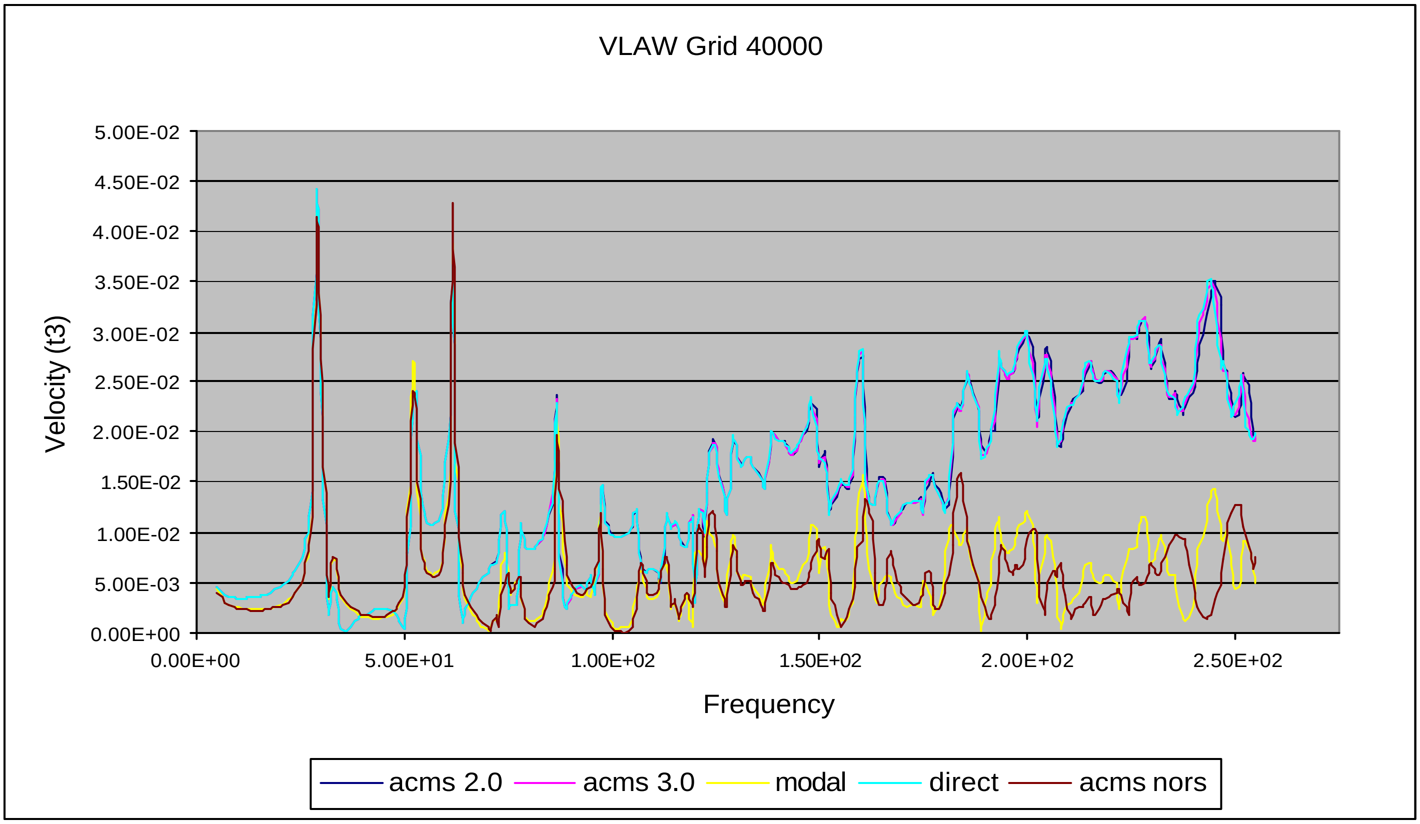
The computer system used to obtain the results presented in this paper was an IBM SP system consisting of three SP3 nodes. Each SP3 node contained four 375 MHz Power3 CPUs and 4GB of memory. Each node also contained eight 18GB disk devices use for scratch disk.

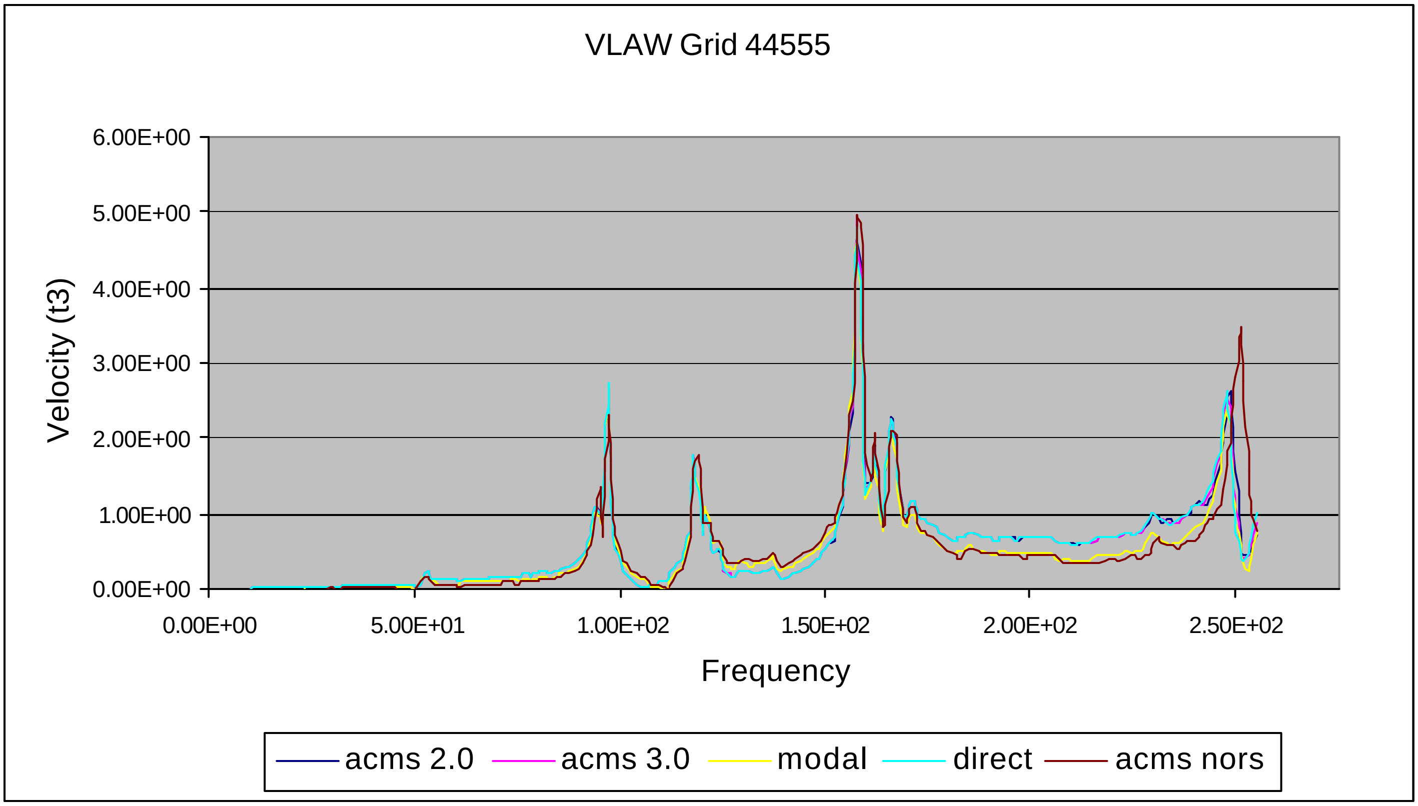
Analysis Results

The following charts compare the frequency response for a subset of the grid points of interest. Shown are results for the Direct approach, the Modal approach, and ACMS. In addition, key ACMS parameters are varied in order to demonstrate their effect on the results. The default ACMS run includes use of residual vectors, as well as an upstream frequency range factor (PARAM,UPFAC) of 2.0. Experiments are made by removing the effect of the residual vectors, as well as by varying the value of PARAM,UPFAC.


VLAW Grid 44587
Velocity (t3)
2.00E-02
1.80E-02
1.60E-02
1.40E-02
1.20E-02
1.00E-02
8.00E-03
6.00E-03
4.00E-03
2.00E-03
0.00E+00
0.00E+00
5.00E+01
1.00E+02
1.50E+02
2.00E+02
2.50E+02
Frequency
acms 2.0
acms 3.0
modal
direct
acms nors


VLAW Grid 44589
Velocity (t3)
1.20E-01
1.00E-01
8.00E-02
6.00E-02
4.00E-02
2.00E-02
0.00E+00
0.00E+00
5.00E+01
1.00E+02
1.50E+02
2.00E+02
2.50E+02
Frequency
ACMS 2.0
ACMS 3.0
Modal
Direct
ACMS nores


VLAW Grid 40000
Velocity (t3)
5.00E-02
4.50E-02
4.00E-02
3.50E-02
3.00E-02
2.50E-02
2.00E-02
1.50E-02
1.00E-02
5.00E-03
0.00E+00
0.00E+00
5.00E+01
1.00E+02
1.50E+02
2.00E+02
2.50E+02
Frequency
acms 2.0
acms 3.0
modal
direct
acms nors


VLAW Grid 44555
Velocity (t3)
6.00E+00
5.00E+00
4.00E+00
3.00E+00
2.00E+00
1.00E+00
0.00E+00
0.00E+00
5.00E+01
1.00E+02
1.50E+02
2.00E+02
2.50E+02
Frequency
acms 2.0
acms 3.0
modal
direct
acms nors

In these charts, the *nores* data represent ACMS results without the use of residual vectors. The *UPFAC* data represents ACMS results obtained by varying PARAM,UPFAC.

The responses shown, computed with ACMS, correlate quite well with the results from the direct approach. The ACMS response for Grid 40000 is actually closer to the Direct response than is the Modal response.

Performance Data

The following tables show the performance data of the PACMS technique for the given problems. Performance is measured as the total turnaround time (elapsed or wallclock time) for the entire analysis.

XLBD model	Elapsed Sec	Speedup
Default SOL 111	46885	--
PACMS dmp=1	19099	2.45
PACMS dmp=2	12969	3.62
PACMS dmp=4	11302	4.15

It should be noted that the use of residual vectors, as well as increased values of PARAM,UPFAC (the upstream frequency range factor) will inhibit performance. In other words, the price paid for greater accuracy is longer running time. An example of this effect is presented in the following table:

XLBD 103	Resid vect	UPFAC	Elapsed Sec
PACMS dmp=4	No	1.5	6140
PACMS dmp=4	Yes	1.5	6696
PACMS dmp=4	Yes	2.0	7439
PACMS dmp=4	Yes	3.0	9460

Discussion

Based on our case studies, one could conclude that the ACMS solution provides sufficient accuracy for most engineering applications. Given the nature of the approximation of the Direct approach by the Modal approach, it is then not unreasonable to approximate the modal solution with a CMS as is described here. In addition, the use of residual vector calculations, and the availability of PARAM,UPFAC provide the engineer with sufficient controls to increase the accuracy of the solution, should this prove to be necessary. It is useful to note, in the charts presented above, that the ACMS solution does not deteriorate at higher frequency levels, evidence that the typical modal truncation effects inherent in the modal approach have been successfully mitigated.

The runtime performance of ACMS improves upon the conventional modal approach in two ways. First, even with dmp=1 (single processor), ACMS is faster. The reasons for this are difficult to determine. It may be that the large I/O cost of a large conventional eigensolution is all but eliminated; instead, many, smaller eigensolutions are run instead. Additionally, the “reducing out” of component interior degrees of freedom, resulting in a smaller overall solution, may also contribute to the increased performance.

Parallel load balance is highly problem dependent. At the outset of the solution, an attempt at load balance is made by making the domains roughly the same size. However, the overriding performance characteristic is the number of eigenvectors produced by each component; components with more modes are more expensive to compute. Currently, there is nothing preventing components that have large numbers of modes to contribute, from being assigned to a single processor, leaving the other processors possibly under-utilized. This is seen as being a major factor inhibiting parallel speedup.

Conclusions

We have shown that the ACMS solution yields results with excellent correlation to direct dynamic analysis. User controls are implemented which may be used to tune the accuracy level desired, with tradeoffs in performance. The user interface is intended to be simple, to allow immediate adoption by the MSC.Nastran user community.

Performance implications of ACMS, both in serial and in parallel, are very promising. Plans for subsequent releases are to improve load balance by dynamically allocating tip superelements to processors, among other enhancements. Of course, feedback from the field will be most useful in improving ACMS.

It is estimated that 70% of the total MSC.Nastran processing time in major automotive companies is spent computing normal modes, usually as part of a modal dynamic analysis. Any meaningful performance breakthrough for MSC.Nastran normal modes computation would have a direct and profitable benefit on the entire user community. With the release and subsequent maturation of ACMS, such a performance improvement may now be at hand.

References

1. Hendrickson, Bruce, and Rothberg, Edward, “Improving the Runtime Performance and Quality of Nested Dissection Ordering,” *SIAM Journal of Scientific Computing*, Vol. 20, No. 2, 1998, pp. 468-489.

2. Metis. Many publications may be found on the World Wide Web at http://www-users.cs.umn.edu/~karypis/publications.

3. *MSC.Nastran Advanced Dynamic Analysis User’s Guide*, The MacNeal-Schwendler Corporation, 1997.

4. George, Alan, and Liu, Joseph W., *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, 1981.