APPLICATION OF FLANIGAN'S MODE ACCELERATION

IN MSC/NASTRAN VERSION 66

by

Thomas G. Butler
BUTLER ANALYSES

and

John C. Muskivitch
MACNEAL SCHWENDLER

Twice before Chris Flanigan has briefed our MSC finite element community on an alternate approach to improve transient solutions using the modal method.[1,2] Not until his 1988 paper did he present his material in so convincing a fashion that everyone was forced to sit up and take notice. What made his approach so imperative was the situation that the analyst faces when solving his modal transient problem using superelements.

In his earlier 1980[1] paper, he showed that his method produced valid results when verified against the standard modal acceleration solutions in conventional analyses. The nub of his argument, then, was not for increased accuracy, but for savings in solution costs. It was left to the reader to write the DMAP to implement the method. In his 1988[2] paper he electrified us

---

1. An Alternate Method for Mode Acceleration Data Recovery in MSC/NASTRAN, by Christopher C. Flanigan, Proceedings MSC/NASTRAN Conference, March 1981.
2. Accurate and Efficient Mode Acceleration Data Recovery For Super Element Models, by Christopher C. Flanigan, Proceedings of MSC/NASTRAN Conference, March 1988.

all by demonstrating marked deficiencies and intractabilities in using standard modal acceleration in superelement problems. Both of these conditions were relieved by his method. He offered improved accuracy at less cost and more access to control one's problem using superelements. Some DMAP alters were included in his paper, but some steps were only outlined and others relied on external programs.

In this paper, the authors do not attempt to advance what Chris has done, but to dilate the theory on which it is based and make some of the points more acceptable; and show how MSC/NASTRAN Version 66 does a nice job in managing the entire solution. Much of what appeared in Chris' earlier papers will be repeated here for completeness. A number of intervening statements are inserted for clarifying the logic of some steps.

Starting with the statement of the equation of motion in geometric coordinates:

(1) $$[M]_{GG}\{\ddot{x}_G(t)\} + [B]_{GG}\{\dot{x}_G(t)\} + [K]_{GG}\{x_G(t)\} = \{p_G(t)\}$$

an eigenvalue problem can be similarly obtained:

(2) $$[M]_{AA}\{\ddot{x}_A(t)\} + [K]_{AA}\{x_A(t)\} = 0,$$

When the eigenvalue problem is solved, one obtains a set of a finite number of eigenvectors $\left[\phi_n\right]$, and eigenfrequencies ω_n. Ordinarily, the eigensolution is performed on the A-sized matrices. But once the modes ϕ_A are obtained, their G-sized counterpart ϕ_G can be recovered so that the response variable $x_G(t)$ of the dynamic equation can be expanded in a transformation involving a finite number of eigenvectors and in terms of a new set of response variables $\left\{q_n(t)\right\}$; i.e.

$$(3) \qquad \{x_G(t)\} \simeq \sum_{i=1}^{n} \phi_{Gi} q_i(t) = [\phi_G]_n \{q(t)\}_n .$$

Equation (3) transforms only the <u>columns</u> of the coefficient matrices. Substituting equation (3) into equation (1), we get:

$$(4) \qquad \left[M_{GG}\right]\left[\phi_{Gn}\right]\left\{\ddot{q}_n(t)\right\} + \left[B_{GG}\right]\left[\phi_{Gn}\right]\left\{\dot{q}_n(t)\right\} + \left[K_{GG}\right]\left[\phi_{Gn}\right]\left\{q_n(t)\right\} \simeq \left\{p_G(t)\right\} .$$

Premultiplying by the transpose of the eigenvectors to act on the <u>rows</u> of M, B, K, and p results in:

$$(5) \quad \left[\phi_{Gn}\right]^T\left[M_{GG}\right]\left[\phi_{Gn}\right]\left\{\ddot{q}_n(t)\right\} + \left[\phi_{Gn}\right]^T\left[B_{GG}\right]\left[\phi_{Gn}\right]\left\{\dot{q}_n(t)\right\} + \left[\phi_{Gn}\right]^T\left[K_{GG}\right]\left[\phi_{Gn}\right]\left\{q_n(t)\right\} \simeq \left[\phi_{Gn}\right]^T\left\{p_G(t)\right\}.$$

If we define each of the triple product coefficient matrices as follows:

$$(6) \qquad \left[\phi_{Gn}\right]^T\left[M_{GG}\right]\left[\phi_{Gn}\right] \equiv \left[{}^{\backslash}\nu_{n\backslash}\right] \quad \text{generalized mass,}$$

$$(7) \qquad \left[\phi_{Gn}\right]^T\left[B_{GG}\right]\left[\phi_{Gn}\right] \equiv \left[{}^{\backslash}2M\omega\zeta_{n\backslash}\right] \equiv \left[{}^{\backslash}\beta_{n\backslash}\right] \quad \text{generalized damping,}$$

$$(8) \qquad \left[\phi_{Gn}\right]^T\left[K_{GG}\right]\left[\phi_{Gn}\right] \equiv \left[{}^{\backslash}\kappa_{n\backslash}\right] \quad \text{generalized stiffness,}$$

equation (5) can be written

$$(9) \quad \left[{}^{\backslash}\nu_{n\backslash}\right]\left\{\ddot{q}_n(t)\right\} + \left[{}^{\backslash}\beta_{n\backslash}\right]\left\{\dot{q}_n(t)\right\} + \left[{}^{\backslash}\kappa_{n\backslash}\right]\left\{q(t)\right\} \simeq \left[\phi_{Gn}\right]^T\{p_n(t)\}.$$

This transforms the equation of motion into modal coordinates. The effect of transforming to modal coordinates is to decouple the dynamic equations. Or stated in another way; each of the coefficient matrices is diagonalized (with special substitutions and assumptions this can be said also of the damping)[3].

Our concern is not in the integration of the modal transient differential equation but in recovering data after integration from a representation which is made using an incomplete or truncated set of modes. If we re-examine the modal transformation equation (3), but view it as partitioned between elastic and rigid body degrees of freedom, we get:

$$\text{(10)} \qquad \{x_G(t)\} \simeq \left[\phi_{Ge} \vdots \phi_{Gr}\right]\left\{\frac{q_e(t)}{q_r(t)}\right\} = \left[\phi_{Ge}\right]\left\{q_e(t)\right\} + \left[\phi_{Gr}\right]\left\{q_r(t)\right\}.$$

Also we partition equation (9) in terms of elastic and rigid body degrees of freedom.

$$\text{(11)} \qquad \begin{bmatrix} \nu_e & 0 \\ 0 & \nu_r \end{bmatrix}\left\{\begin{matrix} \ddot{q}_e(t) \\ \dot{q}_r(t) \end{matrix}\right\} + \begin{bmatrix} \beta_e & 0 \\ 0 & 0 \end{bmatrix}\left\{\begin{matrix} \dot{q}_e(t) \\ \dot{q}_r(t) \end{matrix}\right\} + \begin{bmatrix} \kappa_e & 0 \\ 0 & 0 \end{bmatrix}\left\{\begin{matrix} q_e(t) \\ q_r(t) \end{matrix}\right\} \simeq \left\{\begin{matrix} \phi_{Ge}^T p_G(t) \\ \phi_{Gr}^T p_G(t) \end{matrix}\right\}$$

At this point the "tricky" manipulations begin. First, isolate the elastic partition of equation (11).

$$\text{(12)} \qquad \left[\nu_e\right]\left\{\ddot{q}_e(t)\right\} + \left[\beta_e\right]\left\{\dot{q}_e(t)\right\} + \left[\kappa_e\right]\left\{q_e\right\} \simeq \left\{\phi_{Ge}^T p_G(t)\right\}$$

Under normal circumstances equation (9), is integrated and q_e, $\dot{q}_e$, and $\ddot{q}_e$ would be known. We could then rearrange the equation to get q_e explicity. Instead of integrating, we extract $\{q_e\}$ explicitly in preparation for making a substitution later:

---

3. Classical Normal Modes in Damping Linear Systems, by T. K. Caughey, J. Appl. Mech. Vol 27 June 1960 pp 269-271.

$$(13) \qquad \{q_e\} \simeq [\kappa_e]^{-1}\{\phi_{Ge}^T p_G(t) - \nu_e \ddot{q}_e(t) - \beta_e \dot{q}_e(t)\}$$

Substituting equation (13) into equation (10) results in an expression of external responses in terms of modal variables:

$$(14) \qquad \{x_G(t)\} \simeq [\phi_{Ge}][\kappa_e]^{-1}\{\phi_{Ge}^T p_G(t) - \nu_e \ddot{q}_e(t) - \beta_e \dot{q}_e(t)\} + [\phi_{Gr}]\{q_r(t)\}$$

Then expand and combine.

$$(15) \qquad \{x_G(t)\} \simeq [\phi_{Ge}][\kappa_e]^{-1}[\phi_{Ge}]^T\{p_G(t)\} - [\phi_{Ge}][\kappa_e]^{-1}[\nu_e]\{\ddot{q}_e(t)\} - [\phi_{Ge}][\kappa_e]^{-1}[\beta_e]\{\dot{q}_e(t)\} + [\phi_{Gr}]\{q_r(t)\} \quad .$$

Here we have terms making up the response referred to geometric coordinates. The enterprise is to find ways to alleviate the effects of truncations to a reduced set of modal variables.

Look at the triple matrix product making up the coefficient of the time dependent loading in the first term, $[\phi_{Ge}][\kappa_e^{-1}][\phi_{Ge}^T]$. This term is the target of improving accuracy (and collaterally resources). To understand its importance in the overall solution we must get a physical hold of its consequences. If we first look at what it is functionally, we notice that its dimensions are displacement per unit force. That, by definition, is a compliance. Its inverse is a stiffness. It will be possible to get a better grasp of this compliance term by considering it from a stiffness standpoint; so the inverse of the triple product in this case is:

$$(16) \qquad [\phi_{Ge}^T]^{-1}[K_e][\phi_{Ge}]^{-1} = [[\phi_{Ge}][K_e]^{-1}[\phi_{Ge}]^T]^{-1} \quad .$$

Equation (16) has generalized stiffness for its middle matrix. Substituting from equation (8) for K_e and collecting terms, the result becomes:

$$\left[\phi_{Ge}^{T}\right]^{-1}\left[[\phi_{Ge}]^{T}[K_{GG}][\phi_{Ge}]\right][\phi_{Ge}]^{-1} =$$

$$= [\diagdown I \diagdown][K_{GG}][\diagdown I \diagdown] = [K_{GG}] \quad .$$

Thus we have shown that the triple product compliance term is the inverse of $[K_{GG}]$, i.e:

(17) $$[\phi_{Ge}][K_e]^{-1}[\phi_{Ge}]^{T} = [K_{GG}]^{-1} \quad .$$

There is a rule in tensor analysis which states that physical quantities remain invariant under coordinate transformations. Then truly the first triple product matrix in equation (15) represents the physical compliance after transpositions of coordinates, so the same physical quantity in another form would be equivalent so long as the resultants for both ended up in the same coordinate system.

We are looking for an expression in the original geometric coordinates for $\{x_G(t)\}$. Certainly the inverse of the original stiffness matrix i.e. K^{-1} would work. But this is an unnecessarily expensive option. Chris came up with a much more practical scheme. In effect we will need only a small number of columns of K^{-1} to satisfy our requirements. His suggestion was to solve the statics problem:

$$(18) \qquad \left[K_{GG}\right]\left[U_{Gi}\right] = \left[P_{Gi}\right] \quad ,$$

where the static loading matrix P_{Gi} consists of i columns of load vectors in which a unit load is put on only one degree of freedom and all other entries in the $i^{\underline{th}}$ vector are zero. The number of vectors "i" that are assembled depends on how many degrees of freedom are loaded. Then the responses U_{Gi} are the amount of displacement resulting from a unit loading for each of the "i" columns. There is another point to be made here; the compliance being sought is limited to the number of columns resulting from the set of individual degrees of freedom of loading that exist instead of a complete G-sized compliance matrix. There is, therefore, merit to the enterprise of engaging in a limited exercise in unit loads to get an economically sized compliance matrix. Looking again at the triple product coefficient it can be seen that its size is G x G. So after U_{Gi} has been solved for, it will have to be inflated with zeroes to G columns to make it commensurate with the triple product. In his paper Chris labeled the inflated U_{Gi} term, ψ; and he labeled the P_{Gi} term, p_1. Adopting his notation we can make this substitution in equation (15) to get:

$$(19) \qquad \left\{x_G(t)\right\} = \left[\psi_{GG}\right]\left\{p_G(t)\right\} - \left[\phi_{Ge}\right]\left[\kappa_e\right]^{-1}\left[\nu_e\right]\left\{\ddot{q}_e(t)\right\} - \\ - \left[\phi_{Ge}\right]\left[\kappa_e\right]^{-1}\left[\beta_e\right]\left\{\dot{q}_e(t)\right\} + \left[\phi_{Gr}\right]\left\{q_r(t)\right\} .$$

This corresponds to equation (5) in Chris' 1988 paper. Thus he has improved the data recovery by incorporating a more complete coefficient of the transient forcing term, p(t). In the finite element sense we are still introducing some error into our solution in solving for ψ by using finite elements instead of experimental data. The point to be made is that the amount of error involved in using modal methods in superelement problems has been considerably reduced by using "modal salvage" methods as opposed to "modal acceleration" methods.

So much for the theory based on G-sized matrices. Our interest starts at the output of the TRD1 module which publishes the response in modal coordinates. It passes these results on to the MODACC subDMAP for transformation back to A-sized physical coordinates. Since these results are usually for a reduced set of modes we can declare this x_A output as defective and determine to apply a correction here. This is a different philosophy than Chris used. He reconstructed every term of the refined response, at the G-level. We propose only to replace the defective term and retain all other parts of the data recovery.

The A-sized equation corresponding to equation (3) is

$$\text{(3A)} \quad \left\{x_A(t)\right\} \simeq \sum_{i=1}^{n} \phi_{Ai} q_i(t) = \left[\phi_A\right]_n \left\{q(t)\right\}_n .$$

The A-sized expression corresponding to equation (15) is:

$$\text{(15A)} \quad \left\{x_A(t)\right\} \simeq \left[\phi_{Ae}\right]\left[\kappa_e\right]^{-1}\left[\phi_{Ae}\right]^T\left\{p_A(t)\right\} - \left[\phi_{Ae}\right]\left[\kappa_e\right]^{-1}\left[\nu_e\right]\left\{\ddot{q}_e(t)\right\} - - \left[\phi_{Ae}\right]\left[\kappa_e\right]^{-1}\left[2\nu_e\zeta_e\omega_e\right]\left\{\dot{q}_e(t)\right\} + \left[\phi_{Ae}\right]\left\{q_r(t)\right\}.$$

Are we justified in applying a correction on the A-sized matrices, instead of trying to reconstruct the recovery at the G-level? We think so, because at this point we are free of modal coordinates and are back in physical coordinates. There is no loss of accuracy for elastic condensation, so substitution at the A-level accomplishes the needed improvement.

Now turn to the implementation of this "modal truncation salvage" method in Version 66 of MSC/NASTRAN. Chris has already provided some ideas that will be helpful. Note that the orientation of this implementation is in DMAP until more schemes are tested before making any recommendation toward setting up permanent modules and solution sequences. As will be shown later it does appear to be a good candidate for permanent status. For purposes of appreciating what operations are pursued, start from an orientation of a pure modal transient problem.

$$(20) \qquad [\nu]\{\ddot{q}\} + [\beta]\{\dot{q}\} + [\kappa]\{q\} \simeq \{\pi(t)\} \quad .$$

If we were to run modal transients with model data for elastic, mass, damping, load, and integration parameters, we would get an intermediate time history in modal coordinates q(t) which MSC/NASTRAN calls UHVT. One step further transposes the responses from modal to physical coordinates called UDV. Here is where we want to intervene.

IMPLEMENTATION

Our strategy is to allow the modal transient solution to continue without interruption through the recovery of responses from modal coordinates to physical coordinates in triplet format {disp, velo, acce}. Due to modal truncation, we recognize these results to be defective and we call it UBAD. In effect, we have recovered to the state of equation (15A) (with the addition of velocity and acceleration). Our plan is to reconstruct the first term of equation (15A) and subtract it from the computed value of UBAD. Chris's approach is to apply an improvement to only the displacement response. This necessitates partitioning time varying dispalcements from the velocities and accelerations in the triplet of responses of UBAD. Name the defective displacements UDEF and the acceptable velocities and accelerations

URETAIN. Call the coefficient $\left[\phi_{Ae}\right]\left[\kappa^{-1}\right]\left[\phi_{Ae}^{T}\right]$ the defective compliance Z_{def}. Construct its replacement ψ and call it the compliance correction Z_{corr}. Finally we need to obtain the time varying amplifier p(t). Once all this is in hand we generate an improved response UIMP:

(21) $\{UIMP\} = \{UDEF\} + [Z_{corr} - Z_{def}]\{p(t)\}.$

This improved displacement vector will be merged with URETAIN and passed to the modal transient solution, SSS 112, to continue its data recovery in either a superelement or in a non superelement mode. With this as the general strategy, each specific implementation task will be described separately to help the reader to relate the solution steps to the mathematics. As a final step, all operations will be integrated into a sequence for computing. Demonstration problems, kept simple so as not to intrude on the task of grasping the concepts, have been solved in five ways to contrast the merits of its implementation. They are: modal without correction, modal with mode acceleration, modal with salvage, superelement with mode acceleration, and superelement with salvage.

Task 1. Run modal transients and interrupt the solution after capturing the responses which were transformed to physical coordinates. If there are rigid body modes, be sure to add SUPORT degrees of freedoms to the model so that the static stresses will correlate with the dynamic stresses. Run SSS 112. The final steps before intercept are:

Main SubDMAP SEMTRAN calls subDMAP MTRANS

SubDMAP MTRANS brings in module TRD1, which solves for the triplet of modal responses

$[\{q(t)\},\{\dot{q}(t)\},\{\ddot{q}(t)\}] = [UHVT]$.

Intermediate steps set some parameter values.

Main SubDMAP SEMTRAN calls subDMAP MODACC.

SubDMAP MODACC operates module MODACC which reduces UHVT from integration to output time steps.

SubDMAP MODACC uses module MYPAD to transform from modal to physical coordinates A-size.

$[\{x_A(t)\},\{\dot{x}_A(t)\},\{\ddot{x}_A(t)\}] = [UDV]$.

UDV is the defective response.

Impose a DMAP ALTER in MODACC to partition displacements from velocity and accelerations in UDV.

Partition UDV, and store both sub partitions on the database. The displacement partiton will be labeled UDEF to mark it as defective. The other will be labeled URETAIN for remerging with the improvement.

Task 2. Reconstruct the defective compliance. Run SSS 112. The series of steps before imposing the ALTER are:

Main SubDMAP SEMTRAN calls subdmap MODERS

SubDMAP MODERS brings in the appropriate module to compute eigenvalues, eigenvectors and modal mass.

Impose a DMAP ALTER in MODERS to determine the number of rigid body modes. Separate the rigid body modes the from elastic modes.

Recover Generalized stiffness by converting column 1 of table LAMA into a column matrix of eigenvalues and by solving for the elastic set of generalized stiffnesses, K = LAMBDA X MI.

Solve for the defective compliance by inverting the generalized stiffness; pre-multiplying by the matrix of eigenvectors; then post-multiplying by the transpose of the eigenvectors. This data block ZDEF is stored in the database.

Task 3. Generate Corrected Compliance. Run SSS 101. (STATICS) We will follow Chris's approach and set up a statics problem with unit static loads at every degree of freedom in every individaul loading case corresponding to dynamics. In other words the components to be given static loads match the dof's on which dynamic loads have already been assembled. One load vector (subcase) will be set up for each loaded degree of freedom. We want to be sure to include SUPORT points in the static model as well as the dynamic model if there are rigid body modes in the dynamic behavior. Compliance is different with free boundaries as compared to a model with inertia relief in the static analysis. The freedoms SUPORTed in the static model should correspond to those SUPORTed in the dynamic model for obtaining proper reference in computing stresses. We will retrieve the A-sized response matrix which will be our corrected compliance, after we inflate the columns to A-size.

Final steps before intercept for recovery of response follows:

Main SubDMAP SESTATIC calls subDMAP STATRS

SubDMAP STATRS operates module SSG3

Insert an ALTER in subDMAP STATRS to store the displacement matrix on the database, and exit SSS101 without performing data recovery.

Task 4. Complete the Improved Response. The equation of the assembly is UIMP = UDEF + (ZCE - ZDE)PPT. Merge UIMP with URETAIN. Run SSS 112. The final steps at intercept are the same as those we previously encountered to get UDV, which is as follows:

Main SubDMAP SEMTRAN calls subDMAP SUPER3
SubDMAP SUPER3 calls subDMAP MODACC
Insert an ALTER in subDMAP MODACC to generate the corrected response, taking advantage of the dynamic load vector PPT which has already been generated in module TRLG and is resident on the data base.
Merge the improved displacement, UIMP, with the uncorcorrected velocity and acceleration, URETAIN, to form the improved UDV.

The improved UDV takes the place of that originally output from MPYAD, and MODACC continues processing undisturbed from this point on. SSS 112 then obeys the selections made in case control.

There would be but two executions in actual practice. The first would be to run SSS 101 to install the corrected compliance in the data base. All other tasks are performed in SSS 112 so that all of the ALTER's can be combined into one packet consisting of several steps.

Author Butler has been campaigning for twleve years to enhance the DMAP capability so as to capture scratch data blocks to avert the necessity of having to re-invent the wheel so-to-speak for regenerating temporarily existing things. The authors wish to thank the writers of the new DMAP code for bringing out so many data blocks, that used to buried as scratch, to become accessible now as output data blocks. This has made the writing of the ALTER packets for this paper ever so much more pleasant.

Due to the publishing time constraint, details of the ALTER packets for production runs will be presented at the World User's Conference.

DEMONSTRATION PROBLEMS

To test the implementation of the Modal Salvage Method the results of the five approaches outlined on page 10 for two models will be presented. Figures 1 and 2 depict a model of a simple cantilever beam. Figures 3 and 4 depict a model of a more complex structure. Details of the results will be presented and made available at the Word User's Conference.

CONCLUSIONS

Chis Flanigan's work has provided a basis for an alternate mode acceleration method of data recovery. This paper has given a firm theoretical footing for the techniques that he used. The successful incorporation of this method entirely within MSC/NASTRAN Version 66 has been accomplished using the new DMAP language.

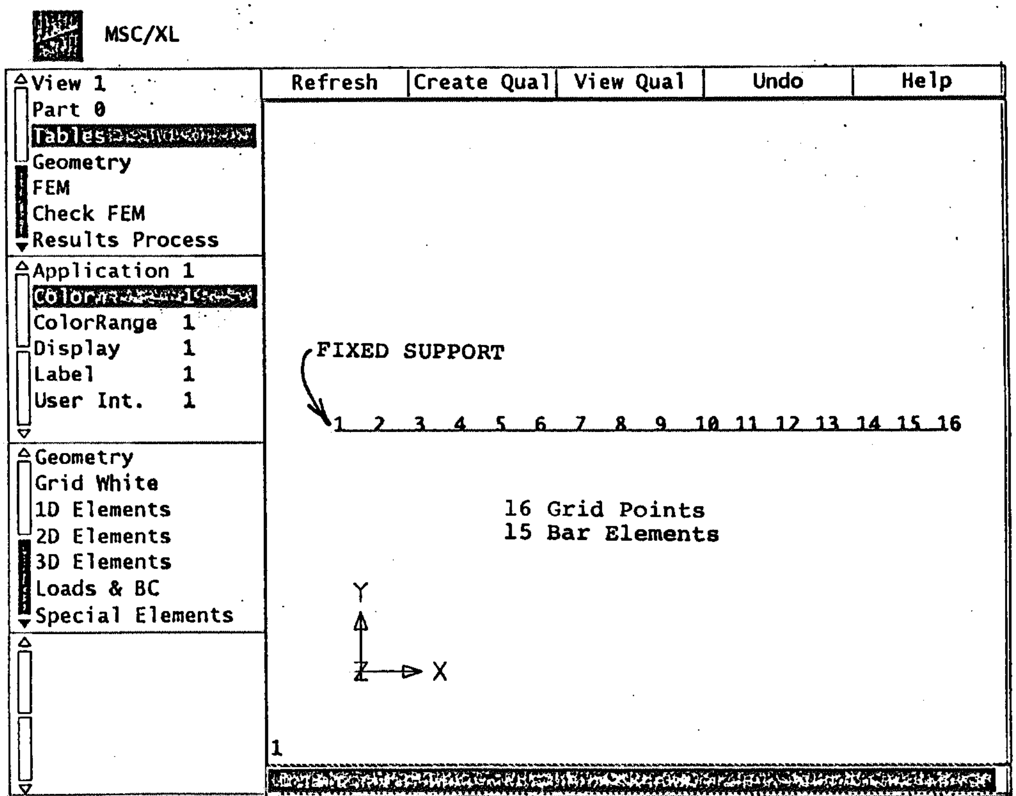

Figure 1 - Simple Model - Basic Configuration

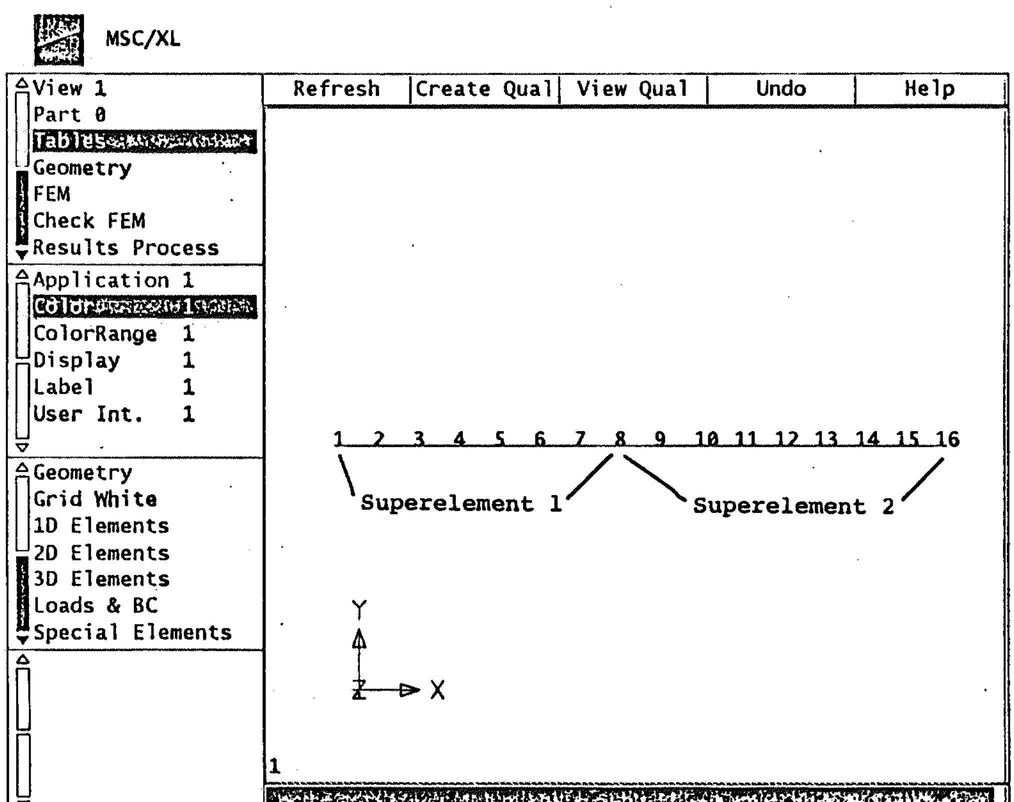

Figure 2 - Simple Model - Superelements

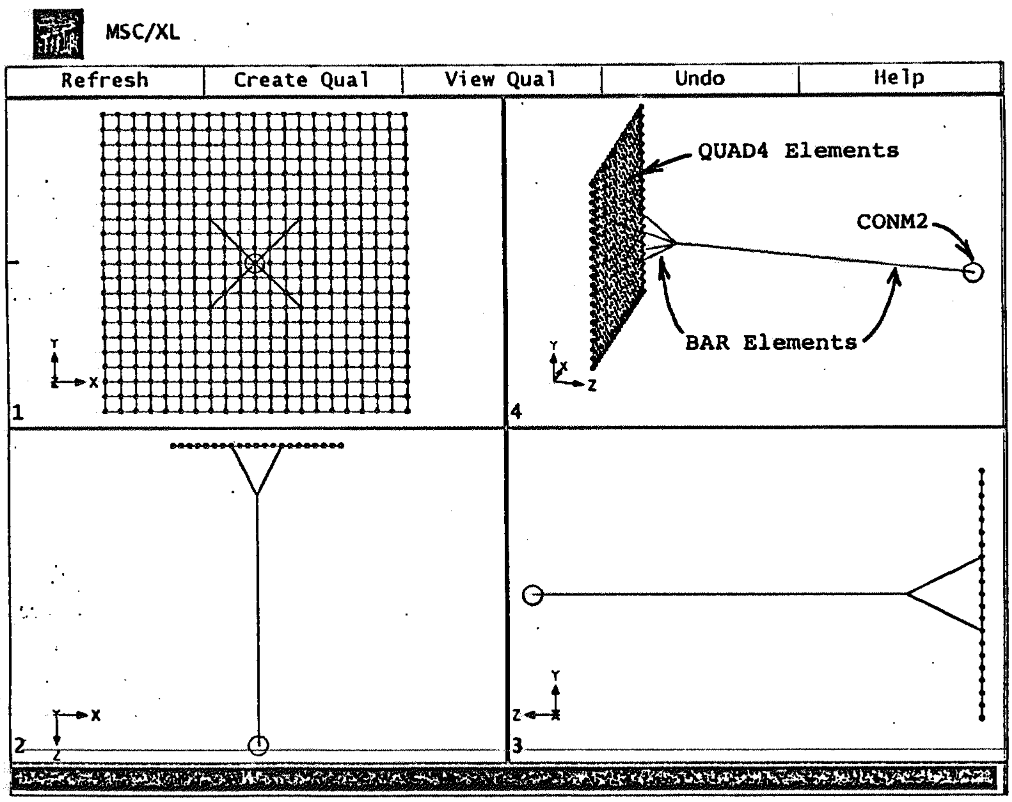

Figure 3 - Complex Model - Basic Configuration

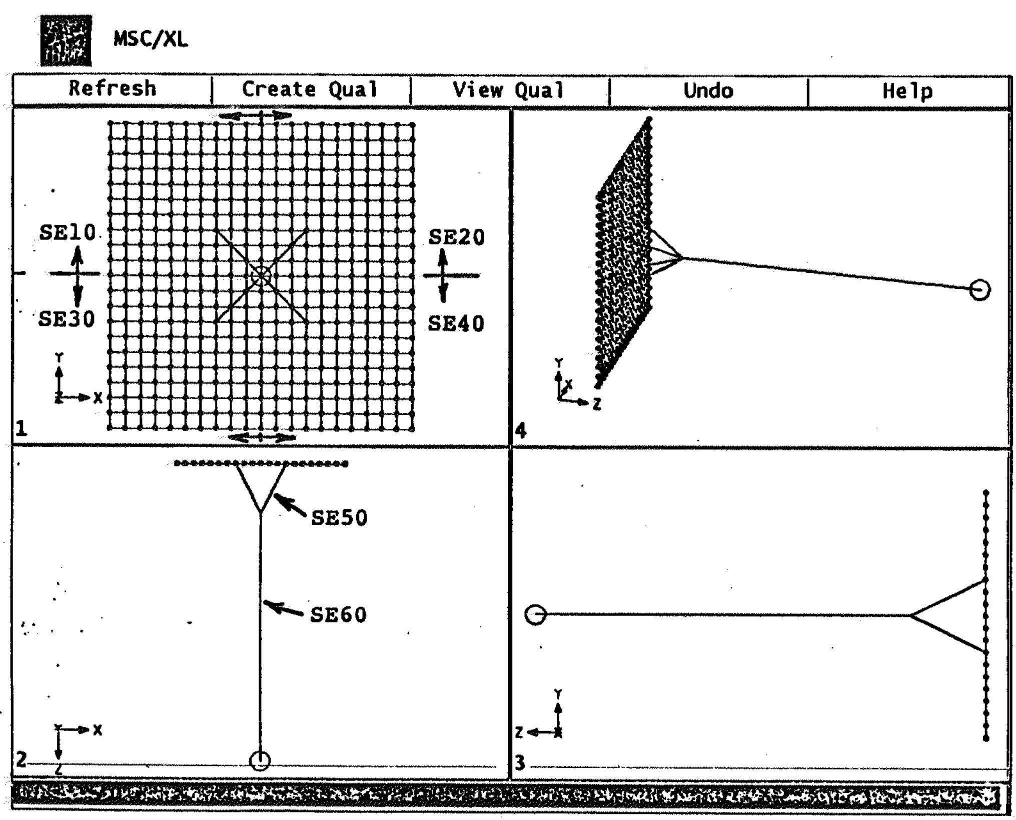

Figure 4 - Complex Model - Superelements