CONDENSING LOADED POINTS FOR 'TRANSIENTS BY SUBSTRUCTURING'

THOMAS G. BUTLER

BUTLER ANALYSES

SUMMARY

This paper is an extension to my article presented at the 7th NASTRAN Colloquium entitled "TRANSIENTS BY SUBSTRUCTURING". In order to capitalize on the economies of (a) condensing of omitted points during substructuring, and (b) of applying BANDIT at the only place where it can enter the analysis, a strategy needed to be worked out so that dynamically loaded points could also be condensed in Phase 1. This would obviate having to retain points during dynamic analysis for no other reason than to make them available for assigning load, and having to suffer the consequences of expensively large matrices. This paper describes a technique, vintage 17.5, for accomplishing these aims. A number of problems arose such as transferring of condensed loading information from R. F. 1 to R. F. 9, and giving this loading a correct time history. The method has been applied to substructure transient solutions according to the approach as outlined at the 7th NASTRAN Colloquium. Only DMAP statements and some manual transfer of data were used to implement this strategy.

PURPOSE

The cost of solving large order transient problems can be high. If the matrix order is reduced by condensation without inflating the band, the cost of solving transients can be decreased. The cost of condensing in itself can be expensive for large matrices, so it behooves one to condense at the component level whenever the matrices are small and the cost is exponentially less. If this condensing of dynamically loaded degrees of freedom could be worked out, then the rder and band of the matrices in the transient solution could also come under the control of the analyst to conform to his needs without operational constraints. The avenue is open in substructuring to do the condensing of each low order component matrix for small cost with a net overall saving. It appears that the ideal is at hand to cut the cost of transients without incurring high processing costs, but the way is not straightforward. The plan should not be coded without considering the needs of banding. One is compelled to condense at the compoonent level when possible, because BANDIT can be invoked in substructuring only at the component level. The purpose of this enterprise is to find a way of condensing dynamic loads in stubstructuring economically, so that the load matrices can be correctly formed when R. F. 9 is entered.

PROBLEMS

How can the dynamic loads be represented in Phase 1? How can the spatial, time amplification, and time delays be coordinated, if the points scheduled for condensation have loading times that differ from those of retained points? How can loading on the retained points be merged with that resulting from condensing operations on the omitted points? Can all these objectives be met without burdensome labor or cost? The answers to these questions will be taken up in the section entitled IMPLEMENTATION. The reason for having to pose the first question is that Phase 1 of fully automated substructuring is operational for statics and eigenvalues only and not dynamics. Consequently, some property of the dynamic load has to be isolated which behaves in a way that can be treated in statics. The answer to the second question seems to be even more elusive than the first. It appears like an attempt to attach a time variation to something which has disappeared from the problem. The merging difficulty posed in the third question seems to be one of retaining some sort of identity of the condensed loads so that the relationships to the loads on uncondensed points can be maintained. The bulk of this report is spent on how these questins were answered.

IMPLEMENTATION

Consider the composition of the dynamic load in its simplest form

$$P(t,x,y,z) = A(x,y,z) \times T(t-p)$$

The component, A(x,y,z), is the spatial specification of the loading. The data for this part is entered via DAREA cards. In that the information contained on DAREA involves position and magnitude, it is similar to stati loads. If one were to set up the TLOAD1 cards for the dynamic loads without regard for any intention to condense loaded points, it would be easy to identify those spatial sets (i.e. DAREA sets) that are loaded synchronously; i.e. those having a common time amplification defined by a single TABLED1 set and a single DELAY set. At completion of defining the total dynamic load there will be as many synchronous sets as there are TLOAD1 sets. There are, at most, the same number of DAREA sets as there are TLOAD1 sets--in general there are fewer DAREA sets. It is now a matter of comparing the omitted degrees of freedom with the DAREA sets to find their intersection. All those DAREA sets having omitted degrees of freedom are isolated. For purposes of illustration, concentrate on just one substructure and assume there are three such DAREA sets in that one substructure and designate them as Q,R & S. Use lower case letters q,r & s for the sets of omitted degrees of freedom in their respective parents. Delete q,r & s from Q,R & S and label these DAREA sets with their condensed d.o.f.'s removed: Q',R' & S'. The case in which all degrees of freedom in one of the DAREA sets are omitted is admissable; for instance q could be identical to Q and Q' could be null. With this notation problem behind us it will ease our discussion of the strategy. These individually synchronous sets (q,r & s) that are both loaded and omitted, will be operated on during substructuring. In effect q,r & s are going to be condensed as static loads in Phase I then delivered back into the dynamic loads as pre-condensed DAREA sets. The logic follows from the fact that spatial condensation in statics operates exactly the same way that spatial condensation works in dynamics. Time varying amplification won't operate any differently on a condensed set that comes pre-condensed from statics and is not further condensed in dynamics than it would on one that comes earmarked to become condensed during transient execution. Consequently, when the loads q,r & s are condensed in substructuring, the resulting redistribution of loads to retained points will be labeled q',r' & s'. The next step is to organize data from q',r' & s' in DAREA format and give them the same set I.D.'s respectively as the parent Q',R' & S' sets. Belonging to the same DAREA set, constitutes a merge of sets in Case Control management, so the q,r, & s that were deleted from Q,R & S will have been replaced with their pre-condensed counterparts q',r' & s'. No changes are made to the dynamic loading data involving TLOAD1, DLOAD, and TABLED1 entries. New DELAY cards will be organized. The sets of synchronously omitted degrees of freedom q,r & s will have to be replaced by the corresponding sets of retained degrees of freedom q',r' & s' without any change to the delay times. When more than one substructure has condensed loading, the same procedure as outlined above should be followed for each component.

So much for talk. It remains to be seen whether this idea has substance or is just a brave show. The DAREA entries of loading coefficients that would have been assigned to the omitted d.o.f.'s of a substructure in a solo transient run are assigned instead to those same omitted d.o.f.'s, as static force loads in a Phase I run of that substructure. Each synchronous set is organized as a static load set and each is scheduled in a succession of subcases. In terms of the symbols of the previous paragraph, the DAREA entries on the set q wil be arranged in a static load set for the first subcase, those on r for the second subcase, and those on s for the third subcase. Of the three solution routes available in automated substructuring, the choice converges on R. F. 2 (Inertia Relief), because it offers options for stiffness, mass, and load matrix generation, while R. F. 1 omits the mass and R. F. 3 omits the static load. If one scans the DMAP sequences for R. F. 2 and compares it with the automated ALTER statements for SUBSTRUCTURE one sees that the load vector is condensed down to retained degrees of freedom in the module SSG2. It is output as the data block PL. NASTRAN needs to be told that all 3 matrices are wanted. In the SUBSCC this is indicated by selecting OPTIONS = K,M,P. Nothing more is essential during Phase I except to emphasize that BANDIT should be enlisted prior to Phase I to resequence all points except mating points at the interfaces between substructures. The user should manually assign the sequence numbers to interface mating points. Even though all the Phase I requirements are met, we are not going to leave it with such an uncaring attitude. Something extra is going to be worked up to give a better "feel" as to how things went during Phase I. We are going to "gin up" an ALTER packet to see what the condensed DAREA components look like before being content with the substructure component runs. The ALTER packet will use the module SOFI to pick off PL, as it is delivered to the SOF, and then will inflate PL with zeroes in a succession of merges until it is G-sized. Then this matrix of "load vectors" is delivered to SDR2 for processing the OLOAD requests in Case Control and reformating according to external grid sequencing. Now the OFP handles the printing chores. Particulars of the DMAP ALTER packet are given in Table I. The above outlined procedure is carried out for every substructure which contains omitted loaded points.

If further condensing is decided on in Phase II, using the REDUCE command, the bookkeeping of the several substructures is kept straight inside the SOF data base. When the Phase II operations have prepared the final pseudo structure, the SOF will contain all of the information that is needed to proceed into transients, provided that OPTIONS = K,M,P has been used for every Phase II run. The stiffness and mass matrices are in final form and the PVEC item of the SOF contains a vector for each synchronous subcase of coefficients for all component substructures in fully condensed form. Now an ALTER uses the module SOFI to take the stiffness, mass, and load data from the SOF and deliver them externally. Use OUTPUTT1 for K & M for subsequent insertion into R. F. 9. Use OUTPUT2 for P if a FORTRAN preprocessor is to be used. If DAREA input is to be prepared manually then the PVERC matrix need only be printed. MATPRN will be sufficient since the DAREA data will be written by internal scalar index number instead of external grid point numbers. Details of the Phase II ALTER are presented in Table II. The highlights of the strategies for the Phase I and Phase II solutions are depicted in Tables V & VI.

EXAMPLES AND CHECK CALCULATIONS

A small two component substructure problem was used to illustrate the method of substructure condensation of dynamic load prior to transient execution. Component BOX is a parallelepiped of BAR elements with BAR diagonals on the rear end to make a connection in the middle. Component END is a two bar appendage. The combined pseudo structure is called ALL. Figure 1 shows P/S ALL with grid points numbered and loading indicated. Grid points 2 and 4 are loaded simultaneously with load F(t) and GP's 6 & 8 are loaded simultaneously with load L(t). GP's 1 & 3 are loaded simultaneously with load F(t) delayed an amount T. GP's 5 & 7 are loaded simultaneously with load L(t) delayed by an amount T. Figure 2 has plots of loadings L(t) and F(t). The direction for both dynamic loads F(t) and L(t) is in component direction 1 as shown by the double shafted arrow in firgure 1. If DAREA and DLOAD coefficients are held at unity then the values from figure 2 will become the TABLED1 entries. This pilot problem was deliberately kept small to test only the new features and not redo all features of transients by substructuring. D.o.f. no. 1 was omitted from point 6 of load L(t) and from point 3 of load F(t) and is indicated with tiny circles in figure 1.

In the spirit of keeping things simple, only displacement responses in the uncondensed component END will be examined as indicated by open brackets in figure 1. Since displacements can be recovered as part of the R.F. 9 output, there will be no need for scheduling Phases 3 & 4. In order to test the method, a separate transient analysis entirely within R. F. 9 was run. The problem flow is diagrammed starting with a Substructure Tree in figure 3 and flow charts of transients first with a substructure preface in figure 4 and then in a stand-alone mode in figure 5.

A measure of the chore of converting the PVEC from substructuring into DAREA data for transients can be obtained by looking at the effects of condensing. Figure 6 shows the DAREA values for the synchronous set of grid points 6 and 8 as it was prepared for the solo transient solution. Figure 7 is a sketch of the points to which the omitted loads of GP 6 and GP 3 were vectored by the SMP module in Phase I. Figure 8 shows the results of manually prepariing the DAREA data of the condensed loading for the synchronous set of grid points 6 and 8 which are now organized according to interior scalar points. It is easy to imagine how burdensome this manual preparation can become for a large structure. A processor program is an obvious necessity. Only one digit of data was used to illustrate the relative magnitudes of the redistribution. The major contributions are circumscribed. Translations are enclosed in boxes and rotations are enclosed in circles. The translation load of GP 6 is essentially reduced to a translation at GP 5 (Scalar dof 7) and rotations about GP 8 (Scalar dof 18) and GP 2 (Scalar dof 35). The box about scalar dof 13 identifies the uncondensed loading of GP 8.

RESULTS

In order to test this method of condensing dynamic loads bv substructuring, a separate analysis was performed in the conventional way by submitting the entire structure with loads defined externally and all other features retained, using R.F. 9 in a solo run. The quality of the substructure analysis was based on making comparisons of only displacements at the tip of the appendage where the amplifications would tend to be the greatest. Comparisons of stresses and forces in elements were omitted for reasons of brevity. Displacement comparisons could be made directly from the two transient outputs, but stresses and forces would have required two extra steps plus additional work in Phase II. Results of the two displacement time histories are assembled side by side for each translational and rotational component in Tables III & IV. Where the displacements are large, the outputs match in 6 digits and start to vary in the seventh place. That is, the axial and vertical translations and the rotations about the transverse axis compare almost exactly. But the small displacements don't even match up in the first digit. In theory all of these extremely small displacements should have been zero, because the load was symmetrical; however the condensation was unsymmetrical. This was a very small model that would be highly sensitive to irregularities, whereas in a large structure much smoothing would take place. This unsymmetrical condensation caused numerical noise to creep into both the solo and the substructure solutions, giving non-zero values 8 orders of magnitude less than the signifigant displacements. Noise is highly dependent on the sequence of operations and the sequences in the two cases under scrutiny were different, so the noise from the two can be expected to be different. The net opinion is that the results compare favorably.

On the basis of good correspondence in the responses from the two approaches, we can say that the technique of spatial condensation of DAREA coefficients by simulating them as static forces is satisfactory. The method of condensing dynamic loads by substructuring has been established.

RECOMMENDATIONS

When transients are automated for substructuring, it appears that the elements contained in this non-automated approach could be used as a basis for the algorithm. The Phase I bulk data could include a DAREA card, called say DAREAS, which would respond to the user's input of condensed loading sets. A caption in the LODS item could append a D to every load subcase that originated from DAREAS input. At transient execution time the user would indicate which sets of DAREAS data would carry the same set I.D. as uncondensed transient DAREA data, and which sets of DAREAS data would be distinct. Similarly, the Phase I bulk data could include a DELAY card, called say DELAYS, which would respond to the user's input of condensed loading. Then the usual preparation of the elements of dynamic loads would proceed as is now done in transients. If further condensation is desired in transients when preceded by substructuring, then caution would have to be exercised in data recovery. Instead of passing the UDVT matrix to Phase IV, the data recovery would have to reconstruct the response histories to the full order of the original Scalar Point compliment, before passing the displacements to Phase IV. These displacements are contained in data block UPV. Modifications to the DMAP ALTER's for R.F.9 as outlined at the 7th Colloquium would be needed to allow for an option to output either UDVT or UPV based on the condition of whether OMIT's are present. The subsequent partitioning would have optional input of either UDVT or UPV.

In effect, the user could be in full command of the condensation of his problem in substructuring by using this proposed technique. He could condense in Phase I where it is most economical and where he could take maximum advantage of BANDIT. He could condense again in Phase II where interface points could be removed. Lastly, he could condense in transients where the final trade-offs are reviewed between the transient solution costs and additional condensation and data recovery costs.

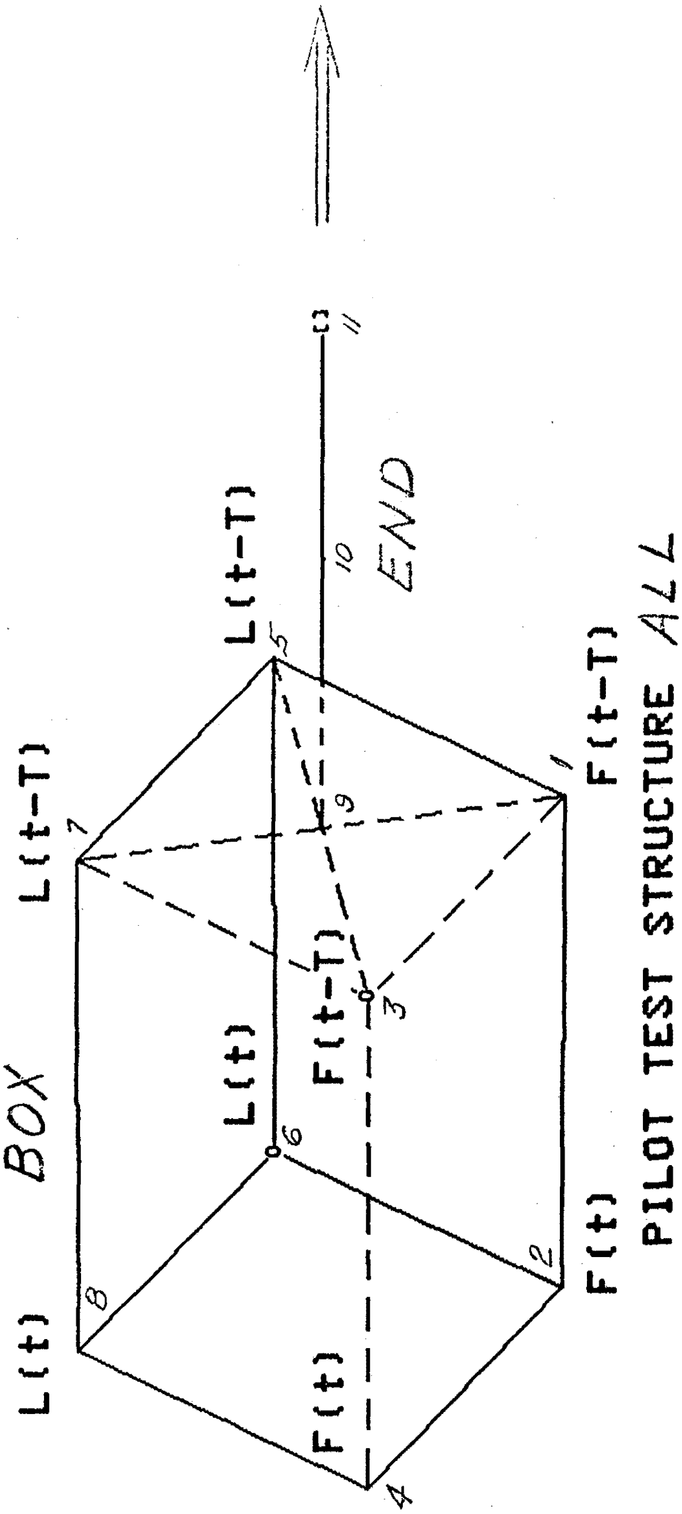

FIGURE 1

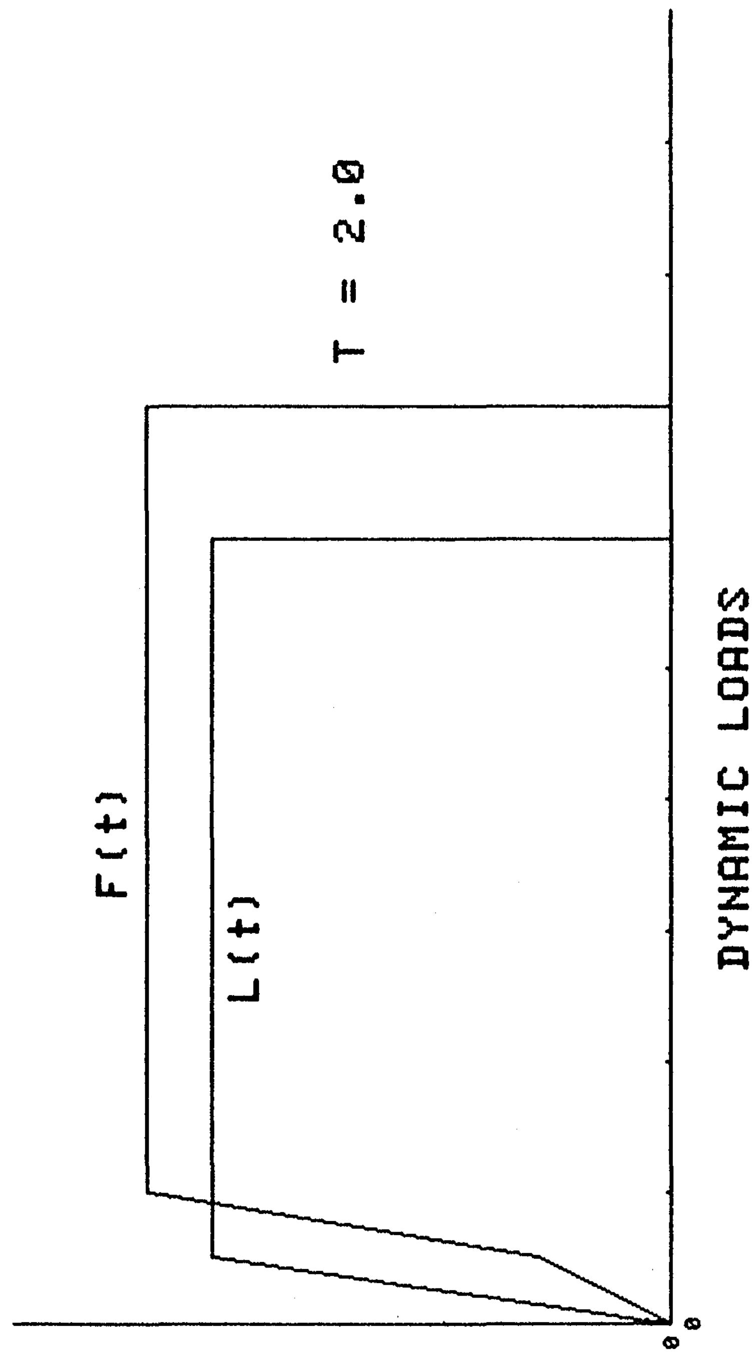

FIGURE 2

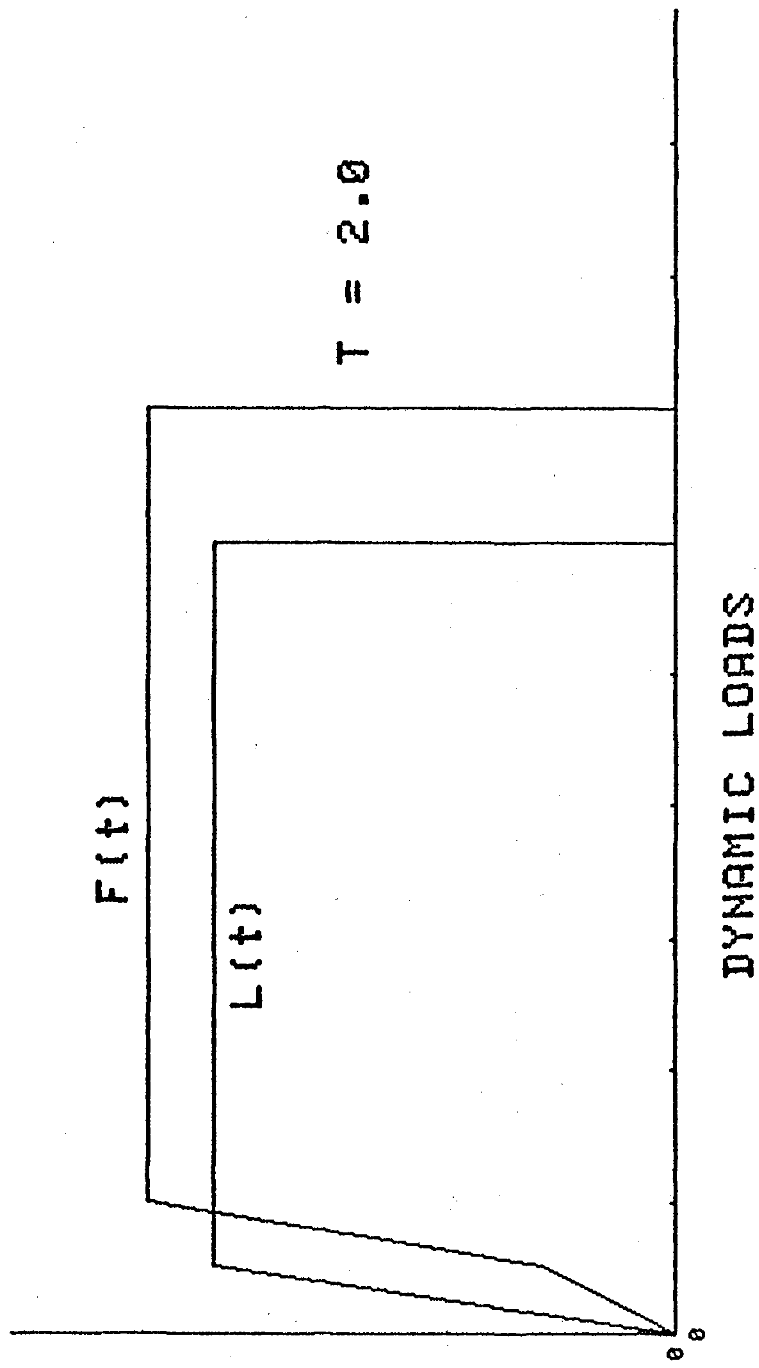

FIGURE 2

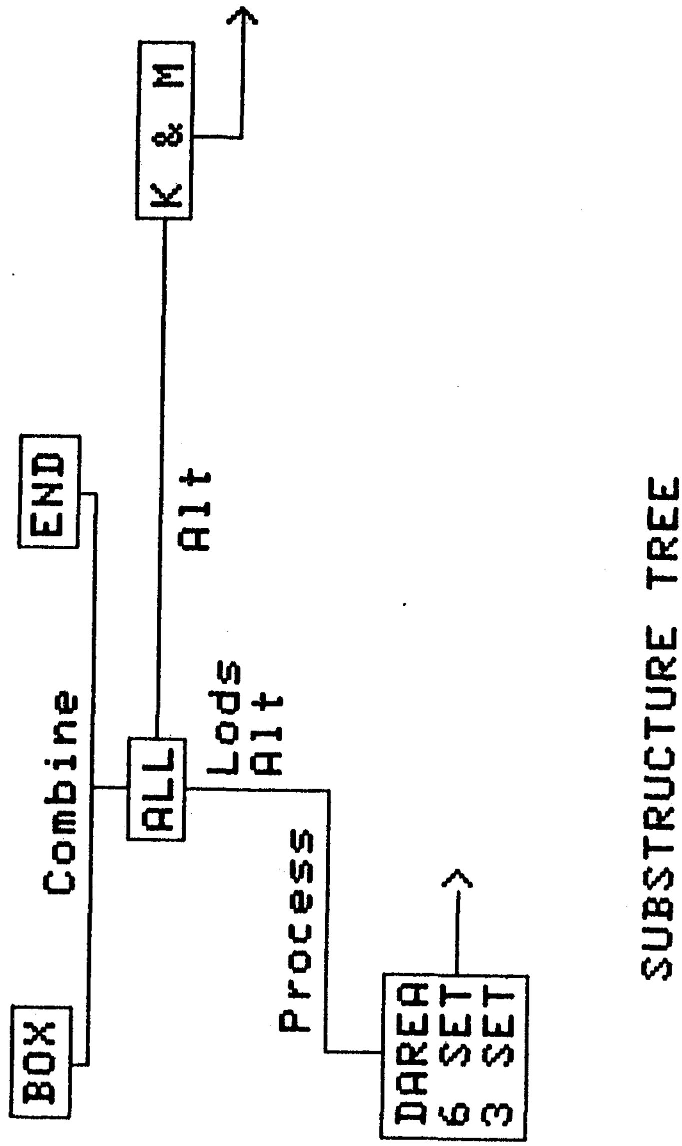
BOX
END
Combine
ALL
Alt
K & M
Lods
Alt
Process
DAREA
6 SET
3 SET
SUBSTRUCTURE TREE

FIGURE 3

S/S - R.F. 9

SCALARS 1 THRU 54

DAREA 2,4 LOAD
5,7 LOAD
COND 6-SET & 8 LOAD
COND 3-SET & 1 LOAD

DLOAD, TLOAD1, TABLED1 (UNMOD)

DELAY 5,7 - SET (UNMOD)
COND 3-SET & 1 TAU

ALT INPUTT1 KMTX, MMTX
ADD K4GG

OUTPUT DISP - 11
OLOAD - 9

FIGURE 4

SOLO

COMPLETE TRANSIENT

BULK OF ALL (BOX & END)

LOAD EXTERNAL BY GP & COMP

OMIT

OUTPUT DISP - 11

OLOAD - 9

FIGURE 5

DAREA

BEFORE OMIT

EXTERNAL SEQ.

G.P. 6

SID	GP	COMP	VALUE
68	6	1	1.0

G.P. 8

68	8	1	1.0

FIGURE 6

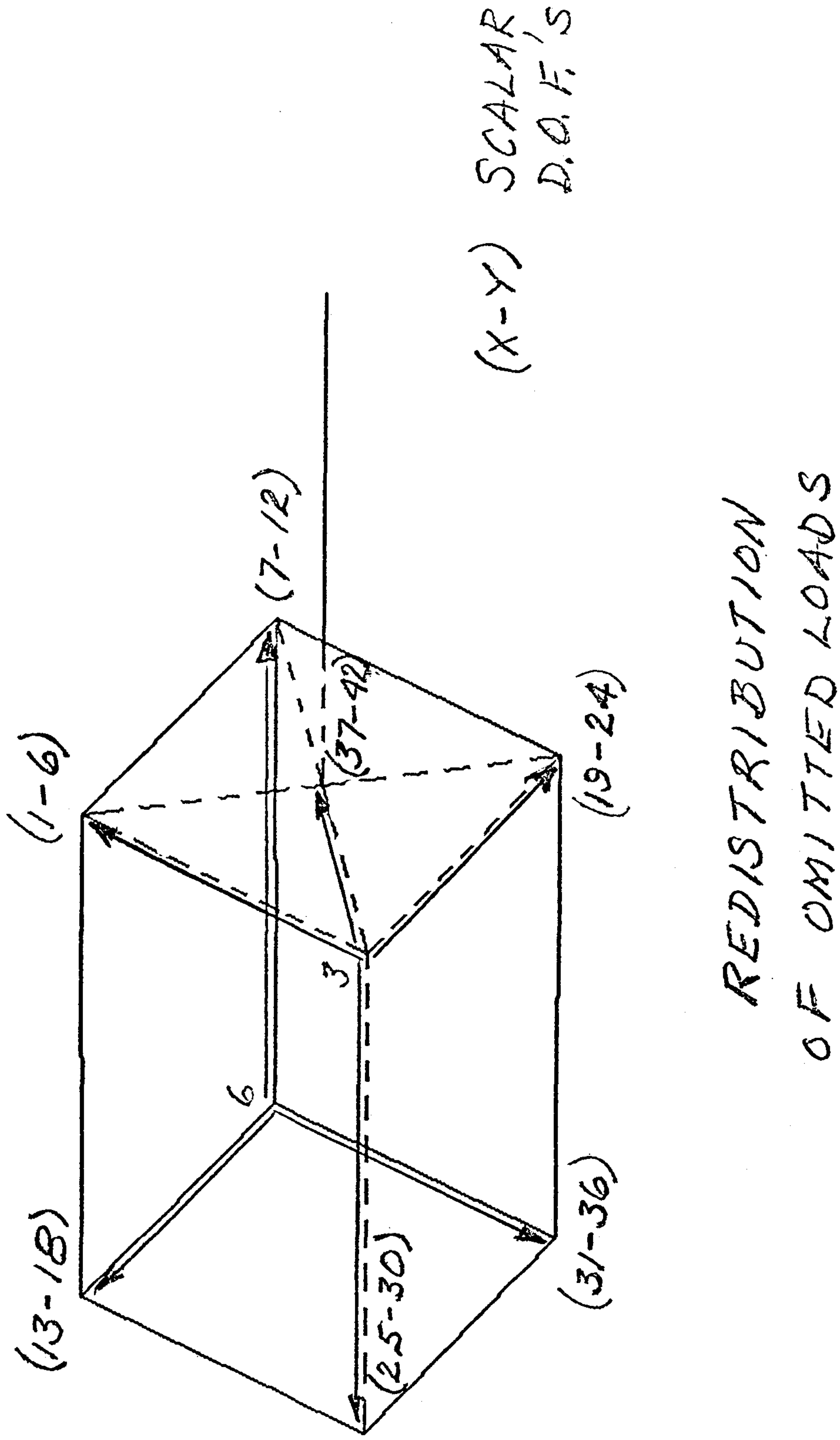
(1-6)
(7-12)
(13-18)
(19-24)
(25-30)
(31-36)
(37-42)
3
6
(X-Y) SCALAR D.O.F.'S
REDISTRIBUTION OF OMITTED LOADS

FIGURE 7

DAREA

AFTER OMIT

INTERNAL SEQ.

SID	GP	COMP	VALUE	GP	COMP	VALUE
68	7	0	.8	8	0	.02
68	9	0	-.02	10	0	.0
68	11	0	-.1	12	0	-.1
68	13	0	1.1	14	0	-.02
68	15	0	.002	16	0	-.01
68	17	0	.06	18	0	.7
68	31	0	.1	32	0	-.002
68	33	0	.02	34	0	.01
68	35	0	.7	36	0	.06

TRANSLATION   ROTATION

FIGURE 8

PHASE I ALTER
PACKET

```
ALTER    136     Just before END$

SOFI     /Pn,,,,/C,N,1/C,N,BOX/C,N,PVEC $
         Selects PL from SOF
         Set n > largest S/S step #
UMERGE   USET,Pn,    /PLTOA/C,N,A/C,N,L/C,N,R  $
UMERGE   USET,PLTOA, /PATOF/C,N,F/C,N,A/C,N,O  $
UMERGE   USET,PATOF, /PFTON/C,N,N/C,N,F/C,N,S  $
UMERGE   USET,PFTON, /PGG  /C,N,G/C,N,N/C,N,M  $
         Inflates PL to G-size with zeroes

CHKPNT   PGG $
SDR2     CASECC,CSTM,MPT,,EQEXIN,SIL,,,BGPDT,,,,,,PGG/
         OPG1,,,,,/C,N,STATICS/C,N,NOSORT2 $
         Formats condensed load to GP's
         according to Case Control
SAVE     NOSORT2
OFP      OPG1,,,,,//V,N,CARDNO $
         Prepares output for Host computer
SAVE     CARDNO $
ENDALTER
```

TABLE I

PHASE II ALTER

PACKET

```
ALTER     136  Just before END $
SOFI           /KN,MN,PN,,/C,N,1/C,N,NAME/C,N,KMTX/C,N,MMTX/C,N,PVEC $
```

The Phase II load vector PVEC preserves the subcase ordering for all of Phase I DAREA cases. Set n > max S/S step#

```
OUTPUT1    KN,MN,,,//C,N,-1/C,N,INP#/C,N,(CONDKM) $
```

K & M matrices of the final solution structure are output to disc for transfer to R.F. 9.

```
MATPRN     PN,,,,// $
```

Listing for man'l prep of DAREA input.

```
OUTPUT2    PN,,,,//C,N,-1/C,N,11/C,N,(LODOMIT) $
OUTPUT2,   ,,,,//C,N,-9/C,N,11 $
```

Matrix of condensed DAREA components is sent to FORTRAN logical disc file TAPE1 for prep of DAREA input by preprocessor.

```
ENDALTER
```

TABLE II

COMPARISONS OF DISPLACEMENT
RESPONSES AT END TIP (G.P.11)
BETWEEN COMPLETE R.F.9 & S/S R.F.9
AT EVERY SECOND FOR 7 SECONDS

T1

TRC	S/S
-4.000940+2	-4.000941+2
-7.335053+2	-7.335056+2
6.445970+2	6.445966+2
5.479066+3	5.479067+3
1.451452+4	1.451452+4
2.738419+4	2.738421+4
4.287670+4	4.287672+4

T2

TRC	S/S
-3.4-5	-7.8-5
-2.2-4	-1.6-4
-5.4-4	-1.1-4
-7.3-4	+2.6-4
-9.9-6	+1.0-3
+3.2-4	+2.1-3
+9.2-5	+1.5-3

T3

TRC	S/S
3.159634+2	3.159632+2
5.792659+2	5.792651+2
3.949432+1	3.949254+1
-1.145368+3	-1.145371+3
-2.290734+3	-2.290735+3
-2.317057+3	-2.317051+3
-1.342825+3	-1.342805+3

R1

TRC	S/S
-1.4-6	+1.5-5
-1.5-5	+6.5-5
-5.7-5	+1.6-4
-1.0-4	+3.2-4
-1.0-4	+5.4-4
+2.2-6	+7.6-4
+2.2-4	+9.6-4

TABLE III

COMPARISONS OF DISPLACEMENT
RESPONSES AT END TIP (G.P.11)
BETWEEN COMPLETE R.F.9 & S/S R.F.9
AT EVERY SECOND FOR 7 SECONDS

R2		R3	
TRC	S/S	TRC	S/S
-1.182428+1	-1.182427+1	-9.0-7	-1.0-6
-2.167783+1	-2.167781+1	-5.9-6	+2.4-6
-1.477996+0	-1.477947+0	-1.4-5	+1.7-5
4.286305+1	4.286311+1	-1.9-5	+4.5-5
8.572603+1	8.572604+1	+4.6-6	+8.0-5
8.671105+1	8.671079+1	+6.3-6	+1.0-4
5.025239+1	5.025163+1	-1.7-5	+3.0-5

TABLE IV

PHASE I STRATEGY
FOR DAREA CONDENSATION

EXECC	SOL 2,0
	ALTER
SUBSCC	OPTIONS = K,M,P
CASECC	OLOAD = ASET D.O.F.'s
	S/C 1
	LABEL = DAREA GP A COMP I VALUE R.R
	LOAD = 1
	S/C 2
	LABEL = DAREA GP B COMP J VALUE S.S
	LOAD = 2
	.
	.
	S/C N
	LABEL = DAREA GP M COMP K VALUE T.T
BULK	FORCE1 . . . . GP A . . COMP I . . VALUE R.R
	FORCE1 . . . . GP B . . COMP J . . VALUE S.S
	FORCE1 . . . . GP M . . COMP K . . VALUE T.T

TABLE V

PHASE II STRATEGY
FOR DAREA CONDENSATION

<u>EXECC</u>	SOL 2,0 ALTER
<u>SUBSCC</u>	OPTIONS = K,M,P ANY COMBINES & REDUCES OUTPUT = SUMMARY OF PSEUDOSTRUCTURE CONNECTIVITIES
<u>CASECC</u>	DNA
<u>BULK</u>	DNA

TABLE VI