

NASA Conference Publication 2018

Sixth NASTRAN Users' Colloquium

**Held at
Lewis Research Center
Cleveland, Ohio
October 4-6, 1977**

NASA



NASA Conference Publication 2018

**Sixth NASTRAN
Users' Colloquium**

**Held at
Lewis Research Center
Cleveland, Ohio
October 4-6, 1977**

NASA

National Aeronautics
and Space Administration

**Scientific and Technical
Information Office**

1977

For sale by the National Technical Information Service
Springfield, Virginia 22161
Price: \$12.50

FOREWORD

NASTRAN (NASA STRUCTURAL ANALYSIS) is a large, comprehensive, non-proprietary, general purpose finite element computer code for structural analysis which was developed under NASA sponsorship and became available to the public in late 1970. It can be obtained through COSMIC (Computer Software Management and Information Center), Athens, Georgia, and is widely used by NASA, other government agencies, and industry.

NASA currently provides continuing maintenance and improvement of NASTRAN through a NASTRAN Systems Management Office (NSMO) located at Langley Research Center. Because of the widespread interest in NASTRAN, and finite element methods in general, NSMO organized the Sixth NASTRAN Users' Colloquium held at Lewis Research Center, October 4-6, 1977. (Papers from previous colloquia held in 1971, 1972, 1973, 1975 and 1976 are published in NASA Technical Memorandums X-2378, X-2637, X-2893, X-3278, and X-3428, respectively.) The Sixth Colloquium provides some comprehensive general papers on the application of finite element methods in engineering, comparisons with other approaches, unique applications, pre- and post-processing or auxiliary programs, and new methods of analysis with NASTRAN.

Individuals actively engaged in the use of finite elements or NASTRAN were invited to prepare papers for presentation at the colloquium. These papers are included in this volume. No editorial review was provided by NASA, but detailed instructions were provided each author to achieve reasonably consistent format and content. The opinions and data presented are the sole responsibility of the authors and their respective organizations.

Cochairmen:

Deene J. Weidman, Manager
NASTRAN Systems Management Office
Langley Research Center
Hampton, Virginia

and

Christos C. Chamis
Lewis Research Center
Cleveland, Ohio

**Page
Intentionally
Left Blank**

CONTENTS

	Page
FOREWORD.	iii
NASTRAN: STATUS, PLANS, AND PERFORMANCE	
Deene J. Weidman.	ix
MANAGEMENT OF NASTRAN DEVELOPMENT AND MAINTENANCE IN A MULTI-DIVISION CORPORATION	
W. D. Mock and R. Narayanaswami	1
<u>FLUIDS AND THERMAL APPLICATIONS</u>	
3-D HYDROELASTIC ANALYSIS IN NASTRAN WITH GENERAL FLUID AND STRUCTURE GEOMETRY	
D. N. Herting, R. L. Hoesly, and D. L. Herendeen.	3
ON THE THERMO-FLUID ELEMENTS AND THEIR APPLICATIONS	
H. P. Lee	25
FINITE ELEMENT SOLUTIONS OF FREE-SURFACE FLOWS	
P. Richard Zarda and Melvyn S. Marcus	27
<u>NASTRAN PROGRAMMING</u>	
MODIFYING THE NASTRAN SYSTEM WITH A NEW CAPABILITY	
John R. McDonough	53
NORCK: A NASTRAN MODULE TO CHECK INPUT DATA AND ELEMENT GEOMETRY	
David T. Zemer.	65
STUDY OF THE NASTRAN INPUT/OUTPUT SUBSYSTEMS	
W. K. Brown and W. F. Schoellmann	75
STRUCTURAL MODEL INTEGRITY	
D. V. Wallerstein, R. S. Lahey, and G. W. Haggemacher	93
<u>SUBSTRUCTURING METHODS</u>	
A DIRECT MATRIX ABSTRACTION PROCEDURE FOR A STATIC SOLUTION DURING A PHASE I SUBSTRUCTURING ANALYSIS	
David T. Zemer.	105
A NASTRAN DMAP ALTER FOR THE COUPLING OF MODAL AND PHYSICAL COORDINATE SUBSTRUCTURES	
Thomas L. Wilson.	119
APPLICATION OF NASTRAN LEVEL 16.0 AUTOMATED MULTISTAGE SUBSTRUCTURING TECHNIQUES	
R. M. Bereznak and C. S. Sahota	131
AUTOMATED MULTI-LEVEL SUBSTRUCTURING FOR SPERRY'S VERSION OF NASTRAN	
Gordon C. Chan and Ronald P. Schmitz	133
UNEXPECTED DIVIDENDS FROM LEVEL 16 AUTOMATED SUBSTRUCTURING	
Eric I. Field	155
SUBSTRUCTURE COUPLING - A DIFFERENT APPROACH	
R. S. Lahey	161

UNIQUE NEW APPLICATIONS

ANALYSIS OF MAGNETIC FIELDS USING VARIATIONAL PRINCIPLES AND CELAS2 ELEMENTS
John W. Frye and Rolf G. Kasper 175

APPLICATION OF SYMBOLIC/NUMERIC MATRIX SOLUTION TECHNIQUES TO THE NASTRAN PROGRAM
E. M. Buturla and S. H. Burroughs 193

ALTERNATE APPROACHES TO VIBRATION AND SHOCK ANALYSIS USING NASTRAN
Richard E. Denver and Joseph M. Menichello. 199

NASTRAN USE FOR CYCLIC RESPONSE AND FATIGUE ANALYSIS OF WIND TURBINE TOWERS
C. C. Chamis, P. Manos, J. H. Sinclair, and J. R. Winemiller 213

LARGE DEFORMATION APPROACHES

NEW LARGE DEFLECTION ANALYSIS FOR NASTRAN
Myles M. Hurwitz. 235

SINGULAR PLASTIC ELEMENT: NASTRAN IMPLEMENTATION AND APPLICATION
M. A. Hussain, S. L. Pu, and W. E. Lorensen 257

APPLICATION OF THE TRPLT1 ELEMENT TO LARGE AMPLITUDE FREE VIBRATIONS OF PLATES
Chuh Mei and James L. Rogers, Jr. 275

GENERAL AUXILIARY PROGRAMS

A CONDENSED FORM OF NASTRAN
James L. Rogers, Jr., Chuh Mei, and W. Keith Brown. 299

INTERACTIVE GRAPHICS SUPPORT FOR NASTRAN
William E. Lorensen 305

A STAND-ALONE INTERACTIVE GRAPHICS FINITE ELEMENT MODELING SYSTEM
Jeffrey Z. Gingerich, Marv M. Abe, Randall L. Vinecore,
Gary J. Romans, and Barry M. Ratihn 327

RINA - AN INTERACTIVE SYSTEM FOR THE RAPID INTERPRETATION OF NASTRAN RESULTS
A. I. Raibstone and A. Pipano 341

SOFTWARE FOR TRANSFERRING NASTRAN DATABLOCKS BETWEEN DISSIMILAR COMPUTERS
Richard Rosencranz, James L. Rogers, Jr., and Reg S. Mitchell 357

FINITE ELEMENT ANALYSIS OF FIBER WRAPPED SHELLS WITH NON-SYMMETRIC LOADS
G. Peter O'Hara 369

SPECIFIC APPLICATIONS

NASTRAN FINITE ELEMENT IDEALIZATION STUDY
W. R. Case and J. B. Mason. 383

STRESS ANALYSIS OF PRESSURE VESSEL ASSEMBLY
Richard T. Eppink, Furman W. Barton, and Lawrence H. Gilligan 405

NEW CAPABILITIES

AN AUTOMATED DATA GENERATOR FOR NASTRAN
Edward L. Stanton 419

DEVELOPMENT OF AN AUTOMATED MULTI-STAGE MODAL SYNTHESIS SYSTEM FOR
NASTRAN

D. N. Herting and R. L. Hoesly. 435

ADDITION OF RIGID ELEMENTS TO NASTRAN

R. R. Pamadi and J. D. Cronkhite. 449

USE OF NEW MATRIX ASSEMBLER FOR DIFFERENTIAL STIFFNESS MATRICES

R. Narayanaswami and J. G. Cole 469

Space
Vibration
Analysis

**Page
Intentionally
Left Blank**

NASTRAN: STATUS, PLANS, AND PERFORMANCE

Deene J. Weidman
NASA Langley Research Center

INTRODUCTION

NASTRAN has been developed and improved over the past thirteen years, and is currently one of the most widely used programs in the U. S. In order to provide users with some indication of the future content of the program, this paper outlines the NASA plans for new capability to be released in Level 17 and the dates this new capability will be completed. The target release date for Level 17 is December 1977 and this version will be leased domestically through COSMIC. As an example of the improved efficiency of NASTRAN Level 16, a table of CPU run-time comparisons is shown.

LEVEL 17 IMPROVEMENTS AND STATUS

The most needed additions to NASTRAN Level 16 have already been selected and are in the final processes of being developed and installed. A list of these improvements, their expected installation dates, and colloquia references describing the additions are given below:

ITEM	DATE	REFERENCE
Matrix Conditioning Checks	6/77	Ref. 1, paper 1
Improved Elements (Rigid, Membrane, Plate, Shell)	7/77	Ref. 2, paper 21 Ref. 3, paper 35
Subsonic Flutter Improvement Package	8/77	Ref. 2, paper 23
Supersonic Aeroelasticity with Gust Analysis	8/77	Ref. 1, paper 1
FEER Eigenmethods (Real and Complex)	8/77	Ref. 2, paper 8
Automated Modal Synthesis	9/77	Ref. 3, paper 34
General Purpose Data Generator Package	2/78	Ref. 3, paper 33

From this table, it appears the only capability that may not be available for Level 17 release in December 1977 is the data generator package. All of this work is progressing rapidly towards completion.

The Error Correction Information System (ECIS) is a system available to all users that supplies complete, up-to-date information on all known errors in the standard version of NASTRAN. This information includes procedures for avoiding errors and allowing users to work around these program faults. It is also used by other installations maintaining their own company-unique versions of NASTRAN since it gives them corrections that are usually applicable to fixing their own versions. This system is working very well, has been extended to allow (1) reporting of errors by users and (2) direct response from the maintenance contractor to the reporting individual, and contains a current "important information" file that can be easily accessed (for each computer) with the latest urgent messages.

The NASTRAN User's Guide (NASA CR-2504) has been available for two years now, and has received much favorable comment. This document allows a new user to become familiar with some aspect of NASTRAN analysis that he hadn't run before, and shows examples using Level 15 to lead him to correct application of the program. This document is currently being improved and up-dated to be applicable to Level 17 and should be available in 1978 to any interested user through the National Technical Information Service in Springfield, Va. 22151.

RUN-TIME COMPARISONS

In order to give an adequate comparison of computer run-times (CPU seconds primarily) between various versions of the standard levels of NASTRAN, tables 1 and 2 have been assembled showing run-times for all of the standard demonstration problems as delivered to the users on the User's Master File. They illustrate a full range of NASTRAN capabilities. It can be noted that Level 16 saves approximately 37% of the CPU time of Level 15 for the IBM 360/95 computers, and over 39% of the Level 15.5 CPU time for the CDC 6600 computers. The UNIVAC values shown are CAUs (Cost Account Units, a general overall cost estimate) for the 1110, and results are not directly comparable with the Level 15 values for the 1108. These values are only shown so that UNIVAC users can estimate their computer run-times for Level 16. Further significant savings in run-times over these Level 16 values could be obtained if its unique features (such as CNGRNT cards) had been utilized. These capabilities were not available in earlier levels.

A comparison of computer run-times after the introduction of a new compiler is shown in table 3. For the nine problems chosen at random, an improvement in run-times of over 30% was noted and indicates that versions of NASTRAN using different compilers during their generation cannot be directly compared to determine the efficiency of the structural coding.

CONCLUDING REMARKS

The status of Level 17 of NASTRAN was discussed and incorporation dates for new capability specified. In addition, computer run-times were presented indicating a significant reduction in CPU times for Level 16.

REFERENCES

1. NASTRAN: Users' Experiences. NASA TM X-3278, 1975.
2. NASTRAN: Users' Experiences. NASA TM X-3428, 1976.
3. Sixth NASTRAN Users' Colloquium. NASA CP-2018, 1977.

TABLE 1.- COMPARISON OF NASTRAN DEMONSTRATION PROBLEM EXECUTION TIMES FOR THE IBM COMPUTER

Demonstration Problems		IBM 360/95 ¹						
No.	Form	Level 15 ²	Level 15.9 ³	Level 16 ³	Level 16-Level 15		Level 16-Level 15.9	
		Sec.	Sec.	Sec.	Sec.	% ⁴	Sec.	% ⁴
1-1-1	UMF	*	17.7	15.667			- 2.0	- 12.8 %
1-1-1A	Restart	7	10.9	12.050	5.1	+42.1 %	1.2	+ 9.9 %
1-1-1B	Restart	15	22.5	19.517	4.5	+23.1 %	- 3.0	- 15.4 %
1-1-2	UMF	**	25.1	18.383			- 6.7	- 36.4 %
1-1-3	UMF	*	21.2	18.333			- 2.9	- 15.8 %
1-1-4	UMF	*	20.2	14.900			- 5.3	- 35.6 %
1-2-1	UMF	*	24.2	22.184			- 2.0	- 9.0 %
1-2-1A	Restart	10	14.2	12.517	2.5	+20.0 %	- 1.7	- 13.6 %
1-3-1	UMF	*	45.0	28.833			-16.2	- 56.3 %
1-3-2	UMF	*	84.7	30.817			-53.9	-175.0 %
1-3-3	UMF	*	89.2	35.433			-53.8	-152.0 %
1-4-1	UMF	.139	112.7	74.400	-64.6	-86.8 %	-38.3	- 51.5 %
1-4-1A	Restart	*	17.0	13.883			- 3.1	- 22.3 %
1-4-2	UMF	179	124.6	93.067	-85.9	-92.3 %	-31.5	- 33.8 %
1-4-3	UMF/INPUT	67	48.4	38.333	-28.7	-74.9 %	-10.1	- 26.4 %
1-4-4	UMF/INPUT	84	54.0	54.217	-29.8	-55.0 %	0.2	+ 0.4 %
1-5-1	UMF	111	143.7	104.367	- 6.6	- 6.3 %	-39.3	- 37.6 %
1-6-1	UMF	10	14.7	12.567	2.6	+20.6 %	- 2.1	- 16.7 %
1-7-1	UMF	21	34.7	33.300	12.3	+36.9 %	- 1.4	- 4.2 %
1-8-1	UMF	16	26.1	19.717	3.7	+18.8 %	- 6.4	- 32.5 %
1-9-1	UMF	*	38.5	23.467			-15.0	- 63.8 %
1-10-1	UMF	*	11.5	11.767			0.3	+ 2.5 %
1-11-1	UMF	30	39.2	19.067	-10.9	-57.1 %	-20.1	-105.2 %
1-11-2	UMF/INPUT	*	24.3	18.900			- 5.4	- 28.6 %
1-12-1	UMF	*	9.2	8.950			- 0.2	- 2.2 %
1-12-2	UMF	*	5.9	6.817			0.9	+ 13.2 %
1-13-1	UMF	**	29.7	32.650			3.0	+ 9.2 %
1-13-2	UMF	**	20.9	15.067			- 5.8	- 38.4 %
1-13-3	UMF	**	21.3	17.700			- 3.6	- 20.3 %
1-14-1	UMF	**	9.0	9.217			0.2	+ 2.2 %
1-15-1	UMF	**	323.3	341.000			17.7	+ 5.2 %
1-16-1	UMF	**	45.5	64.416			18.9	+ 29.3 %

* Results Not Available; ** New for Level 15.9; *** New for Level 16

TABLE 1.- CONTINUED

Demonstration Problems		IBM 360/95 ¹						
No.	Form	Level 15 ²	Level 15.9 ³	Level 16 ³	Level 16-Level 15		Level 16-Level 15.9	
		Sec.	Sec.	Sec.	Sec.	% ⁴	Sec.	% ⁴
2-1-1	UMF	*	12.2	11.467	-		- 0.7	- 6.1 %
2-2-1	UMF	***	***	10.450				
2-2-2	UMF	***	***	7.583				
2-2-3	UMF	***	***	8.617				
2-2-4	UMF	***	***	64.215				
2-2-5	UMF	***	***	10.067				
2-2-6	UMF	***	***	13.283				
2-2-7	UMF	***	***	36.917				
3-1-1	UMF	105	113.0	53.350	- 51.6	- 96.6%	- 59.6	-111.6 %
3-1-2	UMF	228	376.0	154.217	- 73.8	- 47.9%	-221.8	-143.8 %
3-1-3	UMF/INPUT	92	54.8	50.499	- 41.5	- 82.2%	- 4.3	- 8.5 %
3-1-4	UMF/INPUT	510	146.0	142.616	-367.4	-257.6%	- 3.4	- 2.4 %
3-2-1	UMF	*	95.3	103.400			8.1	+ 7.8 %
3-3-1	UMF	79	56.8	72.167	- 6.8	- 9.4%	15.4	+ 21.3 %
3-4-1	UMF	*	34.3	35.317			1.0	+ 2.8 %
3-5-1	UMF	**	9.4	9.433			0.0	- 0.0 %
3-6-1	UMF	**	12.1	11.767			- 0.3	- 2.5 %
4-1-1	UMF	31	40.0	27.133	- 3.9	- 14.4%	- 12.9	- 47.6 %
5-1-1	UMF	*	119.4	88.783			- 30.6	- 34.5 %
6-1-1	UMF	353	595.9	456.750	103.8	+ 22.7%	-139.1	- 30.5 %
7-1-1	UMF	*	75.4	75.783			0.4	+ 0.5 %
7-1-2	UMF/INPUT	*	49.9	49.900			0.0	- 0.0 %
7-2-1	UMF	*	295.4	273.733			- 21.7	- 7.9 %
7-2-2	UMF	*	181.7	171.117			- 10.6	- 6.2 %
8-1-1	UMF	*	46.5	23.967			- 22.5	- 93.8 %
8-1-2	UMF	*	178.6	86.900			- 91.7	-105.5 %
8-1-3	UMF/INPUT	*	24.2	23.850			- 0.3	- 1.3 %
8-1-4	UMF/INPUT	*	70.2	80.133			9.9	+ 12.4 %
9-1-1	UMF	*	14.9	14.783			- 0.1	- 0.7 %
9-2-1	UMF	81	112.0	105.617	24.6	+ 23.3%	- 6.4	- 6.1 %
9-2-2	UMF/INPUT	78	101.6	99.567	21.6	+ 21.7%	- 2.0	- 2.0 %

* Results Not Available; ** New for Level 15.9; *** New for Level 16

111x

TABLE 1.- CONCLUDED

Demonstration Problems		IBM 360/95 ¹						
No.	Form	Level 15 ²	Level 15.9 ³	Level 16 ³	Level 16-Level 15		Level 16-Level 15.9	
		Sec.	Sec.	Sec.	Sec.	% ⁴	Sec.	% ⁴
9-3-1	UMF	286	234.8	179.000	-107.0	- 59.8%	- 55.8	- 31.2 %
9-4-1	UMF	**	13.4	13.366			0.0	- 0.0 %
10-1-1	UMF	*	28.4	27.333			- 1.1	- 4.0 %
10-2-1	UMF	**	31.9	23.900			- 8.0	- 33.5 %
11-1-1	UMF	*	31.5	24.534			- 7.0	- 28.6 %
11-1-1A	Restart	8	*	10.233	2.2	+ 21.6%		
11-2-1	UMF	271	189.6	126.650	-144.3	-113.9%	- 62.9	- 49.6 %
11-2-2	UMF/INPUT	124	*	169.233	45.2	+ 26.7%		
12-1-1	UMF	*	79.1	76.133			- 3.0	- 3.9 %
13-1-1	UMF	**	37.3	34.200			- 3.1	- 9.1 %
14-1-1	UMF	**	23.2	20.617			- 2.6	- 12.6 %
15-1-1	UMF	**	31.4	23.500			- 7.9	- 33.6 %
Totals		2,935	5,045.3	4,343.583	-794.8	- 37.14%	-1,032.3	- 25.72%
No. Demos		25	66	75				

* Results Not Available; ** New for Level 15.9; *** New for Level 16

1. All times are seconds of CPU. PLOT module times have been subtracted. No adjustments have been made to any times to compensate for differences in core speeds.
2. Level 15 jobs were run in high speed core.
3. Level 15.9 and Level 16 jobs were run in low speed core.
4. Level 16.0 is used as the base. A positive percentage indicates the relative increase in run-time for Level 16 when compared to the earlier level.

APX

TABLE 2.- COMPARISON OF NASTRAN DEMONSTRATION PROBLEM EXECUTION TIMES FOR THE CDC AND UNIVAC COMPUTERS

Demonstration Problems		CDC Level 15.5.0 or 15.5.1		CDC Level 16.0.0		CDC Level 16-Level 15.5		UNIVAC Level 15	UNIVAC Level 16.0.1.A
No.	Form	6400	6600	6400	6600	Sec.	%	1108	1110
1-1-1	UMF	*	41.414	*	44.440	3.0	+ 6.8	30	19.966
1-1-1A	Restart	*	*	*	32.566			17	13.603
1-1-1B	Restart	*	44.894	*	48.610	3.7	+ 7.6	41	26.714
1-1-2	UMF	***	***	116.522	*			***	28.518
1-1-3	UMF	138.494	*	*	69.404			*	32.457
1-1-4	UMF	*	35.978	*	43.502	7.5	+ 17.3	*	23.849
1-2-1	UMF	111.856	*	*	69.256			55	67.045
1-2-1A	Restart	*	*	*	38.842			25	16.967
1-3-1	UMF	174.596	*	*	61.994			86	41.586
1-3-2	UMF	*	482.770	*	84.280	-398.5	-472.8	*	50.213
1-3-3	UMF	*	139.910	*	78.470	- 61.4	- 78.3	*	58.765
1-4-1	UMF	*	327.466	657.622	*			510	261.686
1-4-1A	Restart	*	*	*	33.492			*	21.226
1-4-2	UMF	989.970	*	816.534	*	-173.4	- 21.2	*	361.145
1-4-3	UMF/INPUT	724.308	*	215.088	*	-509.2	-236.7	*	148.404
1-4-4	UMF/INPUT	966.022	*	308.268	*	-657.8	-213.4	*	232.945
1-5-1	UMF	*	414.832	*	331.834	- 83.0	- 25.0	477	226.269
1-6-1	UMF	*	29.152	*	39.108	10.0	+ 25.5	22	16.985
1-7-1	UMF	*	121.502	*	132.152	10.7	+ 8.1	115	65.562
1-8-1	UMF	95.224	*	69.140	*	- 26.1	- 37.7	47	24.821
1-9-1	UMF	156.226	*	88.246	*	- 68.0	- 77.0	76	32.535
1-10-1	UMF	47.238	*	57.176	*	9.9	+ 17.4	18	13.120
1-11-1	UMF	*	82.418	96.048	*			83	32.197
1-11-2	UMF/INPUT	*	*	93.008	*			*	31.479
1-12-1	UMF	*	25.838	41.512	*			18	9.247

* Results Not Available; *** New for Level 16

1. All times are seconds of CPU on the indicated machine except for the UNIVAC 1110 times which are CAU times. PLØT module times have been subtracted except for the UNIVAC 1110 times. No adjustments have been made to any times to compensate for differences in core speeds.

2. Level 16 is used as the base.

TABLE 2.- CONTINUED

Demonstration Problems		CDC Level 15.5.0 or 15.5.1		CDC Level 16.0.0		CDC Level 16-Level 15.5		UNIVAC Level 15	UNIVAC Level 16.0.1.A
No.	Form	6400	6600	6400	6600	Sec.	%	1108	1110
1-12-2	UMF	*	*	32.364	*			*	6.666
1-13-1	UMF	***	***	*	102.296			***	61.391
1-13-2	UMF	***	***	*	50.444			***	22.262
1-13-3	UMF	***	***	*	63.776			***	27.549
1-14-1	UMF	***	***	*	27.920			***	8.552
1-15-1	UMF	***	***	*	1,429.846			***	1,437.254
1-16-1	UMF	***	***	296.868	*			***	10.706
2-1-1	UMF	*	32.542	*	55.122	22.6	+ 41.0	24	13.123
2-2-1	UMF	***	***	*	28.546			***	10.706
2-2-2	UMF	***	***	*	26.466			***	7.083
2-2-3	UMF	***	***	*	20.522			***	7.403
2-2-4	UMF	***	***	*	168.594			***	167.610
2-2-5	UMF	***	***	*	48.608			***	10.807
2-2-6	UMF	***	***	*	38.492			***	15.358
2-2-7	UMF	***	***	*	153.858			***	70.577
3-1-1	UMF	*	312.764	*	104.322	-208.4	-199.8	422	113.463
3-1-2	UMF	*	*	*	346.310			*	392.472
3-1-3	UMF/INPUT	*	336.234	*	102.000	-234.2	-229.6	*	113.516
3-1-4	UMF/INPUT	*	1,055.020	*	327.756	-727.3	-221.9	*	375.690
3-2-1	UMF	*	281.092	*	199.692	- 81.4	- 40.8	405	202.142
3-3-1	UMF	*	256.436	*	171.948	- 84.5	- 49.1	354	126.040
3-4-1	UMF	173.432	*	*	83.780			107	107.776
3-5-1	UMF	***	***	*	31.862			***	7.531
3-6-1	UMF	***	***	*	33.276			***	10.781
4-1-1	UMF	167.110	*	*	76.702			102	57.780

* Results Not Available; *** New for Level 16

1. All times are seconds of CPU on the indicated machine except for the UNIVAC 1110 times which are CAU times. PLOT module times have been subtracted except for the UNIVAC 1110 times. No adjustments have been made to any times to compensate for differences in core speeds.
2. Level 16 is used as the base.

TABLE 2.- CONCLUDED

Demonstration Problems		CDC Level 15.5.0 or 15.5.1		CDC Level 16.0.0		CDC Level 16-Level 15.5		UNIVAC Level 15	UNIVAC Level 16.0.1.A
No.	Form	6400	6600	6400	6600	Sec.	%	1108	1110
5-1-1	UMF	715.536	*	*	288.168			287	233.927
6-1-1	UMF	*	1,339.750	*	1,303.570	- 36.2	- 2.8	1,361	878.606
7-1-1	UMF	381.146	*	*	231.044			188	115.252
7-1-2	UMF/INPUT	*	199.974	*	213.238	13.3	+ 6.2	*	103.291
7-2-1	UMF	1,910.568	1,269.600	*	1,518.812	222.2	+14.6	765	675.982
7-2-2	UMF	1,479.252	*	*	680.444			386	259.866
8-1-1	UMF	243.788	*	*	56.634			115	33.182
8-1-2	UMF	1,142.622	*	*	202.384			*	158.321
8-1-3	UMF/INPUT	*	113.340	*	58.860	- 54.5	-92.6	*	32.557
8-1-4	UMF/INPUT	1,130.660	*	*	198.436			*	151.776
9-1-1	UMF	68.256	*	*	86.338			24	106.576
9-2-1	UMF	*	315.958	*	273.244	- 42.7	-15.6	318	162.989
9-2-2	UMF/INPUT	*	290.290	*	268.846	- 21.4	- 8.0	*	153.864
9-3-1	UMF	*	345.632	*	368.528	22.9	+ 8.5	422	349.216
9-4-1	UMF	***	***	*	41.650			***	13.716
10-1-1	UMF	*	98.382	*	90.420	- 8.0	- 8.8	54	27.037
10-2-1	UMF	***	***	*	75.370			***	27.730
11-1-1	UMF	106.248	*	142.064	*	35.8	+25.2	70	62.511
11-1-1A	Restart	*	*	44.140	*			16	11.517
11-2-1	UMF	*	477.300	*	382.114	- 95.2	-24.9	495	429.125
11-2-2	UMF/INPUT	*	486.894	*	393.350	- 93.5	-23.8	*	427.388
12-1-1	UMF	229.808	*	*	158.808			167	193.060
13-1-1	UMF	***	***	178.596	*			***	56.280
14-1-1	UMF	***	***	89.398	*			***	*
15-1-1	UMF	***	***	104.256	*			***	37.239
Totals		11,152.360	8,957.382	3,446.850	11,690.346	-3276.2	-39.3		
No. Demos		21	27	18	57	31			

* Results Not Available; *** New for Level 16

1. All times are seconds of CPU on the indicated machine except for the UNIVAC 1110 times which are CAU times. PL0T module times have been subtracted except for the UNIVAC 1110 times. No adjustments have been made to any times to compensate for differences in core speeds.

2. Level 16 is used as the base.

TABLE 3.- COMPARISON OF NASTRAN DEMONSTRATION PROBLEM EXECUTION TIMES
ON THE CDC CYBER 175 USING A NEW COMPILER

Demonstration Problem	Run Time (Plot Time), CPU Sec.		% Reduction of RUNX Times
	RUNX Compiler,L16.0.1	FTN Compiler,L16.0.8	
1-1-1	11.825	10.021	15.3
3-6-1	8.025	6.417	20.0
8-1-3	16.996	15.473	9.0
9-4-1	8.983 (.004)	7.728 (.003)	14.0
10-2-1	18.083 (.002)	14.493 (.003)	19.9
12-1-1	75.205 (23.979)	66.808 (12.141)	11.2
15-1-1	18.353	16.956	7.6
1-12-2	4.849	3.827	21.1
7-2-1	493.356	312.307	36.7
TOTALS	655.675	454.03	30.75

MANAGEMENT OF NASTRAN DEVELOPMENT AND MAINTENANCE
IN A MULTI-DIVISION CORPORATION

W. D. Mock* and R. Narayanaswami**

Rockwell International

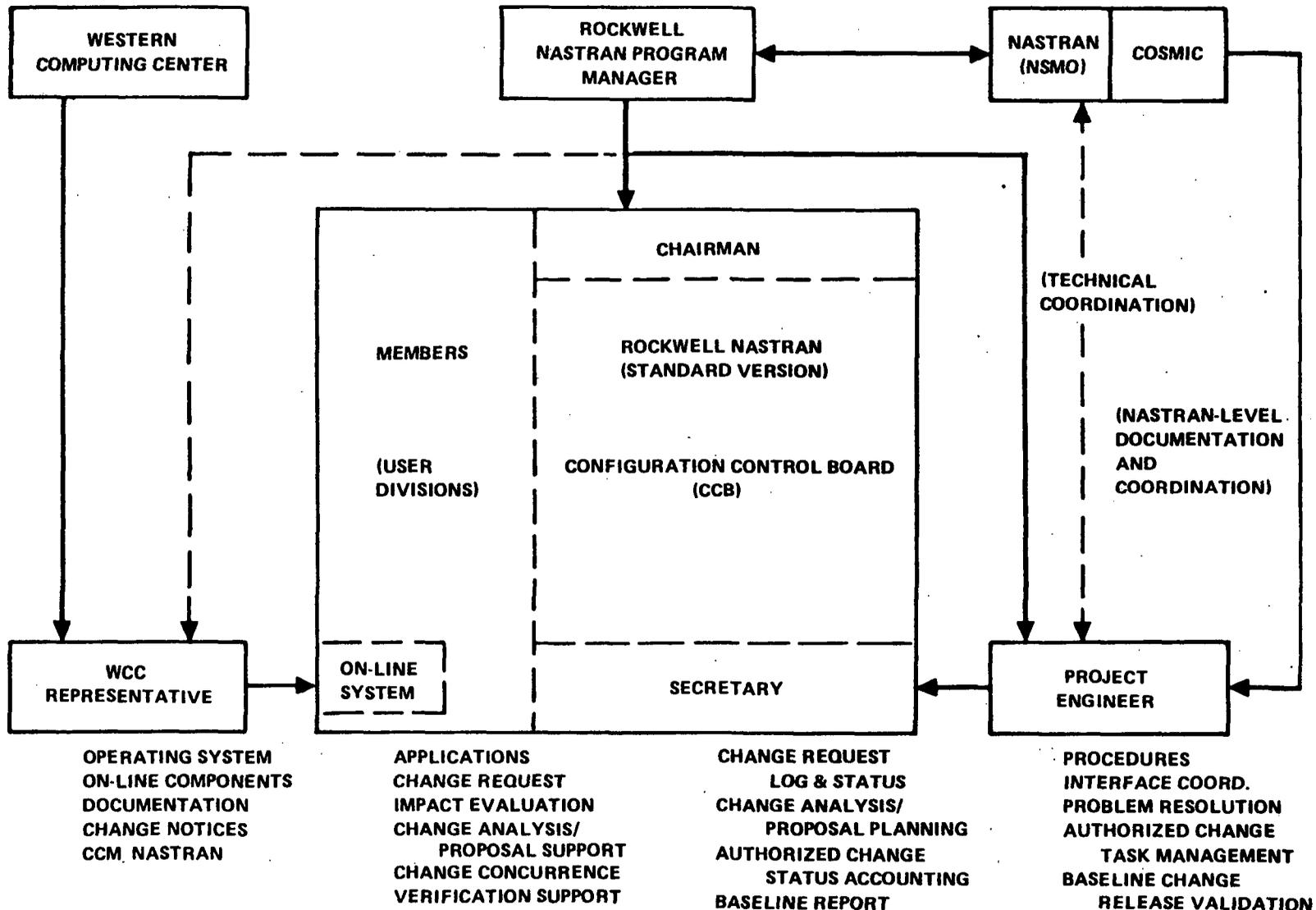
ABSTRACT

Managing the NASTRAN program development and maintenance in a multi-division aerospace corporation is an involved process that presents technical and management-related challenges. This paper presents an overview (attached schematic) of the NASTRAN program management system developed and implemented at Rockwell International. The NASTRAN Level 16.0 as released through COSMIC was installed on IBM OS 370/168 and CYBER 175 computing systems located at Rockwell's Central Computing Center. The basic program has been modified to incorporate technical and efficiency improvements. Eleven divisions of Rockwell International participate in NASTRAN Group Service, which develops and maintains the Rockwell NASTRAN program system. A Rockwell NASTRAN Configuration Control Board with representation from the participating divisions provides divisional inputs to the program development. The Rockwell NASTRAN program manager and project engineer administer the management and technical direction of the Rockwell NASTRAN program development.

* Rockwell NASTRAN Program Manager.

** Rockwell NASTRAN Project Engineer.

ROCKWELL NASTRAN CONFIGURATION CONTROL BOARD



3-D HYDROELASTIC ANALYSIS IN NASTRAN WITH GENERAL FLUID AND STRUCTURE GEOMETRY

D. N. Herting, R. L. Hoesly and D. L. Herendeen
Universal Analytics, Inc.

SUMMARY

The implementation of a general three-dimensional hydroelastic mode analysis capability in NASTRAN is presented. Finite elements with polyhedral shapes define the fluid; existing NASTRAN plate elements define the fluid/structure interface. Efficient solution methods were implemented to allow a separate structural matrix reduction and to allow connection of the fluid mass directly to a small set of grid points or modal coordinates representing the structure. Test case results for the various solution options are presented.

INTRODUCTION

This paper describes the implementation of a general three-dimensional hydroelastic capability in the NASTRAN (NASA Structural Analysis) computer program. Although NASTRAN had provided capabilities for the analysis of compressible fluids with axisymmetric geometry, a more general capability and more efficient solution procedure were desired. The basic approach described in this paper extends the capabilities to provide for arbitrary fluid shapes, including tilted free surfaces, and allows for more efficient methods of obtaining the solutions.

Although the present Level 16 NASTRAN hydroelastic capability, described in Reference 1, will handle non-axisymmetric structures, the fluid finite element model must represent an axisymmetric volume. Furthermore, the NASTRAN method of formulating the solution matrices for compressible fluids results in large, unsymmetric stiffness and mass matrices, requiring costly complex eigenvalue extraction methods.

However, most applications do not require the compressibility effects. With this assumption, a modified method of formulating the matrices may be used, resulting in symmetric matrices with the fluid represented by a mass

matrix. Programs using this technique, as described in References 3, 4, 5, and 8, are typically restricted to axisymmetric fluids and result in large-order, full mass matrices connecting all of the structure node points. The method described herein reduces the order of those matrices and provides a more efficient solution for large-order problems.

The primary goal of the program development was to provide a general method for analyzing the combined mode shapes of arbitrary fluid and structure finite element models. The fluid is modeled with three-dimensional solid elements with options for tetrahedral, wedge, and hexahedral shapes. The elements are connected to fluid grid points which define the pressure in the fluid at the specified location. The structure may be modeled arbitrarily using the existing NASTRAN elements. The fluid/structure interface and the free surface are defined by the user with special NASTRAN boundary elements. A special purpose mesh generator program was used to generate the actual NASTRAN data cards for the fluid, the structure, and the boundary elements for typical tank-type models.

A second goal was to provide efficient solutions for large-order problems. This was provided by a method in which the structural matrices are processed separately and may be reduced. Matrix condensation procedures (\emptyset MIT) or modal formulation using the normal modes of the empty structure as solution coordinates may be used. The fluid matrices are then transformed and connected to the reduced structure coordinates resulting in small, symmetric solution matrices. This approach is particularly valuable when several different fluid levels are to be analyzed for one structure. The structure formulation and reduction is processed only once. The additional calculations for each different fluid case require only fluid matrix operations and solution processing.

User convenience was provided in the system with the implementation of several alternate solution paths and modeling options. These options, which allow a wide variety of problem types and provide the user with efficiency and accuracy trade-offs, are summarized below.

- The Direct formulation option uses structure grid point coordinates and free surface displacements as solution degrees of freedom. The structural matrices may be reduced using the NASTRAN matrix condensation technique (ASET or \emptyset MIT data) for more economical processing of large-order problems.
- The modal formulation option uses the mode shapes of the empty structure as generalized solution coordinates.
- Gravity effects are provided which affect both the free surface displacements and the structure-fluid interface. The gravity effects may also be deleted on user option.

- Symmetric boundaries with symmetric or antisymmetric solution cases may be modeled.
- Compressibility effects may be modeled by either providing a factor to define the overall pressure versus volume change or provide a constraint on the volume change.
- Restart logic is provided in the DMAP (Direct Matrix Abstraction Program) to allow changes in the fluid model without reformulating the structure matrices, or generating structure modes.
- A special purpose NASTRAN mesh/input generator, MESHGEN, is provided. This feature allows both the finite element idealization of a structural shell and its three dimensional fluid contents. This is a highly versatile stand-alone utility controlled by an english language based control structure, MESHLAN, that is oriented toward the structural analyst using terminology that is familiar. This package significantly reduces engineering time in solving hydroelastic problems.

All of the above capabilities were specifically designed for the large-order finite element models anticipated for use in the analysis of the Space Shuttle tanks.

THEORETICAL DEVELOPMENT

In this section the basic theory for the general three-dimensional hydroelastic analysis in NASTRAN is summarized. Both the structure and the interacting fluid will be idealized as general, three-dimensional finite element models. Effects of free surfaces and steady-state gravity will be included. The fluids are assumed to be incompressible, irrotational, and nonviscous. Small motions of both structure and fluid relative to the static solution will be analyzed.

The basic development of the finite element equations for small motions of fluids is described in Reference 1. In Reference 2, the basic equations are cast in the form of integrals representing the time derivatives of Energy and Work using the fluid pressures as the unknown coordinates. The scalar pressures, rather than three displacements, will be used as degrees of freedom at each point in the fluid, which avoids extraneous rotational motions and directly provides for incompressibility. The disadvantage is that the structure and fluid are not automatically connected at the boundary. The pressures in the fluid must be related to the displacements of the boundaries through area factors and flow relationships.

Fluid Field Equations

In Reference 2 the fluid field equations are developed in the form of energy integrals using principles of variational calculus. The basic result for the compressible case is the equation:

$$\delta \left[\int_V \left(\frac{1}{2\beta} \dot{p}^2 + \frac{1}{2\rho} \nabla p \cdot \nabla p \right) dV \right] - \int_S \delta p \left(\frac{1}{\rho} \nabla p \right) \cdot d\hat{S} = 0 \quad (1)$$

where: p is the pressure

\dot{p} is the time derivative of pressure

β is the bulk modulus

ρ is the mass density

V is the volume

$d\hat{S}$ is an incremental surface vector (outward)

∇ is the vector gradient operator

δ is the variational operator

With the incompressible fluids, the bulk modulus is assumed infinite and the \dot{p}^2 term disappears. On the exterior surface the pressure gradient may be replaced with the acceleration vector using the basic momentum equation:

$$\nabla p = - \rho \hat{u} \quad (2)$$

Equation (1) therefore becomes:

$$\delta \int_V \frac{1}{2\rho} (\nabla p \cdot \nabla p) dV + \int_S \delta p \hat{u} \cdot d\hat{S} = 0 \quad (3)$$

or

$$\delta U + \delta W = 0 \quad (4)$$

In the finite element method of solution, a set of variables, p_i , equal to the value of p at specific points, is chosen and the volume is divided into subregions, called fluid elements, with vertices defined by the location of the variables. Using finite elements, Equation 3 may be expressed in the following matrix form:

$$[K^f]\{p\} + [B]\{\ddot{u}\} = 0 \quad (5)$$

where

$$B_{ij} = \frac{\partial}{\partial p_i} \frac{\partial}{\partial u_j} \int_s p \vec{u} \cdot d\vec{S} \quad (6)$$

and

$$K_{ij}^f = \frac{\partial}{\partial p_i} \frac{\partial}{\partial p_j} \int_v \frac{1}{2\rho} \nabla p \cdot \nabla p \, dV \quad (7)$$

On the other hand, the fluid produces work on the structure by applying forces over the structure surface area. The structure forces $\{F\}$ may be defined as:

$$\{F\} = [A]\{p\} \quad (8)$$

where

$$A_{ij} = \frac{\partial}{\partial u_i} \frac{\partial}{\partial p_j} \int_s p \vec{u} \cdot d\vec{S} \quad (9)$$

Comparing Eqs. (6) and (9) we observe that

$$[A] = [B^T] \quad (10)$$

If $[M^S]$ and $[K^S]$ are the mass and stiffness matrices of the structure, the matrix equation for the structure coordinates is

$$[M^S]\{\ddot{u}\} + [K^S]\{u\} - [A]\{p\} = \{0\} \quad (11)$$

Equations (5) and (11) become the system of equations for a solution.

Finite Fluid Elements

Three types of fluid elements are used to represent the three-dimensional fluid: the 4-point tetrahedron, the 6-point "wedge," and the 8-point hexahedron. The wedge and hexahedral elements are fabricated from three and ten

tetrahedra, respectively, as shown in Figure 1. The pressure function within each tetrahedron is assumed to be a linear function in the three directions, or:

$$p = q_0 + q_1x + q_2y + q_3z \quad (12)$$

Element stiffness matrices are formulated with the same equations as the present NASTRAN solid elements except that the factor $(1/p)$ is used as an effective scalar "stiffness" coefficient.

Fluid/Structure Boundary Matrices

From the general development, the area matrix [A] is defined as

$$A_{ij} = \frac{\partial}{\partial u_i \partial p_j} \int_S pu \, dS \quad (13)$$

where u and p are the displacements and pressures at the surface, S . The intersecting areas of the structure and the fluid are specified by the user as fluid-structure elements pairs. From elementary geometry, the locations of the fluid points and the structure points are obtained in a coordinate system on the fluid face. Equation (13) is evaluated for each intersecting area of structure and fluid. For simplicity, only triangular structure elements are considered below. Quadrilateral elements are treated as four overlapping triangles.

Several possible examples of overlapping areas are shown in Figure 2. Clearly, the number of combinations is too numerous to identify each case and provide a specific set of equations. Rather, a general algorithm was developed.

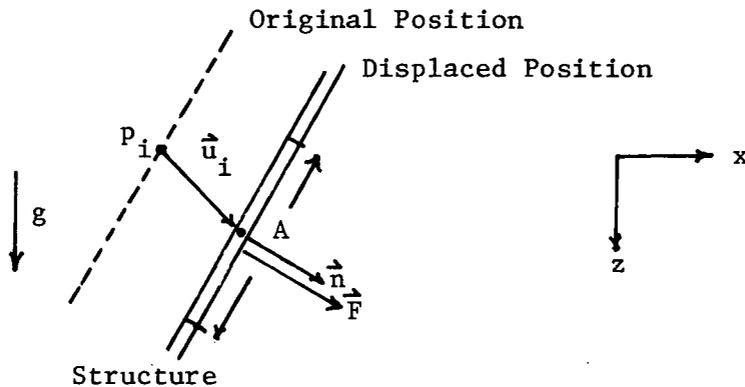
Briefly the method may be described as follows:

- (a) The intersection points defining the planar projection of the shaded area are obtained from the geometry of the two basic elements.
- (b) The pressure and displacement fields within the associated elements are assumed to be linear functions of the corner points.
- (c) Integration over the shaded area is performed using Eq. 14 for each of the area coefficients A_{ij} .

Gravity Effects

When a steady-state acceleration such as gravity is present in a hydro-elastic problem, additional terms must be added to the fundamental equations to account for the steady-state pressure gradient. In the fluid formulation, the Euler equations assume that the pressure is defined at points fixed in space, and the fluid particles flow across the point. In the structure formulation, a Lagrange assumption is used whereby the grid points remain attached to the moving system, and the forces are applied at the displaced location. These contradicting assumptions require formulation of additional matrix terms. The conventional Lagrange integral methods may not be used, but rather direct evaluation of the physical terms must be derived.

A change in force on the structure is illustrated in the sketch below.



The normal force, F_n , required to support the pressure is:

$$\vec{F} = -A[p_i + \rho(\vec{g} \cdot \vec{u}_i)]\vec{n} \quad (14)$$

The term Ap_i is included in the area matrices discussed previously. The second term on the right-hand side of Eq. (14) takes the form of a stiffness. The matrix takes the form:

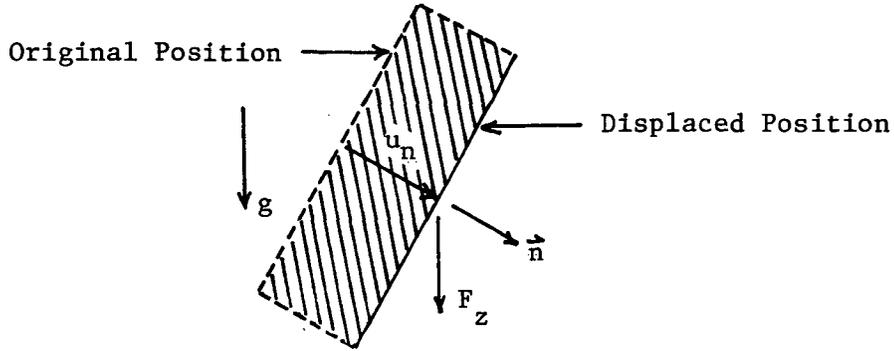
$$\begin{Bmatrix} F_x \\ F_z \end{Bmatrix} = [K] \begin{Bmatrix} u_x \\ u_z \end{Bmatrix} \quad (15)$$

where

$$[K] = -\rho|\vec{g}|A \begin{bmatrix} 0 & n_x \\ 0 & n_z \end{bmatrix} \quad (16)$$

Note that the matrix is not symmetric if $n_x \neq 0$. This violates the fundamental rule that symmetric system matrices must occur for the conservation of energy.

Another method of viewing the problem resolves the non-symmetric issue. If the structure moves, the total fluid weight changes as illustrated below.



The additional weight on the structure, w , due to the motion is:

$$w = \rho g A u_n = \rho g A (\vec{n} \cdot \vec{u}) \quad (17)$$

Since each point, including free surface points, may move independently of the others, the increased vertical force must be applied locally and the force required to support the load is:

$$F_z = - \rho g A (\vec{n} \cdot \vec{u}) \quad (18)$$

The corresponding stiffness matrix is:

$$[K] = - \rho g A \begin{bmatrix} 0 & 0 \\ n_x & n_z \end{bmatrix} \quad (19)$$

Comparing Eqs. (16) and (19), we observe that the lower right-hand terms are equal, but the off-diagonal terms are reversed. The conclusion is that each approach missed an off-diagonal term, and the true result is:

$$[\Delta K_g] = - \rho g A \begin{bmatrix} 0 & n_x \\ n_x & n_z \end{bmatrix} \quad (20)$$

These stiffness terms may be processed along with the fluid-structure area coefficients described previously. The intersecting structure fluid areas are used to define the factor A. The displacements and resulting forces are assumed to be variable on the surface. These integrals are evaluated in a manner similar to those developed in the area matrix calculations.

Free Surface Effects

A free surface is defined as a moving boundary with no restraints. When gravity effects are neglected, the boundary condition, $p = 0$, may be enforced by simply applying single-point constraints (SPC) to the input which causes the rows and columns corresponding to zero pressure to be removed from the matrix equations. However, when gravity is present we must remember that the pressure may not be zero since it is actually measured at a point fixed in space. For an upward displacement, u_f , of the free surface, the pressure at a point defined at the surface is:

$$p = \rho g u_f \quad (21)$$

(For a downward displacement, it is also convenient to use the same equation, measuring a fictitious negative pressure above the surface.)

In the actual solution of the free surface points, it is convenient to implement Eq. (21) in the following form:

$$- Ap + \rho g A u_f = 0 \quad (22)$$

where A is the free surface area associated with the fluid point. The terms in the above equation may be implemented directly into the matrix formulation. In effect, the free surface points are treated as though they were structure points, although no structural stiffness is present. The area factors A are identical to the fluid/structure interface matrices defined previously. The terms $(\rho g A)$ are, in effect, positive springs providing the stiffness terms, $[K_f^g]$, for the normal displacements, u_f , and causing the "sloshing" modes.

Equation (22) may therefore be written in terms of the generalized coordinate vectors in the form:

$$- [A_f]\{p\} + [K_f^g]u_f = 0 \quad (23)$$

System Matrix Solution

The previous development has provided the basic matrix equations to define the fluid, the fluid structure interface, and the free surface. For the general case, when gravity is present, all the previously derived matrices will occur. The desired form of the solution matrices are:

$$[\bar{M}]\{\ddot{u}\} + [\bar{K}]\{u\} = \{F\} \quad (24)$$

where $\{u\}$ is a vector containing both structure and free surface displacements and $\{F\}$ is the applied load vector.

Combining Eqs. (5), (11), and (23), into the form of Eq. 23 we obtain:

$$[\bar{M}] = \begin{bmatrix} M_s & | & 0 \\ \hline 0 & | & 0 \end{bmatrix} + [\bar{A}][K^f]^{-1}[\bar{A}]^T \quad (25)$$

$$[\bar{K}] = \begin{bmatrix} K_s + \Delta K_g & | & 0 \\ \hline 0 & | & K_f^g \end{bmatrix} \quad (26)$$

where

$$[\bar{A}] = \begin{bmatrix} A_s \\ A_f \end{bmatrix}$$

We observe that the matrices $[\bar{M}]$ and $[\bar{K}]$ are symmetric, and may be processed as normal structure matrices.

Unfortunately, the effect of the fluid mass terms in Eq. (25) is to fill the mass matrix, resulting in potentially time-consuming solutions for large structures. However, it is typical for large structures that a reduction procedure is employed. Defined symbolically, this reduction may be defined as:

$$\{u_s\} = [G]\{u_a\} \quad (27)$$

where the vector $\{u_a\}$ is defined by a much smaller number of degrees of freedom than $\{u_s\}$. Components of the vector $\{u_s\}$ are removed by application of constraints through the "Guyan" reduction procedure or through a modal formulation where the columns of $[G]$ are eigenvectors of the empty structure normal modes. The area factor matrix may be treated as a set of load factors in the reduction process.

For the Guyan reduction $[A]$ is reduced using the equation:

$$[A_a] = \left[\begin{array}{c|c} G & 0 \\ \hline 0 & I \end{array} \right]^T [\bar{A}] \quad (28)$$

thus,

$$[A_a] = \left[\begin{array}{c} G^T A_s \\ \hline A_f \end{array} \right] \quad (29)$$

The "reduced" matrices M_{aa} , K_{aa} , A_a , etc. may be used in Eq. 24 and 25 in place of the full size matrices.

Note that as the size of the matrix $[\bar{A}]$ is reduced, the evaluation of the matrices for Eqs. (26) and (27) will be more economical. In the actual formulation, the columns of the matrix $[K_f^g]$ may be treated as load vectors on the structure, and the NASTRAN reduction procedure for the load vectors may be applied directly. The gravity "stiffness" matrix may be reduced in the NASTRAN system with the same algorithm as the mass matrix reduction process.

Treatment of Completely Enclosed Fluids

When the fluid boundary is completely enclosed (by the structure and free surfaces), the incompressible fluid effects must be considered. The incompressible fluid, in effect, provides a steady state constraint on the motions of the boundary. Furthermore, the fluid matrix $[K_f]$ is singular because a constant pressure defines zero flow. Mathematically, a unit pressure vector, defined as $\{I\}$, produces the result:

$$[K_f]\{I\} = \{0\} \quad (30)$$

Since the matrix $[K_f]$ has a singularity of order one, a constraint must be supplied. Because of incompressibility, we know that the total flow must be zero. The basic pressure-flow relationship is:

$$[K]\{p\} = \{Q\} \quad (31)$$

The pressure is obtained by removing one row and column, and solving Equ. 31 in partitioned form:

$$\left[\begin{array}{c|c} K_{11} & K_{1j} \\ \hline K_{j1} & K_{jj} \end{array} \right] \left[\left\{ \begin{array}{c} 0 \\ p_j \end{array} \right\} + \alpha \{I\} \right] = \left\{ \begin{array}{c} Q_1 \\ Q_j \end{array} \right\} \quad (32)$$

After some algebraic steps we obtain the equations:

$$\{Q'_j\} = [H_j]\{Q\} \quad (33)$$

$$\{p\} = [H_j]^T\{p'_j\} \quad (34)$$

and

$$[H_j] = \left[-\frac{1}{N}\{I\} \mid [I] - \frac{1}{N}\{I\}\{I\} \right] \quad (35)$$

The matrix "inverse" may be written symbolically as:

$$[K_f]^{-1} = [H_j]^T [K_{jj}]^{-1} [H_j] \quad (36)$$

Furthermore, it may be proven by examples that $[K_{jj}]$ may be obtained by partitioning any fluid point, p_1 , from the matrix. If the matrix $[K_f]$ is singular (of order 1), the results are exactly the same regardless of the choice.

As described above, the net volume change due to boundary movement is eliminated from the fluid inertia matrix. However, the incompressibility of the fluid requires that the volume change due to structure and free surface displacements be restricted. This constraint could be implemented by supplying a constraint equation of the form:

$$\Delta Vol = \sum_i \sum_j A_{ji} u_j = 0 \quad (37)$$

or, in terms of the matrices:

$$\Delta Vol = [I][A]^T\{u\} = 0 \quad (38)$$

For this approach, one of the displacements, u_j , is removed from the matrices, redistributing its associated mass and stiffness to the other degrees of freedom.

In the alternate method, we add a compressibility factor such that the net volume change will be small. If we define the factor, B , such that for the static case:

$$\{\bar{p}\} = \{I\} B \Delta Vol \quad (39)$$

then the static compressibility may be defined by the stiffness matrix $[K_c]$ where:

$$[K_c] = B[A]\{I\}\{I\}[A]^T \quad (40)$$

This matrix is added to the structure/free surface stiffness matrix and provides an effective approximation to the overall fluid compressibility. It does not account for local compressibility effects or acoustics.

TEST RESULTS

The choice of demonstration problems had to be limited to cases with known results from experimental tests and/or published analyses. Large-order detailed models representing the Space Shuttle External Tanks have also been analyzed by NASA using the program. Results of these tests are forthcoming from NASA. The basic test and demonstration problem analyzed by UAI is described below.

SRI Test Tank

As a test on the performance of the 3-D analysis of a typical problem, a series of analyses were run on a real tank model. This actual model was built and tested by Southwest Research, Inc., and the experimental results are described in Reference 6. Other analytic results were obtained using the DYNASØR axisymmetric program described in Reference 7.

The finite element NASTRAN model is shown in Figure 3. A 15° sector was modeled with one layer of elements and two layers of grid points to solve for the axisymmetric modes. The mesh size was chosen such that when it was extended to a three-dimensional half model (12 sections), the number of degrees of freedom (2900) would result in a reasonable running time.

The effects of nearly all of the available options in the hydroelastic system were evaluated with the SRI model. The results are summarized in Table 1. The error ratios in terms of the test results are given in Table 2. Each of the analysis cases is described below.

NASTRAN - Phase I Program: The first system delivery contained limited options and a crude method of calculating area coefficients. No overlapping structure/fluid elements were allowed. All runs were made using the direct formulation method with no matrix condensation.

Test 1: Model A - Compressible. This model was generated by simply converting the DYNASØR data to the NASTRAN format. The mesh was similar to that shown in Figure 3 except that only four-sided elements were used. The compressibility factor was obtained from the properties of water.

Test 2: Model B - Compressible. This was the basic test case using the model shown in Figure 3 with overall compressibility of water. The second and third modes were excellent but the first mode was suspiciously high.

Test 3: Model B - Incompressible. The incompressible option was used in this model to determine its effect. The first mode became worse but the second and third modes were hardly affected.

Test 4: Model B - 1/6 Compressibility. The compressibility factor was divided by a factor of 6. The first mode frequency became lower than the test results with no change in the second and third modes. This indicated that fluid compressibility had affected the test results.

NASTRAN - Phase II Program: The final delivered program contained more accurate area factor calculations and the complete set of user options. The tests given below were run on this version. For comparison with the preliminary version, Model B with the calculated compressibility was used as the basic model.

Test 5: Direct - Not Reduced. The direct method without matrix condensation was used to compare results with the Phase I program. The results for the more accurate area factor calculations became slightly lower in frequency than for the original area averaging method. It was determined that the structural stiffness was causing low frequencies and that the first method provided an error in the opposite direction. It was decided that the use of a better structural model would be more preferable than trying to compensate for the inaccurate structure with less accurate area factors.

Test 6: Direct - Reduced. In this case the solution matrices were reduced from 257 degrees of freedom to 60 degrees of freedom to represent only shell displacements at every other point. This reduction would be equivalent to reducing the three-dimensional model to 300 degrees of freedom for eigenvalue extraction.

Test 7: Direct - Ignore Gravity. The gravity effects were removed from the problem which reduced the solution size and the running time.

Test 8: Modal - 30 Modes. The modal formulation was used in this problem to reduce the structure matrices to 30 modal coordinates representing the modes of the empty structure. Only three of the 30 empty structure modes participated to any extent in the first mode of the combined fluid and structure systems.

CONCLUDING REMARKS

From the experience of running the test and demonstration problems, several conclusions may be made regarding the NASTRAN hydroelastic system. These are listed below.

- Accuracy of the system was better than expected for the mesh sizes used in the demonstration problems. With only linear elements and averaged area factors representing the fluid, three good slosh modes were obtained from only eight degrees of freedom. It appears that the accuracy for hydroelastic modes is limited more by the existing NASTRAN structure elements than by the fluid formulation. Results indicate that 15° sectors are adequate for a cylindrical or spherical shaped fluid model.
- The results were relatively insensitive to modeling procedures. On each of the problems, different methods of subdividing the fluid space into elements were tested. For similar mesh sizes, the changes in results were insignificant. The use of either Modal Formulation or Guyan reduction to condense the structural degrees of freedom tends to increase the natural frequencies of the system. For the relatively small demonstration problems, their effects on execution cost were small. Since the hydroelastic formulation produces dense solution matrices, large-order problems will require one of these reduction methods.
- Although free-surface gravity effects are necessary to obtain pure sloshing modes, their effect on the hydroelastic modes for most problems is small. The alternate method of constraining the free surface pressures to zero is more efficient and requires less data input. Also, the overall compressibility factor used in the new methods provides a simple, efficient manner of treating enclosed fluids. It will produce more accurate results for very stiff tanks such as those used in the SRI demonstration problem.

REFERENCES

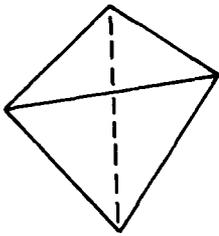
1. R. H. McNeal (Ed.), NASTRAN Theoretical Manual, Chapter 16, NASA SP-222 (01), National Aeronautics and Space Administration, Washington, D.C.
2. Zienkiewics, O.C., The Finite Element Method in Engineering Science. pp. 240-247, McGraw-Hill, New York, 1971.
3. Guyan, R. J., Ujihara, B. H. and Welch, P. W., "Hydroelastic Analysis of Axisymmetric Systems by a Finite Element Method." Proceedings of the 2nd Conference on Matrix Methods in Structural Mechanics, Air Force Flight Dynamics Lab, AFFDL-TR-68-150, 1969.
4. Hsiung, H., "Dynamic Analysis of Hydroelastic Systems Using the Finite Element Method," PhD Dissertation, University of Southern California, Los Angeles, California, 1973.
5. Rush, R. H., "A Modal Analysis of the Fluid Shell Dynamics Interaction Problem," ME Thesis, University of Alabama, Huntsville, 1975.
6. "A Study of Axisymmetric Modes in Thin-Walled Uniform Thickness Hemispherical and Hemispherical-Spherical Tanks Containing Liquid," Final Report, Contract NAS8-30270, Southwest Research, December 1975.
7. "Dynamic Analysis of Shells of Revolution," Document No. D5-14293-1, Boeing Co. Southeast Division, Huntsville, Alabama, February 1971.
8. Cappolino, R.N. "A Numerically Efficient Finite Element Hydroelastic Analysis" NASTRAN User's Experiences" TM X-3428, October 1976.

TABLE 1. COMPARISONS OF FREQUENCIES FOR SRI TEST TANK

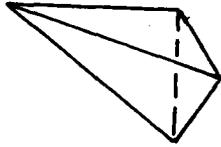
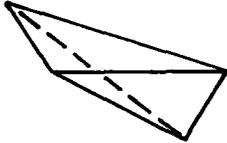
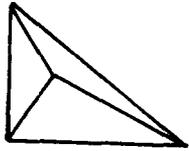
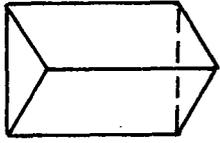
Analysis Case	Mode Frequencies		
	Mode 1	Mode 2	Mode 3
Test Results	495	835	1255
DYNASØR Program	531	807	1179
NASTRAN - Phase 1 Program			
Model A - Comp.	519	822	1239
Model B - Comp.	516	826	1239
Model B - Incomp.	541	828	1240
Model B - 1/6 Comp.	423	821	1234
NASTRAN - Phase 11 Program			
(Model B - Comp.)			
Direct - Not Reduced	513	809	1174
Direct - Reduced	612	914	1279
Direct - Ignore G	539	811	1175
Modal - 30 Modes	568	814	1185

TABLE 2 COMPARISONS OF FREQUENCY ERRORS
FOR SRI TEST TANK

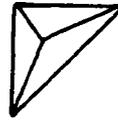
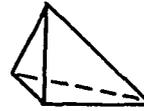
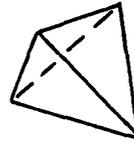
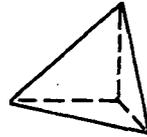
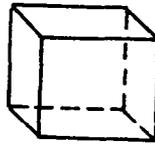
Test No.	Analysis Case	Frequency Difference Ratios (%)		
		Mode 1	Mode 2	Mode 3
	Test Results	0	0	0
	DYNASØR Program	7.3	-3.35	-6.1
	NASTRAN - Phase 1 Program			
1	Model A - Comp.	4.85	-1.56	-1.28
2	Model B - Comp.	4.25	-1.08	-1.28
3	Model B - Incomp.	9.3	-0.83	-1.20
4	Model B - 1/6 Comp.	-12.5	-1.68	-1.67
	NASTRAN - Phase 11 Program			
5	Direct - Not Reduced	3.7	-3.1	-6.5
6	Direct - Reduced	23.6	9.5	1.9
7	Direct - Ignore G	8.9	-2.9	-6.4
8	Modal - 30 Modes	14.8	-2.5	-5.6



(a) Tetrahedron



(b) Wedge and one of its six decompositions



(c) Hexahedron and one of its two decompositions

Figure 1. Three-dimensional fluid elements.

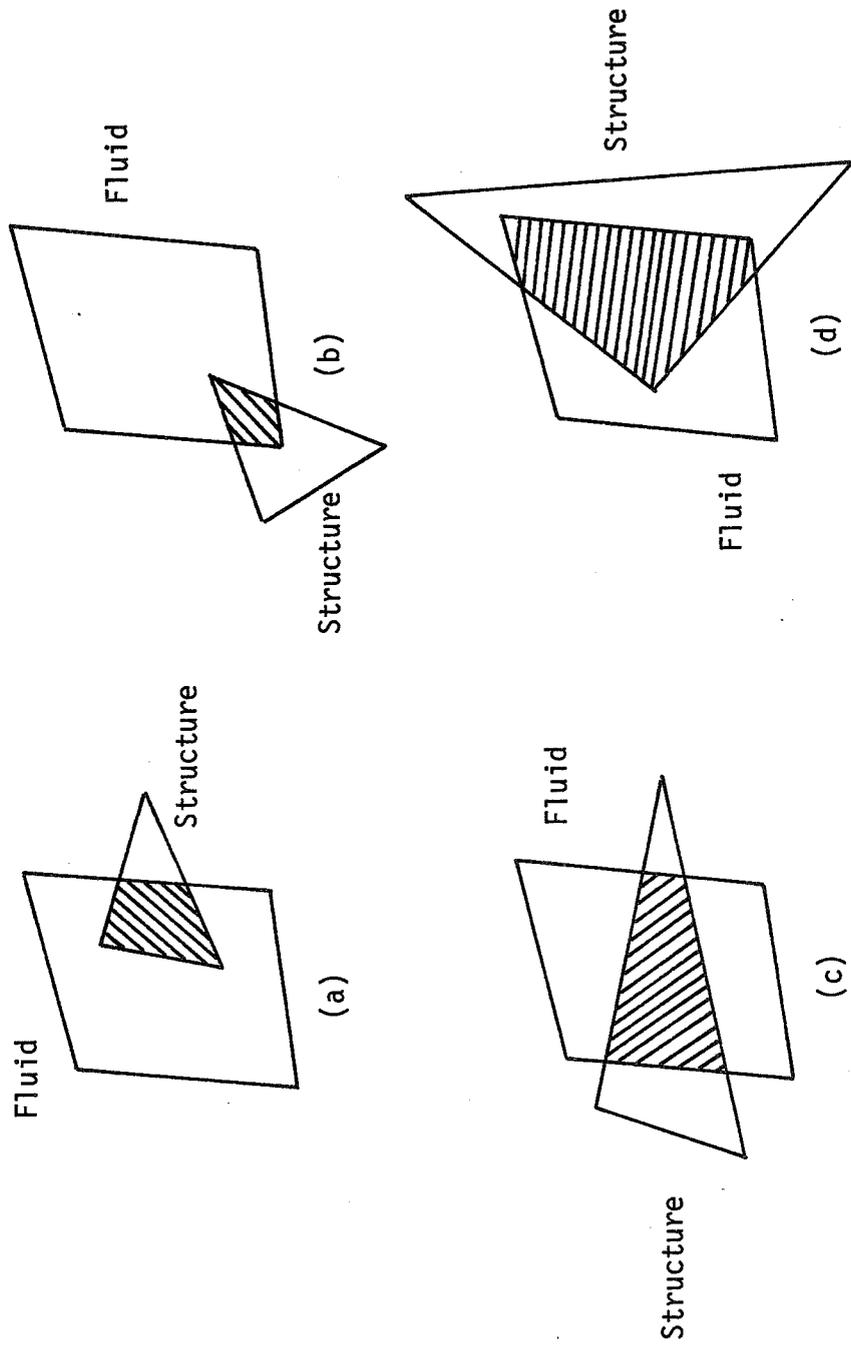


Figure 2. Examples of overlapping fluid and structure element faces.

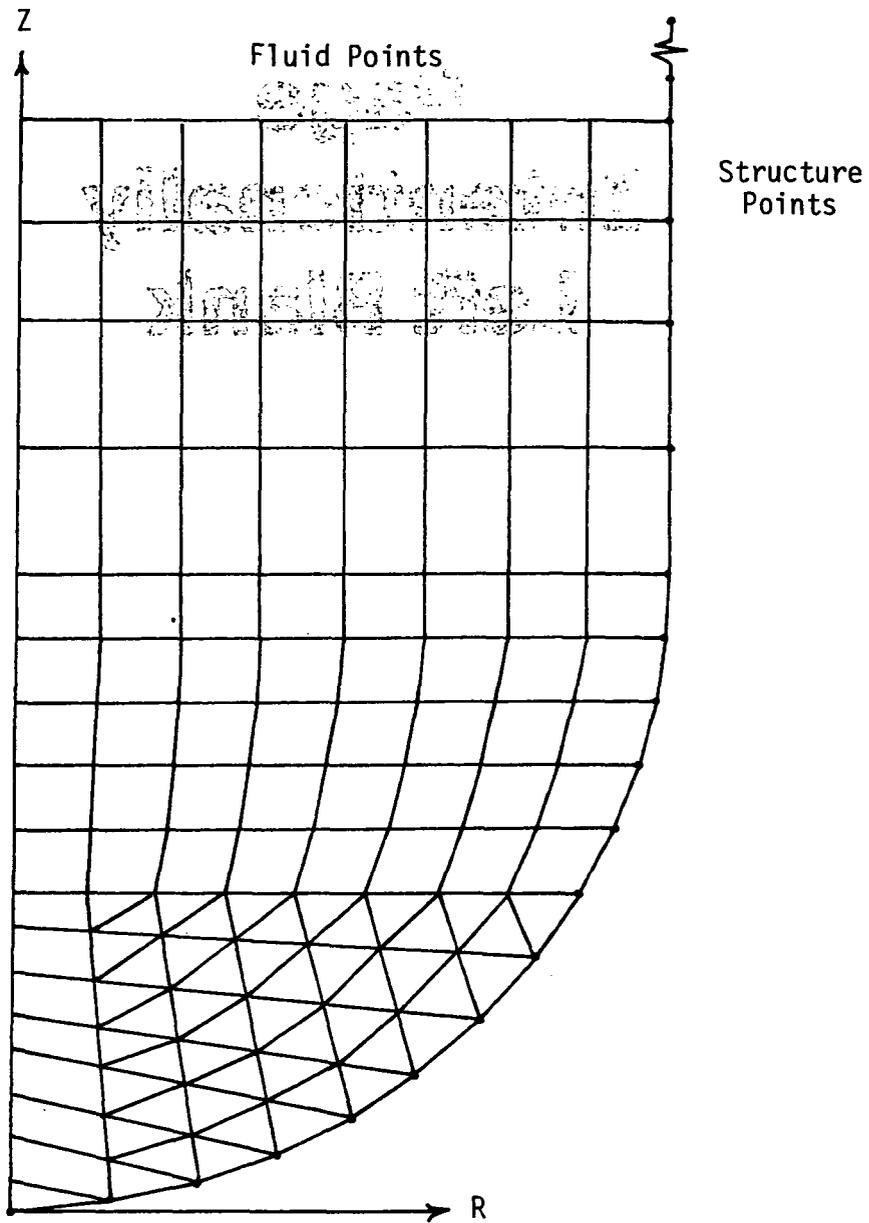


Figure 3. Finite element model of the SRI tank.

**Page
Intentionally
Left Blank**

ON THE THERMO-FLUID ELEMENTS AND THEIR APPLICATIONS

H. P. Lee,
NASA Goddard Space Flight Center

ABSTRACT

Two types of one-dimensional thermo-fluid elements have been developed and added to the NASTRAN Thermal Analyzer (NTA). One is an element of a linearly tapered bore with the constant diameter as a special case. The other, being treated as a nonlinear load, permits the specification of a time-dependent flow rate in a transient problem. Both types are capable of simulating the effect of a flowing fluid in a fluid loop, either in a closed or an open system, to transport thermal energy.

The objective of this paper is to present the basic formulation of these elements including descriptions of pertinent input data cards with emphasis on applications.

Detailed listing of input data cards of demonstration problems explicating modeling techniques and essentials are given. Accuracy of solutions obtained via the NTA are discussed and compared with those based on other numerical methods. Engineering applications of this capability are exemplified by a radiator panel, which simulates a segment of the payload bay door of a space shuttle orbiter, and a solar water-heating system, which consists of a solar collector, an energy storage tank and an associated plumbing system.

Page intentionally left blank

FINITE ELEMENT SOLUTIONS OF FREE SURFACE FLOWS

P. Richard Zarda and Melvyn S. Marcus
David W. Taylor Naval Ship Research and Development Center

SUMMARY

This paper presents a procedure for using NASTRAN to determine the flow field about arbitrarily shaped bodies in the presence of a free surface. The fundamental unknown of the problem is the velocity potential which must satisfy Laplace's equation in the fluid region. Boundary conditions on the free surface may involve second order derivatives in space and time. In cases involving infinite domains either a tractable radiation condition is applied at a truncated boundary or a series expansion is used and matched to the local finite elements. Solutions are presented for harmonic, transient, and steady state problems and compared to either exact solutions or other numerical solutions.

INTRODUCTION

The pressure distribution and flow field about submerged bodies are important in the determination of hydrodynamic variables such as lift and wave resistance and the calculation of boundary-layer characteristics. The investigation of these variables can be realistically modeled by assuming the fluid to be inviscid and incompressible. In this case the equations of motion can be reduced to the solution of Laplace's equation in the fluid region. The linearized free surface condition (small wave amplitude) may involve second derivatives of the velocity potential ϕ in both space and time and considerably complicates the problem. The free surface flows investigated in this paper can be divided into three areas: harmonic, transient, and steady state.

An exhaustive list of literature for forced harmonic motion or diffraction problems may be found in Wehausen (ref. 1). Problems of this type were generally solved by using a distribution of sources or dipoles on the body boundary with an appropriate Green's function for the problem. The boundary condition on the body is used to determine the strength of the source distribution (for example, Hess and Smith, ref. 2). Such solutions are only appropriate for problems of infinite or constant depth.

Bai (refs. 3-6) uses finite elements to model both harmonic and steady state problems of arbitrary geometry. Similar methods which employ variational functionals have been used by Berkhoff (ref. 7) and Chen and Mei (ref. 8). For steady state problems Bai developed a localized finite element method (ref. 6) in which finite elements are used in a localized region around the body and a series expansion is used in the remainder. The finite element representation

is matched to the series expansion along the common boundary to form a consistent set of equations for the nodal potentials and series coefficients.

Finite elements solutions for transient free surface flows are given by Visser and van der Wilt (ref. 9). Unfortunately, for the transient problem there seems to be no suitable method to construct a completely absorbing boundary in cases involving radiation conditions. For that reason truncated boundaries are taken far enough away so as not to affect the region of interest.

It is the purpose of this paper to present a procedure, using NASTRAN, to model the three types of problems described above using finite elements. The procedure described is presented for either 2-D or axisymmetric problems but is readily extendable to 3-D problems using the existing 3-D capability within NASTRAN.

FREE SURFACE EQUATIONS

For an inviscid, incompressible fluid in an irrotational flow field, the equations of motion and continuity reduce to

$$\nabla^2 \phi = 0 \quad (1)$$

where ϕ is the velocity potential (ref. 10). The pressure p in the fluid can be determined from Bernoulli's equation,

$$-\frac{p}{\rho} = \frac{\partial \phi}{\partial t} + \frac{1}{2} \left[\left(\frac{\partial \phi}{\partial x} \right)^2 + \left(\frac{\partial \phi}{\partial y} \right)^2 \right] + gy \quad (2)$$

where ρ is the density of the fluid and g is the gravitational constant. In Fig. 1 the deflection of the free surface η is assumed to be small compared to the depth d . In that case the linearized conditions on the free surface are (ref. 10)

$$\frac{\partial \eta}{\partial t} = \frac{\partial \phi}{\partial y} \quad \text{on } y=0 \quad (3)$$

and

$$\frac{\partial \phi}{\partial t} = -\frac{p}{\rho} - g\eta \quad \text{on } y=0 \quad (4)$$

The surface elevation η may be eliminated from Eqs. (3) and (4) at the cost of increasing the order of the time derivatives by one. This gives

$$\frac{\partial^2 \phi}{\partial t^2} = -\frac{1}{\rho} \frac{\partial p}{\partial t} - g \frac{\partial \phi}{\partial y} \quad \text{on } y=0 \quad (5)$$

Once the potential ϕ is determined, the surface elevation η may be determined

from Eq. (4).

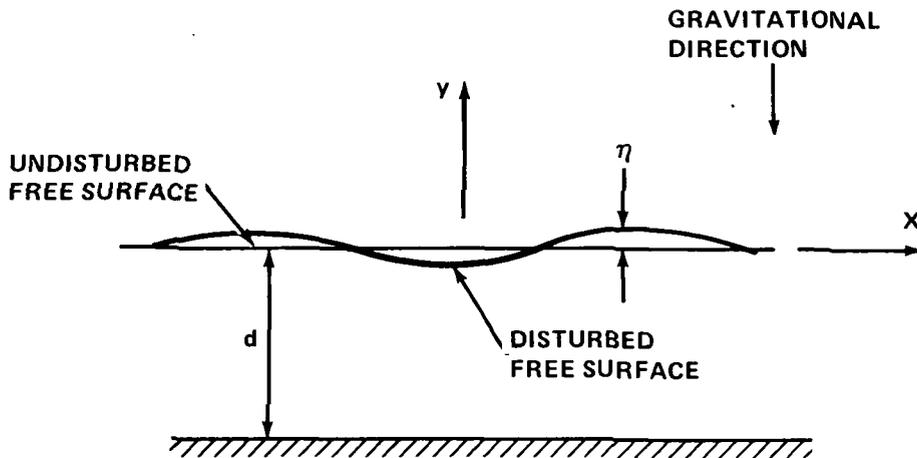


Figure 1. Free Surface Wave

HARMONIC FREE SURFACE PROBLEMS

2-D Wave Maker

Consider the 2-D wave maker shown in Fig. 2. At $x=0$, a wall is oscillating in simple harmonic motion with velocity V . For the harmonic problems, assume

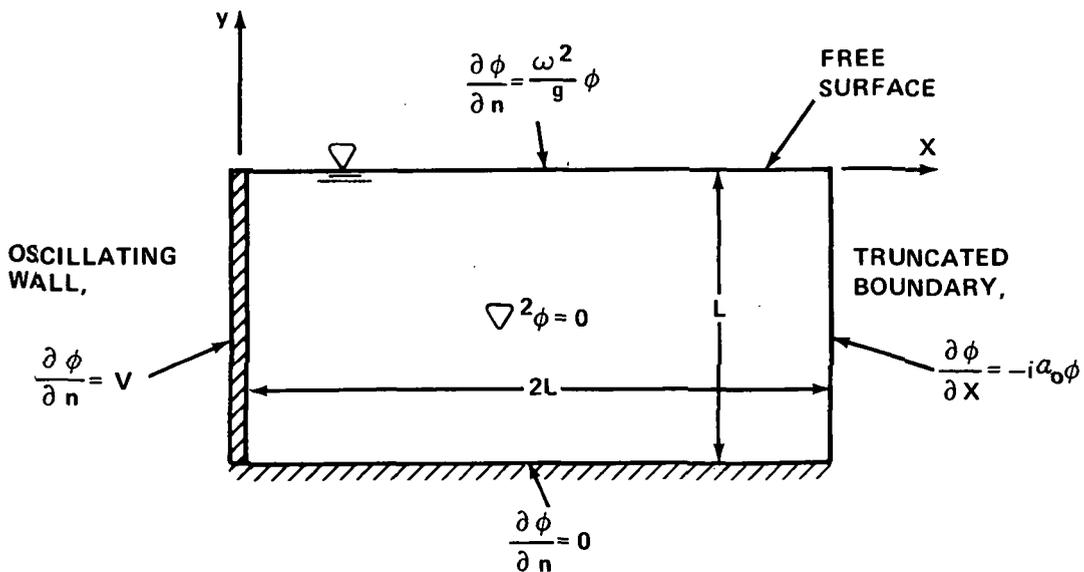


Figure 2. Geometry and Boundary Conditions for 2-D Wave Maker

$$\phi(x,y,t) = \phi(x,y) e^{i\omega t} \quad (6)$$

and

$$p = 0 \quad \text{on} \quad y = 0 \quad (7)$$

Then Eqs. (6) and (1) give

$$\nabla^2 \phi = 0 \quad (8)$$

Using Eqs. (5), (6), and (7), the free surface condition becomes

$$\frac{\partial \phi}{\partial y} = \frac{\omega^2}{g} \phi \quad \text{on} \quad y = 0 \quad (9)$$

At the wall,

$$\frac{\partial \phi}{\partial n} = V \quad \text{on} \quad x = 0 \quad (10)$$

and, along the bottom,

$$\frac{\partial \phi}{\partial n} = 0 \quad \text{on} \quad y = -d = -L \quad (11)$$

The solution of this problem can be obtained by separation of variables (see Bai, ref. 3) and is given by

$$\phi(x,y) = A_0 \cosh \alpha_0(y+d) e^{-\alpha_0 i x} + \sum_{N=1}^{\infty} A_N \cos \alpha_N(y+d) e^{-\alpha_N x} \quad (12)$$

where

$$\frac{\omega^2}{g} = \alpha_0 \tanh(\alpha_0 d) \quad (13)$$

$$\frac{\omega^2}{g} = -\alpha_N \tan(\alpha_N d) \quad \text{for all } N \quad (14)$$

$$A_0 = \frac{-4i}{\sinh(2\alpha_0 d) + 2\alpha_0 d} \frac{\sinh(\alpha_0 d)}{\alpha_0} \quad (15)$$

$$A_N = \frac{4}{\sin(2\alpha_N d) + 2\alpha_N d} \frac{\sin(\alpha_N d)}{\alpha_N} \quad (16)$$

The first term of Eq. (12) represents a traveling wave in the x-direction, while the succeeding terms are local terms that are only significant for small x. Thus

$$\frac{\partial \phi}{\partial x} = -i \alpha_0 \phi \quad (17)$$

for large x . Eq. (17) is a tractable radiation condition which can be applied at suitable boundary far enough away from the wall. Eq. (8), together with boundary conditions given by Eqs. (9), (10), (11), and (17), constitute a well-posed problem for Laplace's equation.

The boundary conditions Eqs. (9), (10), (11), and (17) all have the form

$$\frac{\partial \phi}{\partial n} + \gamma \phi = \beta \quad (18)$$

where γ and β are constants. The functional form for Laplace's equation with the mixed boundary condition of Eq. (18) is

$$F(\phi) = \frac{1}{2} \iint_A \left\{ \left(\frac{\partial \phi}{\partial x} \right)^2 + \left(\frac{\partial \phi}{\partial y} \right)^2 \right\} dA + \int_B \left(\frac{1}{2} \gamma \phi^2 - \beta \phi \right) ds \quad (19)$$

where B is the boundary of region A . When variations are taken with respect to ϕ such that

$$\delta F = 0 \quad (20)$$

then Eqs. (8) and (18) are satisfied.

Eqs. (19) and (20) can be approximated with finite elements using NASTRAN structural elements. A procedure for using structural elements to model fluid domains which satisfy the wave equation (or, as a special case, Laplace's equation) is given by Everstine et al (ref. 11), and has been successfully applied using NASTRAN on several problems by Schroeder and Marcus (ref. 12), Marcus (ref. 13), and Everstine (ref. 14). A translational degree of freedom (in this case the x displacement) is chosen to represent the potential ϕ , and all other degrees of freedom at a node are permanently constrained. The linear isoparametric membrane element, QDMEM1 (NASTRAN Level 16), was used. The material matrix \underline{G} and the mass density ρ_e of the QDMEM1 elements are chosen as follows:

$$\underline{G} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \rho_e = 0 \quad (21)$$

NASTRAN's Rigid Format 8, with governing equation given by

$$(-\omega^2 \underline{M} + i \omega \underline{B} + \underline{K}) \underline{\phi} = \underline{F}(\omega) \quad (22)$$

is chosen as the analysis method. The stiffness matrix \underline{K} generated by the QDMEM1 elements with material properties given by Eq. (21) is equivalent to the finite element representation of the first term in Eq. (19).

The free surface condition, Eq. (9), corresponds to $\gamma = \omega^2/g$ and $\beta = 0$ in the second term of Eq. (19). A consistent formulation for this term is implemented using NASTRAN by inserting the matrix

$$(M2PP)_{i,j} = \frac{\Delta x}{6g} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad \begin{matrix} i=k,\ell \\ j=k,\ell \end{matrix} \quad (23)$$

into the mass matrix \underline{M} in Eq. (22) using DMIG data cards. In Eq. (23), k and ℓ represent the two nodes which lie on the free surface for each of the QDMEM1 surface elements, while Δx is the spacing between nodes k and ℓ . The frequency ω is inserted into Eq. (22) using a FREQ data card.

The radiation condition, Eq. (17), corresponds to $\beta = 0$ and $\gamma = i\alpha_0$ in Eq. (19). A consistent formulation is obtained by inserting the matrix

$$(M2PP)_{i,j} = -i \frac{\alpha_0}{\omega^2} \frac{\Delta y}{6} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \quad \begin{matrix} i=k,\ell \\ j=k,\ell \end{matrix} \quad (24)$$

into the mass matrix \underline{M} in Eq. (22) using DMIG cards. In Eq. (24) k , ℓ , and Δy are defined as in Eq. (23) except that in this case the relevant boundary surface is the truncated boundary.

The bottom condition, Eq. (11), is a natural boundary condition which is automatically satisfied within the finite element approximation. The boundary condition at the wall, Eq. (10), is implemented by inserting the vector

$$F_i = V \Delta y \begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix}, \quad i=k,\ell \quad (25)$$

into the forcing function $\underline{F}(\omega)$ in Eq. (22) using DAREA data cards. The relevant boundary for the quantities k , ℓ , and Δy in Eq. (25) is the oscillating wall.

The above procedure was used to compute the fluid response for the oscillating wall problem illustrated in Fig. 2. All data is presented in non-dimensionalized form using the length L and the velocity V . Results are shown in Fig. 3 for dimensionless spacing $\Delta x = \Delta y = .0625$ which corresponds to approximately 10 nodes per wave length for the linear elements. In Fig. 3, the amplitude of the surface elevations are plotted. The NASTRAN solutions obtained by both consistent and lumped formulations, as well as the analytic solution, are presented. The lumped formulation is determined by using diagonalized matrices in Eqs. (23) and (24) where diagonal terms are determined by adding together all terms in the corresponding row. The consistent formulation is a significant improvement over the lumped formulation. In subsequent problems only a consistent formulation will be used.

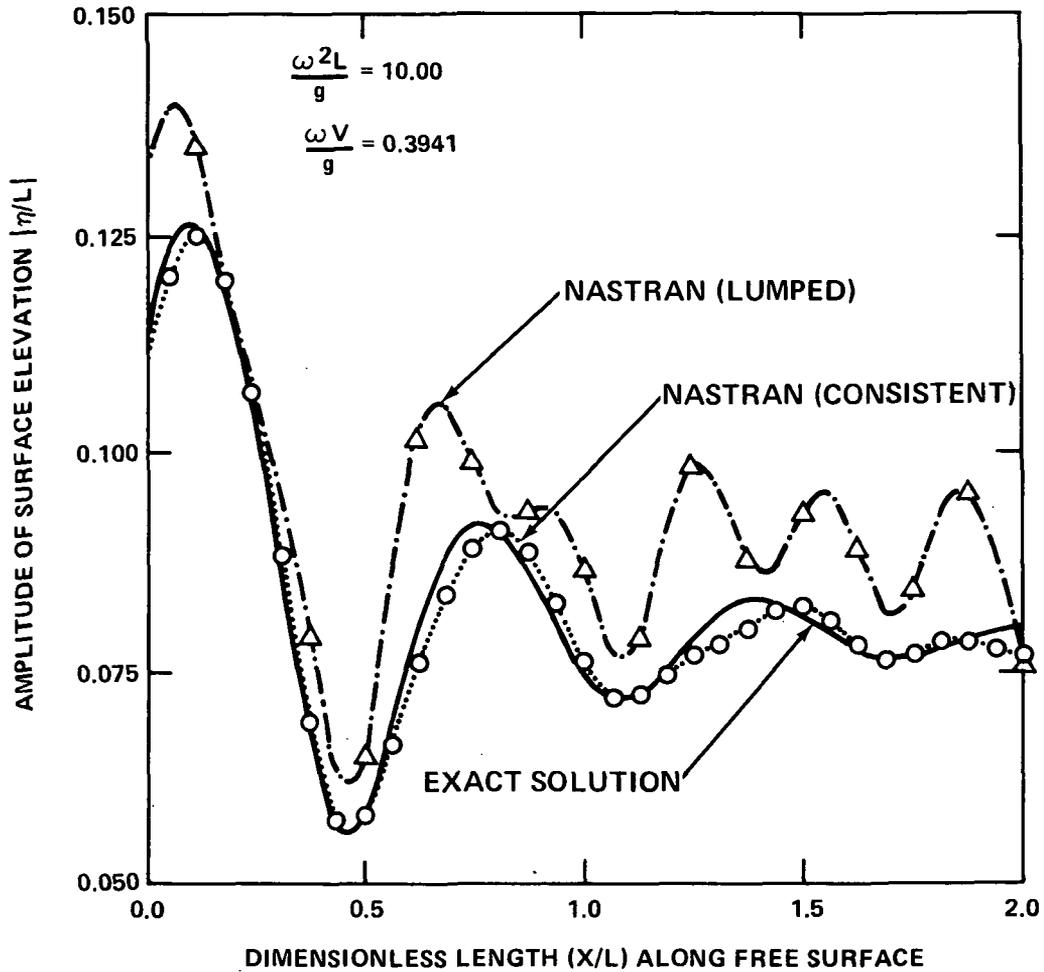


Figure 3. Amplitude of Surface Elevation for the 2-D Wave Maker

Axisymmetric Wave Maker

The geometry and boundary conditions for the axisymmetric wave maker is shown in Fig. 4. Boundary conditions are the same as the 2-D wave maker except for the additional term in the radiation condition. The radiation condition is determined by investigating the exact solution (see Bai, ref. 3):

$$\phi(r, z) = B_0 H_0(\alpha_0 r) \cosh \alpha_0(z+d) + \sum_{N=1}^{\infty} B_N H_0(-\alpha_N i r) \cos \alpha_N(z+d) \quad (26)$$

where

$$B_0 = \frac{4 \sinh(\alpha_0 d)}{H_1(\alpha_0 r_0) \alpha_0 \{ \sinh(\alpha_0 d) + 2\alpha_0 d \}} \quad (27)$$

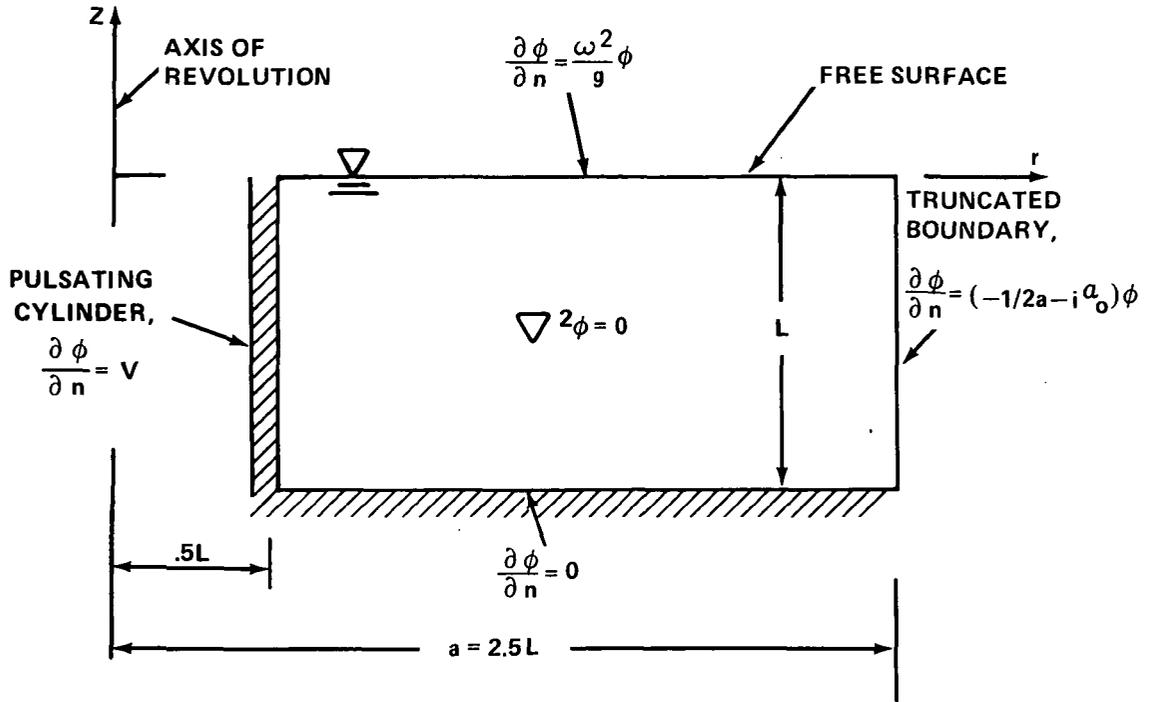


Figure 4. Geometry and Boundary Conditions for Pulsating Cylinder

$$B_N = \frac{4i \sin(\alpha_N d)}{H_1(-\alpha_N i r_0) \alpha_N \{\sin \alpha_N d + 2 \alpha_N d\}} \quad (28)$$

and where α_0 and α_N are given by Eqs. (13) and (14), r_0 is the inner radius of the cylinder, and H_0 , H_1 are Hankel functions of the second kind of order 0 and 1, respectively. The first term of Eq. (26) is an outgoing wave and the second terms represent local disturbances. Thus it can be shown that

$$\frac{\partial \phi}{\partial r} = - \left\{ \frac{1}{2a} + i \alpha_0 \right\} \phi \quad \text{for large } r \quad (29)$$

where a is defined in Fig. 4.

This problem was modeled using NASTRAN's Rigid Format 8. CTRAPRG elements were used (Everstine, ref. 14) with dimensionless spacing given by $\Delta x = \Delta y = .0625$ (all variables are non-dimensionalized with respect to V and L). This corresponds to approximately 10 nodes per wave length. Results showing the amplitude of the surface elevation along the free surface are presented in Fig. 5. These results are based on applying consistent boundary conditions, and are in good agreement with the series solution.

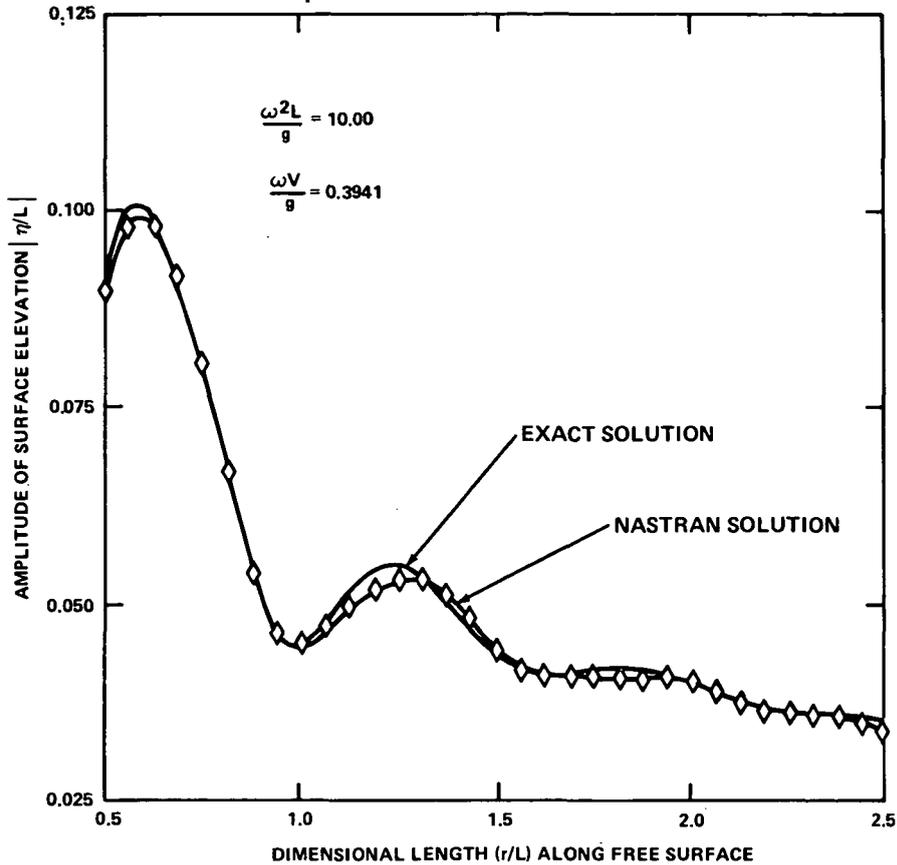


Figure 5. Amplitude of Surface Elevation for the Pulsating Cylinder

Refraction Problems

A surface wave, given by

$$\eta(x,t) = A e^{i(\omega t - \alpha_0 x)} \quad (30)$$

is incident upon the bottom obstacle shown in Fig. 6. The potential ϕ_I corresponding to the incident wave is given by

$$\phi_I(x,y) = \frac{Ag i}{\omega} \frac{\cosh \alpha_0 (y+d)}{\cosh(\alpha_0 d)} e^{-i \alpha_0 x} \quad (31)$$

where ω , α_0 , g and d satisfy Eq. (13). In order to determine the total potential ϕ of the fluid corresponding to the incident wave, the potential ϕ is divided into

$$\phi = \phi_I + \phi_R \quad (32)$$

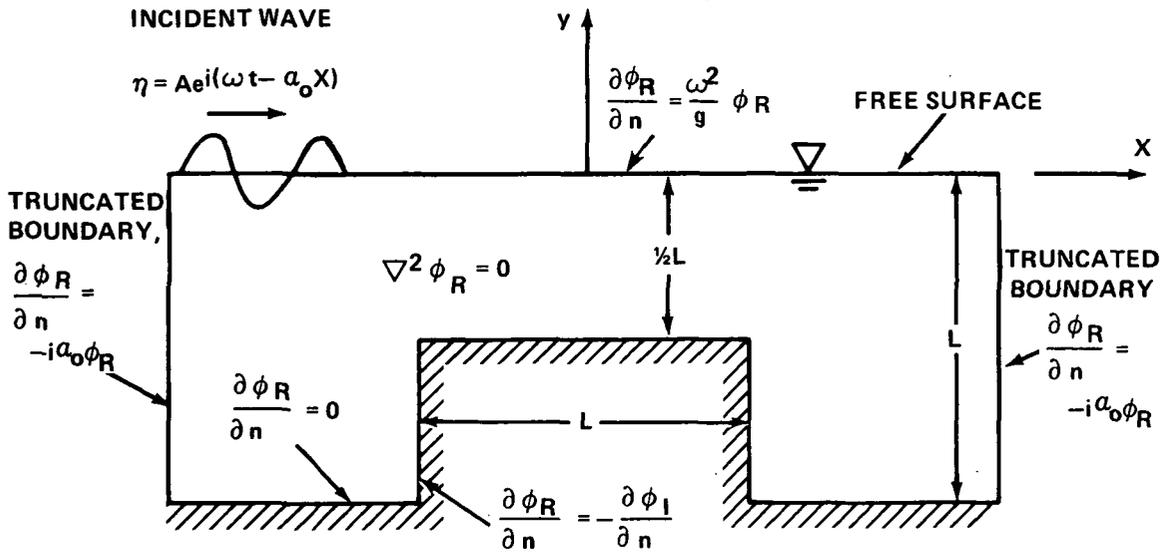


Figure 6. Geometry and Boundary Conditions for Refracted Waves Due to a Bottom Obstacle

where ϕ_R is the refracted potential. The boundary conditions and governing equations on ϕ_R are shown in Fig. 6. All variables are non-dimensionalized with respect to the length L and frequency ω , and boundary conditions are specified in a consistent formulation.

The NASTRAN results shown in Figs. 7 and 8 are presented for dimensionless spacing $\Delta x = .125$ and $\Delta y = .0625$ which corresponds to approximately 41 nodes per wave length. These results compare favorably with the finite element solution recently re-computed by Bai as a correction to his originally published (ref. 5) results. Accuracies within 4% have also been obtained using coarser grids of 10-20 nodes per wave length.

A similar free surface problem is illustrated in Fig. 9. The dimensionless spacing used was $\Delta x = \Delta y = .125$ which corresponds to approximately 42 nodes per wave length. Again, the NASTRAN results shown in Figs. 10 and 11 compare well with the finite element solution recently re-computed by Bai (ref. 5).

TRANSIENT PROBLEMS

Consider the transient free surface problem shown in Fig. 12 illustrating the time dependent pressure distribution on the free surface. The pressure distribution is given by

$$p(x,t) = P(x) \sin \omega t \quad (33)$$

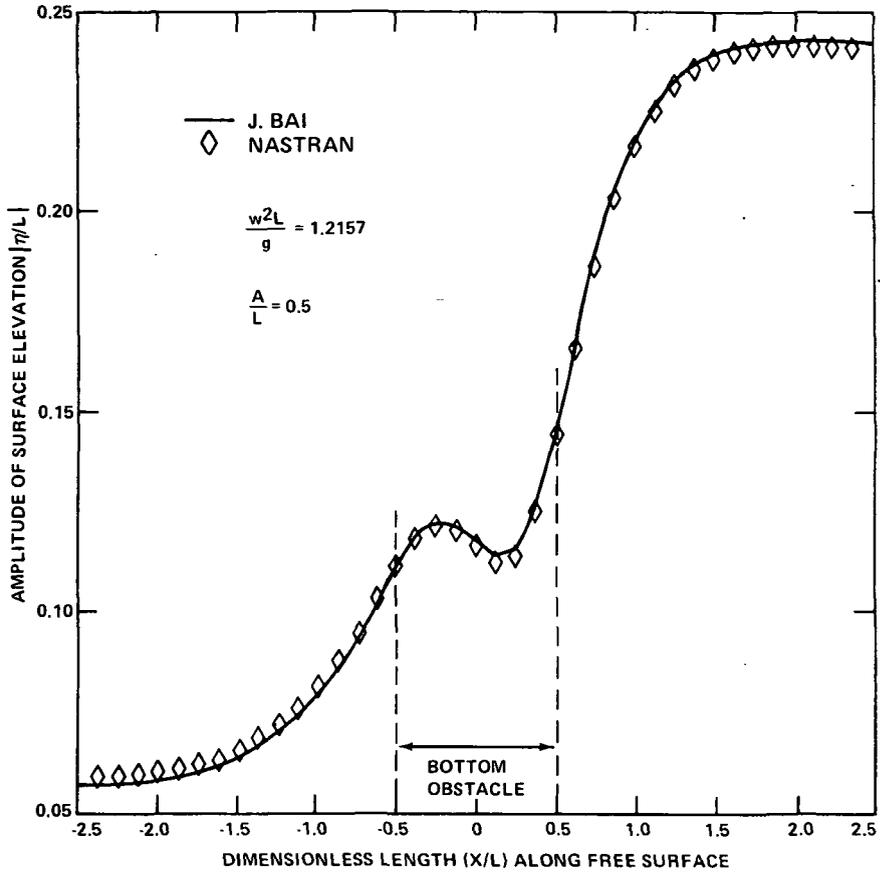


Figure 7. Amplitude of Refracted Waves for the Bottom Obstacle

where

$$P(x) = \begin{cases} P_0 & 0 \leq x \leq 0.3 \\ \frac{P_0}{2} \left[1 - \sin\left(\frac{\pi(x-0.5)}{0.4}\right) \right] & 0.3 \leq x \leq 0.7 \\ 0 & x \geq 0.7 \end{cases} \quad (34)$$

and where P_0 is the maximum pressure.

Initial conditions which could be specified are on the free surface are

$$\frac{\partial \eta}{\partial t} = f_1(x) \quad y = 0, t = 0 \quad (35)$$

and

$$\eta = f_2(x) \quad y = 0, t = 0 \quad (36)$$

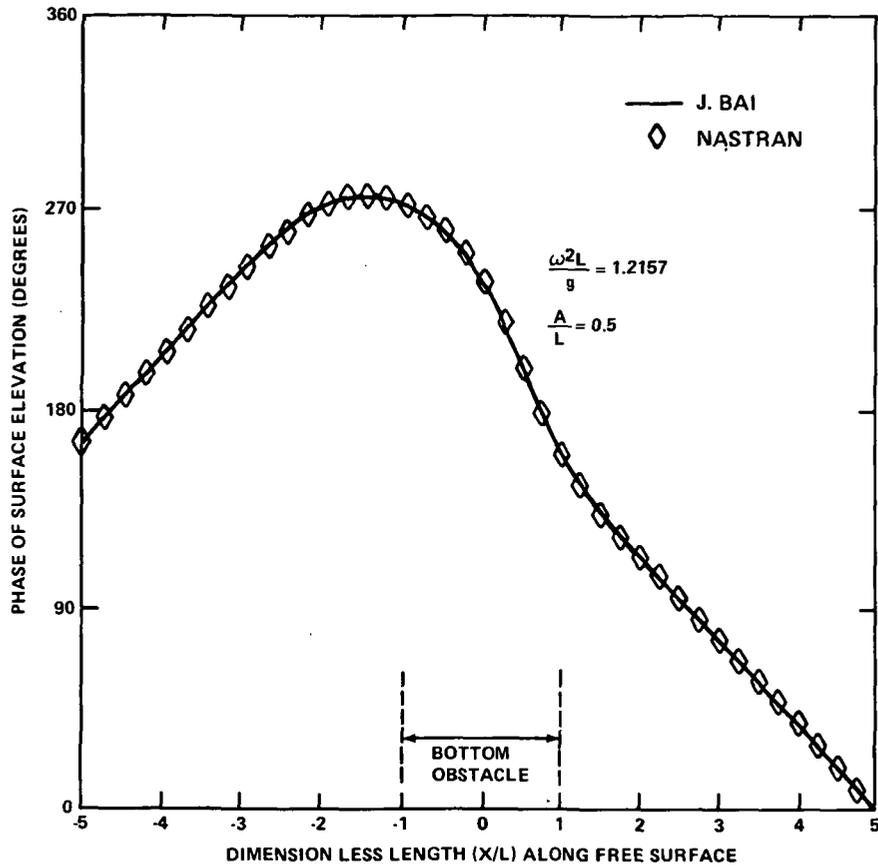


Figure 8. Phase of Refracted Waves for the Bottom Obstacle

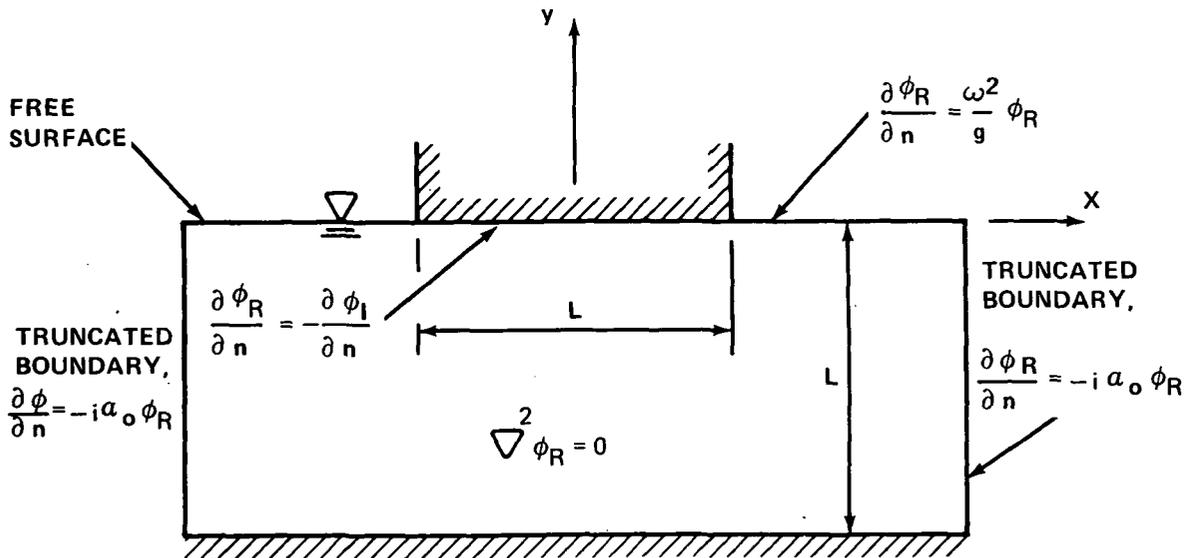


Figure 9. Geometry and Boundary Conditions for Refracted Waves Due to a Surface Obstacle

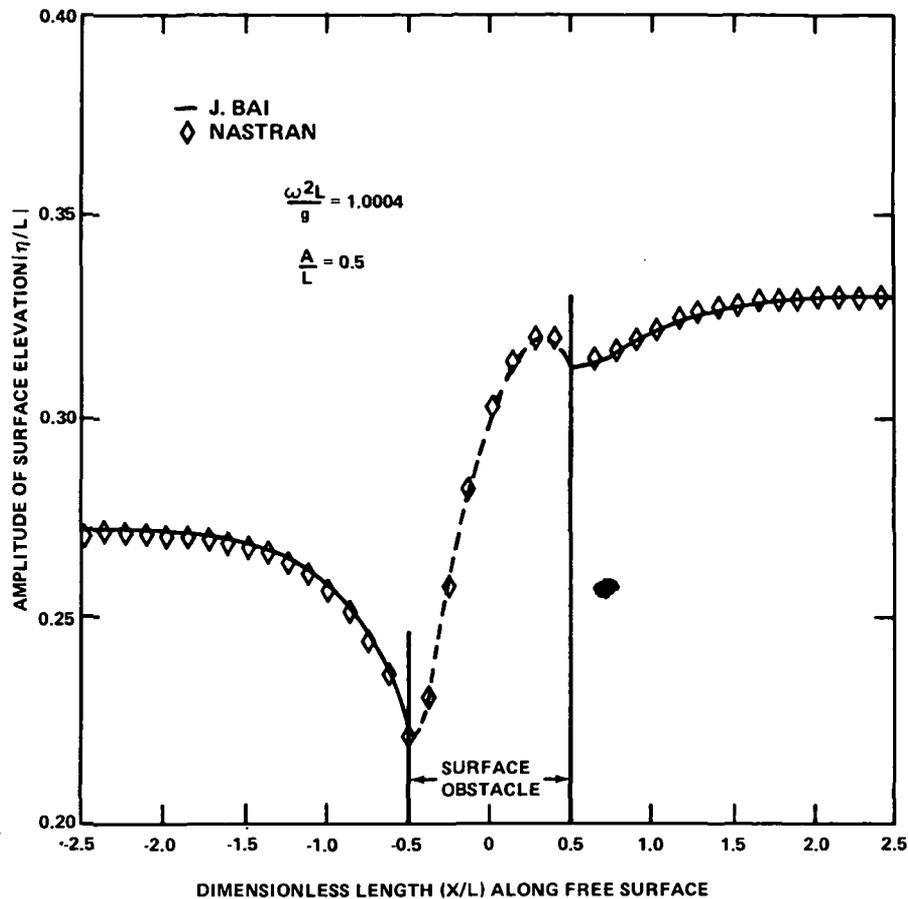


Figure 10. Amplitude of Refracted Waves for the Surface Obstacle

Eqs. (35) and (36) must be put in terms of ϕ and $\partial\phi/\partial t$ since this is the only suitable input to NASTRAN. Specifying $\partial\eta/\partial t$ on $y=0$ is equivalent by Eq. (3) to specifying $\partial\phi/\partial y$ on $y=0$. Then Laplace's equation may be solved with the boundary conditions shown in Fig. 12, except that $\partial\phi/\partial y$ is specified on the free surface. This will determine ϕ everywhere initially. Similarly, specifying η on $y=0$ is equivalent by Eq. (4) to specifying $\partial\phi/\partial t$ on $y=0$. Then the procedure just described may be repeated to determine $\partial\phi/\partial t$ everywhere initially, since $\partial\phi/\partial t$ also satisfies Laplace's equation and the boundary conditions shown in Fig. 12 (not including the free surface condition). This determines ϕ and $\partial\phi/\partial t$ everywhere initially.

The variational form for the free surface problem shown in Fig. 12, based on Hamilton's principle (see Courant and Hilbert, ref. 15), is

$$\begin{aligned}
 F(\phi) = & \frac{1}{2} \int_0^{t_1} \int_A \left\{ \left(\frac{\partial\phi}{\partial x} \right)^2 + \left(\frac{\partial\phi}{\partial y} \right)^2 \right\} dA dt + \int_0^{t_1} \int_B \left(\frac{1}{2} \gamma \phi^2 + \beta \phi \right) ds dt + \int_0^{t_1} \int_{\text{Free Surface}} \frac{1}{2g} \left(\frac{\partial\phi}{\partial t} \right)^2 dx dt \\
 & + \int_0^{t_1} \int_{\text{Free Surface}} \frac{1}{\rho g} \frac{\partial p}{\partial t} \phi dx dt \quad (37)
 \end{aligned}$$

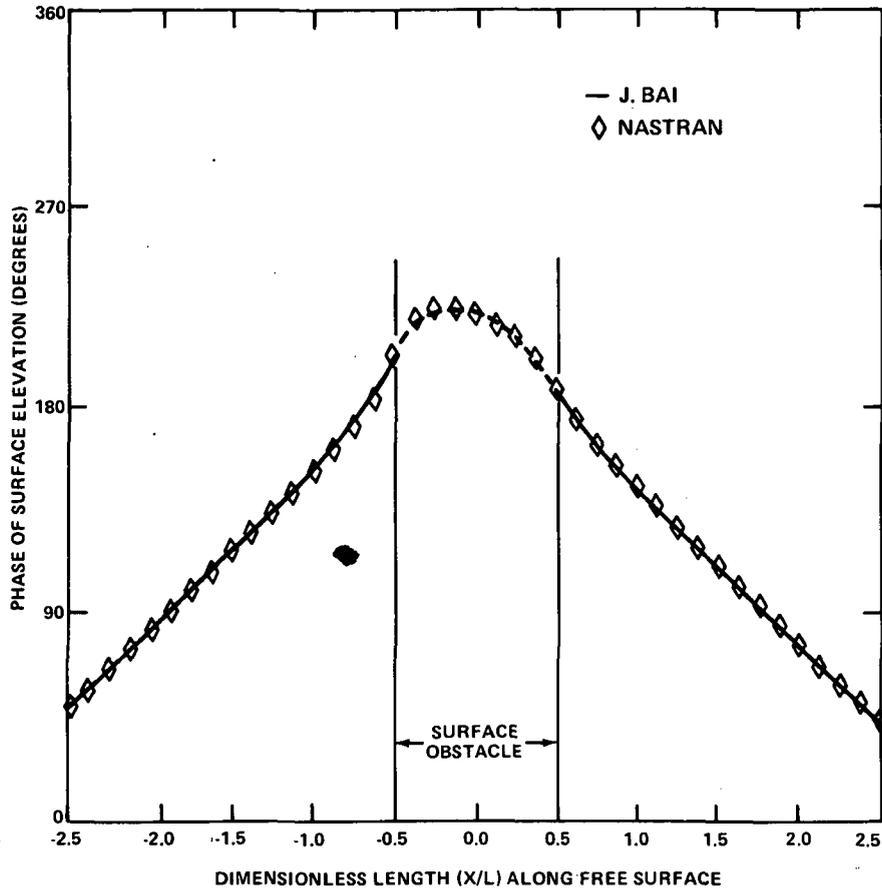


Figure 11. Phase of Refracted Waves for the Surface Obstacle

where B is the boundary of the region A , and all geometric boundary conditions are enforced. If variations of $F(\phi)$ taken with respect to ϕ equal zero, Eqs. (1), (5), (18), (35), and (36) are satisfied for zero initial conditions ($f_1=f_2=0$ in Eqs. (35) and (36)). Non-zero initial conditions can be easily incorporated into Eq. (37).

The finite element representation based on Eq. (37) was implemented using NASTRAN by modeling the fluid with QDMEM1 elements where material properties are given by Eq. (21). Any translational degree of freedom can be used to correspond to ϕ , but all remaining degrees of freedom are permanently constrained. The analysis method chosen is NASTRAN's Rigid Format 9, with the governing equation given by

$$\underline{M} \ddot{\underline{\phi}} + \underline{B} \dot{\underline{\phi}} + \underline{K} \underline{\phi} = \underline{F}(t) \quad (38)$$

The stiffness matrix \underline{K} generated by the QDMEM1 elements is equivalent to the finite element representation of the first term of Eq. (37).

The last two terms of Eq. (37) represent the free surface condition and may be incorporated into NASTRAN as follows: Let ϕ for any point on the free

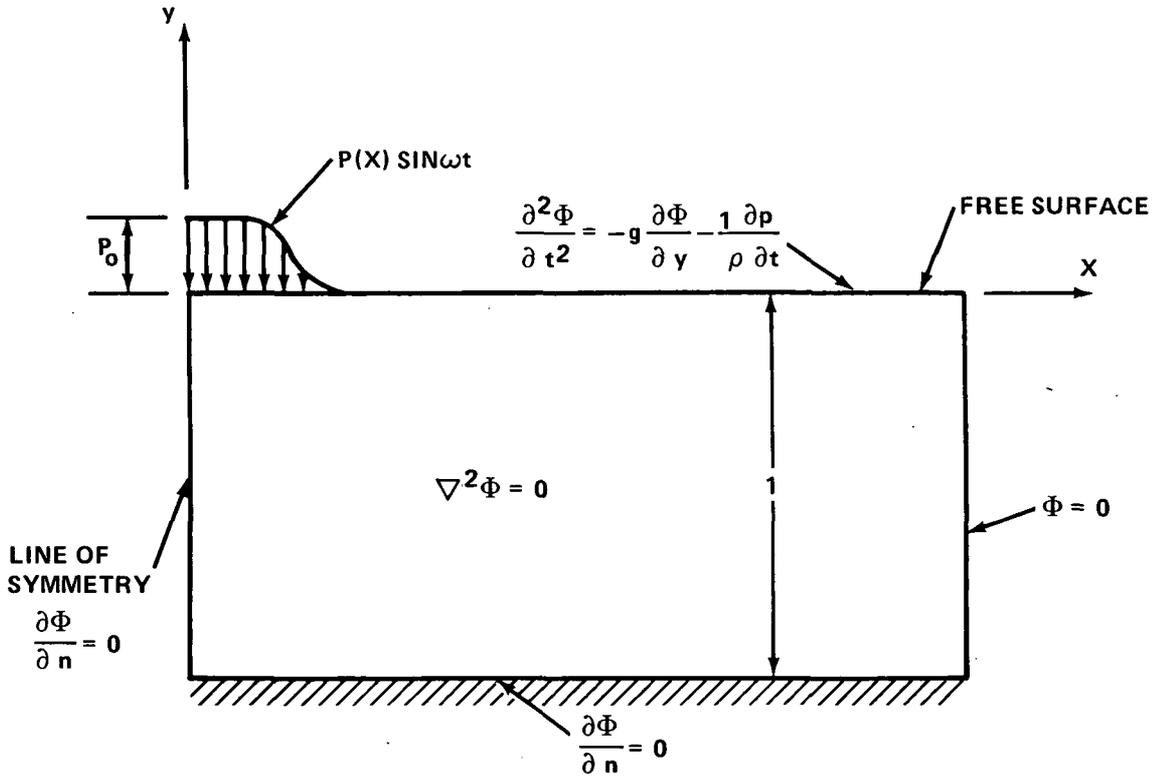


Figure 12. Stationary Pressure Distribution Oscillating on Free Surface

surface be given by

$$\Phi = \sum N_i \phi_i \quad (39)$$

where N_i is the shape function for node i and ϕ_i is the nodal potential. Then the finite element formulation for the third term of Eq. (37) is implemented using NASTRAN by inserting the matrix

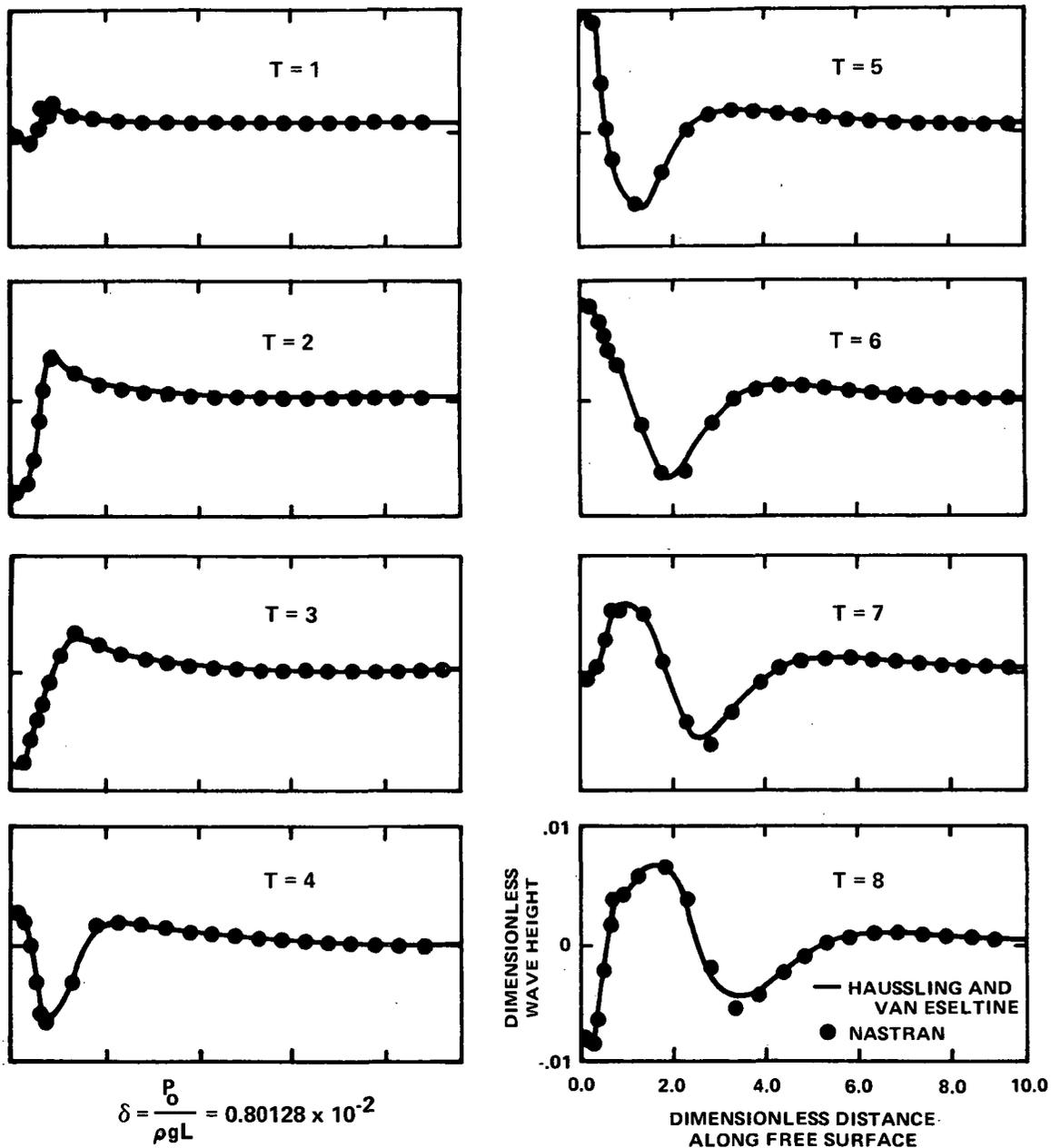
$$(M2PP)_{ij} = \frac{1}{g} \int_{\text{Free Surface}} N_i N_j dx \quad (40)$$

into the mass matrix M in Eq. (38). The finite element representation of the last term of Eq. (37) is implemented using NASTRAN by inserting the vector

$$F_i = - \int_{\text{Free Surface}} \frac{1}{\rho g} \frac{\partial p}{\partial t} N_i dx \quad (41)$$

into the forcing function $\underline{F}(t)$ in Eq. (38).

Referring to Fig. 12, the natural boundary condition $\partial\phi/\partial n = 0$ (corresponding to $\gamma = \beta = 0$) on the bottom and left face are automatically



$$\delta = \frac{P_0}{\rho g L} = 0.80128 \times 10^{-2}$$

$$\frac{\omega^2 L}{g} = 0.99379$$

Figure 13. Development of Surface Elevations for the Pressure Distribution Problem

satisfied. The geometric boundary condition $\phi = 0$ is implemented by constraining $\phi_i = 0$ at all nodes i on the downstream boundary.

The above procedure was used to model the geometry and boundary conditions shown in Fig. 12. All variables have been put in dimensionless form using the pressure P_0 , the depth L , and the gravitational constant g .

This procedure was used to model the geometry and boundary conditions shown in Fig. 12. QDMEM1 elements were used with dimensionless spacing $\Delta x = 0.1$ to 0.5 , and $\Delta y = 0.25$; this would correspond to approximately 13 to 60 nodes per wave length, where the wave length is based on the steady state problem. A dimensionless time step of $\Delta t = .1$ was used. Rules of thumb for estimating spacing and time steps are given by Visser and van der Wilt (ref. 9). In this case approximately 60 time steps per period of the forcing function were used.

In Fig. 13, the NASTRAN results are compared to a Fourier series solution given by Haussling and Van Eseltine (ref. 16). The wave heights are in good agreement with the series solution and illustrate the capability of NASTRAN to model transient water wave problems.

STEADY STATE PROBLEMS

Consider the steady state problem shown in Fig. 14 where a cylinder of diameter L is moving at constant velocity U below the free surface. Steady state solutions are sought for which all variables are independent of time when referenced to a coordinate system moving with the body, that is, the x - y coordinate system shown in Fig. 14. In this coordinate system it can be shown that the potential ϕ must satisfy Laplace's equation, and the free surface condition expressed in Eq. (6) becomes (with $p=0$ on free surface)

$$\frac{\partial \phi}{\partial y} = - \frac{U^2}{g} \frac{\partial^2 \phi}{\partial x^2} \quad (42)$$

The boundary condition on the rigid cylinder shown in Fig. 14 is

$$\frac{\partial \phi}{\partial n} = - U \cos \theta \quad (43)$$

where θ is the angle between the x -direction and the normal to the body directed out of the fluid. No upstream waves are allowed and the Froude number,

$$F = \sqrt{\frac{U^2}{gL}}$$

is such that downstream waves are allowed (see Bai, ref. 6). Considerable effort was devoted to developing tractable radiation conditions for the upstream and downstream boundaries, resulting in the conclusion that none were possible. For this reason a series expansion is used in the regions beyond the upstream and downstream truncated boundaries and matched (at these boundaries) to the finite element solution. This technique was developed and successfully applied by Bai for both steady state problems (ref. 6) and frequency response problems (ref. 4). A similar finite element-series expansion technique for an acoustical fluid has been implemented using NASTRAN by Zarda (ref. 17).

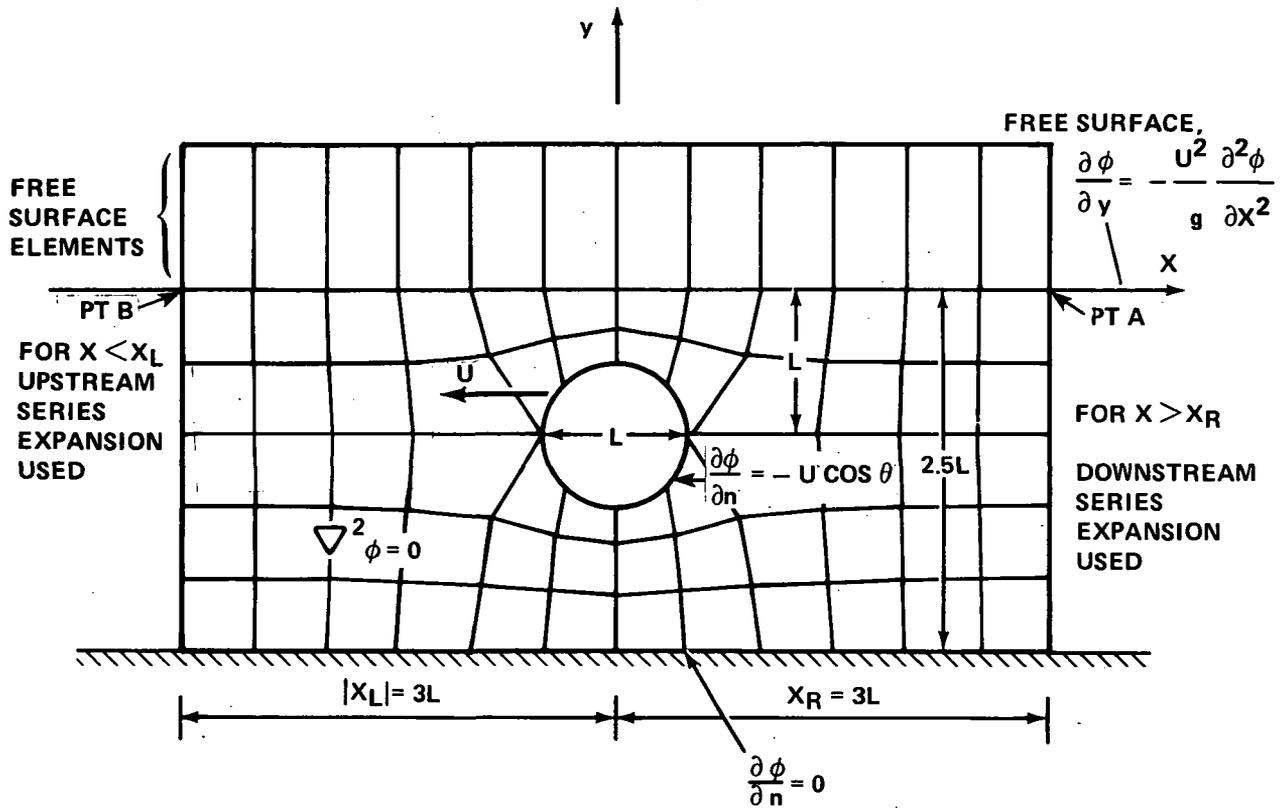


Figure 14. Finite Element Model for Cylinder Moving Below the Free Surface - Coarse Grid

It can be shown, using separation of variables, that downstream from the body

$$\phi = \sum_{j=1}^{N+3} A_j f_j \quad (44)$$

where

$$f_j = \begin{cases} \cos \alpha_j (y+d) e^{\pm \alpha_j x} & 1 \leq j \leq N \\ 1 & j = N+1 \\ \frac{\cosh \alpha_0 (y+d)}{\cosh \alpha_0 d} \cos \alpha_0 x & j = N+2 \\ \frac{\cosh \alpha_0 (y+d)}{\cosh \alpha_0 d} \sin \alpha_0 x & j = N+3 \end{cases} \quad (45)$$

and

$$\frac{U^2}{g} \alpha_0 = \tanh \alpha_0 d, \quad \frac{U^2}{g} \alpha_j = \tan \alpha_j d$$

Upstream from the body,

$$\phi = \sum_{j=1}^{N+1} B_j f_j \quad (46)$$

The sign in the exponential is (+) for the upstream boundary, and (-) for the downstream boundary. Furthermore, N is the number of series terms chosen (the same number is assumed upstream and downstream, although this is not necessary), and d is the depth. Eqs. (44) and (46) satisfy Laplace's equation and the boundary conditions on $y=0$ and $y=-d$. The first N terms represent local terms that decay away from the cylinder, and the last two terms in Eq. (44) represent an outgoing downstream wave; no such waves are allowed in the upstream expansion.

Consider the variational functional given by

$$F(\phi) = \frac{1}{2} \int_A \left\{ \left(\frac{\partial \phi}{\partial x} \right)^2 + \left(\frac{\partial \phi}{\partial y} \right)^2 \right\} dA - \frac{U^2}{2g} \int_{x_A}^{x_B} \left(\frac{\partial \phi}{\partial x} \right)^2 dx + \int_{\text{Body}} U \cos \phi ds$$

Free Surface (47)

$$- \int_{x=x_L}^{\text{Upstream}} \left[\frac{\partial \phi}{\partial n} \right]_{x=-x_L} \phi dy - \int_{x=x_R}^{\text{Downstream}} \left[\frac{\partial \phi}{\partial n} \right]_{x=x_R} \phi dy - \frac{U^2}{g} \left[\frac{\partial \phi}{\partial x} \right]_{x=x_R} \phi_B + \frac{U^2}{g} \left[\frac{\partial \phi}{\partial x} \right]_{x=x_L} \phi_A$$

$y=0$ $y=0$

where points A and B and boundaries x_L and x_R are defined in Fig. 14, and n is the normal to the boundary directed out of the fluid. If independent variations of F with respect to ϕ , ϕ_A and ϕ_B are set equal to zero, then Laplace's equation and the boundary conditions shown in Fig. 14 are satisfied, and $\partial \phi / \partial n$ is continuous on the upstream and downstream boundaries. No variations of the bracketed expressions in Eq. (47) are allowed, and these expressions can be evaluated in terms of the series coefficients by taking the appropriate derivatives using Eqs. (44) and (46). This will increase the number of unknowns by the number (2N+4) of series coefficients. The corresponding additional equations come from requiring that the potential ϕ is continuous at the upstream and downstream boundaries. Let the finite element representation at the truncated boundaries be given by

$$\phi = \sum_{i=1}^{NN} N_i \phi_i \quad (48)$$

where NN is the number of nodes on the truncated boundary. Then, for continuity of ϕ on the downstream boundary, it is required that

$$\sum_{i=1}^{NN} N_i \phi_i = \sum_{k=1}^{N+3} A_k f_k \quad \text{on } x=x_R \quad (49)$$

and on the upstream boundary

$$\sum_{i=1}^{NN} N_i \phi_i = \sum_{k=1}^{N+1} B_k f_k \quad \text{on } x= x_L \quad (50)$$

Eq. (49) is multiplied by f_j , $j=1$ to $N+2$, and integrated from $-d$ to 0 . This gives a system of equations

$$\sum_{i=1}^{NN} G_{ij} \phi_i = \sum_{k=1}^{N+3} H_{jk} A_k \quad \text{on } x=x_R \quad (51)$$

$j= 1 \text{ to } N+2$

where

$$G_{ij} = \int_{-d}^0 N_i f_j dy \quad \begin{array}{l} i = 1 \text{ to } NN \\ j = 1 \text{ to } N+3 \end{array} \quad (52)$$

and

$$H_{jk} = \int_{-d}^0 f_j f_k dy \quad j, k = 1 \text{ to } N+3 \quad (53)$$

Eqs. (51) are $N+2$ equations involving the $N+3$ unknowns A_k . Multiplying Eq. (49) by f_{N+3} and integrating from $-d$ to 0 does not determine an independent equation since f_{N+2} is proportional to f_{N+3} for fixed x .

Multiplying Eq. (50) by f_j , $j=1$ to $N+2$, and integrating from $-d$ to 0 gives

$$\sum_{i=1}^{NN} G_{ij} \phi_i = \sum_{k=1}^{N+1} H_{jk} B_k \quad \begin{array}{l} x = x_L \\ j = 1 \text{ to } N+2 \end{array} \quad (54)$$

Eqs. (54) are $N+2$ equations in the $N+1$ unknowns B_k . The additional equation, determined by multiplying Eq. (50) by f_{N+2} , correspond to the condition that no upstream waves are allowed (see Bai, ref. 6). Eqs. (51) and (54) give the additional $2N+4$ equations involving the $2N+4$ unknowns A_j and B_j .

The procedure just described can be modeled using NASTRAN. CIS2D8 elements are used to model the fluid (see refs. 18 and 19). These second order isoparametric elements with the material properties given by Eq. (21) determine a stiffness matrix equivalent to the finite element representation of the first term of Eq. (47).

The second term of Eq. (47) is modeled using additional CIS2D8 elements along the free surface as shown in Fig. 14. For these elements, the height in the y -direction is unity, and all nodes having the same value of x are constrained to move together. This is equivalent to having 1-D isoparametric

elements along the free surface. The material properties for these elements are given by Eq. (21) except that the material matrix \underline{G} is multiplied by the constant factor $(-U^2/g)$.

The third term of Eq. (47) represents a loading term. It is implemented using NASTRAN by entering

$$F_i = - \int_{\text{Body}} U \cos \theta N_i ds \quad (55)$$

as nodal forces, where N_i is the shape function for node i on the body.

The fourth and fifth terms of Eq. (47) represent coupling terms at the upstream and downstream boundaries. Using Eqs. (44) and (46) to determine the normal derivatives, the finite element modeling yields, for the downstream boundary,

$$(K2PP)_{i,j} = \sum_{j=1}^{N+3} \int_{-d}^0 - \frac{\partial f_j}{\partial x} \Big|_{x=x_R} N_i dy \quad \begin{array}{l} i = 1 \text{ to } NN \\ j = 1 \text{ to } N+3 \end{array} \quad (56)$$

where the matrix K2PP is added to the stiffness matrix. In order to implement this condition, $N+3$ scalar unknowns A_j are created using SPOINT data cards. Then the matrix term $(K2PP)_{i,j}$ in Eq. (56) refers to node i on the downstream boundary and to the SPOINT representation of the unknown A_j . Similarly, for the upstream boundary

$$(K2PP)_{i,j} = \sum_{j=1}^{N+1} \int_{-d}^0 \frac{\partial f_j}{\partial x} \Big|_{x=x_L} N_i dy \quad \begin{array}{l} i = 1 \text{ to } NN \\ j = 1 \text{ to } N+1 \end{array} \quad (57)$$

For the last two terms in Eq. (47), the finite element representation yields

$$(K2PP)_{B,j} = - \frac{U^2}{g} \frac{\partial f_j}{\partial x} \Big|_{x=x_R} \quad j = 1 \text{ to } N+3 \quad (58)$$

$$(K2PP)_{A,j} = \frac{U^2}{g} \frac{\partial f_j}{\partial x} \Big|_{x=x_L} \quad j = 1 \text{ to } N+1 \quad (59)$$

Eqs. (51), (54), and (56) through (59) are entered into NASTRAN using DMIG cards and complete the set of equations to solve for the nodal potentials and the upstream and downstream series coefficients. NASTRAN's Rigid Format 1 (Static Analysis) does not accept DMIG cards. Therefore, Rigid Format 9 was used for one time step. (Since no mass or damping matrix exists, static equilibrium is reached for any time step.)

Computations, with all quantities being non-dimensionalized with respect to the cylinder diameter L , velocity U , and fluid density ρ , were carried out using NASTRAN for the grids shown in Figs. 14 and 15. Each mesh has approximately the same number of unknowns since the series solution is used for $|x| \geq 3.0$ on the coarse grid and for $|x| \geq 1.5$ on the fine grid. Approximately 9 and 17 nodes per wave length were used for the coarse and fine grids, respectively.

Wave height along the free surface is plotted in Fig. 16. Results for both the coarse and fine NASTRAN grids are seen to compare favorably with a solution obtained by Giesing and Smith (ref. 20) using a distribution of sources. The solutions shown here all satisfy the condition that no upstream waves are allowed. (In this case, since the Froude number based on the depth is less than one, downstream waves are generated.)

The pressure distribution on the cylinder may be determined from Bernoulli's equation. Assuming the flow about the cylinder is steady, then, in the x - y coordinate system that is moving with the body, Eq. (2) becomes

$$p = -\rho U \frac{\partial \phi}{\partial x} - \frac{1}{2} \rho \left\{ \left(\frac{\partial \phi}{\partial x} \right)^2 + \left(\frac{\partial \phi}{\partial y} \right)^2 \right\} \quad (60)$$

Fig. 17 illustrates a plot of the dimensionless pressure as a function of the x coordinate on the surface of the cylinder. Results are shown for both the fine and coarse grids shown in Figs. 15 and 16. The discontinuities of the curves occur at element junctures on the cylinder. Although the potential ϕ is necessarily continuous, $\partial\phi/\partial x$ and $\partial\phi/\partial y$ are not necessarily continuous within the finite element approximation, and discontinuities in these terms are magnified in determining the pressure in Eq. (60). Also shown in Fig. 17 is a table showing computed values of the wave resistance and lift coefficients, C_D and C_L , defined by

$$(\rho U^2 L) C_D = - \int_{\text{Body}} p \, dy \quad (61)$$

$$(\rho U^2 L) C_L = \int_{\text{Body}} p \, dx \quad (62)$$

The values of C_D and C_L computed using NASTRAN compare favorably with those given by Giesing and Smith (ref. 20).

CONCLUSIONS

The problems illustrated here demonstrate the capability of NASTRAN to successfully model linearized free surface flow problems for harmonic, transient, and steady state cases. Although the results presented here are for arbitrary 2-D and axisymmetric geometries, the procedures described are directly applicable to 3-D flow problems and readily extendable to the coupled problem of fluid flow about an elastic body.

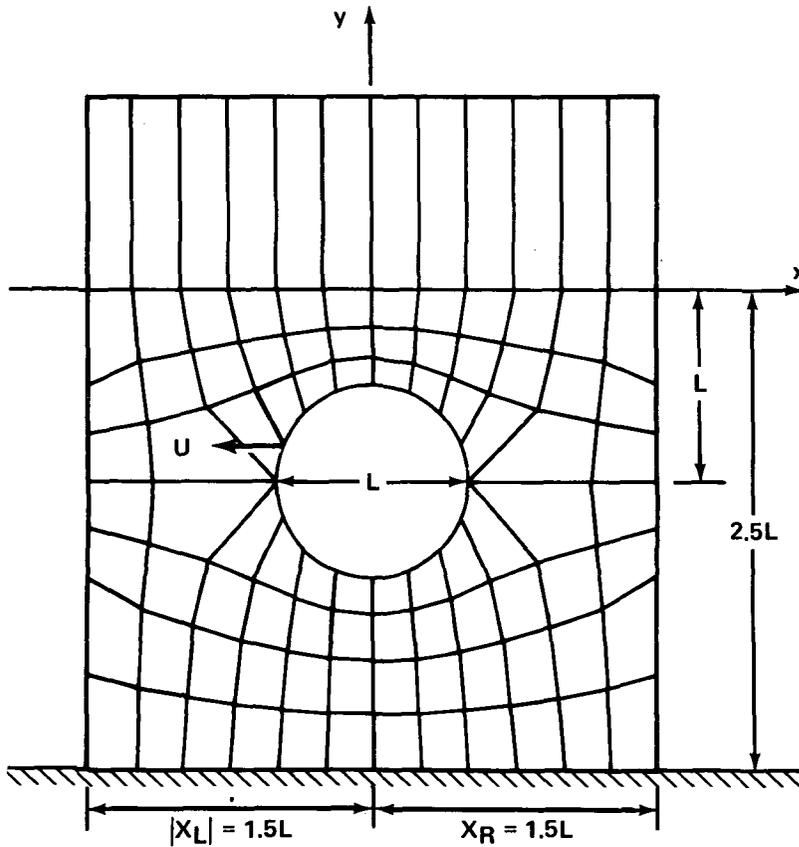


Figure 15. Finite Element Model for Cylinder Moving Below the Free Surface - Fine Grid

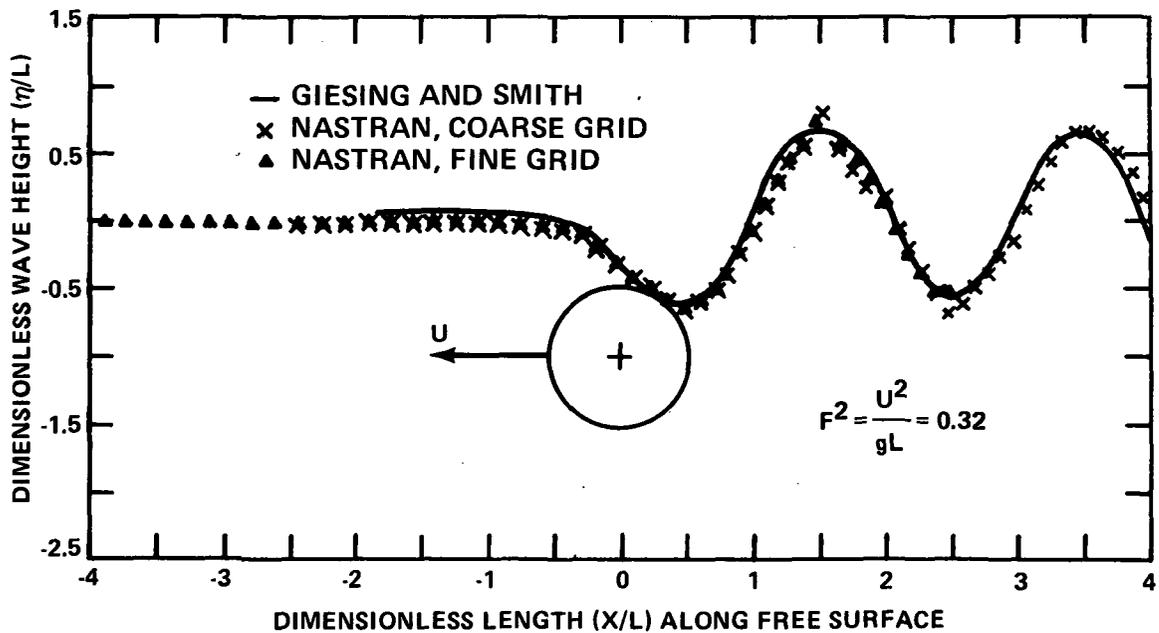


Figure 16. Wave Height Along the Free Surface for Moving Cylinder

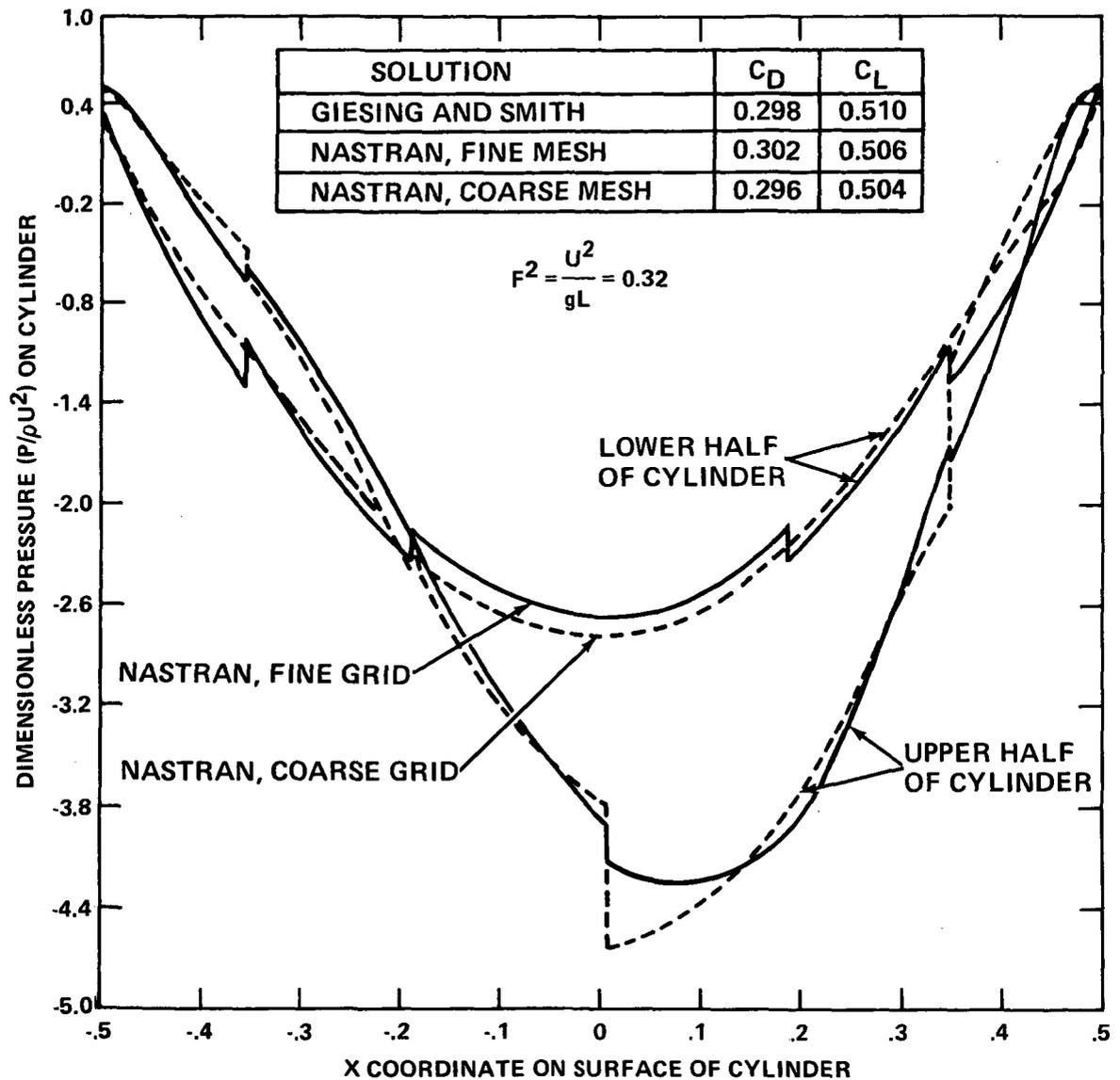


Figure 17. Pressure Distribution on the Cylinder

The steady-state flow due to a cylinder moving below the free surface was computed using the technique of coupling finite elements with a classical method at an appropriate common boundary. Finite elements are used to model irregular geometry over to some specified regular boundary, and classical solution methods are used beyond this boundary. The coupling of the series solutions to the finite element model may be regarded as determining a stiffness matrix for a "classical finite element." Such "elements", if available in the libraries of finite element computer codes, would broaden the range of problems efficiently handled using finite elements. Furthermore, the enhancement of the NASTRAN capability described here may be used to investigate the coupled

problem of fluid flow about an elastic body near or on a free surface. In such a case both the structure and surrounding fluid would be modeled using existing NASTRAN elements and would be coupled at the fluid-structure interface.

REFERENCES

1. Wehausen, J.V.: The Motion of Floating Bodies. Annual Review of Fluid Mechanics, vol. 3, 1971, pp. 237-268.
2. Hess, J.L. and Smith, A.M.O.: Calculation of Potential Flow About Arbitrary Bodies. Progress in Aeronautical Sciences, vol. 8, 1967, pp. 1-137.
3. Bai, K.J.: A Variational Method in Potential Flows With a Free Surface. College of Engineering, Univ. of Calif., Report No. NA 72-2, 1972.
4. Bai, K.J. and Yeung, R.W.: Numerical Solutions to Free-Surface Flow Problems. Tenth Symposium on Naval Hydrodynamics, 1974. Cooper, R.D. and Doroff, S.W., editors.
5. Bai, K.J.: Diffraction of Oblique Waves by an Infinite Cylinder. J. Fluid Mechanics, vol. 68, part 3, 1975, pp. 513-535.
6. Bai, K.J.: A Localized Finite-Element Method for Steady, Two-Dimensional Free-Surface Flow Problems. Proceedings of the First International Conference on Numerical Ship Hydrodynamics. Schot, J.W. and Salvesen, N., editors, October 1975, pp. 209-229.
7. Berkhoff, J.C.W.: Computation of Combined Refraction-Diffraction. Proceedings 13th Coastal Engineering Conference, 2, ASCE, 1972, pp. 471-490.
8. Chen, H.S. and Mei, C.C.: Oscillations and Wave Forces in a Man-Made Harbor in the Open Sea. The 10th Symposium on Naval Hydrodynamics, Office of Naval Research, 1974.
9. Visser, W. and van der Wilt, M.: A Numerical Approach to the Study of Irregular Ship Motions. Finite Elements in Fluids, vol. 1, Gallagher et al, editors, 1975, pp. 233-249.
10. Stoker, J.J.: Water Waves. Interscience Publishers, N.Y., 1957.
11. Everstine, G.C., Schroeder, E.A., and Marcus, M.S.: The Dynamic Analysis of Submerged Structures. NASTRAN: Users' Experiences, NASA TM X-3278, Sept. 1975, pp. 419-429.
12. Schroeder, E.A. and Marcus, M.S.: Finite Element Solution of Fluid-Structure Interaction Problems. David Taylor Naval Ship Research and Development Center, Report 76-0145, Oct. 1976.

13. Marcus, Melvyn S.: A Finite Element Method Applied to the Vibration of Submerged Plates. Journal of Ship Research, to appear.
14. Everstine, G.C.: A NASTRAN Implementation of the Doubly Asymptotic Approximation for Underwater Shock Response. NASTRAN: Users' Experiences, NASA TM X-3428, Oct. 1976, pp. 207-228.
15. Courant, R. and Hilbert, D.: Methods of Mathematical Physics. Interscience Publishers, New York, 1963, vol. I.
16. Haussling, H.J. and Van Eseltine, R.T.: Unsteady Air-Cushion Vehicle Hydrodynamics Using Fourier Series. Journal of Ship Research, vol. 20, no. 2, June 1976, pp. 79-84.
17. Zarda, P.R.: A Finite Element-Analytical Method for Modeling a Structure in an Infinite Fluid. NASTRAN: Users' Experiences, NASA TM X-3428, Oct. 1976, pp. 251-272.
18. Hurwitz, M.M.: Additions to the NASTRAN User's Manual and Theoretical Manual for a Thermostructural Capability for NASTRAN Using Isoparametric Finite Elements. David W. Taylor Naval Ship Research and Development Center, Report 4134, 1973.
19. Hurwitz, M.M.: Programmer's Manual Additions and Demonstration Problems for a Thermostructural Capability for NASTRAN Using Isoparametric Finite Elements. David W. Taylor Naval Ship Research and Development Center, Report 4656, 1975.
20. Giesing, J.P. and Smith, A.M.O.: Potential Flow About Two-Dimensional Hydrofoils. J. Fluid Mechanics, vol. 28, Part I, 1967, pp. 113-129.

MODIFYING THE NASTRAN SYSTEM WITH A NEW CAPABILITY

John R. McDonough
Computer Sciences Corporation

SUMMARY

The purpose of this paper is to discuss the need for a new capability/maintenance interface specification for the NASTRAN system. The paper highlights those problems most difficult to resolve when a new capability is delivered for inclusion into an archive level. The guidelines presented show the objectives of new capability integration as they relate to design and development, delivery, checkout, and documentation. Examples of new capability/maintenance interface already done on an informal basis are presented to illustrate the "test conditions" of the idea of defining implementation objectives.

INTRODUCTION

Since the first release of NASTRAN to the user community, the program has been modified with numerous enhancements, modifications, and the addition of a substantial number of new capabilities. Quite frequently, new capabilities are developed in separate efforts outside the maintenance cycle. The problem subsequently encountered is the difficult task of integrating a new capability into an archive level that has itself been modified since the original base level the new capability was developed for. It was apparent that guidelines were needed to specify requirements for new capability additions and accompanying documentation in order to lessen the impact of the new capability on the archive level.

INTEGRATION OBJECTIVES

The purpose of developing a new capability for NASTRAN generally fits one or more of the following reasons: to correct a design deficiency incorporated in present versions; to install an enhancement that is an advancement in the state of the art (mathematically or conceptually); to add a feature in response to user requirements; or to relieve the user from the burden of manual data preparation or interpretation in favor of automated input or detailed output.

It is assumed, for the purposes of this paper, that the new capability is not developed by the maintenance contractor but is developed by a separate ("New Capability") contractor. Once the new capability is delivered to the NASTRAN Systems Management Office (NSMO), it is the responsibility of the

maintenance contractor to formally install it in the current archive (in-house) level. The maintenance contractor must (if not previously done to some extent by the new capability contractor) certify that the newly installed capability functions according to its intended design and that, when installed, does not degrade the existing system.

The most evident problem, and clearly the most significant, is to integrate the new capability into an existing level. Usually, the two contractors proceed along their respective paths which may diverge. Furthermore, the longer the period of time the new capability is in development, the more divergent are the paths. This occurs because, during the new capability development, the maintenance contractor advances the new capability contractor's common base level in the performance of his tasks. When the integration effort is initiated, the two systems do not necessarily merge; in fact, the chances are they will not merge without further modifications.

This type of situation has occurred ever since NASTRAN itself was developed and released to the public. One of the earliest examples occurred in 1971. By that time, Computer Sciences Corporation (CSC) was performing maintenance of Level 12 on the UNIVAC 1108. At the same time, The MacNeal-Schwendler Corporation was modifying an improved Level 11 on the IBM 360 to develop the Thermal Bending and Hydroelastic capabilities (References 1, 2, and 3). Both contractors' systems were separately functioning satisfactorily, however when the new capability was installed in an archive Level 12 system, modifications had to be performed to address problems associated with different levels and different machines. Fortunately, the impact was lessened in that there was a set of formal demonstration problems (Reference 4) that served as a basis for comparison.

To minimize the integration effort as much as possible, there exists sufficient justification for a set of specifications to delineate the responsibilities of both the maintenance and new capability contractors. The proposed specification¹ is intended to address the problems which repeatedly occur during the integration and consequently provide reasonable and logical objectives.

DISCUSSION OF PROBLEMS

The problems most often encountered in the integration process encompass one or more of the following:

- Deliverables associated with the new capability
- Programming definitions of the new capability code

¹ Unpublished model specification tentatively entitled NASTRAN General Purpose Interface Requirements Document principally developed by Frank J. Douglas (general), W. Keith Brown (programming), and John R. McDonough (system testing) of Computer Sciences Corporation under NASA Contract NAS1-12969.

- Documentation additions to the NASTRAN manuals
- Installation, Testing, and Certification

In order to address these problems, the proposed specification defines the new capability development cycle to be in six phases. These are:

- Definition
- Design
- Programming
- System Testing
- Installation
- Acceptance

Definition Phase

This phase establishes the specific requirements the new capability will satisfy; i.e., the problem to be solved and the Mathematical Specification (MS) needed to obtain the solution.

The MS has been the basic tool utilized throughout the initial development of and subsequent additions to NASTRAN to define the problem and propose its solution. Typically, an MS contains a theoretical development whose content is of sufficient detail that it can readily be inserted into the NASTRAN Theoretical Manual (Reference 1). If new input cards are to be developed, they are defined for the appropriate NASTRAN data deck and are specified in the same content and format as are the cards in the NASTRAN User's Manual (Reference 2). If new functional modules, data blocks, rigid formats, or elements are proposed, they are defined in the same manner as applicable sections of the NASTRAN Programmer's Manual (Reference 3). In essence, the MS clearly describes the intended direction the development of the new capability will take from concept to certification.

Design Phase

This is the phase which contains the New Capability Contractor's (NCC) definition of the types of subroutines, overlays, and operating system requirements needed to fulfill the solution proposed in the Definition Phase. In addition, this phase contains a test plan to define the objectives of test cases that will be used to check the new capability. This type plan roughly approximates appropriate objectives shown in the tables in the front of the NASTRAN Demonstration Problem Manual (Reference 5). The new capability contractor is assumed to develop specific tests to validate his addition in the form of original tests or modified NASTRAN Demonstration Problems.

Programming Phase

This is the actual coding phase in which subroutines, block data programs, and overlays are constructed. Subroutines are expected to contain certain elements in sufficient detail so as to assist any future modifications. These include, in commentary form, the purpose of the subroutine, the Direct Matrix Abstraction Program (DMAP) calling sequence, input variables, output variables, and other descriptive material as may be needed for the subroutine. The need to code comments throughout the subroutine is an essential aspect of the effectiveness of the subroutine's design since this technique serves a dual purpose of producing a form of documentation.

System Testing Phase

The purpose of the System Testing Phase is to verify that current NASTRAN capabilities still exist and to validate the new capability with respect to its intended design. To accomplish this goal, system testing is divided into three categories.

The first category consists of a sample of problems from the existing standard NASTRAN Demonstration problems. These are a specifically chosen set of problems which exercise the NASTRAN operational capabilities (Checkpoint, Restart, Plot, Punch, Rigid Format Alter, Rigid Format Switch) and the NASTRAN computational disciplines (statics, normal modes, buckling, frequency response, transient response, cyclic symmetry, conductive heat transfer, radiation heat transfer, convective heat transfer and aeroelastic flutter analyses). These test cases are executed on the contractors' common base level prior to new capability development and again on the proposed final integration level which includes the new capability.

The second category of system test problems are those devised to specifically emphasize the area encompassing the new capability. These may be special modeling problems exhibiting the new capability's features or an alteration of an existing NASTRAN Demonstration Problem to show its results are repeatable or improved.

The third category of tests are those not usually associated with structural modeling. These are special tests to verify the mathematical computational results of selected stand-alone areas of the new capability contractor's code.

Installation Phase

The Installation Phase is a critical phase in which the New Capability Contractor (NCC) and the Maintenance Contractor (MC) merge their respective areas of expertise to actively integrate the new capability code into two archive levels. One level is, by definition, on the CDC equipment at Langley Research Center. The other level is on one of the other two NASTRAN machines

(IBM or UNIVAC). The third machine integration is performed by the MC alone with consultation by the NCC.

Although the installation of the new capability is designated as a separate phase near the end of the new capability development, it may not occur in such simple terms. It must be recognized that by the time this phase is initiated, the basic problem of integrating a new capability into an archive level may be hindered due to the wide disparity between the New Capability Contractor's modified base level and the Maintenance Contractor's current archive level. Obviously this Phase may have to be adjusted to occur at different times throughout the new capability development cycle. If the cycle is short, it need only be done once. If the cycle is extended, it may have to be done several times.

Acceptance Phase

The final Phase in the new capability development is the acceptance by NSMO of all the elements of the new capability. The elements of a new capability include the required deliverables associated with each phase and the necessary documentation. (The format of the documentation has been previously presented in Reference 6.) In the context of this paper, documentation refers to the formal and final additions to the four NASTRAN manuals (sections of which have been previously referenced). These new capability elements are discussed separately.

ELEMENTS OF A NEW CAPABILITY

Definition Phase Elements

The deliverable associated with the definition phase is the Mathematical Specification (MS). The MS is a statement of the technical problem to be solved; the new capability requirements to be met; and the mathematical solution proposed.

A typical MS would contain the motivation for the new capability, the software requirements to implement it, reliability requirements or limitations, and expected documentation. The definition of the capability is specified by a rigorous and detailed derivation of the equations needed to solve the problem with an in-depth discussion of the theory developed or a relation compared to current technical literature. The style, content, and detail of this development is produced in the same format as the NASTRAN Theoretical Manual with the ultimate goal being that much of it will eventually be placed in that manual. The MS would also define new data cards anticipated in order to utilize the new capability.

Design Phase Elements

Two deliverables are associated with the Design Phase. These are the Design Specification and a proposed Test Plan and Test Specification.

The contents of the Design Specification would detail the information users would need in order to take advantage of the features associated with the new capability and a formulation of the software requirements needed to implement and support the new capability. That information associated with user features and software requirements would be sufficient in content to be easily incorporated into the NASTRAN User's and Programmer's Manuals, respectively.

Relative to the user, the Design Specification would contain appropriate information (where applicable) concerning a discussion of modeling techniques, input deck(s) data cards, changes or additions to the Rigid Formats, diagnostic messages, plotting instructions and definitions of terms. Relative to the software requirements, the Design Specification would contain (where applicable) descriptions of necessary data blocks, tables, subroutines, functional modules, and links. If a new element is defined, the highlights of the mathematics associated with it would be included.

The Test Plan and Test Specification would also be produced in the Design Phase to formulate the anticipated tests required at various steps in the new capability development. These are subsystem tests and installation tests. The purpose of a subsystem test is to independently test software in a stand-alone environment while an installation test is more general to include a formal interface with all of the NASTRAN capabilities. The test plan would include a discussion of the features to be tested and operations required (checkpoint, restart, plot, etc.). The test specification would define the objectives of the test, the model(s) employed to perform the test, and a checklist to verify the expected output and results.

Programming Phase Elements

Since the Programming Phase is the one in which the software is being implemented and subsystem tests are being performed, it is presumed that the definitions put forth in previous phases (which produced the Mathematical Specification and the Design Specification) would, in this phase, be assembled for preliminary documentation.

Based on the results of subsystem testing, the first design criteria are essentially checked. Therefore, a preliminary manuscript of the information intended for inclusion in the NASTRAN manuals would be produced. In addition it would be required, at the end of this phase, to produce compilation listings of altered source code and load maps.

System Testing Elements

This phase requires full scale installation system testing to verify the successful incorporation of the new capability in the NASTRAN environment. The tests would encompass executions of selected standard NASTRAN Demonstration Problems plus the NCC's special purpose test problems. At the completion of these tests, a report of test results would be produced to discuss the impact of the new capability on the existing NASTRAN system. At the completion of the phase, a final edition of compiled source, object, and execution tapes would be delivered as well as decks of alters needed to produce them on two computers.

RECENT INTERFACE EXPERIENCE

During the course of the current NASTRAN System maintenance contract, a model specification for integrating a new capability into an archive level has been formulated², reviewed and critiqued³, revised (to incorporate critiques), and is in the final stages of release for publication. Concurrently, portions of the specifications have been tested with various new capability contractors under varying degrees. Table 1 shows the new capabilities under development.

Automated Multi-stage Substructuring and Improved DMAP

The Automated Multi-stage Substructuring (AMSS) and Improved DMAP capabilities were developed by Universal Analytics, Inc., (UAI). The basic purpose of the AMSS capability (Reference 7) is to provide the NASTRAN user with the ability to analyze components (substructures) of a large structural model to reduce data preparation and computational time and to have the matrices associated with the substructures automatically assembled (Reference 8). The primary purpose of the Improved DMAP capability (Reference 9) is to provide the user with a DMAP language that is less stringent (in terms of format rules) and is more flexible (in terms of statements required) than the language that was originally developed for NASTRAN Level 12 and remained in existence through Level 15.5. Level 16 supports both forms of the DMAP language (Reference 10).

Both capabilities were initiated using Level 15.5 as a base and were installed in an archive Level 15.9. Because UAI is a subcontractor to CSC on the NASTRAN Maintenance Contract, a less formal relationship exists between the two contractors than one which would exist if UAI were not performing some of the

² See Footnote 1.

³ Richard S. Pyle, Charlene Welch, Dr. P. R. Pamidi (Computer Sciences Corporation); Dr. Eric I. Field, David N. Herting (Universal Analytics, Inc.); Allen R. Curtis (Lockheed California Company); Keith H. Redner (Consultant).

maintenance tasks in addition to the new capability development. In this case, the installation and checkout of the new capabilities was incorporated in conjunction with other activities such as error corrections, demonstration problem evaluation, and documentation additions.

At the time the CSC-UAI subcontract was initiated, what would have been the Definition, Design, and Programming phases of the UAI new capability development were virtually complete. It remained to proceed through a joint effort consisting of the rudiments of System Testing, Installation, and Acceptance phases. The new capability code was jointly installed on the Langley Research Center's CDC 6000 series computer and the first and second category tests were executed. Code changes necessary to correct errors discovered, installation and testing of both capabilities on the Goddard Space Flight Center's IBM 360/95 and the Lewis Research Center's UNIVAC 1110 computers, and the finalization of the documentation were completed in time to include them with the public versions of Level 16.

Fast Eigenvalue Extraction Routines

The purpose of the Fast Eigenvalue Extraction Routines (FEER) is to incorporate a tridiagonalization technique to extract roots but to allow the user the option of limiting the number of roots extracted. The capability is intended to be applicable in both real and complex analyses.

Analytical Mechanics Associates (AMA) is the development contractor of this capability. Presently, the real version is complete (Reference 11) and the complex version is still in development.

This was the first instance in which the Maintenance Contractor and New Capability Contractor were able to define a course of action through NSMO that was based on a first draft of the proposed interface specification. This was not contractually binding on either contractor but was a device to test the concept at an early point in the development of the new capability.

At the completion of the Design phase of the new capability, a plan was developed to define the NCC support to be provided by the MC during the Programming phase. It also defined the preferred condition of the Programming phase deliverables since the new capability was being developed on the GSFC IBM 360/95 (with an installation on the LaRC CDC) but installation on the LeRC UNIVAC would be done by the MC alone.

Development began on an archive Level 15.9. Since the IBM was the development machine, one installation was performed as soon as the IBM public Level 16.0.1 became available. The first and second category tests were executed at GSFC by AMA and at LaRC by CSC and AMA. The final integration of the real version of FEER was completed on an archive Level 16.

Matrix Conditioning

This is another capability that was developed by Universal Analytics, Inc. The purpose of this capability is to incorporate a greater level of confidence in the computations associated with singularity conditions and element stress and force calculations. In addition, the Matrix Conditioning package provides the user with the option of requesting output due to forces contributed by multi-point constraints.

The CSC-UAI relationship has previously been described. However, with respect to the development of this capability, another aspect of the proposed specification was tested. This was the development of a Test Plan and Test Results report associated with a System Testing phase. By executing selected NASTRAN Demonstration problems and supplementary problems as well as special purpose problems designed for the new capability, certain areas of code were identified that needed to be changed in order to guarantee acceptability. The Installation phase is presently in progress on an archive Level 16.

Higher Order Plate, Membrane, and Shell Elements

The purpose of the development of the higher order plate, membrane, and shell elements is to provide the user with additional structural elements that incorporate greater accuracy in calculating displacements and defining thermal effects. In addition, the shell element, which couples the bending and membrane properties of the other two elements, provides the user with a modeling tool developed from the mathematics of thin shell theory.

These elements were developed under a NASA research grant with Old Dominion University. The motivation behind the development of these elements can be found in the technical literature (Reference 12). Because this research was monitored by NSMO, in effect, NSMO became the NCC in so far as the new capability integration was concerned. The steps taken through each of the proposed phases resulted in a final installation of the new elements on an archive Level 16. Of particular interest, is the report (Reference 13) that encompasses the deliverables associated with the Programming and System Testing phases.

Aeroelastic Response and Additions to Aeroelastic Flutter Analysis

The Aeroelastic Flutter Analysis, which is available in the public versions of NASTRAN Level 16 (Reference 14), and the Aeroelastic Response capability were developed outside the scope of the NSMO NASTRAN maintenance responsibility. The Aeroelastic Flutter Analysis provides the user with the capability to predict flutter conditions associated with a model that contains structural elements as well as aerodynamic surfaces. The basis for the computations is derived from the Doublet Lattice theory and the k-method of analysis (Reference 15). The

Aeroelastic Response capability will provide the user with the ability to analyze an aerodynamic model under frequency or time dependent conditions in the subsonic or supersonic regimes (Reference 16). The purpose of the new capability activity is to add the Aeroelastic Response and to incorporate enhancements to the existing Flutter Analysis.

This activity is presently in progress. To date, a working archive level has been created and System Testing is in progress on the LaRC computer.

CONCLUDING REMARKS

Six phases were defined in a model specification for the purpose of integrating a new capability into an archive level of NASTRAN. Five new capabilities have been or are being delivered to the maintenance contractor through NSMO. As each new capability was delivered, certain phases of the proposed specification were tested under "field" conditions to evaluate the propositions as defined and to incorporate critiques and comments made from different points of view.

The proposed specification defines the means to add new capabilities to the NASTRAN system to facilitate the management of these activities by NSMO and to provide the user with a level of confidence when they are incorporated and released in the next public level.

REFERENCES

1. The NASTRAN Theoretical Manual, NASA SP-221(03), March, 1976, Section 16.1.
2. The NASTRAN User's Manual, NASA SP-222(03), July, 1976, Sections 1.5, 1.7, 2.4.
3. The NASTRAN Programmer's Manual, NASA SP-223(03), July, 1976, Sections 8.2, 8.5, 8.6, 8.15.
4. The NASTRAN Demonstration Problem Manual, NASA SP-224(03), March, 1976, Sections 1.10, 1.11, 3.2, 3.3, 7.2, 9.3.
5. The NASTRAN Demonstration Problem Manual, NASA SP-224(03), March, 1976, pp. 10-24.
6. Douglas, Frank J., "NASTRAN Documentation from an Historical Viewpoint", NASTRAN: Users' Experiences, September, 1971, pp. 517-532.
7. Field, E. I., Herting, D. N., Herendeen, D. L., and Hoesly, R. L., "The Automated Multi-stage Substructuring System for NASTRAN", NASTRAN: Users' Experiences, NASA TM X-3278, September, 1976, pp. 571-592.
8. The NASTRAN User's Manual, NASA SP-222(03), July, 1976, Sections 1.14, 2.7.
9. Herendeen, David L., "An Improved DMAP Capability", NASTRAN: Users' Experiences, NASA TM X-3278, September, 1975, pp. 595-602.
10. The NASTRAN User's Manual, NASA SP-222(03), July, 1976, Section 5.
11. Newman, Malcolm and Flanagan, Paul F., Eigenvalue Extraction in NASTRAN by the Tridiagonal Reduction (FEER) Method - Real Eigenvalue Analysis, NASA CR-2731, August, 1976.
12. Narayanaswami, R., "New Triangular Plate-Bending Finite Element with Transverse Shear Flexibility", AIAA Journal, Vol. 12, No. 12, December, 1974, pp. 1761-1763.
13. Narayanaswami, R., Addition of Higher Order Plate and Shell Elements into NASTRAN Computer Programs, Technical Report 76-T19, December, 1976.
14. The NASTRAN Theoretical Manual, NASA SP-221(03), March, 1976, Section 17.
15. Doggett, Robert V. and Harder, Robert L., "Subsonic Flutter Analysis Addition to NASTRAN", NASTRAN: Users' Experiences, NASA TM X-2893, September, 1973, pp. 507-529.
16. Harder, Robert L., MacNeal, Richard H., and Doggett, Robert V., "A Design Study for the Incorporation of Aeroelastic Capability into NASTRAN", NASTRAN: Users' Experiences, NASA TM X-2378, September, 1971, pp. 779-795.

TABLE 1. RECENT INTERFACE EXPERIENCES

CONTRACTOR	NEW CAPABILITY
Universal Analytics, Inc.	Automated Multi-stage Substructuring and Improved DMAP
Analytical Mechanics Associates	Fast Eigenvalue Extraction Routines (FEER)
Universal Analytics, Inc.	Matrix Conditioning
Old Dominion University	Higher Order Plate, Membrane, and Shell Elements
MacNeal-Schwendler Corporation	Aeroelastic Response and Additions to Aeroelastic Flutter Analysis

NORCK: A NASTRAN MODULE TO CHECK INPUT DATA AND ELEMENT GEOMETRY

David T. Zemer
NASTRAN Utilization Improvement Program
Northrop Corporation

SUMMARY

NORCK, a new Direct Matrix Abstraction module, has been written which allows the user the ability to define error limits for finite element geometry and properties within the solution flow, and to find these errors before the more expensive parts of the analysis are run.

INTRODUCTION

As the computer becomes faster and computer memory core increases, the NASTRAN user increases the complexity of his structure. His normal method of checking his input data, aside from reading it one card at a time, is through plots and then later by analyzing the output. The best check to date is plotting the structure in sets and trying to interpret the plots. However, this method cannot find all of the errors which may be in the structure or in the properties of the structural elements. For a large project, several engineers or even several groups of engineers may be involved and they constantly change and up-date data. This often leads to common errors, some of which are due to keypunching, misinterpretation of information, and of elements and properties being poorly defined.

NASTRAN finds obvious errors such as missing grid points or blank element properties, but the user has had no method of defining limits to geometry or property data. This can lead to mistakes which are never found or only found after a job is run and the output data has been analyzed. With very large complex models containing tens of thousands of structural data cards which may constantly be updated, it is almost a certainty that errors will be induced allowing geometry failures such as very large or small lengths, areas, and volumes, poorly defined internal angles and out-of-plane quadrilaterals. Complex structures quite often have many property and material cards and a bad property is very difficult to find, especially for matrix type input.

OBJECTIVE

Due to the many different needs of NASTRAN users, an open ended method of checking input errors was deemed necessary. It was required that the limits be user-defined because of the large variety of models which would be run. Other requirements were: compatibility with existing NASTRAN modules, an easy method for the user to define his boundary limits, and a user definition of output form for different models.

NORTHROP DATA CHECKING MODULE - NORCK

Purpose

- A. Print the element ID, connecting external grid ID's and the length of every linear element which has a length less than or greater than given in Table CHECK.
- B. Print the element ID, connecting grid ID's, areas, internal angles, aspect ratio and "warp" angle of every planar element which exceeds:
 - 1. minimum/maximum internal angles
 - 2. minimum/maximum areas
 - 3. maximum aspect ratios
 - 4. out-of-plane or warp angles for quadrilateralsall of which are defined in Table CHECK.
- C. Print the element ID, connecting grid ID's and volume of every solid element which exceeds the volume limit given in Table CHECK.
- D. Print the property card when an element property lies out of the limits given in CHECK.
- E. Print the materials card when any element material property lies outside the limits set in CHECK.
- F. Allow the user to print out only the errors found above, or all geometry and property information with the use of a parameter.
- G. Allow the user to print out the information in floating point format and to choose the number of places following the decimal or exponential format as a default.

DMAP Calling Sequence

NORCK ECT, GPECT, SIL, GPL, EST, EPT, MPT, CHECK//V,N,P1/V,N,P2/V,N,P3\$

Input Data Blocks

ECT - Element Connect Table
GPECT - Grid Point Element Connection Table
SIL - Scalar Index List
GPL - Grid Point List
EST - Element Summary Table
EPT - Element Property Table
MPT - Material Property Table
CHECK - Direct Table Input from User with Limits Defined.

Output Data Blocks

None.

Parameters

P1 - Input-integer-default = 0. This parameter provides user control over output if:

P1 is +1 then full output is obtained
P1 is -1 then only errors are printed.

P2 - Output-integer-default = 0. If a duplicate element is found, set:

P2 = -1.

~~P3 - Input-integer-default = 0. This parameter provides user output print control.~~

P3 = 0 gives exponential format
P3 = 1 to 4 gives printout in floating point format where P3 is the number of digits behind the decimal point.

Any other number gives exponential format.

Remarks

For geometry type testing, Tables ECT, GPECT, SIL, GPL, CHECK are necessary.

For element property testing, Tables EPT and CHECK are necessary.

For material property testing, Tables MPT and CHECK are necessary.

Direct Table Input CHECK

The following card input is necessary.

1	2	3	4	5	6	7	8	9	10
DTI	CHECK	0	1	2	3	4	5	6	0001
+001	LINE					Min	Max		0002
+002	TRIA					Min	Max		0003
+003	QUAD					Min	Max		0004
+004	SOLID					Min	Max		0005
+005	ANGL					Min	Max		0006
+006	WARP					Max			0007
+007	AR					Max			0008
+008	MT1	n				Min	Max		0009
+009	MT2	n				Min	Max		00010
+0010	PROP	n	m			Min	Max		00011
+0011	ENDREC								

<u>Field</u>	<u>Contents</u>
LINE	Two grid element length
TRIA	Three grid element area
QUAD	Four grid element area
SOLID	Solid element volume
ANGL	Internal angle
WARP	Out-of-plane warpage for quadrilateral
AR	Aspect ratio = longest side/smallest side
MT1	For MAT1 type cards
MT2	For MAT2 type cards
Min	Minimum value allowable
Max	Maximum value allowable
n	Field number on this particular type of material/property card
m	Internal card number for property cards output from IFP (User's Manual, Page 2.3-16)

Remarks

If a value is not defined, default values are in the module.

Example

The set of elements in Figures 1 and 2 were used to test the boundary conditions given in DTI CHECK (Table 1).

Element 62300 has had the grid order of the second and fourth nodes on the element card reversed which plots a hour-glass figure.

Element 62400 has one grid (6242) out of the plane formed by the other three. The other elements are used to test the min-max internal angles, lengths, areas, etc., and the answers shown in Table 2 are printed for the errors found.

FUTURE WORK

The names of the limits for DTI CHECK will be changed to match NASTRAN convention and the output will be put into a more legible form.

RECOMMENDATIONS

More features which could be included would be loads integration at planes specified by the user and other checks for input data.

CONCLUSION

NORCK gives the user, and especially the user with a large complex structure and without a graphics capability, more confidence in his analysis and allows him to define his limits for element geometry acceptability before he runs the more expensive steps in his solution.

DTI	CHECK	0	1	2	3	4	5	6	00000570
+0000570ENDREC									
DTI	CHECK	1	1	2	3	23	5	6	00000580
+0000590LINE									
+0000600TRIA									
+0000610QUAD									
+0000620ANGL									
+0000630AR									
+0000640WARP									
+0000650MT1									
+0000660MT1									
+0000670MT1									
+0000680MT2									
+0000690MT2									
+0000700PROP									
+0000710PROP									
+0000720PROP									
+0000730ENDREC									
									00000740

TABLE 1

***** MATERIAL AND ELEMENT PROPERTY CHECK *****

MATERIALS		E	G				NU	RHO	A	TREF	GE	ST	SC	SS	
MAT1	MIN	0.100E+08	0.100E+07				0.28	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
	MAX	0.500E+08	0.200E+08				0.40	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
MATERIALS		G11	G12	G13	G22	G22	G33	RHO	A1	A2	A12	GE	ST	SC	SS
MAT2	MIN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	MAX	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

***** GEOMETRY CHECK DEFAULT VALUES *****

INTERNAL ANGLE MINIMUM 7.0 MAXIMUM 165.0 DEGREES

ASPECT RATIO MAXIMUM 3.0

WARP ANGLE MAXIMUM 4.0

NUMBER OF NODES	MINIMUM AREA	MAXIMUM AREA
2	1.000	45.000
3	50.000	7500.000
4	100.000	15000.000

***** VALUES USED IN GEOMETRY CHECK *****

INTERNAL ANGLE MINIMUM 7.0 MAXIMUM 165.0 DEGREES

ASPECT RATIO MAXIMUM 3.0

WARP ANGLE MAXIMUM 4.0

NUMBER OF NODES	MINIMUM AREA	MAXIMUM AREA
2	1.000	45.000
3	50.000	7500.000
4	100.000	15000.000

TABLE 2(a)

***** ELEMENT PROPERTY CHECK *****

----- BEGIN QDMEM2 4 NODES -----

EL ID	AREA	GRID	ANGL	SIDE	GRID	ANGL	SIDE	GRID	ANGL	SIDE	GRID	ANGL	SIDE	WARP	AR
62000	16.0000	6201	90	4.0000	6202	90	4.0000	6203	90	4.0000	6204	90	4.0000	0	1
62100	4.0000	6202	90	1.0000	6212	90	4.0000	6211	90	1.0000	6203	90	4.0000	0	4
62200	3.5000	6203	153	1.0000	6211	3	11.1803	6222	177	7.6158	6221	27	4.4721	0	11
62300	7.5000	6204	45	4.0000	6203	29	9.8995	6231	67	7.2801	6232	51	6.4031	180	2
62400	19.5959	6241	60	5.6569	6242	90	5.6568	6202	90	4.0000	6201	90	4.0000	60	1

5 QDMEM2 ELEMENTS HAVE BEEN CHECKED THE MIN-MAX VALUES ARE:
 MIN ANGLE OF 3.3664 AT ELEMENT 62200
 MAX ANGLE OF 176.6338 AT ELEMENT 62200
 MIN SIZE OF 3.5000 AT ELEMENT 62200
 MAX SIZE OF 19.5959 AT ELEMENT 62400
 MIN SIDE LENGTH OF 1.00000 AT ELEMENT 62100
 MAX SIDE LENGTH OF 11.18034 AT ELEMENT 62200
 MAX ASPECT RATIO OF 11.18 AT ELEMENT 62200

----- END OF QDMEM2 -----

----- BEGIN ROD 2 NODES -----

EL ID	LENGTH	GRID	ANGL	SIDE	GRID
-------	--------	------	------	------	------

2 ROD ELEMENTS HAVE BEEN CHECKED THE MIN-MAX VALUES ARE:
 MIN SIZE OF 4.0000 AT ELEMENT 101
 MAX SIZE OF 4.0000 AT ELEMENT 101

----- END OF ROD -----

----- BEGIN TRIA2 3 NODES -----

EL ID	AREA	GRID	ANGL	SIDE	GRID	ANGL	SIDE	GRID	ANGL	SIDE
600	7.5000	6201	8	6.4031	6211	4	17.4929	6222	168	11.1803
601	8.0000	6201	45	4.0000	6202	45	5.6569	6203	90	4.0000

2 TRIA2 ELEMENTS HAVE BEEN CHECKED THE MIN-MAX VALUES ARE:
 MIN ANGLE OF 4.3988 AT ELEMENT 600
 MAX ANGLE OF 167.9053 AT ELEMENT 600
 MIN SIZE OF 7.5000 AT ELEMENT 600
 MAX SIZE OF 8.0000 AT ELEMENT 601
 MIN SIDE LENGTH OF 4.00000 AT ELEMENT 601
 MAX SIDE LENGTH OF 17.49286 AT ELEMENT 600

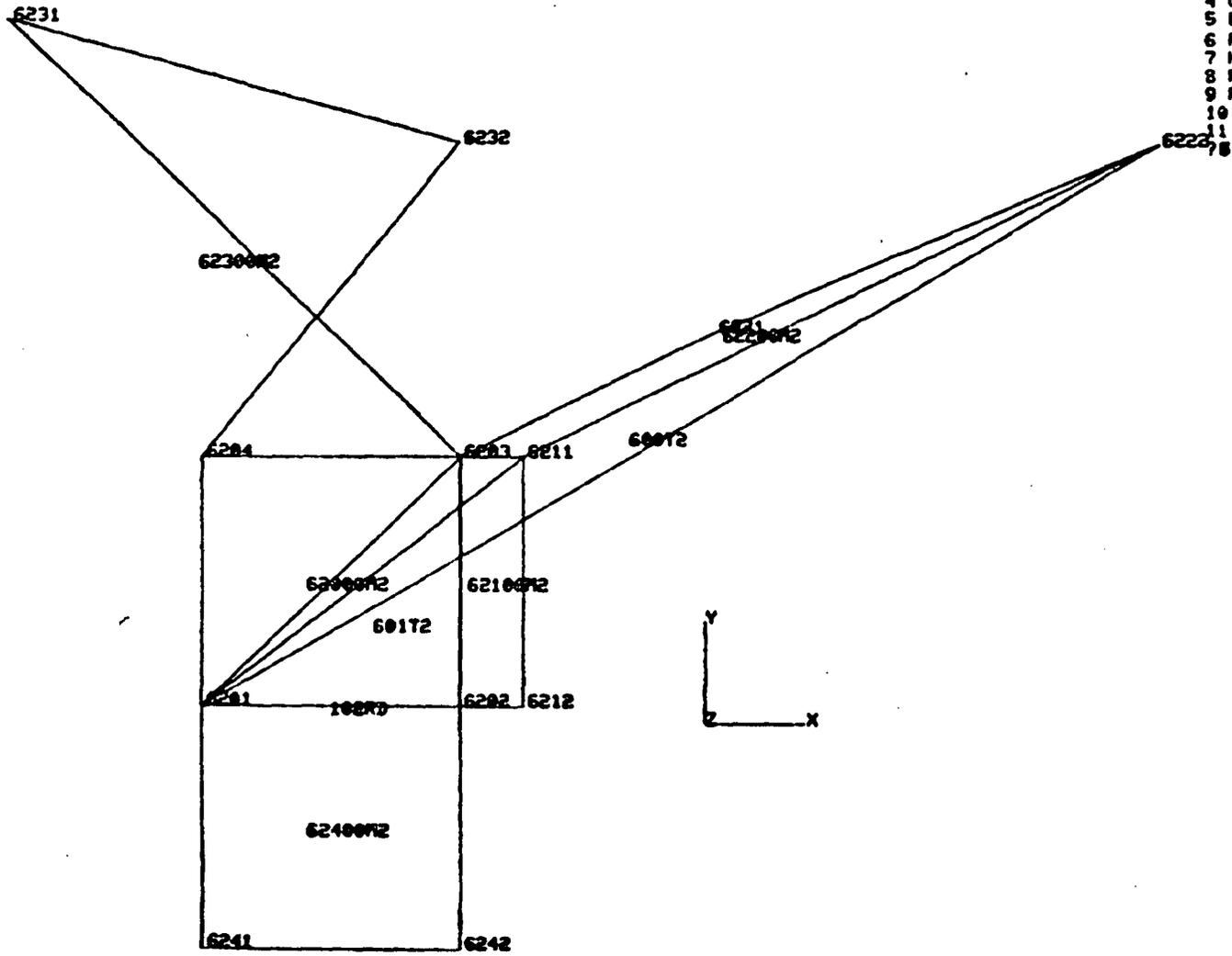
----- END OF TRIA2 -----

----- TOTAL NUMBER OF ELEMENTS CHECKED = 9

TABLE 2(b)

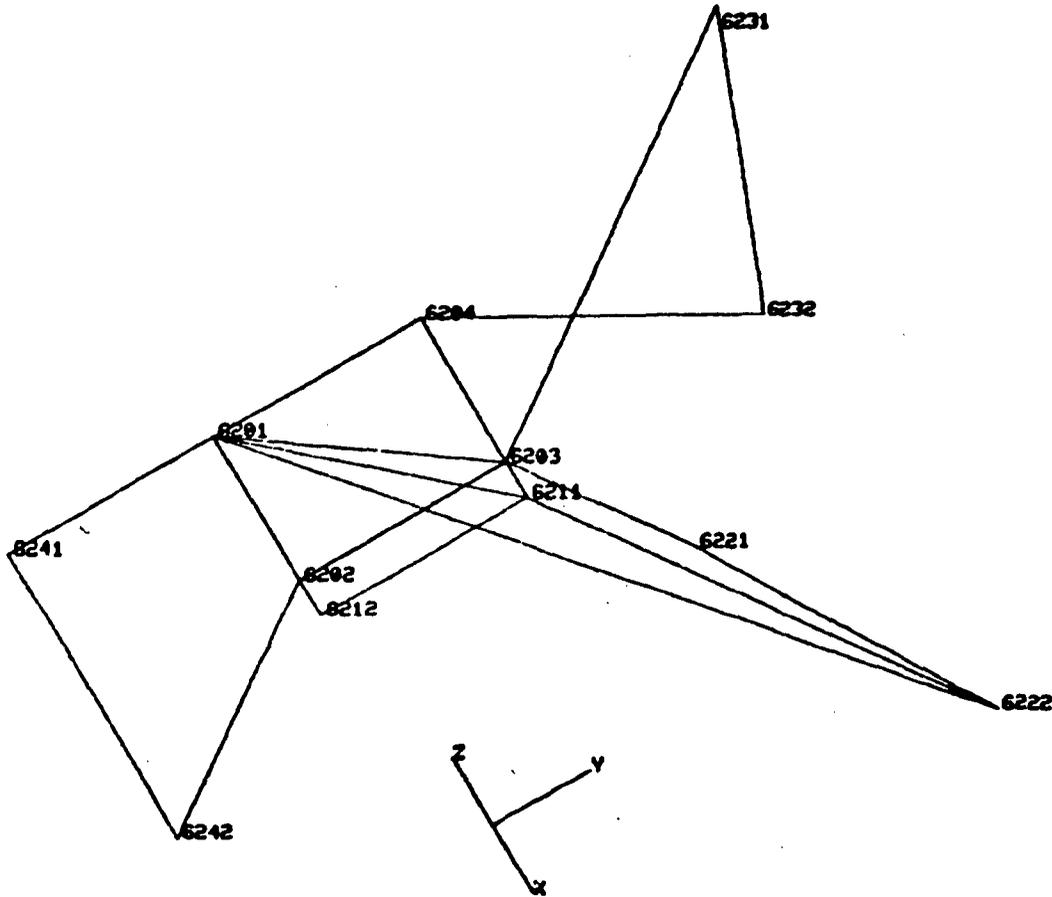
- 1 ZOOM
- 2 NEW ROTATION
- 3 NEW SET
- 4 GRID POINT LABELS
- 5 ELEMENT LABELS
- 6 PLOT AXES
- 7 HIDDEN LINE PLOT
- 8 REPAINT PLOT
- 9 PANEL ARROWS
- 10 MOVE
- 11 END

73



ALP,BET,GAM= 0.0 90.00 90.00
 ELEMENT TYPES PLOTTED: ALL

FIGURE 1



- 1 ZOOM
- 2 NEW ROTATION
- 3 NEW SET
- 4 GRID POINT LABELS
- 5 ELEMENT LABELS
- 6 PLOT AXES
- 7 HIDDEN LINE PLOT
- 8 REPAINT PLOT
- 9 PANEL ARROWS
- 10 MOVE
- 11 END
- 78

ALP,BET,GAM= 30.00 45.00 0.0
ELEMENT TYPES PLOTTED: ALL

FIGURE 2

STUDY OF THE NASTRAN INPUT/OUTPUT SUBSYSTEMS

W. K. Brown
Computer Sciences Corporation

and

W. F. Schoellmann
Computer Sciences Corporation

SUMMARY

The basic characteristics of the NASTRAN Level 16 I/O Subsystem are presented with particular reference to blocking/deblocking aspects, I/O methods used on the IBM, CDC, and UNIVAC machines, definition of basic NASTRAN I/O control tables, and portability of parts of the I/O subsystem to other programs outside the NASTRAN environment. Included is an explanation of the IBM primary, secondary, and tertiary files defined by the data definition (DD) cards in the NASTRAN JCL procedure. The explanation is intended to enlighten users as to the purpose of these DD cards, how they relate to one another, and why there are no similar type definition cards required on the CDC and UNIVAC versions. Enhancements designed to increase overall efficiency and decrease core requirements are also recommended.

INTRODUCTION

The Level 16 NASTRAN I/O Subsystem is divided into two parts: the blocking and deblocking of data and the actual I/O interface with the host operating system. The blocking and deblocking of data essentially involve two different types of data. The first type is general data which may be any one or a combination of alphanumeric, real, or integer words. The second type involves matrix data that is stored in string format. The blocking techniques for these different types of data are discussed.

The I/O interface with the host operating system is different for IBM, CDC, and UNIVAC machines. This interface is explained and enhancements to improve this area of the I/O Subsystem are recommended.

DATA PACKING

Blocking and Deblocking of Data

The blocking/deblocking part of the NASTRAN I/O Subsystem is extremely flexible. The capabilities supported by this part of the I/O Subsystem are as follows:

- read a complete logical record
- blast read a logical record
- read part of a logical record
- read one non-zero term of a matrix column
- read all non-zero terms of a matrix column
- read a complete matrix column with zero padding
- convert precision of a matrix column
- convert type of a matrix column (i.e., real, complex)
- read matrix data directly from the I/O Subsystem buffer

The blocking and deblocking of data is done by subroutine GINØ which is driven by modules for general data, and, in the majority of cases, by subroutine PAKUNPK for matrix data. The data is blocked and deblocked out of buffers pre-allocated from open core by modules. The format of the buffer for each machine is defined in Figures 1, 2 and 3. Blocking/deblocking is best explained through examples.

Consider a call to entry WRITE in subroutine GINØ to write a 10-word logical data record as the first record of a file. The logical data record will be prefaced by a one word record header and appended by a one word record trailer. The format of record headers and trailers are defined in Table 1 for the three machines. Example 1 in Figure 4 shows the twelve words that will be generated in the physical record block by this call to WRITE.

Now consider a call to entry point PACK in subroutine PAKUNPK to write a column of a real single precision matrix into the physical record block. Assume the column contains ten rows of which rows 2, 3, 9 and 10 contain nonzero terms. String packing implies only the non-zero terms will be stored. A string is defined as a set of contiguous non-zero terms in a column. Therefore, in this example, there will be two strings. One string will contain the values of rows 2 and 3 and the second string will contain the values of rows 9 and 10. The column will be prefaced by a record header and a column header and appended by a column trailer (on option) and record trailer. Each string will be prefaced by a string header and appended on option by a string trailer. Note that the type of data, i.e., real single precision, real double

precision, etc., is stored in the column header and trailer. This allows the option for the calling subroutine to specify what precision and what type of data is to be written to or read from the buffer. For example, a subroutine may have a complex single precision matrix column in core but may specify that only the real values of the column be written in the physical record block thus losing the imaginary values within the column. Example 2 in Figure 4 shows the words that are generated for this example assuming the call to PACK was made after the call to WRITE in the first example. To maintain double word boundary, a dummy string definition word is available for insertion into the physical data block buffer.

A call to the entry point CLØSE in subroutine GINØ will result in an end-of-file and end-of-block definition word to be written into the physical record block. Example 3 in Figure 4 shows the result of this call. Once a call to CLØSE is made, the physical record block will be written to mass storage. It should be noted that the trailer information is not written as part of the file on mass storage but is kept in core within the File Allocation Table (FIAT) (see Reference 1).

·NASTRAN I/Ø SUBSYSTEM/OPERATING SYSTEM INTERFACE

Characteristics of the CDC NASTRAN I/Ø Interface

The CDC I/Ø Subsystem issues I/Ø requests through Peripheral Processor (PP) calls and calls to Combined Input Output (CIØ) in subroutine XIØRTN. The I/Ø requests are initiated by the blocking/deblocking subroutine GINØ and the I/Ø initialization subroutine GNFIAT through calls to subroutine IØ6600. Subroutine IØ6600 maintains the Subindex array pointer and calls XIØRTN for the actual I/Ø request. The CDC I/Ø Subsystem processes requests for both sequential and index files. For index files, subroutine IØ6600 maintains the index for each record written in the Subindex array of the buffer (see Figure 1). Once the Subindex array is full, this array is then written on the file and the index for that record is maintained by IØ6600 in the Master Index array permanently located in core (see Figure 5). There is a 62-word array in the Master Index for each file that is maintained in the FIAT (see Reference 1).

The only subroutine in the CDC NASTRAN I/Ø Subsystem package that is portable is subroutine XIØRTN. Subroutine XIØRTN needs the following information:

- an array for the FET
- unit reference number to apply the I/Ø request
- buffer to receive/write data from/to the I/Ø unit
- flag to receive I/Ø status
- index for read operations
- number of words to read/write

- flag to indicate complete or incomplete record
- length of circular buffer in the FET

The purpose of subroutine XIØRTN is to perform the following functions:

- set up the FET
- issue I/Ø requests
- check I/Ø status and set flag for the calling routine
- read/write data from/to supplied buffer

Table 2 documents the entry points in XIØRTN and the operations they perform. This subroutine has been used by several programs outside the NASTRAN environment.

Files that are maintained in the FIAT are dynamically opened by XIØRTN on call from the preface subroutine GNFIAT. Any file that is not preassigned by the user will be dynamically assigned during this open operation.

Characteristics of the IBM NASTRAN I/Ø Interface

The IBM I/Ø Subsystem is the most complex of the NASTRAN I/Ø Subsystems. Due to the complexity of dynamically assigned files during execution, external files are assigned by data definition (DD) cards in a JCL procedure file. The majority of the DD cards assign files with names prefaced by the characters "PRI" (primary), "SEC" (secondary), and "TER" (tertiary) used by the I/Ø Subsystem to store data. The number of "PRI" files assigned determines the maximum number of files to be maintained in the FIAT. Each "PRI" file represents one unit in the FIAT. The "SEC" and "TER" files are spill files that become extensions of the "PRI" files when primary space is exhausted. A deficiency in the logic regarding the extension of a "PRI" file to "SEC" or "TER" files is that no attempt is made to determine whether secondary space may be used on the "PRI" file before connecting a "SEC" or "TER" file since the DD card may have specified secondary space allocation on the "PRI" file. Also, since there may be several "SEC" and "TER" files assigned to the "PRI" file, the same deficiency exists when primary space on the "SEC" or "TER" files is exhausted and other available "SEC" or "TER" files are connected without first trying to use secondary space that may exist on the current "SEC" or "TER" file.

In future releases of NASTRAN, the FIAT printout obtained by requesting DIAG 2 (see Reference 6) will be expanded to tell the user how many "SEC" and "TER" files are connected to each "PRI" file and also tell how much space was used on the "PRI", "SEC" and "TER" files. This supplies the user valuable information regarding file utilization and the actual sizes in words of the data files that NASTRAN is generating. This will aid users in determining which files are costly in terms of storage and in determining reasonable space allocations for certain types of NASTRAN runs.

The File Control Block (FCB) is the means by which the connection between "PRI", "SEC" and "TER" files are maintained by the NASTRAN I/O Subsystem. Table 3 is a description of the FCB. One FCB resides in core at all times for each "PRI", "SEC" and "TER" file. Figure 6 shows core allocation for the IBM NASTRAN System. The parameter FCBNEXT connects the next "SEC" or "TER" file to the current "PRI" file. The parameter FCBPREV is used for "SEC" and "TER" files to refer to the previous connected file which may or may not be the "PRI" file. Through these parameters the chain of connected files is maintained.

A Data Control Block (DCB) (see Reference 2) is generated by the Preface subroutine GNFIAT for every file assigned by a DD card with the exception of the FORTTRAN files. The DCB permanently resides in core (see Figure 6).

Space is allocated for the Data Event Control Blocks (DECB) (see Reference 3) by subroutine GNFIAT (see Figure 6). These DECBs are used only for files that are open and, therefore, the DECBs are used repeatedly. Because of this, the number of DECBs allocated is the maximum number of files (see Reference 4) that NASTRAN may have open at any point in time. Subroutine NASTIØ assigns the DECBs when a file is requested to be opened.

The IBM I/O Subsystem uses the IBM Basic Sequential Access Method (BSAM) (see Reference 5) to issue I/O requests. This method was chosen because NASTRAN's blocking/deblocking capability eliminated the need for any blocking to be done by the operating system. Thus, BSAM allows the capability to write or read a block of data either sequentially or randomly. Most users desire the block of data to be a full track and therefore set the BUFFSIZE parameter (see Reference 6) accordingly. Since every block written goes to a separate track, the relative block number maintained by the I/O Subsystem becomes the relative index for the PØINT supervisor call (see Reference 5). All calls to subroutine NASTIØ are made from subroutine GINØ.

Because of the complexity of the IBM NASTRAN I/O Subsystem, no I/O subroutine is easily portable to be used outside the NASTRAN environment.

Characteristics of the UNIVAC I/O Interface

The UNIVAC I/O Subsystem requests I/O operations through NTRAN (see Reference 7). The files that are written through NTRAN are dynamically assigned by subroutine GNFIAT which allocates 1360 tracks for each file. There will be "MAXFIL" (see Reference 4) files assigned by GNFIAT and they have the file names of 1, 2, 3, 4, etc. The number of files assigned is the maximum number of files in the FIAT plus 5. In order to use NTRAN, three alters are necessary to allow the maximum number of files to be available at one time. The altered NTRAN is the element NTRAN\$ in the NASTRAN source library. The alters do the following:

- set the NTRAN Control Table (NCT) length to 15
- set the number of packets to 37

- set the packet size to 9

The UNIVAC I/O Subsystem maintains a File Control Block (FCB) for each file assigned. The FCB is used only to keep the current block number of a file. On each I/O operation the FCB is compared with the block number of the block just read to insure the correct block was read (see Figure 3). The FCB is part of the /GINØX/ common block. All calls to NTRAN are made from subroutines IØ1108 and SØTRAN (used only for substructuring).

The only I/O subroutines that lend themselves to portability outside the NASTRAN environment are subroutines IØ1108 and SØTRAN. The inputs required for IØ1108 and SØTRAN are:

- I/O operation desired
- I/O buffer

The following operations are performed by IØ1108 and SØTRAN:

- rewind a file
- write a block
- read a block
- backspace one block
- read requested block (IØ1108 only)
- swap tapes for a multi-reel file (IØ1108 only)

The only alter required is to either maintain a FCB when using IØ1108 or to delete all code referencing the FCB.

RECOMMENDATIONS

Enhancements to the CDC I/O Subsystem

It is suggested that the concept of maintaining a Master Index and a Subindex be deleted, thus freeing up core for other purposes. The indexes that were stored in these arrays may be calculated since it only reflects the relative Physical Record Unit (PRU). The relative PRU is a function of the NASTRAN buffer size and the PRU size.

A feature that will be included in the next NASTRAN release is the extension of the FIAT printout (obtained by requesting DIAG 2) to include the size of each file in words.

Enhancements to the IBM I/Ø Subsystem

The allocation of the right amount of disk space for program work files has always been a problem on the IBM operating system. The problem arises because IBM assumes a program user can make a good estimate of the amount of disk space needed for each work file. Making a good estimate, however, requires a knowledge of the NASTRAN I/Ø Subsystem which most users should not be required to have. It is for this reason that the design of NASTRAN includes the concept of the spill work files (i.e., the "SEC" and "TER" files). The algorithm for logically concatenating the spill files to the primary ("PRI") files has been a point of interest to many NASTRAN system programmers and consequently it has been changed many times. The present algorithm is usually sufficient for small or medium size NASTRAN applications, but, large problems require a large amount of disk space overkill or, as stated before, a knowledge of the NASTRAN I/Ø Subsystem.

Presently, NASTRAN avoids using the user specified (via the JCL) secondary space allocation since a program abort will result if the secondary space is not available on the disk. An improvement would be a check in NASTRAN to see if secondary space is available and allow for secondary space allocation before using a spill work file to logically extend the primary file. An IBM system supervisor call already exists to determine the amount of available space left on a disk. The coding changes necessary to utilize the LSPACE supervisor call (see Reference 8), are not extensive and should result in a further NASTRAN refinement to a long existing problem.

Enhancements to the UNIVAC I/Ø Subsystem

The subroutine NTRAN, because of its generality, requires a considerable amount of time to process an I/Ø list (see Reference 7). It is suggested that subroutine NTRAN be replaced by other subroutines employing IØW\$ executive request calls. This will allow faster processing of I/Ø lists, better error handling techniques and messages, and an overall general improvement.

A feature that will be included in the next NASTRAN release is the extension of the FIAT printout (obtained by requesting DIAG 2) to include the size of each file in words.

REFERENCES

1. The NASTRAN Programmer's Manual, NASA SP-222(03), July, 1976, Section 2.4.1.2.
2. IBM System/360 Operating System: System Control Blocks, GC28-6628-5, pp. 37-58.
3. IBM System/360 Operating System: System Control Blocks, GC23-6628-5, pp. 123-125.
4. The NASTRAN Programmer's Manual, NASA SP-222(03), July, 1976, Section 2.4.1.8.
5. IBM System/360 Operating System Supervisor and Data Management Macro Instructions, GC28-6647-5, pp. 45-46, pp. 147-151.
6. The NASTRAN User's Manual, NASA SP-222(03), July, 1976, Section 2.1.
7. UNIVAC 1100 Series Systems Memorandum, Revision 8, Section 8.2.1.7.
8. ØS DADSM Logic, GY28-6607-8, pp29-30.

Table 1. GINØ Definition Words

Definition Word	CDC				IBM				UNIVAC			
	59 54	53 36	35 18	17 0	0 3	4 19	20 31	35 30	29 12	11 0		
Record Header	0	11111	0	NBR WDS	F	1111	NBR WDS	77	11111	NBR WDS		
String Data	0	22222	0	NBR WDS	F	2222	NBR WDS	77	22222	NBR WDS		
Record Trailer												
A. Contained in Block	0	33330	BLK NØ	CLR	2	-	CLR	11	BLK NØ	CLR		
B. Last of Continued Record	0	33332	BLK NØ	CLR	2	-	CLR	12	BLK NØ	CLR		
C. Record Continued in Next Block	0	33333	BLK NØ	CLR	3	-	CLR	13	BLK NØ	CLR		
End of Block	0	55555	0	0	F	5555	0	77	55555	0		
End of File	0	77777	0	0	F	7777	0	77	77777	0		
Column Header	0	CØL NBR	7	8*TYPE+FØRM	4	CØL NBR	16*TYPE+FØRMAT	21	CØL NBR	8*TYPE+FØRM		
Column Trailer	0	CØL NBR	377777	8*TYPE+FØRM	8	CØL NBR	16*TYPE+FORMAT	27	CØL NBR	8*TYPE+FØRM		
String Header * IBM	0	RØW NBR	17	NBR TERMS	F	8888	NBR TERMS	31	RØW NBR	NBR TERMS		
String Trailer	0	RØW NBR	37	NBR TERMS	F	9999	NBR TERMS	37	RØW NBR	NBR TERMS		
Dummy String Def. Word	0	0	777	0	F	AAAA	0	77	33333	0		
End of Block String Word	0	0	77	0	F	BBBB	0	77	44444	0		

* For IBM the next word contains the row number.

Table 2. Entry Points in XIØRTN

Entry Point Name	Operation Performed
XØPEN	Generates FET and opens the file
XCLØSE	Closes the file but does not purge it
XEVICT	Releases space occupied by the file on disk
REINDX	Redefines the index in the FET
XWRITE	Writes partial or complete records on option
XREAD	Reads partial or complete records on option
XREWIND	Rewinds the file
XBKREC	Backspaces the file one record
XFRDREC	Forwardspaces the file one record
WRITEX	Performs blast writes of complete records
READX	Performs blast reads of complete records

Table 3. Description of the FCB

Parameter	Length (Bytes)	Description
FCBFRST*	2	Points to first block of core file
FCBCURNT*	2	Points to current block of core file
FCBLAST	2	Last block number of file
FCBFLAGS	1	I/O flag
FCBDECB	1	Points to DECB assigned to file
FCBBLKNØ	2	Block number at which file is currently positioned
FCBPREV	1	Previous unit assigned to the file
FCBNEXT	1	Next unit assigned to the file
FCBLØW	2	First block number of file for this unit
FCBHIGH	2	Last block number of file for this unit
FCBLKPRI	2	Number of blocks in this unit's primary allocation
FCBLKSEC	2	Number of blocks in this unit's secondary allocation
FCBCLAST*	2	Last block number in core for a core file
FCBNBPT	2	Number of blocks per track
FCBBUFF*	4	Address of I/O buffer assigned
FCBMAX	2	Usage statistics
FCBTIØT	2	Offset in TIØT to entry for this file

* Used only for files kept in core

NASTRAN
I/Ø
Subsystem
Buffer

Physical
Record
Block
to be
written

Word Count	Contents
1	Data Block Name
2	Current Buffer Pointer (bits 35-18) Current Logical Record Pointer (bits 17-0)
3	Pointer to Last Control Word
4	Current Block Number
5	Logical Record Header
NBUFF	End of Block Definition Word
NBUFF + 30	File Environment Table (FET)
NBUFF + 30 + 126	Subindex Array

Figure 1. Buffer Format for the CDC I/Ø Subsystem

NASTRAN
I/O
Subsystem
Buffer

Physical
Record
Block
to be
written

Word Count	Contents
1	Data Block Reference Name
2	Current Buffer Pointer
3	Current Logical Record Pointer
4	Current Block Number
5-6	8 Bytes Containing Full Disk Address in the Form of MBBCCHHR (DCBFDAD)
7	1 Byte Containing the DCB Open Flags (DCBØFLGS) 3 Bytes of Track Balance (DCBTRBAL)
8	Pointer to Last Control Word
9	Logical Record Header
NBUFF	End of Block Definition Word

Figure 2. Buffer Format for the IBM I/O Subsystem

NASTRAN
I/O
Subsystem
Buffer

Physical
Record
Block
to be
written

Word Count	Contents
1	Current Buffer Pointer
2	Current Logical Record Pointer
3	Address of End of Buffer - 2
4	Current Block Number
5	Pointer to Last Control Word
6	Logical Record Header
NBUFF	End of Block

Figure 3. Buffer Format for the UNIVAC I/O Subsystem

Example	Operation	Number of Words	Description
1	Call to WRITE	1	Record Header
		10	Logical Data Record
		1	Record Trailer
2	Call to PACK	1	Record Header
		1	Column Header
		1	String Header
		2	Rows 2 and 3 Stored in Real Single Precision
		1	String Trailer
		1	String Header
		2	Rows 9 and 10 Stored in Real Single Precision
		1	String Trailer
		1	Column Trailer
1	Record Trailer		
3	Call to CLØSE	1	End of File Definition Word
		1	End of Block Definition Word

Figure 4. An Example of Blocked Logical Records in a Physical Record Block

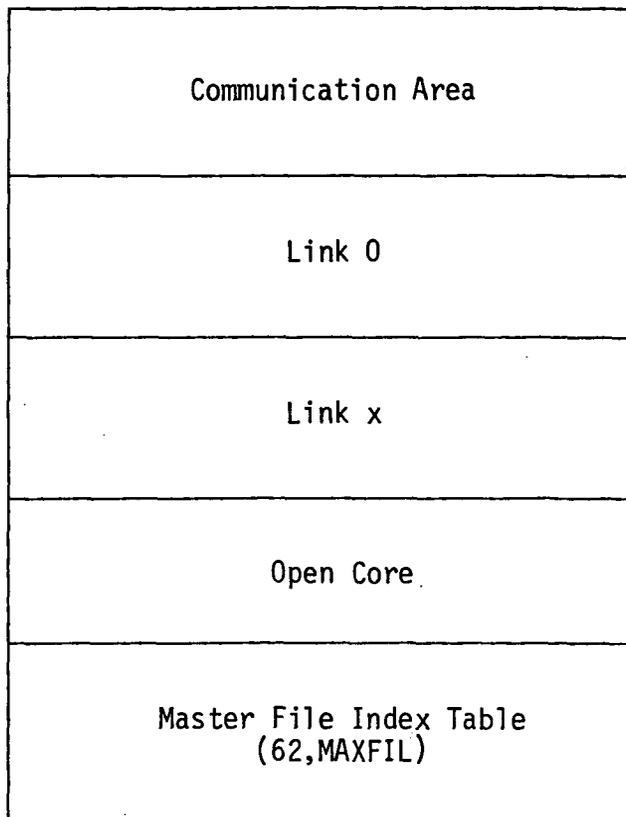


Figure 5. Core Allocation for the CDC NASTRAN System

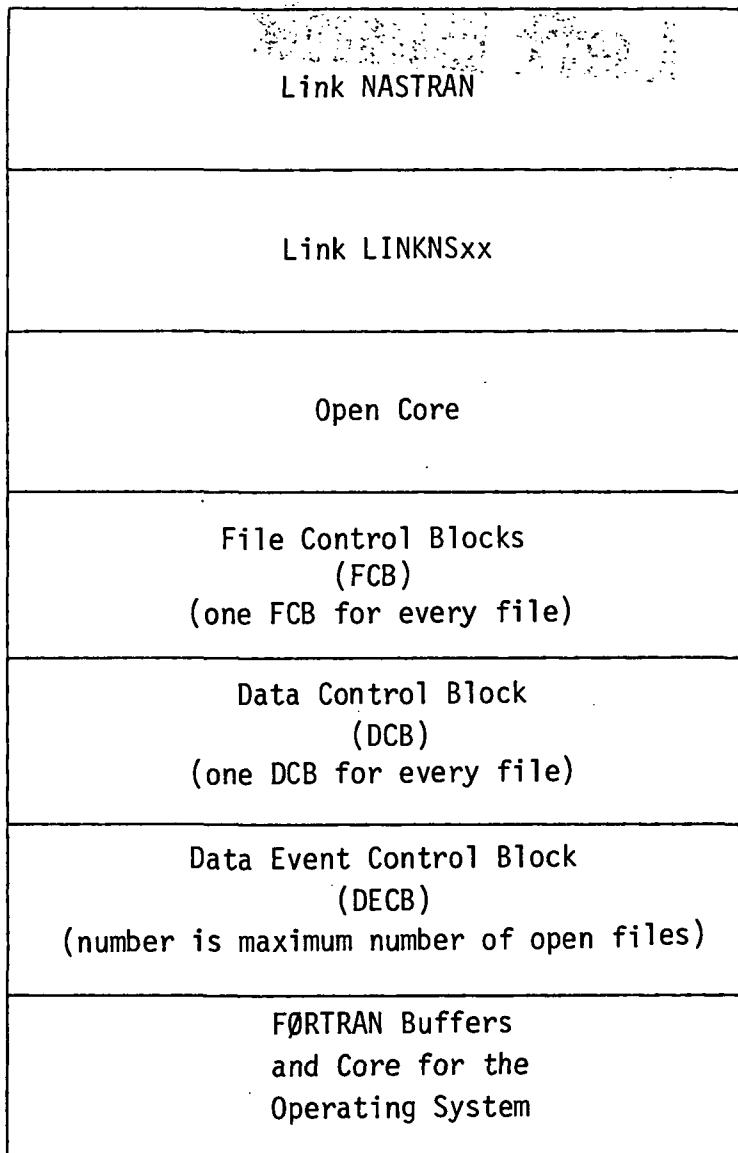


Figure 6. Core Allocation for the IBM NASTRAN System

**Page
Intentionally
Left Blank**

STRUCTURAL MODEL INTEGRITY

D. V. Wallerstein, R. S. Lahey, and G. W. Haggemacher
Lockheed-California Company

SUMMARY

Many of the practical aspects and problems of ensuring the integrity of a structural model are discussed, as well as the steps which have been taken in the Lockheed-California Company's NASTRAN System (Level 15.1 and 16.1) to assure that these checks can be routinely performed. Model integrity as used in this paper applies not only to the structural model but also to the loads applied to the model. Emphasis is also placed on the fact that when dealing with substructure analysis, all of the checking procedures discussed should be applied at the lowest level of substructure prior to any coupling.

INTRODUCTION

The error checking methods prevalent among NASTRAN users appear to fall into four categories: the line-by-line check of the bulk data deck echo; the use of computer generated mesh plots for the detection of improperly shaped or connected finite elements; the use of condition numbers to measure the accuracy of the solution; and the final results look strange method. All of these techniques are both necessary and irreplaceable. There is, however, often a vast amount of data to be checked and questions to be answered once an error is detected by the last two categories. Is the trouble in the applied loads? If the condition number indicates ill-conditioning, is it because of poor sequencing, are the constraints improper, or is the structure unstable? What equations are causing the trouble (if indeed there is any trouble)?

To attempt to answer these questions, the various checks discussed below have been introduced as standard procedure in the Lockheed-California Company's NASTRAN (NASTRAN-LCC). In keeping with the spirit of the philosophy contained in the generation of mesh plots, namely, visibility; an attempt has been made to make the checks as visual as possible. The checks, which have been introduced into NASTRAN-LCC through new modules and modifications to existing modules, fall into the following categories and subcategories:

- Equilibrium
 - External loads - unit load distribution
 - MPC equations
 - Structure boundary stiffness and loads

- Rigid body checks for compatibility of constraints
 - A full structure with support degrees of freedom in the A-set
 - A substructure with boundary points in the A-set
- Equation conditioning (singularities)
 - Stiffness matrix
 - MPC equations
- Selection of simple load conditions to examine model behavior
 - 1-g loads
 - Pressure loads

These checks are, in general, executed through simple DMAP Alters.

SYMBOLS

g-set	Unconstrained grid point displacement set
[PG]	External static load matrix - g size
GPWG	Grid point weight generator module (NASTRAN-LCC)
BGPDT	Basic grid point definition table
CSTM	Coordinate system transformation matrices table
EQEXIN	Equivalence of internal and external indices table
MATPRN	Utility module for printing matrices
[UG]	Displacement matrix - g size
#	Number of load conditions
a-set	NASTRAN analysis set
[RG] = [RMM; RMN]	Multipoint constraint equation
[UM]	MPC dependent degrees of freedom
[UN]	MPC independent degrees of freedom
[PM]	External loads corresponding to [UM]

PS Constraint set on grid cards

SPC Constraint set on SPC cards

MPC Multipoint constraints

DIAGMAT A module to form diagonal matrices (NASTRAN-LCC)

USET Displacement set definition table

[GO] Transformation matrix of stiffness matrix partitioning

SDRP1 Stress data recovery - Phase 1 module (NASTRAN-LCC)

Pcr Critical buckling load

RBMG2 Rigid body matrix generator - Part 2

[GM] Multipoint constraint transformation matrix

EQUILIBRIUM

In the analysis of aircraft structures, the number of applied load conditions can number in the hundreds. While no guarantee can be made that the loads are correct in the sense that a given column of the [PG] matrix represents correctly a desired flight condition, i.e., $N_z = +6.00$, not $+7.20$, each load condition can be checked for static balance. To facilitate this check, the NASTRAN module GPWG has been modified to form a static load summation for each load condition in the [PG] matrix. The actual check consists of the Alter

```
GPWG BGPDT,CSTM,EQEXIN,PG/PGCK/-1/0.0 $
MATPRN PGCK,,,,//$(1)
```

The check is based on the fact the GPWG forms a rigid body matrix $\begin{bmatrix} D \\ g \times 6 \end{bmatrix}$ such that

$$\begin{bmatrix} UG \\ g \times \# \end{bmatrix} = \begin{bmatrix} D \\ g \times 6 \end{bmatrix} \begin{bmatrix} U_0 \\ 6 \times \# \end{bmatrix} \quad (2)$$

holds. $[U_0]$ is the vector of six rigid body motions of the reference point. Hence, from the principle of virtual work, the following load transformation holds

$$\begin{bmatrix} \text{PGCK} \end{bmatrix} = \begin{bmatrix} \text{D} \end{bmatrix}^T \begin{bmatrix} \text{PG} \end{bmatrix} \quad (3)$$

6 x # 6 x g g x #

Rows 1 through 6, respectively, correspond to the PX, PY, PZ, MX, MY, and MZ load summations about the basic coordinate system origin. Table 1 shows that for a unit load in one particular degree of freedom, the values of the three moments about the origin are numerically equal to the location coordinates (basic) to that degree of freedom. Thus, if the applied external loads represent a zero force system (D'Alembert's principle), the [PGCK] matrix should be numerically zero.

An important bonus of this check procedure is that it is useful for determining missing or superfluous PLOAD cards. The PLOAD cards really represent pressure panels whose continuity can not be determined via mesh plots. A pressure vessel must be closed, i.e., produce zero resultant due to pressure. The equations of table 1 can give useful information in determining the approximate location of a hole.

Often, the [PG] matrix represents a matrix of unit load distributions. When this is the case, the [PGCK] matrix directly indicates the locations of the resultant of the unit load distribution as shown in table 1.

Thus far, the equilibrium check has been considered in the context of the [PG] external load matrix. Many other matrices can, however, be considered as load matrices for the purpose of a load check. Two such matrices are the boundary stiffness matrix of a substructure, and the [RG]^T matrix of the multipoint constraint equation. In the former case, each of the columns of any stiffness matrix must, if correctly formed, sum up to zero. In the latter case, the MPC equation

$$\begin{bmatrix} \text{RG} \end{bmatrix} \begin{bmatrix} \text{UM} \\ \text{UN} \end{bmatrix} = \begin{bmatrix} \text{O} \end{bmatrix} \quad (4)$$

via the principle of virtual work implies the static equilibrium relationship

$$\begin{bmatrix} \text{PN(m)} \end{bmatrix} = - \begin{bmatrix} \text{Rmn} \end{bmatrix}^T \begin{bmatrix} \text{Rmm} \end{bmatrix}^{-1} \begin{bmatrix} \text{Pm} \end{bmatrix} \quad (5)$$

This equation expresses equilibrium between [PN(m)] and [Pm]. The resultant [PGCK] is then

$$\begin{bmatrix} \text{PGCK} \end{bmatrix} = \begin{bmatrix} \text{D} \end{bmatrix}^T \begin{bmatrix} \text{Pm} \\ - \text{Pn(m)} \end{bmatrix} = \begin{bmatrix} \text{O} \end{bmatrix} \quad (6)$$

If we let each of the constraint forces necessary to enforce the MPC equation, in turn, equal unity;

[PM] becomes

$$[P_m] = [R_{mm}]^T$$

and

$$[PN(m)] = -[R_{mn}]^T$$

hence

$$[PGCK] = [D]^T \begin{bmatrix} R_{mm}^T \\ R_{mn}^T \end{bmatrix} = [D]^T [RG]^T = [0] \quad (7)$$

and $[RG]^T$ then plays the part of [PG].

RIGID BODY CHECK

The rigid body check is the principal check on the compatibility of all constraints (PS, SPC, and MPC). The rigid body check should be executed after the MPC check, as outlined above, so that the evaluation of results can concentrate on the correct specification of PS and SPC constraints. Errors may be a PS-constraint on a nonsingular degree of freedom, or misspecified SPC constraints. Currently, NASTRAN's rigid body check consists of the print-out of a single error ratio whose nonzero value indicates the existence of a problem somewhere. The check outlined below pinpoints the error by causing a nonzero constraint force to be printed for each incompatible constraint in the standard NASTRAN output. The essentials of the check are the same whether or not it is used on a full structure or a substructure. In the former case, the static supports are on ASET cards instead of SUPORT cards (in essence a substructure analysis has been formed) and in the latter case, the ASET cards represent substitute boundary degrees of freedom.

The check applies rigid body motions to the structure for which all resulting constraint forces should be null. Consider the beam shown in figure 1 which is supported at gridpoints 1 and 3 as shown, with the supports in the ASET. Assume a unit load acts at each ASET degree of freedom and at no other. Form the diagonal matrix

$$\begin{matrix} [UAA] & = & [I] \\ \text{axa} & & \text{axa} \end{matrix} \quad (8)$$

Merge this matrix into a (g x a) matrix

$$[LDA] = \begin{bmatrix} [I] \\ [0] \end{bmatrix} \quad (9)$$

g x a

Treat [LDA] as a static load vector and form

$$[ULT] = \begin{bmatrix} \bar{D}^T & \bar{\bar{D}}^T \\ 6xa & 6x(g-a) \end{bmatrix} \quad \begin{bmatrix} [I] \\ [0] \end{bmatrix} = [\bar{D}]^T \quad (10)$$

by use of NASTRAN-LCC's GPWG. The [ULT] matrix represents the static balance at the reference point for unit loads at each of the grid points. Then [ULV] = [ULT]^T are the rigid body displacements in the a-set corresponding to unit (rigid body) displacements at the reference point. This is so because

$$[UG] = \begin{bmatrix} \bar{D} \\ \bar{\bar{D}} \end{bmatrix} [RBM] \quad (11)$$

Module SDRL, modified in NASTRAN-LCC to double precision, then forms [UO] = [GO] [ULV], and by merging techniques forms the full solution [UGV].

The Alter for Level 16 NASTRAN is

ALTER	98
DIAGMAT	USET,/UAA//1.0/*A* \$
EQUIV	UAA/KAA/ALWAYS \$
ALTER	121
VEC	USET/VPA/*G/*A/*COMP* S
MERGE	UAA,,,,,VPA/LDA/1/2/2 \$
GPWG	BGPDT,CSTM,EQEXIN,LDA/ULT/-1/0.0 S
TRNSP	ULT/ULV \$

In the last step of the Alter, [ULT]^T is output directly as [ULV] since no SUPORT cards are permitted and hence the ASET and LSET are equivalent. The normal solution sequence of NASTRAN is now continued. To find incompatible constraints, simply look for numerically nonzero values in the constraint forces output.

EQUATION CONDITIONING

As a prelude to equation conditioning checks, a positive method of equation identification is desirable. To accomplish this, NASTRAN module GP4 has been modified to increase the visibility of the USET table print between internal-external degree-of-freedom numbers and subset indication (DIAG = 21) by printing the equation number within each subset rather than just an asterisk. Figure 2 shows the new print. The advantage of this print lies in the fact that, if the user thinks something is wrong in equation 216 of the OSET for example, he can immediately correlate this to external grid point 1002, degree-of-freedom 1.

Ill-conditioning arises from several sources. A column under an applied load equal to P_{cr} is ill-conditioned if a small lateral load is applied (Rigid Format 4). More commonly, however, ill-conditioning arises because of modeling oversights such as improper constraints on a grid point or because of computational difficulties due to computer number manipulation. There are many ways to measure ill-conditioning; unfortunately, none of the methods guarantee that the matrix is ill-conditioned and most do not yield information as to where the ill-conditioning is occurring. For example, the check procedure based on use of a condition number defined as the ratio of the maximum eigenvalue to minimum eigenvalue can be expensive to compute and may, as in the case of a structure made up of trains of linear springs numbered from tip to root, be overly pessimistic as to the quality of the solution.

A simple test, which Lockheed-California Company has had good experience with for over 15 years, is the independence index. This index provides a measure of independence for each row/column of the matrix being decomposed and has the attraction of being inexpensive. If d_i represents the value of the diagonal coefficient just before it acts as a pivot and K_{ii} is the original diagonal coefficient, the index is defined as

$$S(I) = d_i/K_{ii} \quad (12)$$

A very low value of the index $S(i)$ indicates an almost zero value of the pivot and hence a singularity. Each negative power of ten for $S(i)$ represents an accuracy loss of that many leading digits in the solution.

The NASTRAN-LCC module SDCK computes $S(I)$ for each diagonal term of the matrix being checked and prints d_i , K_{ii} , and $S(I)$ for each term as well as the equation number. Additionally, however, to increase visibility, the module also prints next to each $S(I)$ an asterisk for each negative power of ten in $S(I)$. A sample print is shown in figure 3. Many asterisks, representing small values of $S(I)$, may readily be scanned for visibility.

Whenever a pattern as delineated in figure 3 appears, a close inspection of the local structure is warranted. In this particular case, the pattern was contrived for demonstration purposes. The Alter to produce the check for the stiffness matrix would be (for Level 16, Rigid Format 1):

```
ALTER 106
SDCK KLL,LLL// $ (13)
```

or in the case where ASET points are boundary points,

```
ALTER 97
SDCK KOO,LOO// $ (14)
```

Matrices other than stiffness matrices can of course be checked. A case in point is the RMM matrix. Module MCE1 has been modified in NASTRAN-LCC to output GM (no change except GM is purged if RMM is singular), RG (instead of from GP4), and RMM. If RMM is multiplied by its transpose, a positive definite symmetric matrix is formed. Putting this result through RBMG2 and its output, LLL, through SDCK yields a check on RMM. The motive behind this check developed when NASTRAN claimed that the MPC's were singular. As it turned out, they were not; rather the pivots during decomposition in MCE1 fell outside the attainable range of the algorithm. Resequencing fixed the problem.

SIMPLE LOAD CONDITIONS TO EXPLORE MODEL BEHAVIOR

This procedure is mentioned for the sake of completeness, as it is common at most installations. The checking process consists of a critical review of standard analysis output to check the rational behavior of the structure. For this, some suitable unit load cases should be chosen so that the physically rational behavior of the structure in terms of deflection and internal force flow can be verified. A fairly realistic condition loading all degrees of freedom can be obtained, through the use of simple 1-g inertia condition, by loading the structure by its own weight. If a density is specified on material cards and the GRAV bulk data card is used to form the gravity acceleration matrix

$$\begin{bmatrix} 1.0 & 0. & 0. \\ 0. & 1.0 & 0. \\ 0. & 0. & 1.0 \end{bmatrix}$$

then a mass matrix generation based on element weight will occur. This is always a reasonable mass matrix. Deflections and stresses can then be judged

on the basis of a 1-g load in the structure. Similarly, for a pressure vessel type structure (such as a fuselage), a unit pressure load is useful.

CONCLUDING REMARKS

While it is impossible to ensure computationally that a structural model correctly represents the structure which the engineer has in mind, it certainly is possible to ensure that the solution is correct and reasonable for what has been modeled. Also, cognizance should be taken of all the nonstructural problems which arise when substructure coupling is involved. Large structures consisting of tens of thousands of degrees of freedom, hundreds of load conditions, and data recovery output that can number a million lines of print for a single substructure will yield their own traumatic surprises in bookkeeping and organization. To attempt such a coupling, without the previous knowledge of successful checks as outlined in categories 1 through 4, can only be considered a masochistic exercise.

TABLE 1. MOMENTS AT REFERENCE POINT FOR UNIT VALUE OF LOAD AT X,Y,Z

Moment Equation	Unit Value of Load		
	PX = 1.0	PY = 1.0	PZ = 1.0
$MX = YxPZ - ZxPY$	0	$-Z(WL-AS)$	$Y(BL-AS)$
$MY = ZxPX - XxPZ$	$Z(WL-SYM)$	0	$-X(MS-SYM)$
$MZ = XxPY - YxPX$	$-Y(BL-AS)$	$X(MS-AS)$	0

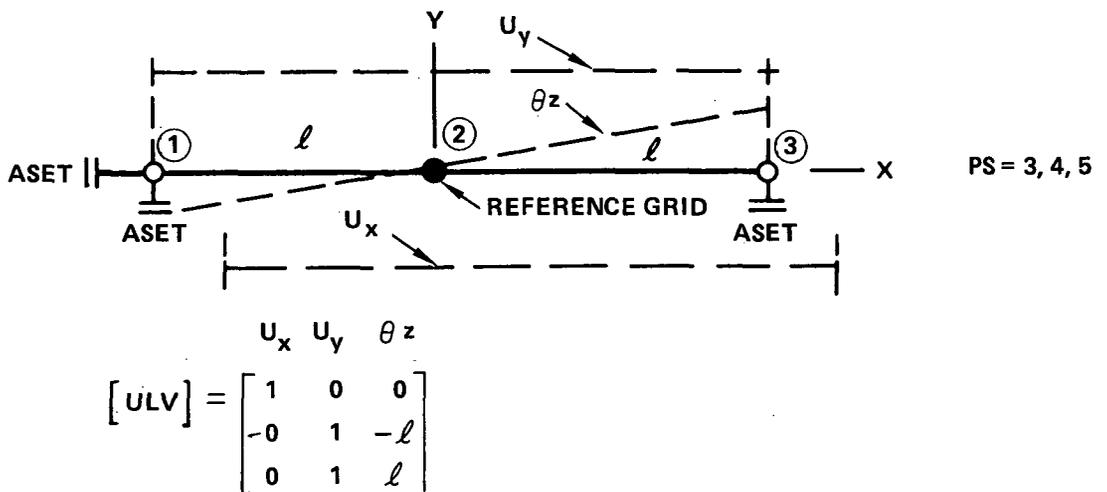


Figure 1. Rigid Body Check

SDCK

SINGULARITY CHECK OF DECOMPOSITION IN RBMG2

I	L(I,I)	M(I,I)	S(I)	Y	I	L(I,I)	M(I,I)	S(I)	Y				
169	8.2687350	06	2.5118020	07	3.291953E-01	*	170	5.8394690	C5	7.6316610	05	7.651635E-01	*
171	2.7610890	05	3.5012930	05	3.247846E-01	*	172	9.9674820	05	1.5847500	06	6.289622E-01	*
173	8.2254710	06	2.5118090	07	3.274720E-01	*	174	5.7696370	05	7.6316640	05	7.560129E-01	*
175	2.7606170	05	8.5021290	C5	3.246971E-01	*	176	9.9666240	05	1.5847500	06	6.289081E-01	*
177	1.0145600	06	1.2587050	06	8.060338E-01	*	178	8.0521250	05	1.4421140	06	5.583558E-01	*
179	2.8749270	04	6.5324300	04	4.400854E-01	*	180	1.1972210	06	1.5272640	06	7.838996E-01	*
181	1.5422300	07	1.9285840	07	7.996694E-01	*	182	1.8276380	07	1.8686650	07	9.780446E-01	*
183	8.4217130	05	1.1969260	06	7.036058E-01	*	184	1.4603930	06	3.1099740	06	4.695836E-01	*
185	1.8699470	07	1.3038950	06	1.434124E-01	*	186	1.5417770	07	1.9285920	07	7.994313E-01	*
187	1.8275290	07	1.8686650	07	9.779364E-01	*	188	8.4214210	05	1.1969480	06	7.035747E-01	*
189	1.4572020	06	3.1099740	06	4.685576E-01	*	190	1.8437600	07	1.3038950	08	1.414040E-01	*
191	1.6035570	06	1.6213500	06	9.890255E-01	*	192	4.8697000	04	7.6918350	04	6.339239E-01	*
193	1.0709500	06	1.7356010	06	6.170486E-01	*	194	1.2504830	06	1.5401080	06	8.121850E-01	*
195	1.6036670	06	1.6214660	06	9.890227E-01	*	196	4.8694230	04	7.6818230	04	6.338890E-01	*
197	1.0708970	06	1.7355940	06	6.170145E-01	*	198	1.2508310	06	1.5401080	06	8.121707E-01	*
199	2.4979070	07	3.0642430	07	8.151790E-01	*	200	7.1212230	05	8.5601360	05	8.319053E-01	*
201	5.9207780	05	3.9186430	05	6.639653E-01	*	202	1.0191090	06	1.5163180	06	6.720942E-01	*
203	2.4965480	07	3.0642470	07	8.147347E-01	*	204	7.1186120	05	8.5601350	05	8.316004E-01	*
205	5.9207000	05	8.9186030	05	6.639035E-01	*	206	1.0191090	06	1.5163180	06	6.720945E-01	*
207	9.0345280	05	1.2870770	06	7.019414E-01	*	208	1.3031070	06	1.6313290	06	7.988006E-01	*
209	2.8242720	04	6.2974660	04	4.484774E-01	*	210	1.2107590	06	1.5087220	06	8.025065E-01	*
211	5.3316390	05	1.5066460	07	3.538746E-02	**	212	1.9864620	04	1.8230610	07	1.089630E-03	***
213	4.2776010	05	8.9965380	05	4.754719E-01	*	214	6.1985560	04	2.1888130	06	2.831926E-02	**
215	3.7150120	04	1.0465660	08	3.549717E-04	****	216	4.4019190	05	1.5066440	07	2.921672E-02	**
217	1.6450650	04	1.8230610	07	9.023638E-04	****	218	3.8748560	05	8.9950530	05	4.307764E-01	*
219	5.7481900	04	2.1888130	06	2.626168E-02	**	220	3.7127400	04	1.0465660	08	3.547545E-04	****
221	3.5796150	06	4.5398980	06	7.884790E-01	*	222	5.5345440	06	5.5555360	06	9.962214E-01	*
223	2.6770800	06	3.4524620	06	7.754118E-01	*	224	7.3212630	05	7.3525640	05	9.957428E-01	*
225	3.5744840	06	4.5375090	06	7.877637E-01	*	226	5.4544810	06	5.4751920	06	9.962173E-01	*
227	2.0950440	06	2.8620360	06	7.320113E-01	*	228	7.3211810	05	7.3525640	05	9.957317E-01	*
229	3.4676890	07	5.7055200	07	6.077778E-01	*	230	7.6346450	06	7.9772810	06	9.570485E-01	*
231	6.8772890	07	6.8980730	07	9.969867E-01	*	232	3.0991640	06	3.2891630	06	9.422349E-01	*
233	6.1962970	07	2.2196500	08	2.791566E-01	*	234	3.4664840	07	5.7055200	07	6.075667E-01	*
235	7.6273760	06	7.9281450	06	9.620632E-01	*	236	6.8204740	07	6.9363280	07	9.983062E-01	*
237	3.0991450	06	3.2891630	06	9.422292E-01	*	238	6.0742620	07	2.2196500	08	2.736586E-01	*
239	1.2780590	07	1.3176330	07	9.699656E-01	*	240	4.3874110	06	8.4518990	06	5.191036E-01	*
241	3.6909450	06	5.4577670	06	6.762555E-01	*	242	2.9617070	06	3.6043340	06	8.217072E-01	*
243	1.2780340	07	1.3176320	07	9.699465E-01	*	244	4.3581580	06	8.3738580	06	5.204480E-01	*
245	3.5985180	06	4.8675970	06	7.392817E-01	*	246	2.9552930	06	3.6043340	06	8.199278E-01	*
247	8.8488970	05	1.0253300	06	8.630295E-01	*	248	1.2847740	06	1.8455100	06	6.961619E-01	*
249	1.2041950	05	1.7785750	05	6.770559E-01	*	250	1.8718860	06	2.1333330	06	8.774465E-01	*
251	3.8238470	06	3.8928760	06	9.822679E-01	*	252	1.1394160	06	5.4537870	06	2.089220E-01	*

TROUBLE AT OR
ABOVE THESE
DEGREES OF
FREEDOM

Figure 3. Portion of Solution Quality Index SCI

A DIRECT MATRIX ABSTRACTION PROCEDURE FOR A
STATIC SOLUTION DURING A PHASE I SUBSTRUCTURING ANALYSIS

David T. Zemer
NASTRAN Utilization Improvement Program
Northrop Corporation

SUMMARY

This paper describes a Direct Matrix Abstraction procedure which allows a static solution to be obtained in Phase I of a three phase NASTRAN substructuring analysis while at the same time storing all information needed to complete the substructuring analysis.

INTRODUCTION

Substructure analysis has been used for several years now in NASTRAN (ref. 1, 2 and 3) and on the whole has succeeded in giving the user a relatively easy method of running large or complex structures in smaller steps. However, one weakness has been the inability of the user to ascertain whether or not his separate substructures were poorly defined or, if they were valid, to obtain usable results before the third and final phase was run.

In the aerospace industry today, it is quite common for the structural analysis of an aircraft to be divided up not only among several groups of engineers within the company, but among several companies or even among several countries (ref. 4). By the very nature of substructuring, the separate substructures could not provide any preliminary analysis data until all substructures were error free and the coupling in Phase II and output of results in Phase III completed. This has meant that a project using the substructuring method would slow down or come to a halt if one section were not completed on time.

The alternative to waiting for all substructures to be completed was to "brickwall" or fix each completed section at its common boundaries for each loading condition in order to check for poorly defined structural elements and to give the waiting stressmen preliminary output data with which to work. This method unfortunately did not provide the necessary NASTRAN tables for further substructure analysis. This report presents a DMAP Alter to the

MacNeal-Schwendler Corporation Version 34, Solution One, Static Analysis which allows the user to do both a static solution as well as obtain the necessary tables for a substructuring solution in one run and with a few minor changes could be incorporated into any NASTRAN substructuring solution.

SYMBOLS

K Stiffness matrix
P Load vector matrix
u Displacement vector matrix
G Transformation matrix

Subscripts:

m Multipoint constraint set
s Single-point constraint set
o Omitted set
r Reaction set
l Left over set
e Extra set

The vector sets obtained by combining two or more of the above sets are (+ sign indicates the union of two sets):

a = r + l, the set used in real eigenvalue analysis
d = a + e, the set used in dynamic analysis by the direct method
f = a + o, unconstrained (free) structural points
n = f + s, all structural points not constrained by multipoint constraints
g = n + m, all structural (grid) points including scalar points.

Superscripts:

T Transpose operator
-1 Inverse operator

Symbols:

[] Matrix

THEORY

The basic theory behind substructuring is well documented (ref. 5 and 6). The differences between the normal NASTRAN solution and this solution with static analysis is discussed here for Phase I, II and III using normal NASTRAN notation and shown in the flow diagram in Figure 1.

Phase I

This solution is basically the same as normal NASTRAN substructuring up to module SSG2 (ref. 7, 8 and 9) (Static Solution Generator - Phase 2). The purpose of this module is to reduce the applied loads vector and enforced load vector applied to the independent displacement coordinate sets. This reduction process is discussed in the NASTRAN Programmer's Manual (ref. 10).

This reduction does not allow the user the choice of fixing his substructure at his boundary points which are where he is normally interested in seeing reaction forces. The loading vector on the analysis degrees of freedom, PL, obtained from SSG2 will not contain the correct degrees of freedom if reaction points are induced by the most logical method, the use of SUPORT cards. Therefore, PA, the loading vector on the analysis set degrees of freedom is obtained by deleting SSG2 and reducing PG step-by-step as shown in the DMAP in Appendix A and explained in Appendix B. PA is then written onto a separate tape for use in Phase II. Now, because PA contains the loading vectors for "R", Reactions, and "L", Leftovers, it is again reduced when SUPORT cards are present and from here a normal static solution is obtained with these as reaction forces.

Phase II

Phase II is unchanged, PA will be read instead of PL.

Phase III

Phase III is basically the same. The major change is the deletion of SUPORT cards from the Phase I old problem tape and replacement by ASET cards.

DISCUSSION AND EXAMPLE

This solution has been found to be quite simple to use. The user has only to replace selected analysis set degrees of freedom with SUPORT (reaction) points in Phase I and then in Phase III delete the SUPORT cards and add identical ASET cards. It also allows substructures to be analyzed as they become available.

A six substructure model (Figure 2) has been analyzed using these alters and the results have matched exactly with both a "fixed" static solution and a normal substructuring solution.

CONCLUSION

Using the powerful matrix manipulations of the NASTRAN Direct Matrix Abstraction method (DMAP) a very effective solution has been obtained which allows a static analysis of a substructure to be made irregardless as to the availability of the connecting substructures and at the same time storing onto user tapes or disk files all information needed for a later substructure analysis.

ACKNOWLEDGMENT

The help of Mr. Carl Hennrich of the MacNeal-Schwendler Corporation is greatly appreciated for both his advise and encouragement on this solution.

REFERENCES

1. Grooms, H.R. and Yahata, S.: Space Shuttle - The need for Substructuring. Space Division, North American Rockwell. NASTRAN: Users' Experiences, NASA TM X-2378, Vol. II, September 1971, pp. 769-778.
2. Narayanaswani, R.: Applications of NASTRAN Substructuring. NASA Langley Research Center, NASTRAN Users' Experiences, NASA TM X-2637, September 1972, pp. 485-496.
3. Hennrich, C.W. and Konrath, E.J., Jr.: Substructure Analysis Techniques and Automation. Software Sciences, Inc., NASTRAN User's Experiences, NASA TM X-2893, September 1973, pp. 323-394.
4. Taig, I.C. and Cheetham, P., British Aircraft Corporation Limited; Gebhard, H., Messerschmitt-Bolkow-Blohm GmbH; Iseni, A., Aeritalia; and Schmalenbach, E., VFW/Fokker GmbH: NASTRAN as a Common Analysis System for a Multicompany Aircraft Project.
5. Przemieniecki, J.S.: Matrix Structural Analysis of Substructures. JAIAA, Vol. 1, No. 1, January 1963, pp. 138-147.
6. Przemieniecki, J.S.: Theory of Matrix Structural Analysis. McGraw-Hill Book Company, New York, 1968.
7. The NASTRAN User's Manual. Level 15.5, March 1974, pp. 1.10-1 to 1.10-26.
8. MSC/NASTRAN Application Manual for IBM 360/370 Series. November 1972, pp. 2.14-1 to 2.14-47.
9. The NASTRAN Theoretical Manual. Level 15.0, December 1972, pp. 4.3-1 to 4.3-6.
10. The NASTRAN Programmer's Manual. September 1972, pp. 4.42-1 to 4.42-4.

APPENDIX A

DMAP ALTER PHASE I

```

$-----
$----- START ALTER TO DO PHASE I AND STATIC SOL. TOGETHER -----
$-----
1. ALTER 3
2. FILE KFS=SAVE/GO=SAVE $
3. FILE LOO=SAVE
4. ALTER 50
5. PARAM //ADD/V,N,LOOPCT/0/1 $
6. ALTER 54
7. PARAM //GT/V,N,JMP/V,N,LOOPCT/1
8. COND LBL7,JMP
9. ALTER 80,80 $
10. PARAM //C,N,PREC/V,N,RPREC $
11. UPARTN USET,KFF/KOO,,KOA,KAAB /C,N,F/C,N,O/C,N,A $
12. CHPNT KOO,KOA,KAAB $
13. DECOMP KOO/LOO, UOO /C,N,1/C,N,0/V,N,MIND/V,N,DET/V,N,POW/V,N,SING$
14. SAVE MIND,SING $
15. CHPNT LOO,UOO $
16. COND LBSING,SING $
17. ALTER 81,81
18. FBS LOO, , KOA/GO/C,N,1/C,N,-1/V,N,RPREC/V,N,RPREC $
19. CHPNT GO $
20. MPYAD KOA,GO,KAAB /KAA/C,N,1/C,N,1/C,N,1/V,N,RPREC $
21. CHPNT KAA
22. ALTER 82
23. OUTPUT1 ASET,KAA//-1//C,Y,USETP1=PIXX $
$-----
$----- REPLACE SSG2 IN ORDER TO DO A -----
$----- STATIC SOLUTION WITHIN PHASE I -----
$----- D. ZEMER -----
$-----
24. ALTER 100,101
$----- MPC
25. EQUIV PG,PN/MPCF1$
26. COND LBLM,MPCF2 $
27. VEC USET/MVEC/C,N,G/C,N,N/C,N,M $
28. PARTN PG,,MVEC/PNBAR,PM,,/1 $
29. MPYAD GM,PM,PBAR/PN/1 $
30. CHPNT MVEC,PBAR,PM,PN $
31. LABEL LBLM $
32. EQUIV PN,PF/SINGLE $
33. CHPNT PF $
34. COND LBS,SINGLE $

```

```

$----- SPC
35. VEC USET/SVEC/C,N,N/C,N,F/C,N,S $
36. PARTN PN,,SVEC/PFBAR,PS,,/1$
37. MPYAD KFS,YS,PFBAR/PF//-1 $
38. CHPNT SVEC,PFBAR,PS,PF $
39. LABEL LBL$
40. EQUIV PF,PA/OMIT $
41. CHPNT PA $
42. COND LBLO,OMIT $
$----- OMIT
43. VEC USET/AVEC/C,N,F/C,N,O/C,N,A $
44. PARTN PF,,AVEC/PO,PABAR,,/1 $
45. MPYAD GO,PO,PABAR/PA/1 $
46. CHPNT AVEC,PO,PABAR,PA $
47. LABEL LBLO $
48. EQUIV PA,PL/REACT $
49. CHPNT PL $
50. COND LBLR,REACT $
$----- REACT
51. VEC USET/RVEC/C,N,A/C,N,L/C,N,R $
52. PARTN PA,,RVEC/PL,PR,,/1 $
53. MPYAD DM,PL,PR/QR/1/-1/-1 $
54. LABEL LBLR $
55. OUTPUT1 PA,,,,//0/0 $
56. MATGPR GPL,uset,sil,ps//c,n,s/c,n,s//c,y,psmin=0.0$
57. MATGPR GPL,uset,sil,qr//c,n,s/c,n,s//c,y,qrmin=0.0$
58. PRTPARM //c,n,o $
59. ALTER 104
60. CHPNT UOOV $
61. ALTER 105,105
62. ALTER 111,111
63. ALTER 137
64. LABEL LBL$ING $
65. PRTPARM //c,n,o/c,n,sing $
66. PRTPARM //c,n,o/c,n,mind $

```

DMAP ALTER PHASE III

```

$----- PHASE III DMAP ALTER FOR SMA RUN
$----- D. ZEMER
$---- IF ASET CARDS WERE REPLACED IN PHASE I BY SUPORT CARDS,
$---- THEY MUST NOW BE INCLUDED AND THE SUPORT CARDS DELETED.

1. ALTER 3
2. PARAM //C,N,SUB/V,N,UTSKP/V,Y,SUBST=0/C,N,1 $
3. COND LBLE1,UTSKP $
4. PARAM //MPY/V,N,NSKIP/1/1 $
5. INPUTT1 /,,,/C,N,-1/C,N,0/V,Y,OUTTAP=PHII $
6. ALTER 20,53
7. GP4 CASECC,GEOM4,EQEXIN,SIL,GPDT,BGPDT,CSTM/RRS, YYS, USET,AASET/V,N,
      LUSET/V,N,MPCF1/V,N,MPCF2/V,N,SINGLE/V,N,OMIT/V,N,REACT/O/
      V,N,REPEAT/V,N,NOSET/V,N,NOL/V,N,NOA/C,Y,SUBID $
8. ALTER 54,109
9. INPUTT1 /ULV,,,/V,N,UTSKP/C,N,/V,Y,OUTTAP $
10. ALTER 112,117
11. ALTER 122,126
12. ALTER 128,137
13. LABEL LBLE1 $
14. PRTPARM // C,N,0/C,N,SUBST $

```

APPENDIX B

DESCRIPTION OF DMAP ALTER FOR A PRELIMINARY STATIC SOLUTION
DURING A PHASE I SUBSTRUCTURING ANALYSIS

11. UPARTN performs a symmetric partitioning of KFF

$$\begin{bmatrix} \text{KFF} \end{bmatrix} = \begin{bmatrix} \text{KOO} & | & \text{KOA} \\ \text{---} & | & \text{---} \\ \text{KAO} & | & \text{KAAB} \end{bmatrix}$$

13. DECOMP decomposes KOO

$$\begin{bmatrix} \text{KOO} \end{bmatrix} = \begin{bmatrix} \text{LOO} \end{bmatrix} \begin{bmatrix} \text{UOO} \end{bmatrix}$$

18. FBS solves the matrix equation

$$\begin{bmatrix} \text{LOO} \end{bmatrix} \begin{bmatrix} \text{KOA} \end{bmatrix} = \begin{bmatrix} \text{GO} \end{bmatrix}$$

for the transformation matrix $\begin{bmatrix} \text{GO} \end{bmatrix}$

20. MPYAD multiplies $\begin{bmatrix} \text{KOA} \end{bmatrix}^T$ and $\begin{bmatrix} \text{GO} \end{bmatrix}$
together and adds $\begin{bmatrix} \text{KAAB} \end{bmatrix}$

$$\begin{bmatrix} \text{KAA} \end{bmatrix} = \begin{bmatrix} \text{KOA} \end{bmatrix}^T \begin{bmatrix} \text{GO} \end{bmatrix} + \begin{bmatrix} \text{KAAB} \end{bmatrix}$$

23. OUTPUT1 puts the psuedostructure information and stiffness matrix onto a user tape for use in Phase II.

26. If no m's are present, go to DMAP 31, Label LBLM

----- MPC reduction if m's are present -----

27. VEC creates a partitioning vector, MVEC, based on Multipoint Constraints.

28. PARTN partitions the global load matrix

$$\begin{bmatrix} \text{PG} \end{bmatrix} = \begin{matrix} \uparrow \\ \text{MVEC} \\ \downarrow \end{matrix} \begin{bmatrix} \text{PNBAR} \\ \text{---} \\ \text{PM} \end{bmatrix}$$

29. MPYAD creates a "N" set loading matrix

$$\begin{bmatrix} \text{PN} \end{bmatrix} = \begin{bmatrix} \text{GM} \end{bmatrix}^T \begin{bmatrix} \text{PM} \end{bmatrix} + \begin{bmatrix} \text{PNBAR} \end{bmatrix}$$

31. Label LBLM \$

32. If no s's are present, equivalence $[PF]$ to $[PN]$ and go to DMAP No. 39,
and label LBLS.
34.

----- SPC reduction if s's are present -----

35. VEC creates a partitioning vector, SVEC, based on single point constraints.

36. PARTN partitions the "N" load matrix

$$[PN] = \begin{array}{c} \uparrow \\ \text{SVEC} \\ \downarrow \end{array} \begin{bmatrix} \text{PFBAR} \\ \text{PS} \end{bmatrix}$$

37. MPYAD creates a "F" set loading matrix

$$[PF] = [KFS]^T [YS] + [PFBAR]$$

39. Label LBLS \$

40. If no o's are present, equivalence $[PA]$ to $[PF]$ and go to DMAP No. 47,
and label LBLO.
42.

----- OMIT reduction if o's are present -----

43. VEC creates a partitioning vector, AVEC, based on the ASET degrees of freedom.

44. PARTN partitions the "F" load matrix

$$[PF] = \begin{array}{c} \uparrow \\ \text{AVEC} \\ \downarrow \end{array} \begin{bmatrix} \text{PO} \\ \text{PABAR} \end{bmatrix}$$

45. MPYAD creates an "A" set loading matrix

$$[PA] = [GO]^T [PO] + [PABAR]$$

47. Label LBLO \$

48. If no l's are present, equivalence $[PL]$ to $[PA]$ and go to DMAP No. 54,
and label LBLR.
50.

-----REACTION reduction if 1's are present -----

51. VEC creates a partitioning vector, RVEC, based on the SUPORT degrees of freedom.
52. PARTN partitions the "A" load matrix

$$\begin{bmatrix} \text{PA} \end{bmatrix} = \text{RVEC} \begin{bmatrix} \text{PL} \\ - \\ \text{PR} \end{bmatrix}$$

53. MPYAD creates a "L" set loading matrix which is the reaction vectors on the support points.

$$\begin{bmatrix} \text{QR} \end{bmatrix} = - \begin{bmatrix} \text{DM} \end{bmatrix}^T \begin{bmatrix} \text{PL} \end{bmatrix} - \begin{bmatrix} \text{PR} \end{bmatrix}$$

54. Label LBLR \$
55. OUTPUT1 puts the load matrix of the ASET and SUPORT points onto a user tape for use in Phase II.

•SUBSTRUCTURE N, N=1,8

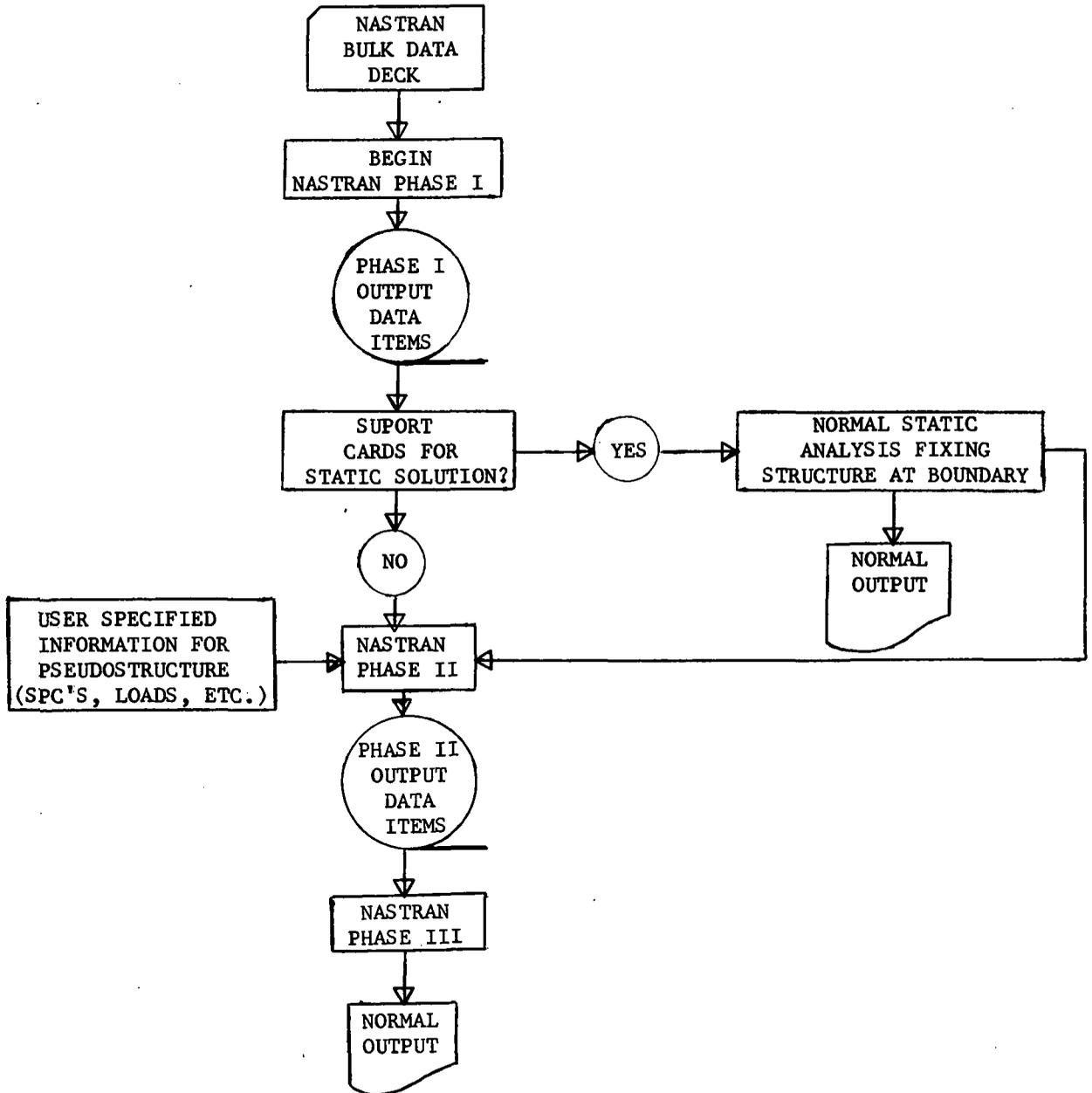
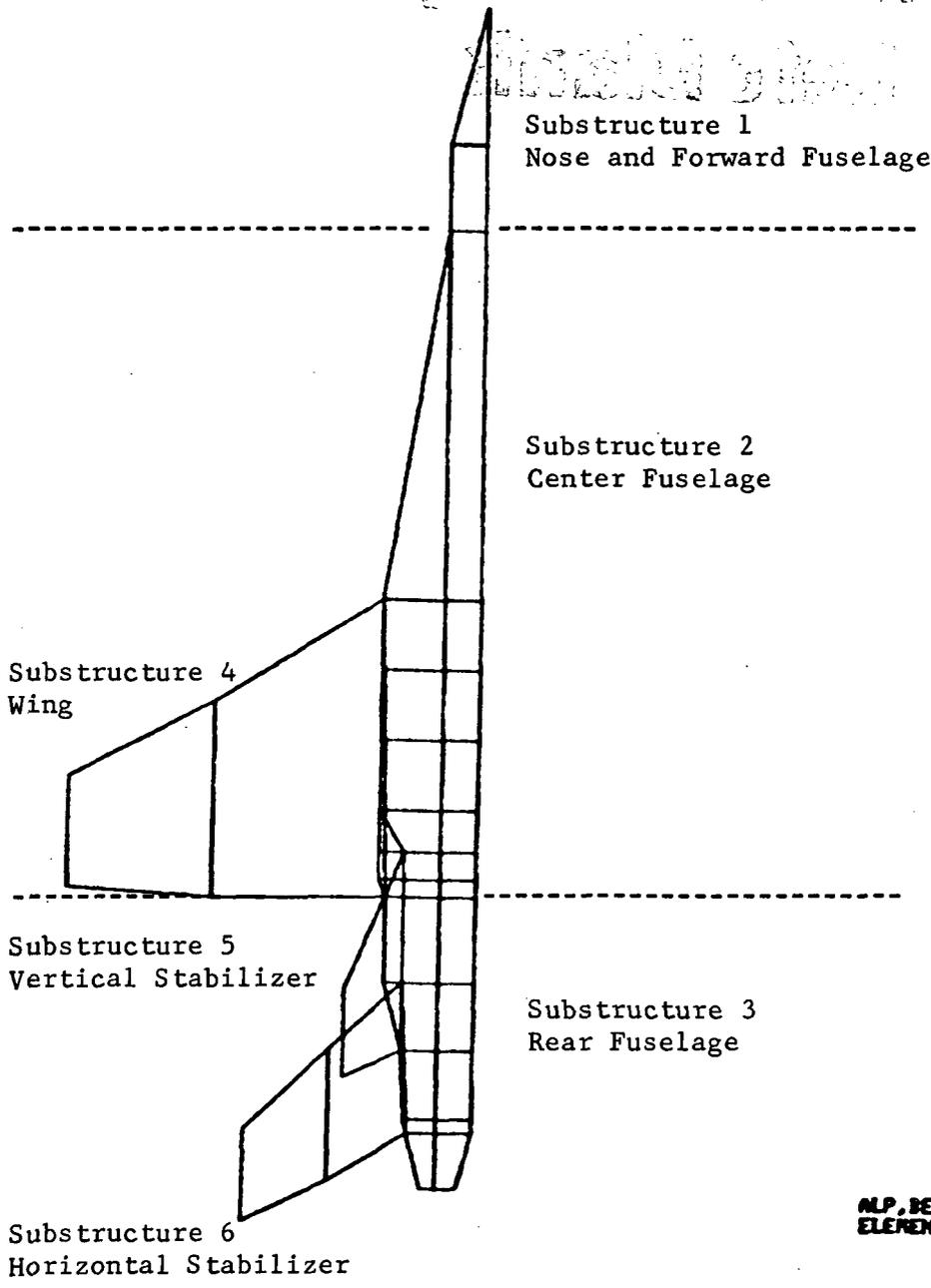


FIGURE 1 - FLOW DIAGRAM FOR NASTRAN SUBSTRUCTURING WITH A STATIC SOLUTION IN PHASE I



- 1 ZOOM
- 2 NEW ROTATION
- 3 NEW SET
- 4 GRID POINT LABELS
- 5 ELEMENT LABELS
- 6 PLOT AXES
- 7 HIDDEN LINE PLOT
- 8 REPAINT PLOT
- 9 PANEL ARROWS
- 10 MOVE
- 11 END
- ?

ALP,BET,GAM= 0.0 90.00 0.0
ELEMENT TYPES PLOTTED: ALL

FIGURE 2 - FIGHTER SUBSTRUCTURE PROTOTYPE

**Page
Intentionally
Left Blank**

A NASTRAN DMAP ALTER FOR THE COUPLING OF MODAL AND
PHYSICAL COORDINATE SUBSTRUCTURES

Thomas L. Wilson
Fairchild Space and Electronics Company

SUMMARY

A method is described to derive a generalized coordinate model consisting of flexible cantilever modes and rigid body modes from a physical coordinate model using a DMAP alter to Rigid Format 3. This model can readily be coupled to other substructures using modal synthesis techniques. It allows the use of a reduced size model for structural analyses while maintaining the capability of recovering the accelerations, forces, stresses, etc., from the original, large, complex model. This output recovery is accomplished with the use of a loads transformation matrix (LTM) which relates the output parameters to the modal coordinate accelerations.

In addition, a method is described to synthesize structural models consisting of hybrid coordinates for use in dynamic response analyses where one structure is described using physical coordinates and the other using generalized modal coordinates.

INTRODUCTION

The motivation for deriving the techniques of model reduction and synthesis stemmed from a series of transfer orbit loads analyses performed in support of the Global Positioning System. The spacecraft models were complex, three-dimensional models which had approximately 2000 degrees-of-freedom but which were reduced to approximately 400 degrees-of-freedom for dynamic analyses using the Guyan reduction (ref. 1). The transfer stage was a simplified, centerline model consisting of approximately 50 degrees-of-freedom. The necessity to perform the loads analyses within a reasonable time and expense required the reduction of the spacecraft model to a much smaller size. The reduction of the spacecraft to model coordinates consisting of flexible cantilever modes and rigid body modes about the interface (refs. 2 and 3) resulted in a significant size reduction while maintaining the capability of recovering the detailed acceleration, displacements, forces, stresses, etc.

SYMBOLS

F force matrix or vector
I identity matrix
K stiffness matrix

M mass matrix
 Q,q generalized displacement vector
 R coefficient matrices in interface compatibility definition
 T coupling transformation matrix
 u,X,x physical displacement vectors
 λ eigenvalues
 ϕ coordinate transformation matrix
 ϕ mode shape

Subscripts:

A analysis degrees-of-freedom
 B boundary partition of mass and stiffness matrices
 GEN generalized (mass or stiffness) matrix
 H generalized coordinates
 L degrees-of-freedom except boundary
 R boundary (interface) degrees-of-freedom
 u,x physical coordinates

Superscript:

T matrix transpose

GENERATION OF MODAL COORDINATE MODEL

Theoretical Description

The equations of motion of the free-free model in terms of the analysis degrees-of-freedom are:

$$\left[M_{AA} \right] \left\{ \ddot{U}_A \right\} + \left[K_{AA} \right] \left\{ U_A \right\} = \left\{ F(t) \right\} \quad (1)$$

which can be written in expanded form as:

$$\begin{bmatrix} M_{RR} & M_{RL} \\ M_{LR} & M_{LL} \end{bmatrix} \begin{Bmatrix} \ddot{U}_R \\ \ddot{U}_L \end{Bmatrix} + \begin{bmatrix} K_{RR} & K_{RL} \\ K_{LR} & K_{LL} \end{bmatrix} \begin{Bmatrix} U_R \\ U_L \end{Bmatrix} = \{F(t)\} \quad (2)$$

where U_R are the boundary coordinates. With U_R constrained, the eigenproblem

$$\left[K_{LL} - \lambda M_{LL} \right] \{U_L\} = 0 \quad (3)$$

yields the cantilever eigenvalues, λ , and mode shapes, ϕ_L , with ϕ_L normalized such that,

$$\{\phi_L\}^T [M_{LL}] \{\phi_L\} = [I] \quad (4)$$

The influence of the boundary coordinates on the unrestrained coordinates is expressed as:

$$\{U_L\} = -[K_{LL}]^{-1} [K_{LR}] \{U_R\} = \{\phi_R\} \{U_R\} \quad (5)$$

where

$$\{\phi_R\} = -[K_{LL}]^{-1} [K_{LR}]$$

The resulting transformation from physical to modal coordinates is

$$\begin{Bmatrix} U_R \\ U_L \end{Bmatrix} = \underbrace{\begin{bmatrix} I \\ \phi_R \end{bmatrix}}_{\text{Rigid body modes}} \quad \underbrace{\begin{bmatrix} 0 \\ \phi_L \end{bmatrix}}_{\text{Cantilever modes}} \begin{Bmatrix} U_R \\ q_H \end{Bmatrix} \quad (6)$$

or

$$\{U_A\} = [\phi] \{q\} \quad (7)$$

The rigid body modes shown in equation (6) are the mode shapes resulting from a unit displacement of each constraint with all other constraints fixed. This maintains the boundary coordinates as physical coordinates in the generalized model which simplifies the synthesis operation.

The generalized mass and stiffness arrays can then be written

$$M_{\text{GEN}} = \begin{bmatrix} M_{\text{BB}} & M_{\text{BH}} \\ M_{\text{HB}} & I \end{bmatrix} \quad (8)$$

$$K_{\text{GEN}} = \begin{bmatrix} K_{\text{BB}} & 0 \\ 0 & \lambda \end{bmatrix} \quad (9)$$

where

$$[M_{\text{BB}}] = [M_{\text{RR}}] + [M_{\text{RL}}] \{\phi_{\text{R}}\} + \{\phi_{\text{R}}\}^T [M_{\text{LR}}] + \{\phi_{\text{R}}\}^T [M_{\text{LL}}] \{\phi_{\text{R}}\} \quad (10)$$

$$[M_{\text{HB}}] = \{\phi_{\text{L}}\}^T \left([M_{\text{LL}}] \{\phi_{\text{R}}\} + [M_{\text{LR}}] \right) \quad (11)$$

$$[M_{\text{BH}}] = [M_{\text{HB}}]^T \quad (12)$$

$$[K_{\text{BB}}] = [K_{\text{RR}}] + \{\phi_{\text{R}}\} [K_{\text{RL}}] \quad (13)$$

If the boundary is determinant, then K_{BB} is null. If the boundary consists of 6 degrees-of-freedom, 3 translations and 3 rotations, then M_{BB} is the rigid body inertia matrix with respect to the boundary.

The recovery of accelerations, displacements, forces, and stresses from the original model is accomplished by a loads transformation matrix which relates the output parameters to the modal coordinate accelerations, \ddot{q} . Acceleration parameters are simply partitioned from the transformation matrix because from equation (7) it follows that

$$\{\ddot{U}_{\text{A}}\} = \begin{bmatrix} \phi \\ \end{bmatrix} \{\ddot{q}\} \quad (14)$$

For forces and stresses (which are recovered in NASTRAN via the displacements) and displacements, it is necessary to determine the inertia loads on the structure with respect to the boundary due to a unit acceleration on each mode,

$$[F_{\text{LL}}] = -[M_{\text{LL}}] \begin{Bmatrix} \phi_{\text{L}} \\ \vdots \\ \phi_{\text{R}} \end{Bmatrix} \{\ddot{q}\} \quad (15)$$

where $\ddot{q} = I$ (unit accelerations). A solution of the equation

$$[K_{\text{LL}}] \{U_{\text{L}}\} = [F_{\text{LL}}] \quad (16)$$

yields the displacements, U_{L} , of the physical model due to unit modal accelerations from which element forces and stresses can be recovered.

DMAP ALTER

1. ALTER 89, 89
2. READ KLL, MLL,,,EED, USET, CASECC/LAMA, PHIL, MI, OEIGS/C,N, MODES/V,N, NEIGV \$
3. ALTER 91
4. UMERGE USET, PHIL, /PHILA/C, N, A/C, N,L/C, N,R \$
5. UMERGE USET, DM, AMR/PHIRB/C,N,A/C, N,L/C,N,R \$
6. MERGE PHIRB,, PHILA,,MP,/PHIA/C,N,1/C,N,2/C,N,2 \$
7. ALTER 96
8. MPYAD MLL, DM, MLR/MIPL/C,N,0/C,N,1/C,N,1/C,N,2 \$
9. MPYAD PHIL, MIPL,/MHB/C,N,1/C,N,1/C,N,0/C,N,2 \$
10. TRNSP MHB/MBH \$
11. MERGE MR, MHB, MBH, MI, MP,/ MGEN \$
12. CHPNT MGEN \$
13. SMPYAD PHIL, KLL, PHIL,,,/KWW/C,N,3/C,N,1/C,N,1/C,N,2/C,N,2 \$
14. TRNSP KLR/KRL \$
15. MPYAD DM,KRL,KRR/KBB/C,N,0/C,N,1/C,N,1/C,N,2 \$
16. MERGE KBB,,,KWW,MP, /KGEN \$
17. CHPNT KGEN \$
18. PARTN PHIG,,ACP/, ACCL,,/C,N,1 \$
19. MERGE DM,,PHIL,,MP,/PHIC/C,N,1/C,N,2/C,N,2 \$
20. MPYAD MLL,PHIC,/LFOR/C,N,0/C,N,-1/C,N,0/C,N,2 \$
21. FBS LLL,ULL,LFOR/LDISP/C,N,1/C,N,1/ C,N,2/C,N,2 \$
22. CHPNT LFOR, LDISP \$
23. UMERGE USET, LDISP,/ADISP/C,N,A/C,N,L/C,N,R \$
24. SDRL USET,,ADISP,,,GO, GM,,KFS,,/PHIZ,,QGI/C,N,1/C,N,REIG \$
25. PARTN PHIZ,,DSP/,DISP,,/C,N,1 \$

26. CHKPNT PHIZ, QGI \$

27. ALTER 105, 107

28. SDR2 CASECC, CSTM, MPT, DIT, EQEXIN,SIL,,,BGPDP, LAMA, QGI
PHIZ, EST,/OQG2, OPHIZ, OES2, OEF2, PPHIG/C,N,REIG \$

29. CHKPNT OES2, OEF2 \$

30. OFP OES2, OEF2,,, //V,N,CARDNO \$

31. SAVE CARDNO \$

32. TBMT OEF2, OES2/OLTF2, OLTS2 \$

33. CHKPNT OLTF2, OLTS2 \$

34. MERGE OLTF2, OLTS2,,,,FSP/ FSLTM/C,N,1/C,N,2/C,N,2 \$

35. MERGE ACCL, DISP,,,,ADP/ADLTM/C,N,1/C,N,2/C,N,2 \$

36. MERGE ADLTM, FSLTM,,,,LTMP/LTM/C,N,1/C,N,2/C,N,2 \$

37. CHKPNT LTM \$

38. OUTPUT2 MGEN, KGEN, LTM, //C,N,-1/C,N,11/C,N, SCMODL \$

39. ENDALTER

DESCRIPTION OF DMAP ALTER STATEMENTS

2. The constrained mass (MLL) and stiffness (KLL) arrays are passed to the eigenvalue-eigenvector module to calculate cantilever modes (PHIL) and eigenvalues (LAMA).
4. The cantilever mode shapes are brought to A-set size by merging zeros into the boundary coordinates.
5. The rigid body modes are formed by merging DM ($=\phi_r$) and AMR which is an identity matrix input through DMI bulk data. The size of AMR is equal to the number of boundary coordinates.
6. The rigid body and cantilever modes are merged together to form the transformation (see eqs. (6) and (7)). MP is a partitioning vector input through DMI bulk data. If there are r boundary coordinates and h cantilever modes, then MP is a column vector with $r + h$ rows, where the first r rows are equal to zero and the remaining h rows are non-zero.
- 8-11. These statements calculate and assemble the generalized mass matrix described in equations (8), (10), (11), and (12).

- 13-16. These statements calculate and assemble the generalized stiffness matrix described in equations (9) and (13). Statement 13 calculates the λ array because LAMA which is output from Module READ is in tabular form and cannot be used as a matrix.
18. The full size (G-set) mode shape array is partitioned to extract the desired acceleration recovery parameters for the LTM (eq. (14)). ACP is a partitioning vector input through DMI bulk data.
- 19-21. The modal forces and resulting displacements are calculated as described in equations (15) and (16).
- 23-25. The displacements calculated in step 24 are brought to G-set size (PHIZ) and then partitioned to extract the desired displacement parameters for the LTM. DSP is a partitioning vector input through DMI bulk data.
28. The desired element forces and stresses are calculated using the displacements (PHIZ) resulting from unit modal coordinate accelerations.
32. TBMT is a dummy module which moves the table forms of element forces and stresses to matrix forms for later assembly into the LTM.
- 37-39. The accelerations, displacements, forces, and stresses are merged together to form the LTM.
41. The generalized mass matrix, stiffness matrix and LTM are stored on tape or disk for later use in the synthesis and response analyses.

Additional Data Requirements

Executive Deck: DIAG 21 and 22 are very useful in determining the partitioning vectors for acceleration and displacement LTM parameters.

Case Control Deck: In addition to the normal eigenvalue data, an appropriate ELFORCE and ELSTRESS specification must be made for those elements desired in the LTM.

Bulk Data Deck: In addition to the partitioning vectors already discussed and the normal finite element model, the SUPORT option for all of the boundary coordinates must be specified. The EIGR card should specify the number of cantilever modes desired and the normalization should be with respect to mass.

COUPLING OF HYBRID COORDINATE SUBSTRUCTURES

This section describes a method used to couple hybrid coordinate substructures, where one substructure consists of modal coordinates as derived previously and the other is a physical coordinate substructure. The coupling is achieved by enforcing displacement compatibility at the boundary between the two substructures.

Theoretical Description

The modal coordinate substructure is described using a set of coordinates, q , where

$$\{q\} = \begin{Bmatrix} U_R \\ q_H \end{Bmatrix} \quad (17)$$

where U_R are the boundary coordinates and q_H are the generalized coordinates corresponding to the flexible cantilever modes of the structure.

The mass and stiffness matrices are M_{GEN} and K_{GEN} , described in equations (8) and (9). The physical coordinate substructure consists of a set of coordinates, X , where

$$\{X\} = \begin{Bmatrix} X_R \\ X_L \end{Bmatrix} \quad (18)$$

where X_R are the coordinates of the interface with the other substructure and X_L are the remaining degrees-of-freedom.

The corresponding mass and stiffness matrices are M_X and K_X .

The coupled model is derived by enforcing displacement compatibility of the interface coordinates

$$\begin{bmatrix} R_X \end{bmatrix} \{X_R\} + \begin{bmatrix} R_U \end{bmatrix} \{U_R\} = 0 \quad (19)$$

The total set of uncoupled coordinates is

$$\{\bar{Q}\} = \begin{Bmatrix} U_R \\ q_H \\ X_R \\ X_L \end{Bmatrix} \quad (20)$$

The uncoupled mass and stiffness matrices are

$$\begin{bmatrix} \bar{M} \end{bmatrix} = \begin{bmatrix} M_{GEN} & 0 \\ 0 & M_X \end{bmatrix} \quad (21)$$

$$\begin{bmatrix} \bar{K} \end{bmatrix} = \begin{bmatrix} K_{GEN} & 0 \\ 0 & K_x \end{bmatrix} \quad (22)$$

The transformation from the uncoupled to the coupled coordinates is

$$\{\bar{Q}\} = [T] \{Q\} \quad (23)$$

where

$$[T] = \begin{bmatrix} -[R_x]^{-1} [R_u] \\ I \end{bmatrix} \quad (24)$$

and

$$\{Q\} = \begin{Bmatrix} U_R \\ q_H \\ x_L \end{Bmatrix} \quad (25)$$

The coupled mass and stiffness matrices are

$$[M] = [T]^T [\bar{M}] [T] \quad (26)$$

and

$$[K] = [T]^T [\bar{K}] [T] \quad (27)$$

which can be used in modal and dynamic response analyses.

Once the response analysis is complete and the coupled model accelerations \ddot{Q} , are calculated, the desired output is recovered as follows:

(a) The physical coordinate substructure output is recovered via standard NASTRAN output requests.

(b) The modal coordinate substructure output is recovered by partitioning the modal coordinate accelerations, \ddot{q} , from the total coordinate acceleration vector, \ddot{Q} , and performing the matrix multiplication

$$\text{OUTPUT} = \left[\text{LTM} \right] \left\{ \ddot{q} \right\} \quad (28)$$

to obtain each parameter value at each time or frequency.

DMAP ALTER

1. ALTER 45
2. INPUT 2/MGEN, KGEN, LTM, /C,N,-1/C,N,11/C,N, SCMODL \$
3. PARAM // C,N,NOP/V,N,NDPM = -1 \$
4. MERGE MGEN, , , , PA, / MG \$
5. MERGE KGEN, , , , PA, / KG \$
6. ADD MGG, MG/MGT \$
7. EQUIV MGT, MGG/NDPM \$
8. ADD KGG, KG/KGT \$
9. EQUIV KGT, KGG/NDPM \$
10. ALTER 151
11. PARIN UPV, , PA/UPVSV, , /C,N,1 \$
12. PARIN UPVSV, ACPR, / , , SVA, /C,N,1/C,N,2/C,N,2/C,N,2/C,N,2 \$
13. MPYAD LTM, SVA, /SVP/C,N,0/C,N,1/C,N,0/C,N,2 \$
14. CHKPNT SVP \$
15. MATPRN SVP, , , , // \$
16. ENDALTER

NOTE: This alter was written for Rigid Format 12, but with slight modifications it is applicable to other rigid formats.

Description of DMAP Alter Statements

NOTE: Space for the generalized substructure is created by defining scalar points 1 through n, where n is equal to the total number of coordinates contained in q.

2. The generalized mass and stiffness matrices and LTM of the modal coordinate substructure are read from the tape or disk where they were previously stored.
3. A parameter, NDPM, is created for later use in Equivalence statements.
- 4-5. The generalized mass and stiffness matrices are merged into a square matrix which is the size of the total set of uncoupled coordinates, Q. The matrices MG and KG are

$$[MG] = \begin{bmatrix} MGEN & 0 \\ 0 & 0 \end{bmatrix}$$

and

$$[KG] = \begin{bmatrix} KGEN & 0 \\ 0 & 0 \end{bmatrix}$$

The partitioning vector PA is input via DMI bulk data.

6. Matrix MG is added to matrix MGG which contains the mass of the physical coordinate model.

$$[MGT] = \underbrace{\begin{bmatrix} MGEN & 0 \\ 0 & 0 \end{bmatrix}}_{MG} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & M_x \end{bmatrix}}_{MGG}$$

7. MGG is made equivalent to MGT.
- 8-9. Steps 6 and 7 are repeated for the stiffness matrices.
11. The generalized coordinate modal responses are partitioned from the total set of calculated responses, UPV.
12. The acceleration response is partitioned from the total response. Responses are stored in UPV in three columns per time step, one column each for displacement, velocity, and acceleration. ACPR is a partitioning vector input through DMI bulk data with every third row non-zero to extract the acceleration data.
13. The modal coordinate substructure responses are calculated as in equation (28).

NOTE: The definition of interface compatibility (eq. (19)) and the transformation to coupled mass and stiffness matrices (eqs. (23) through (27)) are accomplished by the use of a set of MPC equations.

Additional Data Requirements

Case Control Deck: An MPC set must be specified which contains the definition of interface compatibility (see eq. (19)). Any desired output for the physical coordinate substructure should be specified using standard output requests. The standard dynamic load requests must be specified.

Bulk Data: The normal finite element model input for the physical coordinate substructure and dynamic load description is required. The partitioning vectors described previously must be input using DMI. Scalar points 1 through n must be defined, where n is equal to the total number of coordinates contained in q .

References

1. Guyan, R. J.: Reduction of Stiffness and Mass Matrices. AIAA Journal, Vol. 3, No. 2, February 1965, p. 380.
2. Hurty, Walter C.: Dynamic Analysis of Structural Systems Using Component Modes. AIAA Journal, Vol. 3, No. 4, April 1965, pp. 678-685.
3. Craig, Roy R., Jr.; and Bampton, Mervyn C. C.. Coupling of Substructures for Dynamic Analyses. AIAA Journal, Vol. 6, No. 7, July 1968, pp. 1313-1319.

APPLICATION OF NASTRAN LEVEL 16.0 AUTOMATED

MULTISTAGE SUBSTRUCTURING TECHNIQUES

R. M. Berezna* and C. S. Sahota*

Rockwell International

ABSTRACT

The NASTRAN Level 16.0 automated multistage substructuring technique was utilized in joining two large substructures of the B-1 wing modeled for NASTRAN to the B-1 wing carry-through structure modeled for ASKA finite element program (fig. 1). The two substructures representing the NASTRAN wing outer panel model have approximately 14 500 degrees of freedom. The multi-substructured ASKA model of the wing carry-through and pivot structure have approximately 30 000 degrees of freedom.

The stiffness and load vectors for these large structural models were reduced to their common boundary nodes. To combine the reduced matrices, two fictitious substructures were defined in the NASTRAN format to represent the wing carry-through and pivot assembly. The fictitious substructure concept was created for boundary interactions between NASTRAN and ASKA to use the transformation matrix capability in the NASTRAN program for various sweep angles of the wing. These techniques, which were used to interface the NASTRAN structural model to the ASKA model in obtaining a common boundary solution, are discussed.

This technical paper will illustrate how the NASTRAN multistage substructuring program significantly contributed to the B-1 wing and wing carry-through internal structural loads analysis used in the detailed stress analysis.

*Member of technical staff.

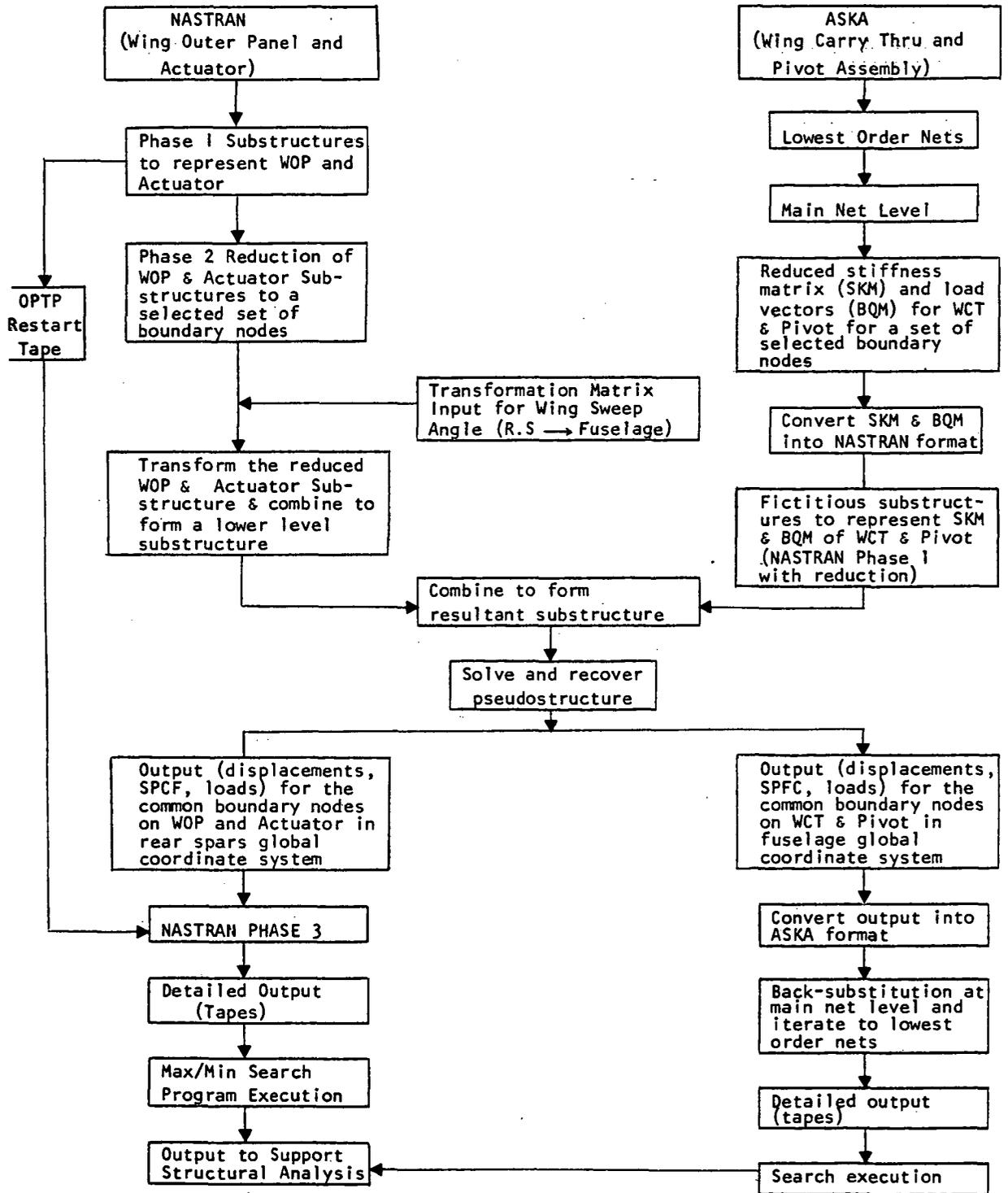


Figure 1. Flow Chart for NASTRAN Wing Model Interface with ASKA Wing Carry Thru Model

AUTOMATED MULTI-LEVEL SUBSTRUCTURING FOR SPERRY'S VERSION OF NASTRAN

Gordon C. Chan and Ronald P. Schmitz
Sperry Rand Corporation

SUMMARY

Automated multi-level substructuring has been successfully implemented in the Sperry version of NASTRAN. The development was achieved with a minimum effort; only seven modules and fourteen new routines were added. Approximately forty subroutines were modified and four new user cards added to NASTRAN. Significant features of the new capability include:

- Multi-level substructuring
- Up to 128 superelements
- Up to 456 boundary grid points per superelement
- Up to 9 user superelement files, each with multi-reel or disc option
- Data storage for one or more superelements per user file
- Congruent, rotated, and reflected symmetry superelements
- Automated MPC, SPC, SPC1, SEQGP, etc. constraint handling and grid point resequencing
- Allowance for equivalent boundary grid points among superelements
- Input data checking and extensive diagnostic messages
- Selective restart option by substructuring phase(s).
 - Phase I - Superelement generation
 - Phase II - Solution of the pseudo structure
 - Phase III - Stress and force recovery
- Application to static and dynamic analyses
- Superelement static Rigid Format

INTRODUCTION

The virtues of substructuring are well known (ref. 1, 2, 3, 4) and the desire for such capability, in the NASTRAN program has been acknowledged since early releases of the Program (ref. 5, 6). Many users have successfully developed sophisticated DMAP programs (ref. 7, 8, 9) for substructuring, and NASA provided documentation for manual one level substructuring with its release of NASTRAN Level 15.1 (ref. 10, 11). The substructuring capability was further extended to automated multi-level substructuring (ref. 12) with the release of NASTRAN Level 16.

However, when the FEDD policy of export restrictions (ref. 12) were placed in Level 16 by NASA Headquarters, it became necessary for Sperry to develop its own substructuring capability in a special Sperry Version of

NASTRAN. Funding for this effort was provided by the Sperry UNIVAC Division and all technical development was performed by the NASTRAN Maintenance Group of the Sperry Huntsville Engineering Operation, Sperry Rand Corporation.

This paper describes the overall capabilities of Sperry's automated multi-level substructuring and discusses several of the significant programming aspects, and future developments planned for the program.

SUBSTRUCTURING PHILOSOPHY

In general, the method used by Sperry involves the concept of "super-elements" (ref. 1, 13). This method allows us to utilize the existing NASTRAN capabilities and program philosophy. Every effort was made to minimize the user interface for substructuring and to develop a truly automated method. As a result, only three new Bulk Data cards and one Executive Control card were added to the program.

In order to implement the simple user requirements, extensive programming changes were required. Seven new modules, and fourteen new subroutines were added, and approximately forty existing subroutines were modified. All coding was achieved in compliance with the NASTRAN Fortran guidelines (ref. 12).

One of the basic principles employed in the superelement approach of substructuring is to treat a substructure component as a simple element. The superelement stiffness matrix is then treated as a simple element such as a CBAR, CTRIA2, or CIS3D20, A 20 node isoparametric element (ref. 14). A CBAR element has two grid points, and therefore produces a 12 x 12 stiffness matrix. The stiffness matrix assembler, SMA1, will accept this 12 x 12 matrix and add it to the system stiffness matrix of the overall structure. Similarly, the CTRIA2 element produces an 18 x 18 matrix, and CIS3D20 produces a 120 x 120 matrix. In each case, the SMA1 module will accept these matrices as they are generated. A superelement may have "n" grid points on its boundary, and therefore, produces a $(6n) \times (6n)$ stiffness matrix. The SMA1 module in general does not discriminate between simple element matrices and the matrix generated for the superelement. Mass matrices and damping matrices are treated similarly in module SMA2. In other words, the superelement is treated as a simple structural element, and the existing modules are utilized to assemble and manipulate the system matrices.

Secondly, the NASTRAN substructuring will handle multi-level superelements with only a minor modeling condition imposed upon the user; where the user must label each higher level superelement with a higher superelement ID number. Since all of the bulk data are sorted in ascending alpha-numeric order during the input phase, the lower-level superelements will be processed first due to their lower superelement ID values. By the time a higher-level superelement is processed, which requires lower-level superelement data, all data is available. There is no need to develop complicated code to process the case control or the executive control decks to achieve multi-level capability.

Finally, where necessary, data blocks will be modified to permit the use of existing NASTRAN modules. This is accomplished by deleting irrelevant data or masking unwanted information during the superelement generation.

SUBSTRUCTURING NEW CARDS

There were four new cards added to Sperry NASTRAN for multi-level substructuring. One card belongs to the executive control deck, and the remaining three to the bulk deck.

EXECUTIVE CONTROL DECK:

SUBSTRUCTURING	PHASE 0 (or \emptyset)
	PHASE I
	PHASE II
	PHASE III
	PHASES I AND II
	PHASES II AND III
	PHASES I, II, AND III (Default)

BULK DATA DECK:

SUPEREL	ID	BSET1	IN	OUT	(ELEM. ID'S LIST)		
BSET1		ID	(GRID POINT LIST)				
SAME1	I_1	J_1	I_2	J_2	I_3	J_3	...

The SUBSTRUCTURING card controls the Phase (0, I, II, or III) to be executed. (Phase 0 is for checking input cards only). The SUPEREL card describes the basic data needed for the superelement generation. The BSET1 card describes the boundary grid points for the corresponding superelement. The SAME1 card allows two or more boundary grid points to be equivalenced. The three new bulk data cards can be placed anywhere in the bulk data deck, and similarly the SUBSTRUCTURING card can be placed anywhere in the executive control deck. The absence of the SUPEREL cards and the SUBSTRUCTURING card in a NASTRAN input deck would make the deck a standard (non-substructuring) NASTRAN run. Appendix A describes the new input data cards in NASTRAN standard format.

SUBSTRUCTURING LIMITATIONS

Several limitations are imposed on the current release of Sperry NASTRAN Level 77.0. Generally they are not very severe, and as time and funding permit, will be removed or altered in future releases of the program.

1. Maximum of 128 superelements and up to 456 boundary grid points per element.
2. Maximum of 9 output tape drives available for the storage of superelement data blocks.
3. The substructure is identified by its superelement ID Number. No alphanumeric ID is allowed.
4. Six-degrees-of-freedom are maintained at each boundary grid point until the pseudo-model (highest level superelement) is formed or until the boundary point is placed in an interior grid point of a higher level superelement. Constraints may be applied to any interior degree of freedom.
5. Scalar points are not allowed in a boundary set.
6. All internal loads must be applied during Phase I superelement generation. Additional loads can be applied only to the grid points of the pseudo-model during subsequent restarts.

SUBSTRUCTURING METHOD

In order to implement NASTRAN substructuring, extensive changes were made in the preface (LINK1). This superelement preparation phase, which requires no user action, organizes the input data into appropriate data blocks, checks input data, and makes preparation for the three substructuring phases, PHASE I, PHASE II, and PHASE III. About forty percent of the total programming effort was spent in this preparation phase.

Input Data Checking and Superelement Preparation - Phase 0

The NASTRAN substructuring input file processor (IFP) will check input data consistency, and will set the NO-GO flag if data errors were made. The following items are checked by the IFP module:

1. Key words, such as SUBSTRUCTURING, SUPEREL, INP1, INP2, THRU, SAME, etc.
2. Validity of grid points and their existence are checked.

3. Existence of valid boundary grid points.
4. Existence of equivalent grid points on the boundary.
5. MPC, SPC, SPC1, OMIT, OMIT1, ASET, ASET1, SUPPORT, SEQGP input card check.
6. Superelement and boundary set cross reference check. A boundary set must exist for each superelement except for the highest-level superelement which may not have a boundary set.

After all of the input cards have been read by NASTRAN's IFP module, sorted, and stored in their appropriate files, GEOM1, GEOM2, MPT, etc., the superelement preparation phase is activated by the presence of a SUPEREL card in the input string. Element data previously stored on "original" files are now copied to a "new" file for permanent storage (e.g. GEOM1 and GEOM2 data are moved to GEOM6). Also, data on some files are temporarily masked out (their coded word changed) such that they cannot be located by NASTRAN during the current Phase I loop. This is done for the load and constraining input cards which are not associated with the current superelement, e.g., SPC, SPC1, FORCE, and SEQGP. Data in the GEOM6 file, which is used to store superelement input data, is altered with new data format and new data added for later use. Many checks are built in at this phase for early detection of possible input errors.

If the user has specified Phase 0 on the SUBSTRUCTURING card of the Executive Control deck, the run will be terminated at the end of the Preface (LINK1). Otherwise, if no error is found, the NO-GO flag remains off, and substructuring Phase I will begin.

Phase I

Substructuring begins by executing the SSPH1B (Substructuring Phase I Beginning) module. It then executes the table generation modules of GP1, GP2, GP3, GP4, and TA1. The superelement matrices (stiffness, mass, and damping) for the substructure are then assembled by the SMA1 and SMA2 modules. If general elements are present, the SMA3 module is executed. Next, the matrix partitioning, due to the presence of MPC and SPC constraints, is performed in modules MCE1, MCE2, and SCE1. Finally the matrices are reduced to those degrees-of-freedom specified by the analysis set, ASET/ASET1 in module SMP1, and the boundary static loads are generated in SSG1 and SSG2 modules. Phase I ends in module SSPH1E (Substructuring Phase I Ending).

There were three new modules developed for the substructuring Phase I. They are SSPH1B, SSPH1E, and GP1S, and their module functions are further described in this section. Two element routines, KSUPER and MSUPER were also developed for the SMA1 and SMA2 modules so that the previously computed superelement can be treated as a "library element" in multi-level substructuring.

SSPH1B Module

The main task of SSPH1B is to reconstitute all of the IFP-generated files with data pertaining only to the present superelement in question, and unmask those previously masked input data that are needed. To illustrate this point, let us assume that there were 2000 grid points in the original bulk data deck, and only 400 of these are needed for the current superelement computation. These 400 grid points, with their coordinates, will be written into the GEOM1 file, together with the required CORDi cards. Similarly, only those SEQGP data that are applicable to the 400 grid points will be placed into GEOM1. The connection cards, such as the simple elements, CBAR, CTRIA2, etc. are placed into GEOM2 only when they are required. The applicable constraint cards are unmasked. At the end of SSPH1B module, a summary table lists the items being brought back for the calculation of the superelement in question. Other forms of data checking are also provided by SSPH1B, e.g., grid points on the SPC and SPC1 cards are checked against the boundary grid list (BSET1), and if they are present, they will be masked from the constraint lists. These masked constraints will be unmasked at a higher superelement which does not contain the grid point in the BSET1 list. MPC cards are also checked. All OMIT, OMIT1, ASET, ASET1, and SUPPORT cards are ignored in all lower-level superelements and are unmasked during the calculation of the pseudo structure. Finally the superelement boundary set, BSET1, which is stored in the GEOM6 file, is copied to the GEOM4 file as an analysis set, ASET1, for the current superelement to be processed by NASTRAN.

If necessary, the load data in GEOM3 is also updated in SSPH1B to reflect any load contribution from previously computed lower-level superelements.

In summary, SSPH1B is an input file manipulator located immediately behind the NASTRAN preface operation, the IFP processor. All other NASTRAN modules will execute and perform their intended function as long as there is no detectable illegal data, or incorrect parameters present.

GP1S Module

A new module, GP1S was added to supplement the GP1 module, which generates the external-internal grid point table, EQEXIN, and the scalar index list, SIL. Since each superelement may be developed by a different user, the boundary grid points may be defined differently on each superelement by each user. A grid point equivalencing list is provided in substructuring via the SAME1 input card. GP1S was developed to process the SAME1 data, and to give the equivalent external grid points the same internal grid number. The updated EQEXIN and SIL tables are then used to generate subsequent data blocks, such as the element summary table, EST, the element connection table, ECT, the element connection and property table, ECPT, static load table, SLT, etc.

SMA1 and SMA2 Modules

KSUPER and MSUPER subroutines were added to the matrix assembler modules SMA1 and SMA2, to handle matrix 'formulation' for the lower-level super-elements. These routines are called each time a higher-level superelement is to have a lower-level superelement incorporated into its structural domain. During this assembly process, the lower superelement is treated as a 'library-resident' simple element. Instead of computing the necessary matrices as in the case of the simple structural element, beam, plate, or solid, the KSUPER and MSUPER open the SEK4MB file, search for the right record, and read in the K matrix (stiffness), K4 matrix (material damping), M matrix (mass), and the B matrix (viscous damping) for the previously calculated lower-level superelement. These matrices are read in a 6 x 6 matrix block format, which was recorded by row and is ready for immediate matrix assembling by SMA1 and SMA2.

SSPH1E Module

SSPH1E module stores the current superelement reduced matrices (K, K4, M, and B), in a 6 x 6 matrix block format at the end of the SEK4MB file. It also computes the reduced load vector, SEQA, due to the loads on the internal points (not on the boundary). From SEQA, load vector FORCE and MOMENT card images are generated and stored in the GEOM3 file, such that these forces and moments can be applied automatically to the appropriate higher-level super-element(s). Finally, SSPH1E copies the EQEXIN, SIL, GO and 19 other files to the user designated output file as specified in the OUT field of the SUPEREL card. All of these files will be recovered in Phase III for force and stress recovery of the current superelement.

The file name specified in the IN field of the SUPEREL card requires some explanation. If a superelement has been processed in a previous run, there is no need to repeat the calculations. The IN field designates the data location of previously calculated superelements which are to be recovered. It must point to one of the INPi (i = 1, 2, . . . , 9) files or tapes where the data has been stored. The SSPH1B module will open the designated file and transfer the [K], [K4], [M], and [B] matrices to the SEK4MB file, and the forces and moments are written to the GEOM3 file for later use.

Congruent superelements can be defined by an alternate SUPEREL input card. The IN and OUT fields contain the words 'SAMEAS' and 'SUPEREL', and the next four fields give the referenced superelement's ID, and three Euler angles of rotation, in degrees. A reflected superelement is similarly treated by using the words 'IMAGE' instead of 'SAMEAS' in the IN field of a SUPEREL input card.

If the current superelement is the highest-level, SSPH1E will be bypassed, and the Phase II operation will begin. Otherwise, SSPH1E is executed, and Phase I will continue by looping back for another superelement operation.

SSPH1B and SSPH1E work together in the task of synchronizing the external-internal relationships of the boundary grid points when the super-element is generated and the internal-external relationships when the super-element is used. The super-element is always referenced by its external grid points since the internal NASTRAN indices vary from super-element to super-element.

Phase II

Phase II begins with the SSPH2B module, followed by the pseudo structure displacement solution module, SSG3 (for static analysis), and ends with SSPH2E. One can execute Phase II directly by the SUBSTRUCTURING PHASE II, or II and III executive control card. Phase II computes the solution of the pseudo structure.

SSPH2B Module

SSPH2B uses data in GEOM6 and generates a super-element hierarchy table, SELEV, to be used for determining the order of super-element stress recovery in Phase III.

SSPH2E Module

SSPH2E is located immediately before Phase III and is used to signal the end of Phase II computation. Currently SSPH2E performs exactly the same function as SSPH3E in Phase III operations.

Phase III

Using the same nomenclature as Phase I, Phase III begins with the Substructuring Phase III Beginning module, SSPH3B, and ends with the Ending module, SSPH3E. In between the two modules, Phase III will execute the force and stress recovery modules SDR1 and SDR2 and satisfy all requests for output. A loop through Phase III will be repeated for each super-element whose forces and stresses are requested for output. Phase III has two modes of stress recovery: A selective mode where only those super-elements specified in a SUBSTRUCTURING PHASE III card (see following section) will be computed, and an automatic mode, where all super-elements in the problem are computed.

SSPH3B Module

The main task of SSPH3B is to restore the appropriate data blocks previously saved by SSPH1E to their original form, so that displacement and stresses can be recovered for a particular super-element. SSPH3B then extracts the proper displacement vector for the current super-element from the known displacement vector of a higher-level super-element, or the pseudo structure.

In the "selective" scheme of operation, SSPH3B consults first the SELEV table to determine whether or not a higher-level superelement needs to be solved prior to the solution of the requested superelement. If a higher-level superelement must be solved first, only the displacement vector is computed (not the forces and stresses). The results are then used in the lower-level superelement calculation.

In the "automatic" scheme of operation, the superelement hierarchy table, SELEV, is also consulted, but the computation automatically starts from the highest-level superelement and proceeds down the hierarchy table to the lower-level elements. This is done in the reverse order of Phase I, where the lower superelements were first generated.

SSPH3E Module

SSPH3E sorts the displacement vector according to the superelement external grid order and stores the data in the SELEV file. The entire SELEV file is also copied to the output file of the last (pseudo) superelement so that PHASE III can be restarted independently. The entire Phase III operation is repeated for each required superelement.

Rigid Format 15

A Rigid Format 15 has been developed for Sperry NASTRAN static analysis with superelements. A DMAP listing of this format is shown in Appendix B.

Rigid Format 15 is used in the example listed in Appendix C.

FUTURE DEVELOPMENTS

Improve Checkpoint/Restart

A fully automated restart capability for substructuring is planned. A normal restart tape, NPTP, will be set up, restart dictionary punched, and data saved on the NPTP tape as before. However, at the end of Phase I, all data files required for Phases II and III have been saved in the user designated output file and a large part of the data information on the NPTP tape becomes obsolete. The restart tape (NPTP) will be rewound, and advanced to the point where the data of a new superelement will be written over the data of an old superelement. Additional checkpoint dictionary cards will be punched replacing the previous set of cards. When a NASTRAN substructuring run is terminated due to an error or "max-time," the restart tape will contain the necessary restart data for the current superelement being calculated. The contents of the NPTP tape, therefore, will always remain small. Phase II will be checkpointed in a normal fashion. Phase III restart/checkpoint capability

will save each superelement solution vector as it is solved. If the problem makes an unscheduled stop, restarting will proceed from the last superelement successfully solved.

Additional Executive Control Option

The NASTRAN substructuring input card will be modified to add several new options. Tentatively it is planned to have the following format:

```
SUBSTRUCTURING      PHASE I, COMPUTES ONLY n1, n2, n3 .....
                    PHASE III, COMPUTES ONLY n1, n2, n3 .....
                    PHASE I, SKIPS n1, n2, n3 .....
                    where n1, n2, n3 ..... are superelement ID's
```

The default value for the "COMPUTES ONLY" is ALL, and that of "SKIPS" is NONE.

Load Re-generation

A load re-generation option will be implemented. In this option the internal load calculation in Phase I of the current substructuring will be removed, and in Phase II, a complete load generation loop will be programmed to permit changing loads without re-calculating the superelements.

Rigid Format 16

All of the modules required for dynamic analysis with substructuring have been incorporated into Sperry NASTRAN. A complete DMAP listing (rigid format 16) with restart will be developed and tested.

EMG/EMA With Substructuring

The new element matrix generation module, EMG, and assembler, EMA, can easily be modified to replace SMA1 and SMA2 operation with substructuring. The superelement matrices in SEK4MB file and the EMG generated files for the element stiffness and mass matrices are closely similar in data structure and logic. DMAP programs, similar to Rigid Formats 15 and 16, will be formulated with EMG/EMA operations.

REFERENCES

1. Przemieniecki, J.S.: Theory of Matrix Structural Analysis, McGraw-Hill, New York, 1968
2. Zienkiewicz, O.C.: The Finite Element Method in Engineering Science, McGraw-Hill, London, 1971
3. Narayanaswami, R.: Application of NASTRAN Substructuring, Second NASTRAN Users' Colloquium, NASA TM X-2637, 1972
4. MacNeal, R.H., ed.: The NASTRAN Theoretical Manual, NASA SP-221(03), March 1976
5. Minutes of the Informal NASTRAN Industry Advisory Panel, McDonnell Douglas, Long Beach, CA., November 28-29, 1972
6. Hennrich, C.W.; and Konrath, E.J.: Substructure Analysis Techniques and Automation, Third NASTRAN Users' Colloquium, NASA TM X-2893, September 1973
7. Filstrup, A.W.: Normal Mode Analysis of a Rotating Group of Lashed Turbine Blades by Substructures, Third NASTRAN Users' Colloquium, NASA TM X-2893, September 1973
8. Field, E.I., et al.: Automated Multi-Stage Substructuring System for NASTRAN, Fourth NASTRAN Users' Colloquium, NASA TM X-3278, September 1975
9. Case, W.R.: Dynamic Substructure Analysis of the International Ultraviolet Explorer (IUE) Spacecraft, Fourth NASTRAN Users' Colloquium, NASA TM X-3278, September 1975
10. McCormick, C.W., ed.: The NASTRAN Users' Manual, NASA SP-222 (01), May 1973 (Revised March 1976)
11. Levy, A.: NASTRAN Multipartitioning and "One-Shot" Substructuring, Third NASTRAN Users' Colloquium, NASA TM X-2893, September 1973
12. The NASTRAN Programmer Manual, NASA SP-223(03), July 1976
13. Aamodt, B., et al.: General Superelement Program (SESAM-69) Users' Manual, NV336, COMPUTAS Norway, 1972
14. Hurwitz, M.M.: Additions to the NASTRAN Users' Manual and Theoretical Manual for a Thermostructural Capability for NASTRAN Using Isoparametric Finite Element, NSRDC Report 4134, May 1973

APPENDIX A
SPERRY'S NASTRAN SUBSTRUCTURING NEW INPUT CARDS

EXECUTIVE CONTROL DECK

SUBSTRUCTURING K

Required when substructuring is requested.

K - Option request for substructuring phases

K = PHASE I

K = PHASE II

K = PHASE III

K = PHASES I and II

K = PHASES II and III

K = PHASES I, II, and III

K = YES (Same as PHASES I, II, and III)

SAVE INDIVIDUAL SUPERELEMENT CHECKPOINT RESTART TAPE (Not available)

Required if individual checkpoint restart tape is to be dismantled and saved for each superelement in a one-shot substructuring problem, during PHASE I operation. Otherwise, the checkpoint restart tape will be automatically rewound and new superelement data will be written on top of the old, and obsolete data of a previous computed superelement.

BULK DATA DECK

Input Data Card SUPEREL

Description: Defines the elements contained within the superelement (substructure).

Format and Example:

SUPEREL	EID	BID	IN	OUT	E1	E2	E3	E4	+abc
SUPEREL	201	01	LIBR	INP2	1	3	5	THRU	+ABC

+abc	E5	E6	E7	E8	E9	E10	etc.		
+ABC	11	4	18	THRU	92	100			

ALTERNATE FORM

SUPEREL	EID	BID	IN	OUT	REID	α	β	γ	
SUPEREL	201	02	SAME	SUPEREL	101	20.0	0.	0.	

Field

Contents

- EID** Superelement identification number (Integer > 0).
- BID** Identification number of a BSET1 boundary card (Integer ≥ 0).
- IN** Location of library elements or existing superelement. Options are:
- LIBRARY - Assemble the superelement from elements defined in the list on this card, e.g., E1, E2, etc. (DEFAULT).
- INP1-INP9 - Superelement previously calculated and resides on the specified user tape.
- SAMEAS - Used to identify a new superelement which is the same as the superelement defined in field REID.
- IMAGE - Used to identify a new superelement which is a reflected image of the superelement defined in field REID.
- OUT** Output location for superelement, options are:
- LIBRARY - Place assembled superelement into library for use by higher level superelements or pseudo-structure. When using this option, INPT user file is utilized for data storage, and must be assigned by user.
- INP1-INP9 - Store superelement on user file for future use. When this option is selected the superelement is automatically stored in the element library and INPi file.
- SUPEREL - Used only when IN=SAME, or IMAGE. Places the new superelement into the library for use in assembling higher level superelements or the pseudo-structure.

BULK DATA DECK

- E1,E2... - A list of identification numbers of all elements contained in the superelement (Integer > 0). The word THRU may be used anywhere in the list.
- REID - Superelement identification number (Integer > 0, and ≠EID).
A previously defined superelement identification number having the same physical characteristics of the superelement currently being defined (Integer > 0).
- α, β, γ - Euler angles of rotation, in degrees, and in order shown (real, default = 0.0).
- α - Used to rotate the superelement matrices about the basic (0) x coordinate
- β - Used to rotate the superelement matrices about the basic (0) y coordinate
- γ - Used to rotate the superelement matrices about the basic (0) z coordinate
- Remarks:
1. Superelement identification numbers must be unique with respect to all other element identification numbers.
 2. There is no restriction on the number of superelement levels, however, superelements are assembled beginning with the lowest EID first. Therefore, higher level superelements must have EID's greater than all lower level superelements.
 3. Elements in the list to be assembled may be unsorted. The word THRU may be used anywhere in the list.
 4. The use of a BID = 0 implies that there is no boundary grid set, and all grid points encountered will be automatically placed in the set for the assembled superelement. This is useful for highest level superelements which are an assemblage of lower level superelements.
 5. The data of one or more superelements can be stored on a user tape. Up to 9 user tapes are available (INP1, INP2,...INP9).
 6. The alternate form of the SUPEREL card is used to define a new superelement using the same physical characteristics of a previously defined superelement. The boundary grid point set (BID) must be unique, and grid points contained on the BSET1 card will be used for the new superelement. This option is useful in defining repetitive structures. Only one set of bulk data is required to define all similar superelements. Rotation of the superelement is obtained by defining the appropriate angles of rotation, α , β and γ . The order of rotation is α , β , γ using positive vector notation.

BULK DATA DECK

Input Data Card BSET1

Description: Defines the boundary grid points for a superelement.

Format and Example:

BSET1	ID	G1	G2	G3	G4	G5	G6	G7	+abc
BSET1	01	1	10	18	15	20	8	30	+ABC

+abc	G8	G9	etc.						
+ABC	35	100							

ALTERNATE FORM

BSET1	ID	GID1	"THRU"	GID2					
BSET	01	21	THRU	28					

FIELD

CONTENTS

ID Boundary grid point set identification number (Integer > 0).

Gi, GIDi Grid point identification numbers (Integer > 0).

- Remarks:**
1. Boundary grid points must agree in number, meaning and direction.
 2. Six degrees of freedom must exist at each boundary grid point.
 3. All grid points referenced within a THRU list must exist.
 4. This card must exist if a SUPEREL card is present, and refers to the identification number in field 2.
 5. This card is optional for the highest level superelement which may have BID=0.
 6. Interior grid points of a superelement may be resequenced using the SEQGP card.
 7. BSET1 standard and alternate forms can be mixed in describing a boundary grid points set.

BULK DATA DECK

Input Data Card SAME1

Description: Defines equivalent grid points among superelements (Substructure).

Format and Example:

1	2	3	4	5	6	7	8	9	10
SAME1	G1	E1	G2	E2	G3	E3	G4	E4	+abc
SAME1	51	151	62	662	51	251	101	1101	+ABC
+abc	G5	E5	G6	E6	G7	E7	etc.		
+ABC	241	4	772	277	88	251			

FIELD

CONTENT

Gi Grid point identification number (Integer > 0)

Ei Equivalent grid point identification number (Integer > 0)

Remark: 1. Gi and Ei must exist in pair

2. Multiple equivalent grid points are allowed

e.g. G1 = E1 = E3 = G7

APPENDIX B
 SPERRY'S NASTRAN RIGID FORMAT 15
 SUBSTRUCTURING ANALYSIS

```

BEGIN      NO.15 SUBSTRUCTURING ANALYSIS - SERIES M3 $
FILE      KNN=SAVE/SEK4MB=SAVE/LLL=TAPE $
SETVAL    //V,N,PHASE/C,N,O/V,N,UNPURGE/C,N,I/V,N,PURGE;NG/C,N,-1 $
SAVE      PHASE,UNPURGE,PURGING $
LABEL     PHASE1 $
SSPHIB    CASECC,GEOM1,GEOM2,GEOM3,GEOM4,GEOM6/SEK4MB/V,N,IDSEL/V,N,
          IDBSET/V,N,INSEL/V,N,OUTSEL/V,N,NOSEL/V,N,PHASE/V,N,LIST $
SAVE      IDSEL,IDBSET,INSEL,OUTSEL,NOSEL,PHASE,LIST $
COND      PHASE2,PHASE $
PURGE     GPDT,PLTSETX,PLOTXI,SLT,ECT,ECPT,GPCT,GPST,OGPST,OGPWG,
          KOO,LOO,UOO,MGG,RG,RUOV/ UNPURGE $
GP1       GEOM1,GEOM2,GEOM4/GPL,EQEXIN,GPDT,CSTM;BGPDT,SIL/V,N,LUSET/
          C,N,123/V,N,NOGPDT $
SAVE      LUSET $
GP1S      GEOM2,GEOM4,EQEXIN,SIL,SEK4MB// $
CHKPNT    GPL,EQEXIN,GPDT,CSTM,BGPDT,SIL $
GP2       GEOM2,EQEXIN/ECT $
CHKPNT    ECT $
PLTSET    PCDB,EQEXIN,ECT/PLTSETX,PLTPAR,GPSETS,ELSETS/V,N,NSIL/
          V,N,JUMPPLOT $
SAVE      NSIL,JUMPPLOT $
PRMSG     PLTSETX// $
SETVAL    //V,N,PLTFLG/C,N,I/V,N,PFILE/C,N,O $
SAVE      PLTFLG,PFILE $
COND      P1,JUMPPLOT $
PLOT      PLTPAR,GPSETS,ELSETS,CASECC,BGPDT,EQEXIN,SIL,,,/PLOTXI/
          V,N,NSIL/V,N,LUSET/V,N,JUMPPLOT/V,N,PLTFLG/V,N,PFILE $
SAVE      JUMPPLOT,PLTFLG,PFILE $
PRMSG     PLOTXI// $
LABEL     P1 $
CHKPNT    PLTPAR,GPSETS,ELSETS $
GP3       GEOM3,EQEXIN,GEOM2,BGPDT/SLT,GPTT/C,N,123/V,N,NOGRAV/C,N,123 $
SAVE      NOGRAV $
PARAM     //C,N,AND/V,N,SKPMGG/V,N,NOGRAV/V,Y,GRDPNT $
PURGE     MGG/SKPMGG $
CHKPNT    SLT,GPTT,MGG $
          ECT,EPT,BGPDT,SIL,GPTT,CSTM/EST,GEI,ECPT,GPCT/V,N,LUSET/
          C,N,123/V,N,NOSIMP/C,N,O/V,N,NOGENL/V,N,GENEL $
SAVE      NOSIMP,NOGENL,GENEL $
PARAM     //C,N,AND/V,N,NOELMT/V,N,NOGENL/V,N,NOSIMP $
COND      ERROR4,NOELMT $
PURGE     GPST/NOSIMP/OGPST/GENEL $
CHKPNT    EST,ECPT,GPCT,GEI,GPST,OGPST $
COND      LBL1,NOSIMP $
SMAI      CSTM,MPT,ECPT,GPCT,DIT,SIL,SEK4MB/KGGX,,GPST/V,N,NOGENL/
          V,N,NOK4GG/V,Y,OPTION $
CHKPNT    GPST,KGGX $
COND      LBL1,SKPMGG $
SMA2      CSTM,MPT,ECPT,GPCT,DIT,SIL,SEK4MB/MGG,/V,Y,WTHASS=1.0/
          V,N,NOMGG/V,N,NOBGG/V,Y,COUPHASS/V,Y,CPBAR/V,Y,CPROD/
          V,Y,CPQUAD1/V,Y,CPQUAD2/V,Y,CPTRIA1/V,Y,CPTRIA2/V,Y,CPTUBE/
          V,N,CPQDPLT/V,Y,CPTRPLT/V,Y,CPTRBSC $
SAVE      NOMGG $
CHKPNT    MGG $
  
```

COND LBL1,GRDPNT S
COND ERROR2,NOMGG S
GPWG BGPDT,CSTM,EGEXIN,MGG/OGPWG/V,Y,GRDPNT=-1/V,Y,WTMASS S
OFF OGPWG,,,,,//V,N,CARDNO S
SAVE CARDNO S
LABEL LBL1 S
EQUIV KGGX,KGG/NOGENL S
CHKPNT KGG S
COND LBL2,NOGENL S
SMA3 GE1,KGGX/KGG/V,N,LUSET/V,N,NOGENL/V,N,NOSIMP S
CHKPNT KGG S
LABEL LBL2 S
PARAM //C,N,HPY/V,N,NSKIP/C,N,O/C,N,O S
GP4 CASECC,GEOM4,EGEXIN,SIL,GPDT/RG,YS,USET,/V,N,LUSET/V,N,MPCF1/
V,N,MPCF2/V,N,SINGLE/V,N,OMIT/V,N,REACT/V,N,NSKIP/V,N,REPEAT/
V,N,NOSET/V,N,NOL/V,N,NOA S
SAVE MPCF1,MPCF2,SINGLE,OMIT,REACT,NSKIP,REPEAT,NOSET,NOL,NOA S
COND ERROR3,NOL S
PARAM //C,N,AND/V,N,NOSR/V,N,SINGLE/V,N,REACT S
PURGE KRR,KLR,QR,DM/REACT/GH/MPCF1/GO,KOO,LOO,UOO,PO,UOOV,RUOV/OMIT/
PS,KFS,KSS/SINGLE/QG/NOSR S
EQUIV KGG,KNN/MPCF1 S
CHKPNT KRR,KLR,QR,DM,GH,GO,KOO,LOO,UOO,PO,UOOV,QQ,PS,KFS,KSS,USET,RG,
YS,RUOV,KNN S
COND LBL3,GENEL S
GPSP QPL,GPST,USET,SIL/OGPST S
OFF OGPST,,,,,//V,N,CARDNO S
SAVE CARDNO S
LABEL LBL3 S
COND LBL4,MPCF2 S
MCE1 USET,RG/GH S
CHKPNT QM S
MCE2 USET,GH,KGG,,,/KNN,,, S
CHKPNT KNN S
LABEL LBL4 S
EQUIV KNN,KFF/SINGLE S
CHKPNT KFF S
COND LBL5,SINGLE S
SCE1 USET,KNN,,,/KFF,KFS,KSS,,, S
CHKPNT KFS,KSS,KFF S
LABEL LBL5 S
EQUIV KFF,KAA/OMIT S
CHKPNT KAA S
COND LBL6,OMIT S
SMP1 USET,KFF,,,/GO,KAA,KOO,LOO,UOO,,,,, S
CHKPNT GO,KAA,KOO,LOO,UOO S
LABEL LBL6 S
EQUIV KAA,KLL/REACT S
CHKPNT KLL S
PARAM //C,N,NOT/V,N,LASTONE/V,N,NOSEL S
COND LBL8,LASTONE S
COND LBL7,REACT S
RBMG1 USET,KAA,/KLL,KLR,KRR,,, S
CHKPNT KLL,KLR,KRR S
LABEL LBL7 S
RBMG2 KLL/LLL,ULL S
CHKPNT ULL,LLL S
COND LBL8,REACT S
RBMG3 LLL,ULL,KLR,KRR/DM S

```

CHKPNT   DM $
LABEL    LBL8 $
SSG1     SLT,BGPDY,CSTM,SIL,EST,MPT,GPTT,EDT,HGG,CASECC,DIT/PG/
          V,N,LUSET/V,N,NSKIP/V,Y,OPTION $
CHKPNT   PG $
EQUIV    PG,PL/NOSET $
CHKPNT   PL $
COND     LBL9,NOSET $
SSG2     USET,GM,YS,KFS,GO,DM,PG/QR,PO,PS,PL $
CHKPNT   QR,PO,PS,PL $
LABEL    LBL9 $
COND     LBL10,OMIT $
SSG3     L00,U00,K00,PO,,,,/U0OV,,RUOV,/C,N,-1/V,Y,IRES=-1/V,N,NSKIP/
          V,N,EPSI $
SAVE     EPSI $
CHKPNT   U0OV,RUOV $
LABEL    LBL10 $
COND     LBL11,IRES $
MATGPR   GPL,USET,SIL,RUOV//C,N,0 $
LABEL    LBL11 $
PURGE    GPDT,PLTSETX,PLOTX1,SLT,ECT,ECPT,GPCT,GPST,OGPST,OGPWG,
          K00,L00,U00,HGG,RG,RUOV/ LASTONE $
SSPHIE   CASECC,GEOM3,GEOM6,SEK4MB,KLL,,,,PO,
          GO,EQEXIN,CSTM,SIL,GPTT,BGPDY,EST,GPL,USET,PG,YS,GM,PS,KFS,
          KSS,QR,PLTPAR,GPSETS,ELSETS,U0OV,/SEQA/V,N,IDSEL/V,N,IDBSET/
          V,N,INSEL/V,N,OUTSEL/V,N,NOSEL/V,N,PHASE/V,N,LIST/V,N,NSKIP/
          V,N,LUSET/V,N,JUMPPLOT/V,N,PLTFLG/V,N,PFILE/V,N,REPEAT/
          V,N,CARDNO/V,N,DUM1/V,N,DUM2/V,N,DUM3 $
SAVE     IDSEL,IDBSET,INSEL,OUTSEL,NOSEL,PHASE,LIST $
REPT     PHASE1,128 $
JUMP     ERROR6 $
LABEL    PHASE2 $
FILE     SELEV=SAVE/QG=APPEND/PGG=APPEND/UGV=APPEND/GM=SAVE $
SSPH2B   CASECC,GEOM6/SELEV/V,N,IDSEL/V,N,IDBSET/V,N,INSEL/V,N,OUTSEL/
          V,N,NOSEL/V,N,PHASE/V,N,LIST $
SAVE     IDSEL,IDBSET,INSEL,OUTSEL,NOSEL,PHASE,LIST $
CHKPNT   SELEV $
COND     PHASE3,PHASE $
SSG3     LLL,ULL,KLL,PL,,,,/ULV,,RULV,/V,N,OMIT/V,Y,IRES/V,N,NSKIP/
          V,N,EPSI $
CHKPNT   ULV,RULV $
COND     LBL12,IRES $
MATGPR   GPL,USET,SIL,RULV//C,N,L $
LABEL    LBL12 $
SDR1     USET,PG,ULV,U0OV,YS,GO,GM,PS,KFS,KSS,QR/UGV,PGG,QG/V,N,NSKIP/
          C,N,STATICS $
CHKPNT   UGV,PGG $
COND     LBL15,REPEAT $
JUMP     ERROR1 $
PARAM    //C,N,NOT/V,N,TEST/V,N,REPEAT $
COND     ERROR5,TEST $
LABEL    LBL15 $
CHKPNT   QG $
SDR2     CASECC,CSTM,MPT,DIT,EQEXIN,SIL,GPTT,EDT,BGPDY,PGG,QG,UGV,EST,/
          OPG1,OQG1,OUGV1,OESI,DEF1,PUGV1/C,N,STATICS $
OFF      OUGV1,OPG1,OQG1,DEF1,OESI, //V,N,CARDNO/V,Y,OPTION $
SAVE     CARDNO $
COND     LBL16,JUMPPLOT $
PLOT     PLTPAR,GPSETS,ELSETS,CASECC,BGPDY,EQEXIN,SIL,ECPT,PUGV1,OESI,
          PLOTX2/V,N,NSIL/V,N,LUSET/V,N,JUMPPLOT/V,N,PLTFLG/V,N,PFILE $

```

```

SAVE          PFILE $
PRTHMSG      PLOTX2// $
LABEL        LBL16 $
SSPHZE       SELEV, EQEXIN, UGV//V,N, IDSEL/V,N, IDBSET/V,N, INSEL/V,N, OUTSEL/
              V,N, NOSEL/V,N, PHASE/V,N, LIST $
SAVE         IDSEL, IDBSET, INSEL, OUTSEL, NOSEL, PHASE, LIST $
PURGE        USET, PG, ULV, UOOV, YS, GO, GM, PS, KFS, KSS, QR, UGV, PGG, QG, CSTM, EQEXIN,
              SIL, GPTT, BGPDT, EST, OPG1, OQG1, OUGV1, OES1, OEF1, PUGV1, PLTPAR,
              GPSETS, ELSETS, PLOTX1, PLOTX2, GPL, PL, PS, RULV, SEQA/ PURGING $
LABEL        PHASE3 $
SSPH3B       CASECC, GEOM6, SELEV/ ULVX,
              GOX, EQEXIX, CSTMX, SILX, GPTTX, BGPDTX, ESTX, GPLX, USETX, PGX, YSX,
              GMX, PSX, KFSX, KSSX, QRX, PLTPAX, GPSETX, ELSETX, UOOVX//V,N, IDSEL/
              V,N, IDBSET/V,N, INSEL/V,N, OUTSEL/V,N, NOSEL/V,N, PHASE/V,N, LIST/
              V,N, NSKIP/V,N, LUSET/V,N, JUMPPLOT/V,N, PLTFLG/V,N, PFILE/
              V,N, REPEAT/V,N, CARDNO/V,N, DUM1/V,N, DUM2/V,N, DUM3 $
SAVE         IDSEL, IDBSET, INSEL, OUTSEL, NOSEL, PHASE, LIST,
              NSKIP, LUSET, JUMPPLOT, PLTFLG, PFILE, REPEAT, CARDNO $
COND         LBL19, PHASE $
SDR1         USETX, PGX, ULVX, UOOVX, YSX, GOX, GMX, PSX, KFSX, KSSX, QRX/
              UGVX, PGGX, QGX/V,N, NSKIP/C,N, STATICS $
CHKPNT       UGVX, PGGX $
COND         LBL17, REPEAT $
JUMP         ERROR1 $
COND         ERRORS, TEST $
LABEL        LBL17 $
CHKPNT       QGX $
COND         LBL18, IDBSET $
SDR2         CASECC, CSTMX, MPT, DIT, EQEXIX, SILX, GPTTX, EDT, BGPDTX, PGGX, QGX,
              UGVX, ESTX, /OPG2, OQG2, OUGV2, OES2, OEF2, PUGV2/C,N, STATICS $
OFF          OUGV2, OPG2, OQG2, OEF2, OES2, //V,N, CARDNO/V,N, OPTION $
SAVE         CARDNO $
COND         LBL18, JUMPPLOT $
PLOT         PLTPAX, GPSETX, ELSETX, CASECC, BGPDTX, EQEXIX, SILX, ECPT, PUGV2, ,
              OES2/PLOTX3/V,N, NSIL/V,N, LUSET/V,N, JUMPPLOT/V,N, PLTFLG/
              V,N, PFILE $
SAVE         PFILE $
PRTHMSG      PLOTX3// $
LABEL        LBL18 $
SSPH3E       SELEV, EQEXIX, UGVX//V,N, IDSEL/V,N, IDBSET/V,N, INSEL/V,N, OUTSEL/
              V,N, NOSEL/V,N, PHASE/V,N, LIST $
SAVE         IDSEL, IDBSET, INSEL, OUTSEL, NOSEL, PHASE, LIST $
REPT         PHASE3, 128 $
LABEL        LBL19 $
JUMP         FINIS $
LABEL        ERROR1 $
PRTPARM      //C,N, -1/C,N, STATICS $
LABEL        ERROR2 $
PRTPARM      //C,N, -2/C,N, STATICS $
LABEL        ERROR3 $
PRTPARM      //C,N, -3/C,N, STATICS $
LABEL        ERROR4 $
PRTPARM      //C,N, -4/C,N, STATICS $
LABEL        ERRORS $
PRTPARM      //C,N, -5/C,N, STATICS $
LABEL        ERROR6 $
PRTPARM      //C,N, -6/C,N, STATICS $
LABEL        FINIS $
END          $

```

APPENDIX C SUBSTRUCTURING EXAMPLE

```

QASG,T   INPS,T,SAVE03 . SUBSTR PHASE I S.E. 101/102/103/201
@XQT,L   @NASTRAN,LINK1
NASTRAN  SYSTEM(65)=15; LOGFILE=80, HICORE=54000
ID        DEMO15 SUBSTRUCTURING
APP       DMAP
@ADD,P   @DMAP.DMAP15
TIME     4
SUBSTRUCTURING PHASES I, II, AND III
DIAG     8,14
CEND

TITLE    = FOUR PIECE SUBSTRUCTURING MODEL
ECHO     = BOTH
DISP     = ALL
FORCE    = ALL
STRESS   = ALL
LOAD     = -1
SPC      = 2
BEGIN BULK
$ *** SUPER ELEMENT 101 ***
SUPEREL 101    01    LIBR    INPS    1    THRU    4    11    +S1
+S1      THRU   14    40     31     30     36     61    THRU   +S11
+S11     64
GRID    1      -10.   20.   0.0
GRID    2      -10.   10.   0.0
GRID    3      -10.   0.    0.0
GRID    12     -5.    20.   0.0
GRID    19     -5.    10.   0.0
GRID    4      -5.    0.    0.0
GRID    11     0.    20.   0.0
GRID    18     0.    10.   0.0
GRID    5      0.    0.    0.0
BSET1   01     3     4     5     18     11
CROD    1     10    1     2
CROD    2     10    2     3
CROD    3     10    3     4
CROD    4     10    4     5
CROD    13    10    5     18
CROD    14    10    39    11
CROD    11    10    11    12
CROD    12    10    12    1
CQUAD2  30    20    1     2     19     12
CQUAD2  31    20    2     3     4     19
CQUAD2  36    20    12    19    18     11
CQUAD2  40    20    19    4     5     18
CBAR    61    60    3     4     2
CBAR    62    60    4     5     19     2
CBAR    63    60    5     18    4     2
CBAR    64    60    39    11    19     2
$ *** SUPER ELEMENT 102 ***
SUPEREL 102    02    LIBRARY INPS    5    THRU    10    37    +S2
+S2      32     39    33    71    THRU   74
GRID    10     5.    20.   0.0
GRID    20     5.    10.   0.0
GRID    6      5.    0.    0.0
GRID    9     10.   20.   0.0
GRID    8     10.   10.   0.0
GRID    7     10.   0.    0.0
GRID    47    10.   0.    10.
BSET1   02     5     6     7     39     11

```

CROD	5	10	5	6						
CROD	6	10	6	7						
CROD	7	10	7	8						
CROD	8	10	8	9						
CROD	9	10	9	10						
CROD	10	10	10	11						
CQUAD2	39	20	11	39	20	10				
CQUAD2	37	20	39	5	6	20				
CQUAD2	33	20	10	20	8	9				
CQUAD2	32	20	20	6	7	8				
CBAR	71	60	5	39	6					2
CBAR	72	60	39	11	20					2
CBAR	73	60	5	6	39					2
CBAR	74	60	6	7	47					2
\$ *** SUPER ELEMENT 103 ***										
SUPEREL	103	03	LIBR	INPS	15	THRU	20	38	+S3	
+S3	35	34	41	THRU	45	81	THRU	84		
GRID	14		-10.	0.	-20.					
GRID	15		0.0	0.	-20.					
GRID	16		10.	0.	-20.					
GRID	13		-10.	0.	-10.					
GRID	21		0.	0.	-10.					
GRID	17		10.	0.	-10.					
BSET1	03	3	THRU	7						
BSET1	03	21								
CROD	15	10	3	13						
CROD	16	10	13	14						
CROD	17	10	14	15						
CROD	18	10	15	16						
CROD	19	10	16	17						
CROD	20	10	17	37						
CQUAD2	34	20	14	15	21	13				
CQUAD2	35	20	15	16	17	21				
CTRIA2	38	30	13	4	21	0.0				
CTRIA2	42	30	13	3	4	0.0				
CTRIA2	43	30	4	5	21	0.0				
CTRIA2	41	30	21	6	17	0.0				
CTRIA2	44	30	21	5	6	0.0				
CTRIA2	45	30	6	37	17	0.0				
CBAR	81	60	3	4	13					2
CBAR	82	60	4	5	21					2
CBAR	83	60	5	6	21					2
CBAR	84	60	6	37	17					2
\$ *** SUPER ELEMENT 201 ***										
SUPEREL	201	00	LIBRARY	INPS	101	102	103	50	+S4	
+S4	51									
GRID	31		20.	0.	0.					
CBAR	50	40	37	31	47					2
CBAR	51	40	37	5	39					2
\$ *** OTHER DATA ***										
SAME1	37	7	5	38	18	39				
PROD	10	1	0.65							
PQUAD2	20	1	0.02							
PTRIA2	30	1	0.42							
PBAR	40	1	1.27	3.51	3.52	1.				+P1
+P1	0.61	0.52								
PBAR	60	2	1.	2.	2.	1.				
MAT1	1	30.+6	12.+6	0.3	0.00073	8.-6	70.			
MAT1	2	30.+4	12.+4	0.3	7.3-5	8.-7	7.			
GRAY	33		386.4	0.	0.0	+1.0				
LOAD	1	1.0	1.0	33						
SPC1	2	6	8	THRU	12					
SPC1	2	6	18	19	20	1	2			
SPC1	2	5	13	THRU	17					
SPC1	2	5	21							
SPC1	2	123456	31	47						
SPC	2	1	32	+1.6						
SEGGP	12	2.5	17	14.5						
ENDDATA										
GXQT *NASTRAN.LINKO										

UNEXPECTED DIVIDENDS FROM LEVEL 16
AUTOMATED SUBSTRUCTURING*

Eric I. Field
Universal Analytics, Inc.

SUMMARY

The new automated multi-stage substructuring capability (AMSS) now available in NASTRAN Level 16 provides a number of unexpected tools for use in both project planning and the structural design cycle. The pay-offs in terms of improved design, more reliable results, and confidence are extra benefits added to the savings in cost and schedule that can be realized. To realize these advantages, some old stereotypical assumptions of what substructuring entails must be challenged.

INTRODUCTION

Substructuring is no longer just a last resort means of getting into the analysis of large complex models. Substructuring can be used very effectively with small-to-medium size structures as well. With less automated systems, the effort to substructure was warranted only for larger problems. Now with AMSS, the simple linguistic commands (like Case Control), the automated file maintenance features, and the independent numbering and naming schemes for each basic substructure, the engineer/user can free himself of tedious book-keeping chores in order to concentrate on modeling, design, and analysis.

Substructuring is an analysis technique which allows a structural model to be subdivided into smaller, more manageable segments. Carried to its logical conclusion, this process can be staged by repeated subdivision of each segment until, finally, only the elementary beam and plate finite elements are left which define the basic substructures of the system.

Once these basic, or elementary, substructure models are defined, the user applies the two basic functions provided to build up the full model for solution. A "combine" operation joins together the segments of the model, and a "reduce" operation effectively omits degrees of freedom to reduce the problem to a more manageable size. By using these "combines" and "reductions" in multiple stages, as with the AMSS system, very effective utilization of computer resources may be achieved. The AMSS system streamlines the data preparation for each of these steps while retaining the full NASTRAN modeling features upon which users depend.

*The AMSS system was developed by Universal Analytics, Inc. for NASA Langley Research Center under contract to, and monitored by McDonnell Douglas Astronautics Company.

A third function also is provided to "recover" the solution data at any level, or stage, of the substructuring sequence. Ultimately, recovery to the basic substructure level is required to obtain the detail element stress output required in the design iteration process. However, AMSS provides for selective output of displacements, loads, and constraint forces at any of the intermediate stages.

The key to the success of AMSS as an analysis tool lies in its data management system, the Substructure Operating File, called the SØF. The SØF provides for automated storage and retrieval of all data required during the analysis. Editing, back-up, and restore capabilities are ample to maintain the file. Because the file can get quite large, multiple physical devices may be used for storage and access during processing. The SØF features are so adaptive, complex models can be built up one step at a time to allow for careful model data checkout at each step.

Knowing that these capabilities exist raises a pertinent question; when should a substructuring analysis with AMSS be considered? It should be considered:

- When the complete structure has repetitive parts.
- When different contractors are responsible for different components of the structure.
- When different groups in an organization are responsible for different components.
- When one or more components may undergo design iteration.
- When the structure may be conveniently divided into easily managed components for detailed analysis.
- When the interaction effects between structural components are of primary interest.
- When the whole structural model would be too large for efficient computer processing.

It should be noted that extreme complexity or size of a model is only one of many reasons for using substructuring. Using the new Level 16 AMSS features can provide considerable advantages to both the engineer/analyst and his management. The unexpected dividends addressed below deal first with the planning process and then with the question of design iteration. Suggestions are made to help keep the cost of processing down and the quality of results high.

PLANNING PAY-OFFS

Many advantages are to be gained from the planning process required to undertake a substructuring effort. The very flexibility of AMSS requires the analyst to think ahead and plan. The completeness of AMSS provides a manager with the freedom and the framework to organize and synthesize the design activities of numerous groups. This preplanning effort and the motivation to carry it out offers pay-offs in each of the following areas of critical management concerns:

- Group objectives - clearly defined
- Substructure interface requirements - simple and well-defined
- Intergroup communications - greatly simplified
- Performance evaluation - easy tracking of both cost and schedule
- Motivation - each group responsible for its own effort and results
- Cost effectiveness - overall labor and computer cost savings can be significant
- Outcome - more reliable results

Of course, many of these advantages are the usual result of all good pre-planning for any effort. However, with substructuring, the unexpected dividend is the simplicity, objectivity, and completeness with which that task can be performed.

For example, the first step in preparing a structure for analysis with substructuring techniques is to define the objectives of the analysis for each segment of the model to be developed. One segment may be highly critical, requiring fine detail in the modeling. Another segment may be subject to design changes and must be kept simple for the time being. Other segments may be important only in their effect on the overall response, but not critical in and of themselves.

Once these objectives are identified, the process of subdividing the structure for modeling can get under way. Often, different design groups are logically associated only with certain segments of the structure and not others. Hence, only the key personnel of each such group need participate in this preplanning activity. The outcome to be achieved is simple. The boundaries between structures must be defined. The sequence for model development and the schedule for its introduction into the system is needed. The plan for stepping through the several analyses involved should be prepared to include individual substructure model verification, design iteration sequences, and final model integration with solution data recovery and final validation.

A significant and crucial element of this preplanning is the cost analysis which will effect, and will be effected by the sequence of steps chosen for building the full model and completing the analyses. The following suggestions should be considered:

- Select stages for the "reduce" operation.
 - a. Use Phase 1 to "omit" undesired degrees of freedom and to constrain (SPC) singularities.
 - b. Several small "reduce" operations usually cost less than one large operation.

- c. For a static analysis, or if only a few mode shapes are desired, do not reduce unless problem size or analysis sequence warrants it.
 - d. Either reduce out very many or very few degrees of freedom. The worst case is to reduce out only half.
- Estimate matrix size and density to anticipate:
 - a. SØF file size requirements
 - b. Run time
 - c. Core requirements
 - Plan use of utility operations to:
 - a. Back up SØF after key steps
 - b. Use standard NASTRAN checkpoint/restart for Phase 3 restart or for the solution step if large matrices are involved.
 - c. Use SØF output diagnostic data to check and verify the results after each "combine" and "reduce" operation.
 - If a complex or modified substructuring analysis is planned, prepare a "simulation" run with simplified models to check:
 - a. NASTRAN deck structure
 - b. SØF manipulations
 - c. Any special NASTRAN features being utilized

Beyond all these considerations, the most important dividend for using AMSS is the freedom to develop each basic substructure independently. The only restrictions imposed are those required to ensure compatibility at each of the substructure boundaries. That is, agreement must be reached as to grid point locations at the boundary and as to the distribution of structural material between substructures. The independent numbering of grids, elements, and loadings for each basic substructure allows each model to be developed and changed independently.

The separate models can be prepared in parallel with only minimal interaction required among the groups responsible. With the flexibility of the SØF features, each model can be loaded and analysed separately. No checkpoint/restart tapes are needed. This checkout processing ensures the quality of the model and completeness of the loads. Similar checkout is possible at each level of combination until the full model is reached. If errors are encountered, their correction involves a minimum of computer costs to rework. Usually only one or a few of the preceding operations need to be repeated and that is a lot cheaper than resolving the model as a whole. Thus, better and more reliable results can be expected from the final model.

The economic and engineering advantages of exploiting this general, multi-stage capability are evident in the analysis cycle. But, to the designer, other applications of AMSS in the design cycle may not be so obvious.

DESIGN ITERATIONS

With suitable preplanning, preliminary design data may be used to model most segments of the full structural system to provide good boundary conditions for detail design analyses of individual components. Thus, entire design cycles can often be skipped, saving both man-time costs and schedule. This is where application of AMSS to smaller structural systems can pay off.

For example, with electronic packaging, with access ports in aircraft or ships, and with penetrations in nuclear reactor design, numerous local design problems can be isolated, each within a separate basic substructure. First, a solution of the whole structure is obtained assuming a reasonable selection of design alternatives for each problem. Thus, with this overall solution available through AMSS, the local effects of other design alternatives can be examined. Upon recovery of the solution results to the Phase 3 level for the approximate model analyzed, several Phase 3 runs can be made, one for each different design alternative.

In effect, the recovered solution vector is being used as an imposed displacement pattern at the boundaries of the basic substructure. These displacements, note, contain the interaction effects of the entire structure and the responses to all the loadings imposed on other substructures. So long as the design changes do not affect the model at the at the substructure boundaries, compatibility with the rest of the structure is maintained. Of course, if the changes are major, they may affect the overall solution. In this case, the whole model may require resolving with those changes included to update the boundary displacement vector.

Only one requirement must be met to exploit this design iteration capability. That is:

- The basic substructure should be "reduced" in Phase 1 to omit all points to be affected by the design changes. All the grid points at the boundaries with all other substructures would make a good set to be kept.

Each of the Phase 3 runs can then be executed simply by submitting the new bulk data deck with the desired design changes incorporated.

For efficiency, the basic substructure subject to design changes should be among the last to be combined into the final whole model. This would minimize the cost of solution vector data recovery and the cost of rebuilding the whole model once the final design is selected.

An alternative exists if one substructure alone would be undergoing design changes. In this case the remainder of the full model would be "reduced" to those grid point degrees of freedom only at the boundary with that basic substructure. With this approach, each design alternative, whether involving major or minor changes, can be "combined" with the rest of the structure and solved "exactly." No approximation is involved here, as above, because the final model being solved is complete.

CONCLUDING REMARKS

Guidelines have been drawn for preparing a plan to carry out a substructuring effort using the new Level 16 NASTRAN automated substructuring capability. The rewards of preparing such a plan offer pay-offs by simplifying the management tasks, reducing costs, and enhancing the reliability and quality of results obtained.

Two novel approaches to solving the design iteration problems were presented. These reduce the cost of analysis and offer much greater flexibility to the designer than he has had available to him in the past.

SUBSTRUCTURE COUPLING - A DIFFERENT APPROACH

R. S. Lahey
Lockheed-California Company

SUMMARY

The substructure analysis of the Lockheed L-1011 Dash 500 long range derivative without use of the NASTRAN Level 16 multistage substructuring capability is described. It is presented as an example of how a large structural analysis can be organized into manageable independent tasks and how preprocessors, model definition conventions, and data management programs can be used to simplify the data management and model documentation. A novel procedure is described to analyze a single substructure by modifying the static solution to solve the interface compatibility equations simultaneously with the substructure solution.

INTRODUCTION

The analysis approach for the L-1011 Dash 500 commercial transport presented many problems not commonly encountered in most finite element analyses. These included: maintaining a close compatibility with a nine year old force method finite element analysis which is still in use; interfacing data with other in-house programs; maintaining over 400 card-image datasets which contain a total of nearly 250,000 cards; and management of several hundred thousand matrices and tables stored on over 1500 magnetic tapes. In order to overcome these problems, the model analysis procedures were standardized; data management programs were used to maintain the data; detail documentation of data interfaces, model definitions, and datasets were compiled; and new programs and NASTRAN modules were developed as necessary.

The approach developed for the Dash 500 analysis has performed successfully through two design iterations on 14 substructures involving a total of 27,000 degrees of freedom and 41,000 elements for a complete, nonsymmetric airplane analysis.

The substructure analysis of the L-1011 Dash 500 was started before the release of the new multistage substructuring capability available in NASTRAN Level 16.1. The coupling procedure used is based on the Lockheed-developed Gridpoint Labeled Matrix concept.

DEVELOPMENT OF THE MODEL

The baseline L-1011 Dash One was analyzed with the Lockheed force method finite element program. These models were developed in 1968, and are still in use for the Dash One configuration. When the model analysis of the Dash 500 derivative (shorter fuselage) of the L-1011 was planned it was decided that all new substructures, as well as the coupling, would be analyzed in NASTRAN. The purpose of this analysis was to calculate internal loads and stresses, structural deformations, and the structural influence coefficients (SICs - the structural flexibility matrix).

The model configuration of the L-1011 Dash 500 consists of twelve basic substructure models, seven of which are NASTRAN models and the remainder force method models taken directly from the basic L-1011 Dash One analysis (see figure 1). These models involve a total of 27,000 degrees of freedom, 41,000 elements, and 1000 coupling degrees of freedom.

The finite element model analysis was planned and developed in four independent tasks:

- Model development and substructure analysis
- External load transformation
- Coupling
- Data recovery

This separation of tasks was made possible by an early definition of the following:

- A unit load network over the complete model which serves to transform external design loads to the model gridpoints
- Substructure boundaries
- A consistent numbering system and other model conventions
- Determination of the force method models to be retained from the Dash One analysis

The definition of the unit load network, which was used to distribute external loads to the model gridpoints, as well as to calculate structural influence coefficients (SICs), was one of the first tasks of the analysis. Its early definition allowed the determination of the external design loads, in terms of these unit load conditions, to be performed independently of the finite element model work. The substructure finite element models were analyzed for the unit loads up to the data recovery step. At this point the solution vectors generated in the coupling step were multiplied by the design external loads, thus forming the solution vectors for the design conditions. This

method of analysis permits each substructure to be analyzed or reanalyzed for different critical design loads in the stress recovery step without violating interface continuity. The unit load distributions were also used to calculate structural influence coefficients (SICs) at the unit load points (using the virtual load method). These were used in the calculation of flexible airplane external loads and other aeroelastic effects.

Initially, SICs were obtained using a drastically simplified finite element model. These preliminary SICs were then superseded as the solution for the full Dash Three model became available.

The definition of the substructure boundaries along with the unit load network allowed the substructure models to be developed and checked out independently. The boundaries were chosen to divide the structure along structural subassembly mating planes, i.e., the fuselage was divided into the following subassemblies: flight station, centerbody (fuselage-wing carry-through structure), fuselage barrel sections, and aft cone. This also helped reduce the number of copies of data recovery results, which average several boxes per substructure, to that number required by the group responsible for the particular subassembly.

The retention of force method models from the L-1011 Dash One analyses was determined early so that transformation of the interface data from the force method model into image NASTRAN models could proceed. The justification for retaining Dash One force method models was based on the following factors:

- Model structures remained unchanged
- Recoding of a model into NASTRAN is costly
- Interface and unit load network definitions remained unchanged
- All substructure data necessary for coupling and data recovery were available for the unit load conditions
- Direct comparison of results between Dash One and Dash 500 configuration would be simplified if model output formats were unchanged
- Methods and procedures to transform force method model data to image NASTRAN substructures were established and proven in prior analyses.

ANALYSIS METHOD AND SPECIAL DMAP MODULES

The substructure analysis of the L-1011 Dash 500 transport was performed using Lockheed-California Company's modified version of NASTRAN Level 15.1 (NASTRAN-LCC) in the environment of the following related software:

- FINDEX - Matrix and table storage and retrieval system and data base
- DCAS - Direct Computer Access System - used to edit data and submit batch jobs
- Other Programs:
 - "Old" force method finite element analysis
 - Matrix algebra (outside NASTRAN), flutter, and design loads programs
 - Margin of safety postprocessors

The Lockheed-modified version of NASTRAN used at the time of the analyses did not contain the Level 16 multistage substructuring capability. The substructuring procedure used relies heavily on the following:

- The FINDEX data management system
- Lockheed-developed NASTRAN DMAP modules:
 - FDXIN - Inputs matrices from FINDEX
 - FDXOUT - Outputs matrices into FINDEX
 - ØMATG - Converts displacement set matrices into Gridpoint Labeled Matrix tables
 - IMATG - Converts Gridpoint Labeled Matrix tables into displacement set matrices
- Lockheed-developed Gridpoint Labeled Matrix concept

The key to the integration of NASTRAN with the other in-house programs is the use of the common data base, FINDEX. The Lockheed-developed data management system, FINDEX, is used by NASTRAN and the other in-house programs to keep track of generated data. This system has a catalog of all datasets stored. This catalog is used by the system to locate, retrieve and perform data management tasks such as writing reports on the status of the datasets for individual projects.

The modules FDXIN and FDXOUT are used to read and write NASTRAN matrix and table data from the FINDEX system. The modules are capable of reformatting data generated outside NASTRAN into NASTRAN matrix format and vice versa. The modules ØMATG and IMATG were introduced into NASTRAN to convert and reconvert displacement set matrices into and from the Lockheed-developed Gridpoint Labeled Matrix table. The ØMATG module uses the USET and EQEXIN tables to identify the matrix elements with the external gridpoint and degree of freedom. This information for both row and column identification and the matrix data are stored in the Gridpoint Labeled Matrix table. The IMATG module uses the USET and EQEXIN

tables for the current structure to convert the Gridpoint Labeled Matrix table into a NASTRAN displacement set matrix.

The Gridpoint Labeled Matrix concept avoids the correlation complexities of the assemblage of the boundary stiffness and load matrices from all substructures during the boundary coupling step, and reduces it to simple matrix additions. The conversion of Gridpoint Labeled Matrix tables to displacement set matrices using module IMATG takes care of differences in gridpoint sequencing and matrix size problems by reordering the data into the order of the current structure as defined by its USET and EQEXIN tables. In the assemblage of the stiffness matrix in the coupling step, all boundary stiffness data of the substructures are converted into the G-set size matrices and the converted matrices are then added together to form the assembled stiffness matrix. In this coupling method the interface gridpoints are required to have the same gridpoint identifier and to be defined in the same coordinate system.

COMPLETE MODEL COUPLING - THREE-STEP COUPLING METHOD

The substructure coupling of the L-1011 Dash 500 involved twelve basic substructure models and two reflected image models. In addition to the actual coupling, the force method models required special processing to transform them into image NASTRAN substructures. A total of 1000 coupling degrees of freedom, and 400 symmetric and 400 antisymmetric unit external load conditions were solved. In the data recovery step, over 1000 unique external design load conditions were analyzed on one or more of the models. The actual number of conditions for which data recovery was performed on any single model ranged from 150 to 350 design external load conditions.

The substructure coupling was performed using a three-step approach:

- Step 1 - Reduction of the substructure to the interface degrees of freedom
- Step 2 - Assemblage of the boundary data and the boundary coupling solution
- Step 3 - Data recovery of the substructures for external design load conditions

Step 1 was performed using an altered Static Solution One. The A-set was used to define the interface degree of freedoms to be coupled. In this step, all data necessary for later data recovery and the boundary stiffness and load matrices (KAA and PA) were saved in the FINDEX system. A rigid body check was also performed. This check serves to ascertain that the model was not self-stressing while undergoing a rigid body displacement (for details, see the "Structural Model Integrity" paper given by D.V. Wallerstein at this conference).

Step 2 also was performed using an altered Static Solution One. In this step, a pseudo structure was formed consisting of boundary gridpoints, SPC, MPC, and

resequencing data. The assembled stiffness and load matrices were generated by reading all substructure boundary data (KAA and PA) in the form of Gridpoint Labeled Matrices from FINDEX. These data were then converted into G-SET matrices using the Gridpoint Labeled Matrix concept and then added together, the results being equivalent to the KGG and PG matrices. The run then appears to NASTRAN as a standard Static Solution One problem. The resultant displacement solution vector, UGV, of the coupling pseudo structure is saved in FINDEX for use in the data recovery phase.

Step Three, the data recovery phase, was performed using approach DMAP. The data recovery step required a separate computer run for each substructure. In this run, the data necessary for data recovery were read from FINDEX; the solution vectors UGV, QG, and PGG were computed; and the data recovery tables corresponding to the element forces and stresses, the displacements, the SPC forces, and the applied loads were generated for all gridpoints, elements, and design external load conditions. These data were saved in the FINDEX system for later postprocessing. For immediate review and checking, the element force and stress tables were scanned (by a NASTRAN-LCC module) to determine the maximum positive and negative conditions for each element and the results of the six largest pairs were printed. In addition, the standard OFP print was produced for three representative external load conditions. Due to the size of the substructures and the large number of external load conditions (150 to 350) analyzed, these runs were very expensive, averaging about one hour of cpu time on the IBM 360-91, and after postprocessing they produced 5000 pages of recovery data and an additional equivalent of 5000 pages on microfiche.

ONE-STEP COUPLING FOR A SINGLE SUBSTRUCTURE

In the course of a major analysis such as the L-1011, it often occurs that several modifications of a single substructure have to be investigated (e.g., failure analyses, behavior with or without structural doors, etc.). A special procedure was developed which is particularly convenient for this situation. It is basically identical to the Static Solution One, with the exception that boundary stiffness and load terms reflecting the surrounding structure are added to the regular stiffness matrix KGG and load matrix PG formed for the single substructure.

This procedure has two phases: (1) a coupling stage which combines all the substructures outside the one being investigated to obtain reduced boundary stiffness and load matrices at the interface with the substructure of interest, and (2) an altered Static Solution One analysis, which is repeated for each modified analysis of that substructure. The alter inputs the boundary matrices calculated in the first phase, using the Gridpoint Labeled Matrix concept, and adds these data to the substructure stiffness (KGG) and load (PG) matrices. In this second phase, the equivalent of the boundary coupling and the data recovery steps required in the three-step coupling method are simultaneously solved during the static solution of the substructure.

The main advantages of this procedure are:

- The reduction of the boundary stiffness and load matrices to the interface of the substructure of interest is done only once.
- Only one run is required to analyze the substructure for each design change.
- Decomposition, and forward/backward substitution of the constrained stiffness matrix is generally not adversely affected.
- Specifying the interface degrees of freedom as the ASET will solve (if necessary) most bandwidth problems caused by the addition of the interface data.

MODEL DEFINITION CONVENTIONS

The model numbering system is an integral part of the data management and documentation of the analysis. The numbering system chosen for the NASTRAN models uses as its basic element a coded gridpoint number. The element numbers are composed of an element-type identification code followed by the gridpoint number of either the first or second point specified on the element connection card. The general rule for numbering gridpoints and elements for the fuselage models is established as follows:

Coded gridpoint number

$$\text{GID} = (\text{FS}) * 100 + \text{LN} \quad (5\text{-digit number})$$

Coded element number

$$\text{EID} = (\text{I}) * 100000 + \text{GID} \quad (6\text{-digit number})$$

where:

GID = Gridpoint identification number

EID = Element identification number

FS = Fuselage station to nearest inch (3-digit number)

LN = Local gridpoint number (2-digit number)

I = Element type identification code

Rules were established to resolve conflicts in numbering when the general conventions failed to be unique.

The coded gridpoint number consists of the fuselage station of the gridpoint (X-coordinate to nearest inch), and a local gridpoint number. Whenever possible the local gridpoint number identifies a common structural model feature, such as a specific longeron. An odd number identifies a point on the left side of the centerline, and the next even number its mirror image on the right side.

For an example of the gridpoint numbering system, see figure 2, which is the model drawing for a typical fuselage barrel frame. Local gridpoint number 17 represents the intersection of the frame and the upper floor on the left side. Using the general rule, the gridpoint numbers at frame stations 709, 729, and 846 are 70917, 72917, and 84617, respectively; and on the right side, the mirror image points have gridpoint numbers 70918, 72918, and 84618, respectively.

A good example of the elements numbering system is that of the frame bar elements whose type code was chosen to be 6. The frame bars are defined from top to bottom. The first gridpoint on the element connection is used to determine the element number. An example of this (see figure 2) is element number 670915, which is a frame bar (I=6) located on frame station 709 and is connected to local gridpoint number 15.

There are several distinct advantages in using these conventions:

- The approximate location of the gridpoint or element is established by its identification number.
- Model documentation is simplified - only exceptions need detailed documentation.
- The recovery output data are easily organized.
- The communication of model information among the many users is simplified.

Additional conventions were established to standardize the following:

- Model drawings
- Load card identification numbers
- Constraint set identification numbers
- Dataset naming

PLOT DECK GENERATION

A plot deck generation preprocessor program was written which takes advantage of the uniform systems of fuselage model definition. Using this program, undeformed plots were generated directly from only the bulk data deck. The

program generates all necessary executive and case control information and submits the final deck for a NASTRAN run. For a typical substructure, over 60 plot frames were generated to check the modeling. Without an automated procedure such a detailed plotting would not have been feasible.

The program produced a set of plots at each fuselage station, for all elements of a given element type and one for all elements in that plane. Side and isometric views were similarly plotted. The format of plot views conformed closely to model drawing conventions; because of the uniform system, identification of the plotted elements could easily be established for the model drawings. See figure 2 for the model drawing of Frame Station 709, and figure 3 for the NASTRAN plot of frame bars for the same frame station.

The use of plots greatly supports the model checkout and adds to model documentation and model confidence. See figure 4, which is a left side view of a fuselage barrel substructure. Note the missing horizontal element in the right-hand bay. One major difficulty in using NASTRAN plots is the lack of identification of the individual plot frames because of the inability to modify the plot title from frame to frame. Identification of the plots becomes a major data management task when plotting large numbers of frames.

PREPROCESSOR AND DATA DECK DIVISION

The use of preprocessors has made it convenient to divide the NASTRAN deck into four datasets which are concatenated at the time of job execution. The following is a typical separation:

- Executive deck
- Casecontrol deck
- Bulk data particular to the current run
- Bulk data defining the basic model

The separation of the deck has eliminated the need for duplicate datasets thus reducing errors when revising the data. The separation has also made it possible to check out individual data decks independently, which is especially useful when dealing with the large decks involved in a substructure analysis (many contain over 6000 cards).

The preparation, generation, and submittal of job decks (JCL) and the preparation of card image datasets were performed interactively on remote computer terminals. The data management of the card image datasets required extensive use of dataset naming conventions and the use of card image dataset management programs to maintain the integrity of these data. The use of preprocessors and the FINDEX data management system required most batch runs to have the following three job steps:

- Step 1 - The preprocess step which generates the complete NASTRAN deck using user-generated card image datasets.
- Step 2 - The FINDEX locate step which locates and collects all inputs from the FINDEX data management system.
- Step 3 - The NASTRAN run and the FINDEX update to incorporate output NASTRAN datasets into the FINDEX system.

For the analysis of the L-1011 Dash 500, three card-image dataset libraries were built. One library contains Executive and Casecontrol decks which were used in the following tasks:

- Plot generation
- Reduction of the substructure to the interface degrees of freedom
- Boundary coupling solution
- Substructure data recovery
- Analysis of a single substructure

A second library contains for each substructure the following bulk datasets:

- Basic model definition
- A-set of interface points
- Unit external load definition
- FINDEX control cards

The third library contains preprocessor programs to generate the decks used in the following tasks:

- Concatenation of datasets
- Plot deck generation
- Bulk data deck updating (modification)
- Generation of the pseudo coupling structure for the boundary coupling solution

The use of standard preprocessor programs and data decks has greatly increased our efficiency, reduced the number of data errors, and simplified the documentation of the analysis when compared to past projects where most runs consisted of physical card decks, and handwritten instructions to the computer operator identifying which tapes were to be used during the run.

CONCLUDING REMARKS

The practical organization and performance of a major structural finite element analysis of a large complex structure involves scores of problems of data acquisition, checking, interdisciplinary communications, etc. Most readers faced with similar tasks will be only too familiar with them. It was not the intention of this paper to describe the complete substructure analysis process, but rather, to discuss selected steps within this process which were major contributors to the successful on-schedule completion of the NASTRAN finite element analysis of the L-1011 Dash 500 commercial transport.

As previously noted, the NASTRAN Level 16.1 with its new multistage substructuring capability was not available at the time the L-1011 Dash 500 analysis was initiated. Since then, the NASTRAN Level 16.1 has been integrated with the Lockheed-California Company system, incorporating all NASTRAN-LCC developments. It is expected that the described procedures will retain their value in conjunction with the new automated coupling capability.

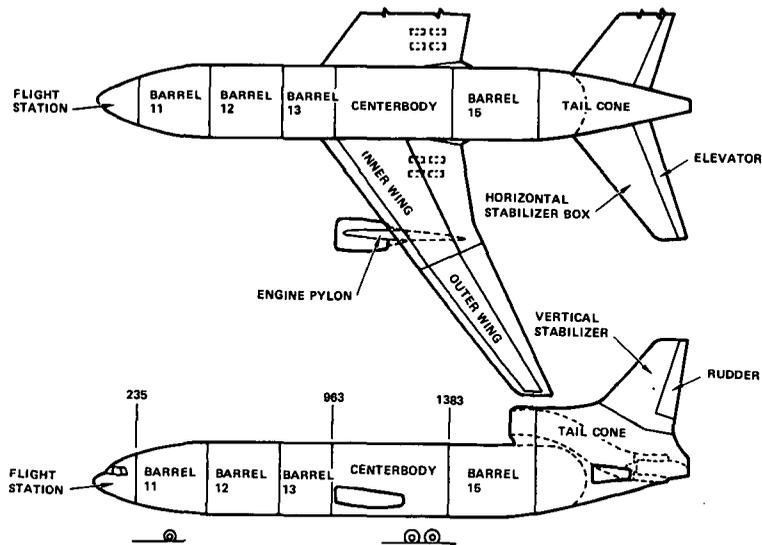


Figure 1. L-1011-500 structural analysis model

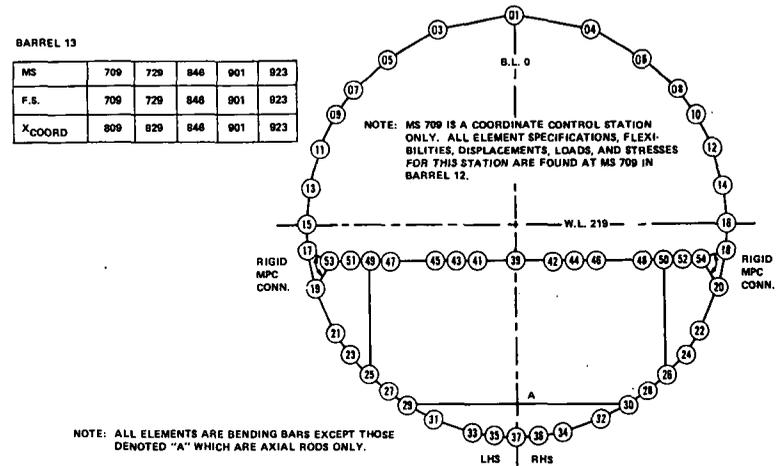
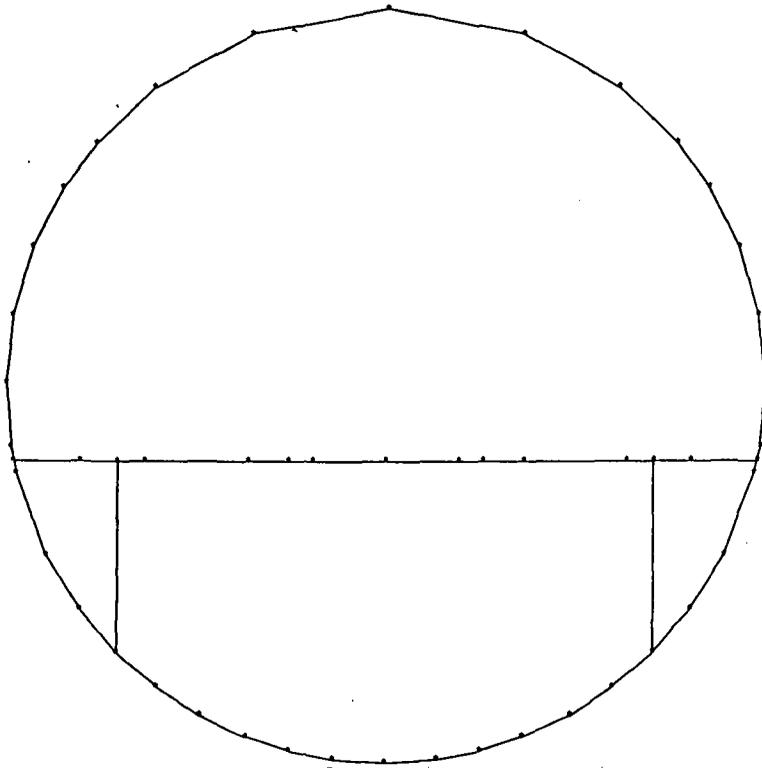
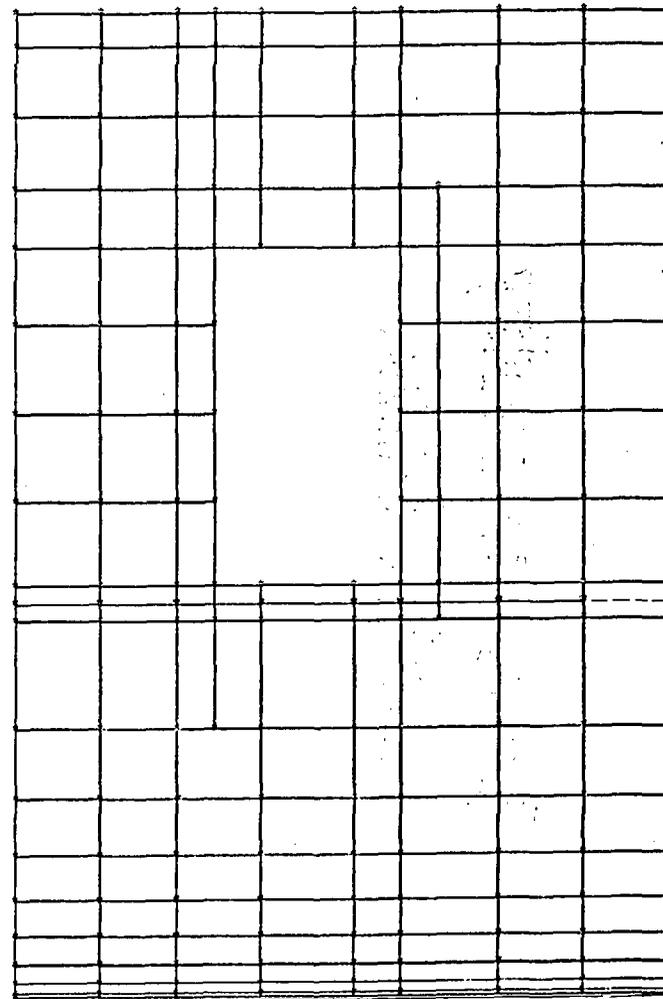


Figure 2. L-1011-500 NASTRAN structural model, barrel 13



L-1011-500 SUBSTRUCTURE MODEL FROM MS = 709 TO MS = 936
UNDEFORMED SHAPE

Figure 3. Frame bar plot



MISSING!

L-1011-500 SUBSTRUCTURE MODEL FROM MS = 709 TO MS = 936
UNDEFORMED SHAPE

Figure 4. Left side view of fuselage
barrel substructure

**Page
Intentionally
Left Blank**

ANALYSIS OF MAGNETIC FIELDS USING VARIATIONAL PRINCIPLES

AND CELAS2 ELEMENTS

John W. Frye and Rolf G. Kasper
Naval Underwater Systems Center

ABSTRACT

Some prospective techniques for analyzing Magnetic Fields using NASTRAN are reviewed. A variational principle utilizing a vector potential function is presented which has as its Euler equations, the required field equations and boundary conditions for static magnetic fields including current sources. The need for an addition to this variational principle of a constraint condition is discussed. Some results using the Lagrange multiplier method to apply the constraint and CELAS2 elements to simulate the matrices are given. Practical considerations of using large numbers of CELAS2 elements are discussed.

LIST OF SYMBOLS

- A - vector potential function (Weber/Meter)
- {A} - vector of discrete vector potentials
- A_i - vector potential at grid point i
- B - magnetic flux density (Telsa, Weber/Meter²)
- c - constant
- D - electric flux density (Coulomb/Meter²)
- [D_I] - stiffness matrix for Lagrange multiplier I = X, Y, Z or r, ψ, z
- D_{I ij} - i, j th element of [D_I] matrix (Ampere·Meter³/Weber)
- E - electric field intensity (Volt/Meter)
- F - volume loading (Newtons/Meter³)
- F_s - surface loading (Newtons/Meter²)

LIST OF SYMBOLS (continued)

\underline{H}	- magnetic field strength (Ampere/Meter)
I	- functional value
\underline{J}	- volume current density (Ampere/Meter ²)
$\frac{J}{s}$	- surface current density (Ampere/Meter)
k	- heat transfer coefficient (Watts/Meter/ ^o C)
$\left[K_{IJ} \right]$	- stiffness matrix of vector potential I = X, Y, Z or r, ψ , z; J = X, Y, Z or r, ψ , Z
$K_{IJ, ij}$	- i, j th element of K_{IJ} matrix (Ampere·Meter ² /Weber)
\underline{n}	- unit normal to a surface
N_i	- finite element interpolation function for grid point i
q	- volume heat input (Watts/Meter ³)
Q	- heat flow (Watts/Meter ²)
r, ψ, z	- cylindrical coordinates
S	- surface (Meter ²)
T	- temperature (^o C)
\underline{u}	- displacement (Meter)
V	- volume (Meter ³)
x, y, z	- rectangular coordinates
ϵ	- permittivity (Farad/Meter)
ϵ_E	- strain
λ	- Lagrange multiplier (Weber/Meter ²)
λ_i	- Lagrange multiplier at grid point i
μ	- permeability (Henry/Meter)
ρ	- volume charge density (Coulomb/Meter ³)
ρ_s	- surface charge density (Coulomb/Meter ²)

LIST OF SYMBOLS (continued)

σ - stress (Pascals, Newton/Meter²)

SUPERSCRIPTS

L - Fourier Harmonic

SUBSCRIPTS

i j - matrix row and column numbers or grid point number

INTRODUCTION

Over the past ten years or so, papers have been appearing in the literature about the analysis of electric and magnetic fields using finite elements (1, 2, 3, 4, 5, 6, 7, 8, 9). A variety of topics have been covered including the analysis of wave guides, wave propagation, and wave scattering phenomena. Until recently, however, a common trait of the papers is that the problems they treated could be resolved by the solution of a set of equations with a single unknown at a grid point. The problems were either posed in terms of scalar potential functions, or they were problems that were restricted such that only a single component of a field or vector potential existed.

For example, in the case of a static magnetic field without currents, the field equations and boundary conditions are:

$$(1) \quad \underline{\nabla} \times \underline{H} = 0$$

$$(2) \quad \underline{\nabla} \cdot \underline{B} = 0$$

$$(3) \quad \underline{B} = \mu \underline{H}$$

$$(4) \quad \underline{n} \times \underline{H} \Big|_{\text{media 1}} = \underline{n} \times \underline{H} \Big|_{\text{media 2}}$$

$$(5) \quad \underline{n} \cdot \underline{B} \Big|_{\text{media 1}} = \underline{n} \cdot \underline{B} \Big|_{\text{media 2}}$$

Problems of this kind can be formulated using a scalar potential function.

$$(6) \quad \underline{H} \approx \underline{\nabla}\phi$$

where ϕ is continuous everywhere in the problem domain. Equation (1) is automatically satisfied since $\underline{\nabla}_x \underline{\nabla}\phi = 0$ for every continuous function ϕ . Equation (2) is satisfied if

$$(7) \quad \underline{\nabla} \cdot \mu \underline{\nabla}\phi = 0$$

Since ϕ is continuous everywhere, the boundary condition of equation (4) is satisfied as it contains only derivations of ϕ tangent to the surface between the two media. If we examine the functional

$$(8) \quad I \approx \iint_V \left[\mu \left(\frac{\partial \phi}{\partial x} \right)^2 + \mu \left(\frac{\partial \phi}{\partial y} \right)^2 + \mu \left(\frac{\partial \phi}{\partial z} \right)^2 \right] dv$$

we find that the Euler equations for this functional satisfy the field equation given by equation (7) and the boundary condition given by (5). This is then a functional we could use to formulate a finite element solution to a magnetic field problem without current sources. Fortunately, this functional is identical to the one used to formulate heat transfer problems in NASTRAN. As a result it is possible to analyze magnetic field problems of this kind through the use of a simple analogy..

potential function ϕ	→	temperature T
permeability μ	→	heat transfer coefficient k
magnetic field flow $\underline{B} = \mu \underline{\nabla}\phi$	→	heat flow $\underline{Q} = -k \underline{\nabla}T$

A similar analogy is possible for static electric fields. In this case, charges may be included as loadings of the problem. The analogy is as follows

potential function ϕ	→	temperature T
permittivity ϵ	→	heat transfer coefficient k
electric field flux $\underline{D} = -\mu \underline{\nabla}\phi$	→	heat flow $\underline{Q} = -k \underline{\nabla}T$
static charge ρ	→	heat input q

Many interesting and useful problems can be solved with scalar potential functions or with fields in which only one component of a vector potential function or field direction need be considered. More general problems however,

do not lend themselves to treatments where only one degree of freedom exists at a grid point. In the case in hand we are concerned with the general static magnetic field problem where currents exist. The following equations serve to define this problem.

$$(9) \quad \underline{\nabla} \times \underline{H} = \underline{J}$$

$$(10) \quad \underline{\nabla} \cdot \underline{B} = 0$$

$$(11) \quad \underline{B} = \mu \underline{H}$$

$$(12) \quad \underline{n}_1 \times \underline{H} \Big|_{\text{media 1}} = \underline{n}_1 \times \underline{H} \Big|_{\text{media 2}} + \frac{\underline{J}_s}{s}$$

$$(13) \quad \underline{n}_1 \cdot \underline{B} \Big|_{\text{media 1}} = \underline{n}_1 \cdot \underline{B} \Big|_{\text{media 2}}$$

It can be seen that the substitution $\underline{H} = \underline{\nabla} \phi$ does not work since equation (9) would not in general, be satisfied. A vector function formulation can be used to get rid of equation (10).

$$(14) \quad \underline{B} = \underline{\nabla} \times \underline{A}$$

Where \underline{A} is continuous everywhere in the domain of the problem.

We know that $\underline{\nabla} \cdot \underline{\nabla} \times \underline{A} = 0$ for any vector function \underline{A} from a theorem of vector calculus. We also know that the boundary condition of equation (13) is automatically satisfied since \underline{A} is continuous and since the boundary condition contains only derivatives of \underline{A} tangent to the surface between the media.

Using equations (11) and (14) in the field equation (9) we get

$$(15) \quad \underline{\nabla} \times \left(\frac{1}{\mu} \underline{\nabla} \times \underline{A} \right) = \underline{J}$$

This equation, along with the boundary condition given by equation (12), must be satisfied in order to solve the general static magnetic field problem with currents.

It is tempting to look for analogies by which those equations may be satisfied. Such analogies could hopefully use finite elements already existing in NASTRAN and would allow immediate formulation of models for magnetic field problems. One interesting analogy makes use of the field equations for general isotropic elastic solids.¹⁰

$$(16) \quad (\lambda + G) (\underline{\nabla} \cdot \underline{U}) + G \underline{\nabla}^2 \underline{U} + \underline{F} = 0$$

If we set $\lambda = -\frac{2}{\mu}$, $G = \frac{1}{\mu}$, $\underline{F} = \underline{J}$, and $\underline{U} = \underline{A}$, in equation (16) we have

$$-\frac{1}{\mu} \underline{\nabla} \cdot (\underline{\nabla} \cdot \underline{A}) + \frac{1}{\mu} \underline{\nabla}^2 \underline{A} + \underline{J} = 0$$

$$-\frac{1}{\mu} \nabla_x \nabla_x A + J = 0$$

$$(17) \quad \frac{1}{\mu} \nabla_x \nabla_x A = J$$

In other words, with the above substitutions, the field equation (15) is evidently simulated at least in regions where μ isn't changing. But what about situations where the permeability μ does change? Typically, μ will change abruptly such as at a surface between a metal and air or free space. The boundary condition given by equation (12) governs such situations. The analogy given above says nothing about this boundary condition; it only provides that the homogeneous field equation be simulated. If we were to solve a problem using this analogy with finite elements, we would make use of a scheme by which the element matrices are calculated. This scheme is derived from an integral equation in which the total strain energy of the problem is evaluated.

$$(18) \quad I = \int \frac{1}{2} \{ \sigma_{\epsilon_E} - \underline{F} \cdot \underline{u} \} dV - \int \underline{F}_S \cdot \underline{u} dS$$

We know that the Euler equations obtained by minimizing this functional are the field equations and surface boundary conditions of an elastic solid. It turns out that with the substitutions of the above analogy, the Euler conditions for the surfaces do not match the boundary conditions given by equation (12). Thus, while this particular analogy is useful in solving some problems, we run into difficulties with it in specifying boundary conditions when problems with complicated non-homogeneous domains are involved.

VARIATIONAL PRINCIPLE FOR MAGNETIC FIELDS WITH CURRENTS

What is really needed for general static magnetic field problems is a functional that when minimized gives Euler equations that correspond to the required field and boundary conditions. Fortunately, such a functional exists. 11, 12, 13, 14, 15

$$(19) \quad I = \int_V \left\{ -\frac{1}{2\mu} (\nabla_x A) \cdot (\nabla_x A) + \underline{J} \cdot A \right\} dV$$

$$+ \int_S \underline{J}_S \cdot A dS$$

The functional of equation (19) has Euler equations corresponding to the field equation (15) and the boundary condition equation (12). One is therefore tempted to make the usual finite element interpolation function approximations, substitute them into equation (19), and minimize the function with respect to the discrete variables.

$$(20) \quad \underline{A} = N_i \cdot \underline{A}_i$$

Where K is the element number

i is the grid point number

N is the interpolation function

Before one does all of this however, it is well to notice that the solutions to equations (15) and (12) are not unique. Further, it turns out that the conditions which must be imposed to provide uniqueness apply not just to the boundaries of the domain, but to the field points as well. The condition that is usually imposed on the field points is

$$(21) \quad \underline{\nabla} \cdot \underline{A} = 0 \text{ everywhere.}$$

The need to impose a constraint on the field points is different from our usual experience in structural mechanics where we need only apply constraints on boundaries to provide uniqueness. Without applying the condition of equation (21), one might expect that assembled matrix equations derived from the variational principle of equation (19) to be singular. One might expect this since singular matrix equations allow non-unique solutions. That is, more than one vector of unknowns may satisfy a singular matrix equation. It has been our experience with the limited number of models we have examined, that this, in fact, is the case for matrices derived using equation (19). Others, for instance, Becker from the University of Texas have noted the same thing. Theoretical work by Konrad also supports this view.¹⁵

To insure that the constraint given by equation (21) is considered in the finite element formulation, a Lagrange multiplier may be used. With a Lagrange multiplier, an additional term is added to the functional.

$$(22) \quad I = \int_V \left\{ -\frac{1}{2} \frac{1}{\mu} (\underline{\nabla} \times \underline{A}) \cdot (\underline{\nabla} \times \underline{A}) + \lambda \frac{1}{\mu} (\underline{\nabla} \cdot \underline{A}) + \underline{J} \cdot \underline{A} \right\} dV \\ + \int_{S-S} \underline{J} \cdot \underline{A} dS$$

where λ is the Lagrange multiplier.

We have tried other methods of imposing this constraint, but these attempts have met with unsuccessful results. The Lagrange multiplier is carried along as an additional unknown in the formulation. It also is approximated by the finite element interpolation functions and discrete unknowns associated with the various grid points.

$$(23) \quad \lambda = N_i \lambda_i$$

If we substitute the finite element approximations of equations (20) and (23) into equation (22) we would have the following functional forms.

$$(24) \quad I = \int_V \left\{ \begin{array}{l} \text{RECTANGULAR COORDINATES} \\ -\frac{1}{2} \frac{1}{\mu} \left(\frac{\partial N_i}{\partial y} A_{z_i} - \frac{\partial N_i}{\partial z} A_{y_i} \right) \left(\frac{\partial N_j}{\partial y} A_{z_j} - \frac{\partial N_j}{\partial y} A_{y_j} \right) \\ -\frac{1}{2} \frac{1}{\mu} \left(\frac{\partial N_i}{\partial z} A_{x_i} - \frac{\partial N_i}{\partial x} A_{z_i} \right) \left(\frac{\partial N_j}{\partial z} A_{x_j} - \frac{\partial N_j}{\partial x} A_{z_j} \right) \\ -\frac{1}{2} \frac{1}{\mu} \left(\frac{\partial N_i}{\partial x} A_{y_i} - \frac{\partial N_i}{\partial y} A_{x_i} \right) \left(\frac{\partial N_j}{\partial x} A_{y_j} - \frac{\partial N_j}{\partial y} A_{x_j} \right) \end{array} \right\}$$

$$\begin{aligned}
& + \frac{N_i \lambda_i}{\mu} \left(\frac{\partial N_j}{\partial x^j} A_{x_j} + \frac{\partial N_j}{\partial y^j} A_{y_j} + \frac{\partial N_j}{\partial z^j} A_{z_j} \right) \\
& + N_i A_{x_i} J_x + N_i A_{y_i} J_y + N_i A_{z_i} J_z \} dV \\
& + \left(J_{sx} N_i A_{x_i} + J_{sy} N_i A_{y_i} + J_{sz} N_i A_{z_i} \right) dS
\end{aligned}$$

CYLINDRICAL COORDINATES

$$\begin{aligned}
(25) \quad I = \int_V \{ & - \frac{1}{2} \frac{1}{\mu} \left(\frac{1}{r} \frac{\partial N_i}{\partial \psi} A_{z_i} - \frac{\partial N_i}{\partial z} A_{\psi_i} \right) \left(\frac{1}{r} \frac{\partial N_j}{\partial \psi} A_{z_j} - \frac{\partial N_j}{\partial z} A_{\psi_j} \right) \\
& - \frac{1}{2} \frac{1}{\mu} \left(\frac{\partial N_i}{\partial z} A_{r_i} - \frac{\partial N_i}{\partial r} A_{z_i} \right) \left(\frac{\partial N_j}{\partial z} A_{r_j} - \frac{\partial N_j}{\partial r} A_{z_j} \right) \\
& - \frac{1}{2} \frac{1}{\mu} \left(\frac{N_i}{r} A_{\psi_i} + \frac{\partial N_i}{\partial r} A_{\psi_i} - \frac{1}{r} \frac{\partial N_i}{\partial \psi} A_{r_i} \right) \left(\frac{N_j}{r} A_{\psi_j} + \frac{\partial N_j}{\partial r} A_{\psi_j} - \frac{1}{r} \frac{\partial N_j}{\partial \psi} A_{r_j} \right) \\
& + \frac{N_i \lambda_i}{\mu} \left(\frac{N_j}{r} A_{r_j} + \frac{\partial N_j}{\partial r} A_{r_j} + \frac{1}{r} \frac{\partial N_j}{\partial \psi} A_{\psi_j} + \frac{\partial N_j}{\partial z} A_{z_j} \right) \\
& + N_i A_{r_i} J_r + N_i A_{\psi_i} J_{\psi} + N_i A_{z_i} J_z \} dV \\
& + \iint_S (J_{rs} N_i A_{r_i} + J_{\psi s} N_i A_{\psi_i} + J_{zs} N_i A_{z_i}) dS
\end{aligned}$$

Having arrived at the functionals, it is a relatively straightforward process to derive the matrix equations that are the conditions of minimization of the functional and that serve to approximate algebraically the field equation and boundary condition of equations (12), (15), and (21). For the functional written in terms of rectangular coordinates the matrix equations have the following form.

$$(26) \quad \begin{bmatrix} [K_{xx}] \\ [K_{xy}] \\ [K_{xz}] \\ [D_x]^T \end{bmatrix}^T \begin{bmatrix} [K_{xy}] \\ [K_{yy}] \\ [K_{yz}] \\ [D_x]^T \end{bmatrix}^T \begin{bmatrix} [K_{xz}] \\ [K_{yz}] \\ [K_{zz}] \\ [D_z]^T \end{bmatrix}^T \begin{Bmatrix} \{A_x\} \\ \{A_y\} \\ \{A_z\} \end{Bmatrix} + \begin{bmatrix} [D_x] \\ [D_y] \\ [D_z] \end{bmatrix} \{\lambda\} = \begin{Bmatrix} \{J_x\} \\ \{J_y\} \\ \{J_z\} \end{Bmatrix} \\
= \{0\}$$

The formulation is seen to be symmetric, but has the peculiar property of not having any diagonal terms multiplying the Lagrange multipliers $\{\lambda\}$. The equations for the matrices of a specific finite element are given below

for the formulation with rectangular coordinates.

$$(27) \quad K_{xx_{ij}} = \int_V \frac{1}{\mu} \left(\frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} + \frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z} \right) dV$$

$$(28) \quad K_{yy_{ij}} = \int_V \frac{1}{\mu} \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z} \right) dV$$

$$(29) \quad K_{zz_{ij}} = \int_V \frac{1}{\mu} \left(\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} \right) dV$$

$$(30) \quad K_{xy_{ij}} = -\int_V \frac{1}{\mu} \left(\frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial x} \right) dV$$

$$(31) \quad K_{xz_{ij}} = -\int_V \frac{1}{\mu} \left(\frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial x} \right) dV$$

$$(32) \quad K_{yz_{ij}} = -\int_V \frac{1}{\mu} \left(\frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial y} \right) dV$$

$$(33) \quad D_{x_{ij}} = -\int_V \frac{1}{\mu} \frac{\partial N_i}{\partial x} N_j dV$$

$$(34) \quad D_{y_{ij}} = -\int_V \frac{1}{\mu} \frac{\partial N_i}{\partial y} N_j dV$$

$$(35) \quad D_{z_{ij}} = -\int_V \frac{1}{\mu} \frac{\partial N_i}{\partial z} N_j dV$$

$$(36) \quad J_{x_i} = \int_V J_{x_i} N_i dV + \int_S J_{x_s} N_i dS$$

$$(37) \quad J_{y_i} = \int_V J_{y_i} N_i dV + \int_S J_{y_s} N_i dS$$

$$(38) \quad J_{z_i} = \int_V J_{z_i} N_i dV + \int_S J_{z_s} N_i dS$$

Similar equations can obviously be developed to cylindrical coordinates. If the geometry of the bodies in the field are axisymmetric, the field may be broken up into different components, each representing the field response of a particular Fourier harmonics. The governing matrix equations separate for each Fourier harmonic and the general three-dimensional problem can be handled as a summation of two-dimensional solutions.

For each Fourier harmonic: L

$$(39) \begin{bmatrix} [K_{rr}^{(L)}] & [K_{r\psi}^{(L)}] & [K_{rz}^{(L)}] \\ [K_{r\psi}^{(L)T}] & [K_{\psi\psi}^{(L)}] & [K_{\psi z}^{(L)}] \\ [K_{rz}^{(L)T}] & [K_{\psi z}^{(L)T}] & [K_{zz}^{(L)}] \\ [D_r^{(L)T}] & [D_\psi^{(L)T}] & [D_z^{(L)T}] \end{bmatrix} \begin{Bmatrix} \{A_r^{(L)}\} \\ \{A_\psi^{(L)}\} \\ \{A_z^{(L)}\} \\ \{A_r^{(L)}\} \\ \{A_\psi^{(L)}\} \\ \{A_z^{(L)}\} \end{Bmatrix} + \begin{bmatrix} D_r^{(L)} \\ D_\psi^{(L)} \\ D_z^{(L)} \end{bmatrix} \{\lambda\}^{(L)} = \begin{Bmatrix} J_r^{(L)} \\ J_\psi^{(L)} \\ J_z^{(L)} \end{Bmatrix} = \{0\}$$

$$(40) K_{rr_{ij}}^{(L)} = \iint \frac{c}{\mu} \left(\frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z} + \frac{N_i N_j}{r^2} L^2 \right)$$

$$(41) K_{\psi\psi_{ij}}^{(L)} = \iint \frac{c}{\mu} \left(\frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial t} + \frac{N_i N_j}{r^2 j} + \frac{\partial N_i}{\partial r} \frac{\partial N_j}{\partial r} + \frac{\partial N_i}{\partial r} \frac{N_j}{r} + \frac{N_i}{r} \frac{\partial N_j}{\partial r} \right) r dr dz$$

$$(42) K_{zz_{ij}}^{(L)} = \iint \frac{c}{\mu} \left(\frac{\partial N_i}{\partial r} \frac{\partial N_j}{\partial r} + \frac{N_i N_j}{r^2 j} L^2 \right) r dr dz$$

$$(43) K_{r\psi_{ij}}^{(L)} = \iint \frac{c}{\mu} \left(\frac{N_i N_j}{r^2 j} L + \frac{N_i}{r} \frac{\partial N_j}{\partial r} L \right) r dr dz$$

$$(44) K_{rz_{ij}}^{(L)} = -\iint \frac{c}{\mu} \left(\frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial r} \right) r dr dz$$

$$(45) K_{\psi z_{ij}}^{(L)} = \iint \frac{c}{\mu} \left(\frac{\partial N_i}{\partial z} \frac{N_j}{r} L \right) r dr dz$$

$$(46) D_{r_{ij}}^{(L)} = -\iint \frac{c}{\mu} \left(\frac{N_i}{r} N_j + \frac{\partial N_i}{\partial r} N_j \right) r dr dz$$

$$(47) D_{\psi_{ij}}^{(L)} = -\iint \frac{c}{\mu} \left(\frac{N_i N_j}{r} L \right) r dr dz$$

$$(48) D_{z_{ij}}^{(L)} = -\iint \frac{c}{\mu} \left(\frac{\partial N_i}{\partial z} N_j \right) r dr dz$$

$$(49) J_{r_i}^{(L)} = \iint c J_r^{(L)} N_i r dr dz$$

$$(50) J_{\psi_i}^{(L)} = \iint c J_\psi^{(L)} N_i r dr dz$$

$$(51) \quad J_{z_i}^{(L)} = \int \int c J_z^{(L)} N_i r dr dz$$

$$\text{where } c = 2\pi \text{ for } L = 0 \\ c = \pi \text{ for } L \geq 1$$

$J_r^{(L)}$, $J_\psi^{(L)}$, and $J_z^{(L)}$ are defined by

$$(52) \quad J_r^{(L)} = \frac{1}{c} \int J_r \cos L\psi \, d\psi$$

$$(53) \quad J_\psi^{(L)} = \frac{1}{c} \int J_\psi \sin L\psi \, d\psi \quad L > 1$$

$$(54) \quad J_z^{(L)} = \frac{1}{c} \int J_z \cos L\psi \, d\psi$$

$$(55) \quad J_\psi^{(0)} = \frac{1}{2\pi} \int J_\psi \, d\psi$$

A SAMPLE PROBLEM AND THE USE OF CELAS2 ELEMENTS

A major problem in trying to apply either of the functionals of equations (24) or (25) is that NASTRAN has apparently no equivalent formulation for any element presently available. The reason for this is probably that the calculation of strain energy involves derivatives of displacement that are different from the derivatives of the vector potential function used to calculate the energy in a magnetic field. The difficulty arises in trying to simulate the finite element matrices without coding a whole new element into the NASTRAN program directly. When testing out a new element formulation with NASTRAN, we would like to use the cheapest and quickest method possible. There are two approaches to simulating finite element matrices with ordinary NASTRAN bulk data cards which are particularly attractive. These approaches call for the use of either the GENEL card or the CELAS element. We have attempted to use the GENEL card, but ran into some problems with it. The problems were encountered with level 15,5 of the program and are perhaps fixed in level 16. We have had some success in using the CELAS elements.

That is not to say that we have found the simulation of finite elements with CELAS elements particularly convenient. A CELAS element is required for each matrix element of every finite element. For even small finite element models large data decks are required. The small model shown in figure 2 contains 200 simplex triangular elements. Each triangular element has three grid points with three degrees of freedom each. It takes 42 CELAS elements to simulate each simplex triangular element. Since there are 200 such elements, 8400 CELAS elements are required to simulate the finite elements of this model. Naturally, these CELAS cards are generated automatically and stored electronically so that they never have to be handled in card form.

There are problems that one runs into when using large numbers of CELAS elements for simulating finite element matrices. The question of how many significant figures you would like to have in the data is important. If the

data is represented in E format, only three significant figures will fit on the standard eight column NASTRAN field. Of course, the expanded field format can be used, but a continuation card is then required for each CELAS element and the number of records in the CELAS element file is doubled. It turns out that NASTRAN's XSORT routine requires one location in core for each continuation card in the bulk data deck. It also turns out that if continuation cards are present in the bulk data the XSORT routine will be called even if the deck is already sorted. If the CELAS element file becomes very large, which can easily happen, then a situation can develop where the NASTRAN program won't execute because of the core requirements of the XSORT routine. A way out of this problem is to scale the model so that for all, or nearly all, elements an F format can be used to represent the stiffness values. In this way a maximum of six significant figures can be put onto a eight column field. It is not the most desirable solution, but at least it avoids the problems attendant with large files of cards with expanded field format.

Even using the standard NASTRAN cards, the XSORT routine can consume a disproportionate amount of time for large data decks. For the model of figure 2, with 8400 CELAS2 cards, the XSORT routine took 134 CPU seconds on a UNIVAC 1108 EXEC 8 system. The decomposition routine SSG3 in comparison for this same problem required only 4.1 seconds of CPU time.

For all the problems encountered with using CELAS elements to simulate finite element matrices, the approach does seem to work, and where we are only interested in running small problems to check out prospective finite element formulations, this is all that really matters,

Figure 1 shows a simple axisymmetric magnetic field problem. A toroidal solid with a square cross section has a sheet of current running around its outer surface. The only magnetic field developed by the current is within the current sheet; outside the current sheet the field is zero. The current loading is axisymmetric and only the zeroth Fourier harmonic of the field need be evaluated. The unknowns of the problem are the vector potentials A_r and A_z and the Lagrange multiplier. The finite element model is shown in figure 2. SPC constraint conditions were imposed on the A_r components at the axis where all radial components must be zero and on an A_z component at one grid point on the axis. This model ran without any serious incident on level 15,5 NASTRAN using a UNIVAC 1108 EXEC 8 system. NASTRAN did give a singularity table for some of the degrees of freedom representing the Lagrange multiplier. These messages may have occurred due to round-off error in assembling certain diagonal components of the stiffness matrix associated with the Lagrange multipliers. Small negative numbers may have been generated instead of zeros.

The distorted mesh pattern is shown in figure 3. Note that the region within the current sheet undergoes a definite rotation while the model elsewhere undergoes very little rotation. The magnetic flux is equal to the curl of \underline{A} which is a measure of rotation of the \underline{A} field at a point. Thus, the mesh distortion is about what should be expected since we expect to see a magnetic flux inside the current sheet and none outside.

A plot of the magnetic flux values computed from the curl of \underline{A} using the

finite element interpolation functions is shown in figure 4. The agreement with the exact solution is good within the region of the current sheet. The solution showed minor deviations from the zero value outside the current sheet for positions away from the axis. More serious deviations from the zero value occurred at elements near the axis. The reasons for these larger deviations at elements near the axis is unexplained at this time, but may have to do with the course mesh size and the numerical integration scheme used to evaluate the stiffness matrices.

RECOMMENDATIONS AND DEVELOPMENT OF ELEMENTS FOR ELECTROMAGNETIC ELEMENTS

It is recommended that elements for general static magnetic fields be included in NASTRAN. The formulations presented here have certain advantages in that the functional used to generate the elements produces Euler equations which correspond to the desired field equations and boundary conditions for magnetic fields.

Variational principles may be able to be applied to time varying fields also. In time varying cases, electric and magnetic fields couple to form electromagnetic fields. A functional for electromagnetic fields which uses scalar and vector potential functions and a Lagrange multiplier to impose a Lorentz gage condition is given below.

$$(52) \quad I = \int_t \int_v \left\{ \frac{1}{2} \left[\epsilon (\underline{\nabla} \phi + \underline{A}) \cdot (\underline{\nabla} \phi + \underline{A}) - \frac{1}{2\mu} (\underline{\nabla} \cdot \underline{A}) \cdot (\underline{\nabla} \times \underline{A}) - \rho \phi + \underline{J} \cdot \underline{A} + \lambda \left(\frac{1}{\mu} \underline{\nabla} \cdot \underline{A} + \epsilon \dot{\phi} \right) \right] \right\} dV \\ + \int_t \int_s (-\rho_s \phi + \underline{J}_s \cdot \underline{A}) dS$$

This functional, when minimized, produces Euler equations and boundary conditions of electromagnetic fields. Simply using finite elements over a limited region of interest is not sufficient however for a complete model of an electromagnetic field since electromagnetic waves propagate through space. Techniques must be established to treat the boundaries of domains modeled with finite elements in order to eliminate false reflections. There are prospective methods of treating these boundaries, and acoustic fields and some electromagnetic fields have already been successfully treated using them.

REFERENCES

1. Arlett, P.; Bahrani, A.; and Zienkiewicz, O.: Application of Finite Elements to the Solution of Helmholtz's Equation, PROC IEE, vol. 115, no. 12, Dec. 1968,
2. Ahmed, S.: Finite Element for Wave Guide Problems, Electronics Letters, vol. 4, no. 18, Sept, 1968,
3. Silvester, P.: High Order Polynomial Triangular Finite Elements for Potential Problems: Int. J, Engrg, Sci, 7,849-861 (1969),
4. Daly, P.: Finite Elements for Field Problems in Cylindrical Coordinates. Int. J, Num, Meth, Engrg, 6, 169-178 (1973),
5. Doner, J.; Guiliani, S.; and Philippe, A.: Finite Elements in the Solution of Electromagnetic Induction Problems, Int. J, Num, Meth, Engrg, 8, 359-367 (1974),
6. Mei, K.: Unimoment Method of Solving Antenna and Scattering Problems, IEEE Trans. on Antennas and Propagation, Nov, 1974,
7. Chang, S.; Mei, K.: Application of the Unimoment Method to Electromagnetic Scattering of Dielectric Cylinders, IEEE Trans, on Antennas and Propagation, vol. AP-24, no, 1, Jan, 1976,
8. Brauer, J.: Finite Element Analysis of Electromagnetic Induction in Transformers, A 77 122-S, IEEE PES Winter Meeting, New York, N. Y., Jan, 30 - Feb, 4, 1977,
9. Brauer, J.: Saturated Magnetic Energy Functional for Finite Element Analysis of Electric Machines. C75 151-6, IEEE PES Winter Meeting, New York, N, Y, Jan, 26-31, 1975,
10. Kasper, R.: Equivalent Electromagnetic/Acoustic Formulations Developed from the Theory of Elasticity. NUSC Technical Report 5533, 9 Dec. 1976,
11. Tolstoy, I.: Wave Propagation, McGraw-Hill Book Company, Inc., N. Y, 1973
12. Finlayson, B.: The Method of Weighted Residuals and Variational Principles Academic Press, New York, 1972.
13. Guancial, E.; DasGupta, S.: Three Dimensional Finite Element Program for Magnetic Field Problems, IEEE Transactions on Magnetics, vol. Mag-13, no. 3, May 1977,
14. Frye, J.: Electromagnetic Finite Elements from Variational Principles. NUSC Technical Report 5517, 26 Nov. 1976,
15. Konrad, A.: Vector Variational Formulation of Electromagnetic Fields in Anisotropic Media. IEEE Transactions on Microwave Theory and Techniques, vol. MTT-24, no, 9, Sept. 1976.

I.D. = 6M
O.D. = 10M
Total Current is one Amp

$\mu = 1.2566$ HENRY/METER

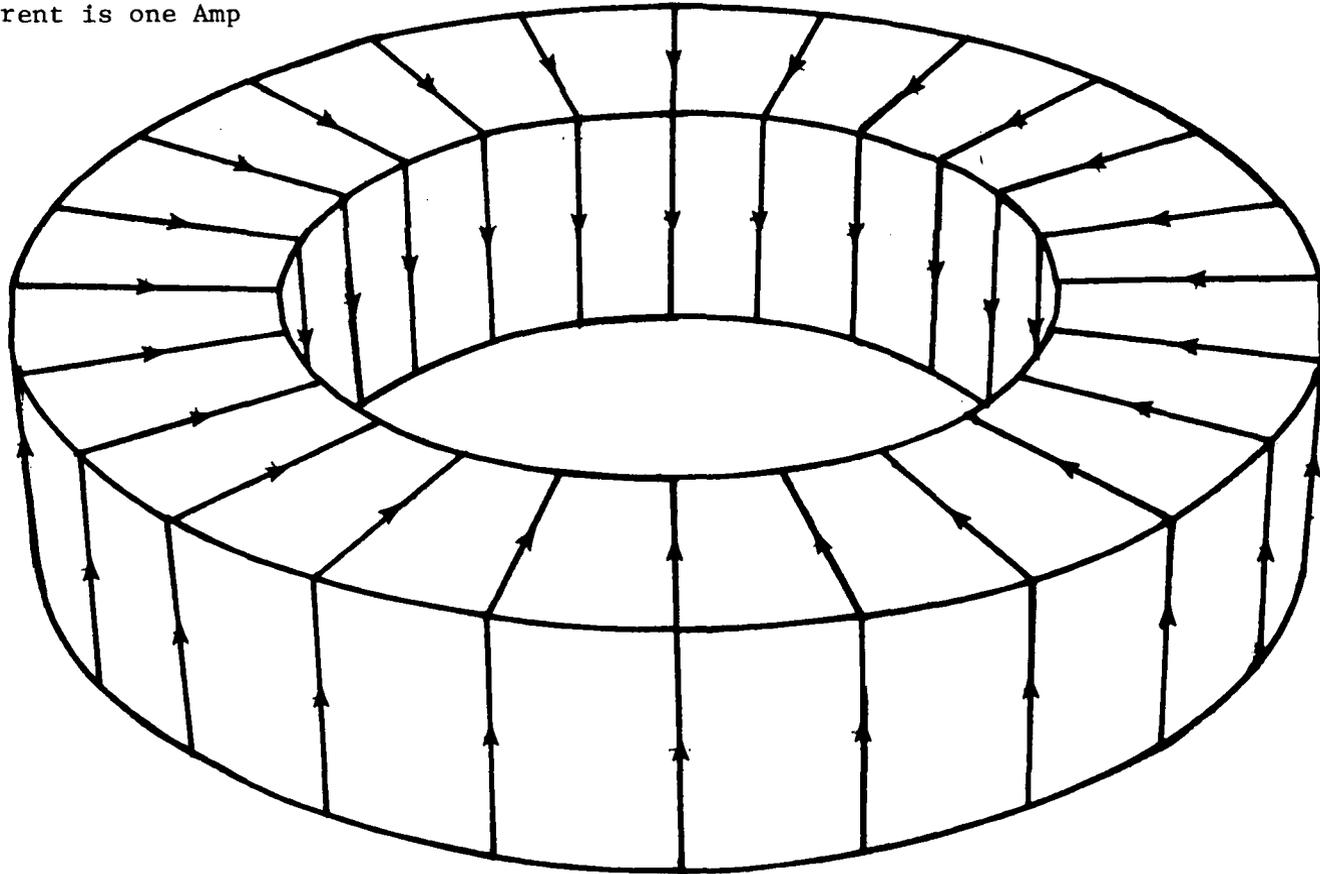


FIGURE 1
SQUARE TOROID SURROUNDED BY CURRENT SHEET

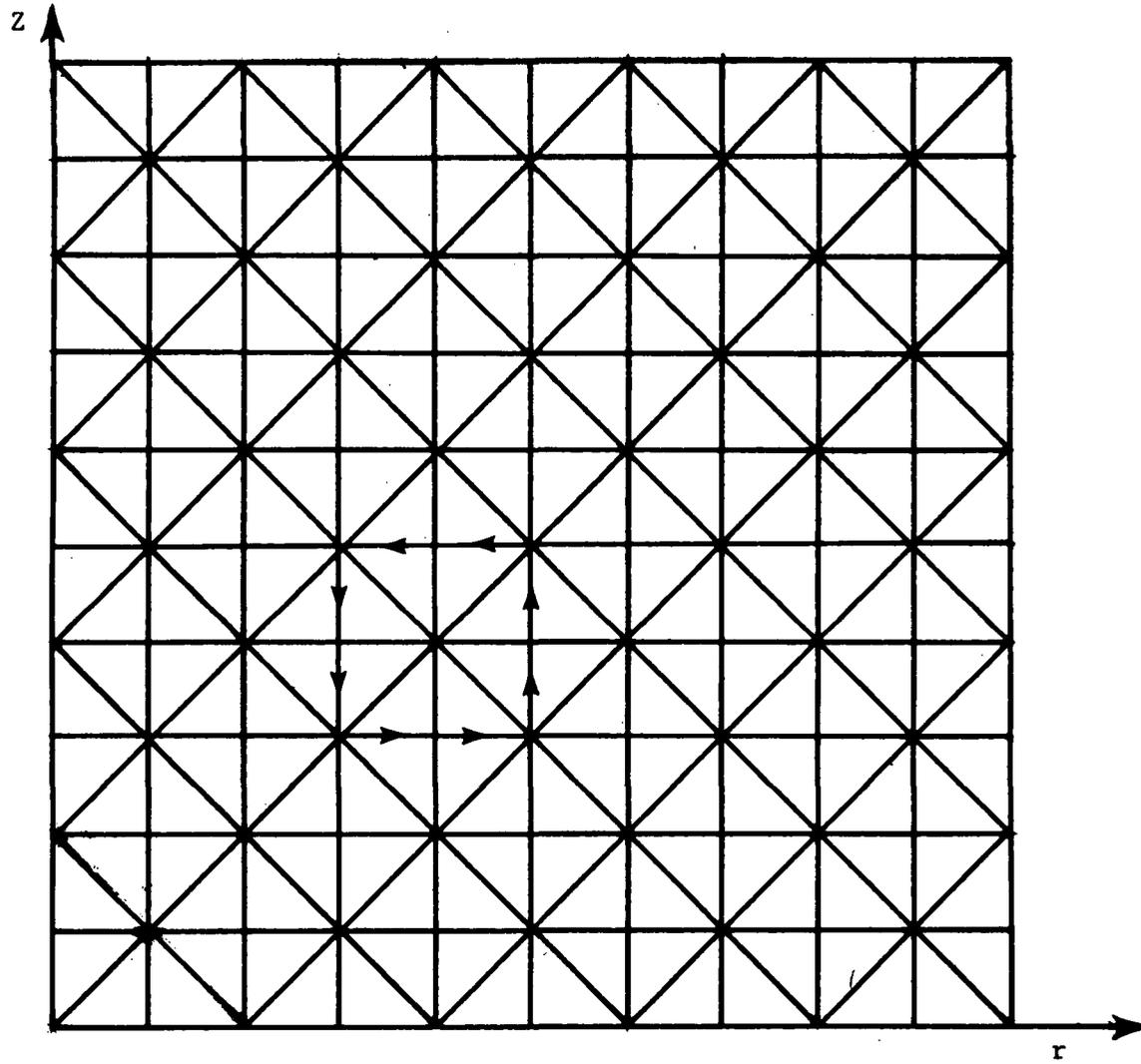


FIGURE 2
FINITE ELEMENT MODEL OF SQUARE TOROID AND FIELD AROUND IT

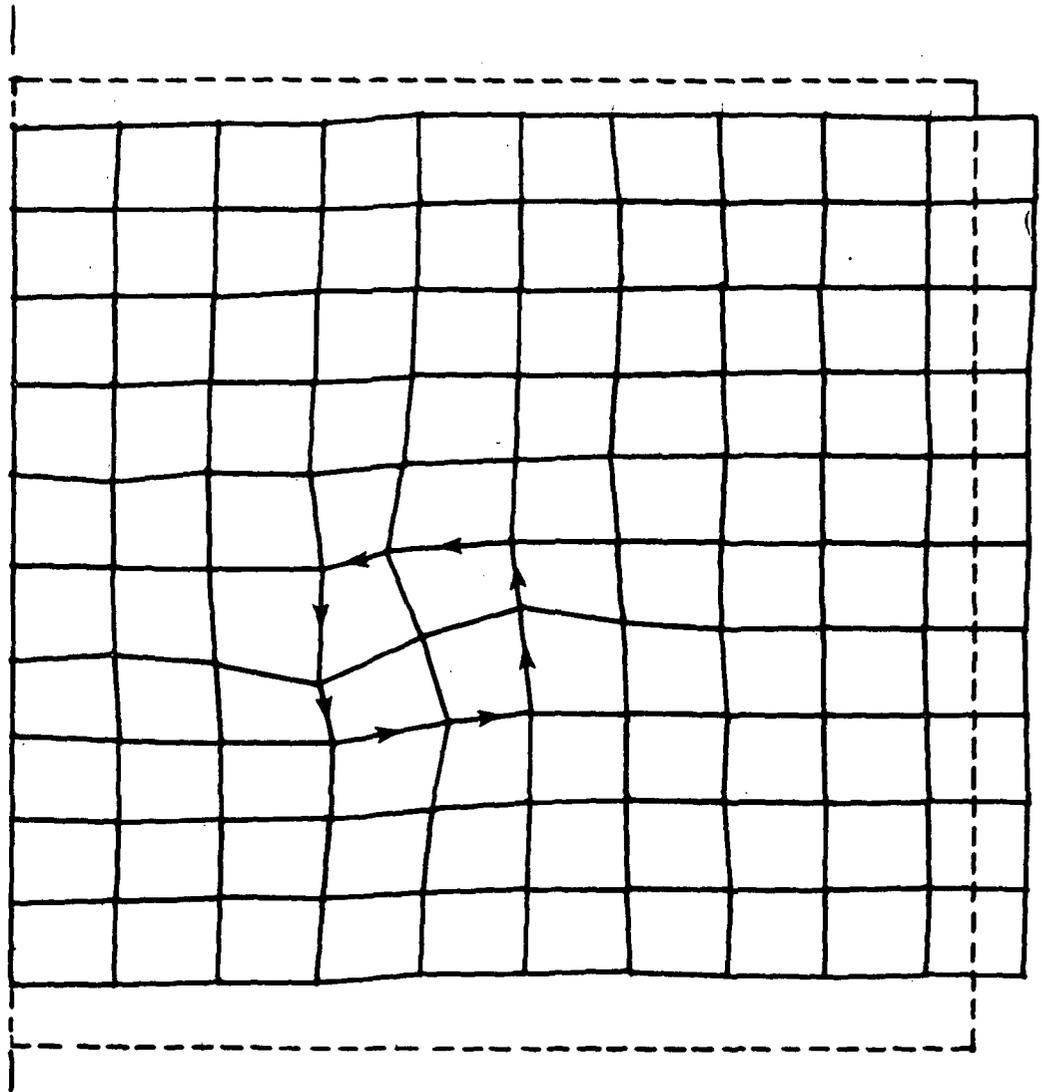


FIGURE 3
"DISPLACED" MESH OF SQUARE SOLINOID

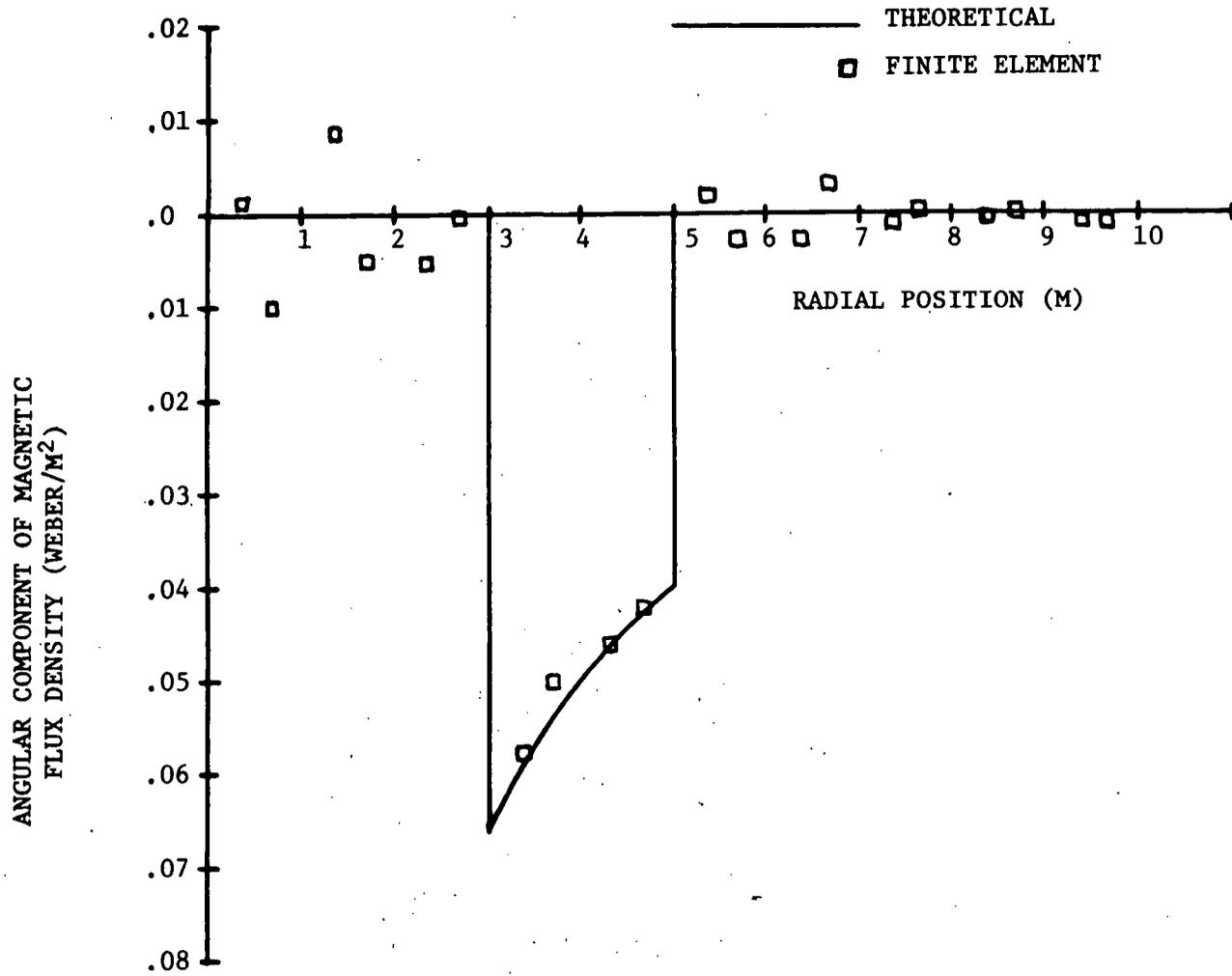


FIGURE 4

FINITE ELEMENT VERSUS THEORETICAL RESULTS FOR
MAGNETIC FIELD IN AND AROUND TOROIDAL SOLINOID

APPLICATION OF SYMBOLIC/NUMERIC MATRIX SOLUTION TECHNIQUES TO THE NASTRAN PROGRAM

E. M. Buturla
S. H. Burroughs

IBM System Products Division

ABSTRACT

The matrix-solving algorithm of any finite element algorithm is extremely important since solution of the matrix equations requires a large amount of elapsed time due to null calculations and excessive I/O operations. In this work we present an alternate method of solving the matrix equations. A symbolic processing step followed by numeric solution yields the solution very rapidly and is especially useful for nonlinear problems.

SYMBOLIC/NUMERIC FACTORIZATION

The concept of using symbolic/numeric matrix solution techniques has been shown to be extremely efficient especially when the system of equations is being solved a number of times, such as transient problems or nonlinear problems (1-4). The symbolic factorization is done only once, after the global coefficient matrix has been assembled. The numeric factorizations can then be found very rapidly.

The basic factorization routines are part of the IBM Program Product SL-MATH (5). A large number of different routines are available; in this work we utilize the routines which assume a positive, definite symmetric matrix since that is the type of matrix encountered with NASTRAN. An ordering algorithm is first employed to reduce the amount of "fill" that occurs in the reduction of the system of equations. This step is somewhat analogous to using BANDIT and has been shown (2) to reduce storage requirements for the factorization stage by up to 40%.

The factorization of the matrix equation

$$[A] \{X\} = \{b\}$$

is by Cholesky's method, resulting in

$$[L] [D] [L]^T \{X\} = \{b\}$$

Where $[L]$ is a lower triangular matrix with unit diagonals,
 $[D]$ is a diagonal matrix, and
 $[L]^T$ is the transpose of $[L]$.

The symbolic factorization determines the positions of the non-zero elements in L . The numeric factorization determines the actual value of the non-zero matrix elements. Note that for a problem where the values of the coefficients change, such as in geometrically nonlinear stress problems, semiconductor transport problems or other nonlinear problems, only numeric factorization need be done when resolving the matrix equations. Also note, however, that for some problems the above will not be true.

The coefficient matrix is stored in compressed form to reduce storage requirements. Only the non-zero terms of the matrix and the row and column pointers are required. The row pointers indicate where in the column pointer vector a particular row is stored. For example, if the full coefficient matrix is

$$\begin{bmatrix} 10 & -1 & 0 & 0 & 3 & 0 \\ & 7 & -3 & 0 & 0 & 4 \\ & & 8 & 0 & -1 & 0 \\ & & & 4 & 0 & 2 \\ & & & & 7 & -1 \\ & & & & & 6 \end{bmatrix}$$

then the compressed vector of matrix terms becomes

$$[A] = [10, -1, 3, 7, -3, 4, 8, -1, 4, 2, 7, -1, 6]$$

the column pointers become

$$\{JA\} = \{1, 2, 5, 2, 3, 6, 3, 5, 4, 6, 5, 6, 6\}$$

and the row pointers become

$$\{IA\} = \{1, 4, 7, 9, 11, 13, 14\}$$

We can characterize any row, say 3, by first looking to $IA(3) = 7$ and $IA(4) = 9$. This indicates that row 3 of the matrix has its column information stored in the 7th and 8th positions in $\{JA\}$, resulting in column values of 3 and 5. The coefficients are stored in the 7th and 8th positions of $[A]$. Note that with this scheme an additional entry to the end of $\{IA\}$ is needed.

Currently, the appropriate SL-MATH routines are called by a stand-alone FORTRAN program which allocates the storage to the arrays used to store the symbolic processing information. The FORTRAN program calls a PL1 program which manipulates NASTRAN data to generate the compressed matrices. The resulting solution is then passed back to NASTRAN.

DMAP MODIFICATION

The NASTRAN program has a programming language of its own called DMAP which stands for Direct Matrix Abstraction Program. There is a DMAP compiler in NASTRAN and it is possible for a user to construct a DMAP procedure which will accomplish a desired set of operations independent of NASTRAN. However, the most common usage of DMAP is when a NASTRAN user executes a particular rigid format. The fourteen types of problems that can be solved using NASTRAN consist of exercising the many stored DMAP sets of instructions. These rigid formats were set up to save the user from remembering all the steps required to solve a certain problem type, but they also offer an opportunity to add, subtract, or otherwise modify a rigid format. The method used to accomplish such a change in NASTRAN is called an ALTER. To accomplish the solution of the problems selected to be run in Rigid Format 1 (linear steady state algorithms), an alter to the rigid format was constructed.

At statement 104, the normal NASTRAN procedure was altered to provide a punched file containing the stiffness matrix $[KLL]$ and load vector $\{PL\}$ for use in the SL-MATH routines. This was done by using the OUTPUT3 module. Statement 104 was selected so that the final solution set U_2 would be used. The U_2 represents the degree of freedom that remains after all constraints and partitioning operations have been completed.

The OUTPUT3 module punched out the $[KLL]$ matrix in the form of a matrix data block of DMI card images. A NASTRAN matrix in packed form has a U word header followed by integer points and non-zero matrix values mixed. The matrix itself had to be unpacked from the DMI form. To accomplish this, a set of three PL1 programs were written which would read the data block and prepare the row and column pointers as well as compress the matrix for SL-MATH as described earlier. PL1 was used because of its superior string handling capability and relative ease with which character information can be converted to either integer or floating point data. It should be recognized that this procedure was utilized on a temporary basis only.

To automate this procedure, the routines will be linked into NASTRAN appropriately so that the matrix and load vector can be passed directly without exiting NASTRAN.

RESULTS

The results in Table 1 represent the application of the procedure to four practical problems. Those problems are:

- 1) A one-dimensional rod problem with an extension force.
- 2) A two-dimensional plate with thermal loading.

- 3) A three-dimensional heat transfer problem.
- 4) A two-dimensional in-plane stress analysis problem.

In each of the four cases, the CPU time was less for the modified procedure than for NASTRAN. The CPU time represented in the table is only the time for actually solving the equation. The time required to process the geometry and assemble the matrices is not included in the tabulated results.

The solution time for problem three is a factor of ten times that for problem four. This inconsistency can only be attributable to the optimal ordering algorithm. The algorithm is not suited to the case where the structure has several layers, but is more suited to the widely dispersed numbering scheme for the two-dimensional problem. Other ordering algorithms are available in SL-MATH, but were not tried.

The NASTRAN runs were made in region sizes of 360K for problems 1 and 2, 600K for problems 3 and 4. The modified procedure ran in a region size of 1500K.

CONCLUSION

The procedure does seem to save time and, therefore, would be of help to users of NASTRAN level 15.5. It is not clear how the procedure will run against level 16.0. Additional work is needed to integrate the procedure into NASTRAN and to test the same problems on level 16.0. A comparison should also be made with USC NASTRAN.

REFERENCES

- 1) Buturla, E. M. and Cottrell, P. E. (1974), "Two-Dimensional Finite Element Analysis of Semiconductor Steady State Transport Equations," International Conference on Computation Methods in Nonlinear Mechanics, Austin, Texas.
- 2) Cottrell, P. E. and Buturla, E. M. (1975), "Steady State Analysis of Field Effect Transistors via the Finite Element Method," IEEE I.E.D.M., Washington, D. C.
- 3) Buturla, E. M. and Cottrell, P. E. (1976), "Two-Dimensional Finite Element Analysis of Semiconductor Transport Phenomena," International Conference on Numerical Methods in Electric and Magnetic Problems, St. Margherita, Italy.
- 4) Buturla, E. M. (1976), "Application of Symbolic Processing Techniques to a Finite Element Algorithm," 1976 IBM Symposium on Mathematics and Computation, Yorktown Heights, NY, RA 82 (#26690).
- 5) "IBM System/360 and System/370, IBM 1130 and 1800 Subroutine Library-Mathematics (SL-MATH)," IBM World Trade, IBM Germany, 700 Stuttgart, Germany.

RESULTS

PROBLEM	D.O.F.	<u>NASTRAN</u>		<u>SYM/NUM</u>		
		CPU	ELAP	CPU	ELAP	OVERHEAD
1	10	2.0	15.0	.13	2.6	.23
2	208	4.0	27.0	.50	5.0	1.60
3	777	39.0	121.0	17.50	44.0	38.50
4	918	49.0	209.0	1.80	4.3	8.10

TABLE 1: COMPARISON TABLE

ALTERNATE APPROACHES TO VIBRATION AND SHOCK ANALYSIS USING NASTRAN

Richard E. Denver and Joseph M. Menichello
IBM Federal Systems Division

ABSTRACT

In analyzing sinusoidal steady-state vibration of structures, the NASTRAN (NASA Structural Analysis) program provides Rigid Formats 8 and 11 for direct and modal frequency and random response solutions. The dynamic solution of the sinusoidal steady state vibration problem usually requires very large memory core allocations and very long computational time. Our experience of analyzing the Proteus system structures revealed an alternate solution set composed of modifying the eigenvector output of normal modes analysis, resulting in significant cost savings. This paper gives a step-by-step approach for the use of this alternate algorithm to the sinusoidal steady-state vibration problem.

The paper then deals with a method that derives an approximate equivalent static load to a base excitation shock analysis. The transient analysis in the current level of NASTRAN, level 16, does not directly provide for either input acceleration forcing functions or enforced boundary displacement. In the suggested alternate analysis format, equivalent force input functions are applied to the constrained locations by using the artifice of placing a large mass, with respect to the total system mass, at the desired acceleration input points. This shortcut static analysis approach is presented to approximate the expensive and time-consuming dynamics analysis approach to the base-excitation shock analysis.

Each of these methods has proved to be a reliable mechanical design guide and has correlated closely with empirical results.

INTRODUCTION

As the Environmental Design Analysis group at IBM's Federal Systems Division (FSD), Owego, NY, facility, our primary function is to guide the mechanical design of FSD hardware during concept and development phases. Vibration and shock requirements associated with these products are often quite severe, as illustrated in figures 1a, 1b, 1c, and 1d.

Structural analysis in support of the mechanical design of electronics equipment is necessary to ensure compliance with the dynamic environmental requirements. Initially, the harmonic and random vibration excitation formats, as well as direct and modal transient shock capability of NASTRAN, have been exercised to validate structural design. As the designs progress, many changes are incurred due to customer request, product improvement, compatibility with

fabrication techniques, and results of analysis, each requiring analytical evaluation. In general, these changes must be assessed quickly and accurately and must result in minimum effect on cost and schedule. Because of long turn-around time and long computation time associated with the rigid formats we have been using, we began a study for an alternate, more efficient algorithm for this solution phase.

The two methods resulting from this investigation for representing structural deformations and stresses under the influence of vibration and shock environments are discussed under the headings "Vibration Stress Analysis" and "Shock Stress Analysis."

VIBRATION STRESS ANALYSIS

The following steps give the structural analyst a simplified shortcut in determining maximum stresses in structural members during sinusoidal steady-state vibration at resonant frequency. This approach foregoes the direct or modal frequency and random response analysis, Rigid Format 8 or 11, respectively, which requires very large memory core allocations and very long computation time, and uses normal modes analysis, Rigid Format 3, and static analysis, Rigid Format 1, with an appropriate ALTER package given in the presentation:

- 1) First, run a normal modes analysis (Rigid Format 3) to determine fundamental structural resonances in three principal directions.
 - a) Specify MAX in the NORM field of the EIGR continuation card. This normalizes maximum structural displacement for use in conjunction with the ALTER package given below.
 - b) Use CHPNT YES in the Executive Control deck to checkpoint all the necessary data and to recover the data in restart. This step will result in significant savings when used with the ALTER package or to calculate additional vibratory modes.
- 2) To restart the checkpointed run to determine structural deformation and vibration stresses at resonance, the following items have to be specified:
 - a) Specify a maximum structural displacement at the structural resonance, (δ). This can be estimated from the following relationship for a single-degree-of-freedom system.

$$\delta = \frac{386.1 \times G \times Q}{4\pi^2 \times f^2}$$

where

δ = single amplitude of response of structure (inches)

G = input steady state sinusoidal peak acceleration

Q = transmissibility at resonance

f = structural natural frequency

In the above relationship, it is assumed that G is a known quantity, and f has been determined in the NASTRAN analysis using Rigid Format 3. The transmissibility at resonance may be determined from test data of a prototype or similar structure. If this data is not available, however, experience has shown that a range of $0.05 \leq C/Cr \leq 0.10$ is typical of three-dimensional frame structures of standard construction. Figure 2 illustrates the interrelationship between structural damping and amplification factor at resonance.

- b) Insert the following ALTER package* in the NASTRAN Executive Control deck, which scales all output, and which includes structural deformations and stresses. (For Level 16)

```
ALTER      108, 108
SDR1       USET,,PHIA,,GO,GM,,KFS,,/
           PHIGG,QG/C,N,1/C,N,REIG $
ADD        PHIGG,/PHIG/C,Y,ALPHA=(1.0.0.0)/
           C,Y,BETA = (0.0,0.0) $
ADD        QGG,/QG/C,Y,ALPHA=(1.0.0.0)/
           C,Y,BETA=)0.0,0.0) $
ENDALTER
```

- c) In conjunction with the ALTER package, the user would also have to specify the value of ALPHA on a PARAM card in the BULK DATA deck. This value of ALPHA corresponds to the maximum structural response, δ , calculated in the step 2a) of the procedure.

Example of this procedure follows:

Input vibration level = 2 g peak sinusoidal

Natural frequency of structure = 68 Hz

Transmissibility of structure = 10

$$\delta = \frac{386.1 \times G \times Q}{4\pi^2 \times f^2} = \frac{386.1 \times 2 \times 10}{4 \times (3.14)^2 \times (68)^2} = 0.042 \text{ inch}$$

Specifying in the BULK DATA deck

PARAM	ALPHA	0.042	0.0		
-------	-------	-------	-----	--	--

will result in linear scaling of all the output data blocks by a factor of 0.042. Obviously, because of the assumptions used, this method is restricted to linear analysis only.

Note: If the user were interested in a RESTART capability, DMAP statement 109 should also be altered to checkpoint data block PHIGG as well as PHIG and QG in the Rigid Format 1.

*The authors wish to express their gratitude to the staff of the NASTRAN Systems Management Office, NASA Langley Research Center, Hampton, Virginia, particularly to Mr. Joseph Walz for assistance offered in preparation of the ALTER packages presented in this paper.

SHOCK STRESS ANALYSIS

The NASTRAN program provides two rigid formats for analyzing shock excitation problems. The transient analysis capability in the current level of NASTRAN, level 16, however, does not directly provide for input acceleration-forcing functions or enforced boundary displacement. To overcome this deficiency in the program, modifications* had to be made to use the program for our specific application of base shock excitation of structures.

The following modifications were necessary to use Rigid Format 9 (Direct Transient):

- 1) Modify the math model as follows:
 - a) Place large seismic masses at all boundary points. These large seismic masses at the support points are necessary to reduce the effects of feedback from the structural responses.
 - b) Use multipoint constraint (MPC) equations so that all boundary points will move together, allowing motion only in the direction of the shock.
 - c) Calculate a forcing function, $f(t)$, which will produce the desired acceleration function on the overall mass, structural mass plus seismic mass, to satisfy the relationship $F=ma$ at different time intervals.
 - d) Input the forcing function, $f(t)$, at the boundary points using the DAREA card in the BULK DATA deck.
- 2) Could start Rigid Format 9.

The following modifications were necessary to use the Rigid Format 12 (Transient Modal Analysis):

- 1) Run a normal modes analysis, Rigid Format 3, (we suggest using the Inverse Power method) to determine the fundamental structural resonance in each principal direction along which shock pulse is to be applied. Be sure to checkpoint this run. In this run, seismic masses should not be attached to the boundary points in the math model.
- 2) Modify the math model as follows:
 - a) Place large seismic masses at all boundary points; approximately 10^5 times the structural mass.

*The authors would like to thank Mr. Leon H. Arnold, IBM Owego, and Dr. Han Chung Wang, IBM Endicott, for their technical assistance in developing the alternate solution methods and implementing them in the NASTRAN program.

- b) Use MPC equations for all the boundary points to allow the boundary points to move only in the direction of the shock.
 - c) Calculate a forcing function in the same manner as in using Rigid Format 9.
 - d) Input the forcing function at the boundary points; these are structural attachment points either at the base or bulkhead of aircraft, shipboard, submarine, or other test fixture frames.
- 3) Restart Rigid Format 12 using the structural resonance data obtained in Rigid Format 3,
- a) Bound the resonance of interest by using the PARAM LFREQ and PARAM HFREQ cards in the BULK DATA deck; the smaller the bandwidth, the more rapid the computation time. To minimize convergence time, specify the same frequency range on the EIGR card as specified on the preceding PARAM LFREQ and PARAM HFREQ cards.

NOTE: In all rigid formats, it is imperative to use a pre-processor program, such as BANDAID or BANDIT to reduce the semi-bandwidth of structural matrices.

The excessive time associated with model preparation and solution to shock analyses using NASTRAN Rigid Formats 9 and 12 reduced their usefulness for the specific application of mechanical design guidance in the earliest phases of design.

The following method has been developed which approximates the effect of the shock pulse on the system through the use of a single-degree-of-freedom idealization, assuming that the fundamental mode parallel to the direction of the pulse is the sole contributor to maximum deformation and corresponding stresses. This method utilizes a combination of Rigid Formats 3 and 1 in place of Rigid Format 9 or 12. Its implementation is described in the following lists:

- 1) Perform a natural frequency analysis, Rigid Format 3, using the Inverse Power method to determine the fundamental structural resonance in each plane in which a shock pulse is to be applied. The use of Inverse Power eigenvalue extraction will prove most efficient since only a few of the resonant modes are of interest.

Note: One can force the program to search for resonant modes in numerically ascending order by specifying ND = small, NE = large (~100xND) in the EIGR card. This will reduce the computation time required to find the lowest natural frequency. It will significantly reduce the possibility of terminating via termination codes 6 or 7 without determining the fundamental mode.

- 2) Calculate the dimensionless parameter t_0/T in which

t_1 = pulse period (seconds) of the shock to be applied

T = fundamental period of responding structure = $2\pi/\omega$ (seconds)

- 3) Using the plotted data in figure 3, extracted in part from reference 4, determine \ddot{x}/A (acceleration amplitude magnification factor)

where

A = peak acceleration of input pulse (g)

\ddot{x} = maximum response acceleration of structure (g)

As shown in figure 3, the solution to the equation of motion and its derivatives for the single-degree-of-freedom idealization are dependent on the degree of viscous damping exhibited by the responding system. This value may be determined from test data by examining the transmissibility (Q) at resonance and correlating Q with the damping factor (C/C_r) through the use of graphs specified in reference 2, repeated in figure 2 for illustration. If test data is not available, experience has shown that a value of damping of $0.05 \leq C/C_r \leq 0.10$ is typical of most three-dimensional framed structures of standard construction.

- 4) Could start in Rigid Format 1 specifying a loading, G , through the use of the NASTRAN GRAV card in which

$$G = A \times \frac{\ddot{x}}{A}$$

where

A = peak acceleration of input shock pulse, g

$\frac{\ddot{x}}{A}$ = dynamic load factor determined in step 3

CONCLUSIONS

The two methods discussed previously have proved to be a valuable engineering aid during the developmental phase of mechanical designs. They have proved to be more cost- and time-effective than their counterparts in NASTRAN Rigid Formats 9, 10, 11, and 12. Tables 1 and 2 summarize the time and cost comparisons of the various solutions. In addition, the solutions have proved to yield acceptable accuracy as evidenced in tables 3 and 4.

REFERENCES

1. Dr. Richard Eppink: A Comparison of Direct Integration vs Modal Methods of Analysis for Transient Response Analysis in NASTRAN. University of Virginia.
2. Application Selection Guide. Barry division of Barry Wright Corporation.
3. C.M. Harris and C.E. Crede: Shock and Vibration Handbook. New York: McGraw Hill, 1961, pp 23-1 through 23-37.
4. Robert R. Luke: The Impact Response of Single-Degree-of-Freedom System with Viscous Damping. University of Texas, Structural Mechanics Research Laboratory, Austin, Texas.
5. NASTRAN User's Manual, NASA SP-222(03). Scientific and Technical Information Office, National Aeronautics and Space Administration, March, 1976.
6. NASTRAN Theoretical Manual, NASA SP-221-(03). Scientific and Technical Information Office, National Aeronautics and Space Administration, March, 1976.
7. NASTRAN Programmers Manual, NASA SP-223(03). Scientific and Technical Information Office, National Aeronautics and Space Administration, March, 1976.
8. Dr. Gordon C. Everstine: The BANDIT Computer Program for the Reduction of Matrix Bandwidth for NASTRAN. Computation and Mathematics Department, Naval Ship Research and Development Center, Bethesda, MD 20034, March, 1972, Report No. 3827.

TABLE 1 - TIME AND COST COMPARISONS OF ALTERNATE SOLUTION METHODS FOR OBTAINING MAXIMUM VIBRATION DISPLACEMENTS AND STRESSES

Unit analyzed	Math model			Solution Method	RF	Unit \$	Total \$	Norma- lized \$	CPU Time (s)	EXCP (I/O operations)
	Grid points	Finite elements	DOF							
Proteus* Analyzer Unit	44	85	228	Modal/freq. random response	3	167.32			480	28,816
					11 (1 subcase)	188.68	356.00	1.0	328	19,128
					11 (3 subcases)	566.00	733.32	1.0	980	57,000
				Alternate method	3	167.32			480	28,816
					1 (1 subcase)	28.48	195.80	0.55	24	8,755
					1 (3 subcases)	85.44	257.76	0.34	72	25,000
Proteus** STM/TACTAS Electronics Rack	165	750	942	Alternate Method	3	242.71			624	32,662
					1 (1 subcase)	23.04	265.75		31	44,760

Notes:

*All solutions performed on IBM System/360 Model 85; NASTRAN level 15.0.1, with BANDIT preprocessing

**NASTRAN Level 15.5.2, IBM System/370-158.

DOF - degree of freedom

RF - rigid format

CPU - central processing unit

EXCP - Execute Channel program

TABLE 2. - TIME AND COST COMPARISONS OF ALTERNATE SOLUTION METHODS FOR OBTAINING MAXIMUM SHOCK DISPLACEMENTS AND STRESSES

Unit analyzed	Grid points	Finite elements	DOF	Solution method	RF	Unit \$	Total \$	Normalized \$	CPU time(s)	EXCP (I/O operations)	Core size (K bytes)
Proteus Anaylzer Unit	45	85	228	Direct transient (3 subcases)	9	267.00	267.00	0.41	288	69,000	500
				Modal transient (3 subcases)	3 12	167.00 480.00	647.00	1.0	480 1251	28,816 91,000	350
				Alternate solution method	3 1	167.00 54.00	221.00	0.34	480 78	28,816 16,500	300
Proteus STM/TACTAS	165	750	942	Direct transient (3 subcases)	9	531.00	531.00	1.0	1017	103,000	750
				Alternate solution method (3 subcases)	3 1	242.00 46.00	288.00	0.54	624 78	32,662 16,600	750 350

TABLE 3. - RESULTS COMPARISON OF ALTERNATE SOLUTION METHODS FOR OBTAINING MAXIMUM VIBRATION DISPLACEMENTS AND STRESSES

Unit analyzed	Solution method	Direction	Fundamental natural frequency (Hz)	Maximum peak axial stress (lbf/in ²)
Proteus Analyzer Unit	Modal frequency & random response	X	66	10,400**
		Y	188	NA
		Z	406	NA
	Alternate solution method	X	66	13,710
		Y	188	-
		Z	406	-
	Engineering development test	X	72*	NA
		Y	190*	NA
		Z	420*	NA

Notes:

*Variation in test/analytical results attributed to weight discrepancy (50 lb) and modified structural configuration
 **Based on 20% damping; 5% damping yielded 40,100 lbf/in² maximum stress

TABLE 4. - RESULTS COMPARISON OF ALTERNATE SOLUTION METHODS FOR OBTAINING MAXIMUM SHOCK DISPLACEMENTS AND STRESSES

Unit analyzed	Eigenvalue extraction			Shock stress analysis	
	Direction	Natural frequency (Hz)	Loading condition	Solution method	Maximum peak axial stress (lbf/in ²)
Proteus STM/TACTAS Electronics Rack	-	-	Navy high impact Medium Weight, inclined, hammer-drop shock test	Direct transient analysis	31,770
	-	-			
	-	-			
	X	124	Alternate solution method	28,905	
	Y	70, 125, 146			
	Z	Not found			
	X	122	Engineering development testing	29,000	
	Y	70, 124, 146			
	Z	Not found			

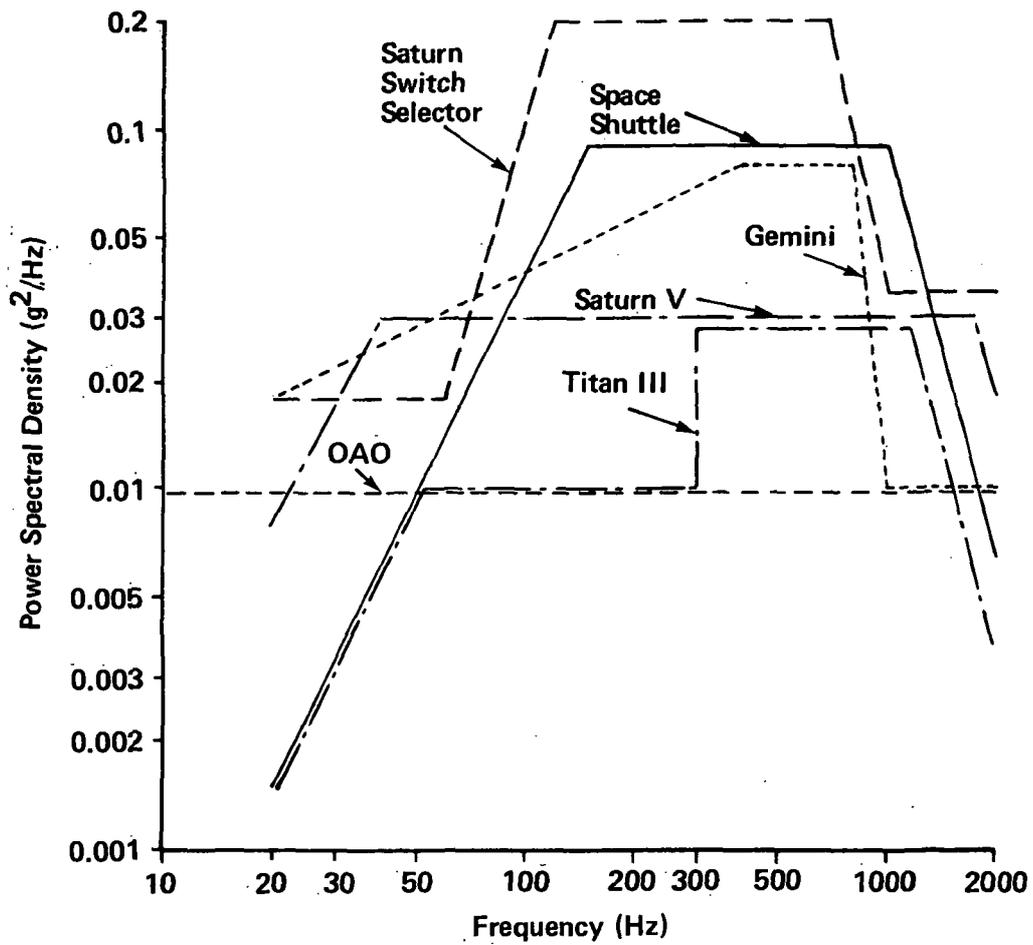


Figure 1a. Typical Spacecraft Random Vibration Levels

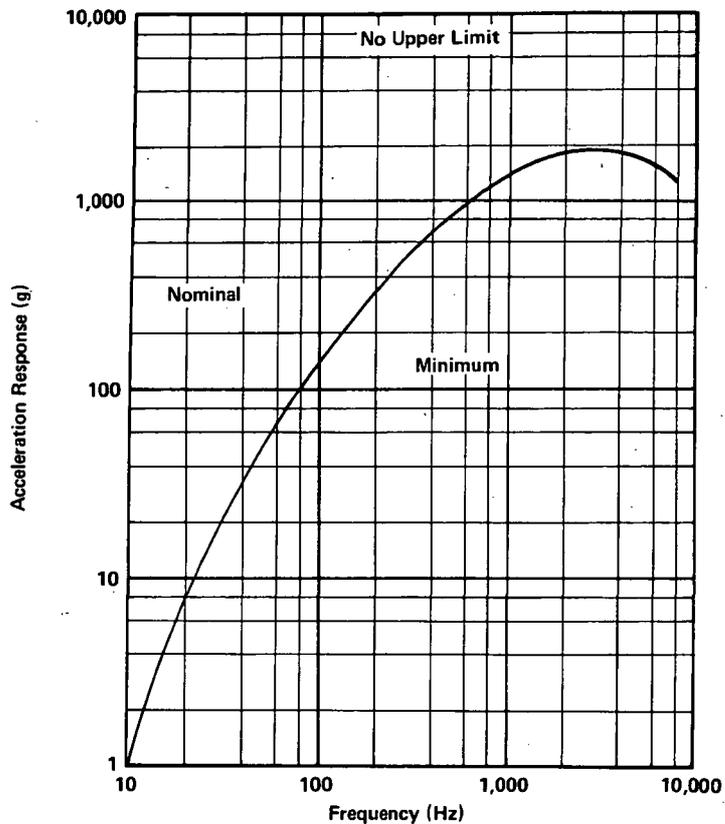


Figure 1b. Shock Requirements, Pyrotechnic Shock Spectrum

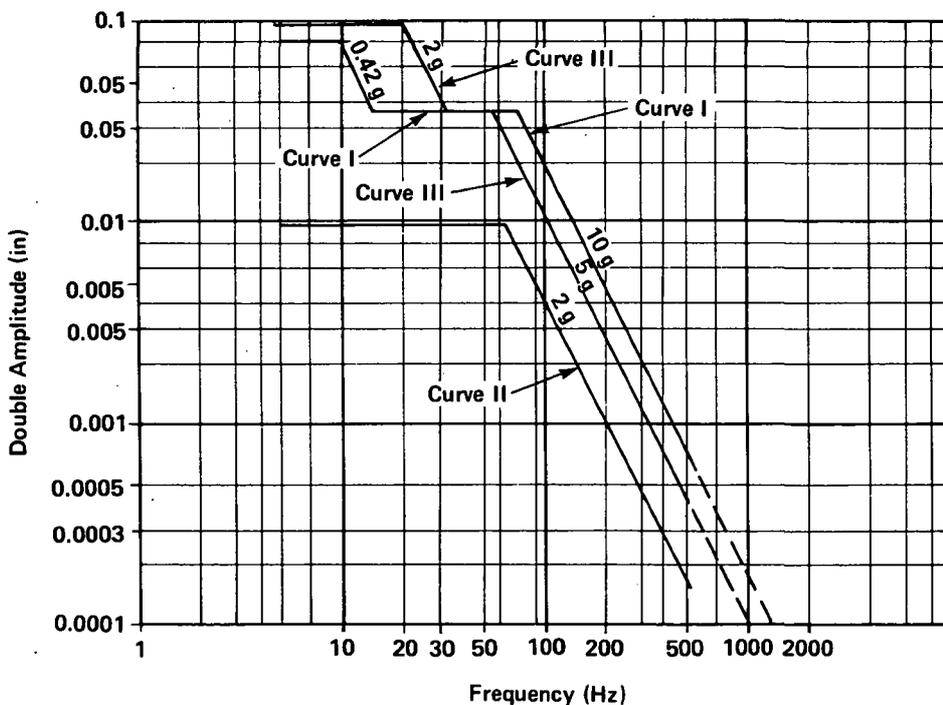


Figure 1c. Typical Aircraft Sinusoidal Vibration Requirements

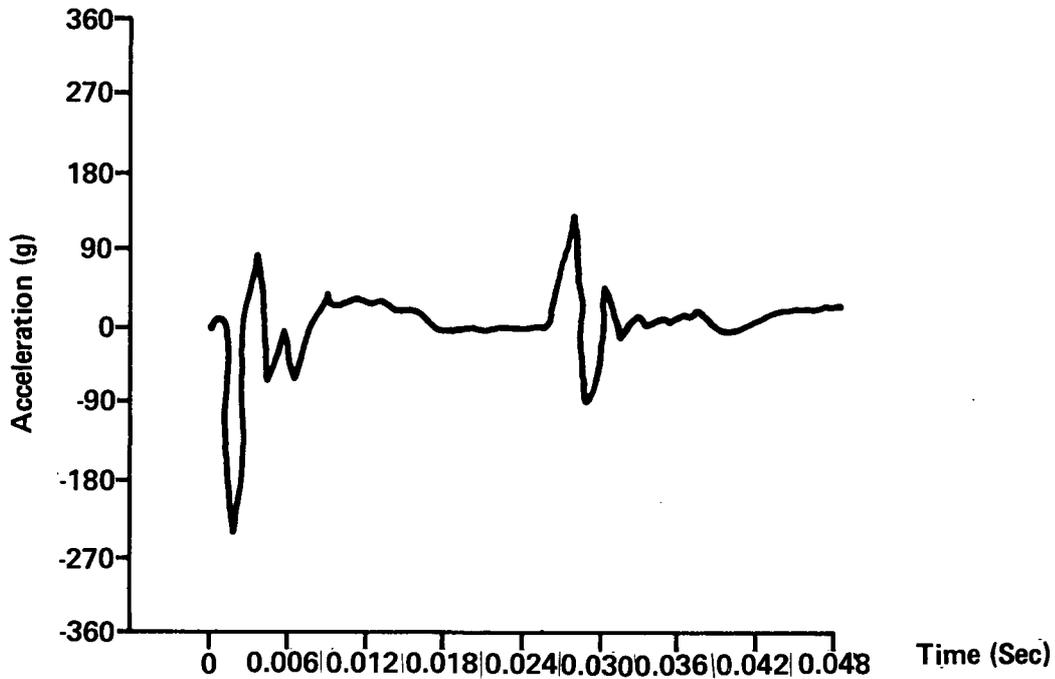


Figure 1d. Typical Navy Medium-Weight-Hammer-Drop Anvil Response

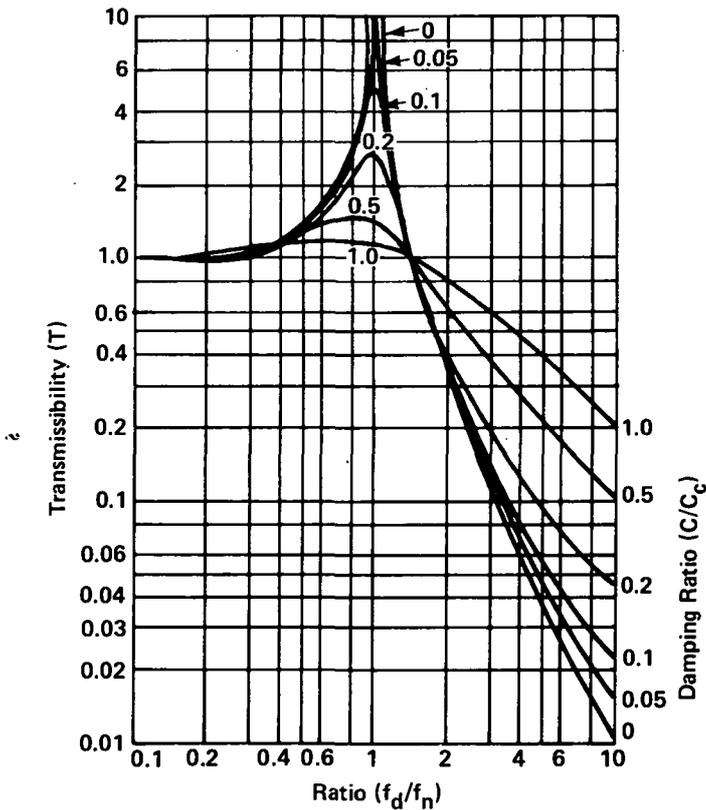


Figure 2. Family of Transmissibility Curves for a Single-Degree-of-Freedom System

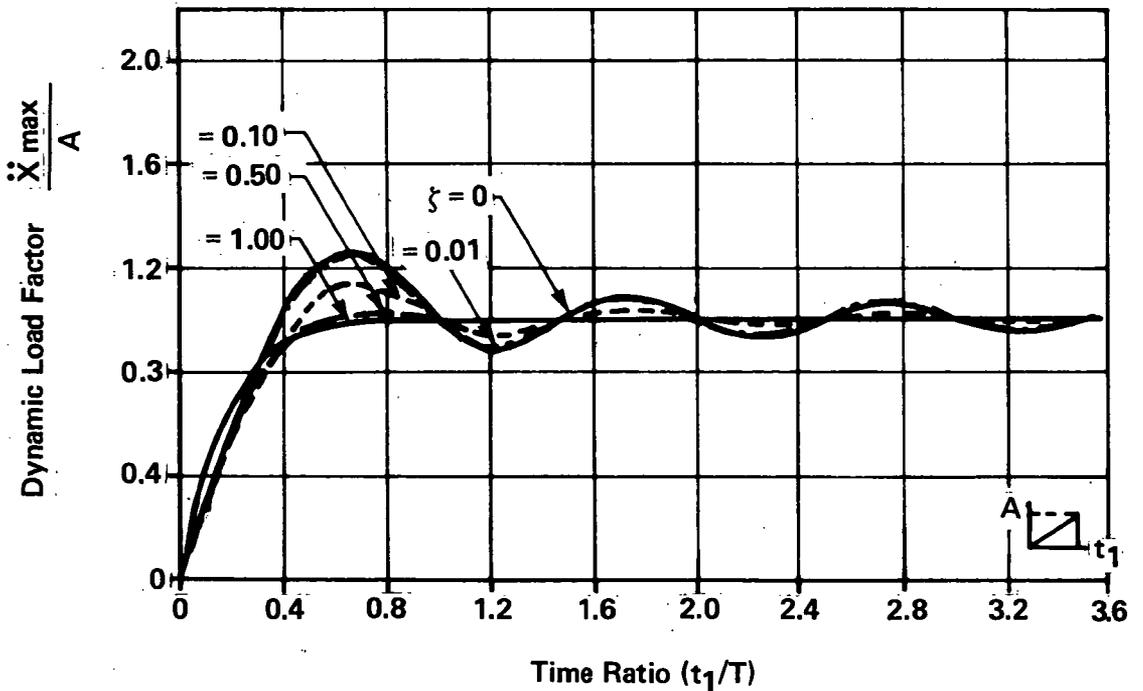
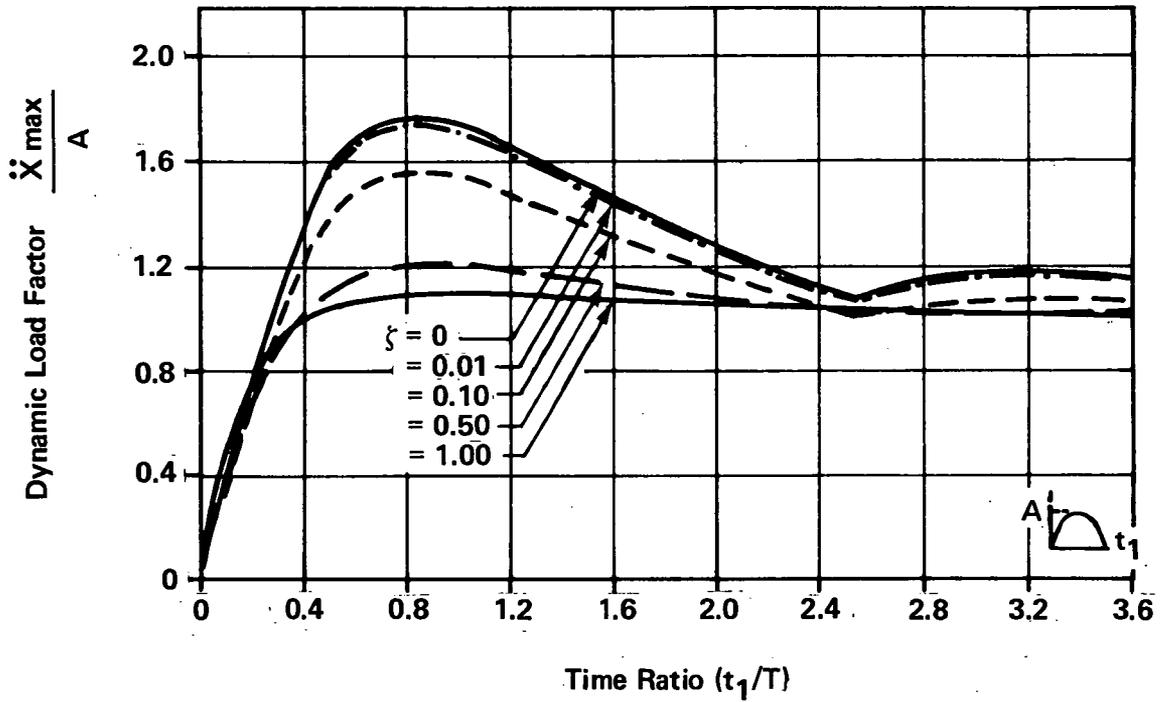


Figure 3. Shock Transmissibility Curves for a Single-Degree-of-Freedom System with Directly Coupled Viscous Damping Subjected to Sinusoidal and Terminal-Peak Sawtooth Shock Pulses

NASTRAN USE FOR CYCLIC RESPONSE AND FATIGUE

ANALYSIS OF WIND TURBINE TOWERS*

C. C. Chamis, P. Manos, J. H. Sinclair, and J. R. Winemiller
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

SUMMARY

A procedure is described which uses NASTRAN coupled with fatigue criteria via a postprocessor to determine the cyclic response and to assess the fatigue resistance (fatigue life) of wind turbine generator towers. The cyclic loads to which the tower may be subjected are entered either in a quasi-static approach through static load subcases (Rigid Format 1) or through the direct dynamic response (Rigid Format 9) features of NASTRAN. The fatigue criteria are applied to NASTRAN output data from either rigid format through an externally written user program embedded in a postprocessor.

INTRODUCTION

One convenient and economical wind turbine generator (WTG) tower configuration is the welded tubular space truss. The fatigue sensitivity of these types of structures and their fatigue resistance over the design life of the WTG needs to be determined. One possible procedure is to determine the cyclic response and fatigue resistance by the use of NASTRAN coupled with user-written programs for applying the appropriate fatigue criteria via a postprocessor. This paper describes the development of such a procedure. A sequel paper will describe the results obtained from using such a procedure and their possible significance to WTG tower design with respect to dynamic response and fatigue resistance.

The procedure consists of modeling the tower using rod and bar finite elements, determining the tower frequencies free of loads and in the presence of force fields, applying the anticipated cyclic loads using either the quasi-static approach or the direct dynamic response and the use of convenient user programs embedded in a postprocessor to apply the fatigue criteria. The details of the procedure are described as it was structured and used to assess the fatigue resistance of the MOD 0A WTG, which is of immediate interest to the ERDA-NASA Wind Power Project.

* Work performed for Energy Research and Development Administration, Division of Solar Energy, under Interagency Agreement E(49-26)-1004.

PROBLEM DEFINITION AND SOLUTION APPROACHES

The problem definition consists of establishing the cyclic load conditions to insure infinite life of the tower. Usually these cyclic load conditions result in a large number of cycles which fall in the category of high-cycle-fatigue for purposes of assessing the fatigue resistance of the tower.

The cyclic loads used herein were determined using MOSTAB (ref. 1) and were provided by the Wind Power Office at Lewis.

The use of NASTRAN in the solution consists of two approaches. Approach one consists of expressing the cyclic load conditions as static load subcases using Rigid Format (RF) 1. Each subcase corresponds to 15° increments (azimuthal positions) of the WTG rotor. The next step in this approach is to determine the stress ranges between the various subcases and compare these ranges to the fatigue allowable. Approach two consists of expressing the cyclic load conditions as forcing functions. The forcing functions are then used in the direct dynamic capability of NASTRAN via Rigid Format (RF) 9. The second approach is the more representative approach in that both the tower inertia effects and damping can be included in the analysis. Detailed descriptions of both approaches are given in later sections.

TOWER DESCRIPTION

The MOD 0A WTG tower is a 4-legged, 93-foot-high lattice structure tapering from 30 feet square at its base to about 7 feet square at the top. This tower is similar to that of MOD 0 100-kW turbine shown in figure 1. The tower supports a two-bladed rotor measuring 125 feet tip to tip, a generator, and the drive trains of a 200-kW wind turbine generator (WTG) designed by NASA for the Energy Research and Development Agency. Total supported weight is approximately 45 000 pounds at 50 inches eccentricity. The tower itself weighs about 46 000 pounds. The rotor is positioned downwind of the tower. Such a location dictated a clean, open tower to minimize shadow effects. Consequently, wherever possible within the swept area of the blades, tubular steel members directly welded to one another with full-penetration groove welds were used. Member sizes varied from 8-inch-diameter pipe for the legs to 3-inch-diameter pipe for low-stressed bracing. Below the swept area, rolled structural steel shapes with bolted joints and tubular members fillet-welded to gussets were used to aid field erection procedures.

Design of the tower was predicated on the following:

(a) Avoid resonant frequencies.

(b) Withstand a 120-mph (at 30 ft above grade) hurricane wind with the rotor in a stowed position.

(c) Safely withstand the cyclic operating loads for infinite life.

All frequency computations and stress analyses were performed using NASTRAN. Maximum computed stresses were limited to the allowable values of the AISC

Specification for the Design, Fabrication and Erection of Structural Steel for Buildings. No increased allowable stress was permitted for wind included in the loading combination.

The tower was modeled for NASTRAN using bar (CBAR) and rod (CROD) elements. End fixity of bars was not relieved. Ninety rod elements and 144 bar elements were required to represent the structure. A computer plot of the NASTRAN model is shown in figure 2.

FATIGUE CRITERIA

To insure infinite life, the range of stress produced in each tower member by the operating loads was computed and compared with the allowable fatigue stress ranges listed in appendix B of the AISC Code (ref. 2).

Rated operating speed of the rotor is 40 rpm. Rotor-induced forces acting on the tower complete a cycle with each half revolution of the rotor. Therefore, less than 420 hours of rated operating time are required to generate 2 million loading cycles. It is necessary, then, to limit stress ranges to those values allowed for Loading Condition 4 (more than 2×10^6 cycles) of the AISC Code.

Seven allowable stress categories are listed under Loading Condition 4, with the admissible range of stress varying from 24 000 psi for Category A to 6000 psi for Category E. Selection of a particular category as the limiting design criteria is dependent upon the member type and fabrication details. For those tower elements with fillet-welded end connections, the minimum stress range of 6000 psi is directly applicable. Most members, however, are not fillet welded and would be permitted a stress range greater than 6000 psi. To examine each tower element and connection detail for a differing stress range would be a cumbersome and a laborious chore. Therefore, the minimum allowable stress range of 6000 psi was initially assumed applicable to all 234 elements of the tower model. The computer was instructed to output only those members whose computed range of stress exceeded 6000 psi. Those members so listed can then be examined more closely by the user himself for conformance with the Code, and if necessary, modified to meet Code requirements.

LOADING CONDITIONS

The loads used in this procedure for the quasi-static and dynamic approaches represent a relatively severe tower loading condition. This loading condition is expected to occur during operation of the MOD OA wind turbine during operation near or at the maximum allowable wind velocity of 40 mph.

The loads used for the NASTRAN static load subcases are shown in table I. As can be seen in table I each subcase corresponds to one azimuthal position. (15° increments of rotor angular position). Both forces and moments in table I were distributed equally to the eight grid points (nodes) at the top of the tower (9301 to 9308) shown in figure 2. The forces were entered by the NASTRAN FORCE Cards and the moments by the MOMENT CARDS. The tower weight was entered

using the GRAV card. The combinations of forces, moments, and gravity for each subcase were selected using the LOAD cards.

The loads used for the direct dynamic response are the same as those for the static subcases. In this case the cyclic loads were continued for two full cycles of the rotor. Each azimuthal position was expressed in terms of time. The cyclic load azimuthal positions are given at 15° intervals. This corresponds to 1/16 second for a rotor speed of 40 rpm. The data used are tabulated in table II. The data in table II were entered in NASTRAN using TABLED1. These loads were distributed equally at the four corners of the tower using DAREA and TLOAD1 cards. A graphical representation of the cyclic loads is shown in figure 3 for forces and figure 4 for moments.

In order to account for tower dead loads in the tower at zero time, the weight of the tower was distributed at the elevation sections as shown in figure 2 which have horizontal members. For the MOD 0A tower these are at elevations 21, 38, 54, 68, and 81 feet, corresponding to grids 2101, 3801, 5401, 6801, and 8101, respectively (fig. 2). The gravity loads at these points were established by using single point constraints for the vertical displacement (Y fig. 2) at the corresponding corner. The results are summarized in table III. These loads were entered using TABLED1 cards assuming that they were constant in the time interval. The gravity loads were distributed at their respective corners using DAREA cards.

The combination of the cyclic loads with gravity loads was selected using the DLOAD card. PARAM cards were used to put in uniform structural damping. A uniform damping of 0.01 was used. This value was the lowest of the values determined by field testing the MOD 0 tower (ref. 3) and it is on the conservative side.

NASTRAN POSTPROCESSOR

A fairly general purpose postprocessor routine has been written which scans any deck of cards that are formatted like NASTRAN card output. This routine prepares a "table of contents," writes informative header records, reformats the data record, and writes both these record types into a file provided by the user.

For the MOD 0A WTG tower NASTRAN prepared the basic data (stresses for selected elements, members in the tower) for output on cards. The EXEC system then diverted this card output to a disk file and the postprocessor got its input from that file. The tower was processed by both the NASTRAN static and transient analysis subsystems, Rigid Formats 1 and 9. In the user program for the static analysis (fig. 5) the output was arrayed so that, for each element, the values of selected stress types could be scanned over the various subcases. In the user program for the transient analysis (fig. 9) the output was arrayed so that the scanning would be over the various time steps. These postprocessor routines are coupled with user programs which apply the fatigue criteria.

The equations used in the user program for applying the fatigue criteria are as follows:

Static approach. - Two sets of equations were used. The first set operates on the maximum stresses (axial combined with bending) at each end of the member (ends A and B). The equations for this case are

$$\left. \begin{aligned} \left| \sigma_{\max,i,k}^{(A)} - \sigma_{\max,i,l}^{(A)} \right| &< \sigma_{FL} \\ \left| \sigma_{\max,i,k}^{(B)} - \sigma_{\max,i,l}^{(B)} \right| &< \sigma_{FL} \end{aligned} \right\} \quad (1)$$

where i scans the elements (members), and k and l are combinations of loading conditions (subcases) as follows: $k = l$ (1) 11 and $l = k + 1$ (1) 12. The superscripts (A) and (B) refer to rod or bar element ends A and B, respectively; σ_{FL} is the fatigue limit, which was taken to be 6000 psi as described earlier.

The second set of equations operates on the sum of the bending stresses and axial stress at the four different points at each end of the element. The governing equations are

$$\left. \begin{aligned} \left| \left(\sigma_{b,m}^{(A)} + \sigma_{a,m}^{(A)} \right)_{i,k} - \left(\sigma_{b,m}^{(A)} + \sigma_{a,m}^{(A)} \right)_{i,l} \right| &< \sigma_{FL} \\ \left| \left(\sigma_{b,m}^{(B)} + \sigma_{a,m}^{(B)} \right)_{i,k} - \left(\sigma_{b,m}^{(B)} + \sigma_{a,m}^{(A)} \right)_{i,l} \right| &< \sigma_{FL} \end{aligned} \right\} \quad (2)$$

The notation in equation (2) is as follows: superscripts (A) and (B) denote element ends A and B, respectively; subscripts a and b denote axial and bending, respectively; subscript m denotes the m point at end A; subscripts i, k, and l have the same meaning as in equation (1); and σ_{FL} is the stress fatigue limit which equals 6000 psi. Equations (2) were necessary to assure that the fatigue limit was not violated by the combination of bending and axial stresses at the different points at each element end.

Dynamic approach. - The equations used in the user program for applying the fatigue criteria to the dynamic approach are

$$\left. \begin{aligned} \left| \left(\sigma_{b,m}^{(A)} + \sigma_{a,m}^{(A)} \right)_{i,t_0} - \left(\sigma_{b,m}^{(A)} + \sigma_{a,m}^{(A)} \right)_{i,t} \right| &< \sigma_{FL} \\ \left| \left(\sigma_{b,m}^{(B)} + \sigma_{a,m}^{(B)} \right)_{i,t_0} - \left(\sigma_{b,m}^{(B)} + \sigma_{a,m}^{(B)} \right)_{i,t} \right| &< \sigma_{FL} \end{aligned} \right\} \quad t_0 \leq t \leq t_f \quad (3)$$

where equations (3) are the same as equations (2), the only difference being that the combinations are compared in the time interval from some starting time

t_0 to a final time t_f . The initial and final times are taken to be an interval representing steady-state response to avoid the high transient occurring at initial time. For the problem considered in this report t_0 is 1.5 seconds and t_f is 3.00 seconds.

RESULTS AND DISCUSSION

The procedure described herein was used to obtain the following results: tower vibration modes, quasi-static fatigue stress ranges, and dynamic response fatigue stress ranges.

Vibration Modes

Vibration modes were determined for the MOD OA tower for two general cases. In the first case, the tower was free of loads. In the second case, the tower was subjected to force fields corresponding to three rotor positions: blades vertical, blades at 45 degrees, and blades horizontal. The vibration modes for the free load case were determined using Rigid Format (RF) 3; those with force fields were determined using RF 13. The results for the frequencies for both cases are tabulated in table IV. The following may be observed from table IV:

1. The first vibration mode frequency of the tower is not affected by the force fields.
2. Force fields produce additional modes with frequencies lying between those of the free load case.
3. The vibration frequencies from the three different force fields associated with the three different rotor positions are approximately the same.

One consequence of observation (2) is that frequency and rotor-speed interference (Campbell) diagrams should be based on tower frequencies calculated with force fields in order to avoid operating the rotor at, or near, resonant frequencies.

Quasi-static Analysis Results (Approach One)

Computer-plotted deformed positions of the tower are shown in figure 7. Tower views "A" and "B" are shown in this figure, with views taken as depicted in the schematic. The deformed tower positions shown in figure 7 correspond to the rotor position with the blades vertical. As can be seen in this figure, the tower mean deformation is considerably more with the wind direction, as one might expect, than it is at 90° to the wind direction.

Corner displacements at the top of the tower are plotted against azimuthal angle in figure 8. As can be seen in this figure, the x-component of the displacement oscillates with a greater amplitude range (0.12 in.) than the z-component (0.03 in.). The y-displacement component (vertical) is the smallest

of the three (0.03 in.) and shows practically no oscillation. Both the x and z displacement components appear to oscillate with a frequency of about 2 cycles per second (1 cycle/120°, or 240°/sec). This frequency corresponds to 3 cycles per rotor revolution, which is one more than the two-per-revolution impulse induced by the blades as they pass through the tower shadow.

The minimum and maximum stresses in the rod members were -1610 and 1600 psi, respectively. The minimum and maximum combined axial and bending stresses in the bar members were -8740 and 8780 psi, respectively. The maximum axial stress in all bar members was less than 2760 psi. Note the minimum and maximum stresses did not occur in the same member. Note also that these stresses are less than 50 percent of the static allowable stress, which is about 20 000 psi.

The stress ranges for the fatigue criteria as determined by the user program were less than the fatigue stress allowable of 6000 psi. The stress ranges in the majority of the members of the tower were less than 3000 psi. The stress ranges were greater than 3000 psi only in the horizontal members at the top of the tower and were mainly due to bending stresses near the end connections.

A significant conclusion from the above is that for the loads considered the stress ranges in the MOD OA tower do not violate the fatigue criteria for infinite life. Two major results are (1) bending stresses near the end connections dominate the stress field; (2) displacements along the wind direction are about twice as large as those transverse to the wind direction and about 10 times larger than the vertical displacements. It is to be noted, however, that the quasi-static approach yields results that may not have practical significance. They are shown herein only for purposes of illustrating the use of NASTRAN for this approach.

Dynamic Analysis Results (Approach Two)

Computed dynamic displacements at one corner at the top of the tower are shown graphically against time and azimuthal angle in figure 9. Note that the displacements are plotted from 1.5 to 3.0 seconds, which represents the second rotor revolution. This was to show displacements representing steady state or close to it and avoid the transients which occur initially.

The interesting points to be observed from figure 9 are

1. The z-component of the displacement (along the wind direction) oscillates about the 0.5-inch mean and remains positive (tends to pitch the tower) throughout the rotor revolution. It reaches a maximum amplitude of about 1.0 inch and a minimum of about 0.1 inch. Its range is about 0.9 inch.
2. The x-component of the displacement (transverse to the wind direction) oscillates about 0-inch mean. It reaches a maximum amplitude of 0.5 inch and a minimum of about -0.7 inch. Its range is about 1.2 inches.
3. The y-component of the displacement (vertical) oscillates about the -0.1-inch mean with a maximum amplitude of about -0.05 inch, a minimum of about -0.15 inch and with a maximum range of about 0.2 inch. The oscillations remain

negative throughout the rotor revolution.

4. The x-component displacement has the greatest amplitude range, 1.2 inches, followed by the z-component, 0.9 inch. The y-component amplitude range may be considered negligible when compared to the other two.

5. All displacement components oscillate with a frequency of about 5.3 cycles per second. It will be recalled from table IV (Vibration Frequencies Summary) that this frequency does not coincide with any of the free frequencies of the tower.

6. Referring to cyclic force (fig. 3) and cyclic moment curves (fig. 4), it can be observed that the z-component displacement is approximately in phase with the thrust, and the x-component with the roll moment.

The stresses calculated using the dynamic approach were relatively small. They ranged from -7050 to 6420 psi. None of these stress ranges violates the fatigue criteria as scanned by the user program described previously.

Comparison of Quasi-Static and Dynamic Analysis Approaches Results

Comparison of the results obtained from the two approaches is of interest to the analyst. Comparing corresponding displacement results from the quasi-static analysis (fig. 8) and the dynamic analysis (fig. 9) and also referring to figures 3 and 4, the following points are observed:

1. The displacements obtained from the quasi-static approach correspond approximately to the minimum displacement amplitudes of the dynamic approach.

2. The maximum amplitude range for the dynamic approach is about 30 times that of the quasi-static (0.90 to 0.03 in.) for the z-component and about 10 times for the x-component (1.20 to 0.12 in.)

3. The displacement x- and z-components obtained from the quasi-static approach appear to oscillate with a frequency of 2 cycles per second; those of the dynamic approach oscillate with a frequency of 5.3 cycles per second.

4. The y-component displacement from the quasi-static approach shows no oscillation (fig. 8); that from the dynamic shows oscillations with a frequency of 5.3 cycles per second and a maximum amplitude range of 0.2 inch.

5. The z-component displacement from the dynamic approach is in phase with the thrust (fig. 3), and the x-component is in phase with the roll moment (fig. 4). The corresponding displacements from the quasi-static approach are not in phase with any of the force or moment components or even the two-per-revolution impulse induced by the rotor.

Stresses from the quasi-static approach and from the dynamic approach with "no" damping and with 1 percent damping as a function of azimuthal angle are shown in figure 10 for comparison purposes. The stresses shown in this figure are maximum combined bending and axial and are for a horizontal member at the

top of the tower for both cases.

The interesting points to be noted from figure 10 are

1. Both quasi-static and dynamic approaches give stresses which exhibit the same general profile.
2. One-percent damping has a negligible effect on the stress determined from the dynamic approach compared to that with "no" damping.
3. The stress determined from the quasi-static approach oscillates about a mean of 6000 psi and has a range of about 5000 psi; that from the dynamic approach oscillates about a mean of 4300 psi and has a range of 3000 psi.
4. Referring to figure 4, both the stresses from both approaches appear to be in phase with the pitch moment.
5. Referring to figures 8 and 9, the stresses calculated by both approaches are not in phase with the displacements calculated from either approach.

The most significant aspects of the above comparisons are (1) the displacements from the dynamic approach are considerably higher than those from the quasi-static in all aspects: mean, amplitude, amplitude range, and frequency; (2) the stresses from the quasi-static approach are about twice as high as the corresponding one from the dynamic approach in mean value, amplitude, and amplitude range; (3) both stresses have the same profile and are in phase with the pitch moment; and (4) 1-percent damping has negligible effect on the stress profile calculated from the dynamic approach.

The above leads to the following important conclusions: (1) In general, a dynamic analysis would be required to realistically assess the fatigue resistance of WTG towers. (2) It may not be known a priori which of the results obtained from the quasi-static analysis are on the conservative side as compared with the dynamic analysis results.

GENERAL COMMENTS

As a consequence of this study, several general comments are in order. Stress analysis to assess the fatigue resistance of a structure can be carried out using NASTRAN once the cyclic load spectrum (profile) is known. The most direct way to do this is to apply the fatigue criteria through a user program outside NASTRAN where the user program operates on NASTRAN output data. Though this may require two independent runs (one for NASTRAN and one for the user program) and more computer time on a per run basis, it would be more cost effective in the long run than embedding the fatigue criteria within NASTRAN because one is not faced with the complexities of the NASTRAN structure. Going through the embedding route, it would normally require considerable debugging time to make the program operational. In the examples investigated herein, the total computer time was 11 minutes (3.3 min CPU) for the quasi-static case and 42 minutes (11.7 min CPU) for the dynamic case in the UNIVAC 1110.

A logical extension in the use of NASTRAN for the fatigue analysis of structures subjected to cyclic loads is to couple NASTRAN with programs which generate the cyclic loads through a user program which is external to both NASTRAN and the programs that generate the cyclic loads. In turn, the NASTRAN output should be coupled with the fatigue criteria through a user program such as is described herein. An example in this direction is the development of the WINTRAN (Wind Turbine Analysis) computerized analysis capability, which couples MOSTAB (ref. 1) and NASTRAN. This analysis capability couples MOSTAB (ref. 1), which generates the aerodynamic forces on the blades, with NASTRAN, which carries out the structural analysis of the integrated WTG system. WINTRAN is presently in the final stages of development.

In general, the most realistic of the two approaches to assess the fatigue resistance of structures subjected to cyclic loads is the dynamic approach. The results obtained using a quasi-static approach require considerable interpretation and judgment on the analyst's part. In addition, results from this approach have a high risk of possible misinterpretation.

It is clear from the procedure described herein, whereby NASTRAN is used to determine cyclic response, that it can be applied to a variety of cyclic loads. The fatigue resistance for these loads can then be assessed using the appropriate fatigue stress allowables.

SUMMARY OF RESULTS AND CONCLUSIONS

The major results and conclusions of an investigation using NASTRAN for the fatigue analysis of wind turbine generators (WTG) are as follows:

1. NASTRAN can be used to assess the fatigue resistance of WTG towers.
2. The most realistic of the two approaches to assess fatigue resistance is the dynamic approach.
3. The quasi-static approach yields results which could be in wide variance with those obtained from the dynamic case for means, amplitudes, amplitude ranges and frequencies.
4. It may not be known a priori which of the results obtained from the quasi-static approach are on the conservative side as compared with the direct dynamic response approach.
5. One effective way to use NASTRAN in the fatigue analysis of structures subjected to cyclic loads is through the use of externally written user programs which couple NASTRAN to cyclic load spectra generating programs and to application of fatigue criteria programs.

REFERENCES

1. Hoffman, John A.: Wind Turbine Analysis Using the MOSTAB Computer Program. MRI-2690-1, Mechanics Research, Inc., 1974.
2. Specification for the Design, Fabrication and Erection of Structural Steel for Buildings. S310, Feb. 12, 1969, American Institute of Steel Construction, Inc.
3. Linscott, Bradford S.; Shapton, William R.; and Brown, David: Tower and Rotor Blade Vibration Test Results for a 100-Kilowatt Wind Turbine. NASA TM X-3426, 1976.

TABLE I. - LOADING CONDITIONS FOR THE NASTRAN QUASI-STATIC MODEL

(MAXIMUM ALLOWABLE WIND VELOCITY, 40 MPH)

Sub-case	Blade ^a azimuthal angle, ψ_b , deg	Force load, lb			Moment load, ft-lb		
		Vertical, F_y	Lateral, F_x	Axial, F_z	Yaw, M_y	Roll, M_x	Pitch, M_z
1	0	-41 000	-600	3500	-9 400	254 300	35 400
2	15	-41 500	-1300	3100	8 700	286 000	54 100
3	30	-42 300	-1900	3000	28 600	293 100	62 200
4	45	-42 400	-1300	3500	42 900	259 100	56 700
5	60	-39 200	1100	4200	55 900	182 000	38 900
6	75	-36 700	1200	4700	29 400	129 900	38 900
7	90	-38 500	0	4800	-3 600	151 600	48 200
8	105	-42 100	300	4500	-3 800	201 800	47 000
9	120	-42 900	1200	4200	7 900	212 300	41 900
10	135	-41 100	300	4000	-7 100	192 300	47 900
11	150	-40 300	-1000	4000	-28 700	193 800	56 100
12	165	-40 700	-1200	3800	-30 000	220 500	38 800
13	180	-41 000	-600	3500	-9 400	254 300	35 400
14	195	-41 500	-1300	3100	8 700	286 000	54 100
15	210	-42 300	-1900	3000	28 600	293 100	62 200
16	225	-42 400	-1300	3500	42 900	259 100	56 700
17	240	-39 200	1100	4200	55 900	182 000	38 900
18	255	-36 700	1200	4700	29 400	129 900	38 900
19	270	-38 500	0	4800	-3 600	151 600	48 200
20	285	-42 100	300	4500	-3 800	201 800	47 000
21	300	-42 900	1200	4200	7 900	212 800	41 900
22	315	-41 100	300	4000	-7 100	192 300	47 900
23	330	-40 300	-1000	4000	-28 700	193 800	56 100
24	345	-40 700	-1200	3800	-30 000	220 500	38 800

^aEach azimuthal angle corresponds to one subcase.

TABLE II. - CYCLIC LOADS FOR DYNAMIC APPROACH^a

(a) First revolution.

Blade azimuthal angle, ψ_b , deg	Time, sec	Force load, lb			Moment load, ft-lb		
		x	y	z	x	y	z
0	0.0	-600	-41 000	3500	254 300	-9 400	35 400
15	0.0625	-1300	-41 500	3100	286 000	8 700	54 100
30	0.125	-1900	-42 300	3000	293 100	28 600	62 200
45	0.1875	-1300	-42 400	3500	259 100	42 900	56 700
60	0.25	1100	-39 200	4200	182 000	55 900	38 900
75	0.3125	1200	-36 700	4700	129 900	29 400	38 900
90	0.375	0	-38 500	4800	151 600	-3 600	48 200
105	0.4375	300	-42 100	4500	201 800	-3 800	47 000
120	0.50	1200	-42 900	4200	212 300	7 900	41 000
135	0.5625	300	-41 100	4000	102 300	-7 100	47 900
150	0.625	-1000	-40 300	4000	193 300	-28 700	56 100
165	0.6875	-1200	-40 700	3800	220 500	-30 000	38 800
180	0.75	-600	-41 000	3500	254 300	-9 400	35 400
195	0.8125	-1300	-41 500	3100	286 000	8 700	54 100
210	0.875	-1900	-42 300	3000	293 100	28 600	62 200
225	0.9375	-1300	-42 400	3500	259 100	42 900	56 700
240	1.0	1100	-39 200	4200	182 000	55 900	38 900
255	1.0625	1200	-36 700	4700	129 900	29 400	38 900
270	1.125	0	-38 500	4800	151 600	-3.600	48 200
285	1.1875	300	-42 100	4500	29 800	-3 800	47 000
300	1.25	1200	-42 900	4200	212 300	7 900	41 000
315	1.3125	300	-41 100	4000	192 300	-7 100	47 900
330	1.375	-1000	-40 300	4000	193 300	-28 700	56 100
345	1.4375	-1200	-40 700	3800	220 500	-30 000	38 800
360	1.50	-600	-41 000	3500	254 300	-9 400	35 400

^aThese loads are the same as those in table I except that they are expressed as a function of time for the dynamic approach.

TABLE II. - Concluded.

(b) Second revolution.

Blade azimuthal angle, ψ_b , deg	Time, sec	Force load, lb			Moment load, ft-lb		
		x	y	z	x	y	z
0	0.0	-600	-41 000	3500	254 300	-9 400	35 400
15	1.5625	-1300	-41 500	3100	286 000	8 700	54 100
30	1.625	-1900	-42 300	3000	293 100	28 600	62 200
45	1.6875	-1300	-42 400	3500	259 100	42 900	56 700
60	1.75	1100	-39 200	4200	182 000	55 900	38 900
75	1.8125	1200	-36 700	4700	129 900	29 400	38 900
90	1.875	0	-38 500	4800	151 600	-3 600	48 200
105	1.9375	300	-42 100	4500	201 800	-3 800	47 000
120	2.0	1200	-42 900	4200	212 300	7 900	41 000
135	2.0625	300	-41 100	4000	192 300	-7 100	47 900
150	2.125	-1000	-40 300	4000	193 300	-28 700	56 100
165	2.1875	-1200	-40 700	3800	220 500	-30 000	38 800
180	2.25	-600	-41 000	3500	254 300	-9 400	35 400
195	2.3125	-1300	-41 500	3100	286 000	8 700	54 100
210	2.375	-1900	-42 300	3000	293 100	28 600	62 200
225	2.4375	-1300	-42 400	3500	259 100	42 900	56 700
240	2.5	1100	-39 200	4200	182 000	55 900	38 900
255	2.5625	1200	-36 700	4700	129 900	29 400	38 900
270	2.625	0	-38 900	4800	151 600	-3 600	48 200
285	2.6875	300	-40 100	4500	201 800	-3 800	47 000
300	2.75	1200	-42 100	4200	212 300	7 900	41 000
315	2.8125	300	-41 100	4000	192 300	-7 100	47 900
330	2.875	-1000	-40 300	4000	193 300	-28 700	56 100
345	2.9375	-1200	-40 700	3800	220 500	-30 000	38 800
360	3.00	-600	-41 000	3500	254 300	-9 400	35 400

TABLE III. - TOWER GRAVITY LOAD DISTRIBUTION FOR
DYNAMIC APPROACH ANALYSIS

Tower elevation, ft	Corner grids				Vertical load/grid, lb
21	2101	2103	2105	2107	8220
38	3801	3803	3805	3807	5980
54	5401	5403	5405	5407	4280
68	6801	6803	6805	6807	2750
81	8101	8103	8105	8107	1460

TABLE IV. - WIND TURBINE GENERATOR MOD OA TOWER
CALCULATED FREQUENCIES

Mode	Frequency cycles per second			
	Free load case	Force field cases corresponding to:		
		Blades vertical	Blades at 45 degrees	Blades horizontal
1	3.69	3.68	3.68	3.68
2	5.90	5.87	5.87	5.87
3	----	5.87	5.87	5.87
4	6.00	5.98	5.94	5.98
5	----	6.00	6.02	6.01
6	----	6.26	6.26	6.26
7	6.89	6.87	6.87	6.87
8	7.27	7.23	6.95	7.23
9	7.38	7.49	7.80	7.57
10	----	8.70	8.70	8.70

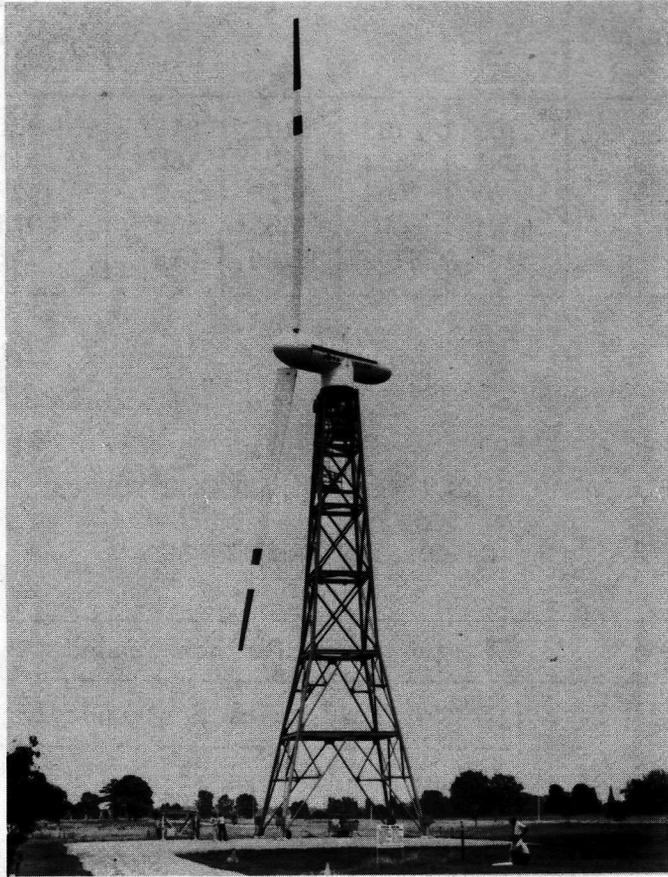


Figure 1. - Mod-0 100 kW wind turbine.

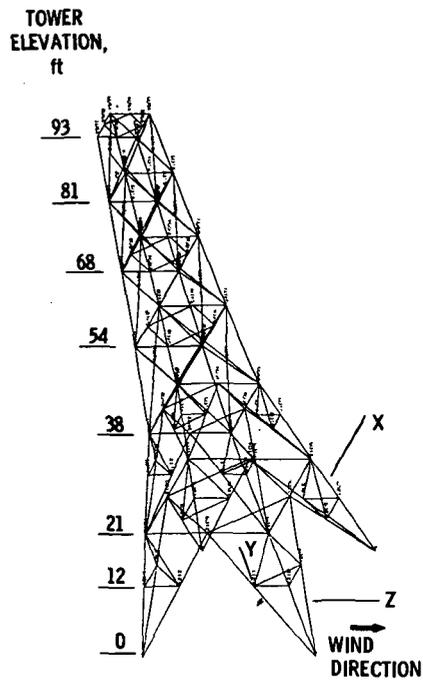


Figure 2 - Nastran model for Mod-0A WTG tower.

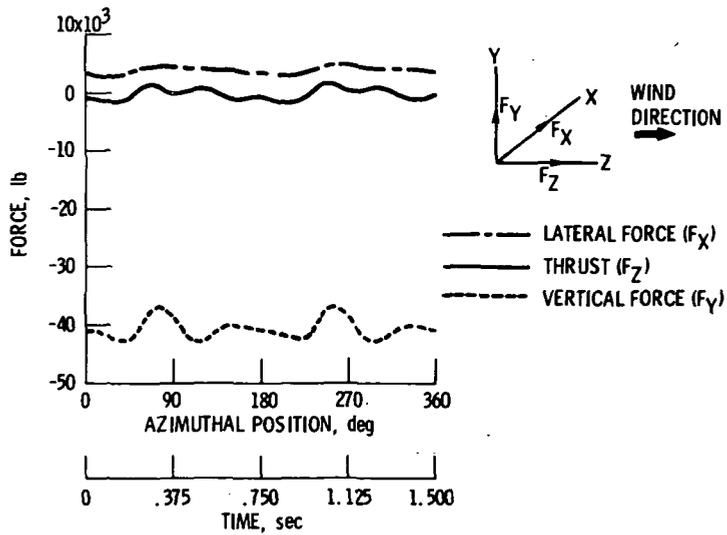


Figure 3 - Cyclic forces for one rotor revolution.

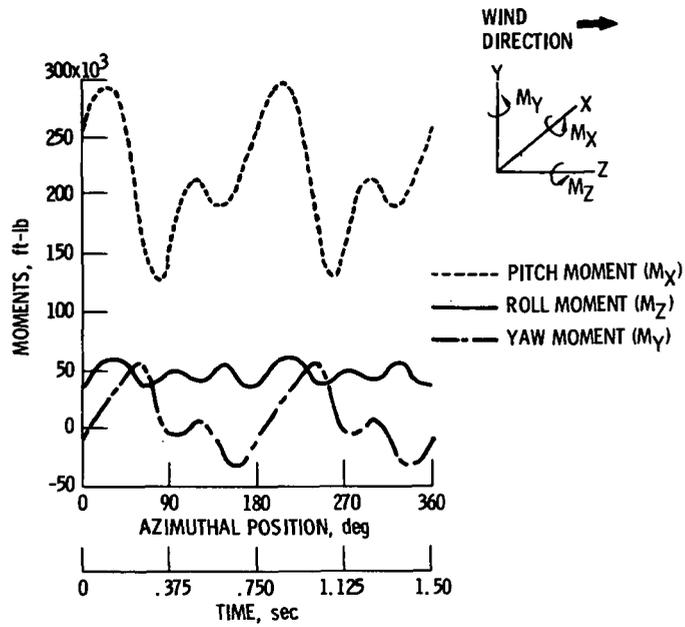


Figure 4. - Cyclic moments for one rotor revolution.

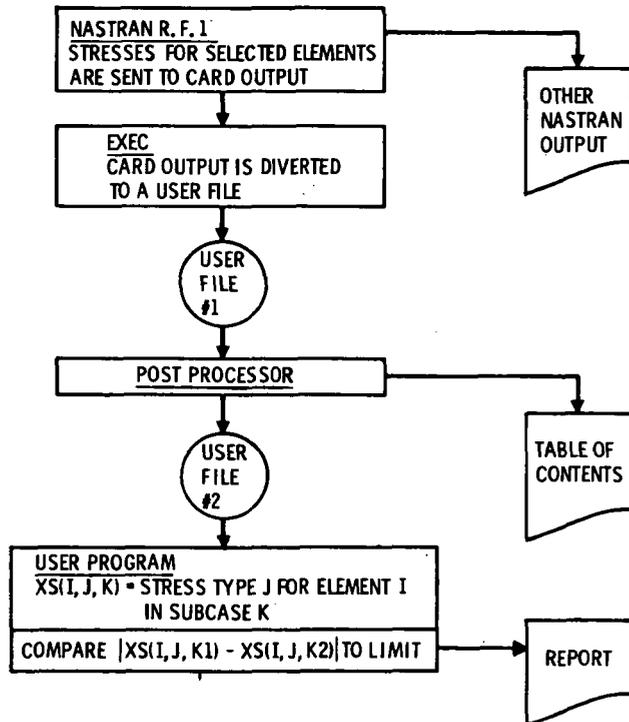


Figure 5. - Flow chart for quasi-static approach data retrieval.

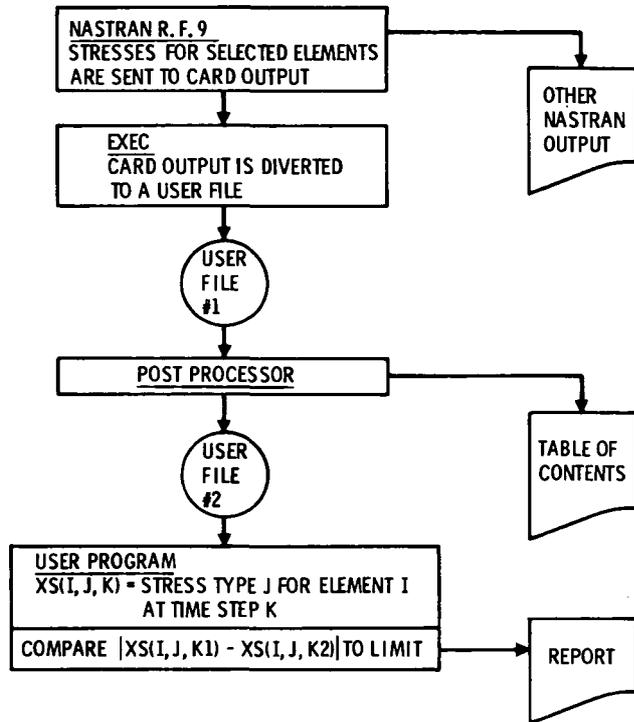


Figure 6. - Flow chart for dynamic approach data retrieval.

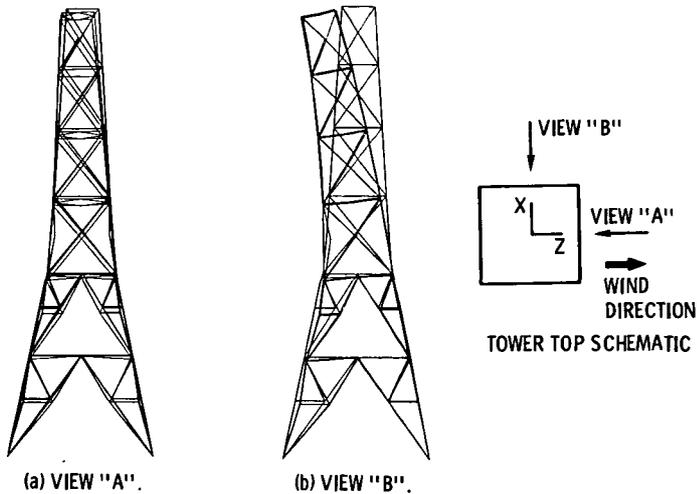


Figure 7. - Computer plots of deformed tower position. Mod-0A WTG at 40 rpm and at 40 mph wind velocity. (Quasi-static approach; blades vertical position; max. def. = 0.237 in.)

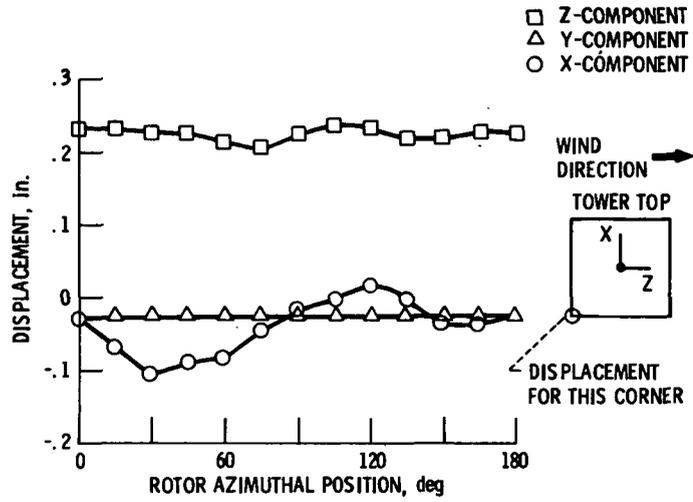


Figure 8. - Quasi-static approach computed top-of-tower corner displacements. Mod-0A WTG operating at 40 rpm and at 40 mph wind velocity.

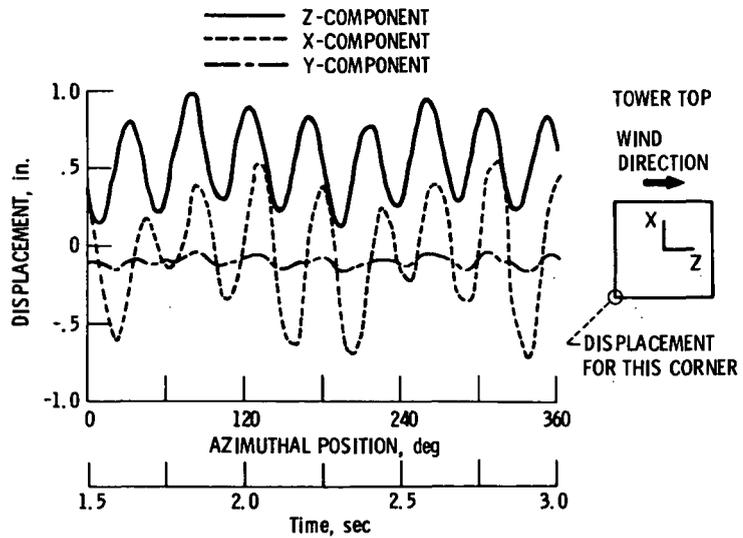


Figure 9. - Computed dynamic top-of-tower corner displacements Mod-0A WTG operating at 40 rpm and at 40 mph wind velocity. (No damping.)

Wind Turbine Structural Analysis

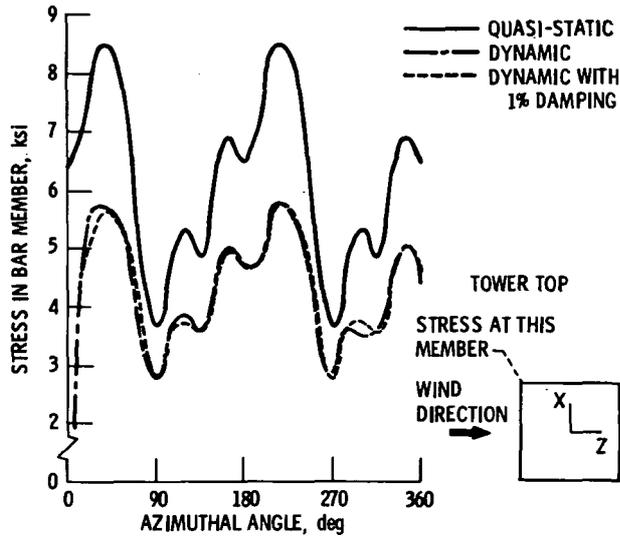


Figure 10. - Comparison of maximum computed stress at top-of-tower. Mod-OA WTG at 40 rpm and 40 mph wind velocity.

**Page
Intentionally
Left Blank**

NEW LARGE DEFLECTION ANALYSIS FOR NASTRAN

Myles M. Hurwitz
David W. Taylor Naval Ship Research and Development Center

SUMMARY

A new large deflection analysis is being developed for NASTRAN Level 16 as an alternative to Rigid Format 4, Static Analysis with Differential Stiffness, since it has been shown that differential stiffness effects alone are not sufficient to accurately solve geometrically nonlinear problems, especially those problems which involve a high degree of nonlinearity.

This paper, which represents a progress report for a long term, general nonlinear analysis NASTRAN project, contains (1) the theory of the structural analysis and numerical analysis methods presently used, and (2) some simple test problems comparing the new analysis with Rigid Format 4.

INTRODUCTION

It has long been realized that differential stiffness effects are only, at best, a first order approximation to the solution of structural analysis problems which contain geometrical nonlinearities, i.e., large deflections (ref. 1, 2, 3). Since flexible structures such as wings, antennas, skirts around advanced naval vehicles, etc. are subject to large deflections, it was decided to develop a large deflection analysis for NASTRAN Level 16 as an alternative to Rigid Format 4, Static Analysis with Differential Stiffness. Although other nonlinear computer programs exist, notably the MARC program, this project was undertaken in order to produce an easy-to-use, well-documented nonlinear capability for NASTRAN, a nonproprietary program already widely used for linear analysis.

The structural analysis theory presently employed is based on the work of Haisler (ref. 4), which uses a stationary Lagrangian coordinate system. In our present work, all terms of the strain-displacement relationships are included for possible development later of a large strain capability. The curvature relationships used in bending include just the linear terms. Therefore, the analysis includes large deflections with small to moderate rotations. All work to date has been limited to small strains and static analysis.

The numerical analysis techniques used to solve the nonlinear equations are based on the work of Haisler et al. (ref. 4,5), and six nonlinear equation solution methods have been included into NASTRAN in conjunction with the new capability.

Thus far, the finite elements which may be used with the new analysis, i.e., those for which a nonlinear stiffness matrix and force vector have been developed and inserted into the program, are BAR, QUAD1, and QUAD2. Insertion of TRIA1 and TRIA2 elements is nearing completion. The theoretical analysis for the isoparametric solid elements has been completed.

The paper will be divided into the following sections: General Theory, which describes the general approach to the large deflection analysis; Computation of K^* and Q^* for NASTRAN Quadrilateral Elements, which describes the application of the general theory to NASTRAN's QUAD1 and QUAD2 elements, Sample Problems, and Future Plans.

GENERAL THEORY

Equilibrium Equation

In this section, the general theory of the large deflection analysis will be presented. What will not be presented, however, are some of the already well-documented (ref. 6) steps in finite element theory.

Using the standard x,y,z Cartesian coordinate system with corresponding displacements u,v,w , the strain-displacement equations are:

$$\begin{aligned}
 \epsilon_{xx} &= u_x + \frac{1}{2} (u_x^2 + v_x^2 + w_x^2) \\
 \epsilon_{yy} &= v_y + \frac{1}{2} (u_y^2 + v_y^2 + w_y^2) \\
 \epsilon_{zz} &= w_z + \frac{1}{2} (u_z^2 + v_z^2 + w_z^2) \\
 \epsilon_{xy} &= u_y + v_x + u_x u_y + v_x v_y + w_x w_y \\
 \epsilon_{xz} &= u_z + w_x + u_x u_z + v_x v_z + w_x w_z \\
 \epsilon_{yz} &= v_z + w_y + u_y u_z + v_y v_z + w_y w_z
 \end{aligned} \tag{1}$$

For the case of small strains and elasticity, it is well-documented in finite element theory (ref. 7) that

$$U = \int_V \frac{1}{2} \{\epsilon\}^T [D] \{\epsilon\} dV \tag{2}$$

where

U is the strain energy,

$[D]$ is the symmetric matrix relating strains to stresses, i.e.,

$$\{\sigma\} = [D] \{\epsilon\},$$

V is the volume of the finite element, and
 $\{\epsilon\} = (\epsilon_{xx}, \epsilon_{yy}, \epsilon_{zz}, \epsilon_{xy}, \epsilon_{xz}, \epsilon_{yz})^T$ as given in equation (1).

Since $\{\epsilon\}$ now contains both linear and nonlinear terms, let us separate the strain energy U into linear and nonlinear portions, U_L and U_{NL} , respectively. That is,

$$U = U_L + U_{NL} \quad (3)$$

Now, since

$$U_L = \frac{1}{2} \int \{\epsilon_L\}^T [D] \{\epsilon_L\} dV \quad (4)$$

where $\{\epsilon_L\}$ is the linear portion of the strain vector $\{\epsilon\}$,

$$U_L = \frac{1}{2} \{u\}^T \int [B]^T [D] [B] dV \{u\} \quad (5)$$

where

$$\{\epsilon_L\} = [B] \{u\}, \text{ and}$$

$\{u\}$ is the vector of grid point displacements.

However, we know (ref. 7) that the strain energy is

$$U_L = \frac{1}{2} \{u\}^T [K] \{u\} \quad (6)$$

so that

$$[K] = \int_V [B]^T [D] [B] dV \quad (7)$$

or

$$[K]_{ij} = \frac{\partial^2 U_L}{\partial u_i \partial u_j} \quad (8)$$

where $[K]_{ij}$ is the (i,j) th term of the linear stiffness matrix. Also,

$$\frac{\partial U_L}{\partial u_i} = [K] \{u\}_i = \sum_j k_{ij} u_j, \quad (9)$$

the i^{th} term of the vector $[K] \{u\}$.

Now, using the Principle of Virtual Work, we arrive at the equilibrium equation

$$[K] \{u\} = \{Q\} - \{Q^*\} \quad (10)$$

where $\{Q\}$ is the vector of externally applied loads, and

$$\{Q^*\} = \left\{ \frac{\partial U_{NL}}{\partial u} \right\} \quad (11)$$

is the vector of pseudo forces due to nonlinearities. Therefore, at this point, all the nonlinear terms of the strain vector $\{\epsilon\}$ have been lumped on the right-hand side. However, for subsequent use, we define

$$[K^*]_{i,j} = \frac{\partial^2 U_{NL}}{\partial u_i \partial u_j} = \frac{\partial Q^*}{\partial u} \quad (12)$$

the stiffness matrix due to the nonlinear terms in the strain-displacement equations (1).

Methods of Solution

For a set of displacements $\{u\}$ which exactly satisfies the equilibrium equation (10), there is no force unbalance. However, when this is not true, the force unbalance can be defined as

$$\{f\} = -[K]\{u\} + \{Q\} - \{Q^*(u)\} \quad (13)$$

Let us assume that the applied load vector $\{Q\}$ is applied incrementally so that

$$\{Q\} = P\{\bar{P}\} \quad (14)$$

where

$\{\bar{P}\}$ is an initially applied load, and

P is a load parameter.

Therefore, $\{u\}$ and $\{Q^*\}$ may be considered as functions of P . Differentiation of equation (13) with respect to P yields

$$\{\dot{f}\} = -[K]\{\dot{u}\} + \bar{P} - \dot{Q}^*(u) \quad (15)$$

Now

$$\dot{Q}_i^* = \frac{\partial Q_i^*}{\partial P} = \frac{\partial Q_i^*}{\partial u_j} \frac{\partial u_j}{\partial P} = \frac{\partial Q_i^*}{\partial u_j} \dot{u}_j \quad (16)$$

or

$$\{\dot{Q}^*\} = \left[\frac{\partial Q^*}{\partial u} \right] \{\dot{u}\}, \quad (17)$$

and, by equation (12),

$$\{\dot{Q}^*\} = [K^*]\{\dot{u}\} \quad (18)$$

Therefore, an alternative formulation to equation (15) is

$$\{\dot{f}\} = - [K+K^*]\{\dot{u}\} + \{\bar{P}\} \quad (19)$$

Differentiating equation (13) a second time yields

$$\{\ddot{f}\} = - [K]\{\ddot{u}\} - \{\ddot{Q}^*(u)\} \quad (20)$$

The methods of solution to be described fall into three categories:

(1) Exact solution procedures which attempt to exactly satisfy the equilibrium equation, i.e., $\{f\} = 0$, (2) Initial value procedures which attempt to minimize the force unbalance, i.e., $\{\dot{f}\} = 0$, and (3) Self-correcting procedures which attempt to correct the response if it deviates from the equilibrium state.

Exact Solution Procedures

Exact solution procedures seek to satisfy $\{f\} = 0$. The best known method in this class is the Newton-Raphson approach.

Using a first-order Taylor series expansion of the i^{th} coordinate of $\{f\}$ about $\{u\}$ yields

$$f_i(u+\Delta u) = f_i(u) + \frac{\partial f_i(u)}{\partial u_j} \Delta u_j \quad (21)$$

Assuming that the unbalance at $u+\Delta u$ is 0,

$$f_i(u) = - \frac{\partial f_i(u)}{\partial u_j} \Delta u_j \quad (22)$$

Using equations (13) and (12), the Newton-Raphson procedure becomes

$$[K + K^*(u)]_n \{\Delta u\}_{n+1} = \{f(u)\}_n \quad (23)$$

and

$$\{u\}_{n+1} = \{u\}_n + \{\Delta u\}_{n+1} \quad (24)$$

Iterations using equations (23) and (24) continue until $\{\Delta u\}_M$ or $\{f(u)\}_M$ becomes small.

This iterative procedure is performed for each load step, making Newton-Raphson very accurate and very expensive (since K^* changes with each iteration). A modified Newton-Raphson technique allows K^* to remain constant for some number of iterations, which may be based on the size of $\{f(u)\}$. The Newton-Raphson procedure, with allowable modifications, has been included in NASTRAN.

Initial Value Procedures

The best known of the initial value techniques, which seek to satisfy $\{\dot{f}\} = 0$, is the incremental stiffness procedure.

Starting with equation (19), we have

$$[K + K^*(u)_n]\{\dot{u}\}_n = \{\bar{P}\} \quad (25)$$

for the n^{th} load increment. Using a forward difference approximation of $\{\dot{u}\}$

$$\{\dot{u}\}_n = \frac{1}{\Delta P} \{\Delta u\}_{n+1} = \frac{1}{\Delta P} \{u_{n+1} - u_n\} \quad (26)$$

where $\Delta P = P_{n+1} - P_n$, yields

$$[K + K^*(u)_n]\{\Delta u\}_{n+1} = (\Delta P)\{\bar{P}\} = \{\Delta Q\} \quad (27)$$

and

$$\{u\}_{n+1} = \{u\}_n + \{\Delta u\}_{n+1} \quad (28)$$

This incremental stiffness method has been included in NASTRAN.

This method is easy to apply and is relatively fast but has a tendency to drift away from the true solution. This problem can be alleviated by taking small load increments at the expense of increased time. The drifting can be controlled with correction terms in equation (27). This brings us to the next set of solution procedures.

Self-Correcting Procedures

The self-correcting solution techniques seek to overcome some of the limitations of other procedures, i.e., the time problem in the Newton-Raphson procedure and the accuracy problem of the incremental stiffness procedure. The first-order self-correcting procedure is characterized by

$$\{\dot{f}\} + z\{f\} = 0 \quad (29)$$

Substituting $z = 1/\Delta P$ (ref. 5) and equations (19) and (13) into equation (29) yields

$$[K + K^*(u)]\{\dot{u}\} + \frac{1}{\Delta P} [K]\{u\} = (1 + \frac{P}{\Delta P})\{\bar{P}\} - \frac{1}{\Delta P} \{Q^*(u)\} \quad (30)$$

One of the solution methods included in NASTRAN solves equation (30) using a fourth-order Runge-Kutta integration procedure.

Using equation (26) in equation (30) yields

$$\begin{aligned}
[K+K^*(u)_n]\{\Delta u\}_{n+1} &= \Delta P\{\bar{P}\} + P\{\bar{P}\} - [K]\{u\}_n - \{Q^*(u)_n\} \\
&= \Delta Q + \{f(u)\}_n
\end{aligned}
\tag{31}$$

which is exactly the same as the incremental stiffness procedure, equation (27), with a correction term, the force unbalance of the previous load increment, included. This method is generally regarded as one of the best solution procedures with respect to both time and accuracy and has been included in NASTRAN.

A third first-order self-correcting method included in NASTRAN is obtained as follows. The exact solution to equation (29) is

$$f(s) = ce^{-zs} \tag{32}$$

where $s=0$ at the start of a load step, and $s = \Delta P$ at the end. Then,

$$\{f(u)\}_n = \{f(0)\} = c \tag{33}$$

and

$$\{f(u)\}_{n+1} = \{f(\Delta P)\} = ce^{-z\Delta P} = ce^{-1} \tag{34}$$

Then, substituting equations (33), (34), and (13) into equation (32) yields

$$[K]\{u\}_{m+1} = ([K]\{u\}_m - P_m\{\bar{P}\} + \{Q^*\}_m)/e + P_{m+1}\{\bar{P}\} - \{Q^*\}_m \tag{35}$$

In addition to the first-order self-correcting procedure, a second-order procedure can be characterized by

$$\ddot{\{f\}} + c\dot{\{f\}} + z\{f\} = 0 \tag{36}$$

where c and z are arbitrary scalars. Following the procedures of references 5 and 8, $z = 2/(\Delta P\sqrt{P\Delta P})$, $c = z^2/2$, and

$$[K]\{u\}_{m+1} = -\{f\}_{m+1} + P_{m+1}\{\bar{P}\} - \{Q^*\}_m \tag{37}$$

where

$$\{f\}_{m+1} = e^{-c\Delta P/2}(A \cos \omega\Delta P + B \sin \omega\Delta P)$$

$$A = -[K]\{u\}_m + P_m\{\bar{P}\} - \{Q^*\}_m$$

$$B = (\dot{\{f\}}_m + cA/2)/\omega$$

$$\omega = \frac{1}{2} \sqrt{4z - c^2}$$

$$\dot{\{f\}}_m = -[K]\{\dot{u}\}_m + \{\dot{P}\} = \frac{1}{\Delta P} [K]\{u_m - u_{m-1}\} + \{\dot{P}\}$$

This method is the sixth one included in NASTRAN.

COMPUTATION OF K* AND Q* FOR NASTRAN QUADRILATERAL ELEMENTS

In order for a NASTRAN finite element to be used in the new large deflection analysis, subroutines computing $[K^*]$ and $\{Q^*\}$, as defined by equations (12) and (11), must be developed for the element. Thus far, such subroutines have been included in NASTRAN for the BAR, QUAD1, and QUAD2 elements. The development of the subroutines for TRIA1 and TRIA2 elements is nearing completion. The theoretical development has been completed for the solid isoparametric elements IHEX1 and IHEX2.

The BAR element development of K^* and Q^* is relatively simple and is given for the two-dimensional case in reference 4. The development for the QUAD1 and QUAD2 elements is much more complicated and is presented here.

The NASTRAN quadrilateral elements are composed of four overlapping basic bending triangles (TRBSC) superimposed over four membrane triangles (TRMEM). The membrane and bending properties are independent, i.e., not coupled. This uncoupling disappears in the nonlinear analysis, and the membrane and bending properties are computed together. The triangle coordinate system is seen in figure 1. Components of displacements parallel to the x and y axes of the element coordinate system are given by u and v, respectively, while deflection w is normal to the x-y plane, positive outward, and rotations α and β follow the right-hand rule.

The shape functions we will be using are (ref. 9):

$$\begin{aligned}
 u &= q_1 + q_2x + q_3y \\
 v &= q_4 + q_5x + q_6y \\
 w &= q_7 + (\gamma_x + q_8)x + (\gamma_y + q_9)y + q_{10}x^2 + q_{11}xy + q_{12}y^2 + q_{13}x^3 + q_{14}xy^2 + q_{15}y^3 \\
 \alpha &= q_9 + q_{11}x + 2q_{12}y + 2q_{14}xy + 3q_{15}y^2 \\
 \beta &= -q_8 - 2q_{10}x - q_{11}y - 3q_{13}x^2 - q_{14}y^2
 \end{aligned} \tag{38}$$

where

q_i , $i=1, \dots, 15$ are the generalized coordinates, and γ_x, γ_y are transverse shear strains.

These shape functions are the same as for the standard TRMEM and TRBSC elements and imply the same conditions concerning continuity and compatibility. The transverse shear strains γ_x, γ_y can be shown (ref. 9) to be:

$$\begin{aligned}
 \gamma_x &= c_{11}q_{13} + c_{12}q_{14} + c_{13}q_{15} \\
 \gamma_y &= c_{21}q_{13} + c_{22}q_{14} + c_{23}q_{15}
 \end{aligned} \tag{39}$$

where c_{ij} are constants which depend only on geometry.

The strain-displacement relationships used for the element are:

$$\{\epsilon\} = \begin{Bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{xy} \\ \chi_x \\ \chi_y \\ \chi_{xy} \end{Bmatrix} = \begin{Bmatrix} u_x + \frac{1}{2}(u_x^2 + v_x^2 + w_x^2) \\ v_y + \frac{1}{2}(u_y^2 + v_y^2 + w_y^2) \\ u_y + v_x + u_x u_y + v_x v_y + w_x w_y \\ -\beta_x \\ \alpha_y \\ \alpha_x - \beta_y \end{Bmatrix} \quad (40)$$

Notice that (1) the coupling of the membrane and bending terms occurs in the first three quantities of the strain vector, and (2) the curvatures $\chi_x, \chi_y, \chi_{xy}$ contain only linear terms. Since the strain vector for the standard linear NASTRAN TRBSC element is exactly the same as the last three terms of equation (40), we will drop those terms from consideration and remember, later, to add in the bending-only terms of the TRBSC stiffness matrix to the terms we derive here. Therefore, $\{\epsilon\}$ is now a 3x1 vector and can be written as follows:

$$\begin{aligned} \{\epsilon\} &= \begin{Bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{xy} \end{Bmatrix} = \begin{Bmatrix} u_x + \frac{1}{2}(u_x^2 + v_x^2 + w_x^2) \\ v_y + \frac{1}{2}(u_y^2 + v_y^2 + w_y^2) \\ u_y + v_x + u_x u_y + v_x v_y + w_x w_y \end{Bmatrix} \\ &= \begin{bmatrix} 1 + \frac{1}{2}u_x & 0 & \frac{1}{2}v_x & 0 & \frac{1}{2}w_x & 0 \\ 0 & \frac{1}{2}u_y & 0 & 1 + \frac{1}{2}v_y & 0 & \frac{1}{2}w_y \\ \frac{1}{2}u_y & 1 + \frac{1}{2}u_x & 1 + \frac{1}{2}v_y & \frac{1}{2}v_x & \frac{1}{2}w_y & \frac{1}{2}w_x \end{bmatrix} \begin{Bmatrix} u_x \\ u_y \\ v_x \\ v_y \\ w_x \\ w_y \end{Bmatrix} \end{aligned} \quad (41)$$

The vector on the right-hand side of equation (41) can be obtained from equation (38), resulting in

$$\{\epsilon\} = [B]\{q\} \quad (42)$$

where $[B]$ is given in table 1.

By evaluating equation (38) at each of the three triangle vertices, we obtain

$$\{u\} = [H]\{q\} \quad (43)$$

where $\{u\}$ is the 15x1 vector of grid point displacements, i.e.,

$$\{u\} = \begin{Bmatrix} u_1 \\ v_1 \\ w_1 \\ \vdots \\ \alpha_3 \\ \beta_3 \end{Bmatrix}$$

Therefore, equation (42) becomes

$$\{\epsilon\} = [B]\{q\} = [B][H]^{-1}\{u\} \quad (44)$$

By rearranging $\{u\}$, $[H]$ becomes a matrix with several blocks of zeroes. Therefore, instead of inverting a 15x15 matrix, only a 6x6 matrix, at worst, need be inverted, and this 6x6 matrix is the same one used in linear analysis.

The strain energy U , equation (3), is

$$U = \frac{1}{2} \int \{\epsilon\}^T [D] \{\epsilon\} dV \quad (45)$$

Using equation (44) in equation (45), we obtain

$$U = \frac{1}{2} \{u\} [H^{-1}]^T \int [B]^T [D] [B] dV [H^{-1}] \{u\} \quad (46)$$

From equations (4), (9), and (12),

$$\begin{aligned} [K + K^*]_{ij} = \frac{\partial^2 U}{\partial u_i \partial u_j} = & \left\{ \frac{\partial u^T}{\partial u_i} \right\} \left[\frac{\partial \bar{K}}{\partial u_j} \right] \{u\} + \{u\}^T \left[\frac{\partial \bar{K}}{\partial u_i} \right] \left\{ \frac{\partial u}{\partial u_j} \right\} \\ & + \left\{ \frac{\partial u^T}{\partial u_i} \right\} [\bar{K}] \left\{ \frac{\partial u}{\partial u_j} \right\} + \frac{1}{2} \{u\}^T \left[\frac{\partial^2 \bar{K}}{\partial u_i \partial u_j} \right] \{u\} \end{aligned} \quad (47)$$

where

$$[\bar{K}] = [H^{-1}]^T \int [B]^T [D] [B] dV [H^{-1}]$$

$$\left\{ \frac{\partial u^T}{\partial u_i} \right\} = (0, \dots, 1, \dots, 0), \text{ where } 1 \text{ is in the } i^{\text{th}} \text{ position so that the third term on the right-hand side of equation (47) reduces to } [\bar{K}]_{ij}$$

$$\left[\frac{\partial \bar{K}}{\partial u_i} \right] = [H^{-1}]^T \int [B^T D \frac{\partial B}{\partial u_i} + \frac{\partial B^T}{\partial u_i} DB] dV [H^{-1}]$$

$$\left[\frac{\partial^2 \bar{K}}{\partial u_i \partial u_j} \right] = [H^{-1}]^T \int \left[\frac{\partial B^T}{\partial u_j} D \frac{\partial B}{\partial u_i} + \frac{\partial B^T}{\partial u_i} D \frac{\partial B}{\partial u_j} \right] dV [H^{-1}]$$

(All volume integrals reduce to area integrals, i.e., $\int dV = t \int dA$, where t is the thickness of the plate.) The integrals of the triple matrix products required by equation (47) are complicated, and the integrations could have been performed using Gauss Quadrature. However, such a procedure would be time-consuming since up to four or five integration points in each of the x and y directions would be required to integrate all the terms. Instead, the matrix products were performed symbolically, and the area integrals of the resulting terms over the triangle area were performed symbolically using a very powerful symbolic manipulation system MACSYMA (ref. 10), available free of charge over the ARPANET computer network. For example, MACSYMA will symbolically compute $B^T D B$, thus showing that terms of the type $x^i y^j w_x w_y$ will have to be integrated. However, these terms are just polynomials in $x^i y^j$. By building a list of those terms required, a two-line MACSYMA program can be written to perform symbolically all required integrals $\int x^i y^j dA$, which will be in terms of x_b , x_c , y_c (figure 1). For example,

$$\int x^3 y^2 dA = y_c^3 (10x_b x_c^3 + 6x_b^2 x_c^2 + 3x_b^3 x_c + x_b^4) / 420$$

MACSYMA returns this result virtually instantaneously. Performing the integration in this manner results in exact integrals and large time savings.

Notice that in equation (47), the quantity $[K + K^*]$ has been computed, not just K^* . The computation of K^* requires almost all the steps required for computing K (plus a lot more, of course), so that the extra cost of computing K is trivial. Since all of the solution techniques require either K or $K+K^*$, if K^* alone were computed, the costly ADD operation would be required for those techniques requiring the sum.

Finally, from equations (11) and (46),

$$\{Q^*\}_i = \left\{ \frac{\partial u^T}{\partial u_i} \right\} [K] \{u\} + \frac{1}{2} \{u\}^T \left[\frac{\partial \bar{K}}{\partial u_i} \right] \{u\} - \{\bar{Q}\} \quad (48)$$

where $\{\bar{Q}\}$ is the vector resulting from the first two terms of equation (48) with $[K]$ replacing $[K]$.

Matrix $[K + K^*]$ is computed in functional module DSMG1 (Differential Stiffness Matrix Generator-Phase 1) with subroutines DQUAD and DTRBSC converted to the present specifications. Vector $\{Q^*\}$ is computed in a new functional module, QSTAR, but draws heavily from existing subroutines TRIQD, QDPLT, and TRBSC, which compute thermal load vectors.

SAMPLE PROBLEMS

Two relatively simple sample problems were run for this progress report. The first is a highly nonlinear, one-degree-of-freedom truss-spring problem (fig. 2). Figure 3 shows the truss-spring results for linear analysis, differential stiffness analysis from Levels 15 and 16, and exact results. The exact results were obtained with the Newton-Raphson procedure using a very small load increment and agrees with reference 4. Figures 4 and 5 show the results of the problem using seven solution methods with $\Delta P=1$. The modified Newton-Raphson did not update K^* at every iteration. Instead, K^* was updated only when the norm of the force unbalance vector exceeded 10% of the load at a particular load increment. Also, notice how, in figure 4, the first-order self-correcting procedure deviates from the exact solution at a load of 8 lbs., but how it "corrects itself" at a load of 9 lbs. Figures 6 and 7 show the results of the same seven methods, but with $\Delta P=3$. (All loads and displacements in figures 3-7 have been plotted as positive for simplicity.) The results show that, at least for this problem, the Newton-Raphson and first-order self-correcting methods are the most accurate. The latter is the one used in the MARC program (ref. 11) and is, in fact, the method recommended for general use since it combines the advantages of good accuracy and reasonable cost.

The second sample problem is seen in figure 8. Only the lower left quarter of the plate was modeled, and a 3x3 mesh of QUAD2 elements was used. The 15 psi load was reached with three load increments of 5 psi each. The solution method was first-order self-correcting. Analytical results (ref. 12) give the normal deflection of the center of the plate as .116 inches. NASTRAN gave .117 inches. The time to compute $K+K^*$ on a CDC 6400 computer was approximately 6.5 seconds per QUAD2 element. This compares with the 2.2 seconds per element required to compute K in EMG/EMA.

This problem was run with standard Rigid Format 4, and null differential stiffness matrices were computed. The problem was also run on the MARC program (ref. 11) using quadratic isoparametric quadrilateral shell elements. The mesh and load increments were the same as those run with NASTRAN, and the same result, .117 inches, was obtained. The time required to compute K^* was approximately 12 seconds per element.

FUTURE PLANS

Some possible future work for this project includes:

- (1) More efficient computation of K^* . The DSMG1 module used to compute K^* was not converted to EMG-type in Level 16, but is still SMA1-type. Converting to EMG-type would immediately reduce the time by approximately a factor of 2.
- (2) Allowing more of NASTRAN's finite elements into the new analysis.
- (3) Stress computation. Functional module SDR2 (Stress Data Recovery-Phase 2) will have to be modified to include the nonlinear effects of

the allowable elements. In fact, computation of stresses in differential stiffness analyses has always been incorrect since the nonlinear terms were never included.

- (4) Plasticity.
- (5) Combined large deflection and plasticity.
- (6) Large strains.
- (7) Extension to dynamic loads.

REFERENCES

1. Marcal, P.V.: Effect of Initial Displacements on Problems of Large Deflection and Stability. Brown Univ. Eng. Report, ARPA E54, Nov. 1967.
2. Mallett, R.H. and Marcal, P.V.: Finite Element Analysis of Nonlinear Structures. J. Struct. Div., Proc. ASCE, V. ST9, Sept. 1968.
3. Carey, G.F.: A Unified Approach to Three Finite Element Theories for Geometric Nonlinearity. Computer Methods in Applied Mechanics and Engineering, 4 (1974), pp. 69-79.
4. Haisler, W.E.: Development and Evaluation of Solution Procedures for Nonlinear Structural Analysis. Ph.D. Dissertation, Texas A&M, Dec. 1970.
5. Tillerson, J.R., Stricklin, J.A., and Haisler, W.E.: Numerical Methods for the Solution of Nonlinear Problems in Structural Analysis. Symposium on Numerical Solution of Nonlinear Structural Problems, ASME 1973 Winter Annual Meeting, Nov. 11-15, 1973, Detroit, Michigan.
6. Zienkiewicz, O.C.: The Finite Element Method in Engineering Science, 2nd Ed. McGraw-Hill, London, 1971.
7. Gallagher, R.H.: Finite Element Analysis Fundamentals. Prentice-Hall, 1975.
8. Massett, D.A. and Stricklin, J.A.: Self-Correcting Incremental Approach in Nonlinear Structural Mechanics. AIAA J., V. 9, No. 12, Dec. 1971.
9. The NASTRAN Theoretical Manual (Level 16.0). NASA SP-221(03), March, 1976.
10. MACSYMA Reference Manual. The MATHLAB Group, Project MAC, Massachusetts Institute of Technology, 1975.
11. The MARC System, Volume A: User Information Manual. MARC Analysis Research Corporation.
12. Boland, P.L.: Large Deflection Analysis of Thin Elastic Structures by the Assumed Stress Hybrid Finite Element Method. Air Force Office of Scientific Research Report TR 76-1111, Oct. 1975.

TABLE I. - [B] MATRIX FOR EQUATION (42)

$$\{\epsilon\} = [B]\{q\}$$

$$\{\epsilon\} = \begin{bmatrix} 0 & 1+\frac{1}{2}u_x & 0 & 0 & \frac{1}{2}v_x & 0 & 0 & \frac{1}{2}w_x & 0 & xw_x & \frac{1}{2}yw_x & 0 \\ 0 & 0 & \frac{1}{2}u_y & 0 & 0 & 1+\frac{1}{2}v_y & 0 & 0 & \frac{1}{2}w_y & 0 & \frac{1}{2}xw_y & yw_y \\ 0 & \frac{1}{2}u_y & 1+\frac{1}{2}u_x & 0 & 1+\frac{1}{2}v_y & \frac{1}{2}v_x & 0 & \frac{1}{2}w_y & \frac{1}{2}w_x & xw_y & \frac{1}{2}(yw_y+xw_x) & yw_x \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{2}w_x(3x^2+c_{11}) & \frac{1}{2}w_x(y^2+c_{12}) & \frac{1}{2}c_{13}w_x \\ \frac{1}{2}c_{21}w_y & \frac{1}{2}w_y(2xy+c_{22}) & \frac{1}{2}w_y(3y^2+c_{23}) \\ \frac{1}{2}w_y(3x^2+c_{11}) & \frac{1}{2}w_y(y^2+c_{12}) & \frac{1}{2}c_{13}w_y \\ \frac{1}{2}c_{21}w_x & \frac{1}{2}w_x(2xy+c_{22}) & \frac{1}{2}w_x(3y^2+c_{23}) \end{bmatrix} \begin{Bmatrix} q_1 \\ q_2 \\ \vdots \\ q_{15} \end{Bmatrix}$$

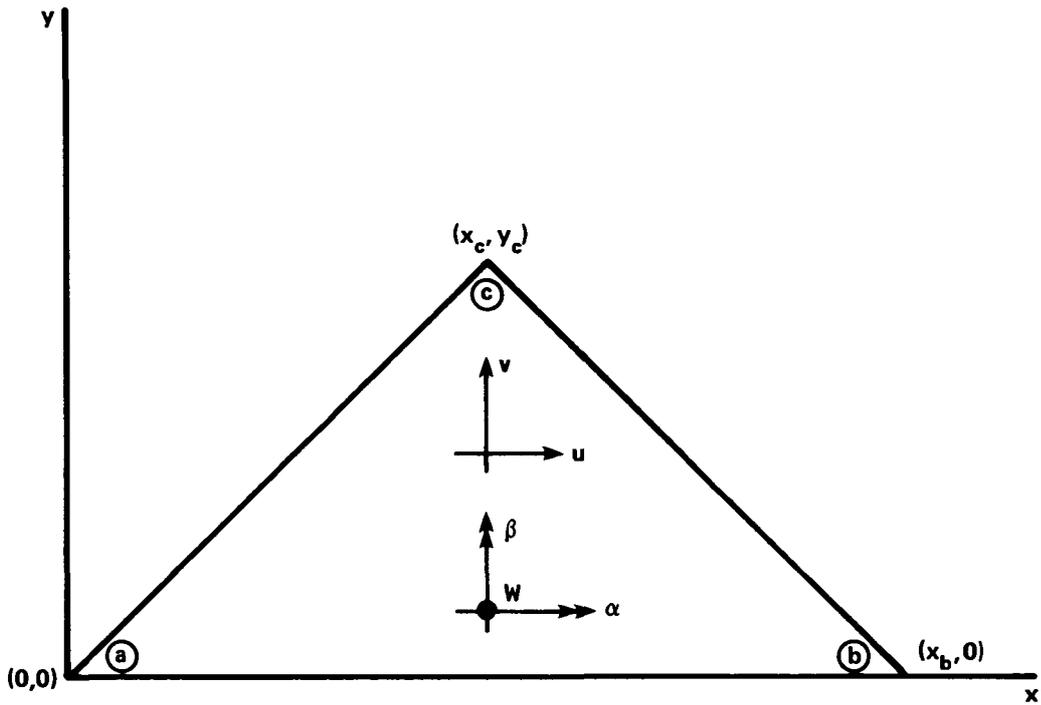
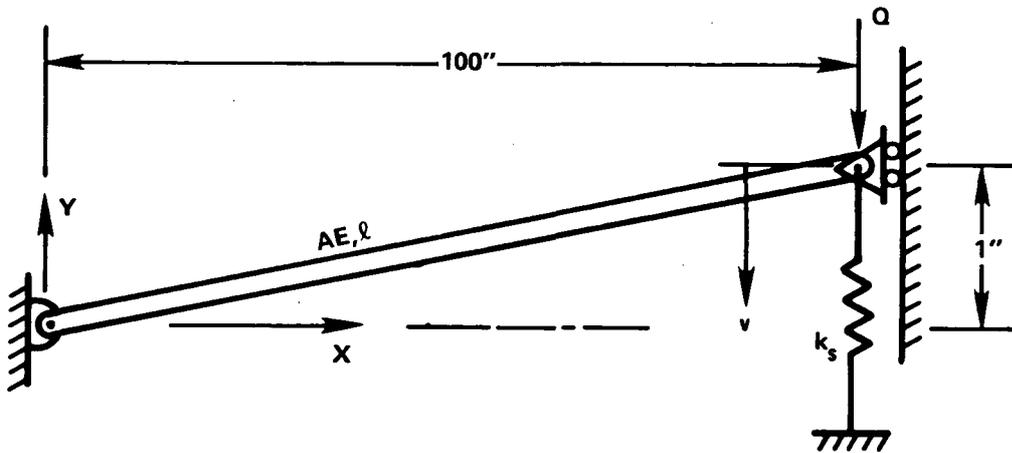


Figure 1 - Triangle Coordinate System



$E = 10^7 \text{ PSI}$
 $A = 1 \text{ IN}^2$

$Q = \text{APPLIED LOAD}$
 $k_s = \text{SPRING CONSTANT} = 6.0 \text{ LB/IN}$
 $v = \text{DISPLACEMENT}$

Figure 2 - Truss-Spring Problem

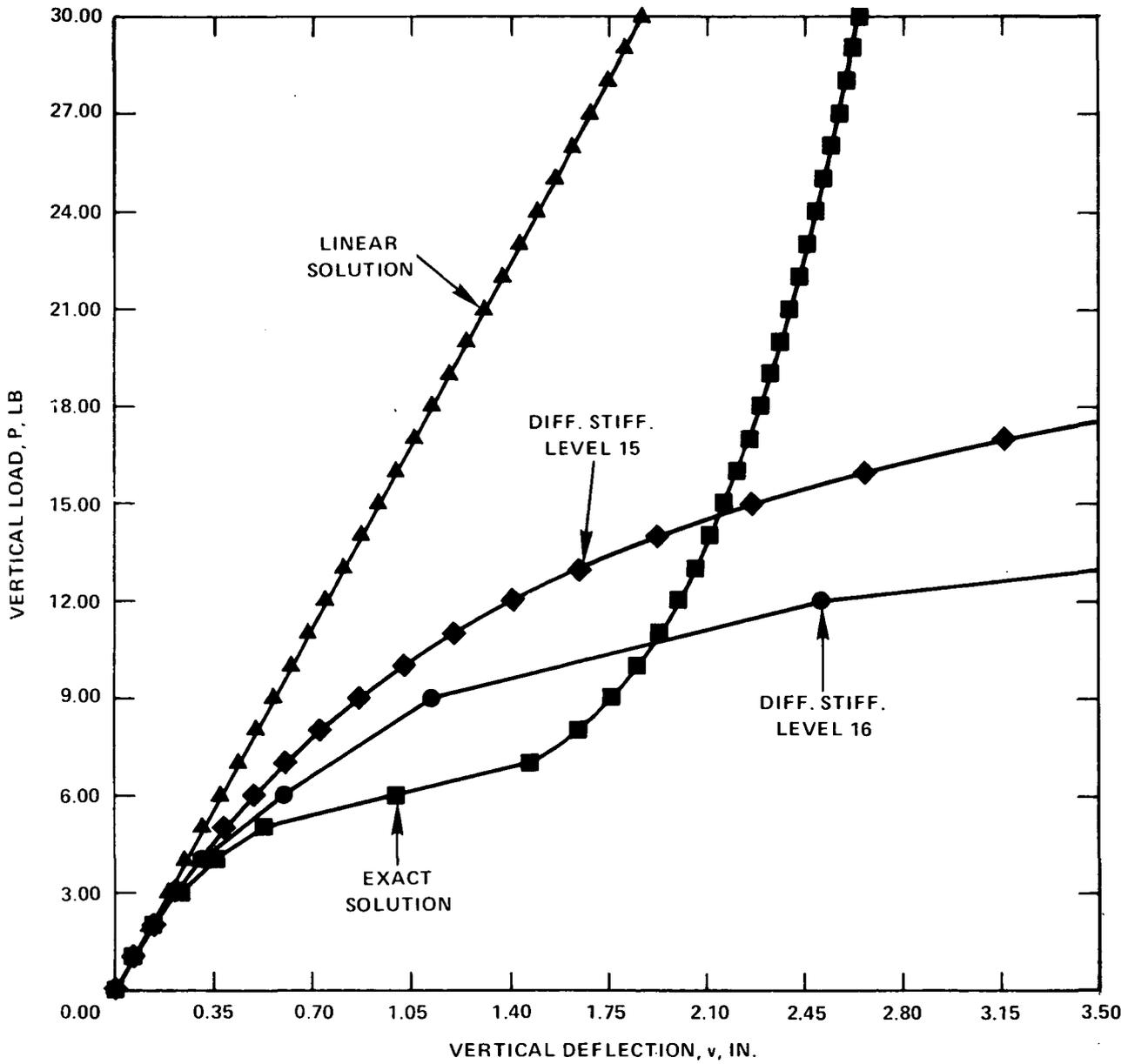


Figure 3 - Load-Deflection Curves for Truss-Spring Problem

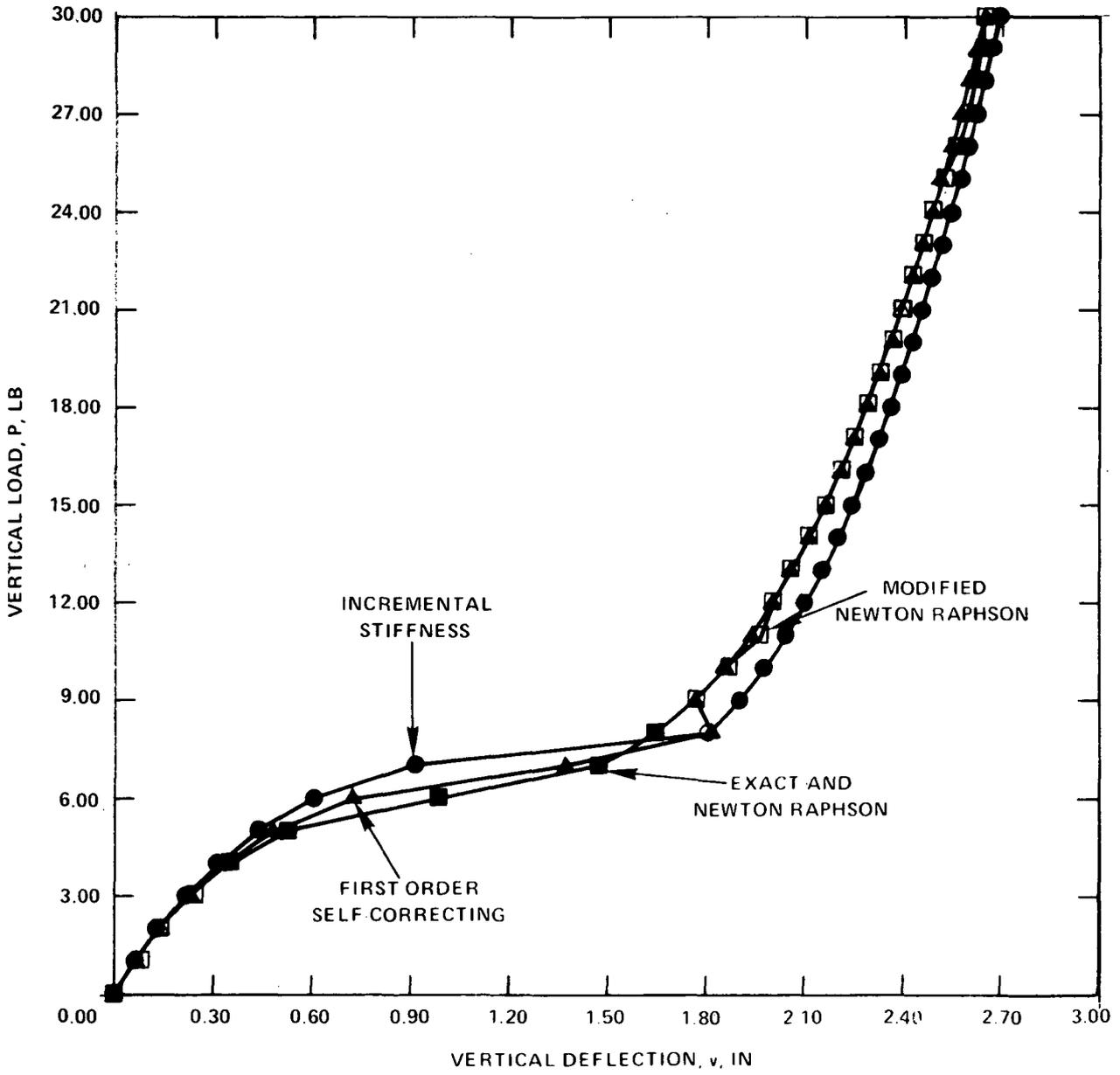


Figure 4 - Truss-Spring Problem Solutions 1 ($\Delta P = 1.0$)

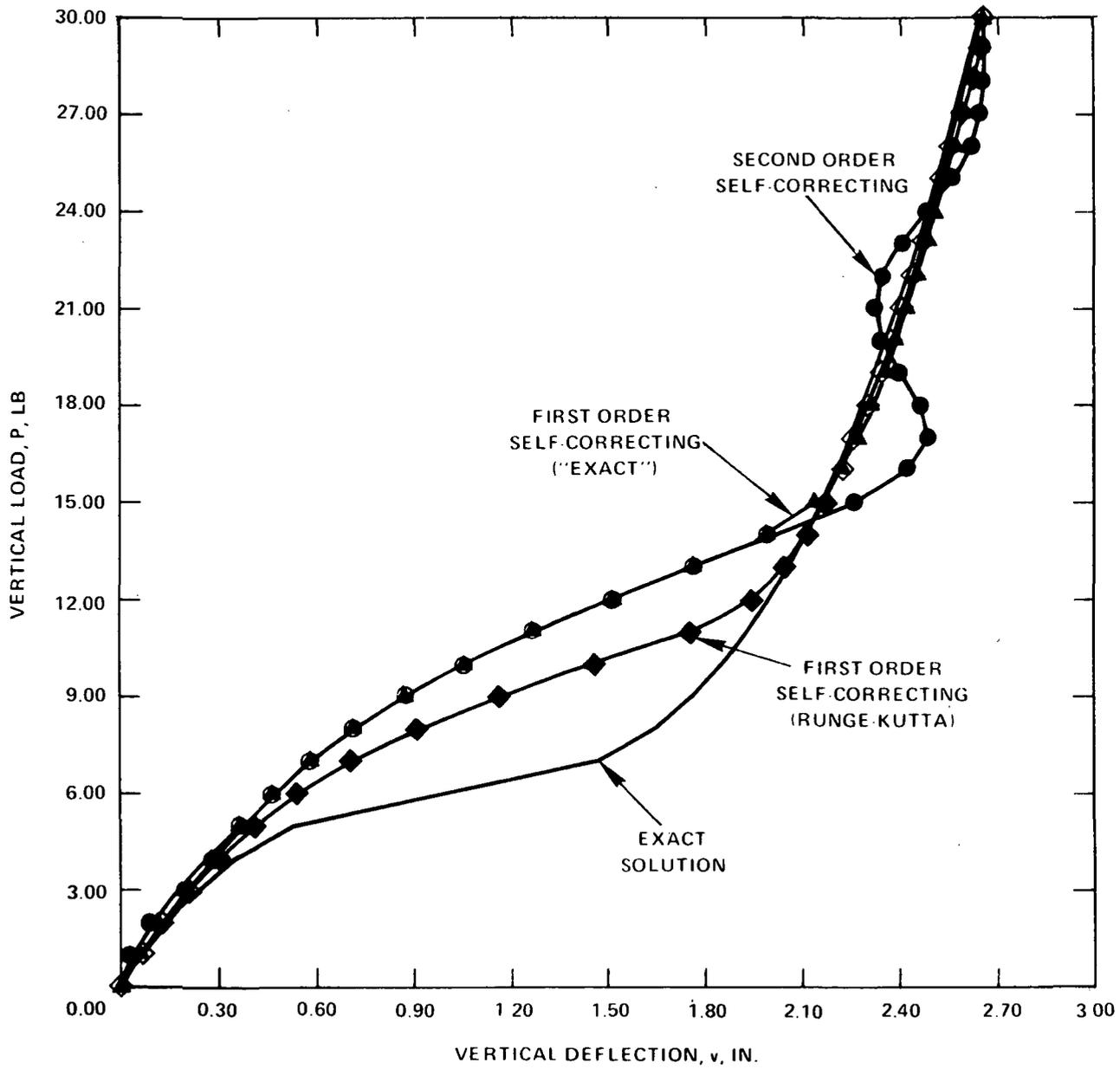


Figure 5 - Truss-Spring Problem Solutions 2 ($\Delta P = 1.0$)

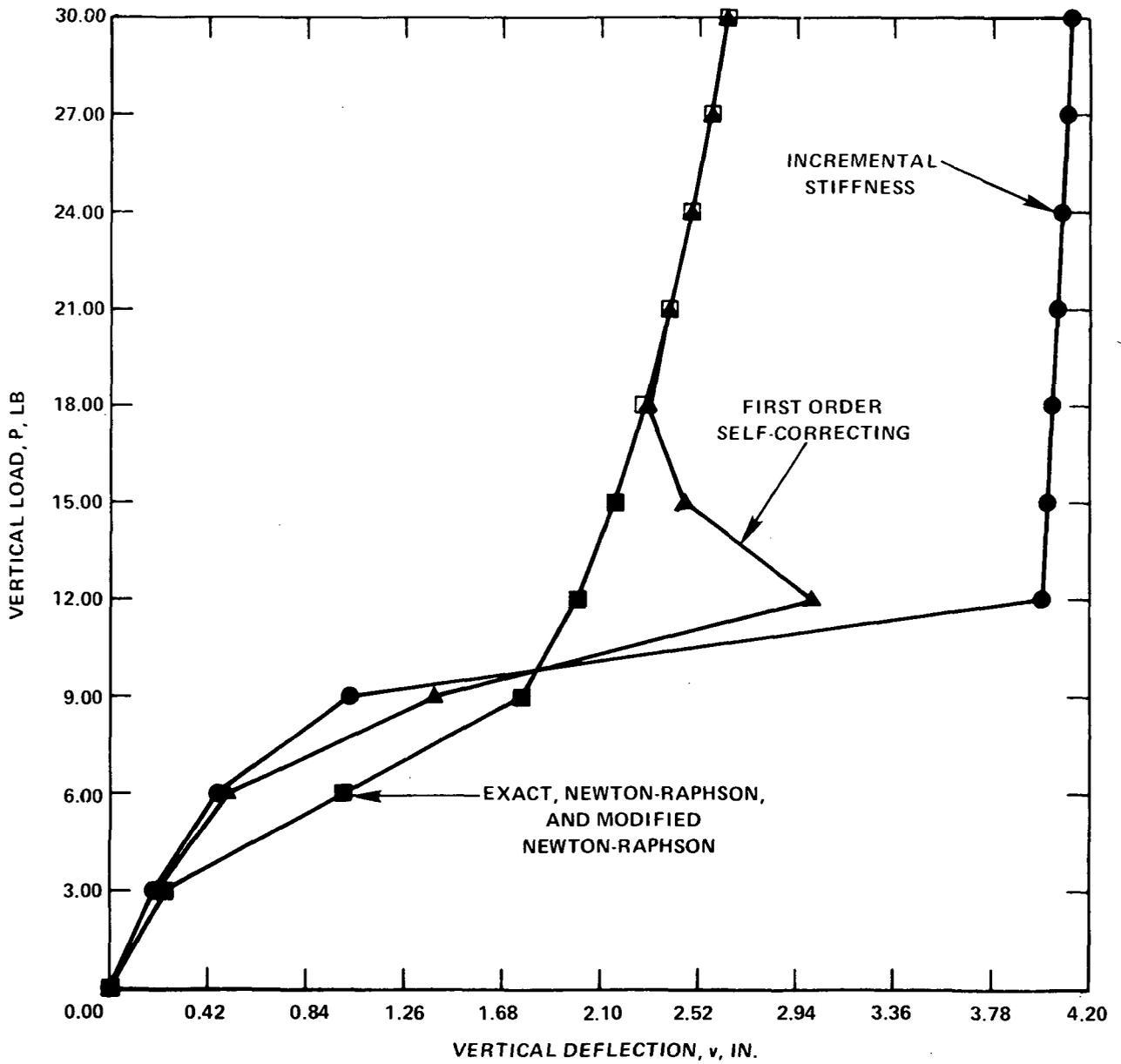


Figure 6 - Truss-Spring Problem Solutions 1 ($\Delta P = 3.0$)

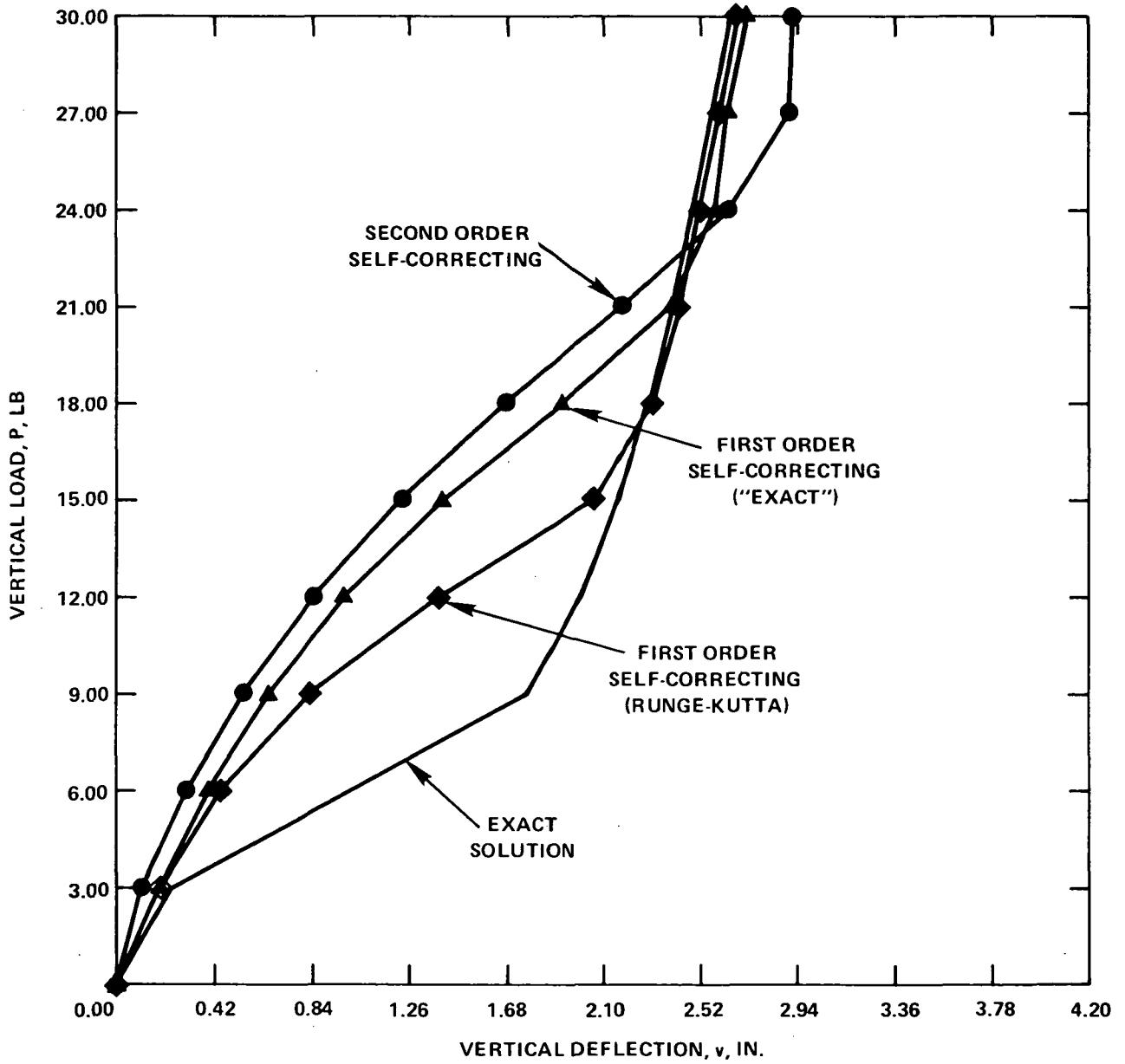


Figure 7 - Truss-Spring Problem Solutions 2 ($\Delta P = 3.0$)

Example
Vibrational
Modal Analysis

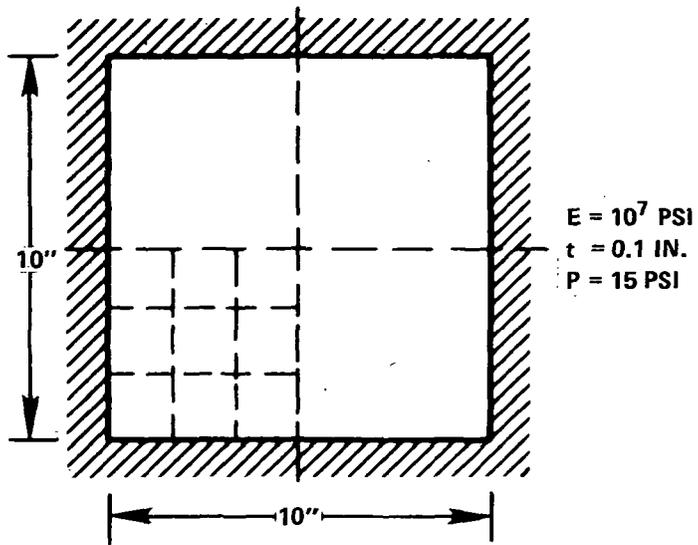


Figure 8 - Clamped Plate Problem

**Page
Intentionally
Left Blank**

SINGULAR PLASTIC ELEMENT:

NASTRAN IMPLEMENTATION AND APPLICATION

M. A. Hussain, S. L. Pu and W. E. Lorensen
Department of the Army
Watervliet Arsenal
Watervliet, New York 12189

SUMMARY

The elastic and plastic singularities near a crack tip are obtained from higher order isoparametric elements. This is simply accomplished by collapsing the quadrilateral element into triangular element and by judicious choice of adjacent mid-side nodes.

Specifically for the cubic element the elastic singularity is obtained by placing the mid-side nodes adjacent to the crack tip at $1/9$ th and $4/9$ th locations. The plastic singularity is constructed using the sliding node concept. These elements have been implemented in NASTRAN as user dummy elements.

INTRODUCTION

In recent years there has been a wide acceptance of linear fracture mechanics resulting in the development of new structural alloys having high fracture toughness and maintaining yield strength close to previous levels.

However, plasticity plays a major role in the application of these materials either in thin cross sections or under mixed mode conditions. Also in some cases, to meet the ASTM requirement for plane strain fracture toughness testing, the specimens required are too large for economical testing. To alleviate some of these problems a number of methods have been proposed, e.g. Irwin's equivalent 'Elastic Crack Length' (ref. 1), Well's Crack Opening Displacement (ref. 2), Rice's Path Independent J-Integral (ref. 3) and Non-linear Energy Methods proposed by Liebowitz and his coworkers (ref. 4), the last two being quite promising. Hence it is necessary to model the plastic condition near crack tip as accurately as possible.

In this paper, we implement higher order isoparametric elements (quadratic and cubic) in NASTRAN's piecewise linear (plasticity) module. By judicious choice of intermediate grid points, and using proper constraints, we develop elastic, and elastic-plastic singular elements.

Specifically, the elastic singular cubic element embodying the square root ($1/\sqrt{r}$) singularity is constructed by placing the midside nodes, adjacent to the crack tip, at 1/9th and 4/9th locations. The plastic singular element is constructed for the Ramberg-Osgood type of material with zero hardening exponent (ideally plastic material) using the 'Sliding Node Concept' of Barsoum (ref. 5).

'Sliding Nodes' are simply achieved by collapsing one side of an element and surrounding the crack tip with these elements, so that the crack tip has multiple independent nodes at one physical location which slide with respect to each other during deformation, due to loading. The proper order for plastic singularity (i.e. $1/r$) is achieved by locating the adjacent midside nodes at 1/9th and 4/9th of the length of the side of the element, as done for the elastic element.

After a brief review of the theory proving the existence of crack tip singularities, we discuss the implementation of these elements in Nastran as user dummy elements. The results of the analysis are compared to a prandtl-slip line field solution.

Many general purpose finite element codes as well as advanced versions of NASTRAN may have these elements. Hence, the method may be quite accessible to many users.

SYMBOLS

$(x,y), (r,\theta)$	cartesian and cylindrical coordinates
(ξ,η)	curvilinear coordinates
x_i, y_i, ξ_i, η_i	grid point coordinates
N_i	shape function at grid point i
u, v	cartesian displacements
ϵ_{ij}	strain tensor
σ_{ij}	stress tensor
S_{ij}, e_{ij}	deviatoric stress and strain tensors
W	strain energy density
J	path independent integral
$[J]$	Jacobian matrix
n	strain hardening exponent

CRACK TIP SINGULARITIES

Consider the path independent integral J developed by Rice (ref. 3,6),

$$J = \int_{\Gamma} (W dy - \bar{\tau} \cdot \frac{\partial \bar{u}}{\partial x} ds) \quad (1)$$

where W is the strain energy density, $\bar{\tau}$ and \bar{u} traction and displacement vectors on the path Γ . Using a circular path of radius r surrounding a crack tip (1) reduces to,

$$J = r \int_{-\pi}^{\pi} \{W \cos\theta - \bar{\tau} \cdot \frac{\partial \bar{u}}{\partial x}\} d\theta \quad (2)$$

The terms in {...} in above are of the form:

$$(\text{stress})(\text{strain}),$$

hence for the nonvanishing contribution to J (which is identical to energy release rate for the elastic case) we have

$$\sigma_{ij} \epsilon_{ij} = O\left(\frac{1}{r}\right) \text{ as } r \rightarrow 0. \quad (3)$$

Equation (3) is quite familiar for the elastic case for which stress and strain each have a singularity of order one half at the crack tip.

Now consider the Ramberg-Osgood type of material given by

$$\tau = G\gamma = \frac{\tau_0}{\gamma_0} \gamma, \quad \text{for } \gamma \leq \gamma_0 \quad (4)$$

$$\tau = T_0 \left(\frac{\gamma}{\gamma_0}\right)^n, \quad \text{for } \gamma \geq \gamma_0 \quad (5)$$

where $\tau = \sqrt{1/2 S_{ij} S_{ij}}$, $\gamma = \sqrt{2 e_{ij} e_{ij}}$, and τ_0 , γ_0 are yield stress and strain in shear and n is the hardening exponent. From (4), (5) and (3), outside the elastic range, we have

$$\begin{aligned} \sigma_{ij} &= O\left(r^{\frac{-n}{n+1}}\right) \\ \epsilon_{ij} &= O\left(r^{\frac{1}{1+n}}\right) \end{aligned} \quad (6)$$

From (6) we have the familiar elastic case for $n = 1$. However when $n = 0$, which is the case of ideally plastic material we have, from (6)

$$\begin{aligned}\sigma_{ij} &= 0(r^0) \\ \epsilon_{ij} &= 0(r^{-1})\end{aligned}\tag{7}$$

indicating a singularity of order one for the strains.

The existence of such singularities for quadratic elements have been given in (ref. 5,7). In the next section we briefly outline the case of cubic element.

SINGULARITIES OF CUBIC ELEMENT

Following the notation of (ref. 8), the geometry of 12-point cubic element is mapped into a normalized square in (ξ, η) plane ($-1 \leq \xi \leq 1$, $-1 \leq \eta \leq 1$) through the transformation,

$$\begin{aligned}x &= \sum_{i=1}^{12} N_i(\xi, \eta) x_i, \\ y &= \sum_{i=1}^{12} N_i(\xi, \eta) y_i,\end{aligned}\tag{8}$$

where the shape function is given by,

$$\begin{aligned}N_i &= \frac{1}{256} (1 + \xi\xi_i)(1 + \eta\eta_i)[-10 + 9(\xi^2 + \eta^2)][-10 + 9(\xi_i^2 + \eta_i^2)] \\ &+ \frac{81}{256} (1 + \xi\xi_i)(1 + 9\eta\eta_i)(1 - \eta^2)(1 - \eta_i^2) \\ &+ \frac{81}{256} (1 + \eta\eta_i)(1 + 9\xi\xi_i)(1 - \xi^2)(1 - \xi_i^2),\end{aligned}\tag{9}$$

(x_i , y_i and ξ_i , η_i are the grid points.)

Collapsing the quadrilateral element as shown in Figure 1 and placing the midside nodes at 1/9th and 4/9th location we have

$$x_1 = x_{10} = x_{11} = x_{12} = 0, \quad x_2 = x_9 = h/9, \quad x_3 = x_8 = 4h/9,$$

$$x_4 = x_5 = x_6 = x_7 = h,$$

$$y_1 = y_{10} = y_{11} = y_{12} = 0, \quad y_2 = -y_9 = -\ell/9, \quad y_3 = -y_8 = -4\ell/9$$

$$y_4 = -y_7 = -\ell, \quad y_5 = -y_6 = -\ell/3 \quad (10)$$

Substituting (10) into equation (8) we have

$$x = \frac{h}{4} (1 + \xi)^2 \quad (11)$$

$$y = \frac{\ell\eta}{4} (1 + \xi)^2$$

Any point at a radial distance, $r = (x^2 + y^2)^{1/2}$, from the crack tip is given by

$$r = \frac{\ell}{4} (1 + \xi)^2 \left[\left(\frac{h}{\ell}\right)^2 + \eta^2 \right]^{1/2}$$

or

$$(1 + \xi) = \sqrt{r} \frac{1}{\left\{ \frac{\ell}{4} \left[\left(\frac{h}{\ell}\right)^2 + \eta^2 \right]^{1/2} \right\}^{1/2}} \quad (12)$$

The Jacobian [J] is given by

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} \frac{h}{2}(1 + \xi) & \frac{\ell}{2}\eta(1 + \xi) \\ 0 & \frac{\ell}{4}(1 + \xi)^2 \end{bmatrix} \quad (13)$$

and the determinant is

$$\det |J| = \frac{h\ell}{8} (1 + \xi)^3 \quad (14)$$

For the inverse functions, we have

$$[J]^{-1} = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{2}{h(1+\xi)} & \frac{-4\eta}{h(1+\xi)^2} \\ 0 & \frac{4}{\lambda(1+\xi)^2} \end{bmatrix} \quad (15)$$

The displacement components of the point (ξ, η) for isoparametric transformation are,

$$u = \sum_{i=1}^{12} N_i(\xi, \eta) u_i$$

$$v = \sum_{i=1}^{12} N_i(\xi, \eta) v_i \quad (16)$$

The derivatives of u, v with respect to ξ, η are

$$\frac{\partial u}{\partial \xi} = \sum_{i=1}^{12} \frac{\partial N_i}{\partial \xi} u_i, \quad \frac{\partial u}{\partial \eta} = \sum_{i=1}^{12} \frac{\partial N_i}{\partial \eta} u_i$$

$$\frac{\partial v}{\partial \xi} = \sum_{i=1}^{12} \frac{\partial N_i}{\partial \xi} v_i, \quad \frac{\partial v}{\partial \eta} = \sum_{i=1}^{12} \frac{\partial N_i}{\partial \eta} v_i \quad (17)$$

where

$$\frac{\partial N_i}{\partial \xi} = \frac{1}{256} (1 + \eta \eta_i) [-10 + 9(\xi_i^2 + \eta_i^2)] (-10\xi_i + 9\xi_i \eta^2 + 18\xi + 27\xi_i \xi^2)$$

$$+ \frac{81}{256} (1 - \eta_i^2) \xi_i (1 + 9\eta \eta_i) (1 - \eta^2)$$

$$+ \frac{81}{256} (1 + \eta \eta_i) (1 - \xi_i^2) (9\xi_i - 2\xi - 27\xi_i \xi^2) \quad (18)$$

$$\frac{\partial N_i}{\partial \eta} = \frac{1}{256} (1 + \xi \xi_i) [-10 + 9(\xi_i^2 + \eta_i^2)] (-10\eta_i + 9\eta_i \xi^2 + 18\eta + 27\eta_i \eta^2)$$

$$+ \frac{81}{256} (1 + \xi \xi_i) (1 - \eta_i^2) (9\eta_i - 2\eta - 27\eta_i \eta^2)$$

$$+ \frac{81}{256} (1 - \xi_i^2) \eta_i (1 + 9\xi \xi_i) (1 - \xi^2)$$

Substituting for nodal values and collecting terms, using MACSYMA, (ref. 9) equations (17) become

$$\frac{\partial u}{\partial \xi} = a_0 + a_1(1 + \xi) + a_2(1 + \xi)^2$$

$$\frac{\partial u}{\partial \eta} = b_0 + b_1(1 + \xi) + b_2(1 + \xi)^2 + b_3(1 + \xi)^3 \quad (19)$$

where

$$\begin{aligned} a_0 = & \frac{1}{32} [9(-3u_{12} + 3u_{11} - u_{10} + u_7 - 3u_6 + 3u_5 - u_4 + u_1)\eta^3 \\ & + 9(u_{12} + u_{11} - u_{10} + u_7 - u_6 - u_5 + u_4 - u_1)\eta^2 \\ & + (27u_{12} - 27u_{11} - 35u_{10} + 9u_9 - 9u_8 - u_7 \\ & + 27u_6 - 27u_5 + u_4 + 9u_3 - 9u_2 + 35u_1)\eta \\ & + (-9u_{12} - 9u_{11} - 35u_{10} + 9u_9 - 9u_8 - u_7 + 9u_6 \\ & + 9u_5 - u_4 - 9u_3 + 9u_2 - 35u_1)] \end{aligned} \quad (20)$$

$$a_1 = \frac{9}{8} [(2u_{10} - 5u_9 + 4u_8 - u_7)(1 + \eta) - (u_4 - 4u_3 + 5u_2 - 2u_1)(1 - \eta)]$$

$$a_2 = -\frac{27}{32} [(u_{10} - 3u_9 + 3u_8 - u_7)(1 + \eta) - (u_4 - 3u_3 + 3u_2 - u_1)(1 - \eta)]$$

$$\begin{aligned} b_0 = & \frac{1}{16} [27(3u_{12} - 3u_{11} + u_{10} - u_1)\eta^2 - 18(u_{12} + u_{11} - u_{10} - u_1)\eta \\ & + u_1 - 27u_{12} + 27u_{11} - u_{10}] \end{aligned} \quad (21)$$

$$\begin{aligned} b_1 = & -\frac{1}{32} [27(3u_{12} - 3u_{11} + u_{10} - u_7 + 3u_6 - 3u_5 + u_4 - u_1)\eta^3 \\ & - 18(u_{12} + u_{11} - u_{10} + u_7 - u_6 - u_5 + u_4 - u_1)\eta^2 \\ & + (-27u_{12} + 27u_{11} + 35u_{10} - 72u_9 + 36u_8 + u_7 \\ & - 27u_6 + 27u_5 - u_4 - 36u_3 + 72u_2 - 35u_1)] \end{aligned} \quad (22)$$

$$b_2 = \frac{9}{16} (2u_{10} - 5u_9 + 4u_8 - u_7 + u_4 - 4u_3 + 5u_2 - 2u_1)$$

$$b_3 = -\frac{9}{32} (u_{10} - 3u_9 + 3u_8 - u_7 + u_4 - 3u_3 + 3u_2 - u_1)$$

The derivatives $\partial v/\partial \xi$, $\partial v/\partial \eta$ are the same except for replacing u_i by v_i . The derivatives of u with respect to x, y are obtained from

$$\begin{aligned} \frac{\partial u}{\partial x} &= \frac{\partial u}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial u}{\partial \eta} \frac{\partial \eta}{\partial x} \\ &= -\frac{4\eta b_0}{h(1+\xi)^2} + \frac{2a_0 - 4\eta b_1}{h(1+\xi)} + \frac{1}{h} (2a_1 - 4\eta b_2) + \frac{1}{h} (1+\xi) (2a_2 - 4\eta b_3) \end{aligned} \quad (23)$$

$$\begin{aligned} \frac{\partial u}{\partial y} &= \frac{\partial u}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial u}{\partial \eta} \frac{\partial \eta}{\partial y} \\ &= \frac{4b_0}{\ell(1+\xi)^2} + \frac{4b_1}{\ell(1+\xi)} + \frac{4b_2}{\ell} + \frac{4b_3}{\ell} (1+\xi) \end{aligned} \quad (24)$$

Similar expressions for $\partial v/\partial x$ and $\partial v/\partial y$ with u_i replaced by v_i in a's and b's.

The stresses and strains are singular when the Jacobian determinant vanishes at $\xi = -1$. From (23), (24) and (12), the singularity is $0(1/r)$ if $b_0 \neq 0$ and is $0(1/\sqrt{r})$ if $b_0 = 0$. A careful study of (21) we see that b_0 depends on the displacements of nodal points at the crack tip. If the nodal points at the crack tip are tied together, i.e.,

$$u_1 = u_{10} = u_{11} = u_{12} \quad \text{and} \quad v_1 = v_{10} = v_{11} = v_{12} \quad (25)$$

then $b_0 = 0$ and the strain field has the inverse square root of r singularity, the correct singularity of linear fracture mechanics. On the other hand if the nodal points at the crack tip are allowed to move independently to one another, the strain field has the $(1/r)$ singularity, a characteristic of perfectly plasticity.

NASTRAN IMPLEMENTATION

The NASTRAN implementation for the quadratic element follows the steps outlined in section 6.8.3.12 of reference 12. The following routines require modification: PLA1, which creates the ECPT's and EST's for the linear and non-linear elements; PLA31 and PLA32, which recover stresses for the non-linear

elements; and PLAYBD, PLA41 and PLA42 which control generation of the updated stiffness matrix. The following new routines are required: PSDUM1, a driver for stress data recovery in PLA3; PSDM11 and PSDM12, phase I and II stress recovery routines; PKDUM1, a driver for stiffness generation for the non-linear elements; PKDM11 and PKDM12, stress recovery routines which generate stresses for the computation of the non-linear material matrix; and PKDM1S, the stiffness matrix generation routine for non-linear elements. The two driver routines, PSDUM1 and PKDUM1 can be modelled after the corresponding routines for the QUAD1 element. The remaining routines are modifications of the stiffness and stress recovery routines (ref. 7) required for rigid format 1, statics. The major modifications to switch from statics to piecewise linear include changing the labelled common areas, building the non-linear material matrix (ref. 10) and calculating incremental stress rather than total stresses.

NUMERICAL EXAMPLE

Consider the problem of small scale yielding. The problem is governed by elastic field at points far away from the crack tip and asymptotically has the elastic singular field. Near the crack tip we have plastic zone. This is schematically represented in figure 2. The plane strain slip line field is also shown.

The problem is modelled in a fashion similar to Barsoum (ref. 5). The geometry is shown in figure 3. The crack tip elements, 1-12, are the singular elements which can either be quadratic or cubic elements. For the symmetric case the corner nodes of the elements are placed on concentric semi-circles, $0 < \theta < \pi$, at $\pi/12$ intervals, of radii, $r = 0, .5, 1.0, 1.625, 1.5^2, 2^2, 2.5^2, 3^2, 4^2, 5.5^2$.

The method of solution, for the plastic problem, is based on Swedlow's piece-wise linear analysis and is well documented in the NASTRAN theoretical manual (ref. 10).

The procedure for the present problem is accomplished via two rigid formats. The static rigid format is first used to obtain the stress distribution and the equivalent stresses at the integration point ($\xi = \eta = 0$) for the elastic increment. This solution is performed with all the collapsed nodes at the crack tip having the same displacement vector (see equation 25). This is accomplished with multipoint constraints. The outer most nodes are subjected to the displacements governed by Westergaard solution, with $K = 1$,

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \frac{K}{2G} \left(\frac{r}{2\pi}\right)^{1/2} \left(\frac{3-\nu}{1+\nu} - \cos \theta\right) \begin{Bmatrix} \cos \theta/2 \\ \sin \theta/2 \end{Bmatrix} \quad (26)$$

where $E = 30 \times 10^6$ psi and $\nu = .3$. The value of $2K_0$ is established from the elastic solution based on the yield stress (σ_0) of 20×10^3 psi for the highest stressed element. For the plastic analysis the stress-strain curve is provided with above constants and yield strain at .2% and hardening exponent $n \approx .3$

(this should be close to zero for perfect plasticity). The nodes at the crack tip are then released for sliding in order to obtain $1/r$ singularity at the crack tip. The load is incremented by $K_0/4$ till the plastic zone has reached first layer of elements.

Preliminary results of the problem are indicated in figures 4a-c and compared with those of (ref. 11). From the static solution it was found that the inception of yielding occurs at $\theta = 68^\circ$ compared to the theoretical value of $\theta = 70^\circ$.

In figures 4a-4c we have also plotted the slip line (plane strain) solution for comparison. The plastic zone also corresponds well with (ref. 11).

CONCLUSION

Higher order isoparametric elements can be effectively used for modelling singular elastic as well plastic problems that arise in the field of fracture mechanics. The procedure in obtaining these do not require any special crack tip elements but are simply constructed by adjusting the adjacent nodes at proper locations and proper constraints. The locations of these nodes should be adhered to as closely as possible for stable answers. Since many general purpose finite elements may have these elements in their library the method, for crack problems, may be accessible to many users.

REFERENCES

1. Irwin, G. R.: Fracture Testing of High-Strength Sheet Materials Under Conditions Appropriate For Stress Analysis. Naval Research Laboratory Rpt. 5486, July 1960.
2. Wells, A. A.: Unstable Crack Propagation in Metals. Proc. Conf. Crack Propagation, Cranfield, England, 1960, p. 120.
3. Rice, J. R.: A Path Independent Integral and the Approximate Analysis of Strain Concentration by Notches and Cracks. Trans. Am. Soc. Mech. Engrs., Journal of Applied Mechanics, 1968, p. 379.
4. Eftis, J.; Jones, D. L.; and Liebowitz, H.: On Fracture Toughness in the Nonlinear Range. Engrg. Fract. Mech., Vol. 7, 1975, p. 491.
5. Barsoum, R. S.: Triangular Quarter-Point Elements as Elastic and Perfectly-Plastic Crack Tip Element. Int. J. Num. Meth. Engrg., Vol. 11, 1977, p. 85.
6. Rice, J. R. and Rosengren, G. F.: Plane Strain Deformation Near A Crack Tip In A Power-Law Hardening Material. J. Mech. Phys. Solids, Vol. 16, p. 1.
7. Hussain, M. A.; Lorensen, W. E.; and Pflegl, G.: The Quarter-Point Quadratic Isoparametric Element As A Singular Element For Crack Problems. NASTRAN Users' Experiences, NASA TM-X-3428, Oct. 1976, p. 419.
8. Zienkiewicz, C. O.: The Finite Element Method In Engineering Science. McGraw Hill, London, 1971.
9. MACSYMA: Math Lab Group, MIT Laboratory for Computer Science (Symbolic Manipulation System), Nov. 1975.
10. The NASTRAN Theoretical Manual. Editor, MacNeal, R. H., NASA SP-221, Sept. 1970, p. 104.
11. Hutchinson, J. W.: Singular Behavior At The End Of A Tensile Crack In A Hardening Material. J. Mech. Phys. Solids, 1968, Vol. 16, p. 13.
12. The NASTRAN Programmer's Manual. NASA SP-223(01), Sept., 1972.

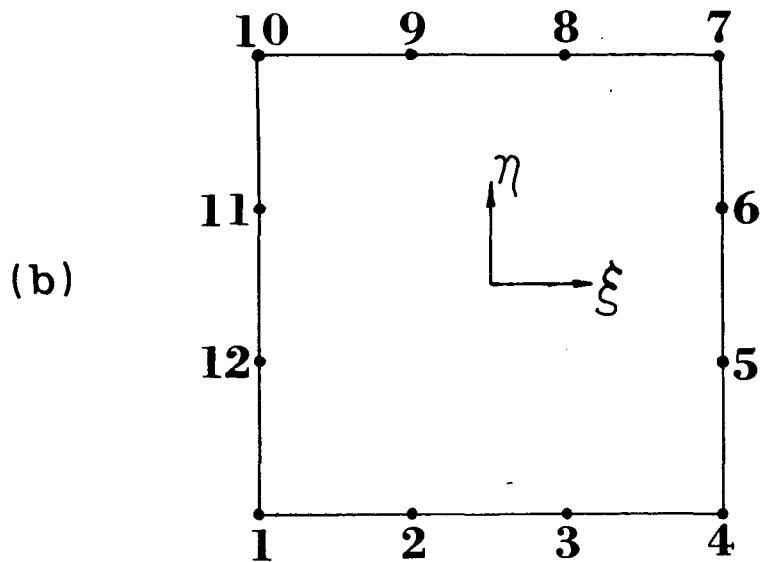
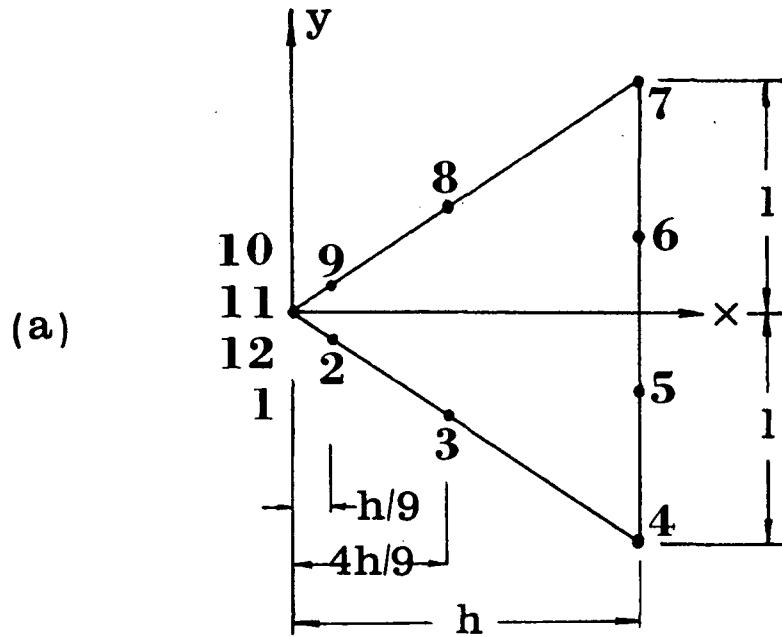


FIGURE 1. (a) 12-NODE CUBIC ELEMENT COLLAPSED TO FORM A SINGULAR ELEMENT;
 (b) THE PARENT ELEMENT.

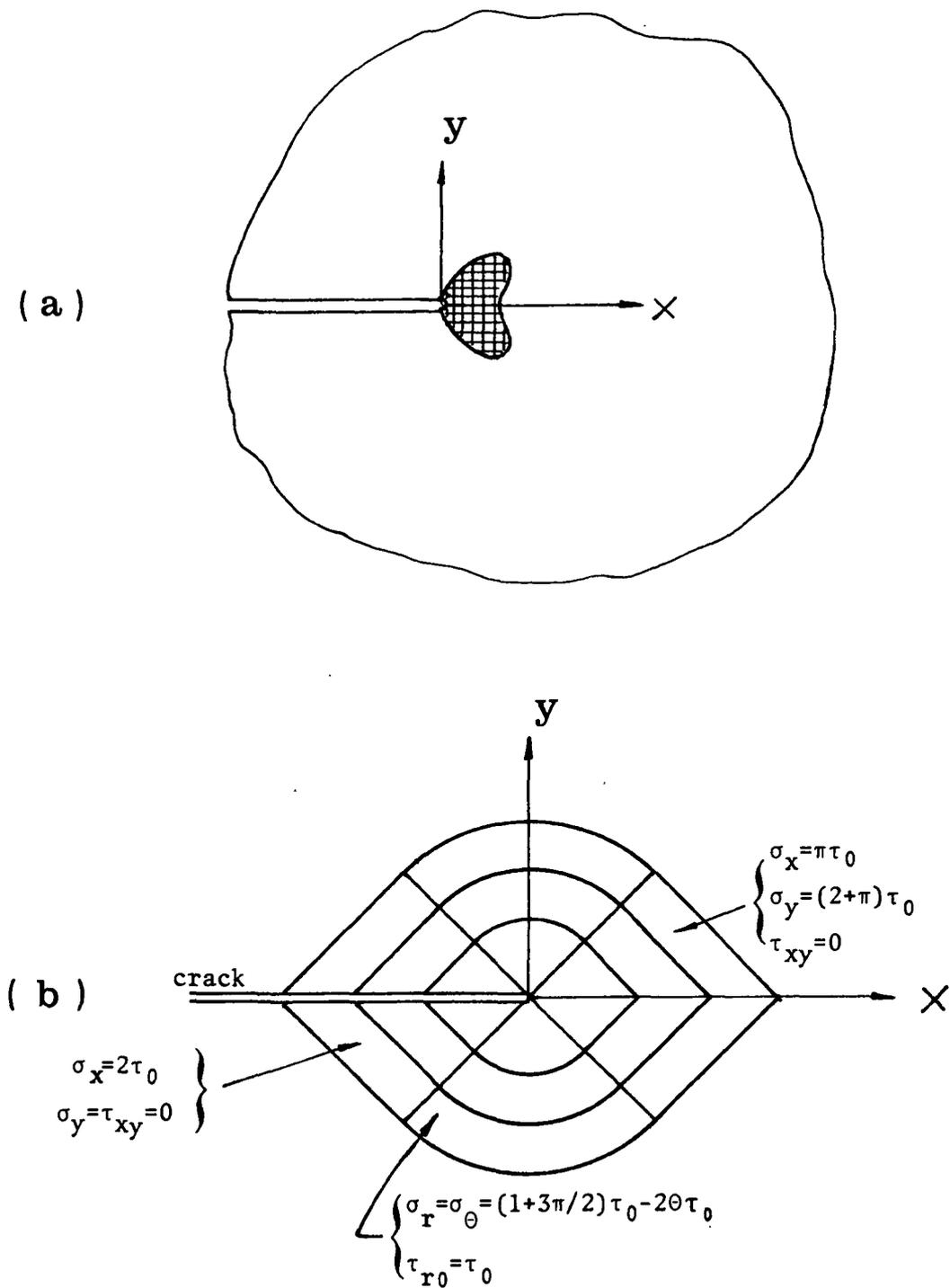


FIGURE 2. (a) SMALL-SCALE YIELDING NEAR A SEMI-INFINITE CRACK;
 (b) PERFECTLY PLASTIC PLANE STRAIN SLIP LINE FIELD AT THE CRACK TIP.

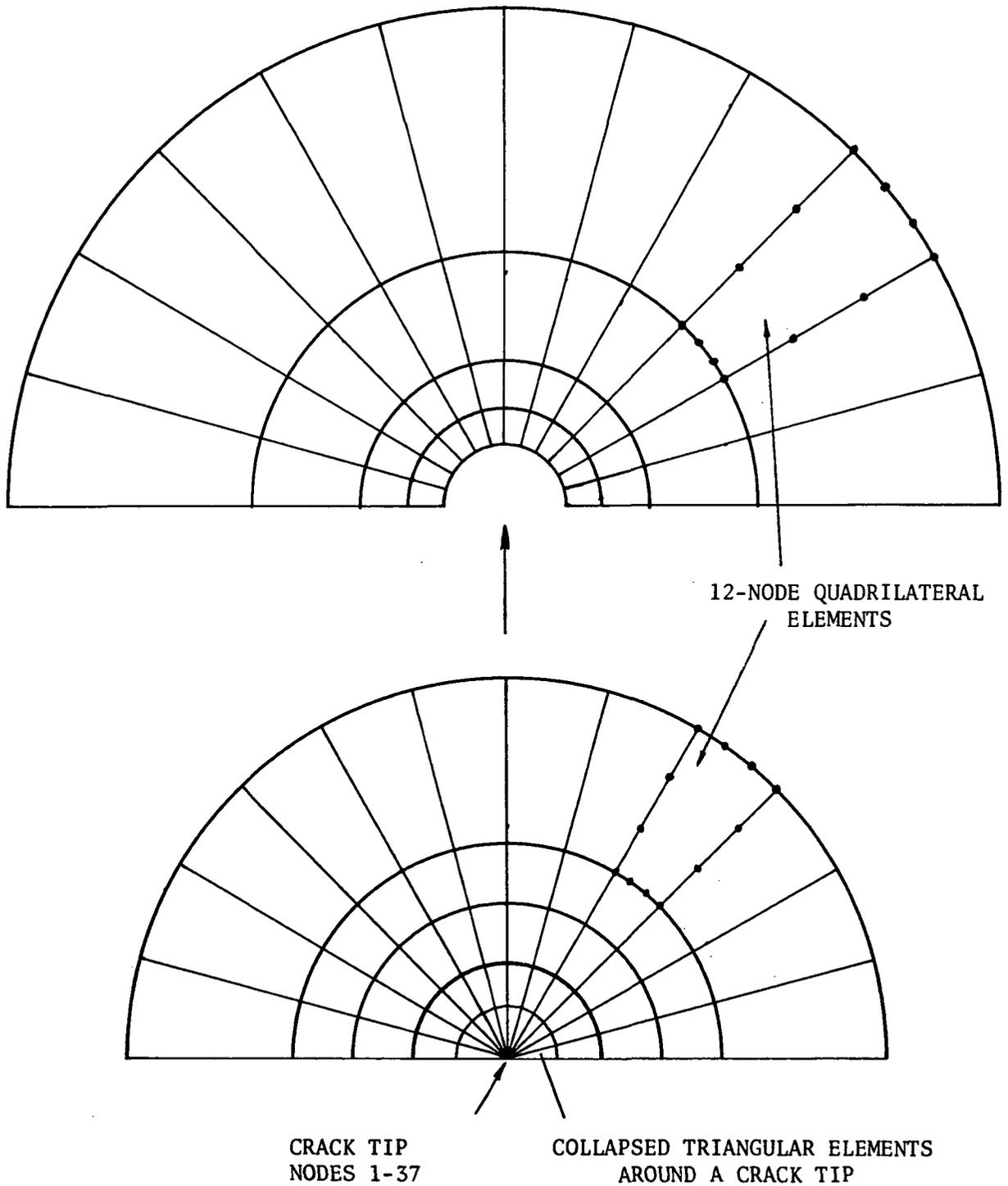


FIGURE 3. FINITE ELEMENT IDEALIZATION OF THE CRACK TIP NEAR FIELD.

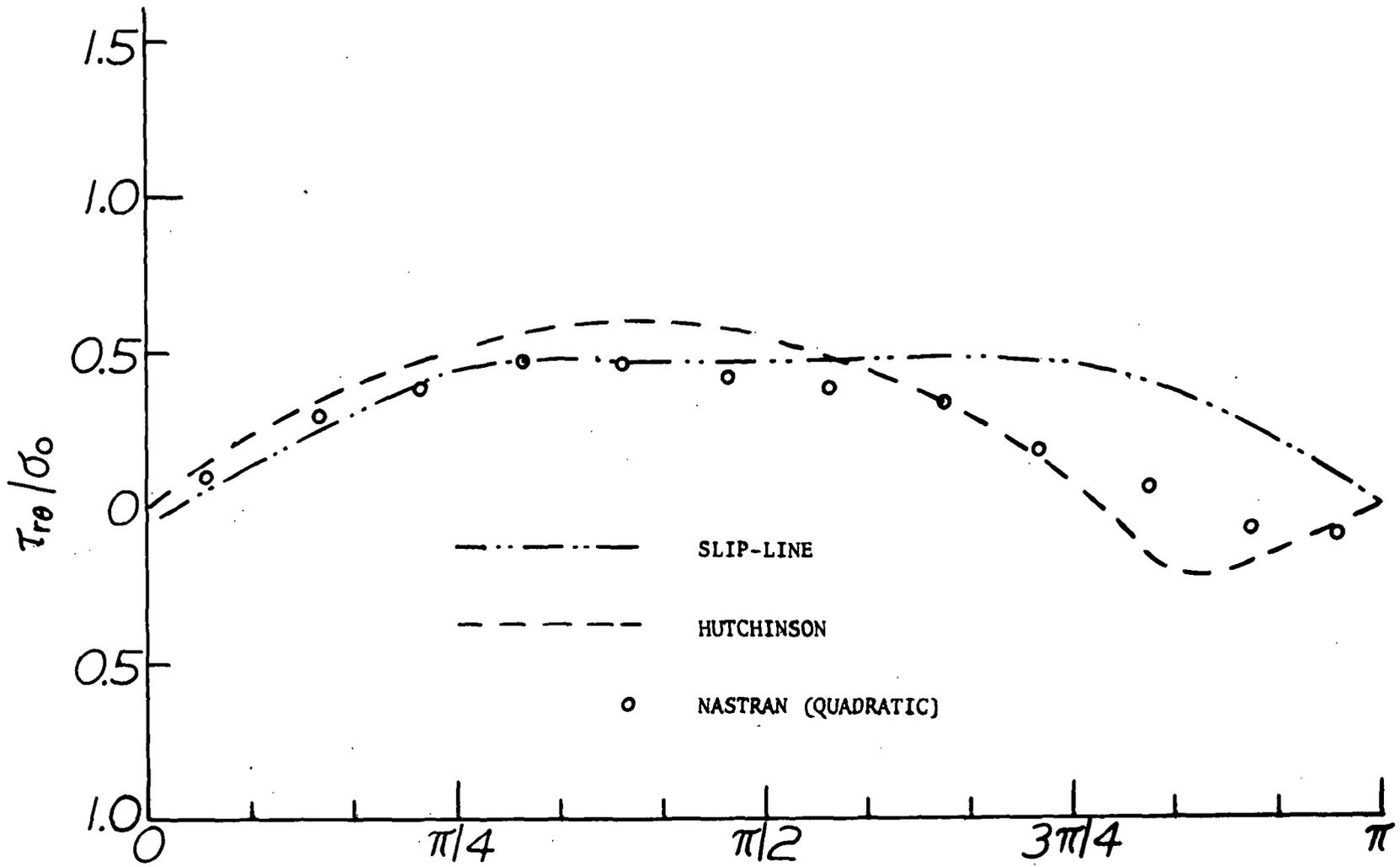


FIGURE 4. (a) SHEARING STRESS DISTRIBUTION FOR THE SINGULAR ELEMENTS COMPARED TO REF. (11) AND SLIP LINE SOLUTION. (PRELIMINARY RESULTS)

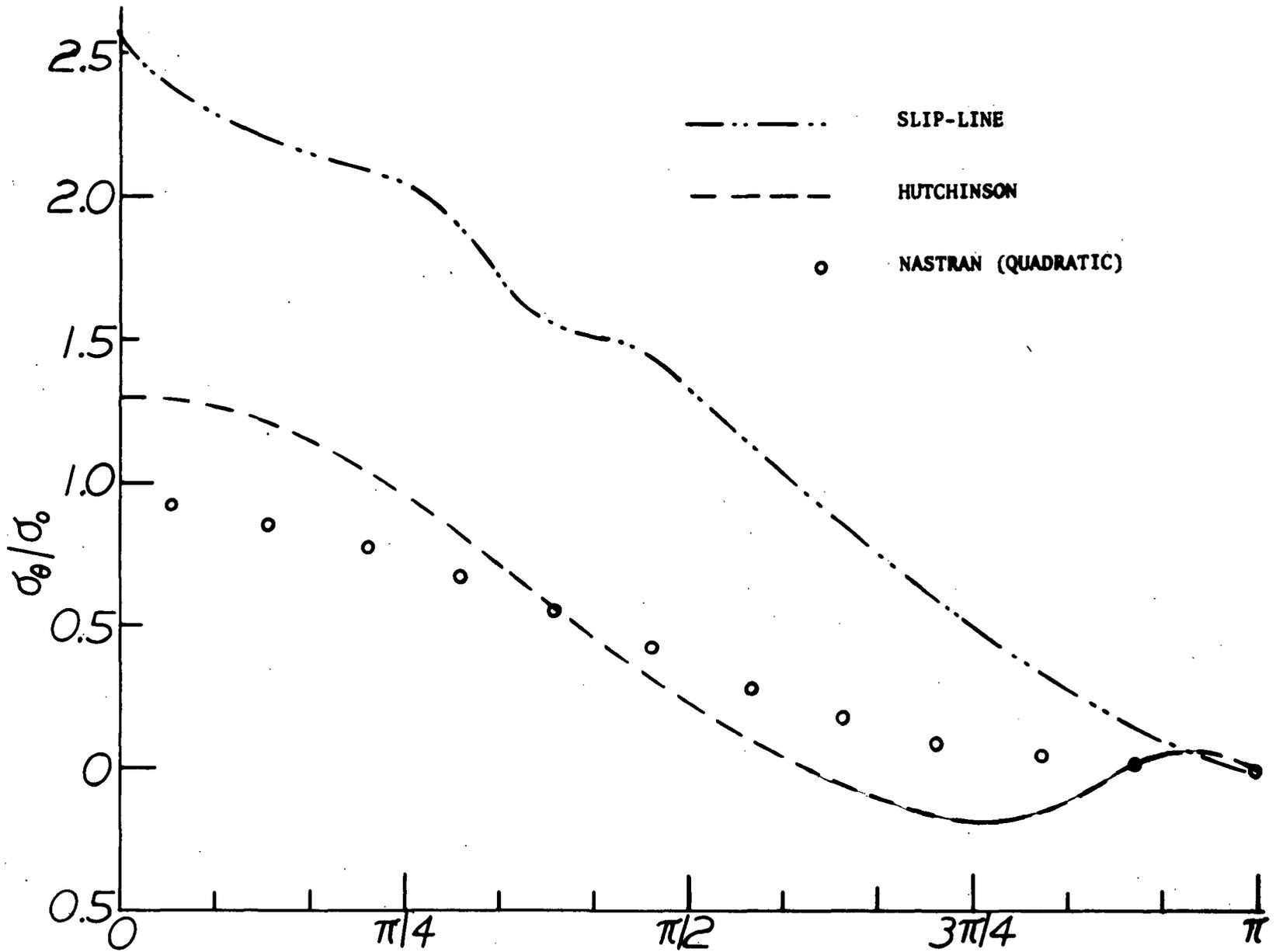


FIGURE 4. (b) TANGENTIAL STRESS DISTRIBUTION FOR THE SINGULAR ELEMENTS COMPARED TO REF. (11) AND SLIP LINE SOLUTION (PRELIMINARY RESULTS).

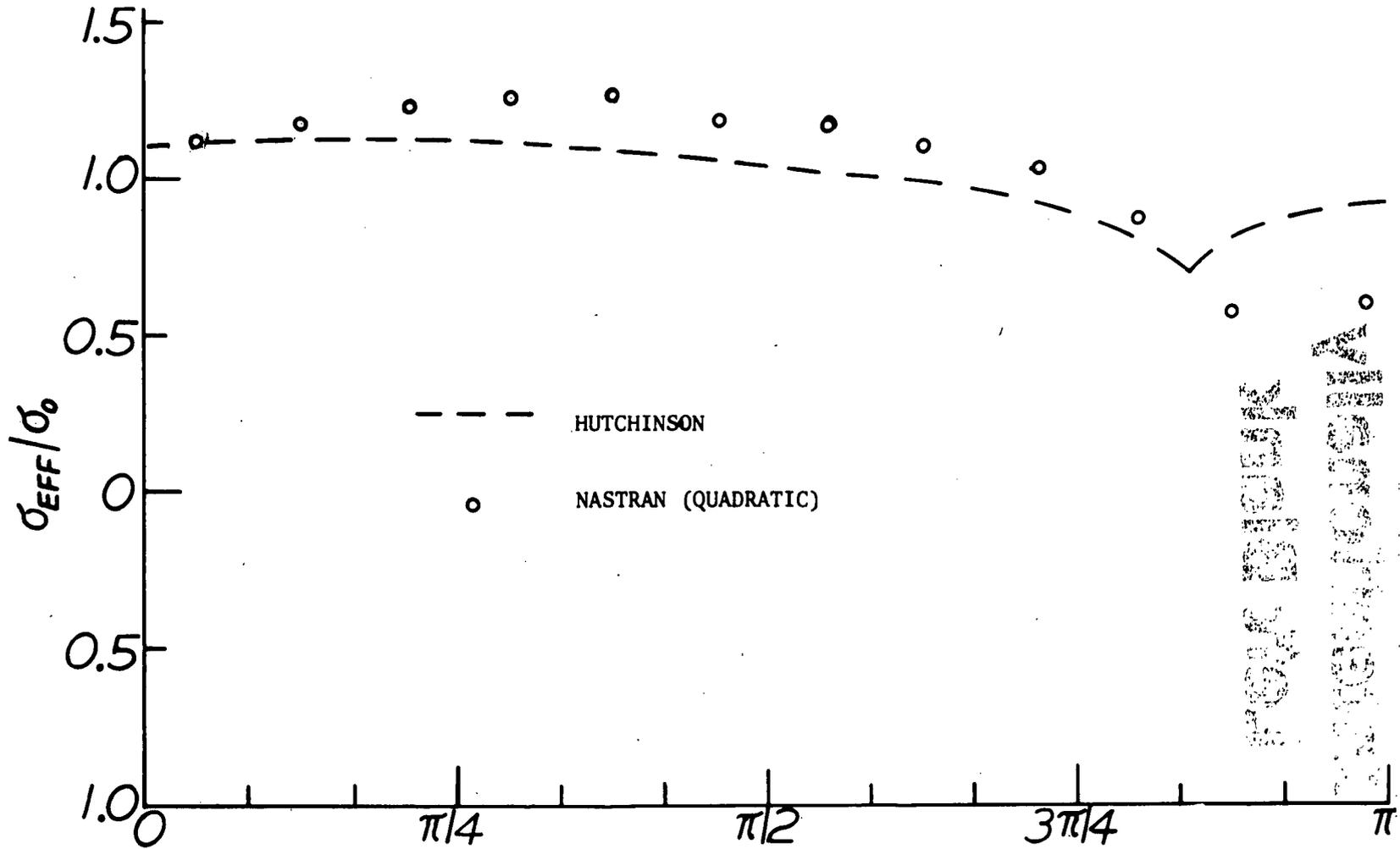


FIGURE 4. (c) EFFECTIVE STRESS DISTRIBUTION FOR THE SINGULAR ELEMENTS COMPARED TO REF. (11) AND SLIP LINE SOLUTION. (PRELIMINARY RESULTS)

**Page
Intentionally
Left Blank**

APPLICATION OF THE TRPLT1 ELEMENT TO
LARGE AMPLITUDE FREE VIBRATIONS OF PLATES

Chuh Mei*
Vought Corporation, Hampton, Va.

and

James L. Rogers, Jr.
NASA Langley Research Center, Hampton, Va.

SUMMARY

A finite element formulation is developed for analyzing large amplitude free flexural vibrations of thin plates in NASTRAN. Stress distributions in the plate, in addition to deflection shapes and nonlinear frequencies are determined. Linearized equations of motion governing large amplitude oscillations of plates and a linearized geometrical stiffness matrix are presented. The solution procedure and convergence characteristics are discussed. The quasi-linear geometrical stiffness matrix for an eighteen degree-of-freedom higher order triangular plate element is evaluated by using a seven-point numerical integration. Nonlinear frequencies for square, rectangular, rhombic, and isosceles triangular plates, with edges simply supported or clamped, are compared with earlier solutions. The present formulation is found to give results entirely adequate for engineering purposes.

NOMENCLATURE

[A]	matrix relating curvatures and coefficients of transverse displacement
{a}	vector of coefficients of transverse displacement polynomial
a_1, a_2, \dots, a_{20}	coefficients of polynomial for transverse displacement
a, b	plate dimensions
[C]	matrix relating membrane forces and strains
c	amplitude of vibration, $(w)_{\max}$
D	bending rigidity

*Presently at the Department of Engineering Mechanics, University of Missouri-Rolla.

[D]	matrix relating bending moments and curvatures
E	Young's modulus
[F]	matrix of linearizing functions
f_1, f_2	linearizing functions
h	plate thickness
[K]	system stiffness matrix
[k]	element stiffness matrix
[K^g]	linearized system geometrical stiffness matrix
[k^g]	linearized element geometrical stiffness matrix
[M]	system mass matrix
[m]	element mass matrix
{N}	membrane forces
n	iteration cycle
{q}	eigenvector or plate deflection
[T]	matrix relating coefficients of transverse displacement and element nodal displacements
[T_1]	matrix relating element nodal displacements and coefficients of transverse displacement
U	strain energy
u, v	inplane displacements
w	lateral displacement
x, y, z	local or element coordinates
{ δ }	element nodal displacements
{ ϵ }	membrane strains
$ \epsilon $	norm
λ	linear frequency parameter
ν	Poisson's ratio

mass density
 vector of nondimensional stresses
 vector of bending curvatures
 linear (small deflection) frequency
 nonlinear (large deflection) frequency

INTRODUCTION

The finite element analysis of nonlinear structural problems in solid mechanics has received considerable attention in recent years. Application of finite elements to large amplitude vibrations of plates was first reported in Ref. 1. The approximation used was based on a modified form of the Berger's formulation (ref. 2). Instead of the sum of stress resultants ($N_x + N_y$) being assumed constant over the complete plate in Refs. 3 and 4; N_x and N_y are assumed constant within each element in Ref. 1. A geometric stiffness matrix dependent on the membrane stresses N_x , N_y and N_{xy} is derived explicitly for a rectangular plate element. Fundamental nonlinear frequencies for rectangular plates with various edge conditions were found to agree with classical solutions of Refs. 3-6. Recently, Rao (refs. 7, 8) et al. presented a simplified formulation which eliminates the inplane displacements from the strain-displacement relations for plates with immovable edges. In addition, an appropriate quasi-linearization of the strain-displacement relations was introduced. This process leads to linear equations of motion containing two unknown functions which are solved by an iterative process used in Ref. 9 for nonlinear beam vibration problems. Results obtained for circular (ref. 7) and rectangular (ref. 8) plates are in good agreement with classical solutions.

In the present paper, a triangular plate element is developed to investigate large amplitude free vibrations of thin plates of arbitrary shape. The formulation of the linearized geometrical stiffness matrix follows Refs. 7 and 8. The solution procedure and convergence characteristics are discussed. Examples treated include plates of rectangular, rhombic, and isosceles triangular shapes with simply supported and/or clamped edges. Comparisons are made with linear (small deflection) frequencies and available fundamental nonlinear (large deflection) frequencies. Nonlinear vibration capability for thin plates has been developed for use with NASTRAN Level 16.0 by means of DMAP alters and addition of two subroutines to the NASTRAN code. The DMAP alter sequence is given in the Appendix.

FORMULATION

The nonlinear strain-displacement relations for a thin elastic plate are given by (ref. 10)

$$\{\epsilon\} = \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_{xy} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial u}{\partial x} + \frac{1}{2} \left(\frac{\partial w}{\partial x} \right)^2 \\ \frac{\partial v}{\partial y} + \frac{1}{2} \left(\frac{\partial w}{\partial y} \right)^2 \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} + \frac{\partial w}{\partial x} \frac{\partial w}{\partial y} \end{Bmatrix} \quad (1)$$

$$\{\chi\} = \begin{Bmatrix} \chi_x \\ \chi_y \\ \chi_{xy} \end{Bmatrix} = \begin{Bmatrix} - \frac{\partial^2 w}{\partial x^2} \\ - \frac{\partial^2 w}{\partial y^2} \\ - 2 \frac{\partial^2 w}{\partial x \partial y} \end{Bmatrix} \quad (2)$$

where $\{\epsilon\}$ is the vector of membrane strains, $\{\chi\}$ is the vector of bending and twist curvatures, u and v are the displacements in the x and y directions, and w is the lateral deflection.

Membrane forces and bending moments are related to strains and curvatures by

$$\{N\} = \begin{Bmatrix} N_x \\ N_y \\ N_{xy} \end{Bmatrix} = [C]\{\epsilon\} \quad (3)$$

$$\{M\} = \begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = [D]\{\chi\} \quad (4)$$

where $[C]$ and $[D]$ are symmetric matrices of elastic constants. For an isotropic plate of uniform thickness h

$$[C] = \frac{Eh}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (5)$$

and

$$[D] = D \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (6)$$

in which E and ν are the Young's modulus and Poisson's ratio, respectively, and $D = Eh^3/12(1-\nu^2)$.

By linearizing the strain-displacement relations and assuming immovable edges (refs. 7,8), Eq. (1) becomes

$$\{\epsilon\} = \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \epsilon_{xy} \end{Bmatrix} = \begin{Bmatrix} f_1 \frac{\partial w}{\partial x} \\ f_2 \frac{\partial w}{\partial y} \\ f_1 \frac{\partial w}{\partial y} + f_2 \frac{\partial w}{\partial x} \end{Bmatrix} \quad (7)$$

where f_1 and f_2 are the linearizing functions defined as

$$\{f\} = \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = \begin{Bmatrix} \frac{1}{2} \frac{\partial w}{\partial x} \\ \frac{1}{2} \frac{\partial w}{\partial y} \end{Bmatrix} \quad (8)$$

The strain energy for a plate element is

$$U = \frac{1}{2} \iint (\{M\}^T \{\chi\} + \{N\}^T \{\epsilon\}) dx dy \quad (9)$$

where the integration is taken over the area of the element. Substituting Eqs. (3) and (4) into Eq. (9), yields

$$U = \frac{1}{2} \iint \{\chi\}^T [D] \{\chi\} dx dy + \frac{1}{2} \iint \{\epsilon\}^T [C] \{\epsilon\} dx dy \quad (10)$$

The first integral in Eq. (10) represents the strain energy due to the bending and twisting of the plate midsurface, and the second integral represents the strain energy due to midsurface stretching induced by lateral deflections.

Consider a triangular plate element in the local coordinate system as shown in Fig. 1. The lateral deflection within the element is taken as (refs. 11 and 12)

$$\begin{aligned}
 w(x,y) = & a_1 + a_2x + a_3y + a_4x^2 + a_5xy + a_6y^2 + \\
 & a_7x^3 + a_8x^2y + a_9xy^2 + a_{10}y^3 + \\
 & a_{11}x^4 + a_{12}x^3y + a_{13}x^2y^2 + a_{14}xy^3 + a_{15}y^4 + \\
 & a_{16}x^5 + a_{17}x^3y^2 + a_{18}x^2y^3 + a_{19}xy^4 + a_{20}y^5
 \end{aligned} \tag{11}$$

The element has 18 degrees-of-freedom: namely, deflection w and its first derivatives at each of the three corner and three midside nodes. They are represented by the vector $\{\delta\}$ defined by

$$\{\delta\}^T = \underline{[w_1, w_{x1}, w_{y1}, w_2, \dots, w_{y5}, w_6, w_{x6}, w_{y6}]} \tag{12}$$

The relationship between the nodal displacements and the generalized coordinates is written as

$$\begin{Bmatrix} \{\delta\} \\ 0 \\ 0 \end{Bmatrix} = [T_1] \{a\} \tag{13}$$

where $\{a\}$ is the vector of coefficients in Eq. (11)

$$\{a\}^T = \underline{[a_1, a_2, \dots, a_{20}]} \tag{14}$$

It is noted that $[T_1]$, a 20 x 20 matrix, is nonsingular for all practical cases. Thus

$$\{a\} = [T_1]^{-1} \begin{Bmatrix} \{\delta\} \\ 0 \\ 0 \end{Bmatrix} \tag{15}$$

which is equivalent to

$$\{a\} = [T]\{\delta\} \tag{16}$$

where the 20 x 18 matrix [T] consists of the first eighteen columns of $[T_1]^{-1}$. For details of matrix $[T_1]$, the reader is referred to Ref. 11.

Strains and curvatures may be evaluated from Eq. (11) as

$$\{\chi\} = \left\{ \begin{array}{c} - \frac{\partial^2 w}{\partial x^2} \\ - \frac{\partial^2 w}{\partial y^2} \\ - 2 \frac{\partial^2 w}{\partial x \partial y} \end{array} \right\} = [A]\{a\} \tag{17}$$

$$\{\epsilon\} = \left\{ \begin{array}{c} f_1 \frac{\partial w}{\partial x} \\ f_2 \frac{\partial w}{\partial y} \\ f_1 \frac{\partial w}{\partial y} + f_2 \frac{\partial w}{\partial x} \end{array} \right\} = [F] \left\{ \begin{array}{c} \frac{\partial w}{\partial x} \\ \frac{\partial w}{\partial y} \end{array} \right\} = [F][B]\{a\} \tag{18}$$

where

$$[F] = \begin{bmatrix} f_1 & 0 \\ 0 & f_2 \\ f_2 & f_1 \end{bmatrix} \tag{19}$$

and [A] and [B] are obtained by the appropriate differentiation of Eq. (11).

Substituting Eqs. (17) and (18) into Eq. (10), then gives

$$U = \frac{1}{2} \{a\}^T ([k_a] + [k_a^g])\{a\} \tag{20}$$

where

$$[k_a] = \iint [A]^T [D] [A] dx dy \tag{21}$$

$$[k_a^g] = \iint [B]^T [F]^T [C] [F] [B] dx dy \tag{22}$$

The generalized stiffness matrix $[k_a]$ has been incorporated in NASTRAN in Ref. 12 as TRPLT1 element. The calculation of the geometrical stiffness matrix $[k^g]$ is the subject of the present paper. Evaluation of $[k^g]$ is based on numerical integration using a seven-point integration scheme (ref. 13) which can exactly integrate functions up to and including quintic order.

The element stiffness and linearized geometrical stiffness matrices in the local coordinate system, by virtue of Eq. (16), are

$$[k] = [T]^T [k_a] [T] \quad (23)$$

and

$$[k^g] = [T]^T [k_a^g] [T] \quad (24)$$

Following Refs. 7 and 8, the matrices $[k]$ and $[k^g]$ are combined to form a single linearized stiffness matrix. The element consistent mass matrix is given in Ref. 11 and has been incorporated in NASTRAN as reported in Ref. 12. By assembling the finite elements, and applying the kinematic boundary conditions, the linearized equations of motion governing large amplitude vibrations of plate may be written as

$$\omega^2 [M] \{q\} = [[K] + [k^g]] \{q\} \quad (25)$$

SOLUTION OF EQUATIONS

Eq. (25) is a standard symmetric eigenvalue problem. However, the functions f_1 and f_2 which are needed for $[k^g]$ are not known a priori. Consequently, an iterative process is adopted to solve Eq. (25). To start with, the linearizing functions are assumed to be zero, and Eq. (25) is solved for the linear (small deflection) frequency ω_L and the corresponding mode shape $\{q\}_L$. Once the linear mode shape is known, it is scaled up corresponding to a prescribed nondimensional amplitude of oscillation (c/h). The element displacements $\{\delta\}$ can then be obtained, and the linearizing functions are evaluated from the expression

$$\begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = \frac{1}{2} [B][T]\{\delta\} \quad (26)$$

Once the linearizing functions are known, the linearized geometrical stiffness matrix, Eqs. (22) and (24), is evaluated by the seven-point numerical integration. Eq. (25) is then solved to obtain the nonlinear (large deflection) frequency ω and corresponding deflection shape $\{q\}$. Again the deflection shape $\{q\}$ is scaled up corresponding to the prescribed amplitude (c/h) . This iterative procedure is repeated until a certain convergence test is satisfied.

The maximum displacement norm criterion used by Bergan and Clough (ref. 14) for large deflection static analysis is employed in this dynamic problem. Also a frequency norm, introduced in Ref. 15 and used in the present study, is defined by

$$||\epsilon||_F = \left| \frac{\Delta\omega}{\omega} \right| \quad (27)$$

where $\Delta\omega$ is simply the change in frequency during the n-th iteration cycle.

A typical plot of the displacement and frequency norms versus number of iterations for a simply supported square plate (3 x 3 gridwork in 1/8 of plate) with amplitude $c/h = 1.0$ is shown in Fig. 2. Both norms exhibit the important characteristics of straightness and parallelism as described in Ref. 14. Therefore, an upper bound or maximum error on displacement and frequency convergence can be estimated, for details the reader is referred to Ref. 14. The numerical examples presented in the following section, convergence is considered achieved whenever either one of the norms reaches a value of 10^{-4} .

The principal stress in the plate and the frequency ω constitute important information for designing plate structures. Once the nonlinear frequency ω and deflection slope $\{q\}$ for a given amplitude (c/h) are determined, nondimensional stresses can be obtained from the equations

$$\{\sigma\} = \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \frac{a^2}{Eh^2} \left(\frac{z}{I} \{M\} + \frac{1}{h} \{N\} \right) \quad (28)$$

where a is the dimension of a rectangular plate parallel to the x-direction. I is the area moment of inertia per unit length. The stresses are evaluated at the extreme fibers of the plate, $z = \pm h/2$. The bending moments and membrane forces in Eq. (28) are computed as

$$\{M\} = [D][A][T]\{\delta\} \quad (29)$$

and

$$\{N\} = [C][F][B][T]\{\delta\} \quad (30)$$

The stresses are evaluated at the three corner nodes and also at the centroid of the element. The principal stresses and the maximum shear stress are

$$\left. \begin{aligned} \sigma_{1,2} &= \frac{\sigma_x + \sigma_y}{2} + \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2} \\ \theta &= \frac{1}{2} \arctan \left(\frac{2 \tau_{xy}}{\sigma_x - \sigma_y} \right) \end{aligned} \right\} \quad (31)$$

and

$$\tau_{\max} = \sqrt{\left(\frac{\sigma_x - \sigma_y}{2}\right)^2 + \tau_{xy}^2}$$

MODIFICATIONS TO THE NASTRAN LEVEL 16

Subroutines DTSHLN and PDW were created to compute the geometrical stiffness matrix $[k_g]$ of Eq. (24). Subroutine PDW is used to calculate the $[F]$ matrix in Eqs. (19) and (26). The core requirements are 14,404₈ locations for DTSHLN and 93₈ for PDW.

To exercise the nonlinear vibration option in NASTRAN, it is necessary to modify the calling sequence for module DSMG1 as follows:

```
DSMG1      CASECC,,SIL,EDT,PHIG,CSTM,MPT,
           ECPT,GPCT,DIT/KDGG/V,N,DSCSET/C,N,1 $
```

The two new subroutines were compiled and appended to the NASTRAN object library. Link 1 and Link 13 were relinked and a new executable NASTRAN was created. Although this procedure was done on a CDC computer, similar procedures will produce similar results on both the IBM and UNIVAC computers. In order to use the nonlinear vibration capability in NASTRAN, extensive alters have to be applied to Rigid Format 5. The appropriate DMAP alter sequence is shown in the Appendix. One additional change must be made to the Bulk Data Deck. The parameter AMP is input via a PARAM card to specify the amplitude of vibration of the structure.

RESULTS AND DISCUSSION

Using the present formulation, nonlinear vibrations of plates of rectangular, rhombic, and isosceles triangular shapes with simply supported and/or clamped edges are studied. The nondimensional linear frequency parameter

$$\lambda = \omega_L a^2 \sqrt{\frac{\rho}{D}} \quad (32)$$

and frequency ratios ω/ω_L for various amplitude ratios c/h are obtained for the fundamental mode. Additionally, frequencies of next three higher modes are obtained for the square plate with simply supported and clamped edges. A value of Poisson's ratio of 0.3 is used in all calculations.

Figure 3 shows the finite element idealizations used. Only one-quarter of the rectangular plate and one-eighth of the square plate are modelled due to symmetry. The total number of independent degrees-of-freedom (d.o.f.) and the number of d.o.f. after applying the Guyan reduction to eliminate rotational d.o.f. are given in the second and third columns of Tables 1 and 2. Note that independent d.o.f. means the difference between the total number of d.o.f. and the number of boundary constraints. Numerical results are given in Tables 1 and 2 for the fundamental frequency parameter λ and frequency ratio ω/ω_L for a simply supported rectangular, and a clamped rectangular plate at different amplitudes c/h . Good agreement with the exact fundamental frequency parameters is obtained even with the 2×2 gridwork for rectangular plate. Results for frequency ratios are also in substantial agreement with the earlier continuum solutions.

The fundamental frequency ratio ω/ω_L versus amplitude c/h for a rhombic plate is shown in Fig. 4. The frequency parameters λ obtained from the analysis are 18.7925 and 33.8541 for simply supported and clamped edges, respectively. The corresponding theoretical values from Ref. 16 are 18.844 and 34.71.

In Fig. 5, the frequency ratio for the fundamental mode is given as a function of amplitude c/h for an isosceles triangular plate with several edge restraints. No comparisons are made for the isosceles triangular and rhombic plates because no approximate solution was available in the published literature.

Nonlinear frequencies corresponding to higher modes (m,n) are obtained for a square plate with simply supported and clamped edges. In Table 3, the frequency parameters and frequency ratios are shown for higher modes up to $m = 3$ and $n = 1$ (or $m = 1$ and $n = 3$), where m and n denote the number of half-waves. A 4×4 gridwork in a quarter of plate was used obtaining the results for higher modes. No previously published data was available for comparison with the latter results.

CONCLUDING REMARKS

A geometrical stiffness matrix consistent with the higher order triangular plate element, TRPLT1, has been developed and incorporated in a non-standard version of NASTRAN level 16.0. An iterative process used to determine nonlinear frequencies has been implemented by the DMAP alters. Numerical results obtained for square and rectangular plates are in good agreement with classical solutions. Nonlinear frequencies are also obtained for plate of rhombic and isosceles triangular shapes. The implementation of the triangular plate element geometric stiffness matrix does greatly enhance the total nonlinear vibration capability of NASTRAN.

APPENDIX

DMAP ALTER SEQUENCE FOR NONLINEAR

VIBRATIONS OF PLATES

```

ID      NLVIB,TRPLT1 ELEMENT
$      DMAP ALTER SEQUENCES FOR NONLINEAR VIBRATIONS OF THIN ELASTIC
$      PLATES OF ARBITRARY SHAPE
APP     DISP
SOL     5.0
TIME    10
ALTER  22,22
GP3     GEOM3,EQEXIN,GEOM2/,GPTT/C,N,123/V,N,NOGRAV/C,N,123 $
ALTER  31
PARAM   //C,N,ADD/V,N,NOMGG/C,N,1/C,N,0 $
ALTER  71
EQUIV   MGG,MNN/MPCF1 $
CHKPNT  MNN $
ALTER  75,85
MCF2    USET,GM,KGG,MGG,,/KNN,MNN,, $
CHKPNT  KNN,MNN $
LABEL   LBL2 $
EQUIV   KNN,KFF/SINGLE /MNN,MFF/SINGLE $
CHKPNT  KFF,MFF $
COND    LBL3,SINGLE $
SCF1    USET,KNN,MNN,,/KFF,KFS,KSS,MFF,, $
CHKPNT  KFS,KSS,KFF,MFF $
LABEL   LBL3 $
EQUIV   KFF,KAA/OMIT /MFF,MAA/OMIT $
CHKPNT  KAA,MAA $
ALTER  88
SMP2    USET,GO,MFF/MAA $
CHKPNT  MAA $
ALTER  90,140
ALTER  146,153
SETVAL  //V,N,BREAK/C,N,1/V,N,LINK/C,N,-1 $
SAVE    BREAK,LINK $
LABEL   NLVIB $
EQUIV   KAA,KDAA/BREAK $
EQUIV   MAA,MDAA/BREAK $
READ    KAA,MAA,,,FEED,USET,CASECC/LAMA,PHIA,,OEIGS/C,N,MODES/V,
        N,NEIG/C,N,2 $
SAVE    NEIG $
CHKPNT  LAMA,PHIA,OEIGS $
OFP     LAMA,OEIGS,,,,//V,N,CARDNO $
SAVE    CARDNO $
COND    FINIS,NEIG $
SDR1    USET,,,PHIA,,,GO,GM,,KFS,,/PHIG,,BQG/ C,N,1/C,N,REIG $
CHKPNT  PHIG,BQG $

```

APPENDIX

EQUIV	PHIG,PHIAMP/BREAK \$
ADD	PHIG,/PHIAMP/V,Y,AMP \$
CHKPNT	PHIAMP \$
EQUIV	PHIAMP,PHIG/LINK \$
TA1	FCT,FPT,BGPD,T,SIL,GPTT,CSTM/X1,X2,FCPT,GPCT/V,N,LUSET/ V, NOSIMP/C,N,0/V,N,NOGENL/V,N,GENEL \$
DSMG1	CASECC,,SIL,EDT,PHIG,CSTM,MPT, FCPT,GPCT,DIT/KDGG/V,N, DSCASET/C,N,1 \$
CHKPNT	KDGG \$
ADD	KDGG,KGG/KDGGG \$
CHKPNT	KDGGG \$
EQUIV	KDGGG,KDNN/MPCF2 /MGG,MDNN/MPCF2 \$
CHKPNT	KDNN,MDNN \$
COND	LRL2D,MPCF2 \$
MCF2	USET,GM,KDGGG,MGG,,/KDNN,MDNN,, \$
CHKPNT	KDNN,MDNN \$
LABEL	LRL2D \$
EQUIV	KDNN,KDFF/SINGLE /MDNN,MDFE/SINGLE \$
CHKPNT	KDFF,MDFE \$
COND	LRL3D,SINGLE \$
SCF1	USET,KDNN,MDNN,,/KDFF,KDFS,,MDFE,, \$
CHKPNT	KDFF,KDFS,MDFE \$
LABEL	LRL3D \$
EQUIV	KDFF,KDAA/OMIT /MDFE,MDAA/OMIT \$
CHKPNT	KDAA,MDAA \$
COND	LRL5D,OMIT \$
SMP1	USET,KDFF,,/GDO,KDAA,KDOO,LDOO,UDOO,,,,, \$
CHKPNT	GDO,KDAA \$
SMP2	USET,GDO,MDFE/MDAA \$
CHKPNT	MDAA \$
LABEL	LRL5D \$
EQUIV	KDAA,KAAL/LINK \$
EQUIV	MDAA,MAAL/LINK \$
REPT	NLVIB,7 \$
ADD	MDAA,KDAA/KMAA \$
ADD	PHIAMP,/PHIM \$
ALTER	169.170
ENDALTER	
CEND	

REFERENCES

1. Mei, C.: Finite Element Displacement Method for Large Amplitude Free Flexural Vibrations of Beams and Plates. Computers and Structures, Vol. 3, Jan. 1973, pp. 163-174.
2. Berger, H. M.: A New Approach to the Analysis of Large Deflections of Plates. J. Applied Mechanics, Vol. 22, Dec. 1955, pp. 465-472.
3. Wah, T.: Large Amplitude Flexural Vibration of Rectangular Plates. Int. J. Mech. Sci., Vol. 5, 1963, pp. 425-438.
4. Wah, T.: The Normal Modes of Vibration of Certain Nonlinear Continuous Systems. J. Applied Mechanics, Vol. 31, March 1964, pp. 139-140.
5. Chu, H. N., and Herrmann, G.: Influence of Large Amplitudes on Free Flexural Vibrations of Rectangular Elastic Plates. J. Applied Mechanics, Vol. 23, Dec. 1956, pp. 532-540.
6. Yamaki, N.: Influence of Large Amplitudes on Flexural Vibrations of Elastic Plates. ZAMM, Vol. 41, 1961, pp. 501-510.
7. Rao, G. V., Raju, K. K., and Raju, I. S.: Finite Element Formulation for the Large Amplitude Free Vibrations of Beam and Orthotropic Circular Plates. Computers and Structures, Vol. 6, 1976, pp. 169-172.
8. Rao, G. V., Raju, I. S., and Raju, K. K.: A Finite Element Formulation for Large Amplitude Flexural Vibrations of Thin Rectangular Plates. Computers and Structures, Vol. 6, 1976, pp. 163-167.
9. Mei, C.: Nonlinear Vibration of Beams by Matrix Displacement Method. AIAA J., Vol. 10, March 1972, pp. 355-357.
10. Timoshenko, S., and Woinowsky-Krieger, S.: Theory of Plates and Shells. McGraw Hill, 1959, Chap. 13.
11. Cowper, G. R., Kosko, E., Lindberg, G. M., and Olson, M. D.: A High Precision Triangular Plate-Bending Element. LR-514, Dec. 1968, National Research Council of Canada, Ottawa, Canada.
12. Narayanaswami, R., and Mei, C.: Addition of Higher Order Plate Elements to NASTRAN. NASA TM X-3428, Oct. 1976, pp. 439-477.
13. Zienkiewicz, O. C.: The Finite Element Method in Engineering Science. McGraw Hill, 1971, p. 151.
14. Bergan, P. G., and Clough, R. W.: Convergence Criteria for Iterative Processes. AIAA J., Vol. 10, Aug. 1972, pp. 1107-1108.

15. Mei, C., and Rogers, J. L., Jr.: Application of NASTRAN to Large Deflection Supersonic Flutter of Panels. NASA TMX-3428, Oct. 1976, pp. 67-97.
16. Leissa, A. W.: Vibration of Plates. NASA SP-160, 1969.

Table 1 Fundamental frequency parameter $\lambda = \omega_L a^2 \sqrt{\rho/D}$ and frequency ratio ω/ω_L
for a simply supported rectangular plate

Gridwork	D.O.F.		λ	ω/ω_L at c/h				
	Total	Reduced		0.2	0.4	0.6	0.8	1.0
Aspect ratio $a/b = 1.0$								
1 x 1	7	3	19.4720	1.0182	1.0704	1.1511	1.2535	1.3717
2 x 2	26	10	19.6698	1.0185	1.0714	1.1526	1.2550	1.3726
3 x 3	57	21	19.7064	1.0185	1.0716	1.1533	1.2563	1.3747
4 x 4	100	36	19.7201	1.0185	1.0717	1.1534	1.2565	1.3751
Chu-Hermann (ref. 5)	-	-		1.0195	1.0757	1.1625	1.2734	1.4024
2 x 2 (Rao et al. ref. 8)	16	16		1.0185	1.0716	1.1530	1.2561	1.3744
4 x 4	64	64		1.0185	1.0717	1.1534	1.2565	1.3753
Aspect ratio $a/b = 0.5$								
1 x 1	12	4	11.7643	1.0233	1.0904	1.1942	1.3268	1.4810
2 x 2	48	16	12.2002	1.0239	1.0918	1.1950	1.3256	1.4711
3 x 3	108	36	12.2745	1.0239	1.0921	1.1960	1.3263	1.4753
Chu-Hermann (ref. 5)	-	-		1.0241	1.0927	1.1975	1.3293	1.4808
2 x 2 (Rao et al. ref. 8)	16	16		1.0238	1.0917	1.1955	1.3261	1.4758
4 x 4	64	64		1.0239	1.0918	1.1957	1.3264	1.4758

Exact value of $\lambda = 19.7392$ ($a/b = 1.0$) and 12.3370 ($a/b = 0.5$).

Table 2 Fundamental frequency parameter $\lambda = \omega_L a^2 \sqrt{\rho/D}$ and frequency ratio ω/ω_L for a clamped rectangular plate

Gridwork	D.O.F. Total Reduced		λ	ω/ω_L at c/h				
				0.2	0.4	0.6	0.8	1.0
Aspect ratio $a/b = 1.0$								
1 x 1	5	3	31.9446	1.0099	1.0387	1.0847	1.1450	1.2167
2 x 2	22	10	35.2474	1.0070	1.0276	1.0609	1.1052	1.1590
3 x 3	51	21	35.6358	1.0070	1.0277	1.0608	1.1046	1.1575
4 x 4	92	36	35.7778	1.0070	1.0277	1.0608	1.1047	1.1577
Yamaki (ref. 6)	-	-	37.22	1.0075	1.0295	1.0653	1.1130	1.1712
2 x 2 (Rao et al. ref. 8)	9	9		1.0070	1.0280	1.0616	1.1067	1.1613
4 x 4	49	49		1.0070	1.0276	1.0608	1.1047	1.1578
Aspect ratio $a/b = 0.5$								
1 x 1	8	4	19.8996	1.0111	1.0438	1.0960	1.1650	1.2478
2 x 2	40	16	23.5777	1.0073	1.0290	1.0644	1.1125	1.1720
3 x 3	96	36	24.1046	1.0074	1.0292	1.0644	1.1116	1.1691
2 x 2 (Rao et al. ref. 8)	9	9		1.0074	1.0293	1.0651	1.1136	1.1736
4 x 4	49	49		1.0074	1.0292	1.0647	1.1123	1.1686

Theoretical value of $\lambda = 35.9990$ ($a/b = 1.0$) and 24.5777 ($a/b = 0.5$) from ref.16.

Table 3 Higher frequency parameters $\lambda = \omega_L a^2 \sqrt{\rho/D}$ and frequency ratios ω/ω_L
for a simply supported and clamped square plate

Mode m n		λ		ω/ω_L at c/h				
		Present	Leissa (ref. 16)	0.2	0.4	0.6	0.8	1.0
Simply supported								
2	1	49.113	49.348	1.0241	1.0930	1.1984	1.3307	1.4822
2	2	78.551	78.957	1.0188	1.0729	1.1560	1.2610	1.3816
3	1	97.058	98.696	1.0118	1.0462	1.1007	1.1722	1.2577
Clamped								
2	1	72.310	73.405	1.0152	1.0583	1.1239	1.2064	1.3015
2	2	106.530	108.237	1.0141	1.0543	1.1152	1.1916	1.2789
3	1	128.250	132.210	1.0094	1.0368	1.0797	1.1356	1.2016

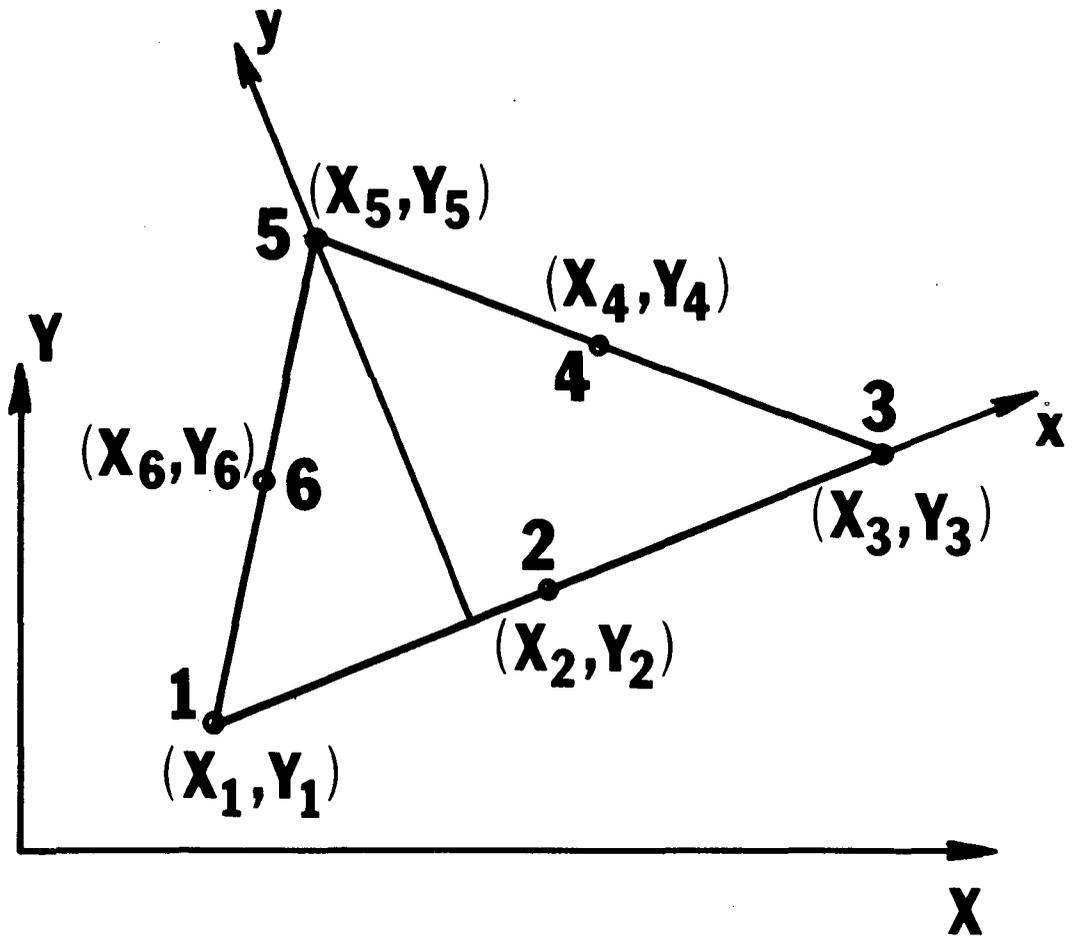


FIGURE 1. GEOMETRY OF TRPLT1 ELEMENT.

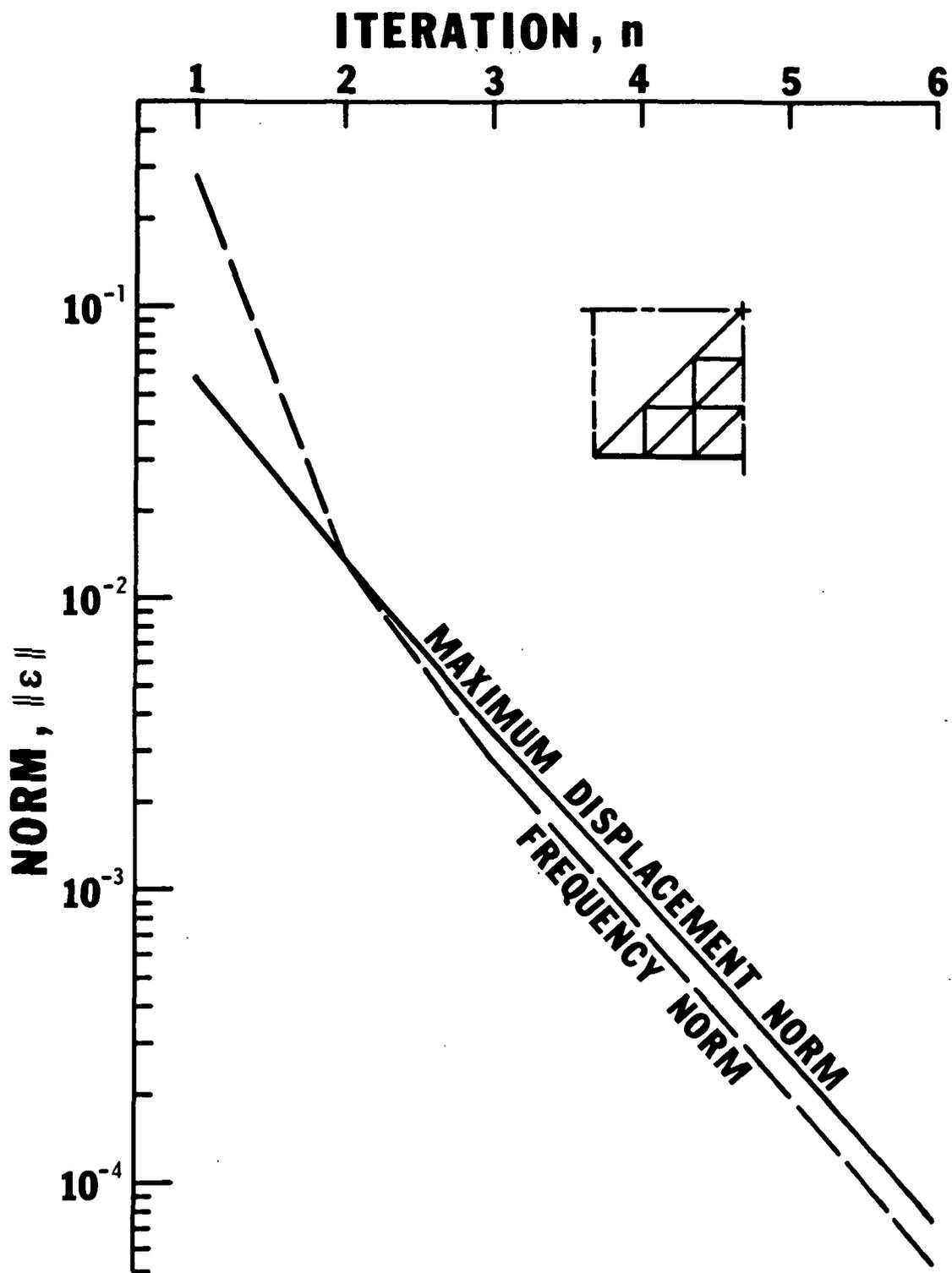
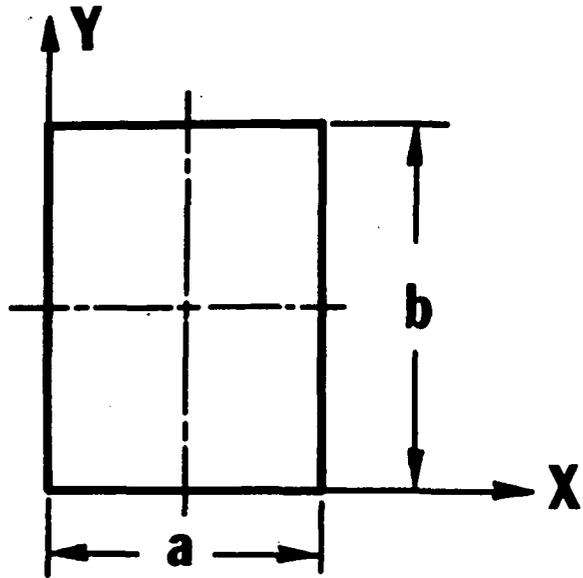
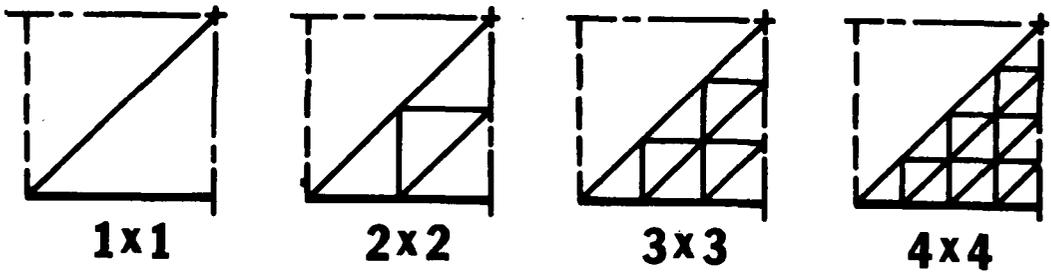


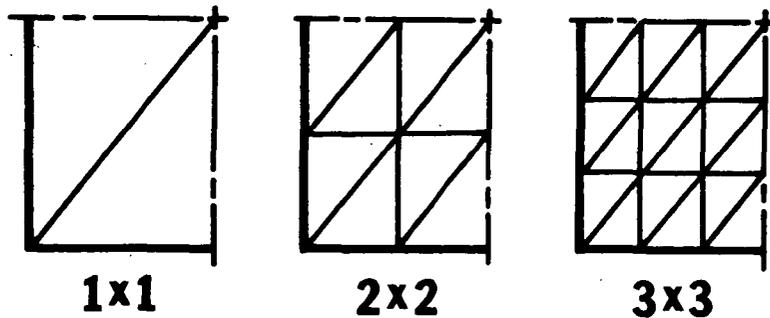
FIGURE 2. CONVERGENCE CHARACTERISTICS FOR A SIMPLY SUPPORTED SQUARE PLATE AT $c/h = 1.0$ WITH 3×3 GRIDWORK.



(a) Rectangular plate.



(b) Gridwork for square plate.



(c) Gridwork for rectangular plate.

FIGURE 3. FINITE ELEMENT IDEALIZATIONS.

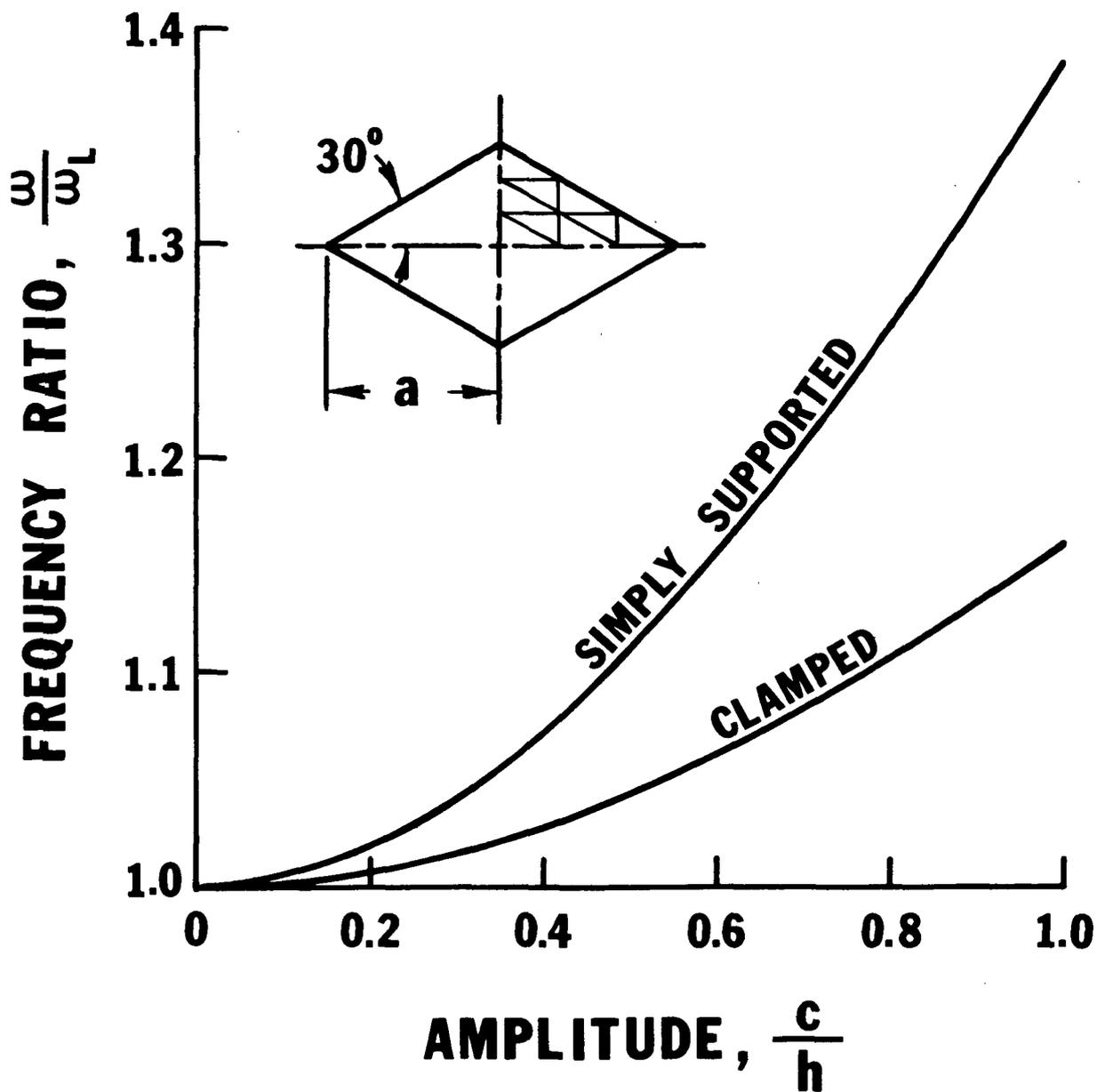


FIGURE 4. RATIO OF NONLINEAR TO LINEAR FREQUENCY AS FUNCTION OF AMPLITUDE FOR RHOMBIC PLATES.

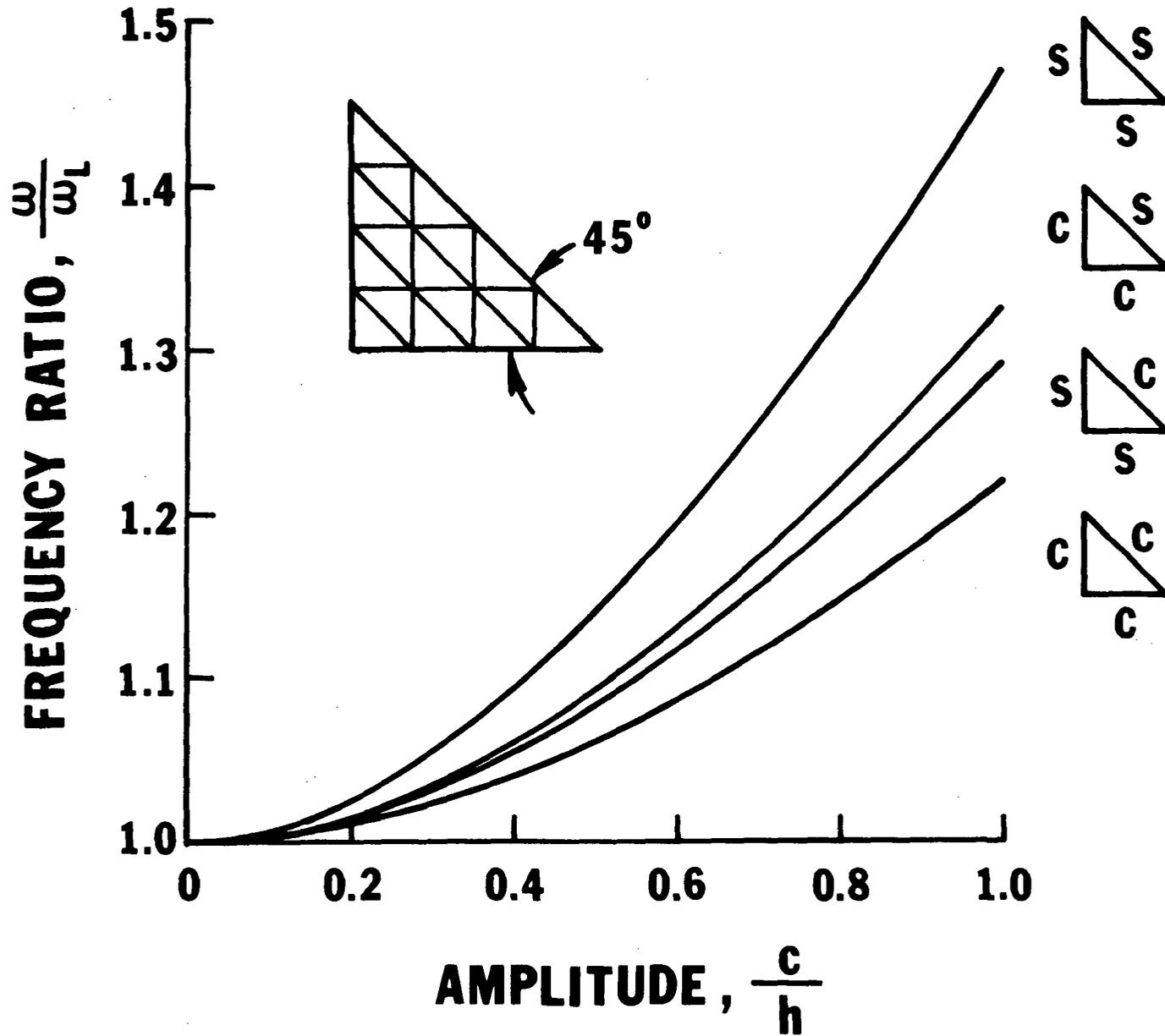


FIGURE 5. FREQUENCY RATIO AS FUNCTION OF AMPLITUDE FOR ISOSCELES TRIANGULAR PLATES.

A CONDENSED FORM OF NASTRAN

James L. Rogers, Jr.
NASA/Langley Research Center

Chuh Mei
Vought Corporation

W. Keith Brown
Computer Sciences Corporation

INTRODUCTION

NASTRAN (NASA Structural Analysis) is a finite element computer program for structural analysis that is intended for general use. As such, it must answer to a wide spectrum of requirements, as well as permit future modifications and continued expansion to new problem areas. But because of its size, cost, generality, and voluminous documentation, NASTRAN has not gained popularity among universities and small consulting firms. These organizations for the most part, neither need all of the capability NASTRAN provides, nor can afford the cost of more recent levels. To provide a form of NASTRAN compatible with their needs a condensed NASTRAN was created. This condensed form of NASTRAN is simply a limited capability form of Level 16.

CAPABILITIES

Capabilities for a condensed form of NASTRAN were selected after discussions with engineers, programmers, and university professors familiar with NASTRAN. Three types of analysis which appeared to be most widely favored were selected:

1. Linear Static Analysis
2. Vibration Analysis
3. Buckling Analysis

Rigid Formats 1, 3, and 5 respectively, solve these types of problems in NASTRAN. The Rigid Formats, however, are not provided in the condensed form of NASTRAN. The DMAP (Direct Matrix Abstraction) programming language is used instead. With DMAP, the user can go beyond the scope of the existing three analyses and also solve a broad range of non-structural matrix problems - all within the framework of NASTRAN. The user will have at his disposal 58 modules with which to do his own DMAP programming. The modules are listed by categories in Table 1.

The number of input cards (Executive, Case Control, and Bulk Data) that are available as input to the condensed form of NASTRAN, has also been limited. A list of these 53 cards is shown in Table 2. Only four finite elements are available for structural modeling; they include BAR (single beam), SHEAR (shear panel), TRIA2 (triangular membrane and bending), and TRIMEM (triangular membrane). In addition, the CONM2 (concentrated mass) is also included. Only one method of eigenvalue extraction, Inverse Power, can be selected using the EIGB and EIGR cards. The user can obtain a graphical display of the structural model by using the NASTRAN plot package (ref. 1). The Stromberg Carlson, CALCOMP, and NASTRAN general purpose plotters are permitted.

FORMATION OF THE PROGRAM

The condensed form of NASTRAN was created by entirely deleting five of the existing seventeen links in the CDC Level 16 version of NASTRAN. These were Links 3, 9, 10, 11, and 15. Next, the remaining links, except Links 0 and 20 (CDC version), were reduced by removing unneeded object decks from the overlay structure. Before reduction, the longest link (Link 13) was 127.5K₈ locations. After reduction, the longest link (Link 8) was only 115.5K₈ locations, a saving of 12K₈ locations. In addition, considerable disk storage was saved by removing unnecessary subroutines from the executable code.

In addition to the overlay structure changes, eight subroutines were modified. Subroutines DS1A, EMGOLD, EMGPRO, SDR2B, and SDR2 were modified to remove calls to deleted element subroutines. Routines XBSBD and XSEM07 were modified to move modules DIAGONAL and SCALAR from Link 15 into Link 7. Subroutine TTLPGE was also modified.

DOCUMENTATION

A separate manual for the condensed form of NASTRAN is planned. This manual is intended to be a substitute for the existing NASTRAN manuals. It will be concise and geared for classroom use, but yet contain all the information required to use the capabilities remaining in the condensed form of NASTRAN.

DEMONSTRATION AND VERIFICATION

Four demonstration problems were developed to test and illustrate the use of the condensed form of NASTRAN. The four problems include:

1. Static Analysis - Deflection of a uniform beam
2. Vibration Analysis - Cantilever beam with a lumped mass at the free end
3. Buckling Analysis - A simply supported triangular plate under uniform compression
4. Test of matrix operations and utility modules not used in problems 1-3

Each problem was tested on both the CDC Level 16 of NASTRAN and the condensed form, and identical results were obtained. Level 16, however, required a field length of 160K_g, while the condensed form required 140K_g with plotting and 125K_g without plotting. Although there were no significant time differences between the two, the disk accesses were cut in half when using the condensed form of NASTRAN.

These demonstration problems will also be run on IBM, UNIVAC, and CDC (with the NOS operating system) computers. Delivery packages for each of these computers should be available to the public in the near future.

CONCLUDING REMARKS

A condensed form of NASTRAN has been created and verified. It is simply a limited capability form of Level 16 and should be compatible with the needs of universities and small consulting firms. The program is designed to perform three main types of analyses (static, vibration, and buckling), although the user can go beyond this scope with DMAP programming. The user has at his disposal 58 modules, 53 input cards including four structural elements (BAR, SHEAR, TRIA2, and TRIMEM), and a plotting package. Although the condensed form of NASTRAN and Level 16 NASTRAN yield identical results for identical problems, the condensed form requires 20K_g - 30K_g less storage locations for execution on the CDC computer and reduces the number of disk accesses by 50%. Documentation is planned for this program in the form of one concise manual that can be substituted for the four existing manuals. Delivery packages should be available to the public on IBM, CDC, UNIVAC computers in the near future.

REFERENCE

1. Wilkinson, M. T.: "Use of NASTRAN as a Technology Aid," NASTRAN: User's Experiences. NASA TM X-2637, 1972, pp. 415-419.

TABLE 1. MODULES WITHIN THE CONDENSED FORM OF NASTRAN

Matrix Operation Modules

ADD	MPYAD
DECOMP	PARTN
FBS	SOLVE
MERGE	TRNSP

Executive Operation Modules

COND	JUMP
END	LABEL
EQUIV	PURGE
EXIT	REPT
FILE	XDMAP

Utility Modules

COPY	PARAM
DIAGONAL	PARAML
INPUT	PARAMR
INPUTT2	PRTPARM
MATPRN	SCALAR
OUTPUT2	SETVAL
	SWITCH

Structurally Oriented Modules

DPD	OFF	SCE1
DSMG1	PLOT	SDR1
EMA	PLTSET	SDR2
EMG	PRTMSG	SMP1
GP1	RBMG1	SMP2
GP2	RBMG2	SSG1
GP3	RBMG3	SSG2
GP4	RBMG4	SSG3
GPSP	READ	TA1

TABLE 2. ALLOWABLE INPUT CARDS FOR THE CONDENSED FORM OF NASTRAN

Executive Control Deck

Case Control Deck

ID	DEFORM	PLOTID
APP	DISPLACEMENT	SDISPLACEMENT
TIME	ECHO	SET
DIAG	ELFORCE	SPC
CEND	LABEL	SPCFORCES
	LINE	STRESS
	LOAD	SUBCASE
	MAXLINES	SUBTITLE
	METHOD	TITLE
	OLOAD	\$ (comment)
	OUTPUT	

Bulk Data Deck

CBAR	DMI	PARAM
CNGRNT	EIGB	PBAR
CONM2	EIGR	PSHEAR
CORD1C	FORCE	PTRIA2
CORD1R	GRID	PTRMEM
CORD1S	LOAD	SPC
CSHEAR	MAT1	SPCADD
CTRIA2	MOMENT	SUPPORT
CTRMEM	OMIT	\$ (comment)

**Page
Intentionally
Left Blank**

INTERACTIVE GRAPHICS SUPPORT FOR NASTRAN

William E. Lorensen
Benet Weapons Laboratory - ARRADCOM

INTRODUCTION

The Interactive Graphics Finite Element System, IGFES, is described along with its supporting analysis software, graphics terminal support package and hardware configurations. IGFES provides an interactive design tool for structural engineers via pre- and postprocessing of finite element data. The system currently runs on an IBM 360/44 OS-MFT system or a PDP 11/40 DOS/BATCH system. Graphics devices are supported using an inhouse developed, device independent terminal support package. Support is available for the Calcomp 563 drum plotter, Tektronix 4002A storage display terminal and the Lundy Electronics 20 inch standalone refresh display system. IGFES and its associated systems are written in Fortran IV.

FINITE ELEMENT ANALYSIS

Even the neophyte finite element user is aware of the drawbacks of the powerful finite element technique. Input preparation, model verification and output analysis is much more expensive than the actual cost of the finite element analysis. To make NASTRAN and other analysis codes cost-effective, many people have developed automatic preparation and data reduction systems. Reference 1 describes a number of these systems. The design goals considered during the development of IGFES included: ease of use, graphics device and computer independence, a uniform interactive man-machine interface and modularity of coding.

COMPUTER GRAPHICS

Computer graphics offers one solution to the finite element input/output problem. Implementation of the graphics solution can take two forms. First, one can modify existing finite element codes to include the interactive capability. This approach offers the advantages of uninterrupted completion of the preparation, verification, analysis and data reduction cycle. However, its disadvantages are obvious. Each finite element code must be modified to provide the graphics capability. Also, most finite element codes require extensive computation times for execution and these delays are intolerable in an interactive environment.

The second technique is the most general and practical. In the pre- and postprocessing approach, the interactive portions of the system are separated from the analysis. Consequently, these programs can support several finite element codes. Also the interactive portions of the system can be run in time-sharing or even minicomputer settings. We chose this second approach and by isolating analysis dependent coding and graphics device dependent coding we have a system designed to support a variety of analysis codes and devices. Furthermore, new interactive support routines are easily added by adhering to the overall design objectives of the system.

Before describing the IGFES components in detail, a few words about graphics support software are appropriate. There is currently no standard graphics support package available. Although industry standardization is not imminent, users should attempt to standardize at the local level. Our installation has three different graphics devices, and we have designed a system of routines to achieve device independence. When doing this, one must be careful not to sacrifice a device's capabilities to achieve independence. An example is appropriate. Most graphics programs should be controlled by the user. He must be able to change parameters, select options and cycle through the program as he wishes. Menus give the user this capability. Our menu routine returns to the caller the x, y (and z) coordinates of a graphics cursor as well as a function code. The Calcomp support package allows the user to enter this function code on cards along with x, y (and z) coordinates. On the storage display terminal the user positions crosshairs with a joystick and presses a key containing the function code. Lastly, on the refresh graphics system, the user positions a tracking cross with the lightpen and selects a menu item with the pen. The software detects which item was selected and passes a corresponding function code to the calling routine. The graphics application program is unaware of the device on which it is running for it only receives a function code and coordinate information.

Hardcopy output is also implemented in a device independent fashion. Plotter users obtain hardcopies during normal operation of the program while storage display users merely press the copy button at their terminals. The refresh system implementation has a pre-assigned function key which may be pressed at any time by the user. This action dumps the current display file onto auxiliary storage. When the session is complete, the data describing the display files is transferred to the Model 44 where it is interpreted and displayed on the plotter.

The former examples are indications of the capabilities possible after careful design of graphics support packages. Similar device independence is achieved in the following areas: screen erasure, alphanumeric and numeric input, symbol generation, vector display using virtual graphics and three-dimensional graphics.

IGFES INPUT MODULE

The IGFES input module consists of three separate programs: GRID2D, a two-dimensional grid generator; GRID3D, a three-dimensional plate-type element and grid generator; and EDITOR, a structural model preview program.

GRID2D and GRID3D operate under user control and generate finite element grid points and elements on arbitrary boundaries. These boundaries are entered by the user as straight or curved segments. The segments are represented internally as two or three piece-wise cubic spline functions of chord length,

$$x = f(c), y = g(c)$$

and in GRID3D,

$$z = h(c), \text{ where } c = \text{chord length measured from the first point on each boundary.}$$

Using a graphics cursor, the user defines a mapping between his arbitrary boundaries and a rectangular area. He does this by defining four "corners" which establish the sides for a rectangle. Along any of the sides he may enforce grid points at specific points, e.g. physical corners. Any points not enforced are generated by the program. Unless otherwise instructed, the program generates these points equally-spaced on each side. The user has the option of defining a spacing algorithm which can vary from uniform to arithmetic to geometric spacing. A separate spacing algorithm can be defined for each side. Four algorithms are available for interior grid point generation. Each algorithm uses the grid points that have been generated for the sides.

Algorithm 1: LaPlacian

The LaPlacian generation is an iterative one. The name is derived from the finite difference LaPlacian operator. Each coordinate of a grid point is calculated from four of its neighbors:

$$x_{ij}(y_{ij} \text{ or } z_{ij}) = \frac{x_{i-1,j} + x_{i+1,j} + x_{i,j-1} + x_{i,j+1}}{4}$$

where $2 \leq i \leq M-1$, $2 \leq j \leq N-1$ with M and N the number of grid points on each side of the rectangular region. Initially all interior coordinates are set to zero. All interior coordinates are calculated in a loop iteratively until a predetermined norm is satisfied. Using this method for concave sections of boundaries can sometimes result in grid points outside the boundary; however, since the user can see this on the display, he can correct the problem by changing the grid point density or selecting another algorithm.

Algorithm 2: Direct Ray

The Direct Ray algorithm draws rays between sides 1 and 3 of the grid region and divides these rays into segments according to the grid point spacing on sides 2 and 4. Choice of sides 1 and 3 should be made so that the rays do not lie outside the boundaries.

Algorithm 3: Coons Blending

This method utilizes an adaptation of Coons blending functions (ref. 2) to define a mapping between a rectangle in (u,v) space and the arbitrary user boundary in cartesian space. The boundary functions are linear interpolates of the generated points on each side and cubic blending functions are used.

Algorithm 4: Isoparametric Mapping

Quadratic isoparametric shape functions are used to define a mapping from $-1 \leq \xi, \eta \leq 1$ into the user defined region. The shape functions are:

$$x = \sum_{i=1}^8 N_i(\xi, \eta) x_i$$

$$y = \sum N_i(\xi, \eta) y_i$$

$$z = \sum N_i(\xi, \eta) z_i, \quad -1 \leq \xi, \eta \leq 1$$

$$N_i = [(1+\xi\xi_i)(1+\eta\eta_i) - (1-\xi^2)(1+\eta\eta_i) - (1-\eta^2)(1+\xi\xi_i)]\xi_i^2\eta_i^2/4 \\ + (1-\xi^2)(1+\eta\eta_i)(1-\xi_i^2)\eta_i^2/2 + (1-\eta^2)(1+\xi\xi_i)(1-\eta_i^2)\xi_i^2/2 \quad (\text{ref. 3})$$

where N_i is the shape function at node i whose cartesian and curvilinear coordinates are (x_i, y_i) and (ξ_i, η_i) respectively. Nodes 1-4 are corner nodes while nodes 5-8 are midside nodes. The shape functions are used for interior generation only. Since this implementation uses points previously generated on the boundaries, the usual limitation of isoparametric mapping, i.e. representation of a restricted class of boundaries, is avoided.

All of the above algorithms may be employed to define a number of grid regions which describe the idealization. The generator automatically eliminates duplicate grid points on adjacent region sides. A number of display and generating options are available including: display of all generated regions, display of nodal numbering, punched output of grid and connection cards, user selection of graphics windowing and user control over grid and element labels. Currently, all NASTRAN triangular and quadrilateral elements are supported. Since the NASTRAN dependence has been isolated to two subroutines, support for other finite element codes is facilitated. Figures 1 through 6 illustrate GRID2D and GRID3D capabilities.

CREATE and EDITOR are the two programs comprising the structural preview segment of IGFES. CREATE establishes a data base from NASTRAN Bulk Data decks, checking syntax as it goes. The data base is an associative data structure (ref. 4) which allows data to be retrieved in a variety of ways. Data is stored by the relation: attribute (object) = value or $A(\phi) = V$. Once CREATE has been run on a Bulk Data deck, EDITOR can retrieve from direct access data sets information relevant to a given grid point or element. EDITOR allows user definition of those items in the structure to be displayed. Grid points can be displayed by id or coordinate system id; ranges of ids can be displayed; and symbols or labels can be displayed at grid point locations. Element display options include: display by element type, element id and/or material id; display of all types except for one selected by user; display of element labels and specification of an element shrink factor whereby elements are drawn reduced in size to show each individual element. Figure 7 illustrates the element shrink option. Another option is available to display element segments defining the boundaries of a structure. These segments are determined by CREATE in the following way. An element segment is considered a boundary segment if it is referenced in an odd number of elements. Figure 8 shows a structure with only boundary segments displayed. The shrink and boundary displays are particularly useful in detecting missing elements. EDITOR runs in three-dimensional display mode on the refresh graphics system and the user has a hardware rotation capability which allows dynamic rotation of his structure. The user can also work with four views simultaneously: front, side, top and arbitrary. Figure 9 shows a four-view display. Lastly, the user can define images at the screen which he can erase or delete selectively. These images can be any collection of structural information he wishes.

IGFES OUTPUT MODULE

The output module consists of programs to interface IGFES with the analysis program and programs to reduce and display the finite element results. The data reduction and display programs work with x,y,z data and can be used to process data from other empirical sources.

NASIGIO is the NASTRAN-IGFES input/output interface program. NASIGIO runs in batch mode, producing a file to be accessed by the graphics application routines. Input to this program consists of a NASTRAN OUTPUT2 tape created by using rigid format alters. The OUTPUT2 tape contains BGPDT, the basic grid point definition table; EQEXIN, equivalence table between external and internal grid points; SIL, the scalar index list; EST, element summary table; and the appropriate stress file, e.g. OES1 for static analyses. Given this information, NASIGIO collects all data associated with each subcase, transforms stresses from the elemental to the global coordinate system, computes maxima and minima for each independent and dependent variable, determines which grid points lie on the boundaries of a structure, and produces the standard interface file shown in Table 1. The dependent variables are stress components at the centroids of elements. Note that NASIGIO compresses all data into x,y,z format, a general form required by the display routines. NASIGIO can be used independently of IGFES as a general NASTRAN data reduction program.

SELECT is the data selection program, also running in batch mode. SELDAT works from the interface file and acting on user specified subcase numbers and dependent variable identifiers, creates a boundary description file and an x,y,z data file. SELECT must be executed prior to the other programs if the user is analyzing NASTRAN data.

SURFACE performs a piecewise doubly cubic spline approximation of the data. The function produced is:

$$\Phi = z(x,y) = \sum_{k=0}^3 \sum_{l=0}^3 a_{ijkl} (x-x_i)^k (y-y_j)^l,$$

$$i=1, \dots, Nx-1; j=1, \dots, Ny-1$$

where Nx = Number of x grid points
 Ny = Number of y grid points.

The x and y grid points form rectangles over which each piecewise spline is valid. The coefficients a_{ijkl} are determined by minimizing

$$\iint \{\nabla\Phi\}^2 dS + \lambda \sum_{i=1}^N w_i^2 \{\Phi(x_i, y_i) - z_i\}^2$$

where N = number of data points.

λ is a smoothing parameter. As $\lambda \rightarrow \infty$, a least squares approximation of the data is realized. As $\lambda \rightarrow 0$, the surface becomes a plane. The basis functions for the surface are the Cardinal Splines (ref. 5) calculated over the approximating grid points. Reference 6 describes the one-dimensional spline interpolation routine used as a foundation for the doubly-cubic routine.

CONTOUR is the IGFES contour display program. Using the doubly-cubic spline produced by SURFACE, CONTOUR draws lines of constant dependent variable superimposed on an outline of the structure. This program runs interactively and allows the user to select the contour levels to be tracked and portion of the structure he wishes to view. Contouring is basically a root-finding and bookkeeping process. CONTOUR uses a grid overlaid on the structure to facilitate bookkeeping. Initially each grid point is checked for its location with respect to the boundaries of the structure. Points outside the boundaries are flagged and the function is evaluated at all interior grid points. Typically twenty grid points are used in each direction. Once the user specifies the level to be tracked, the grid is searched for sides which contain this level. The equation $z(x,y) - \text{level} = 0$ is solved repetitively for fixed x or y obtaining the corresponding y or x. Tracking continues in this manner until:

1. No root is found in three attempts. An attempt is defined as fixing x (or y) and incrementing y (or x) in one direction.
2. The x or y limits set by the user are exceeded.

3. The starting grid cell is reentered.
4. A contour point is found outside the boundary.

Once tracking stops, the algorithm returns to the first point (unless condition 3 is met) and tracking continues in the opposite direction. As each cell is entered, it is flagged to prevent starting in the cell. Contours are labeled with alphabetic or numeric symbols and a legend containing label-level correspondences is provided. Other options available include: drawing an axis, changing tracking stepsizes, changing the number of bookkeeping grid points and suppressing the labeling. Sample CONTOUR output is illustrated in figures 10 and 11. A detailed description of an earlier version of the program is given in reference 7. CONTOUR can be easily modified to handle other analytic functions.

PERSPV allows the perspective display of the function produced by SURFACE. It uses a modification of Kubert's algorithm (ref. 8) to perform hidden-line elimination. Of course hidden-line elimination for analytic surfaces is much easier than that for polyhedra or arbitrary collections of geometric entities. Once the vantage point is established a ray is envisioned from the vantage point to the point on the surface whose visibility is in question. Every surface line between that point and the bounds of the surface is checked for its relation with the ray. If all points are below (above) the ray, the point is visible from the top (bottom). If some are above and some below, the point is not visible. When one point defining a line segment is visible and one is not, a search for the visible segment is performed. Since visibility tests are time consuming, the user has the option of turning the tests on and off. Other interactive options include entering of x, y, or z limits of view, selecting the number of surface lines, viewing of points visible only from the top or bottom and display of side bars at the edges of the surface. Figures 12 and 13 illustrate PERSPV output.

NORMAL permits the dynamic display of NASTRAN normal mode analysis output. DMAP alters are used in rigid format 3 to place onto an OUTPUT 2 tape elements, grid points and eigenvector results for the problem. An interface program, NASTODYN processes the OUTPUT 2 tape and rearranges the data into a form required by NORMAL. NORMAL runs on the PDP 11/40 with the refresh graphics display. The user selects the eigenvector he wishes to view, the range of the structure he wants to see and the number of frames to create. Each frame is a deformed plot with the eigenvector displacements multiplied by $\sin(2\pi(i-1)/N)$ maximum deformation, where i = frame number and N = number of frames. NORMAL builds frames of the eigenvector one at a time and stores the resultant display file on a direct access device. Once the frames of the sequence have been created and stored, the user can play back the sequence, either continuously or a single step at a time. As in EDITOR, a hardware rotation capability is present, so the user can change the view dynamically, or watch the normal mode vibrations in four views simultaneously.

CONCLUSIONS

A modular, high performance alternative to conventional finite element input preparation and data reduction has been described. IGFES emphasizes man-graphics interaction thus motivating the engineer and increasing his confidence in the finite element analysis. Due to its modular nature, IGFES can easily be extended and improved. Proposed extensions include solid element generation in three dimensions, more extensive work in the surface approximation area and real-time display of finite element transient response output.

REFERENCES

1. Structural Mechanics Computer Programs. University Press of Virginia, 1974.
2. Coons, S. A.: Surfaces for Computer-Aided Design of Space Forms. MIT Report, MAC-TR-41, 1967.
3. Zienkiewicz, O. C.: The Finite Element Method in Engineering Science. McGraw Hill, London, 1971.
4. Symonds, A. J.: Auxiliary-Storage Associative Data Structure for PL/I. IBM Systems Journal, Vol. 7, Nos. 3&4, 1968, pp. 229-245.
5. Ahlberg, J. H.; Nilson, E. N.; and Walsh, J. L.: The Theory of Splines and Their Applications. Academic Press, New York, 1967.
6. Scanlon, R. D.: An Interpolating Cubic Spline Fortran Subroutine. Watervliet Arsenal Tech. Report, WVT-7010, 1970.
7. Bair, R.: Computations of Smooth Contours from Non-Uniform Data. Watervliet Arsenal Tech. Report, WVT-7265, 1972.
8. Kubert, B.; Szabo, J.; and Guilieri, S.: Perspective Representation of Functions of Two Variables. Journal of the ACM, Vol. 15, No. 2, April, 1968.

TABLE I. - NASIGIO INTERFACE FILE

<u>RECORD NUMBER</u>	<u>DESCRIPTION</u>
1	Number of files on tape, number of data sets per file
2	Number of words per control record
3	Control record: number of title records, number of words per title record, number of distinct boundaries, number of boundary nodes, number of dependent variables, number of data records
4	Pointers to boundary node vector, defining distinct boundaries
5	Boundary node vectors
6	Dependent variable titles
7	Independent and dependent variable maxima and minima
8	ID, independent variables, dependent variables
.	
.	
.	
7 +	Number of data points .

All records except record 1 are repeated for each subcase, element type, eigenvalue, and so on.

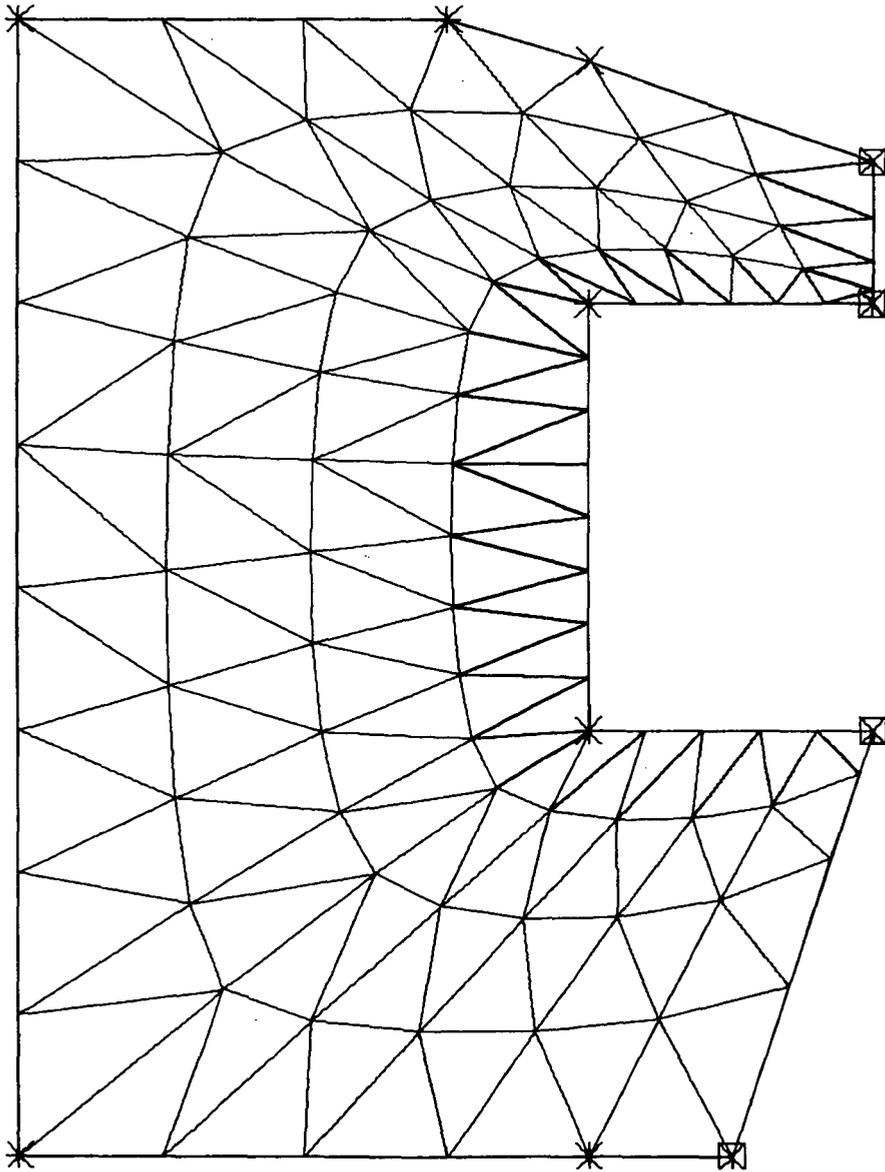


FIGURE 1. NOTCHED SPECIMEN, CHANGE IN SIDE NODAL SPACING FOR THE TWO SHORT SIDES.

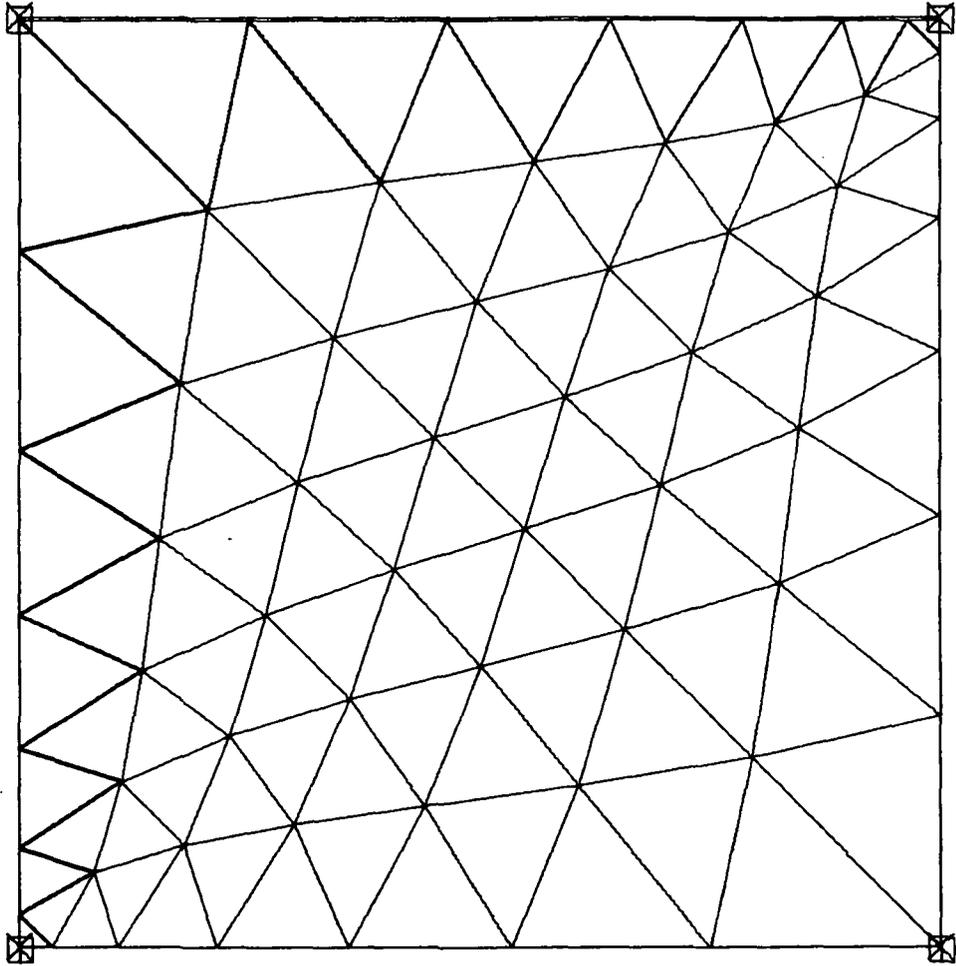


FIGURE 2. SQUARE PLATE ILLUSTRATING EFFECT OF DIFFERENT NODAL SPACING ON EACH SIDE.

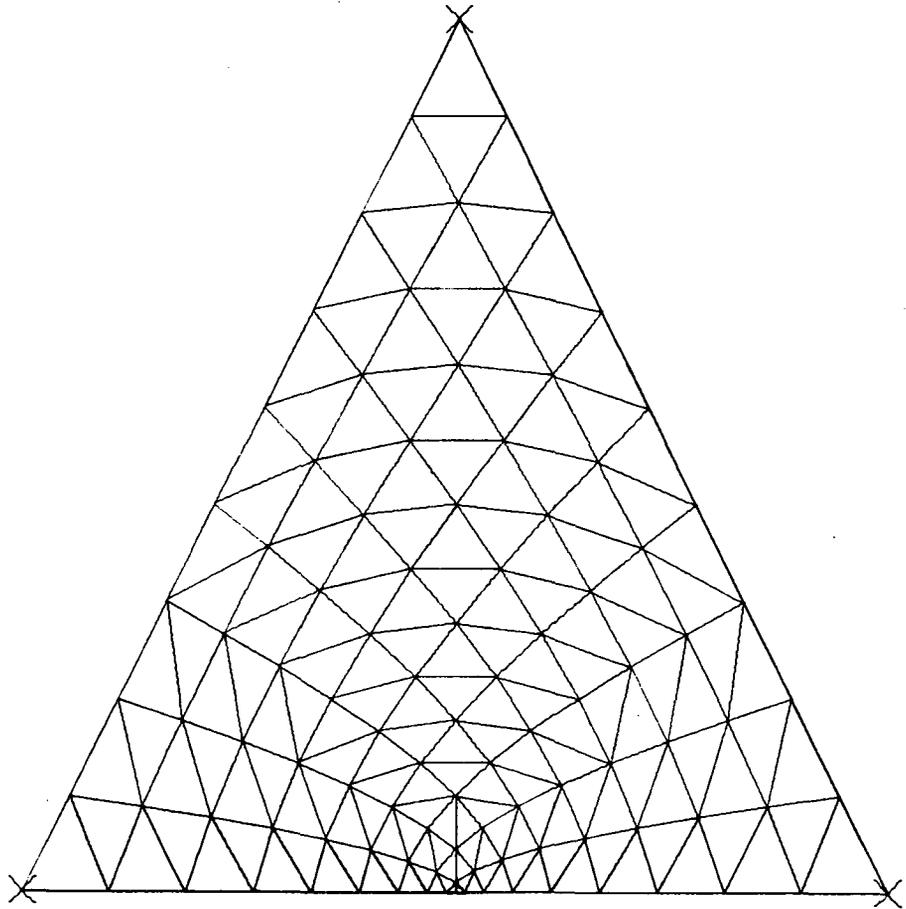


FIGURE 3. GENERATION ON A TRIANGULAR BOUNDARY.

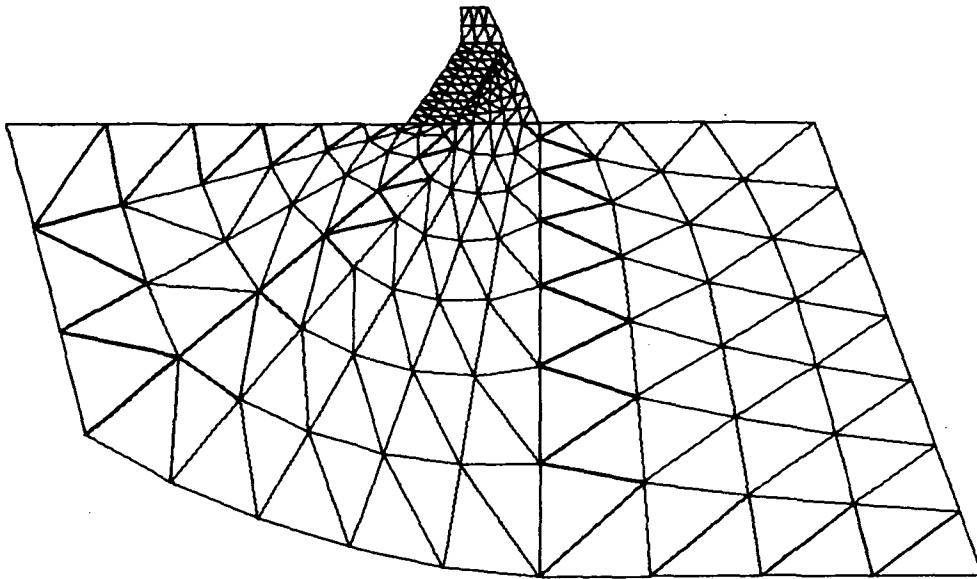


FIGURE 4. MULTIPLE GENERATIONS, WINDOWING USED FOR FINE REGIONS.

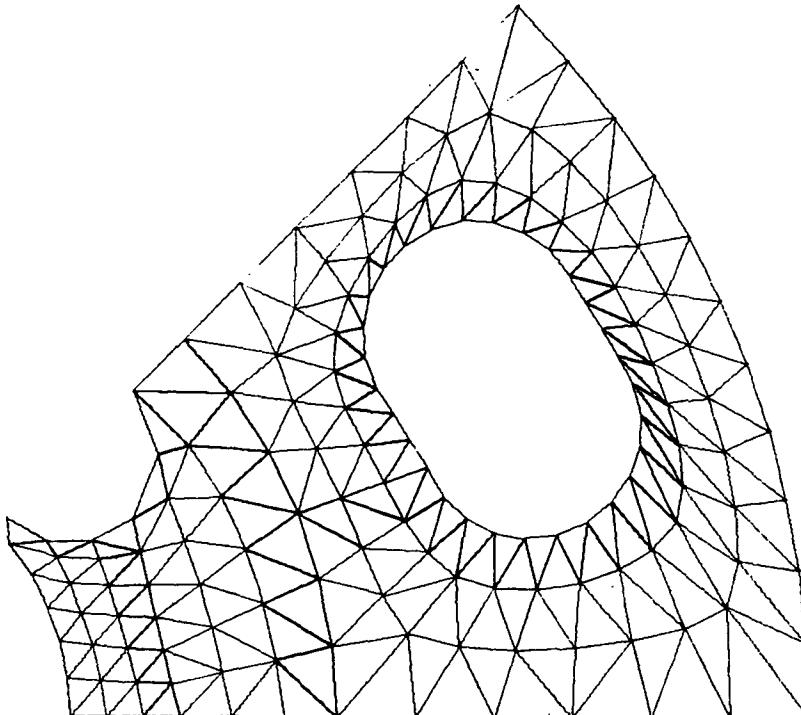


FIGURE 5. GENERATION AROUND A HOLE.

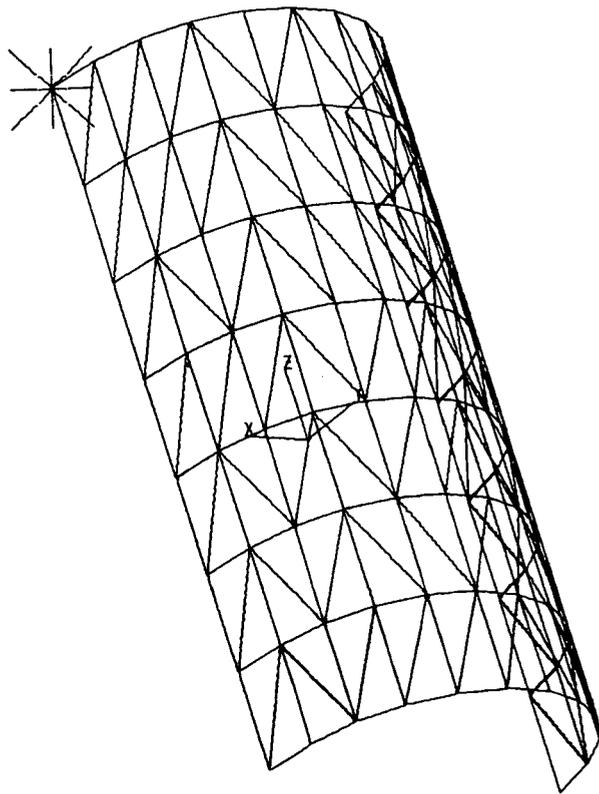


FIGURE 6. GRID3D GENERATION.

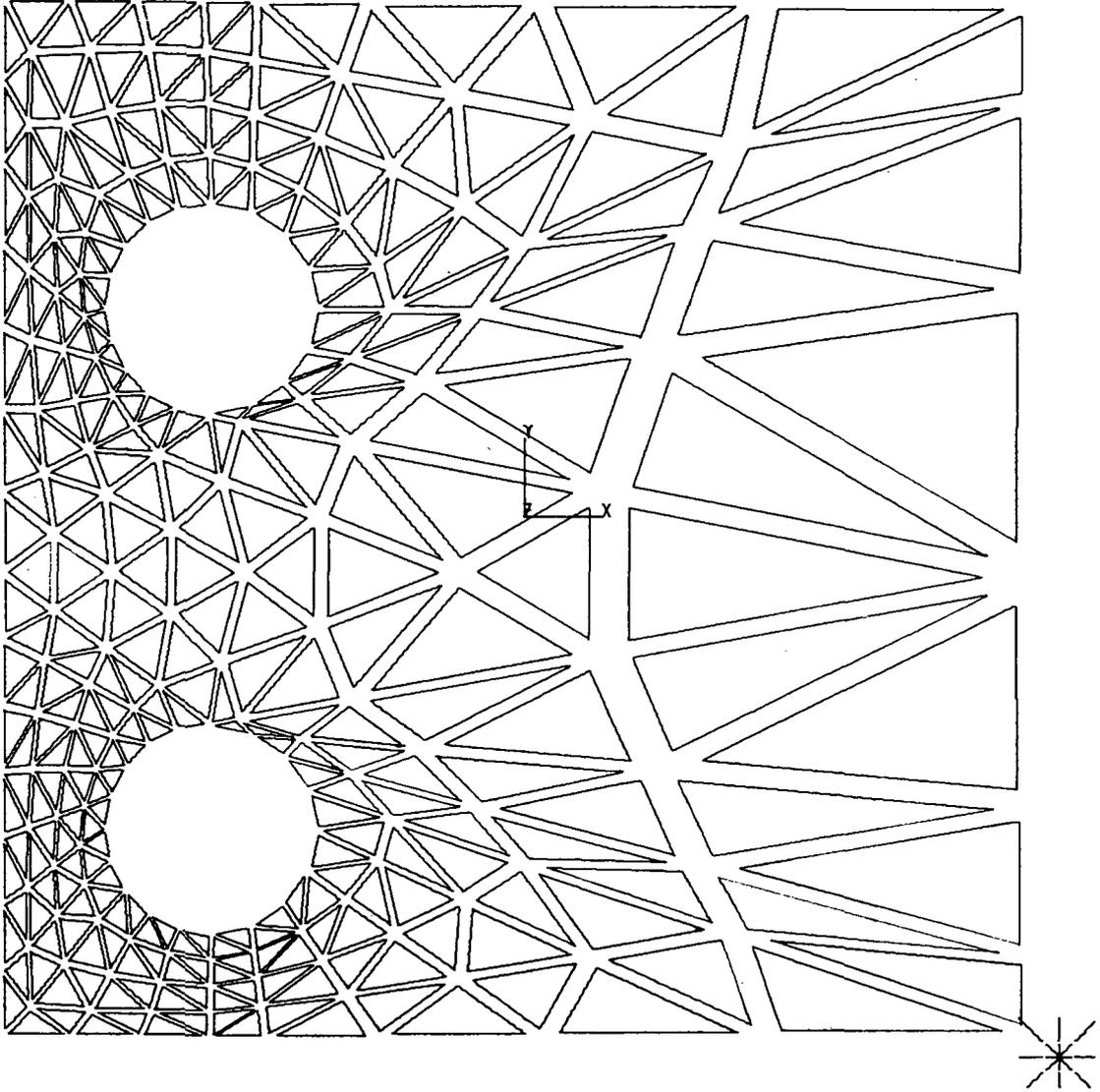


FIGURE 7. ELEMENT SHRINK OPTION.

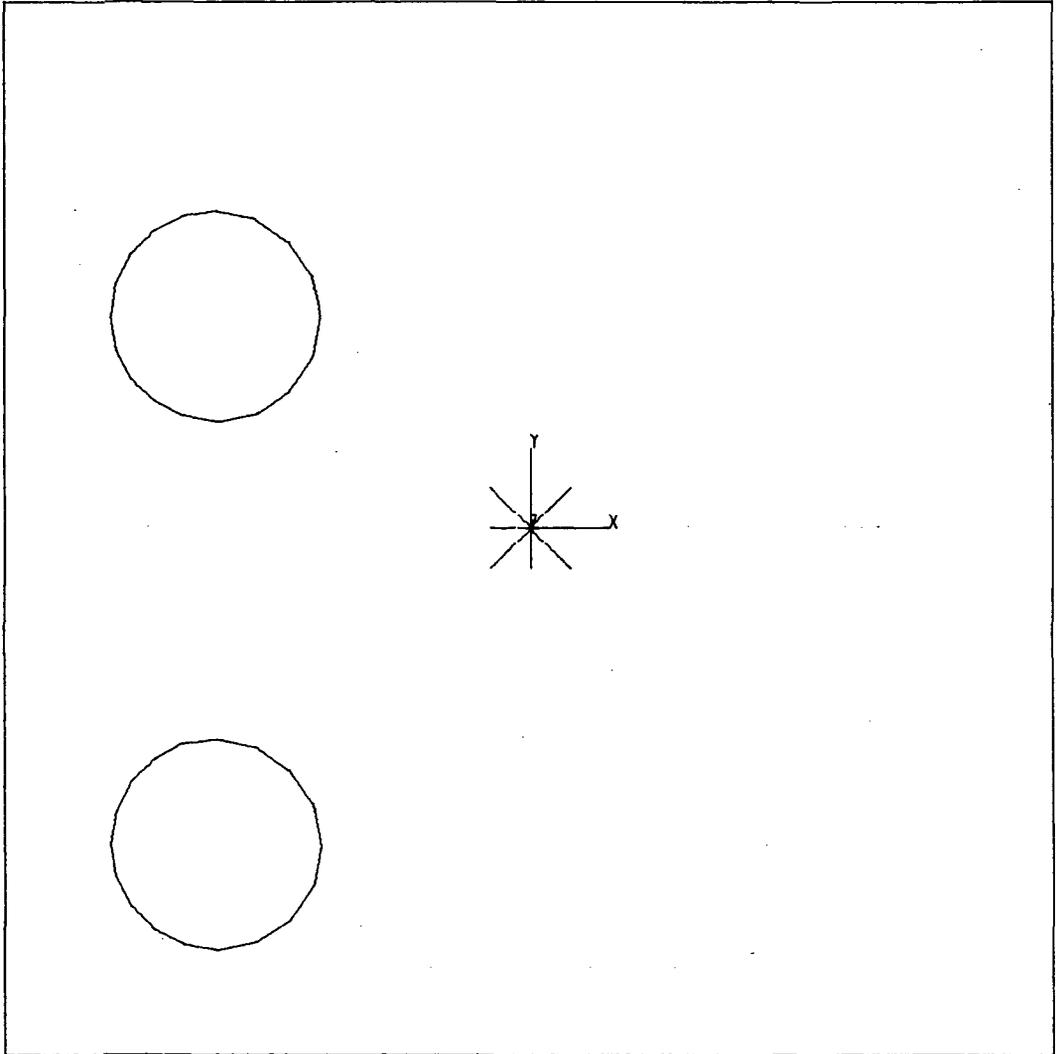


FIGURE 8. BOUNDARY SEGMENT DISPLAY.

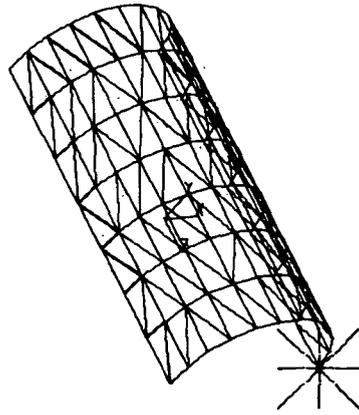
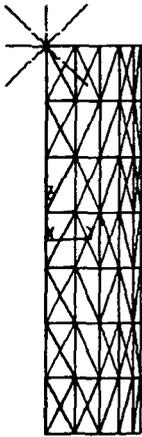
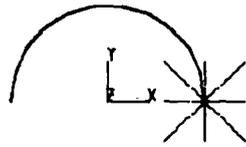
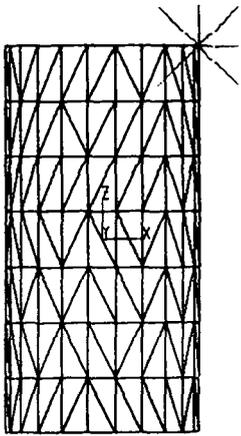


FIGURE 9. FOUR VIEW CAPABILITY.

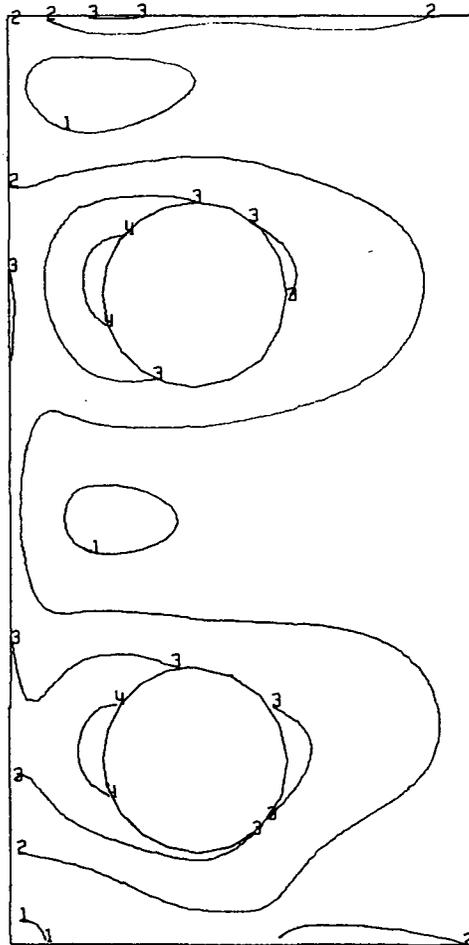


FIGURE 10. CONTOURS AROUND HOLES.

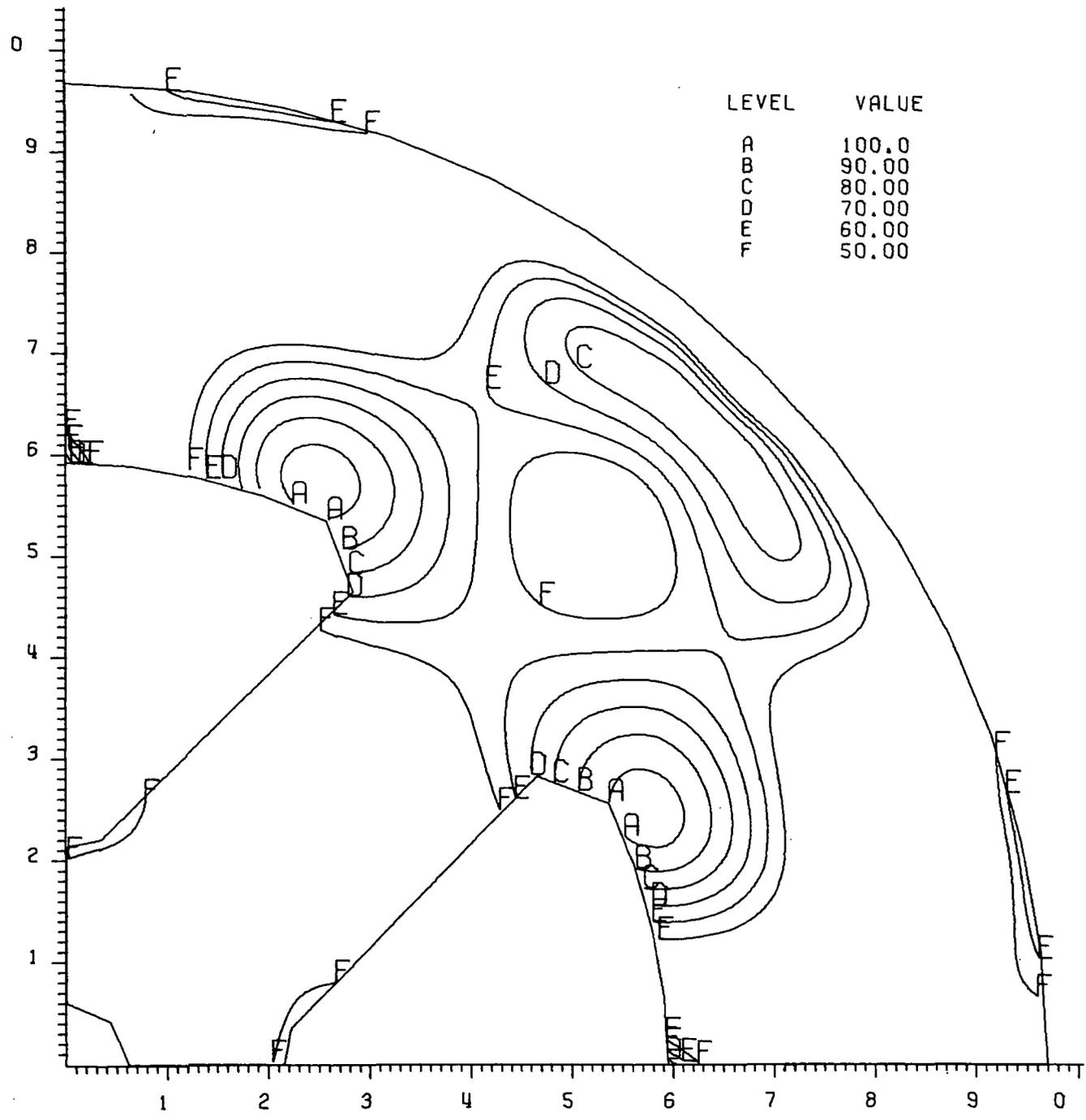


FIGURE 11. CONTOURS WITH AXIS.

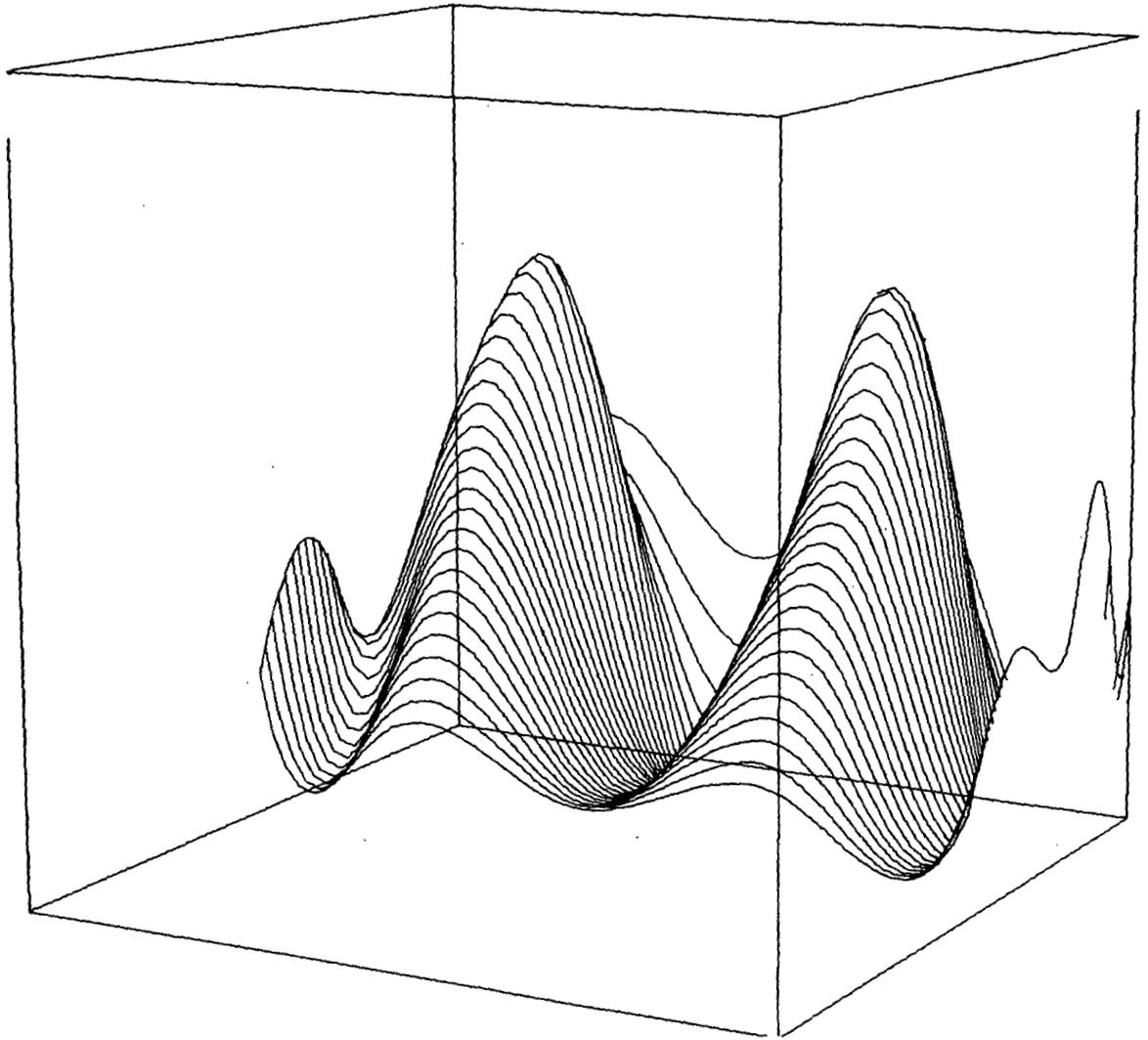


FIGURE 12. PERSPECTIVE OF STRESS SURFACE.

Page
intentionally
left blank

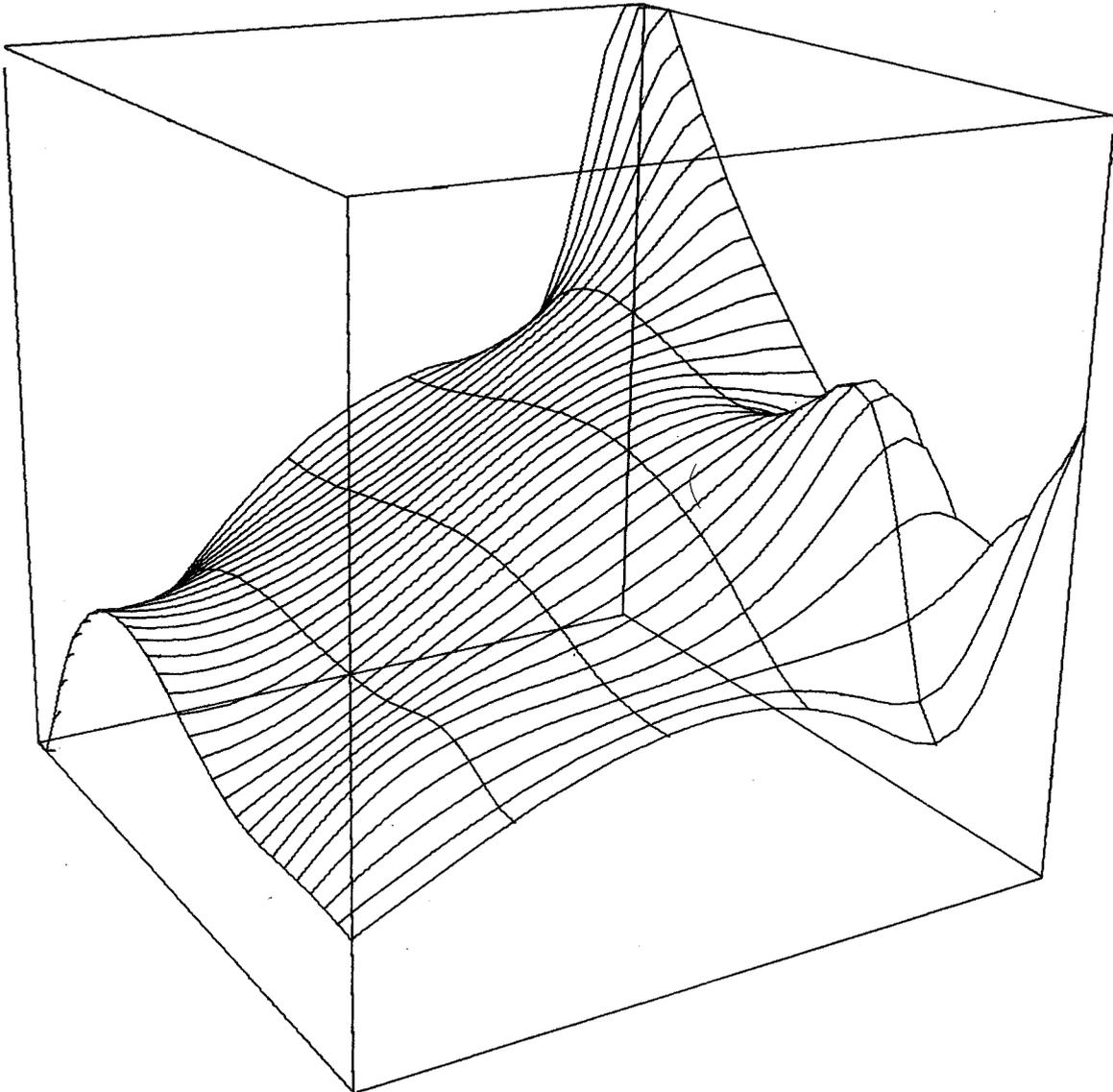


FIGURE 13. ANOTHER STRESS SURFACE.

**Page
Intentionally
Left Blank**

A STAND-ALONE INTERACTIVE GRAPHICS FINITE ELEMENT MODELING SYSTEM

Jeffrey Z. Gingerich

Marv M. Abe

Randall L. Vinecore

Gary J. Romans

Barry M. Ratihh

Tektronix, Inc.

SUMMARY

The finite element modeling (FEM) system described here is an interactive graphics system designed for use with an intelligent terminal. It offers an economical alternative to the conventional methods (hand compilation and remote terminals) of 2 and 3-dimensional model generation.

Using local computing and local storage, this FEM system allows the operator to create and display 3-dimensional models entirely off-line from the host computer. This eliminates costly on-line computer and host-processing time, and makes the user more effective by greatly reducing computer response time.

The system allows model generation by down-loading node and element data from the host computer or by using the system's model generation operations for digitizing and automatic creation. The system then generates completed model information in the desired analysis package's bulk data format and transmits it to the host computer for finite element analysis.

INTRODUCTION

In the past twenty years, finite element analysis has become a widely accepted design technique. The use of computers made feasible this numerical analysis and recently has assisted in the preparation and reviewing of data for finite element analysis.

Model generation is definitely the most time-consuming part of the finite element analysis (FEA) of a three-dimensional object. There are three distinct ways to do that modeling:

- hand compilation.
- timesharing (remote) graphic terminal modeling.
- off-line (stand-alone) interactive graphic terminal modeling.

See figure 1.

The first alternative, *hand compilation*, is very tedious and time consuming. It is also extremely error-prone. A misplaced decimal can go totally unnoticed until a preview geometry plot graphically reveals the error or, worse yet, an unexpected analysis result voids an expensive computer run.

The second choice, using a *time-sharing graphic terminal*, offers the convenience of node digitization from drawings, "automatic creation" commands and step-by-step visual verification of the modeling process. However, remote graphic terminals require constant line connection to a host time shared computer. Constant line connection is not only costly, but is also slow for graphic displays and command executions, especially if the tie-line data exchange rate is 1200 baud or less.

The third choice, using an *off-line intelligent graphic terminal*, offers the same advantages as a remote graphic terminal: ease of use and constant visual verification. Additionally, the stand-alone system eliminates the expense and slow response of the host computer.

A FEM SYSTEM

Three Tasks

Figure 2 shows the equipment used in the off-line FEM system (a Tektronix 4081) described here. The system has three main tasks:

- generate node/point spatial data.
- define element connectivity and some property information related to the nodal data.
- create the model bulk data for finite element analysis (FEA).

Node/Point Data

Nodal and element creation is handled by the system in the off-line mode. This independence from the timesharing service is complimented by many automatic creation commands that make model generation quick and efficient. Nodes can be digitized from drawings, down-loaded from the host computer, automatically generated by command, or typed in if high resolution is a must.

Element Connectivity

Element connectivity can also be downloaded from the host computer, automatically generated by command, or can be individually specified, graphically, using this sytem. Many display features allow the model generation to be visually verified. The 3-dimensional rotations, scaling and selective displaying of the model also allow more convenient model generation by selecting the proper view. Node numbering, element connectivity orientation, and element libraries are user defined and controlled.

Bulk Data

The model bulk data is also generated by the system in the off-line mode. The model information for nodes, elements and properties is compiled and formatted for any FEA package through the system's bulk data formatter module. This formatting module retrieves the model information, and compiles and formats it into the card image required by the selected analysis package. The formatting is user controlled by simple but powerful commands. The bulk data file is written onto disk storage in an 80-column, card-image format which may be transmitted to the host.

Operation

The operation of the FEM system is shown in figure 3.

The data loader is useful for generating models from existing bulk data decks or other sources. It can also be used as an initial source of node and element data. The model generation module can then edit the nodes, properties and connectivity when necessary.

Flexibility

The system makes use of the intelligent terminal's capabilities, and it is as easy to use as possible while still being powerful. The operations are all repeatable without repeating commands. For example, if the user wants to

digitize nodes, all that need be done is to tell the system to digitize nodes until done. The compliment of this capability is the ability to bail out of the current operation at any time. For example, if the user is creating a 20-node element with this system, and discovers that he doesn't need the element after all, he doesn't have to complete all 20 nodes and delete the element. He can bail-out at any time.

The operations are listed in a menu structure that is selected by positioning a box around the commands displayed on the terminal screen. The box and a cursor for selecting parts of the model are controlled by moving a pen on a digitizing tablet. When the user pushes the pen point down over a point on the tablet, the operation or the selection is transmitted to the system for processing.

The Screen Format

The display screen is sectioned into the four following areas: the static menu, the dynamic menus, the prompt band and the work area. See figure 4.

The *static menu* lists the main operations available to the user for model generation and display. The *dynamic menu* lists sub-operations for the selected main operation. The prompt band is a message space that displays the current status of the system. This area constantly tells what operation the user is in, what the system is expecting, or what the user has done wrong. The work area is the largest display area.

Example User's Session

The following example shows how the system creates a model. (Minor operational details are not discussed.)

The Object

The object to be modeled is a pipe elbow with a 60 cm material center diameter cross-section, a downward spiral of 3 cm every 20° along the 50 cm radius.

The Model

The model is composed of quadrilateral elements (QUAD2). The user may choose to digitize in the circular section of the pipe or, in this example, the user may employ the node creation commands to generate the circular section. By typing the values of $x = 80$ cm, $y = 0$ cm, $z = 0$ cm, the first node is generated

and displayed. This first node is then copied by using the COPY operation with the rotation parameters $x = 0^\circ$, $y = 0^\circ$, $z = 20^\circ$, new origin $x = 50$ cm, $y = 0$ cm, $z = 0$ cm and 17 iterations to produce the circular section as displayed in figure 6. Now nodes 1 thru 18 are copied 9 times with rotation parameters $x = 0^\circ$, $y = 20^\circ$, $z = 0^\circ$, translation parameters $x = 0$ cm, $y = 3$ cm, $z = 0$ cm and nine iterations. The resulting automatic node generation is shown in figure 7.

Commands

There are three major types of commands for model generation: environmental display, utility, and model generation. Each category has a series of operations and sub-operations. The operations are always displayed (in storage mode) in the static menu as shown in figure 8. The sub-operations are displayed (in refresh mode) in dynamic menus when the corresponding operation is requested by the user.

Table 1 is a complete list of all the operations and suboperations available with this FEM system.

Quadrilateral Elements

After completing the nodes, the user can next create the quadrilateral elements by indicating a master set of nodes that define the element's connectivity. By using the master set as a template, the elements can be generated automatically as shown in figures 9, 10, and 11. The pipe elbow model has 162 nodes and 144 elements and takes about 20 minutes to generate. The geometry of the model is visually correct, so the bulk data deck can now be generated.

Card Image

The bulk data formatter portion of the system allows many different card formats to be output from the system's data base. This is accomplished by using a command language to format the card image. The commands used for this model need to generate the nodal point data and the quadrilateral elements. (The user created these commands beforehand and stored them in a disk file that generates NASTRAN formatted card images.) The two commands used for the blade model are:

generation command

```

GEN, NODE, ALL, generate all nodes;
  1, A4, 'GRID', grid type;
  9, I8, ID, grid identification number;
 25, F8.3, X, x coordinate;
 33, F8.3, Y, y coordinate;
 41, F8.3, Z, z coordinate/

GEN, QUAD2, ALL, generate all quad2 elements;
  1, A6, 'CQUAD2', element type;
  9, I8, ID, element identification number;
 17, I8, P1, property card id. number;
 25, I8, N1, grid point number 1;
 33, I8, N2, grid point number 2;
 41, I8, N3, grid point number 3;
 49, I8, N4, grid point number 4/

```

card field definition syntax:
card column start, format,
internal variable or
constant, optional command

These two commands tell the system to generate the nodes and plates in the NASTRAN format. (In the example above, properties and control cards aren't shown.)

The command language allows the user to access data in the system model data base and output it in card format according to the card field specified. The internal variables used by the bulk data formatter are listed in Table 2.

Editing

The user should now edit the bulk data deck for the properties, loads, and control cards that were not handled by the system. Then the system transmits the data to the host timeshared computer for analysis, thus completing the pre-processing.

Table 1. The complete menu of operations and suboperations available with this FEM system.

Model Generation Commands

Model	
Cre	create model
Del	delete model
Get	retrieve model
Zone	
Cre	create zone
Del	delete zone
Get	retrieve zone
Lim	set node range for zone
Element	
Cre	create element
Del	delete element
Mod	modify element
Cpy	copy master element to create elements
Node	
Lin	automatically create nodes on line defined by two nodes
D-C	digitize nodes and create connectors at same time
Dig	digitize nodes
XYZ	type in coordinates for node creation
Del	delete node
Mod	modify node coordinates
Num	set node numbering scheme for generation
Ren	renumber a node
T-O	orient tablet to drawing
Property	
Cre	enter or replace a floating point number in the property table
LPT	list the property table
N-A	associate a range of nodes with a property
N-D	disassociate a range of nodes from its properties
N-L	list a range of nodes and their properties
E-A	associate a range of element numbers with a property
E-D	disassociate a range of elements from their properties
E-L	list a range of elements and their properties
Arc	
Cre	create arc
Connector	
Cr2	create connector by detecting two nodes
Cr1	create connector by detecting one node (connector goes to last node detected)
Del	delete connector
Mul	create multiple connectors between node sequences
Copy	
Org	copy nodes, elements, connectors
Trn	new origin for copy transformation
Rot	translate on the 3 axes
Scl	rotate on the 3 axes
Iti	scale about the 3 axes
Go	iteration number
	start the copy

Environmental Display Commands

Rotate	Rotate model about any of the model axes
X	
Y	Rotate model about the screen axes
Z	
Cont	Continuously rotate about screen axes
ZoomIn	Scale up model display
ZoomOut	Scale down model display
Paint	Display model
Options	
ZFg	turn Z-plane clipping on and off
Z-P	set front and back Z-plane values for clipping
Vew	adjust viewpoint for perspective projections
Iso	turn isometric projection on and off
Zon	control zone display range
Ety	control element display type range
Ela	control element tags display
ENo	control element display range
Arc	control node display range
NLa	control node tags display
Con	control connector display

Utility Commands

Status	generate model status report
Help	generate help listing of commands
End	terminate session
Geometry	
Loc	type out node id's and coordinates
Dis	determine and type out distance between nodes
Tol	set cursor locate tolerance
Misc	
ElmLib	
Cre	create element library entry
Del	delete element library entry
Mod	modify
Dis	display element library entries
Men	redisplay the static menu
TBW	specify tablet window size
PLOT	generate plot of model

Table 2. Internal variables used by the bulk data formatter.

INTERNAL VARIABLE	DATA BASE INFORMATION	DATA TYPE
M	Model name	character
TN	Total number of nodes	integer
TT	Total number of element types	integer
TEn	Total number of elements where n is element type (1-60)	integer
GPn	General property value array where n is array location (1-97)	real
ID	Node or element identification number/tag	integer
X	Node X value	real
Y	Node Y value	real
Z	Node Z value	real
ET	Element type number	integer
EN	Number of nodes in element	integer
Nn	Element node points where n is element node number (1-64)	integer
Pn	Node or element property value (1-2)	real

Modeling: Three Approaches

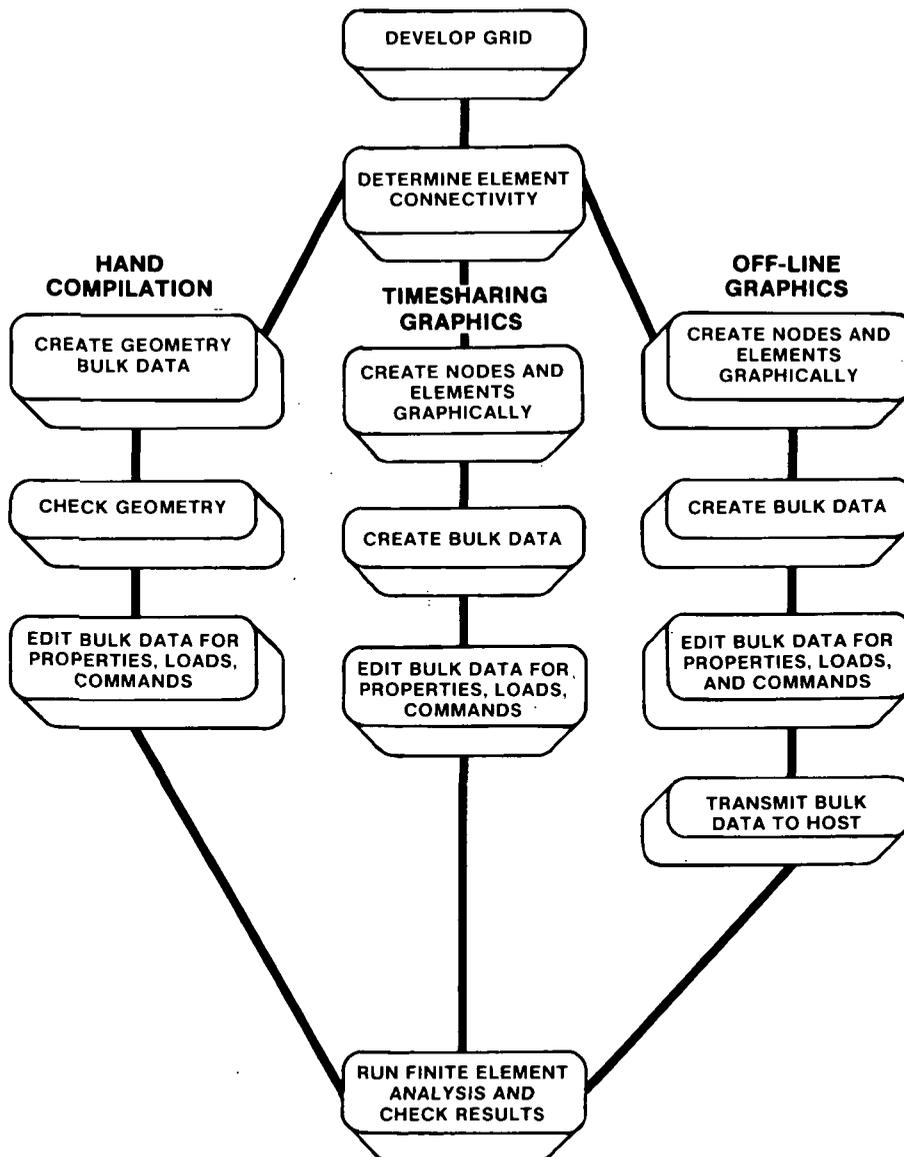


Figure 1. There are three main approaches to modeling. Hand compilation is tedious, time consuming, and prone to error. Using timesharing graphics offers many advantages, but it is costly. Using off-line graphics (with a stand-alone terminal) offers the advantages of timesharing graphics, but at greatly reduced cost.

FEM Hardware System

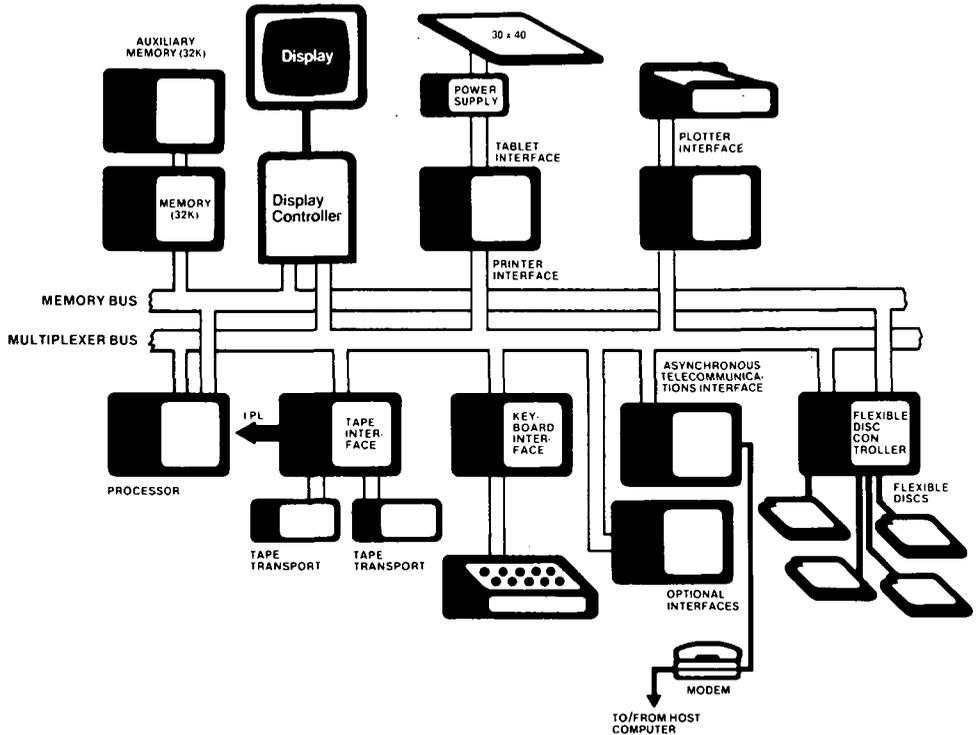


Figure 2. The equipment used for the FEM system described.

FEM System Operation

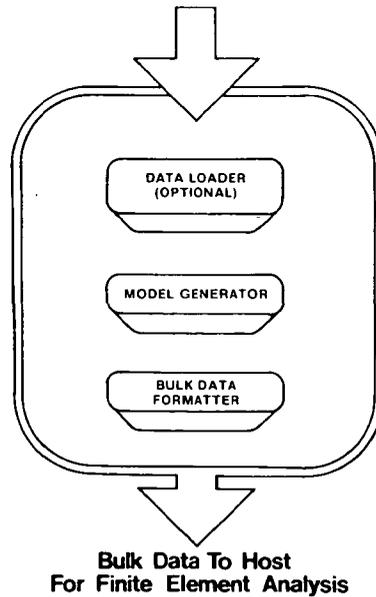


Figure 3. Finite element modeling system operation.

Display Screen Format

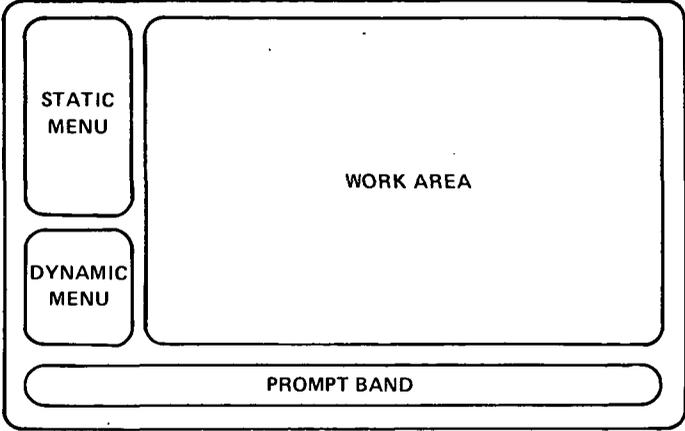


Figure 4. Display screen format.

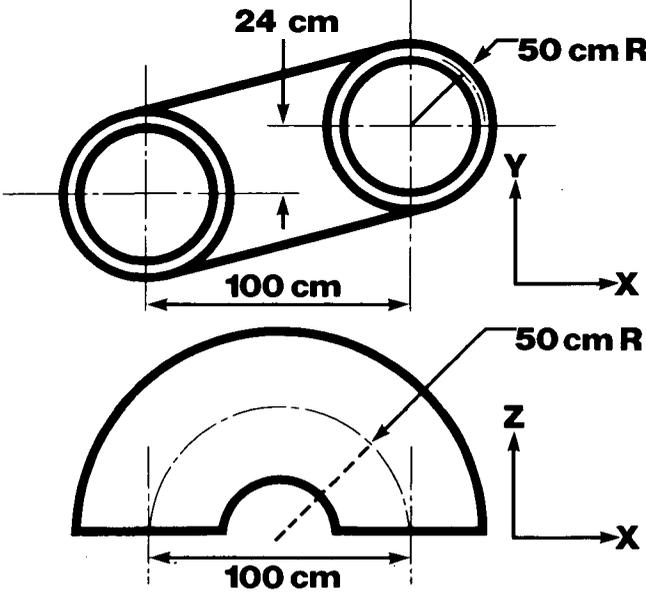


Figure 5. The example pipe elbow.

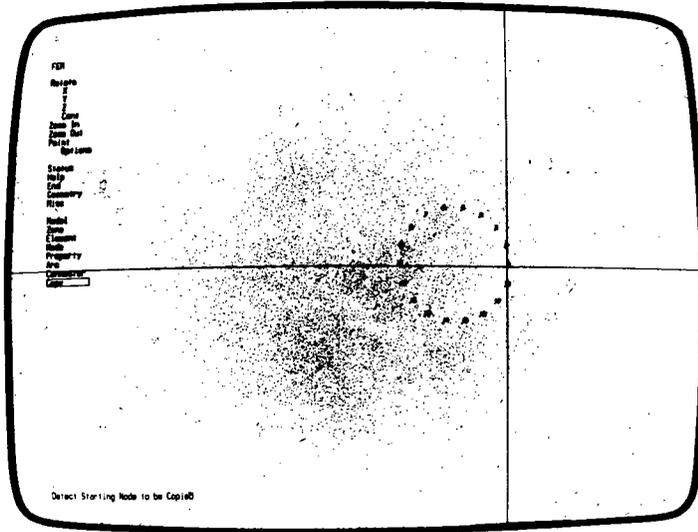


Figure 6. Pipe elbow, initial node generation.

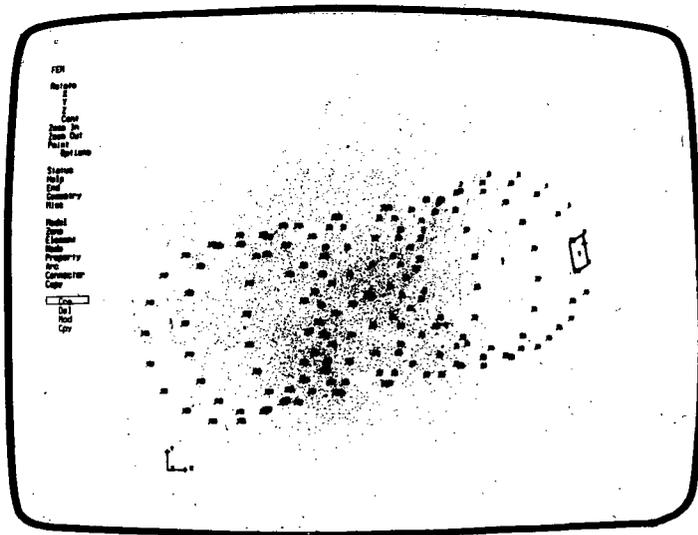


Figure 7. Pipe elbow, final node generation.

FEM	
Rotate X Y Z Cont	Environmental Display Commands
Zoom In Zoom Out Point Options	
Report Help End Geometry Misc	Utility Commands
Model Zone Element Node Property Arc Connector Copy	Model Generation Commands

Figure 8. The static menu is a display of the operations available in each of the three categories of commands.

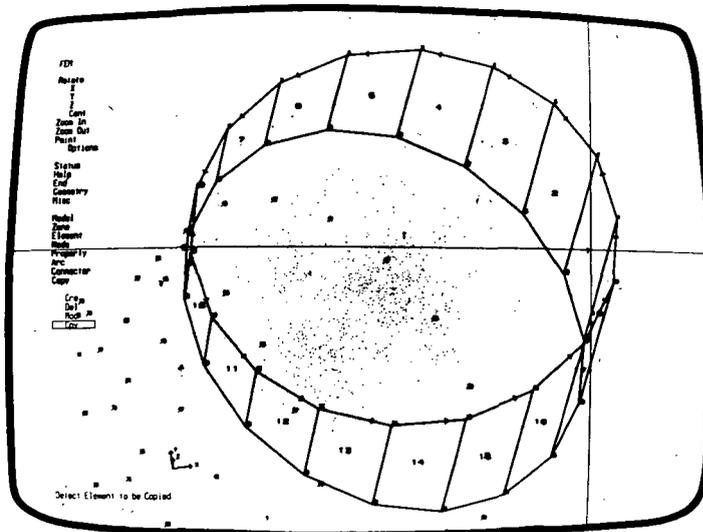


Figure 9. Pipe elbow, element generation.

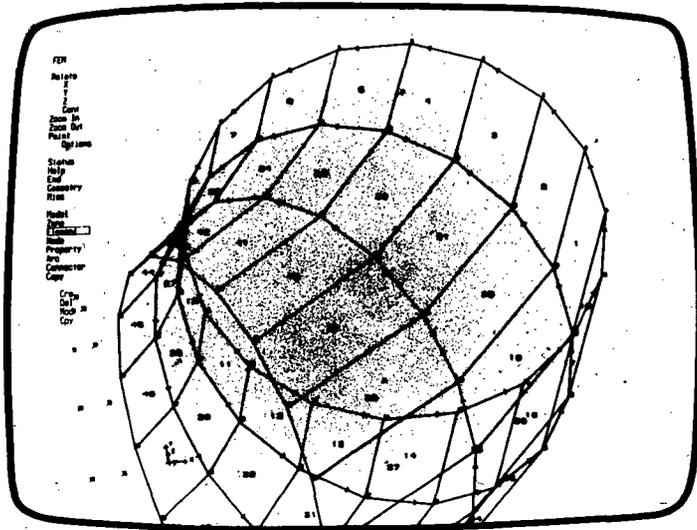


Figure 10. Pipe elbow, element generation.

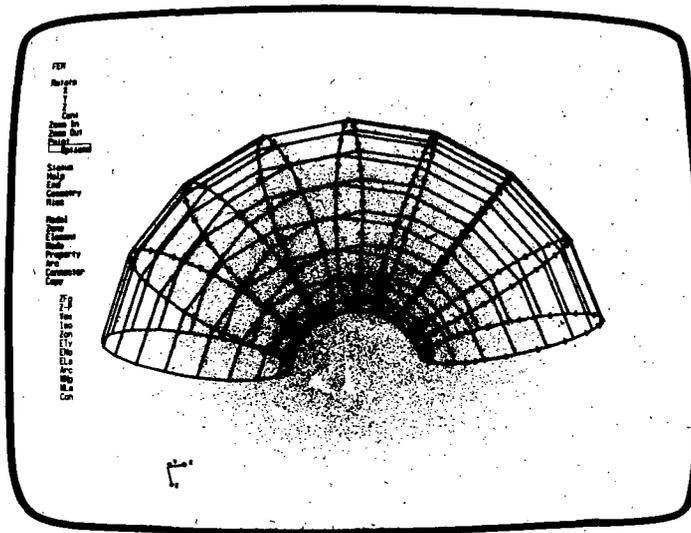


Figure 11. Pipe elbow, completed node and element generation.

RINA - AN INTERACTIVE SYSTEM FOR THE RAPID INTERPRETATION OF NASTRAN RESULTS

A. I. Raibstone and A. Pipano
Israel Aircraft Industries Ltd.

SUMMARY

RINA is a general post-processor system which performs data reduction, post processing and post analysis of NASTRAN results. RINA can be run in either batch or time-sharing mode. The system, which is an expansion of the NASDAT program, performs the following tasks: (a) Scan of extreme values for displacements, forces, stresses and margins of safety (b) Computations of envelopes for displacements, forces, stresses and margins of safety (c) Computations of allowables and margins of safety and (d) Generation of NASTRAN tables needed for reanalysis computations. Usage of all these options provides the analyst with an efficient tool for the study and interpretation of NASTRAN analysis results and their presentation for project documentation.

INTRODUCTION

For the efficient design of complex structures it is necessary to investigate several structural configurations. Consequently, the analyst is bound to devote a substantial portion of his time in visual scanning, processing and interpretation of a large volume of finite element analysis results. This process, being time-consuming and error-prone, leaves little time for engineering-oriented decision making.

The way to alleviate this problem is to automate the scanning and interpretation of results and let the computer program suggest to the analyst possible modifications of element stiffnesses in order to obtain a distribution of material which conforms with the existing flow of internal forces. The final result should be a structure which has a certain optimum stiffness configuration based on purely engineering-oriented criteria.

Israel Aircraft Industries' NASDAT system (Reference 1) has been a successful attempt to integrate the automated scanning, interpretation, post-analysis and graphic presentation of finite element analysis results. For this reason, the NASDAT system has been expanded and modified into RINA in order to meet the growing demands of the structural analyst in the company.

The new system, RINA (Rapid Interpretation of Finite Element Analysis) has presently the following features:

1. The system may be operated either in batch mode or in time-sharing mode (T/S); in which it is easily called on-line by the analysts in a dialogue oriented manner.
2. The system requires only 20K central memory computer words for executing a job whether large or small.

This paper describes briefly the RINA system and its capabilities, and demonstrates its application utilizing most of its features on an actual aircraft structure.

DESCRIPTION OF THE RINA SYSTEM

The execution of the RINA program may be divided into four main phases:

- a) In phase 1, regular NASTRAN output data blocks are sorted and stored in a compact form on a disk file for subsequent use. This file may be saved after termination of the RINA execution. Thus, a restart file, which contains all necessary information from the NASTRAN analysis, is created to be used in phases 2, 3 and 4.
- b) In phase 2, the packed and sorted NASTRAN output is scanned and efficient data reduction is performed in accordance with the user's requirements.
- c) In phase 3, post-analysis - including computations of allowables and margins of safety - is carried out.
- d) Finally in phase 4, NASTRAN tables are updated for reanalysis computations and stiffness suggestions are made.

RINA may be run either in conjunction with NASTRAN as a single job, or as a separate run following a NASTRAN analysis for which the necessary output data blocks have been saved. RINA may be restarted as many times as required.

RINA CAPABILITIES

The following is a brief description of the different capabilities presently available in RINA:

a) Maximum and Minimum Values

The user may define a given set of elements and/or gridpoints from which the program will single out and identify those elements and grid points bearing extreme values of force, stress and/or displacement; the magnitude of these extrema is also produced. In addition, it is possible to specify the upper and lower bounds beyond which the search for maximum and minimum values is to begin.

b) Envelopes of Displacements, Forces, Stresses and Margins of Safety

When analyzing several loading cases or checking various boundary condition configurations, the user has the option of obtaining envelopes of displacements, forces, stresses and margins of safety for a specified set of subcases, grid points and/or elements.

c) Reduced Output

In some cases the user may require the results in a certain region of interest to be printed out separately in a specified sequence. In addition, if upper and lower bounds on the output values are also defined, then only those elements of the set with results above these bounds are printed. The user may also obtain results which are within the above defined upper and lower bounds.

d) Combined Loading Conditions

NASTRAN analysis results, obtained for any previously defined loading condition, may be linearly superimposed in RINA. This option enables the analyst to check his structure for any possible combination of loading configurations, without the burden of obtaining superfluous results, as is the case with the SUBCOM option in NASTRAN. Furthermore, the need for large central memory requirements is avoided.

e) Computations of Allowables and Margins of Safety

The margins of safety of any Rod, Shear Panel, Membrane and Plate element can be computed using allowables either defined by the user or automatically computed by the program. For the latter option the user may either define the section properties via manual input or allow the program to retrieve the geometrical and mechanical properties of these sections from tables generated in the regular NASTRAN analysis.

f) Reanalysis

At the user's request the program will generate a new, ready-to-run NASTRAN input file for the reanalysis of the regular structure employing a direct modification procedure (Reference 2). The program updates the Element Summary Table (EST) and Material Property Table (MPT) and computes the Boolean transformation matrices which relate the modified degrees of freedom, affected by the modifications in the stiffness matrix, to the unmodified degrees of freedom. Finally, the appropriate DMAP Alter package is generated.

g) Stiffness Suggestions

At the user's request the program will issue suggestions for possible new stiffness properties for some regions in the structure in order to keep the level of margins of safety within desired limits. The necessary stiffness requirements are either input by the user or computed automatically by the program, employing the options of Allowable Stresses and a slightly changed version of the Reduced Output option.

THE RINA INPUT

The main input to RINA comes from NASTRAN data blocks generated in the regular analysis. In order to ensure that this information is readily available for RINA the following DMAP statements, for static analysis, must be included into the Executive Control Deck of the NASTRAN deck:

1. ALTER 121
2. OUTPUT1 LLL,ULV,KFS,,//C,N,-1/C,N,8/C,N,USERT1 \$
3. OUTPUT1 USET,EQEXIN,SIL,BGPDT,//C,N,0/C,N,8 \$
4. OUTPUT1 GM,CSTM,YS,PS,KSS//C,N,0/C,N,8 \$
5. OUTPUT2 MPT,EST,,//C,N,-1/C,N,12/C,N,USERT2 \$
6. OUTPUT2 CASECC,GPL,USET,CSTM,YS//C,N,-1/C,N,11/C,N,USERT3 \$
7. OUTPUT2 MPT,EST,,//C,N,0/C,N,11 \$
8. OUTPUT2 OUGV1,OQG1,OEF1,OES1,//C,N,0/C,N,11 \$
9. ENDALTER

It is worthwhile noting that statement 7 saves the necessary geometrical and mechanical properties of the structure on NASTRAN Fortran file UT1 and Statement 8 saves all the analysis results.

As previously mentioned the main input to NASDAT comes from the above data blocks stored on file. However, additional input is required to specify the user's request regarding data reduction and post-analysis of results.

The input for the user's requests is prepared in one of two different formats depending on whether RINA is used in batch or in T/S mode. The input for the batch option is described in Reference 1. Input for the T/S mode of RINA is performed in a dialogue oriented manner which is self explanatory. Examples for the input options and formats are shown in the following section.

NUMERICAL EXAMPLE

The capabilities of RINA, employing the T/S options, are demonstrated on the results obtained from the finite element analysis of the engine support structure of the Westwind 1124 aircraft. The structure was idealized using membrane elements for the skin panels and rod elements for the stringers. The complete mathematical model, shown in Figure 1, was represented by 360 CONROD elements, 450 CQDMEM elements and 420 GRID points representing 1200 unconstrained degrees of freedom. The structure was subjected to 10 loading conditions.

Based on engineering considerations related to the structural configurations and loading conditions the analyst has asked for and has obtained the following items: (presented here as an example)

- (i) Information regarding extreme values of the structure's displacements in the Z-direction (3-direction) for all the loading conditions. This request enabled the analyst to check in a quick and concise manner if the displacements obtained in the regular analysis were within prescribed limits. In the user's request, shown in Figure 2, the analyst defined upper and lower bounds, such that only displacements which were larger than 0.110 in (2.75mm) in magnitude were output. When the output seems to be too voluminous to be printed on the low-speed console it may be disposed to a high-speed printer. The output for the request explained above is shown in Figure 3.
- (ii) Information regarding critical forces and/or stresses in the rod elements. In this request the user defined the scan to be performed on all the element for subcases 1,2,3 and 9. Output is to be printed for all the elements whose axial forces are less than -4000 lbs (1815kg) and larger than 4000 lbs and whose axial stresses are greater than 10000 psi (7 kg/mm^2) in magnitude. The input and output formats for this request are shown in Figures 4 and 5.

- (iii) The analyst then asked for envelopes of the von Mises equivalent stresses for all the membrane elements covering all the subcases defining 20000 psi (14 kg/mm²) as the upper bound. The input and output for this example are shown in Figures 6 and 7. With the help of this option the analyst quickly detected that SUBCASE 2 and SUBCASE 5 were critical. As a result, a request for more detailed information concerning these two subcases was obtained via the batch option of RINA. An output example is shown in Figure 8.
- (iv) In the next step a number of panels which displayed high stress levels were selected. Allowables and margins of safety for the three selected panels were computed (for SUBCASE 2 and SUBCASE 5) employing RINA's "MS" option. In this example the analyst requested that the program retrieves the geometrical and mechanical properties of the defined finite elements. When margins of safety for a large number of elements are to be computed then RINA's batch option should be utilized. The input and output formats for the above example are shown in Figures 9 and 10.
- (v) Having studied some of the results, the analyst may now decide to investigate the effect of superimposing a number of loading conditions. In this example the analyst has superimposed three loading conditions and requested output which exceeded the values defined by the MAX/MIN SET values in the input questionnaire. As a result he obtained 11 elements which exceeded the defined extreme values. This option is basically equivalent to the SUBCOM option within NASTRAN. However, while the SUBCOMs have to be defined prior to the NASTRAN analysis or in a restart job, in RINA the option may be activated whenever needed and may be coupled with a data reduction request. The input for this capability is shown in Figure 11 and its output is presented in Figure 12.
- (vi) Finally, after the analyst has studied the structure's overall response, some structural details may have to be investigated. In this example the analyst decided to check the effectiveness of two access panels. These panels were defined in the regular analysis as being fully effective. In this case, the analyst decided that the thickness of the access panels should be reduced in ten steps. The final step simulating an equivalent zero stiffness (cut-out) for the four finite elements. As a result, the program generated a ready-to-run NASTRAN input file including the necessary DMAP Alter package for the reanalysis run. The input and the output for this example are shown in Figures 13 and 14.

Having concluded all the tasks described in the above example, the analyst now decides which relevant analysis results have to be obtained in order to be included in his final technical report. He then prepares a Data Deck for RINA which is submitted employing RINA's batch option.

The complete listing of the regular NASTRAN analysis results would have amounted to 600 pages of computer output for each analysis step (10 analysis steps were performed). The cycles of analysis, manual data reduction, interpretation, post-analysis and reanalysis would have required several man-months of project time.

However, by employing the capabilities of the RINA system, the same job - reducing and interpreting results, performing post-analysis computations and re-analyzing the structure - was performed within a few days. The complete set of results, representing all values of engineering interest, could be reduced to 20 to 50 pages, depending upon the user's request. The format of the output is such that it can easily fit into a technical report, without any additional effort on the part of the analyst, while giving a comprehensive description of the analysis results.

CONCLUSION

Scanning of analysis results, post-analysis and reanalysis, may be performed with the aid of RINA in an automated fashion, thereby eliminating all possible errors and waste of valuable man hours both of which are bound to occur when performing the above options in a manual and/or visual manner. As a result the analyst is free to devote a larger portion of his time to engineering-oriented decision-making based upon results obtained in an organized and comprehensive form.

The combined usage of RINA's batch and interactive capabilities provides the analyst with an efficient and convenient tool for the study of NASTRAN analysis results and their presentation for project documentation.

REFERENCES

- 1 Raibstein, A.I., Emil, S. and Pipano, A. : A Post-processor System for the Data Reduction and Postanalysis of NASTRAN Analysis Results. NASTRAN User's Experiences, NASA TM X-3428, Oct. 1976, pp. 555-580
- 2 Raibstein, A.I., Kalev, I. and Pipano, A. : Efficient Reanalysis of Structures by a Direct Modification Method. NASTRAN: User's Experiences, NASA TM X-3428 Oct. 1976, pp. 21-51.

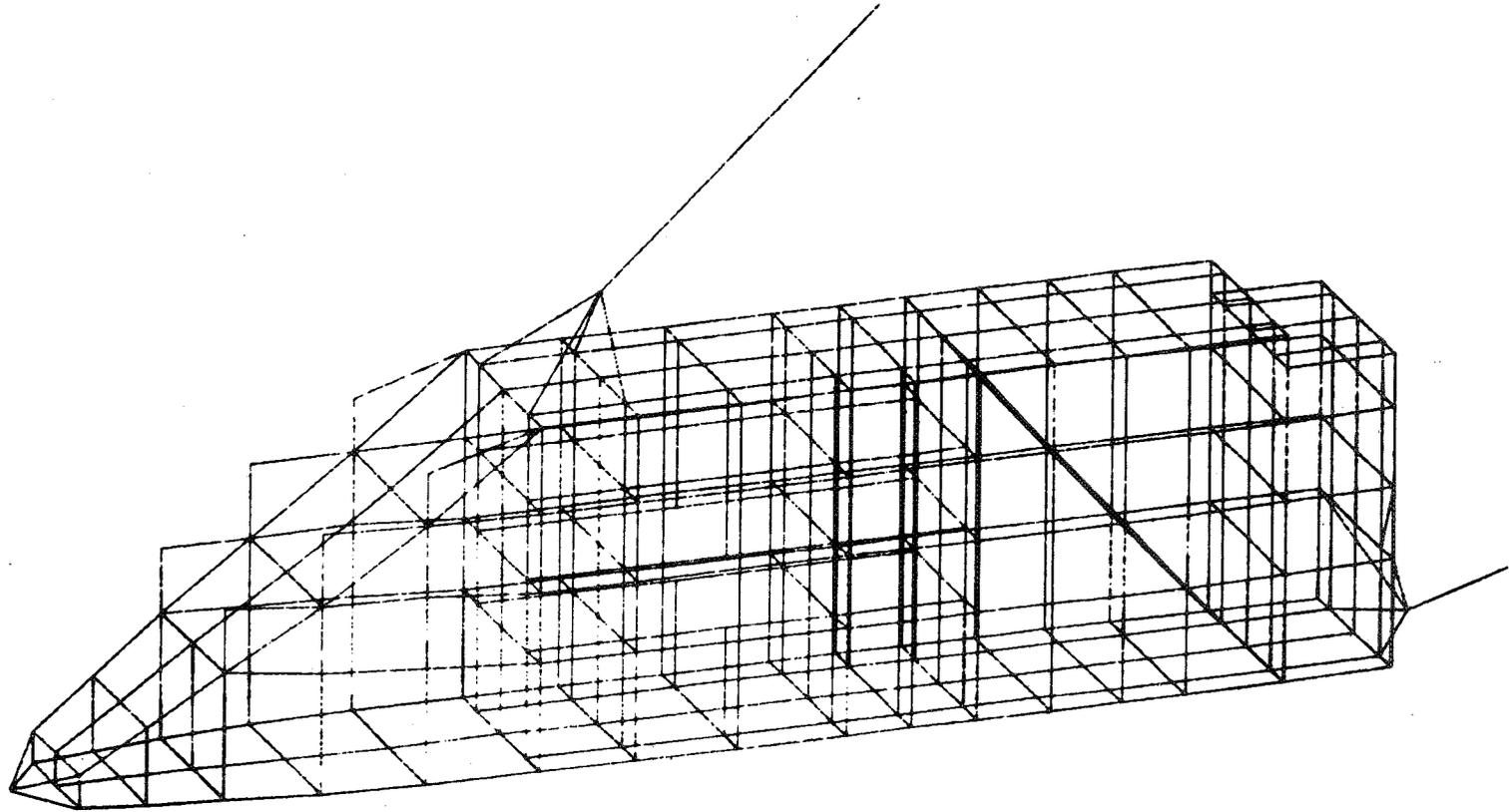


Figure 1 : Finite Element Model of Engine Support Structure

RIN31

```
*****
* WELCOME TO "P. I. N. A." *
* VERSION 3.1 (T/S) *
*****

DATE AND TIME: 77/07/21. 15.09.40.

DO YOU WANT TO CREATE A RESTART FILE ("YES"(Y) OR "NO"(N))
? Y
DO YOU WISH TO SAVE THE RESTART FILE ("YES"(Y) OR "NO"(N))
? Y
ENTER A PERMANENT FILE NAME FOR THE RESTART FILE
? DNASA

**NASTRAN OUTPUT WAS RECOVERED FROM FILE UTAPE **

ENTER DATA REDUCTION OPTION
OPTIONS=MINMAX, ENVELOP, REDOUT, SUPPOS, NS, ADVISE, REAN
? MINMAX
STRESS (S) OR DISPLACEMENT (D)
? D
ENTER DISPLACEMENT SET (A SET OF GRID NOS OR "ALL" (A))
? A
ENTER SUBCASE SET (A SET OF SUBCASE NOS OR "ALL" (A))
? 1

ENTER MIN/MAX VALUES FOR:
DISPLACEMENT IN T1 DIRECTION
? 0.0,0.0
DISPLACEMENT IN T2 DIRECTION
? 0.0,0.0
DISPLACEMENT IN T3 DIRECTION
? -0.11,0.11
R1 ROTATION
? 0.0,0.0
R2 ROTATION
? 0.0,0.0
R3 ROTATION
? 0.0,0.00
```

Figure 2 : Sample Input for Minimum and Maximum Values of Displacements, Option - MINMAX

```
YOU HAVE REQUESTED THE FOLLOWING OPTIONS:
EXECUTION OPTION - MIN/MAX ; DATA FILE NAME UTAPE
TYPE OF RESULTS - DISPLACEMENTS
THE FOLLOWING RESULTS WILL BE PRINTED:
T3 DISPLACEMENTS WHICH ARE LESS THAN -1.1000E-01 OR LARGER THAN 1.1000E-01
IN DESCENDING ORDER OF ABSOLUTE VALUE

SUBCASE 1
-----
0 MINIMUM (NEGATIVE) DISPLACEMENTS EXCEED INDICATED VALUES
5 MAXIMUM (POSITIVE) DISPLACEMENTS EXCEED INDICATED VALUES
DO YOU WANT DISPLACEMENTS TO BE PRINTED ("YES"(Y) OR "NO"(N))
? Y
GRID ID T1 T2 T3 R1 R2 R3
-----
293 3.5521E-02 9.1845E-02 1.1583E-01 0. 0. 0.
295 3.9135E-02 9.8229E-02 1.1581E-01 0. 0. 0.
301 4.5538E-02 1.0658E-01 1.1415E-01 0. 0. 0.
289 3.8496E-02 9.5894E-02 1.1049E-01 0. 0. 0.
292 2.9520E-02 8.0201E-02 1.1048E-01 0. 0. 0.

CONTINUE (C) OR QUIT (Q)
? C
```

Figure 3 : Sample Output for Option MINMAX

```

ENTER DATA REDUCTION OPTION
      OPTIONS=NINMAX,ENVELOP,REDOUT,SUPOS,MS,ADVISE,PEAR
? REDOUT
  STRESS (S) OR DISPLACEMENT (D)
? S
  ENTER ELEMENT TYPE (ANY OF THE NASTRAN ELEMENT NAMES)
? CONROD
  ELEMENT CONROD IS NOT INCLUDED IN YOUR ANALYSIS; TRY AGAIN
  ENTER ELEMENT TYPE (ANY OF THE NASTRAN ELEMENT NAMES)
? CONROD
  ENTER ELEMENT SET (A SET OF ELEMENT ID NOS OR "ALL" (A))
? A
  ENTER SUBCASE SET (A SET OF SUBCASE NOS OR "ALL" (A))
? 1,2,3,9
  DEFINE MIN/MAX VALUES FOR:
  AXIAL FORCES
  ? -4000.,4000.
  AXIAL STRESSES
  ? -10000.,10000.

```

Figure 4 : Sample Input for Reduced Output of Rod elements, Option - REDOUT

```

YOU HAVE REQUESTED THE FOLLOWING OPTIONS:
EXECUTION OPTION - REDOUT , RESTART FROM FILE TAPE6
TYPE OF RESULTS - FORCES AND STRESSES FOR ELEMENT CONROD
THE FOLLOWING RESULTS WILL BE PRINTED:
AXIAL FORCES WHICH ARE LESS THAN -4.0000E+03 OR LARGER THAN 4.0000E+03
AXIAL STRESSES WHICH ARE LESS THAN -1.0000E+04 OR LARGER THAN 1.0000E+04

```

SUBCASE 1		
11 ELEMENTS EXCEED INDICATED VALUES		
DO YOU WANT RESULTS TO BE PRINTED FOR THIS SUBCASE (YES (Y) OR NO (N))		
? Y		
ELEMENT ID	AXIAL FORCES	AXIAL STRESSES
164	-1.47080E+03	-1.13140E+04
167	-1.42330E+03	-1.09490E+04
193	-2.61720E+03	-1.04690E+04
313	-1.21080E+03	-1.00150E+04
314	-1.22730E+03	-1.02280E+04
320	2.80390E+03	1.12160E+04
1119	-3.12130E+03	-1.00690E+04
1120	-3.12950E+03	-1.00950E+04
1202	3.78940E+03	3.15780E+04
1427	5.07340E+03	2.02940E+04
1430	-5.07690E+03	-2.03080E+04

```

SUBCASE 2
26 ELEMENTS EXCEED INDICATED VALUES
DO YOU WANT RESULTS TO BE PRINTED FOR THIS SUBCASE ( YES (Y) OR NO (N) )
? N

SUBCASE 3
17 ELEMENTS EXCEED INDICATED VALUES
DO YOU WANT RESULTS TO BE PRINTED FOR THIS SUBCASE ( YES (Y) OR NO (N) )
? N

SUBCASE 9
0 ELEMENTS EXCEED INDICATED VALUES

CONTINUE (C) OR QUIT (Q)
? C

```

Figure 5 : Sample Output for Option REDOUT

```

ENTER DATA REDUCTION OPTION
                OPTIONS=NIN/MAX, ENVELOP, REDOUT, SUPOS, MS, ADUISE, PEAN
? ENVELOP
STRESS (S) OR DISPLACEMENT (D)
? S
ENTER ELEMENT TYPE (ANY OF THE NASTRAN ELEMENT NAMES)
? CQDMEM
ENTER ELEMENT SET (A SET OF ELEMENT ID NOS OR "ALL" (A))
? A
ENTER SUBCASE SET (A SET OF SUBCASE NOS OR "ALL" (A))
? A

ENTER MIN/MAX VALUES FOR:
STRESSES NORMAL-X
? 0.,0.,
STRESSES NORMAL-Y
STRES < ERROR, RETYPE RECORD AT THIS FIELD
? 0.,0.,
SHEAR STRESSES
? 0.,0.,
NON RISES STRESSES
? 0.,20000.

```

Figure 6 : Sample Input for Stress Envelopes of Membrane Elements, Option - ENVELOP

```

YOU HAVE REQUESTED THE FOLLOWING OPTIONS
EXECUTION OPTION - ENVELOP , RESTART FROM FILE TAPE6
TYPE OF RESULTS - STRESSES FOR ELEMENT CQDMEM
THE FOLLOWING RESULTS WILL BE PRINTED:
NON RISES STRESSES WHICH ARE LESS THAN 0.          OR LARGER THAN 2.0000E+04
29 ELEMENTS EXCEED INDICATED VALUES
DO YOU WANT STRESSES TO BE PRINTED ("YES"(Y) OR "NO"(N))
? Y

```

ELEMENT ID NO.	SUBCASE NO.	MAXIMUM VALUE
2305	5	2.73090E+04
2306	5	2.84380E+04
2307	5	2.39200E+04
2308	2	2.46300E+04
2321	5	2.27620E+04
2322	5	2.34850E+04
2323	2	2.43170E+04
2324	2	3.76800E+04
2333	4	2.34050E+04
2336	5	2.11220E+04
2349	4	2.27830E+04
2352	5	2.06490E+04
2353	2	2.24680E+04
2354	2	2.18600E+04
2369	2	2.04390E+04
2381	2	2.28390E+04
2397	5	2.51000E+04
2400	5	2.54500E+04
2401	5	2.16670E+04
2409	5	2.11270E+04
2412	2	2.30480E+04
2416	2	2.47990E+04
2424	5	2.07080E+04
2428	5	2.38370E+04
2465	2	3.02410E+04
2467	5	3.37740E+04
2468	5	2.67930E+04
2471	5	2.51080E+04
2472	5	3.36540E+04

```

CONTINUE (C) OR QUIT (Q)
? C

```

Figure 7 : Sample Output for Option ENVELOP

P R E F E R R E D S E Q U E N C E F O R Q U A D R I L A T E R A L M E M B R A N E E L E M E N T - C O D M E M -

FOR VALUES LESS THAN -5.0000E+03 -5.0000E+03 -4.0000E+03 0. IN ELEMENT SET
 AND VALUES LARGER THAN 5.0000E+03 5.0000E+03 5.0000E+03 5.0000E+03

S U B C A S E 5

EXTRACTION ORDER	ELEMENT ID.	STRESSES NORMAL -X	STRESSES NORMAL -Y	SHEAR STRESSES	VON MISES STRESSES
1	2222	-2.7642E+03	1.3020E+02	2.8080E+03	5.6277E+03
2	2241	-3.4088E+03	2.1499E+02	2.1397E+03	5.1122E+03
3	2242	-3.8151E+03	6.4618E+02	4.5587E+03	8.9322E+03
4	2243	-2.7814E+03	3.8418E+02	4.4519E+03	8.2711E+03
5	2251	1.4834E+03	-3.3634E+03	-1.5148E+03	5.0384E+03
6	2262	-2.4296E+03	1.7650E+03	2.7559E+03	6.0076E+03
7	2263	-1.5340E+03	1.2401E+03	2.8692E+03	5.5218E+03
8	2269	-2.9083E+03	-1.2161E+04	-3.3507E+03	1.2436E+04
9	2270	4.7331E+02	-1.0834E+04	-1.6665E+03	1.1448E+04
10	2271	3.7849E+03	-9.9465E+03	-9.5736E+02	1.2396E+04
11	2272	8.1750E+02	-1.2444E+04	-7.5364E+01	1.2873E+04
12	2281	-1.2270E+04	-9.2845E+02	8.0425E+03	1.8278E+04
13	2282	-3.9335E+03	4.3977E+02	-4.1470E+03	8.3059E+03
14	2283	-1.6732E+03	6.4651E+02	-6.2096E+03	1.0953E+04
15	2284	-3.7156E+03	-2.4511E+02	-8.1631E+03	1.4590E+04
16	2285	-1.1926E+04	-3.6062E+02	2.9117E+03	1.2787E+04
17	2286	-5.4232E+03	-8.8417E+02	-2.0452E+03	6.1601E+03
18	2287	-1.3536E+03	-7.0087E+02	-3.2045E+03	5.6728E+03
19	2290	-7.0166E+03	-1.4222E+03	-2.2740E+03	7.5359E+03
20	2303	-1.2999E+03	2.4349E+03	2.8160E+03	5.8798E+03
21	2307	6.1669E+03	-1.7876E+04	-5.8970E+03	2.3920E+04
22	2313	-1.3814E+03	6.8858E+02	-6.4897E+03	1.1388E+04
23	2314	-4.0661E+03	1.1115E+03	-7.6241E+03	1.4024E+04
24	2317	-1.0582E+02	3.5904E+03	-2.8844E+03	6.1840E+03
25	2318	2.1581E+03	4.7583E+03	3.1906E+03	6.8971E+03
26	2319	-3.1216E+03	1.9064E+03	3.2937E+03	7.2024E+03
27	2320	-1.8173E+03	1.4317E+03	-3.9310E+03	7.3697E+03
28	2321	-5.4182E+03	-2.1453E+04	6.9467E+03	2.2762E+04
29	2322	2.2841E+03	-2.2259E+04	-2.9234E+01	2.3485E+04
30	2329	-1.7515E+03	4.7749E+03	-8.9463E+03	1.6563E+04
31	2331	-7.3333E+03	-5.5377E+03	8.8715E+03	1.6732E+04
32	2332	-1.7415E+03	-9.7903E+03	8.4562E+03	1.7215E+04
33	2333	3.6097E+03	3.5194E+03	1.3057E+04	2.2894E+04
34	2336	-4.9992E+03	-6.8719E+02	1.1890E+04	2.1122E+04
35	2338	-3.3562E+03	-1.2970E+04	-3.6629E+03	1.3275E+04
36	2344	4.4816E+03	1.3103E+03	2.4042E+03	5.7680E+03
37	2345	4.0840E+03	6.5221E+03	5.8545E+03	1.1636E+04
38	2349	-2.1139E+03	-7.3574E+01	1.3058E+04	2.2712E+04
39	2352	1.4401E+03	1.5266E+03	1.1891E+04	2.0649E+04
40	2361	1.0517E+03	1.1477E+03	4.8093E+03	8.4026E+03

352

Figure 8 : Detailed Output via high speed printer for Membrane Elements, Option - REDOUT

```

ENTER DATA REDUCTION OPTION
      OPTIONS=NINMAX,ENVELOP,REDOUT,SUPOS,MS,ADVISE,REAN
? SUPOS
STRESS (S) OR DISPLACEMENT (D)
? S
ENTER ELEMENT TYPE (ANY OF THE NASTRAN ELEMENT NAMES)
? COINM
SET DEFINITIONS AS IN PREVIOUS RUN ("YES"(Y) OR "NO"(N))
? N
ENTER ELEMENT SET (A SET OF ELEMENT ID NOS OR "ALL" (A))
? 2320 THRU 2324, 2360 THRU 2417
ENTER SUBCASE SET (A SET OF SUBCASE NOS OR "ALL" (A))
? 1,2,3

ENTER DATA REDUCTION OPTION FOR THE SUPERPOSED RESULTS.
      (OPTIONS =NINMAX,REDOUT,MS)
? REDOUT
ENTER MULTIPLICATION FACTORS FOR LOADING CONDITIONS
SUBCASE 1
? 1.0
SUBCASE 2
? 2.0
SUBCASE 3
? -1.0
ENTER MIN/MAX VALUES FOR
STRESS NORMAL-X
? -15000.,15000.
STRESS NORMAL-Y
? -15000.,15000.
SHEAR STRESS
? -14000.,14000.
VON MISES STRESS
? -0.,17000.

```

Figure 9 : Sample Input for the Superposition of Stresses in Membrane Elements, Option - SUPOS

```

YOU HAVE REQUESTED THE FOLLOWING OPTIONS:
EXECUTION OPTION - SUPOS , RESTART FROM FILE TAPE6
TYPE OF RESULTS - SUPERPOSED STRESSES FOR ELEMENT COINM
THE FOLLOWING RESULTS WILL BE PRINTED:
NORMAL-X STRESSES WHICH ARE LESS THAN -1.5000E+04 OR LARGER THAN 1.5000E+04
NORMAL-Y STRESSES WHICH ARE LESS THAN -1.5000E+04 OR LARGER THAN 1.5000E+04
SHEAR STRESSES WHICH ARE LESS THAN -1.4000E+04 OR LARGER THAN 1.4000E+04
VON MISES STRESSES WHICH ARE LESS THAN 0. OR LARGER THAN 1.7000E+04
8 ELEMENTS EXCEEDED THE INDICATED VALUES
DO YOU WANT STRESSES TO BE PRINTED (YES (Y) OR NO (N))
? Y

```

ELEMENT ID.	STRESS NORMAL-X	STRESS NORMAL-Y	SHEAR STRESS	VON MISES STRESS
2321	1.7285E+03	3.1395E+04	5.6400E+02	3.0573E+04
2322	1.1803E+04	-3.6110E+03	-1.5339E+04	3.0013E+04
2400	-6.5340E+02	-9.2901E+02	-1.5327E+04	2.6560E+04
2401	3.0100E+01	-1.5168E+03	1.4177E+04	2.4603E+04
2402	-1.1406E+04	-1.2679E+03	7.8858E+03	1.7429E+04
2404	1.3185E+03	4.7247E+02	1.7971E+04	3.1148E+04
2408	-4.3059E+03	-6.7104E+03	-1.8739E+04	3.2987E+04
2409	-1.5020E+03	1.3390E+03	1.7646E+04	3.0663E+04

```

CONTINUE (C) OR QUIT (Q)
? C

```

Figure 10 : Sample Output for Option SUPOS

```

ENTER DATA REDUCTION OPTION
      OPTIONS=MINMAX, ENVELOP, REDOUT, SUPOS, MS, ADVISE, REAN
? MS
STRESS (S) OR DISPLACEMENT (D)
? S
ENTER ELEMENT TYPE (ANY OF THE NASTRAN ELEMENT NAMES)
? QADMEN
SET DEFINITIONS AS IN PREVIOUS RUN ("YES"(Y) OR "NO"(N))
? N
ENTER ELEMENT SET (A SET OF ELEMENT ID NOS OR "ALL" (A))
? 2307,2321,2322
ENTER SUBCASE SET (A SET OF SUBCASE NOS OR "ALL" (A))
? 2,5

ATTENTION: IN THE FOLLOWING ENTRIES A ZERO VALUE CAUSES
      RETRIEVAL OF THE CORRESPONDING CONSTANT FROM THE RESTART FILE
DEFINE THE FOLLOWING CONSTANTS:
SIDE A (A)
? 0
SIDE B (B)
? 0
YOUNG MODULE (E)
? 0
POISSON RATIO (NU)
? 0
PANEL THICKNESS (T)
? 0
RADIUS OF CURVATURE (R)
? 0.0

```

Figure 11 : Sample Input for Computations of Margins of Safety, Option - MS

```

YOU HAVE REQUESTED THE FOLLOWING OPTIONS:
EXECUTION OPTION - MS , RESTART FROM FILE TAPE6
TYPE OF RESULTS - STRESSES AND MARGINS OF SAFETY FOR
      ELEMENT QADMEN
THE FOLLOWING PHYSICAL PROPERTIES OF THE CHOSEN ELEMENTS ARE USED:

```

ELEMENT ID	A/B	T	E	NU	R
2307	.92	.05	1.03E+07	.33	0.00
2321	2.00	.05	1.03E+07	.33	0.00
2322	.61	.05	1.03E+07	.33	0.00

```

SUBCASE 2
      ELEMENT STRESSES AND THEIR MARGINS OF SAFETY

```

ELEMENT ID	STRESS NORMAL-X	STRESS NORMAL-Y	SHEAR STRESS	WON NISES STRESS	N	N	MS
2307	6.461E+03	-1.558E+04	2.644E+03	2.203E+04	1	1	5.0
2321	4.892E+03	-1.072E+04	7.278E+02	1.389E+04	1	1	-3
2322	7.406E+03	-1.455E+04	-6.687E+03	2.255E+04	1	1	1.0

```

SUBCASE 5
      ELEMENT STRESSES AND THEIR MARGINS OF SAFETY

```

ELEMENT ID	STRESS NORMAL-X	STRESS NORMAL-Y	SHEAR STRESS	WON NISES STRESS	N	N	MS
2307	6.167E+03	-1.788E+04	-5.897E+03	2.392E+04	1	1	4.0
2321	-5.419E+03	-2.145E+04	6.947E+03	2.276E+04	1	1	-7
2322	2.284E+03	-2.226E+04	-2.923E+01	2.349E+04	1	2	0.0

```

CONTINUE (C) OR QUIT (Q)
? C

```

Figure 12 : Sample Output for Option MS

```

ENTER DATA REDUCTION OPTION
OPTIONS=NINMAY,ENVELOP,PEOUT,SUPOS,NS,ADVISE,REAN
? REAN
ENTER ELEMENT TYPE (ANY OF THE NASTRAN ELEMENT NAMES)
? CODMEN
SET DEFINITIONS AS IN PREVIOUS RUN ("YES" (Y) OR "NO" (N))
? N
ENTER ELEMENT SET (A SET OF ELEMENT ID NOS OR "ALL" (A))
? 2403,2404,2471,2472
ENTER SUBCASE SET (A SET OF SUBCASE NOS OR "ALL" (A))
? A

ENTER NEXT ELEMENT TYPE (IF ONLY ONE ELEMENT TYPE IS USED ENTER BLANK)
?
DEFINE THICKNESS (T), YOUNG MODULUS (E) AND POISSON RATIO (NU)
FOR THE FOLLOWING ELEMENTS:
ELEMENT ID 2403
? 0.05,1.03E+7,0.33
ELEMENT ID 2404
? 0.05,1.03E+7,0.33
ELEMENT ID 2471
? 0.085,1.03E+7,0.33
ELEMENT ID 2472
? 0.085,1.03E+7,0.33
DEFINE NEW FORCES ACTING ON GRIDS OF INDICATED ELEMENTS
(ENTER NO OR N AFTER THE FIRST ELEMENT ID IF FORCES REMAIN UNCHANGED)
? N
DEFINE INITIALIZATION FACTOR ALPHA1 AND STIFFNESS MULTIPLICATION FACTOR ALPHA2
ALPHA1
? 1.0
ALPHA2
? -0.10
DEFINE NUMBER OF INCREMENTAL STIFFNESS STEPS
? 10
DO YOU WANT INCREMENTAL RESULTS TO BE COMPUTED
(YES (Y) OR NO (N))
? Y
ENTER PERMANENT FILE NAME FOR THE REANALYSIS SUBMIT JOB FILE
? REAN1

```

Figure 13 : Sample Input for Reanalysis, Option - REAN

```

REANALYSIS EXECUTION OPTIONS:
ELEMENT TYPE(S) CHANGED: CODMEN
THE FOLLOWING ELEMENTS ARE CHANGED:
ELEMENT      GRIDS      CHANGES
  ID   G1   G2   G3   G4   T       E       NU
-----
 2403 168  173  172  167   .050  1.03E+07  .330
 2404 169  174  173  168   .050  1.03E+07  .330
 2471 206  231  226  201   .085  1.03E+07  .330
 2472 201  226  221  197   .085  1.03E+07  .330
FORCES NOT CHANGED
ALPHA1= 1.0      ALPHA2= -0.1      NUMBER OF STEPS= 10

DO YOU WANT RESULTS TO BE DISPOSED TO PJE (P) OR TO CENTRAL CITE (C)
? C
REANALYSIS JOB IS SAVED ON FILE REAN1

JOB REAN1 SUBMITTED      STOP RINA EXECUTION
EXECUTION COMPLETED AT 77/07/21. 16.07.09.

```

Figure 14 : Sample Output for Option REAN

**Page
Intentionally
Left Blank**

SOFTWARE FOR TRANSFERRING NASTRAN
DATABLOCKS BETWEEN DISSIMILAR COMPUTERS

Richard Rosencranz
Johnson Space Center

James L. Rogers, Jr.
Langley Research Center

Reg S. Mitchell
Goddard Space Flight Center

SUMMARY

This paper reports the successful culmination of efforts to freely exchange all types of NASTRAN datablocks between the internalized forms in any of three makes of NASTRAN-class computers and that of any other such make. The medium used is magnetic tape formatted in BCD (Binary Coded Decimal) character and numeral fields. The method used involves a FORTRAN-coded post-processor and a pre-processor, mostly portable but with a small assembly-coded component peculiar to each make of machine. Snags in a method reported in an earlier NASTRAN colloquium (ref. 1) have been overcome and functional requirements upgraded as well.

INTRODUCTION

The NASTRAN computer program is capable of executing on the CDC 6000 and CYBER series computers, IBM 360 and 370 series computers, and the UNIVAC 1100 series computers. High interest but also severe difficulties exist in projected conversion to specialized fast arithmetic processors such as the CDC STAR 100 or ILLIAC IV (ref. 2, 3, 4, and 5). It is highly desirable to be able to exchange NASTRAN datablocks among these several makes of computers for two chief reasons:

- o Large structures are sometimes analyzed by several co-operating groups which have ready access to different makes of computers;
- o It becomes desirable even though not yet feasible to perform compute-bound tasks on the very fastest arithmetic processing computers, whether near or far away.

One exchange medium which exists for matrices is DMI (Direct Matrix Input) bulk data cards and the existing modules for generating them and reading them. These, however, do not carry as many guard digits as is desirable to minimize the tendency to lose engineering significance during prolonged sequences of computation. Another medium is a special NASTRAN module or pair of modules for transmitting decimal coded tapes with large fields expressing matrix elements only. Such a pair was developed recently by RI (Rockwell International/Space Division) for exchange between their IBM equipment and the UNIVAC systems at the JSC (Johnson Space Center), and other such channels may exist.

The medium we report herein also uses decimal-coded (actually BCD) tapes with large fields as the readiest form in which data blocks can be exchanged. All types of NASTRAN internalized data--alphanumeric, integer, real, and double precision--are correctly transferred.

PREVIOUS WORK

In the Third NASTRAN Colloquium, held at Langley Research Center in 1973, Rogers (ref. 1) reported on a similar piece of work. BCD tapes were the exchange medium, and FORTRAN code was expected to execute on CDC, IBM, and UNIVAC main frames in a post-processor and also a pre-processor. The NASTRAN OUTPUT2 tapes (or files) were subjected to the post-processor on one host, then the pre-processor was used on another host computer, to convert the BCD tapes into INPUTT2 tapes (or files).

Unfortunately, it was found that the Level 15.0 NASTRAN modules OUTPUT2 and INPUTT2 utilized forms of datablocks which were quite machine-dependent. For instance, in UNIVAC the representation of a matrix column in an OUTPUT2 file was in string form, while it was not in string form on the CDC. There was also a subtle dependence upon the GINO buffer size. The effect was to make portable code of the processors inadequate, and the method was accordingly scrapped. Two facts are worth mentioning: (1) that utilizing NASTRAN modules OUTPUT1 and INPUTT1 would be impractical for the same reason, and (2) that in Level 16.0 the representation of matrix columns in OUTPUT2 files was made substantially simpler, at the cost of explicitly representing all zero elements even in sparse matrices.

OVERVIEW OF THE PRESENT METHOD

The present method resembles that of 1973 in its schematic flow and in the nomenclature of its main processors and sub-

routines. The transmission of datablocks begins with the writing of a so-called USER file or tape by invoking the NASTRAN Level 16.0 module OUTPUT2, on any make of computer in the NASTRAN class. Next, on the same host computer, the new post-processor named RDUSER reads the user file and causes the output of a BCD tape. Card inputs are used to set file pointers and other parameters, and then to select by name any subset of the datablocks to be transcribed in a manner meeting the requirements, terminating on a blank card.

Upon transferring the BCD tape to a different host computer, the new pre-processor named WRTUSR transcribes in an inverse manner any selected subset of the datablocks from the BCD tape to a file or tape of USER form. This is next processed (internalized) using the NASTRAN module INPUTT2, completing the desired transfer. It will be understood that familiarity with the DMAP language of NASTRAN, with respect to both the INPUTT2 module and follow-on computations, is required. Most often, the follow-on computations are done using the DMAP approach when matrices only are transferred, as in the JSC usage mentioned in the Introduction. However, when tables also are transferred, there is a better possibility of using the NASTRAN Rigid Format approach or even the Automated Substructuring capability, instead of purely DMAP language.

Figure 1 illustrates what has been described as the flow path between two computers. Considering that three different makes are involved and that bi-directional paths are distinct as to software, there are six paths of functional interest. In addition, the RDUSER output on each make of computer can be used to test the WRTUSR pre-processor on the same make. Such tests involve three more flow paths which are of use only for checkout purposes.

Figure 2 shows exactly which of the nine possible flow paths have been successfully checked out at the time this is written.

The decision to interface new software with so-called user files rather than internal NASTRAN files simplified the task, but also imposed special requirements on the design of the two processors.

FUNCTIONAL REQUIREMENTS

A. For Datablocks which are Matrices:

With respect to matrices, the chief requirement is to exchange the largest number of guard decimals which any two of the three main NASTRAN machines can represent in their internal

data types of choice. Omitting the uncommon practice of choosing FORTRAN type DOUBLE PRECISION in CDC (which is not the default), we then have the following numbers of significant decimals to exchange:

From and to CDC (Single Precision): 15 decimals (approximate)
From and to IBM (Double Precision): 16 decimals (approximate)
From and to UNIVAC (Double Precision): 18 decimals
(approximate)

The statements are approximate. If a number has a 9 as its most significant digit, the number of trailing digits which are exactly representable is less than if a 1 or 2 is the most significant digit.

This boils down to a requirement to represent 16 decimals of significance, with sign and exponent, in tape fields. Any further decimals would either be meaningless in the originating machine or in the receiving machine.

Another requirement with respect to matrices is that each matrix column, in unpacked form, is allowed to occupy at most 8000 single precision words in both host machines. However, this limit is rather easy to relax by changing two lines of coding in the pre-processor or post-processor, or both.

A final requirement with respect to matrices is that sparse matrices should be represented without explicit zeros, in the BCD tape, whenever that will tend to reduce the amount of tape required.

The above requirements do not prevent transmitting double precision matrices from a CDC internal representation if that is the given condition. Neither does it prevent transmitting single precision matrices from the IBM or UNIVAC systems. But in the first case, much significance will be truncated before transmission, while in the second case, some of the digits transmitted will be only noise. The end results will reflect the lower of two precisions in whichever host it occurs.

B. For Datablocks which are Tables:

The functional requirements for tables are simply stated: to copy correctly single precision words which are either alphanumeric strings in A4 format, or are numeric of type integer or type real. It is known that numbers of type double precision are not, as a rule, contained in NASTRAN tables. In the 1973 work, there was no attempt to copy alphanumeric strings correctly.

DESIGN FEATURES OF THE TAPE

The tape, for transmittal between different makes, is written with 120 BCD characters per block, basically because that form is interchangeable while using the nearest to completely portable FORTRAN code. It is true that blocks of 10-100 times that many characters would require less tape and less input/output time while still using a reasonably small amount of fast storage as buffer or buffers. However, it was felt the machine dependencies so introduced were best deferred.

The 120-character size of each basic block corresponds to a whole number of words, no matter whether it is on CDC, IBM, or UNIVAC equipment. This fact simplifies the exchange of data. To avoid confusion the basic block of 120 characters will hereafter be called a line.

The first five characters in each line are invariably written as a single integer in I5 format. This fact is useful in distinguishing several characteristics of the rest of the line, and of the NASTRAN logical record of which it is a part. These characteristics include the following:

- o A zero value of the I5 integer field denotes either the end of the tape label or the end of a NASTRAN datablock.
- o Two successive zero values denote the end of the tape, just prior to a hardware end-of-file.
- o A negative value indicates the last line of either (1) a column of matrix, or (2) a logical record of a table. The absolute value counts the same entities.
- o In tables, a 5-digit decimal number (always positive and hence unsigned) is present in all lines except the last of a logical record and an initializer. This number contains encoded information concerning the data types of each successive field in the following line. Some such encoding is essential to transmit correctly the mixed information, as required by the enhanced functional specifications.

The remaining 115 character spaces of each line are apportioned into single precision, double precision, integer, or alphabetic fields in ways to suit each subclass of datablocks: dense matrix, sparse matrix, and table. Whether matrices are real or complex is also taken into account and made clear on transmission. Design details to do this will not be discussed further in this presentation.

VERIFICATION OF CORRECT TRANSMISSION

To verify that matrix or table datablocks are transmitted correctly to a large number of decimals, it is not sufficient to use the available NASTRAN display modules MATPRN and TABPRT, which print out the floating point elements in only 5 to 7-decimal precision, depending on the module and perhaps individual system settings.

The checking method employed is to use NASTRAN module SOLVE to obtain the high precision solutions of several sets of four simultaneous equations with given complex co-efficients. The result is another matrix having complex elements. In other words, the module was required to compute matrix X such that (with K and B given matrices) $K X = B$.

On the originating machine, and also on the receiving machine, the NASTRAN module MPYAD is used to obtain the matrix named ZERO, such that $ZERO = K X - B$.

It is naturally found then, using MATPRN, that ZERO is close to the null matrix. An example of ZERO computed after a UNIVAC-to-UNIVAC transfer is printed as Table 1. Since care is taken that the given matrices K and B are not unusually replete with commensurate whole numbers, truncation errors always occur causing most of the elements of ZERO to be non-zero numbers in the noise level of the least precise components of the hardware and software. It is to be expected that the noise level printed out on the receiving machine will slightly exceed that in the transmitting machine. Yet it has been found that the error growth in transmissions is nominal in all the different paths tested. It follows that most guard digits are reliable.

To verify tables, the eyeball method is sufficient. The main concern is not in the exact large number of guard digits, since all are limited by single precision on the two less precise machines, and follow-on sequence of computing do not have to get many trailing digits from the tables anyway. Instead, the concern is that no real number shall be erroneously transmitted as a character string, and vice versa. This possibility is discussed in the following section. If eyeball tests indicate trouble with certain tables, it may be possible to input a premise on a data card which will effect a workaround when complete success cannot otherwise be guaranteed.

MACHINE DEPENDENCY IN DISTINGUISHING INFORMATION TYPES

It was stated, under Functional Requirements, that integers,

reals, and alphabetic strings should all be transmitted correctly. It turns out that any real zero may, without error, be transmitted in the BCD tape as an integer zero. This constitutes an exception to the following rule:

- o Correct transmission of any table items requires correct recognition of their types within the transmitting machine.

It is easy to distinguish integers from both floating point and alphabetic words. This is done by observing that if the absolute value of a word (equivalenced to a FORTRAN type integer) is less than $2^{23} = 8,388,608$, the type is neither floating point non-zero nor alphabetic, since each of these types contain bits as high order or higher than the 23rd from the right. Supposing that a word is too high order to be type integer, the ways to distinguish real from alphabetic are highly machine-dependent. The ways are not entirely reliable. By using different branches, depending upon the identity of the host machine, one FORTRAN code supports all the following details:

CDC - The easiest distinguishing method applies to CDC machines. Since every NASTRAN alphabetic string is restricted to four (A4-format) characters, left justified, with blanks to the right, a test for all the right-hand blanks has either great or small distinguishing power, depending on their number. On CDC machines, six blanks--amounting to 36 bits--are found in every true word of an alphabetic string. The method is to shift off all four meaningful characters, then test the rightmost part. Among random real floating point numbers, only one in about 6.9×10^{10} would be expected to appear with such a bit pattern as to match six rightmost blanks. Thus, there is no appreciable trouble expected using this method on CDC machines.

UNIVAC - The next easiest distinguishing method occurs on UNIVAC equipment, where two blanks--represented by 12 bits--are at the right end of every true alphabetic string word. Random floating point numbers have a 1 in 4096 probability of matching that same pattern in their rightmost 12 bits. Since this is an undesirably high probability, the method used is first to class as floating point those words which do not pass this test, and then, to subject the survivors (eligible to be alphabetic strings) to tests of each of its four 6-bit subfields on the left. If any of these fail to conform with the bit pattern of any of a restricted set of 49 characters, the word is classified as a real number. Otherwise, the word qualifies as alphabetic. By imposing the extra tests, the probability that a random floating point number will be misclassified as alphabetic is reduced from 1 in 4096 to 1 in 11,921 (approximately).

The restricted character set includes the 26 letters,

10 digits, all punctuation marks allowed in the most basic of FORTRAN, and the dollar and semi-colon symbols.

IBM - The hardest task and least certain of distinguishing reals from alphabetic strings occurs when the transmitting machine is IBM. The reason is that all 32 bits of a word are needed to express the four characters allowed in such a string, leaving no portion of the word restricted to the pattern of blanks.

Each 8-bit byte of the word is tested for conformity to any of the characters in the restricted set defined above. The probability that a random real number passes this test, and hence is erroneously accepted as alphabetic, is about 1 in 745. (The probability is readily computed by observing that there are 256 different configurations of each 8-bit byte, and for all four, independently, to be in the restricted set of 49 acceptable conformations, has the probability of $(49/256)$ to the fourth power. This has the approximate value of $1/745$.)

Last Resort - Recognizing that the real numbers may be mistaken for character strings on either IBM or UNIVAC--with unacceptable frequency--a provision is made in the code to work around this using prior knowledge about any particular table datablock. The user may rule out the whole class of alphabetic string words in all but a given number of the earlier logical records of a datablock. The given number may in many cases be equal to 1, in that the first record, as a rule, contains the name of the table. A field on an input data card (whose main function is to select a datablock by name) can be used to set this switch in the post-processor program.

FUTURE WORK

One area for future work is to enable the same information to be shared using BCD tapes containing blocked records of substantially larger size. We expect that the code for doing this will not be portable across all three makes of machines.

It would be desirable and practical to convert the processors into NASTRAN module code of the usual mostly-portable type. The objective would be to eliminate the operation of modules OUTPUT2 and INPUTT2.

Projecting forward in time, it may become possible to use fourth-generation computers, i.e., those of the class of the CDC STAR and the ILLIAC IV, to execute certain compute-bound modules of NASTRAN before the implementation of all NASTRAN on the same machine(s). For this to be done without an unreasonable burden of input/output operations, it would be appropriate to accomplish both the above described enhancements.

REFERENCES

1. Rogers, James L., Jr. A Method of Transferring NASTRAN Data Between Dissimilar Computers. NASA TM X-2893, 1973, pp. 633-639.
2. Anonymous (Control Data Corporation): "Study of the Modifications Needed for Efficient Operation of NASTRAN on the Control Data Corporation STAR 100 Computer", NASA CR-132644, 1975.
3. Field, Eric I. (Universal Analytics, Inc.): "Feasibility Study for the Implementation of NASTRAN on the ILLIAC IV Parallel Processor", NASA CR-132702.
4. Golden, Michael E.; Hurwitz, Myles M.: "NASTRAN on the Texas Instruments Advanced Scientific Computer: Feasibility Study and Experimentation", Naval Ship Research and Development Center Report 76-0092, November 1976.
5. Rogers, James L., Jr.: "The Impact of "Fourth Generation" Computers on NASTRAN", NASA TM X-3278, 1975, pp. 431-447.

TABLE I.-DISPLAY OF THE ZERO - K X - B MATRIX

PLAYBACK OF A USER TAPE MADE ON UNIVAC FROM A UNIVAC BCD TAPE
 THE ZERO MATRIX IS CALC BY APPLYING MPYAD TO A TRANSMITTED X

NASTRAN 16.0 9/7/76

(WHICH SHOULD BE EXPECTED TO LOSE A LITTLE FROM IDENTICAL ZERO)

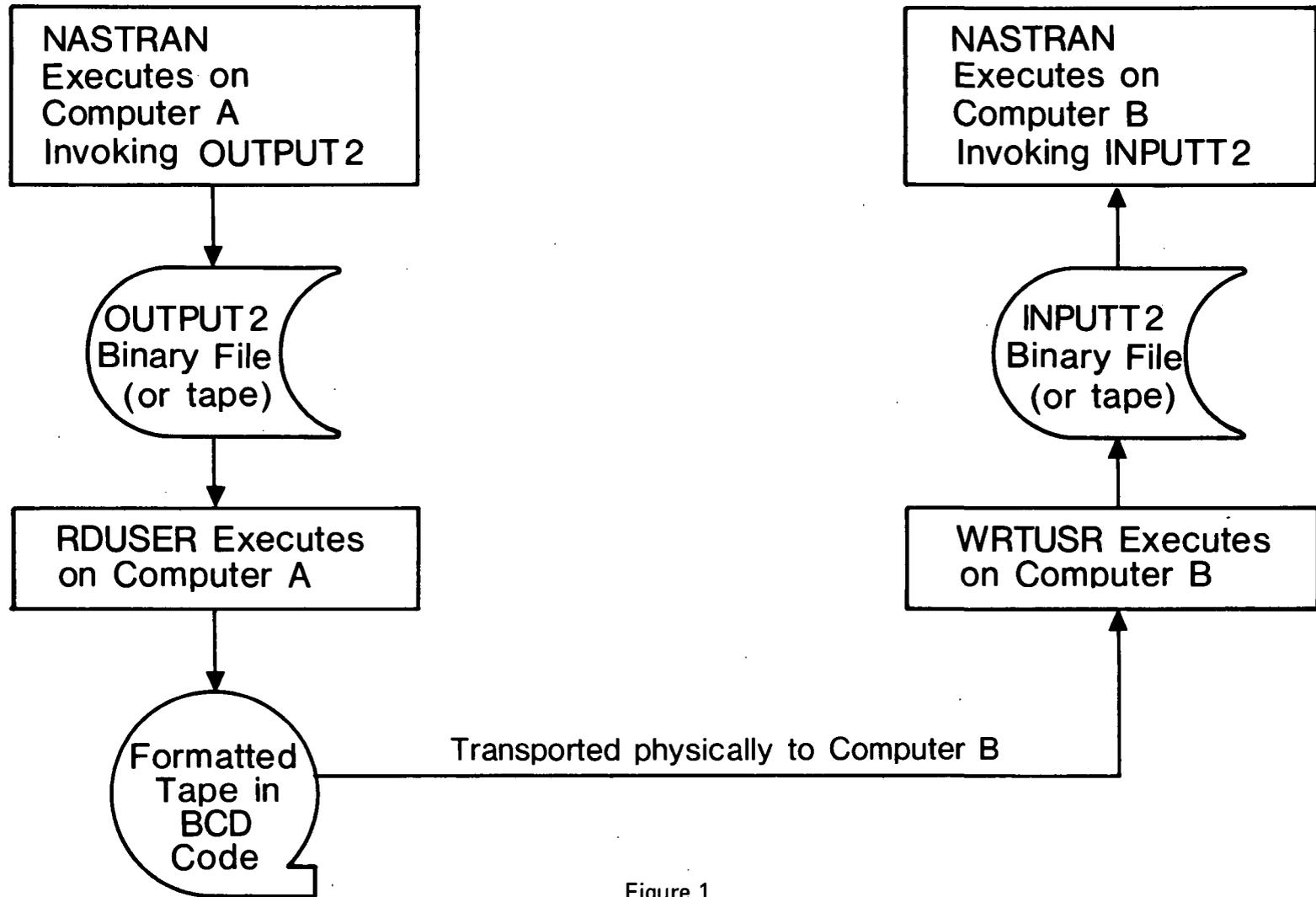
MATRIX ZERO (GINO NAME 102) IS A COMPLEX 4 COLUMN X 4 ROW SQUARE MATRIX.

COLUMN	1	ROWS	1 THRU	4	-----			
	-5.7680-17+	-7.6762-17I	-2.2985-17+	2.5153-17I	-6.0715-18+	1.1059-16I	-5.4319-17+	-4.7773-17I
COLUMN	2	ROWS	1 THRU	4	-----			
	-2.0817-17+	-1.8431-17I	5.6379-17+	-8.6302-17I	-5.0741-17+	-4.2718-17I	-2.0037-17+	4.0251-17I
COLUMN	3	ROWS	1 THRU	4	-----			
	-6.7654-17+	2.3852-17I	3.9031-17+	8.7387-17I	7.2858-17+	-5.4861-17I	-5.0947-17+	2.8081-17I
COLUMN	4	ROWS	1 THRU	4	-----			
	1.3791-16+	-5.1174-17I	-4.9873-17+	-1.1774-16I	-2.1250-17+	3.9899-17I	2.0871-18+	7.5054-17I

THE NUMBER OF NON-ZERO WORDS IN THE LONGEST RECORD = 16

THE DENSITY OF THIS MATRIX IS 100.00 PERCENT.

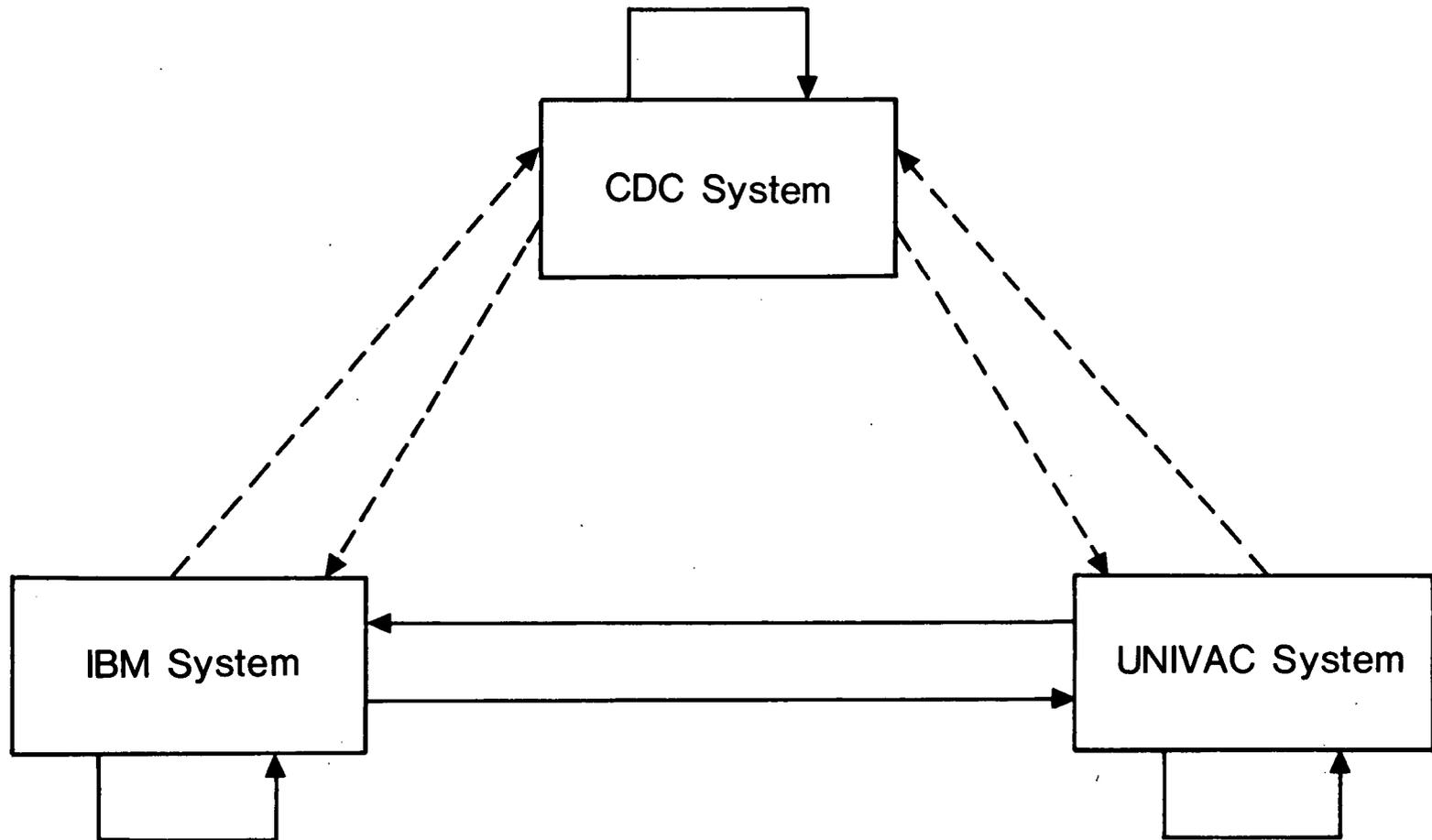
DATA PATH BETWEEN COMPUTERS



367

Figure 1

PROGRESS IN CHECKOUT OF DATA PATHS



Solid lines signify tape was correctly processed at its destination
Dotted lines mean tape formats still gave trouble, July 1977

Figure 2

FINITE ELEMENT ANALYSIS OF FIBER WRAPPED

SHELLS WITH NON-SYMMETRIC LOADS

G. Peter O'Hara
Department of the Army
Watervliet Arsenal
Watervliet, New York 12189

SUMMARY

The structural analysis of filament wound composite shells presents several rather large problems; the composite material properties are anisotropic, the properties vary along the length of the shell, the profile may be very complex, the thickness may vary along the length of the shell, more than one fiber type may be used, an isotropic liner may be used, large non-symmetric loads may be present, etc. Though all of these things tend to make the finite element method very attractive, the data preparation is rather large. The paper describes two FORTRAN IV programs to aid in the preparation of the important grid and material information. Also some examples of the application of these programs is presented.

INTRODUCTION

Over the last century so much theoretical work has been done on shell analysis that I shall not even try to cover it in any detail. I shall only say that many problems have been solved and indeed membrane shell problems are very easy. There are, however, many problems which are not well defined and are indeed mathematically difficult or impossible. These include the large class of problems where the contour of the surface cannot easily be described by a closed form and problems where the material conditions are complex such as anisotropic materials. Also, conditions arise when engineering answers must be obtained quickly to meet deadlines. It is in these cases where the finite element code like NASTRAN becomes a powerful tool.

This work will concentrate on axisymmetric shell problems using anisotropic materials and discontinuous loadings. It will also consider the question of relatively thick shells. There are three types of elements (ref. 1) in NASTRAN that can be used for axisymmetric shell problems. First is the conical shell element (CCONAX) which is an element with the shape of a portion of a cone. It is simple to grid because each element requires 2 grid points defined in the R-Z plane. It will accept nonsymmetric loads with the use of a Fourier method of analysis where each harmonic adds 2 degrees of freedom to each grid point. This element has four problems:

- (1) It will accept only isotropic materials.
- (2) It cannot be mixed with other element types.
- (3) The accuracy depends on the number of harmonics chosen.
- (4) Special input cards are required.

The second element type is the Toroidal ring (CTORDRG) element which is a curved shell element connecting 2 grid points, it will accept orthotropic material properties. However, this element has 2 problems:

- (1) It will accept only symmetric loads.
- (2) It cannot be mixed with other element types.

The last available method is to build up the shell from quadrilateral or triangular plate elements, which requires 3 dimensional grid. This method has the disadvantage of using many elements and grid points with 5 or 6 degrees of freedom per grid point. However, it can take anisotropic materials, non-symmetric loads and can be mixed with other element types, i.e., rods, bars, solids, etc. The element also has a partial capability to handle thick plates.

The object of this work is to study filament wound structures of various types using the plate elements (CQUAD1, CQUAD2, CTRIA1, CTRIA2) where the efficient generation of grid points and elements requires a special generation program with the following limitations:

- (1) The body is symmetric.
- (2) Material and section properties are symmetric.
- (3) One or more planes of loading symmetry exist.
- (4) The material orientation must be defined in a consistent manner.

The material properties and section properties must be assembled from the data for each ply of the finished product and this must be done at the centroid of each element row because the properties are assumed to be symmetric. The elements should be arranged in rows so that properties can be assembled for sets of elements. This process is carried out in the program PLAPROP. In this program the following properties can be varied for each ply matrix material, fiber material, fiber volume fraction, void content, ply thickness, and fiber angle.

The anisotropic material properties in each ply are defined as follows:

$$\begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ & C_{22} & C_{23} \\ \text{SYMMETRIC} & & C_{33} \end{bmatrix} \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \gamma_{12} \end{bmatrix} \quad (1)$$

The plate properties are defined as: (ref. 2)

$$\begin{bmatrix} N_x \\ N_y \\ N_{xy} \\ \hline M_x \\ M_y \\ M_{xy} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & | & B_{11} & B_{12} & B_{13} \\ & A_{22} & A_{23} & | & & B_{22} & B_{23} \\ \text{SYM} & & A_{33} & | & \text{SYM} & & B_{33} \\ \hline & & & | & D_{11} & D_{12} & D_{13} \\ & & [B]^T & | & & D_{22} & D_{23} \\ & & & | & \text{SYM} & & D_{33} \end{bmatrix} \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma \\ \hline K_x \\ K_y \\ K_{xy} \end{bmatrix} \quad (2)$$

Where

$$A_{ij} = \sum_{i=1}^n C_{ij} (T_i - T_{i-1})$$

$$B_{ij} = \sum_{i=1}^n C_{ij} (T_i^2 - T_{i-1}^2) / 2.0$$

$$D_{ij} = \sum_{i=1}^n C_{ij} (T_i^3 - T_{i-1}^3) / 3.0$$

Where T_n is the distance from the extreme fiber of a layer to the neutral plane (Figure 1).

In NASTRAN the bending and membrane forces are uncoupled, i.e. it is assumed that the B partition is zero. The only input matrix is the material ([C]) matrix (called [G] in the "Users Manual"), and the [A] and [D] are defined as follows: (ref. 3)

$$[A] = T_m [C] \quad (6)$$

$$[D] = I [C] \quad (7)$$

The NASTRAN system also allows the inclusion of the transverse shear deformation of the plate elements. However, this has only been implemented for isotropic materials. Relation may be stated as : (ref. 3)

$$\begin{bmatrix} V_x \\ V_y \end{bmatrix} = T_s \begin{bmatrix} G & 0 \\ 0 & G \end{bmatrix} \begin{bmatrix} \gamma_x \\ \gamma_y \end{bmatrix} \quad (8)$$

Where G is the shear modulus of an isotropic material.

In NASTRAN two options are available, either the plate thickness (T_m) and the material ([C]) is input and the rest is calculated from standard relations (CTRIA2, CQUAD2) or the problem can be split and T_m , I and T_s input along with three different material properties, one each, for membrane, bending and shear properties.

Element and generation both require tedious calculation and each relies on a different data base. For this reason two separate programs have been developed with only one simple interface. That interface is the printout of the coordinates of the centroid of the elements in each row, along with the appropriate section property card number. These two programs PLAPRO and SGEN will be discussed in this paper along with several test problems.

SYMBOLS

T_m	membrane thickness
T_s	transverse shear thickness
I	area moment of inertia ($T_m^3/12$)
G	shear modulus
[C]	material property matrix
[A]	inplane partition of plate stiffness matrix
[B]	coupling partition of plate stiffness matrix
[D]	bending partition of plate stiffness matrix

σ_y, σ_x	extensional stress
τ_{xy}	shear stress
ϵ_x, ϵ_y	extensions
N_x, N_y, N_{xy}	plate force
M_x, M_y, M_{xy}	plate moment
K_x, K_y, K_{xy}	curvature
T_i	length from the extreme fiber of a layer to the neutral axis
$\gamma_{xz}, \gamma_{xy}, \gamma_{yz}$	shear strain
V_y, V_x	transverse shear load
E_x, E_y	Young module
ν	Poission ratio
R	plate thickness ratio

SGEN

The primary objective of a grid generation scheme is to make both input data and the finished grid as easy to understand as possible. It is also desirable to eliminate any unusual sensitivity of the generation scheme to the input variables. With these and the previously cited limitations the concept of generating both elements and grid point in uniform rows from the minimum to the maximum angle coordinate is the most effective. This method will usually produce a good matrix bandwidth as well. With this in mind the data deck can be divided into 5 sections with one or more input parameters in each:

1. Definition of the R-Z contour with a set of 2 or more input points in R and Z. Only the first and last point must become grid point coordinates.
2. The angular range over which the grid is to be generated where the two extremes are assumed to be planes of load symmetry.
3. Element type, one of the following CTRIA1, CTRIA2, CQUA1 or CQUA2.
4. Number of grid points in the angular direction at the first R-Z point.
5. Approximate element size increase factor to control element size growth rate.

Generation proceeds in the following way. First a spline function approximation for the R-Z contour is generated from the input data (ref. 4). After this element sides are layed out along the contour so that the length along the contour is about the same as the length in the angular direction and the number of grid points in the angular direction is a constant. If the last point does not fall on the last data point a scale factor is changed and the process is repeated until a good fit is obtained. This produces a uniform increase in element size with an increase in radius. There are two exceptions, the first is when the ratio of an element size to the reference (the first) element exceeds the "size increase factor". At this point the element density is doubled, a transition row is generated, and a new reference size is set. This process will tend to control element size with an increase in radius without limiting element size. It is also possible to enforce the R-Z coordinate of a grid point row when this is a modeling requirement.

Once the R-Z coordinates of all element rows and the number of points in each row are established the appropriate grid and element cards can be written along with a coordinate system (CORD2R) and ENDDATA card. The grid point cards have the proper constraint placed on them to account for structural symmetry at the angular extremes and the 5 degrees of freedom at the 4 permissible element types. The element cards have a different section property card number for each row and a printout of the R-Z centroid position of the elements in each row.

Two grids generated by this scheme are shown in Figures 2 and 3. Both of these structures are of similar size and happen to have identical element numbers at similar locations.

PLAPROP

This program is essentially the FORTRAN IV extension of the theory previously stated in this report. The program is set up to compile the data given for 1 to 20 layers into plate information and this process can be carried out for many plates. The data is input for each layer independently. In order to cut down on the number of input cards only the first layer of the first plate in the deck needs to be fully defined. After that the only data which needs to be entered are those values which change. The code then examines the changes that have been made from the previous layer to find which parameters must be recalculated. At the end of each plate the summations are made and the following NASTRAN cards are output:

- (1) A section property card which refers to 3 material property card where a set identification number (PID) is input.
- (2) An anisotropic material card (MAT2) which contains the C matrix with a set identification number (MID=PIDX10).
- (3) An anisotropic material card (MAT2) which contains the D matrix with a set identification number (MID=PIDX10+3).

(4) An isotropic material card (MAT1) which has the average value for G with a set identification number (MID=PIDX10+6).

All these cards are compatible with NASTRAN and may be placed directly into a bulk data deck.

The data deck for PLAPROP must be assembled by hand which could be an extremely long operation for complex structures where the layer thickness and fiber angle vary rapidly. There is work now being done on the computation of the fiber angles, etc. for wrapped products. It does not seem reasonable, at this time, to attempt an interface in the three programs.

The calculation of the [C] matrix follows the work of Steyer (ref. 5) with some modification. The physical model of the work is shown in figure 4 in which the basic elemental cube is broken down into 4 subvolumes, 3 of which are matrix material and the other is the fiber. Using the assumptions that the (1) sides of the cube remain flat and parallel at all times and (2) shear on adjoining faces of the sub-volumes is zero, strain equilibrium and compatibility relations can be written for the cube. For a uniform extension of the cube on any of the 3 axis, 3 equilibrium, 3 compatibility and 12 strain equations can be written. If the extension is along the direction of the fiber, the solution can be reduced to a set of 6 equations and the axial modulus can be calculated. Extension in the transverse direction yields a system of 7 equations and the transverse modulus. From this data and the axial modulus, 2 poissions ratios can be calculated. The rest of the poissions ratios can then be calculated from the Maxwell reciprocal relation.

The basic assumption of flat parallel faces for a basic cube of an orthotropic material yields the following relation for shear modulus:

$$G_{ij} = \frac{E_i E_j}{E_i (1 + \mu_{ji}) + E_j (1 + \mu_{ij})}$$

It is then a simple matter to write the basic C matrix and transform it to the proper fiber angle for each layer.

SMALL TEST PROBLEM

These test cases were used to evaluate the necessity for shear stiffness and were either 1 or 2 element cantilever plate problems. When the problem is approached from the viewpoint of the ratio of plate thickness over element size (R) it can be seen that when the plate is thin $R = .10$ shear deformations may be less than 1% of the total deformation for a plate. However, if plate thickness is large ($R = 1.0$) shear may account for 40% of the total deformation in an isotropic case. In the case of a fiber wrapped composite the transverse shear properties are a function of angle and if the thickness was of the same order as the element size, serious errors could result. This situation is

common in Army structures. The use of a constant shear modulus is only a first order approximation to the correct answer.

APPLICATION TO A FLAT CONE

The analysis of a flat cone test specimen was used as a test case. This cone was 50 cm in diameter, 1.79 cm thick and had a 5 cm central hole. Load was applied through a steel plug epoxied in the central hole while the outer edge of the cone rested on the steel table of a 120,000 pound Universal Testing Machine. Deflection was measured using a machinist dial indicator.

The NASTRAN solution was carried out 38 grid points and 45 CTIRAL elements to model an 18° wedge of the cone. The material and section properties were calculated from this solution using published data for the fiber and matrix materials (E-glass, epoxy) and fiber angles of 0°, 60°, and 120° for a larger number of layers. This type of a layup produces nearly isotropic properties which was reflected in the constitutive matrix. A fiber volume fraction of 50% was used and the void content was assumed to be zero because no valid data was available.

The analysis was carried out both with and without transverse shear deformations. The results are shown in Table I in terms of the compliance of the structure.

The most interesting result is that in the solution with shear average round off error Epsilon sub E was smaller at 1.18189×10^{-12} compared with 6.683×10^{-12} for the solution without shear. This improved matrix behavior is also reflected in the length of the Grid Point Singularity Table, 6 possible singular grid points with shear and 11 without shear.

APPLICATION TO A ROADWHEEL

The two computer programs described in this report have been used in the analysis of a roadwheel from a tracked vehicle which is about 0.5 meters in diameter. Two grids used in the roadwheel shown in figures 2 and 3. Figure 2 is the grid for a standard aluminum wheel and figure 3 is the grid used for the analysis of a prototype composite wheel. An analysis was performed on both of these wheels and both were available for testing. Unfortunately, fabrication problems eventually produced a composite wheel which was sufficiently different from the fiber wrap angles used for analysis to prevent any valid comparison between analysis and test. The aluminum wheel has a problem in that the plate thickness is not well defined on the drawings. This resulted in analysis for the minimum and maximum thickness. For this paper compliance in a diameter compression loading will be used for comparison. The two finite element runs give a compliance of 6.382×10^{-5} and 9.469×10^{-5} meters per newton while the test produces 8.525×10^{-5} .

DISCUSSION AND CONCLUSION

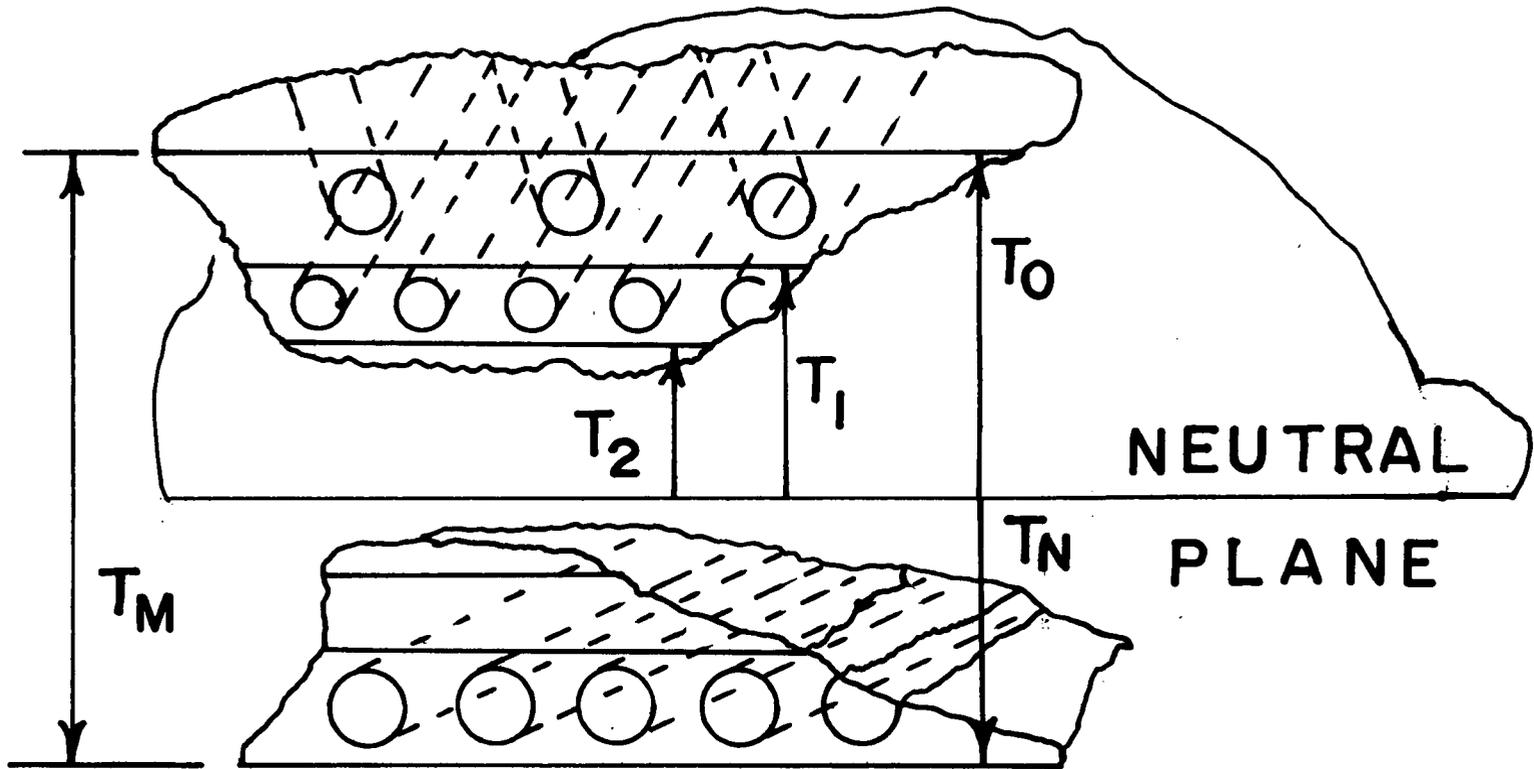
In the analysis of multilayer composite plate structures, the material definition becomes extremely important and as in the case of wrapped axis-symmetric shell problems, the material-section property problem can be more complex than the structural shape itself. However, it should be apparent that good structural analysis is dependent on good definition of both the structure and the material. Positive steps have been taken in each of these areas to provide computer programming to take the burden of data preparation off the designer and allow him to concentrate on what the structure is.

REFERENCES

1. The NASTRAN User's Manual, C. W. McCormick, Editor, NASA SP 222(01).
2. Primer on Composite Materials, J. E. Ashton, J. C. Halpin, P. H. Petit, Technomic Publishing Co., 1969.
3. The NASTRAN Manual, R. H. MacNeil, Editor, NASA SP 221.
4. Lorensen, W., An Interactive Graphics Finite Element System, Structural Mechanics Computer Programs, W. Pilkey, Editor, University Press, Virginia, 1974, pp. 991-1001.
5. Steyer, Carl, C., Analysis of a Thick-Walled Anisotropic Elastic Cylinder Subjected to Radial Pressure, Journal of Spacecraft, Vol. 3, No. 4, April 1966.

TABLE I

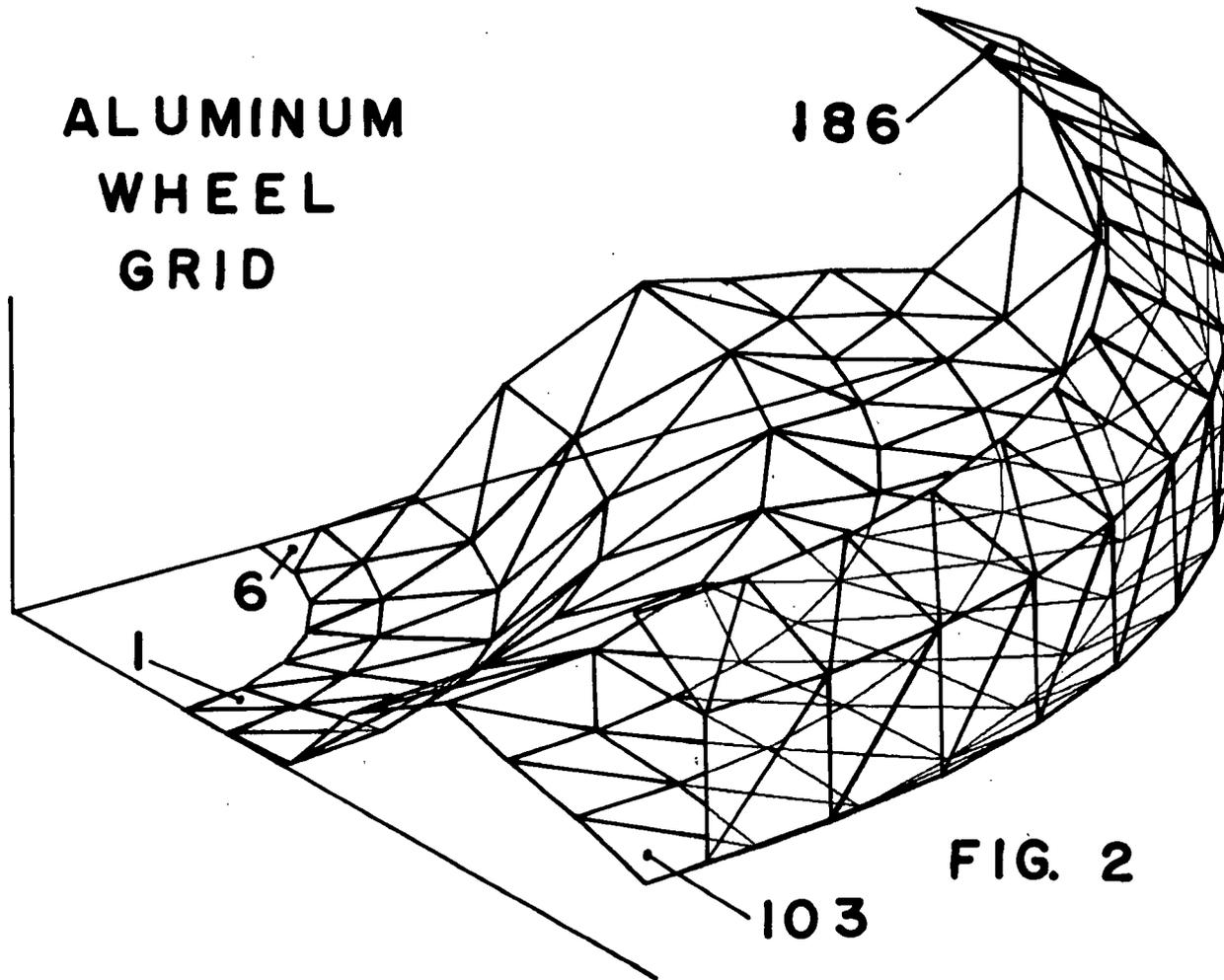
Test = 1.244×10^{-6} Meter/Newton
NASTRAN with shear = 1.126×10^{-6}
NASTRAN without shear = 1.113×10^{-6}



THICKNESS DEFINITION

FIG. 1

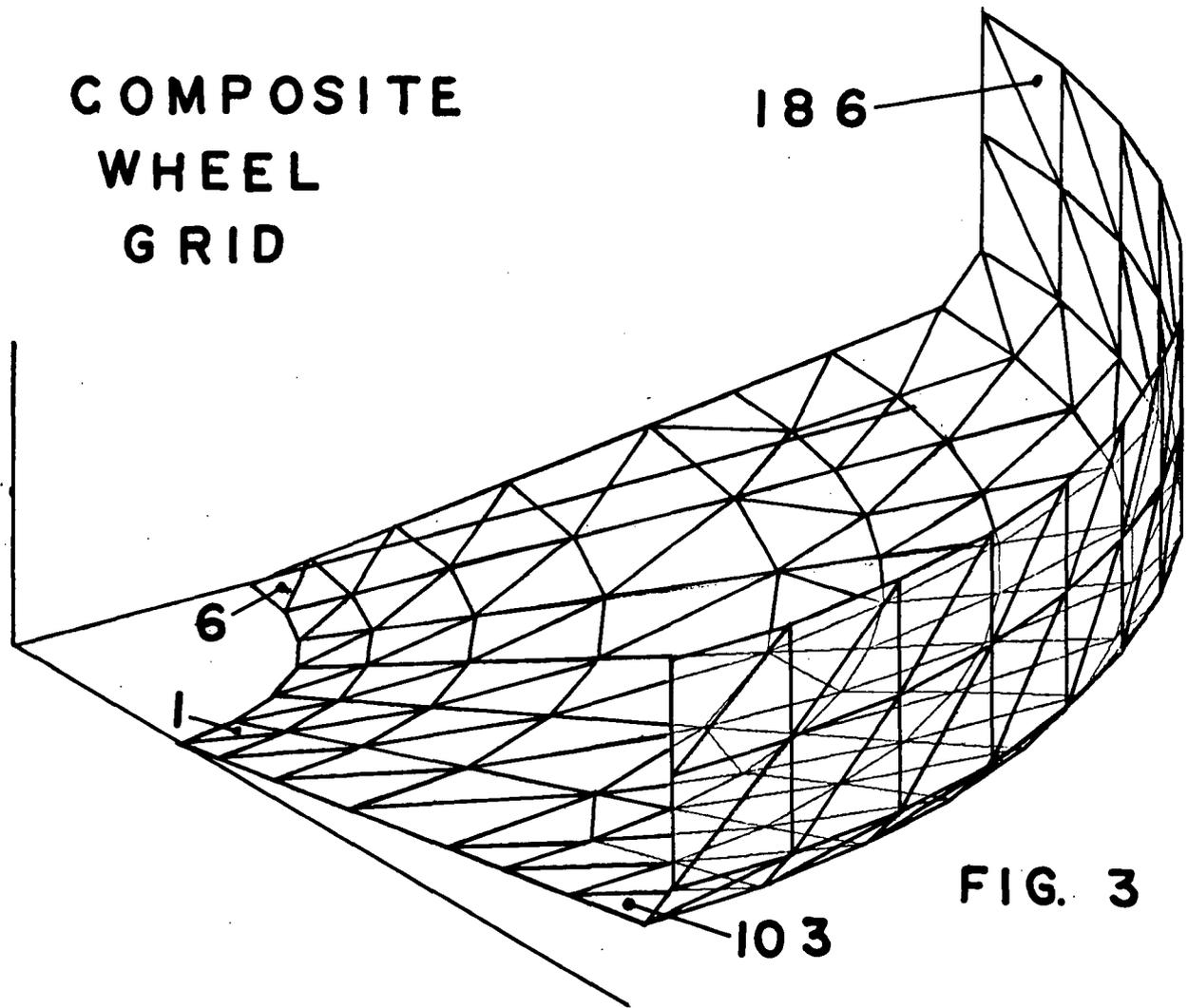
**ALUMINUM
WHEEL
GRID**



379

FIG. 2

**COMPOSITE
WHEEL
GRID**



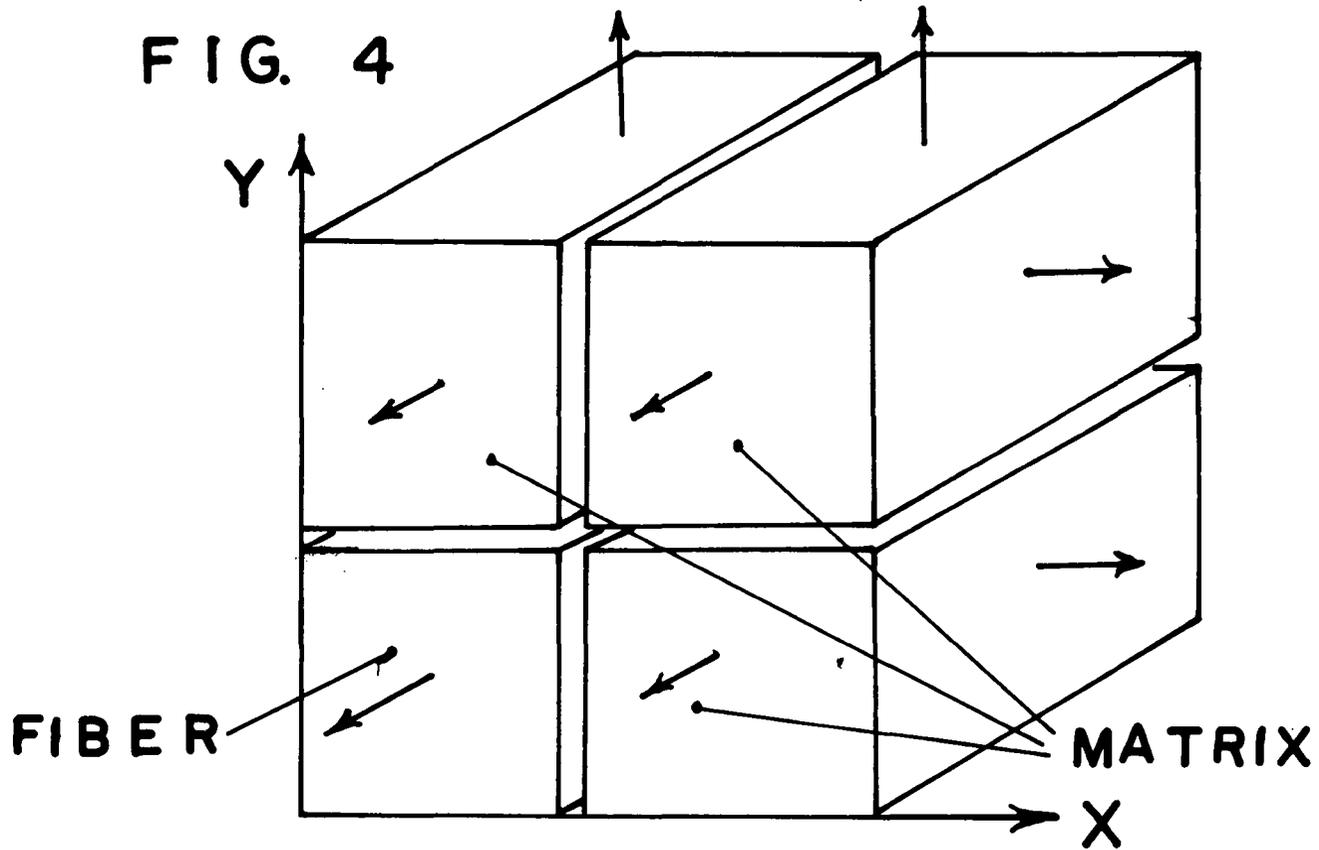
186

6

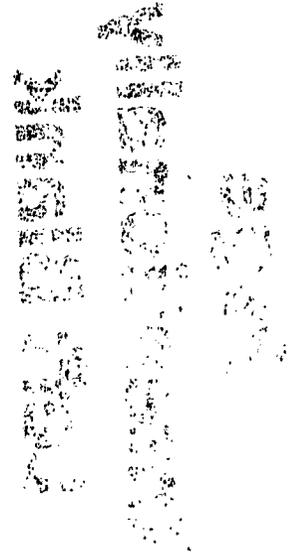
1

103

FIG. 3



ELEMENTAL CUBE



**Page
Intentionally
Left Blank**

NASTRAN FINITE ELEMENT IDEALIZATION STUDY

W. R. Case and J. B. Mason
NASA Goddard Space Flight Center

SUMMARY

The paper presents the results of a study to provide quantitative data to the analyst which can be used as a guide for the idealization of complex structures exhibiting plane stress, plane strain, or axisymmetric behavior. The NASTRAN membrane elements TRMEM, QDMEM, QDMEM1, and QDMEM2 were utilized in the study. In addition to providing convergence and accuracy type information for particular elements and mesh patterns, the data can be employed on a comparative basis to assess the amount of approximation relative to mesh refinement and element aspect ratio.

Beam type structures of constant rectangular cross section were selected for study. The basic beam configuration employed was a deep cantilever of unit depth and beam aspect ratio (length/depth) of two. The element aspect ratio portion of the study utilized unit depth beams with aspect ratios ranging between 0.2 and 20.0. The plane stress models were subjected to the following applied loading conditions: (1) end moment, (2) end shear. The applied loads were sized to give a maximum direct stress of unity, and boundary conditions were imposed which were sufficient to remove only the rigid body motions.

The investigation of the effects of variations of mesh refinement and mesh pattern were conducted using a basic rectangular mesh pattern. When employing the constant strain TRMEM element, the basic rectangular pattern was subdivided into triangles. This subdivision employs two different triangular patterns and allows results to be obtained which demonstrate the effect of modelling bias. Errors in tip deflection, direct stress, and shearing stress as a function of mesh size and element aspect ratio were obtained as well as mid-span stress distributions. All problems were solved on an IBM 360/95 computer using MacNeal-Schwendler Version MSC-38 Rigid Format-1 of the NASTRAN computer program. While NASTRAN uses double precision arithmetic for the solution of the global equations for displacements, subsequent computations to obtain element stresses are carried out in single precision. This suggests that some improvement in stress recovery might be expected when using the higher precision CDC machines.

FINITE ELEMENT MODEL PROPERTIES AND LOADING

Figure 1 shows the geometry, coordinate system, boundary conditions, and basic beam physical properties used in the study. The beam was divided into a

rectangular mesh in the x-y plane using the indicated elements. Figure 2 shows the mesh subdivision technique used.

For the TRMEM element studies, two mesh patterns were investigated. These are indicated as mesh Pattern 1 and 2 on Figure 2. Mesh Pattern 1 contains a modelling bias that should generally be avoided in favor of Pattern 2. For all problems, the mesh size is indicated by NL x ND where NL and ND are the number of rectangular subdivisions of the beam along the length and through the depth respectively. For the sample mesh patterns shown in Figure 2, NL = 6 and ND = 4.

The finite element model was loaded with work equivalent grid point forces to simulate the boundary stresses for an end moment and an end shear. This was done to simulate the applied loads as well as the reactions at the cantilevered end. Thus, displacement boundary conditions had to be supplied only for the purpose of restraining rigid body motion. The loading conditions and theoretical solutions are discussed below.

I. End Moment Loading

Forces are applied to the ends of the beam to simulate the linear direct stress indicated on Figure 3:

$$\sigma_x = 2\sigma_o y/d, \quad \sigma_o = 689.5 \text{ N/m}^2 \text{ (1.0 psi)}$$

The theoretical stresses along the beam are independent of x and equal to the end values above. The theoretical lateral displacement on the neutral axis at the free end ($x=l, y=0$) is

$$v(l,0) = -\frac{\sigma_o l^2}{Ed}$$

II. End Shear Loading

In this case, the theoretical stress distribution is

$$\sigma_x = 2\sigma_o y/d (1 - x/l)$$
$$\tau_{xy} = -\sigma_o d/4l (1 - 4y^2/d^2)$$

The beam is loaded with the stress values above at the two ends ($x = 0$, $x = \ell$). The theoretical lateral displacement at $x = \ell$, $y = 0$ is

$$V(\ell, 0) = -2/3 (\sigma_0 \ell^2 / Ed) \left(1 + \frac{4 + 5\nu}{8} d^2 / \ell^2 \right),$$

$$\sigma_0 = 689.5 \text{ N/m}^2 \text{ (1.0 psi)}$$

RESULTS OF ANALYSES

In order to assess the relative performance of the various elements investigated, the results of the study are presented, in the main, as convergence diagrams. These show the error in displacements and stresses at specific points of the beam as a function of mesh refinement. As the mesh becomes finer, one would expect convergence to zero error for all of the elements tested. At smaller mesh sizes, however, some elements will yield more accurate answers due to their higher order displacement polynomials.

There are two major questions to be answered; how fine a mesh is required to obtain accuracy within a given percentage, and to what degree is the accuracy a function of element aspect ratio. Thus, for the models investigated, results are presented which show error as a function of mesh size and of element aspect ratio. Also presented are plots showing the stress distributions through the beam depth and how it varies with element type, element aspect ratio, and mesh size.

Constant element aspect ratio results ($AR_e=1.0$) are shown in Figures 4 through 7. These figures display errors in deflection, direct stress, and shear stress as a function of mesh refinement. As expected, for the same mesh size, the bilinear quadrilateral element QDMEM1 gave the best results. Note that the TRMEM mesh Pattern 2 gives superior results in comparison to Pattern 1. This is due to the fact that there is less element orientation bias for Pattern 2.

Figure 6 shows midspan direct stress error at the outer fiber for the end moment and end shear loadings. Results are independent of loading and so nearly identical that they could be plotted on the same graph. In order to obtain the stresses at the outer fiber of the beam midspan, the element stresses had to be averaged. Since there are an even number of elements along the length, stresses from elements on either side of the beam midspan were averaged to obtain stresses at the midspan. Following this averaging, the direct stresses were linearly extrapolated using the outermost element stress value to obtain outer fiber stresses.

Figure 7 shows shear stress error at the neutral axis ($y=0$) of the beam. Again, shear values were obtained by averaging the values from the two elements on either side of the neutral axis. Note that the shear stress values are significantly less than those for the tip deflection and direct stresses.

The results described thus far show how the errors in the deflections and stresses vary as the mesh is refined while holding the element aspect ratio constant. In order to determine the effect of element aspect ratio on the errors, several runs were made where NL and ND were held fixed at 24 and 12 respectively, while the length of the beam was varied from 0.2 through 20. This means that the element aspect ratio,

$$AR_e = \ell_e / d_e = (\ell/d) (ND/NL)$$

varies. Figure 8 shows the effect of this variation on the tip deflection errors. As the element aspect ratio increases from unity (square elements) the errors get large for all but the QDMEM1. As element aspect ratio decreases the effects are far less pronounced. The trend is clear, however, and indicates that element aspect ratios much above 2.5, regardless of orientation, should not be used for the constant strain elements in the NASTRAN library. This aspect ratio sensitivity has been evidenced in other elements, notably the bending plate elements, which also exhibit sensitivity to changes from a rectangular shape to a swept or rhombic shape.

Figures 9 through 11 show stress distributions through the beam depth. The QDMEM, QMEM1, and QDMEM2 are compared and it can be seen that the direct stresses are, as expected, best for the QDMEM1 element. Comparing Figures 9 and 10 with Figure 11 demonstrates how the stress distributions vary with element aspect ratio. The superior behavior of the QDMEM1 is evident.

As was mentioned previously, the direct stresses had to be averaged in order to obtain values at the midspan. These averaged values were then extrapolated to give outer fiber stresses to generate the curves. The averaging process to obtain stress values at the midspan was found to have a smoothing effect on the TRMEM stresses that depended on the averaging technique used. For the quadrilateral elements, it can be seen from Figure 2 that in order to obtain stresses at the midspan, a simple average of the element stresses on either side of the midspan line is the obvious choice. For the TRMEM mesh Pattern 2, however, there are several possibilities for averaging as indicated below, with reference to Figure 2.

- (1) Average adjacent triangles such as j and k
- (2) Average adjacent triangles such as i and l
- (3) Average triangles i and j to obtain the stress in the rectangular area to the left of the midspan. Repeat for the right side and then average the result. This is the same as averaging (1) and (2).

Figure 12 shows the results of the three averaging techniques for an NL = 4, ND = 4 mesh deep beam subjected to an end moment. Either methods (1) or (2) give erratic results, whereas method (3), which is the average of (1) and (2), gives a very smooth curve for the direct stress. The erratic behavior of methods (1) or (2) are probably due to a bias introduced by the use of triangles that get better as more elements in an area are used to obtain the

average, as in method (3). Method (3) is the one used in all of the stress error convergence plots previously discussed.

All of the comparisons have indicated that the QDMEM element is the least accurate followed by the TRMEM, QDMEM2 and the QDMEM1, which is the most accurate. These comparisons have been presented on the basis of equal mesh size and the results are as expected; the higher order elements yield higher accuracy for a given problem size. The rather poor behavior of the QDMEM is explained by the TRMEM bias which is automatically introduced when this element is employed, see further discussion below. What is just as important, however, is the comparison of accuracy of these elements on a basis of solution cost. Since the studies reported in this chapter were all performed on the same computer using the same finite element program, a cost comparison can be made. For the deep beam with an end moment loading, the errors as a function of total solution time are shown on Figure 13 for models with NL = ND ranging from 4x4 to 24x24. This figure is from Reference 4 and based on NASTRAN Level 15.5 results. The solution time is the central processor unit (CPU) time required to process the complete job. As indicated on Figure 13, the TRMEM element is more costly than the others followed by the QDMEM and QDMEM1. However, the higher order elements are not grossly better as was indicated in the comparisons based on problem size. This is due to the fact that the higher order element stiffness matrices take longer to generate. On the IBM 360/95 using NASTRAN Level 15, the stiffness matrix generation times are

<u>Element Type</u>	<u>Stiffness Matrix Generation Time (CPU Sec)</u>
2-TRMEM elements	.04
1-QDMEM element	.18
1-QDMEM1 element	.32

The CPU times are those required to generate the stiffness matrix for an equivalent quadrilateral area for all three element types.

GENERAL MODELLING CONSIDERATIONS

The fundamental modelling technique governing the application of the elements is the objective of minimizing errors which arise due to the failure of the elements to satisfy equilibrium everywhere on the structure. Based on the known element characteristics and the results of the sample problems, several general observations can be made:

1. Since constant stress fields are inherent in the TRMEM, QDMEM, and QDMEM2 elements; the grid point refinement must be increased in regions of high stress (strain) gradient. While this is obvious since the overall stress state is being approximated in a discontinuous manner, it is important to note that for uniform (i.e. constant) stress fields, the elements should give exact results regardless of model refinements. As the stress condition

departs from the uniform condition, more refinement will be required.

2. Free edge boundary conditions should be treated with special care. As in 1 above, a free edge will often suggest the presence of a steep stress gradient as rapid stress decay takes place. Again, mesh refinement near the edge is indicated.

3. Elongated elements must be avoided. As the results indicate, numerical accuracy is severely reduced for the constant strain elements as the aspect ratio deviated from unity. This degradation is evident for the TRMEM, QDMEM and QDMEM2 elements and is explained by the fact that the TRMEM stiffness does not reduce to that of an axial element as elongation occurs. This suggests that the analyst should attempt to keep his TRMEM elements close to equilateral and the QDMEM and QDMEM2 elements near square.

4. Since a linear strain state is not admitted by the TRMEM, artificial residual stresses will be predicted by the unconstrained elements when loaded by linear temperature gradients. This effect is generally small, however.

5. Work equivalent representation of external loadings should always be utilized.

6. Numerous pattern arrays are normally available to the analyst for treating any particular problem. A general goal, however, should be to minimize mesh bias. As demonstrated by the Pattern 1 and 2 sample problem results for the TRMEM elements, different solutions are obtained for the same number of d.o.f. and elements. The pattern with the minimum bias, i.e. Pattern 2, gives superior results.

7. Stress averaging over two or more elements should always be used when employing the TRMEM elements. The most simple of averaging techniques can be utilized to improve stress definition for the area of interest. It is important, however, to be consistent in the methodology as to both the calculation of stress magnitude and location.

8. If properly employed, satisfactory results can be obtained with any of the elements tested. For equal mesh sizes, the QDMEM1 element provides generally superior results, followed by the QDMEM2, TRMEM2, and QDMEM elements, respectively. When compared on a solution time versus accuracy basis, however, the higher order elements do not exhibit such a dramatic improvement. As indicated earlier, the poor performance of the QDMEM can be attributed to bias of the constant strain TRMEM element mesh which NASTRAN produces when the QDMEM is employed. Actually the QDMEM is the average of two TRMEM patterns, each of which has a modelling bias (like the upper or lower half of pattern 1 in Figure 2). Thus, it should be expected that results using the QDMEM would be worse than those using the TRMEM if the TRMEM model is patterned judiciously (as in pattern 2). In fact, the QDMEM has errors about the same as the biased TRMEM pattern 1.

9. The MSC-38 QDME1 yields generally superior results when compared to the NASTRAN Level 15.5 and 16.0 QDME1. The reader should consult Reference 4 for a summary of NASA NASTRAN QDME1 performance.

REFERENCES

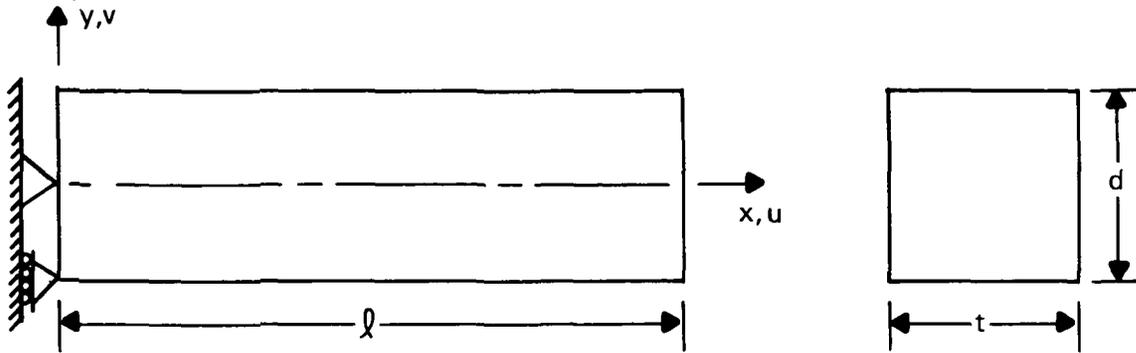
1. Love, A.E.H. A Treatise on the Mathematical Theory of Elasticity, Fourth Edition, Dover Publications, New York, 1944.
2. Bruhn, E. F. Analysis and Design of Flight Vehicle Structures, Tri-State Offset Co., 1965.
3. Gallagher, R. H. Finite Element Analysis Fundamentals, Prentice-Hall, Inc., 1975.
4. Case, W. R. and Mason, J. B. The Constant Strain Triangle and Associated Quadrelateral, Guidelines For Finite Element Idealization, ASCE Nat. Structural Engineering Convention, New Orleans, April 1975; Preprint 2504.

APPENDIX—NOTATION

AR	=	BEAM ASPECT RATIO (l/d)
AR _e	=	ELEMENT ASPECT RATIO (l_e/d_e)
d	=	BEAM DEPTH
d _e	=	ELEMENT DEPTH (d/ND)
E	=	YOUNGS MODULUS
l	=	BEAM LENGTH
l_e	=	ELEMENT LENGTH (l/NL)
ND	=	NUMBER OF ELEMENTS THROUGH BEAM DEPTH
NL	=	NUMBER OF ELEMENTS ALONG BEAM LENGTH
t	=	BEAM THICKNESS
u,v	=	BEAM DEFLECTIONS
ν	=	POISSON RATIO
σ_x, τ_{xy}	=	BEAM STRESSES

FIG. 1

BEAM GEOMETRY AND PROPERTIES



$$l = \begin{cases} .0508 \text{ M (2.0 IN) BASIC DEEP BEAM FOR MESH STUDY} \\ \text{VARIABLE FOR ASPECT RATION STUDY} \end{cases}$$

$$d = .0254 \text{ M (1.0 IN)}$$

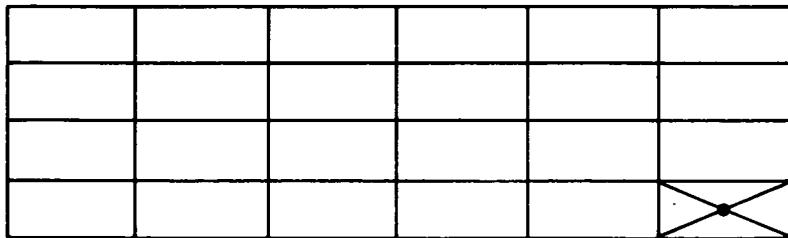
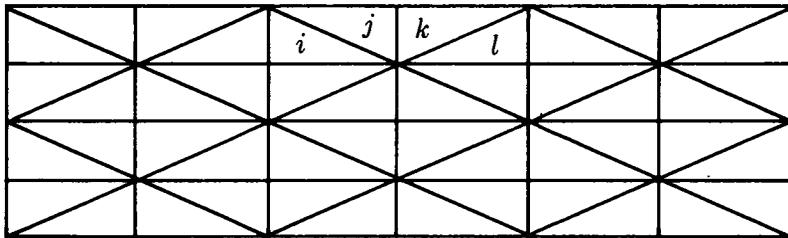
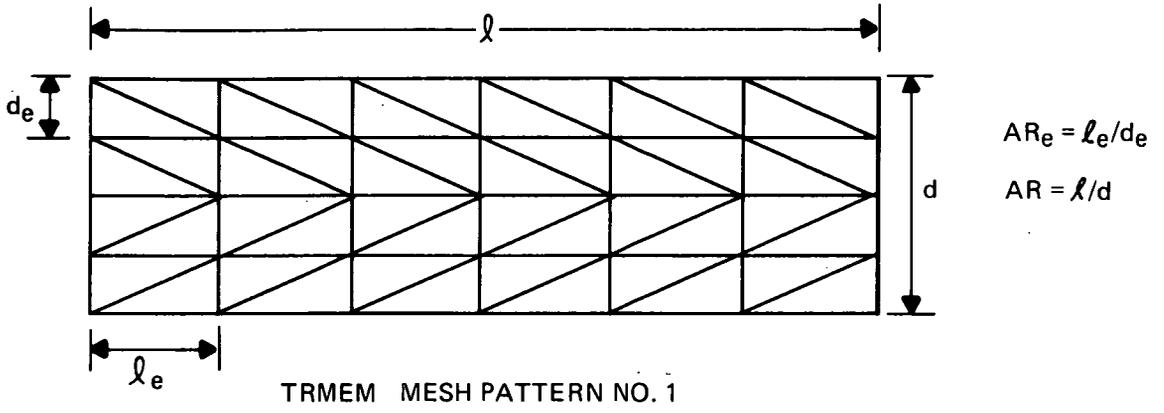
$$t = .0254 \text{ M (1.0 IN)}$$

$$E = 1.9305 \times 10^{10} \text{ N/m}^2 \text{ (} 28 \times 10^6 \text{ LB/IN}^2\text{)}$$

$$\nu = 0.3$$

$$u = v = 0 \text{ AT } x = y = 0 \text{ AND } u = 0 \text{ AT } x = 0, y = -d/2$$

FIG. 2
FINITE ELEMENT MODEL MESH PATTERNS



} EXTRA NODE—QDMEM2
} ELEMENT ONLY

FIG. 3
BEAM LOADS

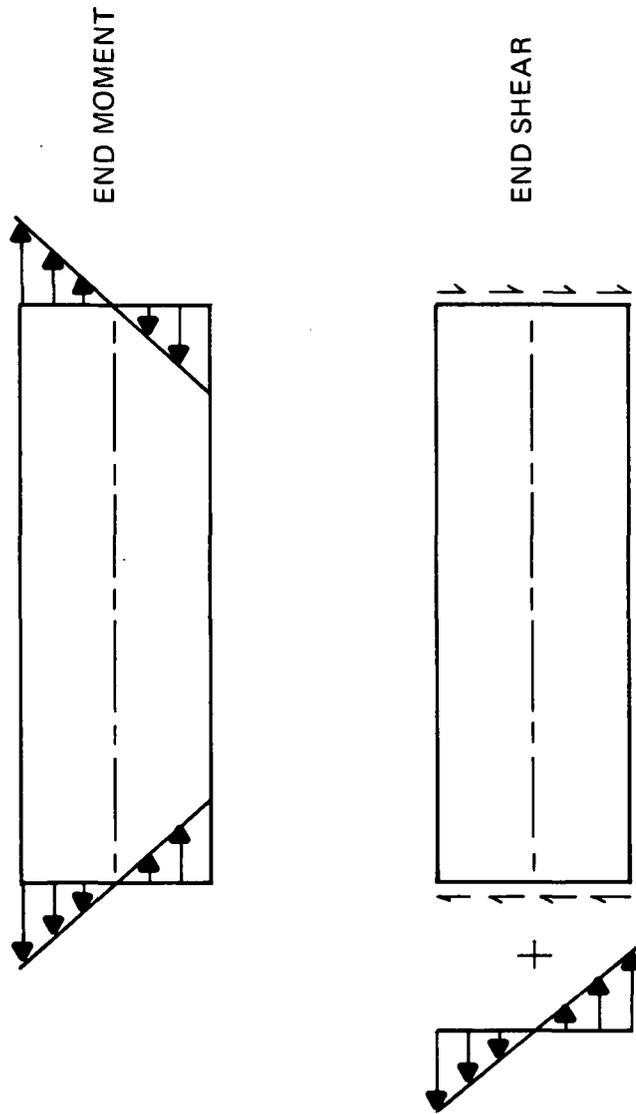


FIG. 4
TIP DEFLECTION ERROR
DEEP BEAM-END MOMENT LOADING
(MESH SIZE STUDY)

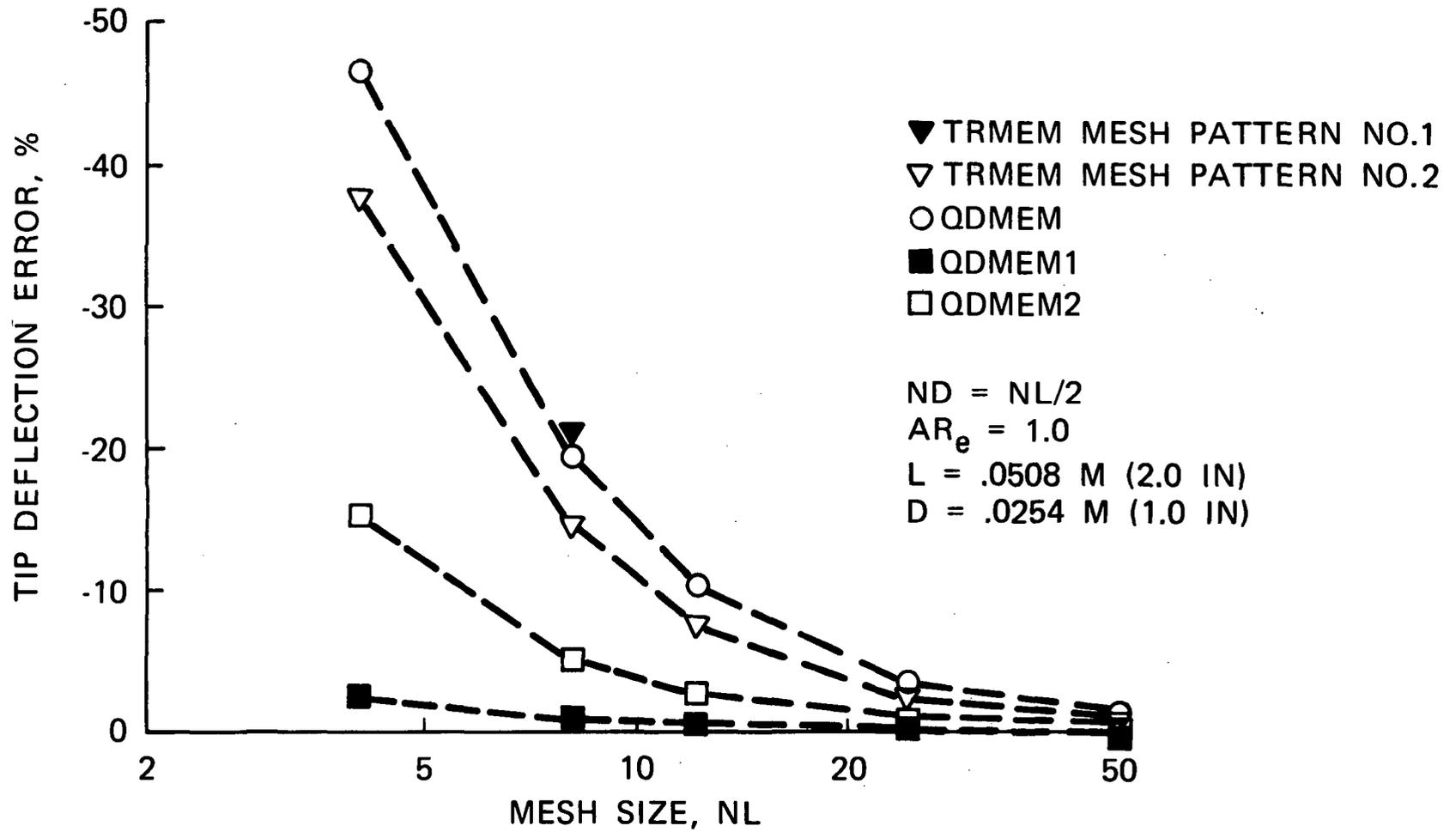


FIG. 5

TIP DEFLECTION ERROR
 DEEP BEAM-END SHEAR LOADING
 (MESH SIZE STUDY)

395

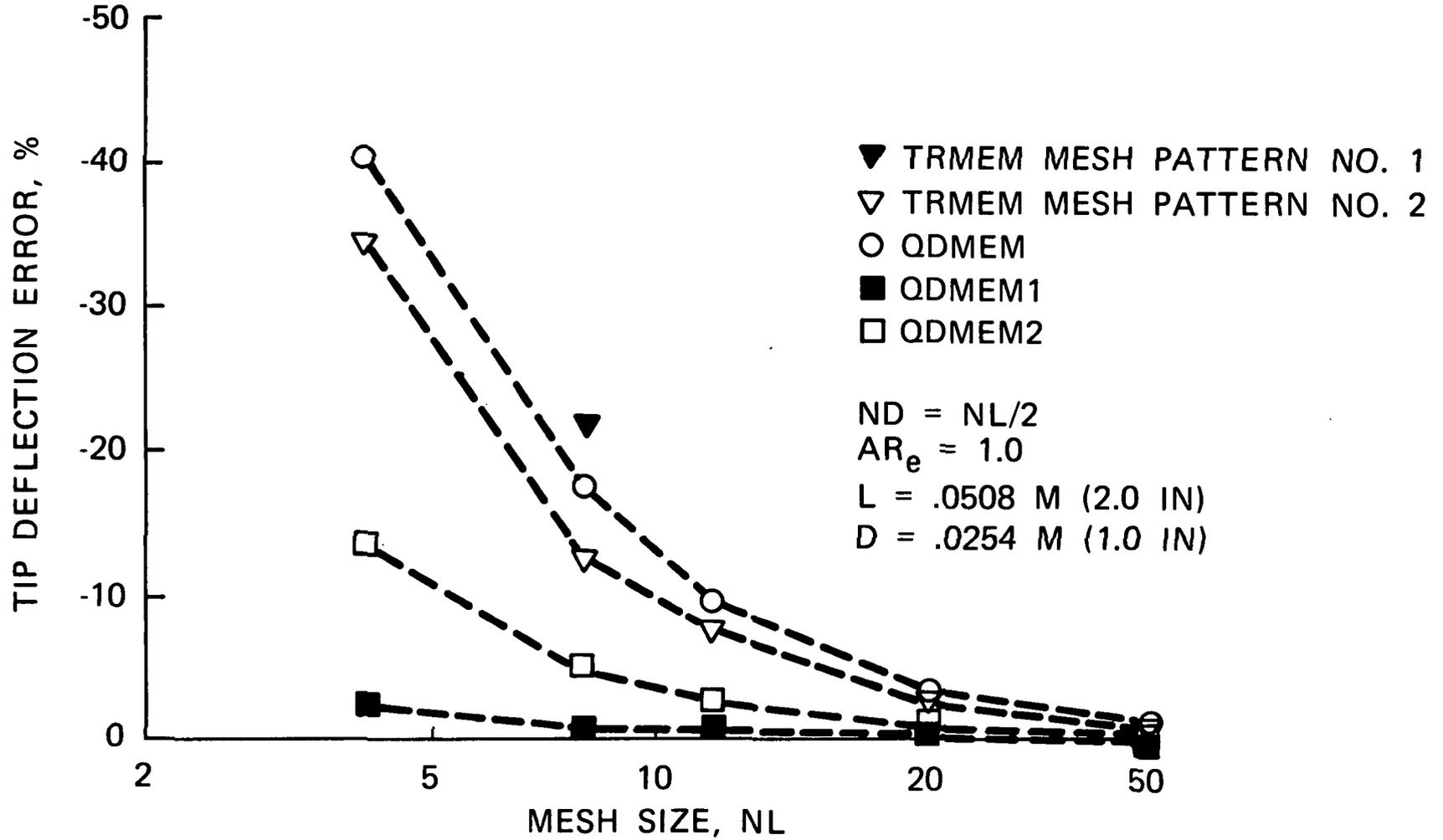


FIG. 6

DIRECT STRESS ERROR
DEEP BEAM-END MOMENT AND END SHEAR LOADING
(MESH SIZE STUDY)

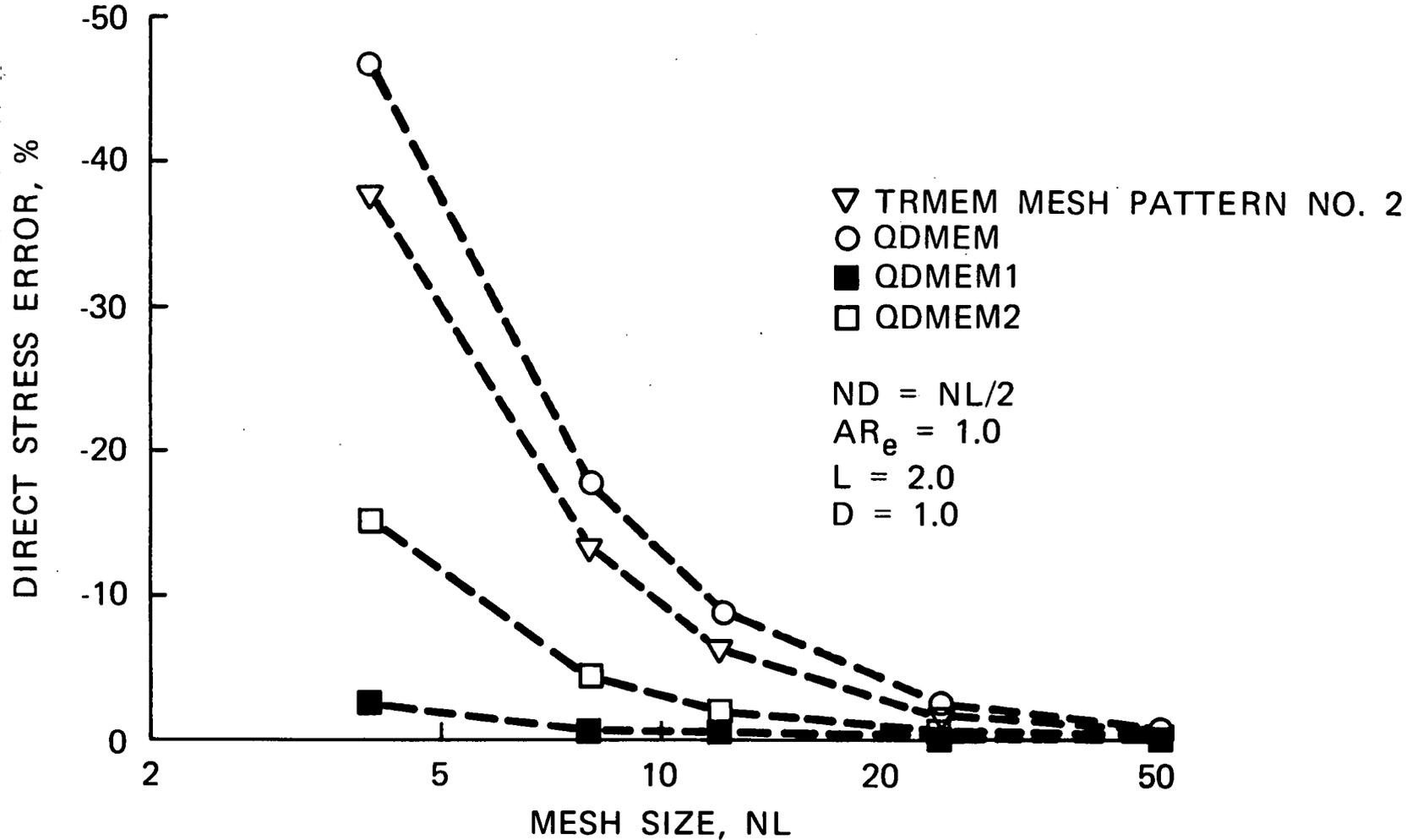


FIG. 7

SHEAR STRESS ERROR
DEEP BEAM-END SHEAR LOADING
(MESH SIZE STUDY)

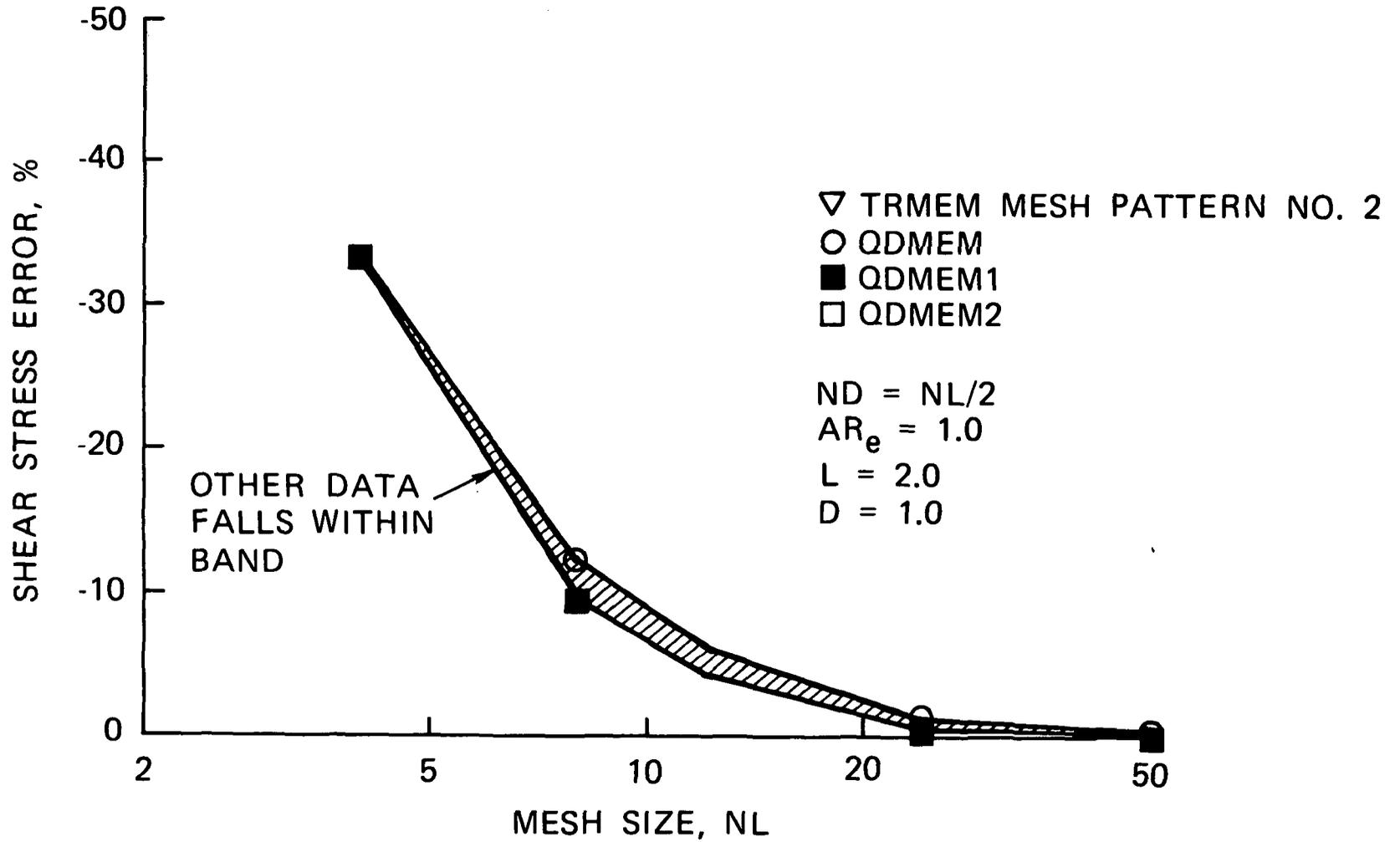


FIG. 8
TIP DEFLECTION ERROR
VARIABLE BEAM-END MOMENT LOADING
(ASPECT RATIO STUDY)

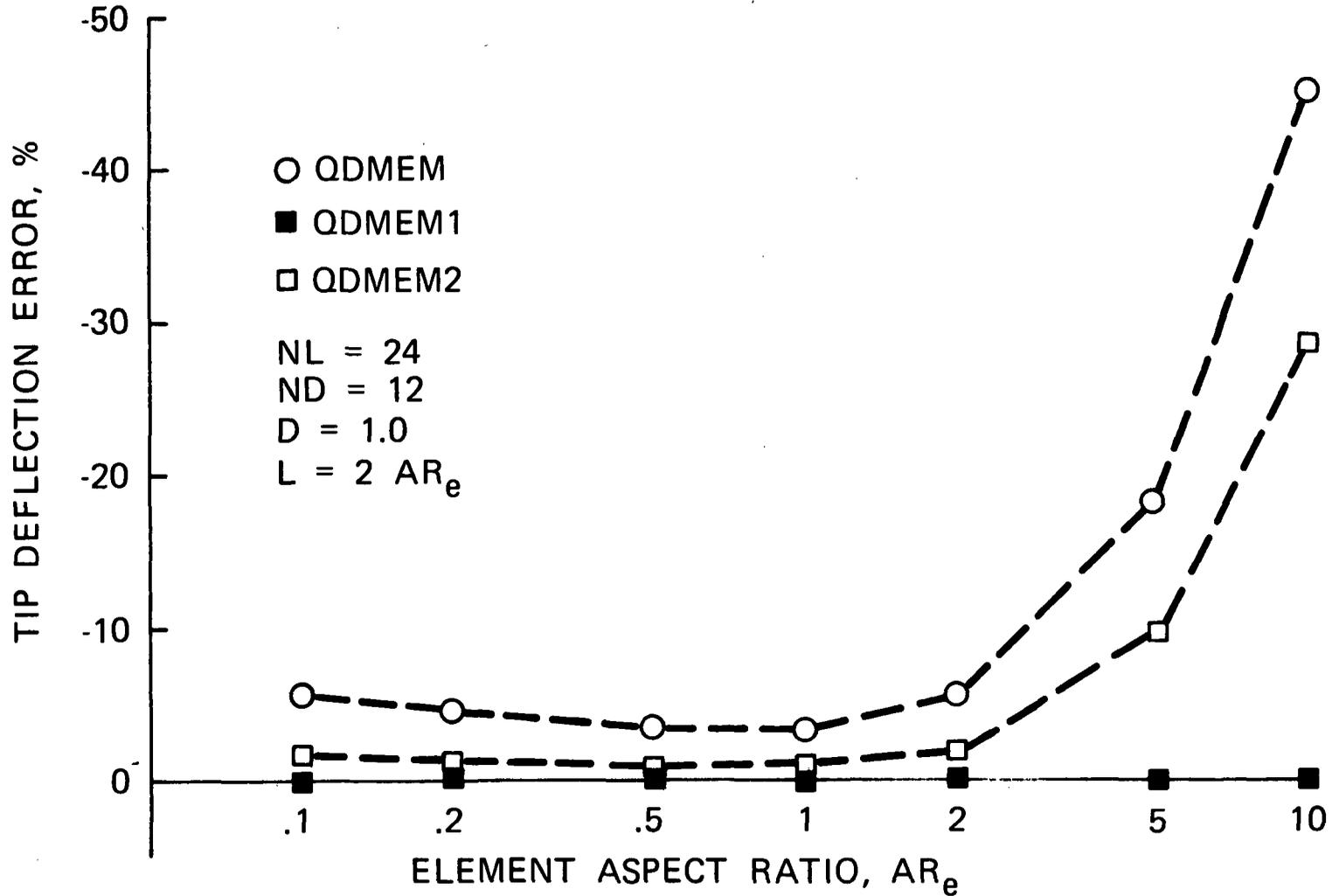


FIG. 9
 MID-SPAN STRESS DISTRIBUTIONS
 DEEP BEAM-END SHEAR LOADING
 NL = 48, ND = 24, AR_e = 1.0

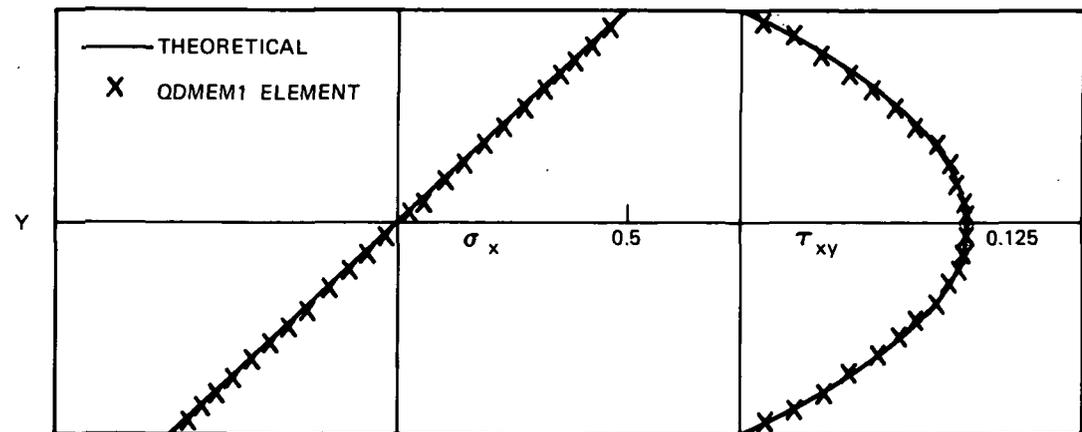
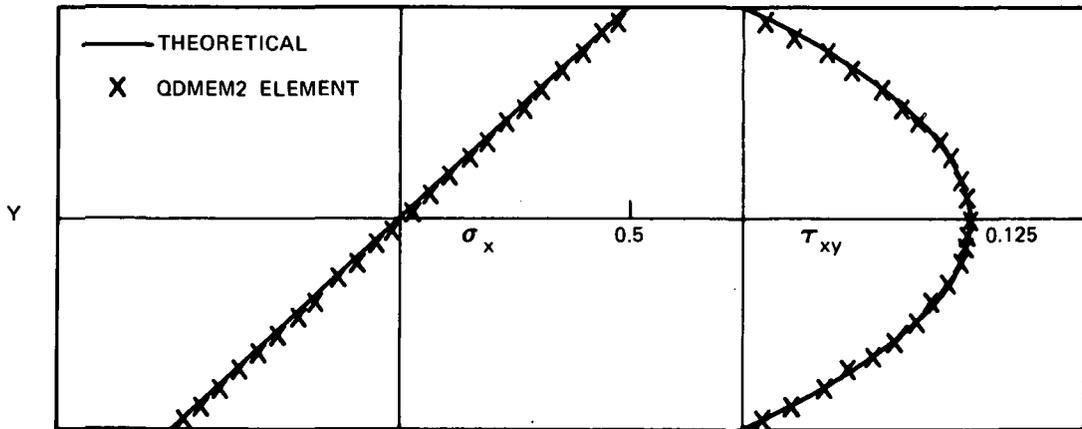
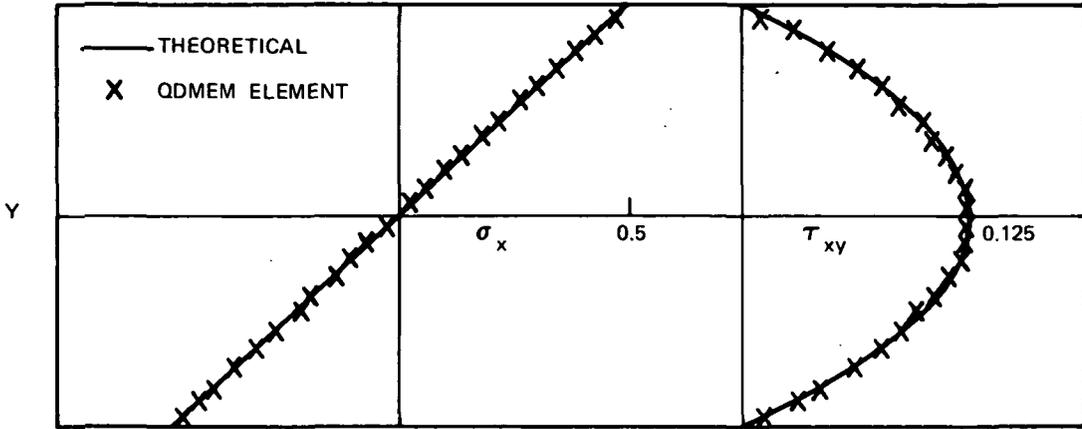


FIG. 10
 MID SPAN STRESS DISTRIBUTIONS
 DEEP BEAM-END SHEAR LOADING
 NL = 4, ND = 2, AR_e = 1.0

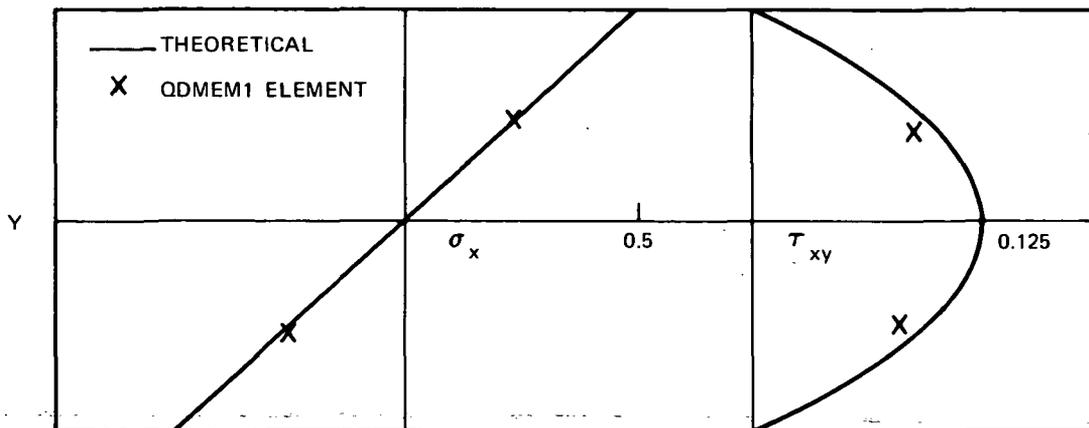
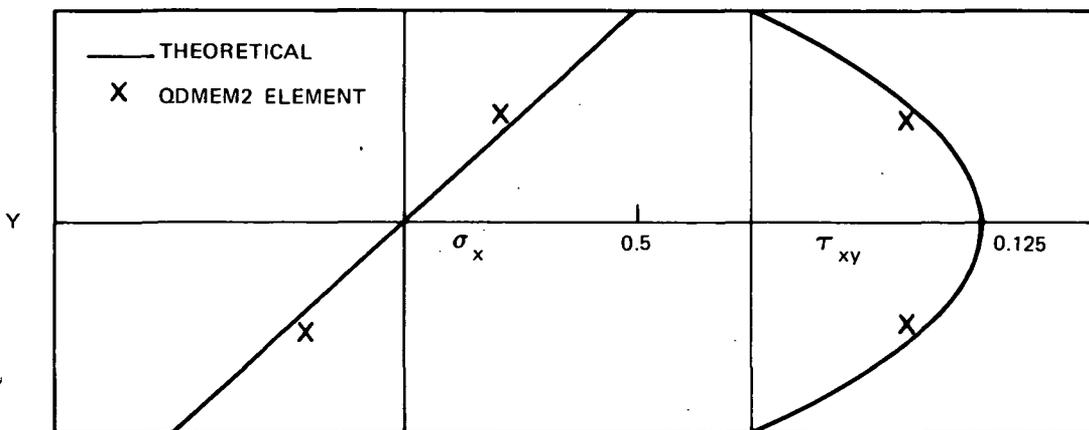
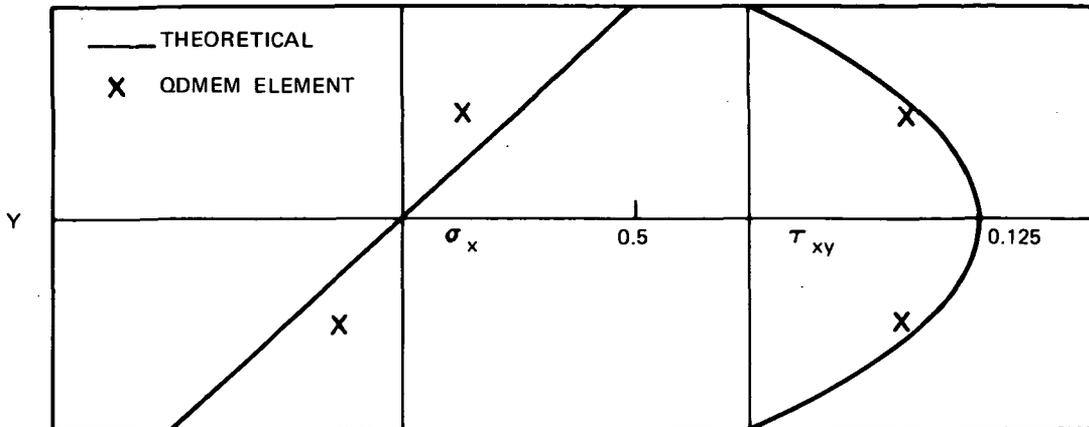


FIG. II
 MID-SPAN STRESS DISTRIBUTIONS
 DEEP BEAM-END SHEAR LOADING
 NL = 24, ND = 12, AR_e = 10.0

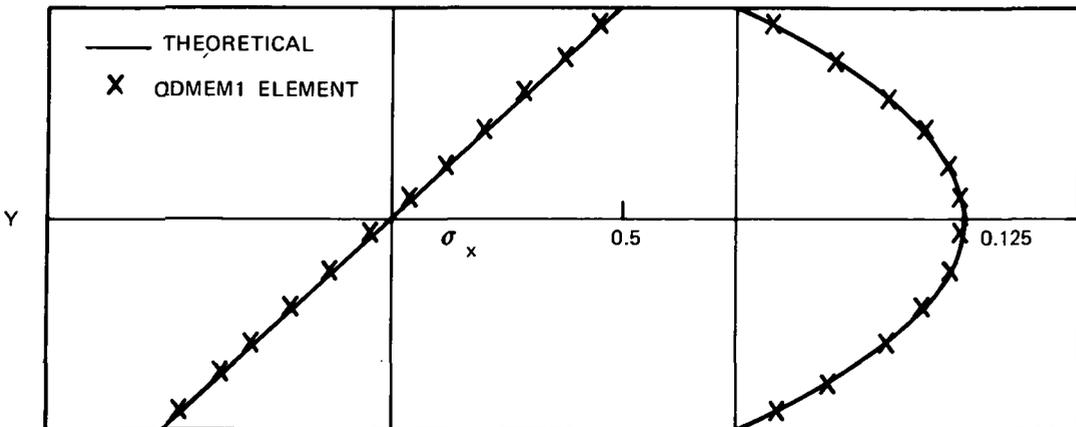
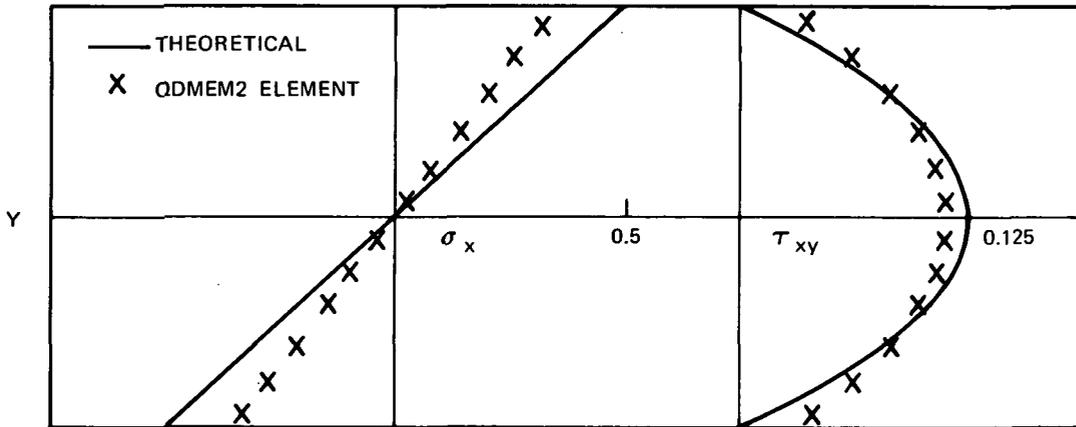
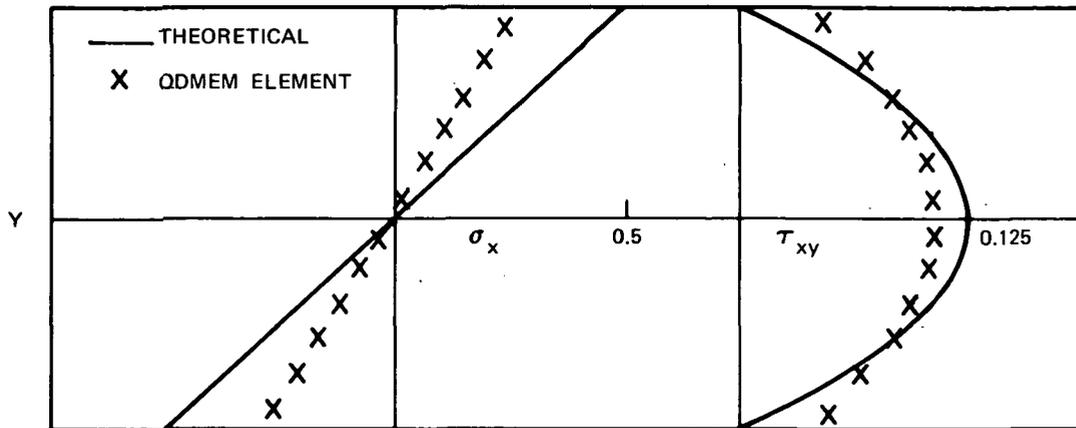


FIG. 12

MID-SPAN BENDING STRESS DISTRIBUTIONS
DEEP BEAM-END MOMENT LOADING
NL = 4, ND = 4

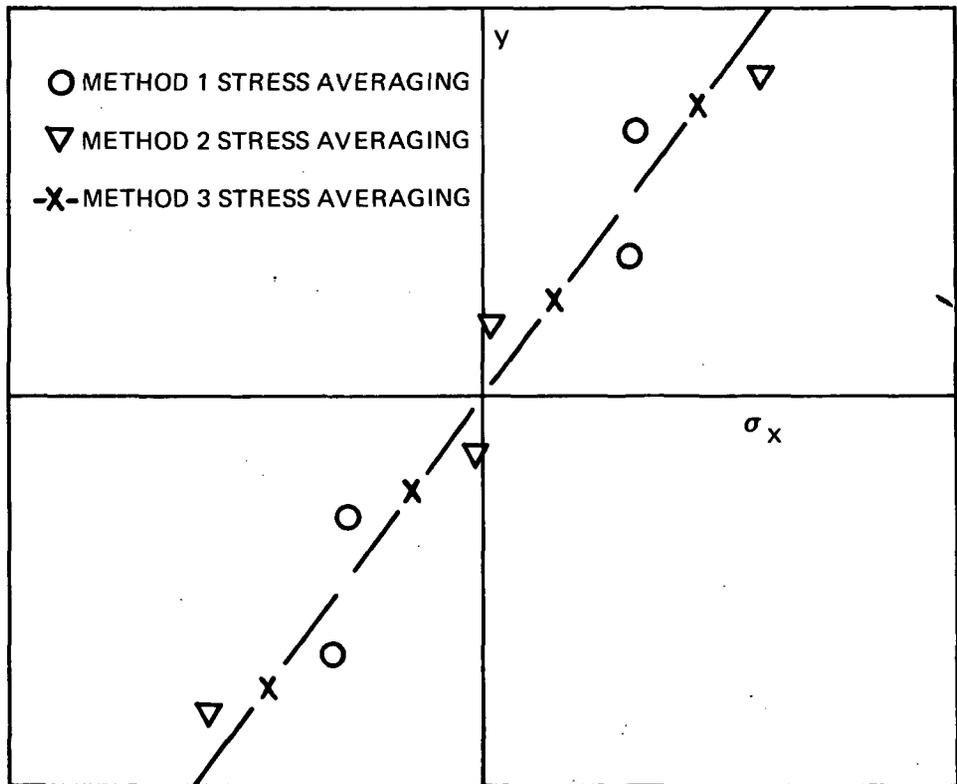
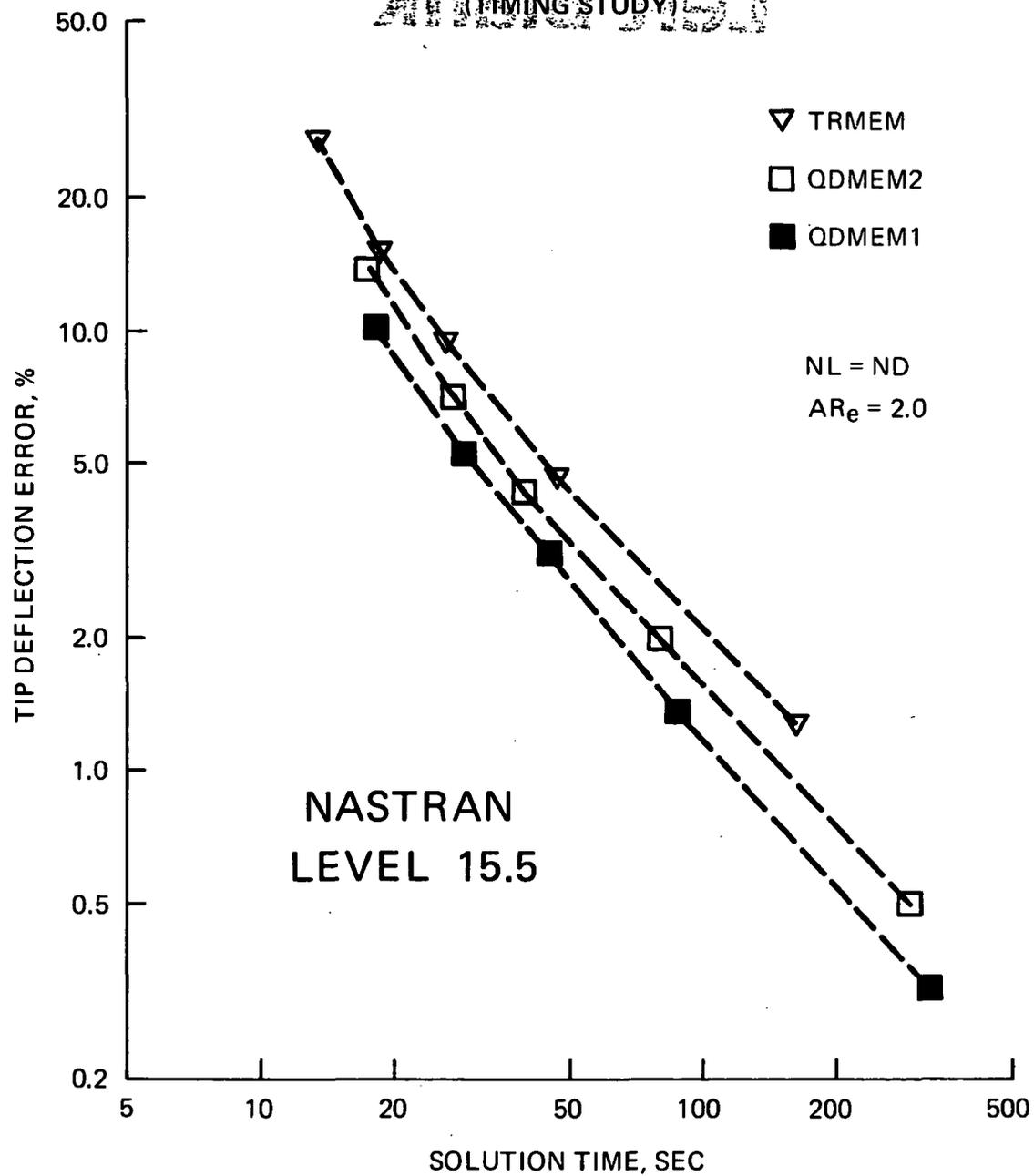


FIG. 13
 TIP DEFLECTION ERROR
 DEEP-BEAM-END MOMENT LOADING
 (TIMING STUDY)



Page intentionally left blank

STRESS ANALYSIS OF PRESSURE VESSEL ASSEMBLY

Richard T. Eppink and Furman W. Barton
University of Virginia

Lawrence H. Gilligan
Sperry Marine Systems

SUMMARY

A radome assembly capable of withstanding large hydrostatic pressures was designed with a sapphire hemispherical dome and a plastic housing. These two parts were connected with a stainless steel flange ring. A finite-element stress analysis of this assembly was accomplished and an experimental load test was performed to verify the predicted stresses.

INTRODUCTION

A recent contract of Sperry Marine Systems with the U.S. Navy required the design of a radome assembly capable of withstanding an external hydrostatic pressure of 7000 kPa (1000 psi) in a sea water environment. The pressure vessel is shown in figures 1 and 2. The complete pressure vessel assembly consists of three parts manufactured from three different materials. The left portion of the vessel (figure 1) is of Noryl plastic. The radome, at the right, is characterized by a hemispherical dome as an end enclosure. The dome is fabricated from a single crystal sapphire selected both for its strength and optical qualities. The dome is attached to the Noryl plastic housing with a stainless steel flange ring. A lapped surface interface between the sapphire and steel provides a high pressure water seal and a retaining ring provides a low pressure seal. The lapped-surface-type seal allows differential contraction between the steel and sapphire along the seal joint.

The lapped surface has a large radius of curvature rather than being flat; it is a portion of a large sphere of radius 165.1 mm (6.5 in.) rather than a shallow cone. Thus side motion is permitted to the extent of several micrometers without affecting the seating. This design reduces stress build-up and the possibility of seal fracture compared to a solid cemented mount.

The overall dimensions of the assembly are about 350 mm (14 in.) long by about 130 mm (5 in.) in diameter. The sapphire dome is approximately 150 degrees hemispherical with a thickness of 9.5 mm (0.375 in.) and an inner radius of 50.8 mm (2 in.).

The overall design required a detailed stress analysis of the proposed configuration and an experimental load test of a prototype model to verify the predicted stresses. The purpose of this study was to perform a complete analytical study of stresses and corresponding deformation in the entire radome assembly to ascertain stress magnitudes throughout the component and to identify regions of high stress where experimental measurements could be subsequently obtained. A primary concern in this analysis was the strength of the sapphire dome, since sapphire had not previously been used in an application of this type. The stress analysis was accomplished using NASTRAN.

Values are given in both SI and U.S. Customary Units. The measurements and calculations were made in U.S. Customary Units.

ANALYTICAL SOLUTION

The stress analysis of the pressure vessel assembly was accomplished using NASTRAN. Both triangular and trapezoidal axisymmetric solid ring elements were utilized in the idealized model of the vessel.

Solutions were obtained for a uniform external normal pressure of 6895 kPa (1000 psi). Results for other values of pressure could be obtained directly since linear behavior was assumed.

Two finite-element models of the pressure vessel were analyzed. One solution was obtained considering the entire pressure vessel assembly. Then a more refined model was studied wherein only the stainless steel ring and the sapphire dome were included in the model. Displacement boundary conditions at the plastic-stainless steel ring interface for this second case were taken to be those calculated from the solution for the complete assembly.

All three materials were assumed to provide linearly elastic behavior. The elastic properties used in the analysis are summarized in the following table:

Material	Modulus of Elasticity		Poisson's Ratio
	GPa	psi	
Stainless Steel	193	2.8×10^7	0.3
Noryl Plastic	2.45	3.55×10^5	0.38
Sapphire	365	5.3×10^7	0.26

The value of modulus of elasticity used for the sapphire represents an average value over different crystal orientations.

Model of Complete Pressure Vessel Assembly

The finite-element model of the complete pressure vessel assembly is shown in figure 3. This model contains 224 grid points and 307 ring elements. The Noryl plastic housing was taken to be fixed along the left edge in figure 3 since the actual plastic housing was restrained by a relatively rigid plate attachment.

The modeling of the interface between the sapphire dome and the stainless steel ring required special attention because both separation and relative sliding motion between the two parts along the interface were permitted. The relative sliding was implemented in NASTRAN through the use of multipoint constraints.

A trial-and-error-type solution was required in order to identify that part of the dome-ring interface over which separation occurred. As a first trial separation between the sapphire and the steel at all points on the interface was prevented. This solution revealed the presence of tensile forces between the dome and ring at two of the grid points on the external side of the dome. All other nodes on the interface had interconnecting compressive forces. Separation of the dome and ring was prevented by the constraints imposed in the analysis. However, the presence of tensile forces indicated that separation of the dome and ring actually would occur. In order to properly account for the separation in the analysis, it was necessary to totally release the connections between those nodes along the dome-ring interface at which separation occurs. This was done in considering the refined model of ring and dome discussed next.

Model of Sapphire Dome and Steel Flange Ring

The finite-element model of the sapphire dome and stainless steel flange ring is shown in figure 4. This model represents a

refinement of the model used previously to represent this portion of the pressure vessel assembly. This refined model contains 188 grid points and 288 elements.

The displacements along the surface of the ring in contact with the Noryl plastic were imposed to be those calculated for the complete pressure vessel assembly.

In a trial-and-error process, three solutions were necessary in order to adequately determine the separation that occurred between the dome and ring interface. In each of these solutions the connections between the grid points along the dome-ring interface were released in succession until the connecting tensile forces between the dome and ring became inconsequential. The results of this procedure showed that separation occurred along approximately two-thirds of the interface on the external side of the dome. Consequently, according to this solution, only about 3 mm (0.125 in.) of the dome thickness is in contact with the steel ring.

EXPERIMENT

The pressure vessel was tested in a pressure tank 330 mm (13 in.) in diameter by 1.5 m (5 ft.) long. It was subjected to hydrostatic pressures up to 7000 kPa (1000 psi). Five strain gages were attached to the vessel at critical regions, identified by the NASTRAN analysis. Signals from these gages were sent to a signal conditioner and then to an automatic data acquisition system.

Two-element gages were attached to the sapphire dome, about 2.5 mm (0.1 in.) from the steel flange interface, and to the steel flange about 2.5 mm (0.1 in.) below the sapphire dome; a single gage was attached to the Noryl housing about 5 mm (0.2 in.) below the steel flange. The locations of the gages are shown in figure 5. The biaxial gages on the dome and steel flange were oriented to measure strains in both the circumferential and meridional directions; the uniaxial gage on the housing was oriented to measure only the circumferential strain.

PRESENTATION AND DISCUSSION OF RESULTS

For an applied normal pressure of 6895 kPa (1000 psi) the maximum calculated stress in the sapphire dome was a compressive stress of 160 MPa (23,200 psi) and occurred at the inside corner where the dome contacts the steel flange ring. The stresses

change rapidly in this region. Consequently, this stress magnitude may be only an indication of the maximum stress. However, this maximum stress is well below the compressive strength of 2070 MPa (300,000 psi) for sapphire.

Stresses at the top of the dome varied from 13.8 MPa (2000 psi) compression at the inner surface to 28.3 MPa (4100 psi) compression at the outer surface. It is of interest to compare these stresses with the theoretical stresses in a comparable spherical shell. For the spherical shell, the stress at the inner surface is 25.7 MPa (3700 psi) and is 22.2 MPa (3200 psi) at the outer surface, both compressive. The calculated stresses indicate a larger stress gradient through the dome thickness for the pressure vessel, but the order of magnitude is about the same.

The maximum calculated stress in the steel flange occurred near the inside corner of the dome-flange interface and had a magnitude of 135.8 MPa (19,700 psi). However, this is a highly localized stress and probably not a good indicator of the maximum stress associated with failure of the steel flange. At the inner surface near the interface between the ring and the Noryl housing a compressive stress of 68.3 MPa (9900 psi) was calculated.

The maximum calculated stress in the Noryl plastic was 39.3 MPa (5700 psi) and occurred about 80 mm (3 in.) from the supported edge of the housing.

The compressive strengths of the steel and plastic are approximately 590 MPa (85,000 psi) and 115 MPa (16,500 psi), respectively. The above results indicate that the design of the pressure vessel for a normal pressure of 7000 kPa (1000 psi) is adequate.

A comparison of the calculated and measured strains is provided in Table 1. Generally, the calculated strains agree satisfactorily with the measured strains. The values presented, however, should be examined and interpreted in an approximate sense because of practical limitations of both the experimental study and the NASTRAN analysis. The strain gage length, although small, is still necessarily finite and it is not possible to determine precisely the point at which the measurement was made. Similarly, the element size used in the analysis does not allow a precise determination of the location of the calculated strain.

The tabulated calculated strains show a linear increase with increasing pressure which is a result of the assumption of linear behavior. The measured strains are also approximately linear. The largest nonlinear behavior was observed in the circumferential strain measurements of the sapphire dome and stainless steel flange.

In the dome the calculated and measured strains in the meridional direction agreed rather well. They were of the same order of magnitude, about 200 $\mu\text{m}/\text{m}$ at 6895 kPa (1000 psi). In the circumferential direction the measured strain was about half the predicted strain. This difference may partly be explained by the assumption in the analysis of a flat (conical) lapped interface between the dome and flange instead of the large spherical radius of the actual interface, consequently, the analysis may predict a higher stress concentration at the inside corner than actually exists due to the combination sliding and rolling action on the spherical interface.

From Table 1 it appears that the measured strains in the flange agree well in the meridional direction. However, in the circumferential direction the table indicates predicted strains of only about a tenth of the measured values. It should be noted, however, that the strain gradients along the meridian of the flange are large. The calculated meridional strain varies from about 200 $\mu\text{m}/\text{m}$ (compression) at the dome edge to about 50 $\mu\text{m}/\text{m}$ (compression) at the Noryl plastic edge for a pressure of 6895 kPa (1000 psi). The circumferential strain varies from about 50 $\mu\text{m}/\text{m}$ (tension) at the dome edge to about 300 $\mu\text{m}/\text{m}$ (compression) at the plastic edge for the same pressure loading. The best calculated value for the circumferential strain corresponding to the gage location is 10 $\mu\text{m}/\text{m}$ but at a distance of about 4 mm (0.15 in.) closer to the plastic housing the calculated strain is about 100 $\mu\text{m}/\text{m}$, about the same as the measured value. The conclusion is that the measured and predicted strains are of the same order of magnitude and agree satisfactorily.

The calculated strain in the Noryl plastic was about a third of the measured value. This may partly be explained by the assumption in the analysis that the plastic and the steel displaced together at their interface. Actually sliding between the two surfaces was possible so that a relative displacement particularly near the inner surface could occur. Consequently, the strain in the plastic might be larger than was calculated. Also, properties of thermoplastic materials may easily deviate greatly from manufacturer's specifications of physical parameters. Since the design of the sapphire dome and not that of the housing was the prime consideration of this study, the large discrepancy in these strains was not investigated further. A more refined finite-element model of the housing would be required to represent the behavior more precisely.

CONCLUDING REMARKS

The finite-element stress analysis of the pressure vessel assembly indicated maximum stresses in the sapphire dome of

160 MPa (23,200 psi) for an external normal pressure of 6895 kPa (1000 psi). The corresponding maximum stress in the stainless steel flange ring is 68.3 MPa (9900 psi). The maximum stress in the Noryl plastic was calculated to be 39.3 MPa (5700 psi). Considering the strengths of the three materials composing the pressure vessel assembly, ultimate failure would occur in the plastic at a pressure level of approximately 20.7 MPa (3000 psi).

A comparison of the strains measured experimentally with those determined from the NASTRAN study shows satisfactory agreement. For the sapphire dome and stainless steel flange the discrepancies can be explained partly by uncertainties in the precise locations of the measured and experimental values.

TABLE 1. COMPARISON OF CALCULATED AND MEASURED STRAINS

Pressure		Location	Measured Strain	Calculated Strain
kPa	psi		$\mu\text{m}/\text{m}$	$\mu\text{m}/\text{m}$
1724	250	Dome Circumferential	38	34
		Dome Meridional	56	52
		Flange Circumferential	42	2
		Flange Meridional	52	45
		Housing	1119	350
3448	500	Dome Circumferential	54	68
		Dome Meridional	106	104
		Flange Circumferential	68	5
		Flange Meridional	99	90
		Housing	2408	700
5171	750	Dome Circumferential	64	101
		Dome Meridional	152	155
		Flange Circumferential	85	8
		Flange Meridional	140	136
		Housing	3674	1050
6895	1000	Dome Circumferential	76	135
		Dome Meridional	191	207
		Flange Circumferential	104	10
		Flange Meridional	183	181
		Housing	5078	1400

413

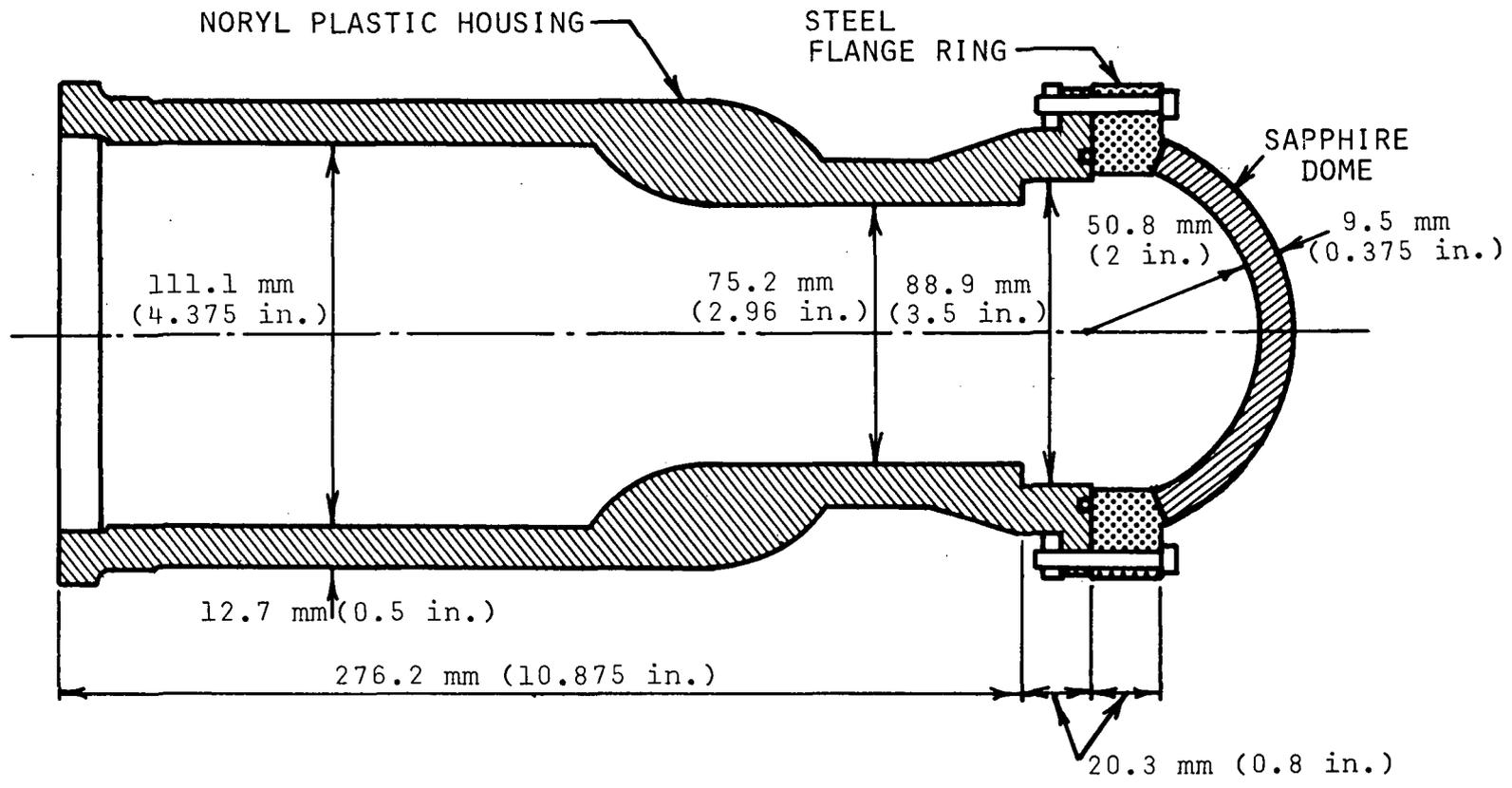


FIG. 1 DRAWING OF PRESSURE VESSEL ASSEMBLY

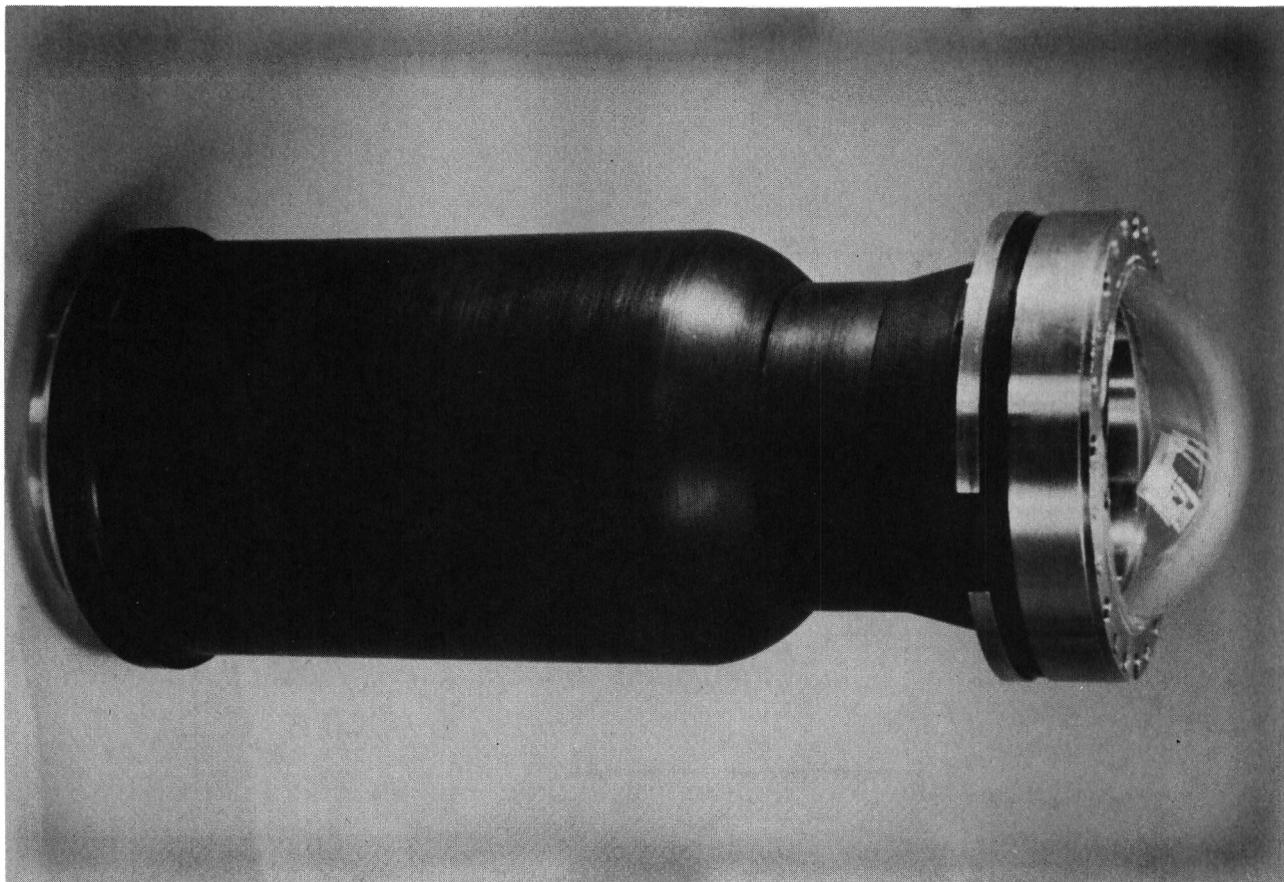


FIG. 2 PHOTOGRAPH OF PRESSURE VESSEL ASSEMBLY

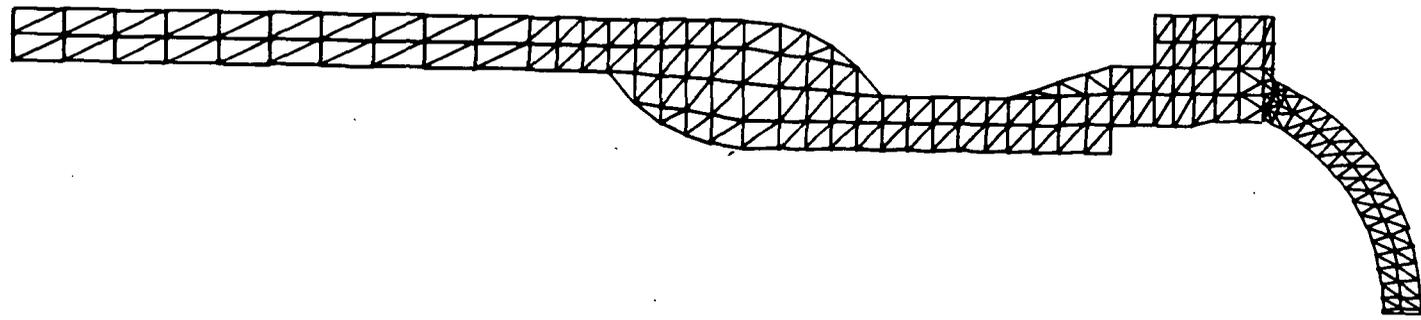


FIG. 3 MODEL OF COMPLETE ASSEMBLY

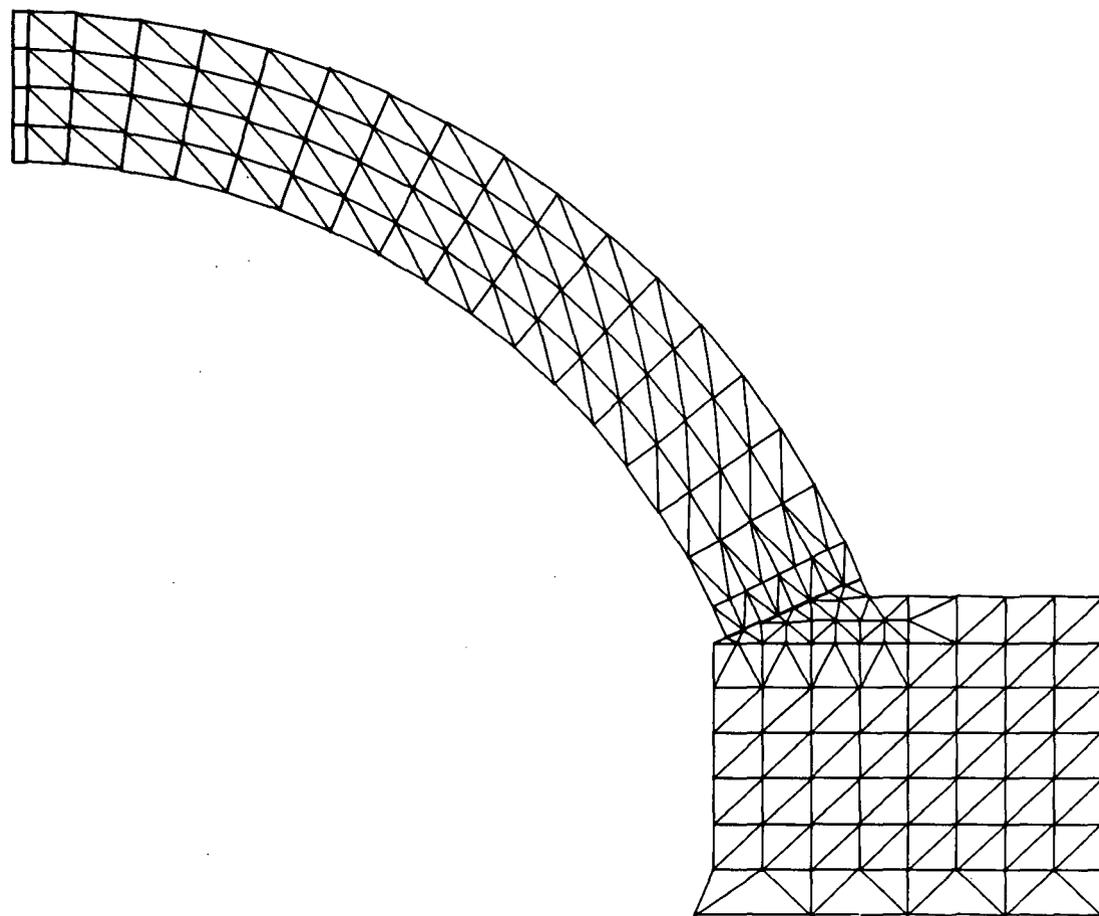


FIG. 4 REFINED MODEL OF DOME AND FLANGE RING

2025
November 2025
Khalid Alkhalid

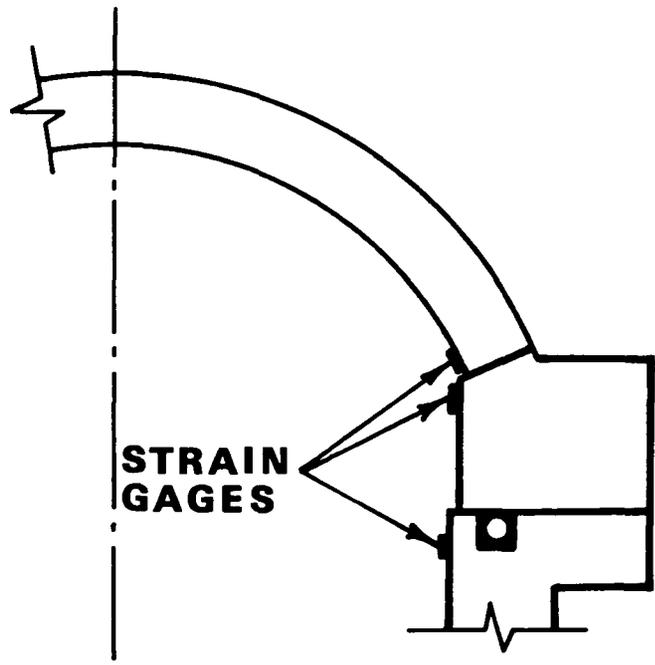


FIG. 5 STRAIN GAGE LOCATIONS

**Page
Intentionally
Left Blank**

AN AUTOMATED DATA GENERATOR FOR NASTRAN

Edward L. Stanton

Prototype Development Associates, Inc.

SUMMARY

A modeling system based on construction-in-context of geometry and physical properties data is adapted for NASTRAN finite element data generation. This system was originally developed for generating three-dimensional finite element data for composite structures in PATCHES-III using parametric cubic models. These finite line, surface and volume models constructed using simple data directives will be subdivided into NASTRAN finite elements and their supporting data in the new data generator. Any geometry can be constructed as illustrated by a compressor fan blade with camber, twist and changing airfoil section and the shape then subdivided using a uniform or non-uniform mesh. Special emphasis is given to the adaption of construction-in-context to composite material property modeling. A PMAT series of directives is described for any one, two or three-dimensionally reinforced composite that allows the phase properties, as well as composite properties, to be generated.

INTRODUCTION

The need for a comprehensive data generator for NASTRAN is manifest in the growing number of pre-processor programs developed in the last few years that provide specialized data generation capability for a variety of industrial and NASA applications. A parametric cubic modeling system based on construction-in-context is presented for the generation of NASTRAN geometry data, physical data and element data, including a comprehensive composite material property data generator. These models for any one, two or three-dimensional region in a structure are then subdivided into a user specified number of NASTRAN finite elements. The particular NASTRAN finite element and mesh density selected determine the grid points generated for that region or substructure. Uniform and nonuniform mesh spacings can be generated and the physical properties can vary over the region. These data are also generated automatically using data line, data patch and data hyperpatch directives for construction. Individual elements with irregular properties may exist within any region and these are handled using direct input which overrides data generator values. To illustrate the generality and simplicity of the system, a compressor fan blade airfoil with camber and twist is modeled using construction-in-context.

Generation of effective modulus composite material properties is available for particulate composites as well as uniaxial, biaxial and multiaxial fiber reinforced composites. Special user convenient formats are provided for laminates. The default property model is a rule-of-mixtures with a number of special purpose models, like the composite cylinder assemblage, available on request. Property models also may use construction-in-context so that ply properties generated by one directive can be used to generate laminate properties by another. This procedure is simple, yet very effective in practice. In addition to generating bulk data for NASTRAN, the data generator will output printed summary information for each region and for the composite material properties generated.

PARAMETRIC GEOMETRY MODELING

Computer aided geometric design of structural shapes has progressed rapidly in recent years and is used for the basic engineering definition of geometry in several industries. To take full advantage of these advances for finite element model construction requires a familiarity with parametric representations for lines, surfaces and volumes; however, no formal mathematical constructions are required. The situation is entirely analogous to having a familiarity with the properties of a finite element without requiring the mathematical details of its derivation. The object of the present discussion is to define basic terms and equations associated with parametric cubic modeling in the PATCHES data generator being adapted for use in NASTRAN. The essential feature of any parametric geometry model is the mapping of a simple shape in parametric coordinates, ξ_i , into a complex shape in Euclidian space, $Z(\xi_i)$, as illustrated in Figure 1. This allows the computation of geometric properties or transformations in convenient normalized coordinates and in a space where slope singularities are not a problem. In the PATCHES system, the mappings are parametric cubics defined over the unit interval for finite lines, over the unit square for finite surfaces or patches, and over the unit cube for finite volumes or hyperpatches. Any parametric cubic shape can be defined uniquely in three different but mathematically equivalent ways:

- Algebraic, Defined by the coefficients of the parametric coordinates ξ_i .
- Geometric, Defined by the value of Z and Z, ξ_i at the corners, $\xi_i = 0, 1$.
- Point, Defined by the values of Z at the one-third points, $\xi_i = 0, 1/3, 2/3, 1$.

The symbols normally used for these coefficients are S_i , B_i and P_i , respectively, and from long experience no single format is efficient or convenient for all constructions. A mathematical development of the properties of parametric cubics for lines and surfaces is provided by Coons (Ref. 1) and elsewhere (Ref. 2) for solids. It is the

relative simplicity with which these properties can be used to construct accurate models for any shape that accounts for their use.

Finite Lines

A parametric cubic line is simply a mapping of the unit interval $[0, 1]$ into a finite length space curve as illustrated in Figure 1 and defined by the following equation.

$$\underline{Z}(\xi) = Z_1(\xi)\underline{e}_1 + Z_2(\xi)\underline{e}_2 + Z_3(\xi)\underline{e}_3 \quad (1)$$

Each coordinate function, $Z_i(\xi)$, is a cubic or lower order polynomial and the \underline{e}_i are the unit vectors for a reference Cartesian coordinate frame. The most important feature of parametric modeling is the ability to adjust the focus (i. e., parametrization) so that regions of rapid change receive the greatest detail. Note that this refers to both initial geometry and deformed geometry, as well as boundary points and interior points. Consider, for example, a parametrization used for crack tip models (Ref. 3) in which the initial geometry is simple but the deformed geometry results in a strain singularity at $\underline{Z}(0)$ in the \underline{e}_1 direction.

$$\underline{Z}(\xi) = (\xi^2/L)\underline{e}_1 + Z_2(\xi)\underline{e}_2 + Z_3(\xi)\underline{e}_3 \quad (2)$$

This parametrization of the coordinate function $Z_1(\xi)$ in algebraic format is trivial to define using a simple LINE directive from Table I. In general, lines constructed using directives of the form LINE __ from this table are piecewise cubic over each segment of the line. The benchmark data associated with each directive usually fits on one or two bulk data cards (Ref. 4) in familiar NASTRAN syntax. The user is not required to specify any of the mathematical details, they are automatically supplied by the directive.

Finite Surfaces

A bicubic surface patch maps the unit square into a finite area surface as illustrated in Figure 1 and defined by the following equation.

$$\underline{Z}(\xi_1, \xi_2) = Z_1(\xi_1, \xi_2)\underline{e}_1 + Z_2(\xi_1, \xi_2)\underline{e}_2 + Z_3(\xi_1, \xi_2)\underline{e}_3 \quad (3)$$

Each coordinate function $Z_i(\xi_1, \xi_2)$ is a bicubic or lower order bivariate polynomial analogous to the univariate polynomials used for line coordinate functions. In fact, they are the product (i. e., tensor product) of the same cubic polynomials, and nonuniform parametrizations (i. e., nonuniform meshes) are available using directives of the form PATCH __ from Table I. Each patch normally has four sides, but three sided patches can also be constructed. Note that Equation (3) completely defines the geometry of the surface. This allows the computation of surface areas, normals, curvatures and any other geometric property without additional user input. Note also that the shape is not restricted to constant curvature or a reference conic surface.

Finite Volumes

A tricubic hyperpatch maps the unit cube into a finite volume hexahedra as illustrated in Figure 1 and defined by the following equation.

$$\underline{Z}(\xi_1, \xi_2, \xi_3) = Z_1(\xi_1, \xi_2, \xi_3)\underline{e}_1 + Z_2(\xi_1, \xi_2, \xi_3)\underline{e}_2 + Z_3(\xi_1, \xi_2, \xi_3)\underline{e}_3 \quad (4)$$

The coordinate functions, $Z_i(\xi_1, \xi_2, \xi_3)$, are tricubic or lower order multivariate polynomials that reduce to bicubic patch functions on external surfaces and cubic line coordinate functions along the edges of the hexahedra. Any of the external surfaces of a hexahedron may degenerate to a line or a point forming a pentahedron or a tetrahedron. This is illustrated in Figure 2 in which all of the six hyperpatches constructed using the HPR directive from Table I are degenerate. Note in particular the first hyperpatch which is a curvilinear tetrahedra. Each hyperpatch has 192 coefficients, 64 for each coordinate function, that for this tetrahedron were automatically generated from two grid cards, one LINE directive, one PATCHR directive and one HPR directive, all of very simple format. Also note that as with a surface patch, the hyperpatch given by Equation (4) completely defines the geometry of the solid for the computation of volumes or any geometric property. The accuracy of shapes such as those in Figure 2, modeled with parametric cubics is very high. Typical maximum deviations for a single line, patch and hyperpatch shown in Table II are all less than 0.1 percent for sectors up to ninety degrees.

Construction-in-Context

The construction procedures long used by designers to produce engineering drawings have been adapted in recent years to computer aided geometric design systems often using interactive graphics equipment (Ref. 5). The essential feature of any such system lies in the use of references to data created by other operations. This can be accomplished interactively from a terminal or it can be accomplished in a batch system using a queuing algorithm to sequence the directives for serial processing. This latter approach is used in the new NASTRAN data generator which makes the system device independent. However, the directives could easily be used in an interactive mode with the proper interface.

To illustrate the generality and ease of construction of the new system, a compressor fan blade airfoil recently analyzed by Chamis and Minich (Ref. 6) was modeled using four hyperpatches. The PATCHES-III code itself was used for this construction in that work on the NASTRAN data generator had just started as of this writing. A plan view of the model showing the ξ_1 one-third lines is shown in Figure 3, and a perspective view in Figure 4 shows the camber and twist of the compressor blade. This model was constructed from airfoil cross-section data in Reference 6 in less than one day, including plots. The directives used are summarized in Table III and include no GRID cards. All points on the three airfoil sections given in Reference 6 were input directly

on LINE cards. The spanwise sections were generated using LINECS directives which fit cubic splines through the chordwise airfoil sections. Upper and lower surfaces were then generated using PATCHL directives and the solid geometry model was obtained by connecting these surfaces with HP2PAT directives. No smoothing or any adjustment to the original data has been made, although that could easily be accomplished if necessary for aerodynamic calculations. It would also have been a simple matter to construct a four patch model of the midsurface of the blade with the blade thickness defined by four data patches which are described in the next section.

PHYSICAL PROPERTY MODELING

Once the geometry for a structure has been defined using parametric cubic shape functions, the distribution of physical data over the model becomes easier to construct. Pressures over a surface patch, for example, require only a definition of magnitude in that the normal to the surface is automatically determined at any point on the surface by the geometry model. The same is true of any load or constraint data whose definition depends in some way on a property of the geometry model. A second important advantage of modeling in parametric coordinates is the ability to use construction-in-context for data lines, data patches and data hyperpatches with directives from Table I analogous to those used for geometry construction. The data generated using these directives are defined independent of physical units which are implicit to the geometry model or other element data.

As a result of parametric data modeling, any finite element property such as thickness may vary over a geometric region. The PFEG card, Figure 5, controls the generation of property cards over regions defined by a line, patch or hyperpatch. The exact format of this card may change before final release, but the basic approach, as shown in Figure 5, is to satisfy blank fields on the NASTRAN property cards using the data models referenced by PFEG. Any data field specified a value by a NASTRAN property card will take precedent over the value from a data model. This allows discontinuities or other local irregularities in properties to be inserted in a field without losing the convenience of data generation for smoothly varying properties such as tapered thickness.

The generation of constraint and load data will follow basically the approach taken in PFEG. Data construction will be performed using the appropriate PATCHES-III option to define the constraint or mechanical load over a region. Any element in this region then has access to these data in a manner analogous to the PFEG access to property data. In keeping with user convenience, the data function value for any parameter may be overridden by NASTRAN input.

Finite Element Connectivity

The CFEG option, shown in Figure 6, controls subdivision of parametric cubic shapes into NASTRAN finite elements. The identification numbers of the data generator grid points at the corners of each shape will be used to define NASTRAN grid point connectivity. This technique was originally developed to generate the connectivity for 64 mesh points in a PATCHES-III solid element from the connectivity of the eight corner grid points. Control of grid point sequencing is thus exercised on a much smaller set which facilitates optimal sequencing. The technique was successful in creating stiffness matrices that NASTRAN matrix utilities could efficiently factor in spite of the very large dimension (192 x 192) of the element stiffness matrices. The generalization of the technique to arbitrary subdivisions is straightforward. A data line, patch or hyper-patch, depending on the shape, is constructed from the corner identification numbers treated as floating point numbers. The value of this data function at any point in parametric space, determined by any subdivision, can be used to uniquely identify that point for connectivity purposes. All common points on "substructures" that share a common boundary automatically are connected by this procedure. If unequal subdivisions are specified, only the common points are connected. A bias constant is allowed if a user wishes to decouple grid point sequencing for a region. The finite element identification numbers are assigned sequentially as in a Fortran array with indices determined by their position in parametric space. A bias constant is input by the user to avoid duplicate element identification numbers. This is compatible with the common practice of making all elements of a given type, such as CQUAD2, have the same first digit, for example the 1XXX series.

Composite Material Properties

The generation of effective modulus data for composite materials is an increasingly important aspect of finite element modeling. At the time of the original NASTRAN development, there was little use of composites in NASA vehicles or in aircraft structures and as a consequence very few features were provided for composite properties. Today, laminated composites are common and a variety of other fiber reinforced composites are increasingly used in structural components. The analyst typically has data on the phase materials and needs properties for the composite to analyze the structure. Pre-processor programs (Ref. 7) are available for laminates that are very useful in preparing data for NASTRAN models. The design of the new data generator adapts models of this type to allow construction-in-context of composite properties. This will be described after first describing the PMAT __ directives.

The generation of composite properties is controlled by PMATL, PMATGL and PMATX directives as described in Figures 7, 8 and 9. A basic rule-of-mixtures approach is the default model when a particular composite property model is not specified. A variety of Voigt-Reuss models, the Halpin-Tsai model and others are being considered for inclusion. The PMATX directive generates properties for any N-phase

composite based on phase properties, orientation and other data such as volume fraction required by the model selected. The PMATL and PMATGL directives are intended as user convenient subsets of PMATX for laminated composites.

The construction-in-context approach used elsewhere in the data generator will also be used for composite properties. This allows the phase material properties to be generated by another PMAT__ card. Ply properties, for example, might be the result of a PMATX execution using fiber and matrix properties which would then be available for a PMATL execution to compute laminate properties. In this instance, construction-in-context eliminates the need for multiple pre-processor runs using different composite material property models. This simple design is easy to use, yet very effective in practice. It has been used to generate thermal, as well as elastic, properties for a wide variety of advanced composites.

REFERENCES

1. Coons, S. A., "Surfaces for Computer Aided Design of Space Forms," Massachusetts Institute of Technology, Report No. MAC-TR-41, 1967.
2. Stanton, E. L., L. M. Crain, and T. F. Neu, "A Parametric Cubic Modelling System for General Solids of Composite Material," *Int. J. Num. Meth. Engng.*, Vol. 11, 1977, pp. 653-670.
3. Henshell, R. D. and K. G. Shaw, "Crack Tip Finite Elements are Unnecessary," *Int. J. Num. Meth. Engng.*, Vol. 9, 1975, pp. 495-507.
4. Stanton, E. L. and L. M. Crain, "PATCHES-III User's Manual," McDonnell Douglas Astronautics Company Report No. MDC G5538, 1974.
5. Peters, G. J., "Interactive Computer Graphics Application of the Parametric Bicubic Surface to Engineering Design Problems," in Computer Aided Geometric Design (Ed. R. E. Barnhill and R. F. Risenfeld), Academic Press, N.Y., 1974, pp. 259-302.
6. Chamis, C. C. and M. D. Minich, "Structural Response of a Fiber Composite Compressor Fan Blade Airfoil," NASA TM X-71623, 1975.
7. Chamis, C. C., "Computer Code for the Analysis of Multilayered Fiber Composites-User's Manual," NASA TN D-7013, 1971.

TABLE I. NASTRAN DATA GENERATOR DIRECTIVES*

1. GEOMETRY

GRID	LARCPC	PATCH	HPATCH	SCALP
	LINE	PATCHB	HPL	SCALPH
	LINEB	PATCHGR	HPHEX	TMOVE
	LINECS	PATCHL	HPN	
	LINEGR	PATCHO	HPR	
	LINEPC	PATCHQ	HP4PAT	
		PATCHR	HP2PAT	
		PATCH4L	HP6PAT	
			HPB	
			HPP	

2. ELEMENT AND PROPERTIES

CFEG	MATAL	MAT1	MTRX-CID
PFEG	MATC	MAT2	MTRX-ID
PMATL	MATE	MAT3	
PMATGL	MATOR	MAT5	
PMATX		MAT6	

3. CONSTRAINTS

SDC10	SDC1	SPC1
SDC20	SDC2	SPC2

4. LOADS

FORCE	FORCEL	FORCET	TEMP
		PLOAD	

5. DATA MODELING

DATAG	DLINE	DPATCH	DHPAT
	DLINCS	DPATA	DHPHEX
	DLINP	DPATEQ	DHPL
	DLINPC	DPATL	DHP2P
		DPATP	DHPSORT
		DPAT4L	DHP4P

*Minor changes may occur before final release.

TABLE II. PARAMETRIC CUBIC SHAPE DEFINITION ACCURACY

	Circular Arc	Spherical Surface	Spherical Volume
θ	$\delta R/R$	$\delta A/A$	$\delta V/V$
45°	.000004	.000013	.000005
90°	.00025	.00070	.00037

TABLE III. COMPRESSOR BLADE GEOMETRY DATA CONSTRUCTION

Directive	Number of Directives	Number of Cards
LINE	12	33
LINECS	6	6
PATCHL	8	16
HP2PAT	4	4

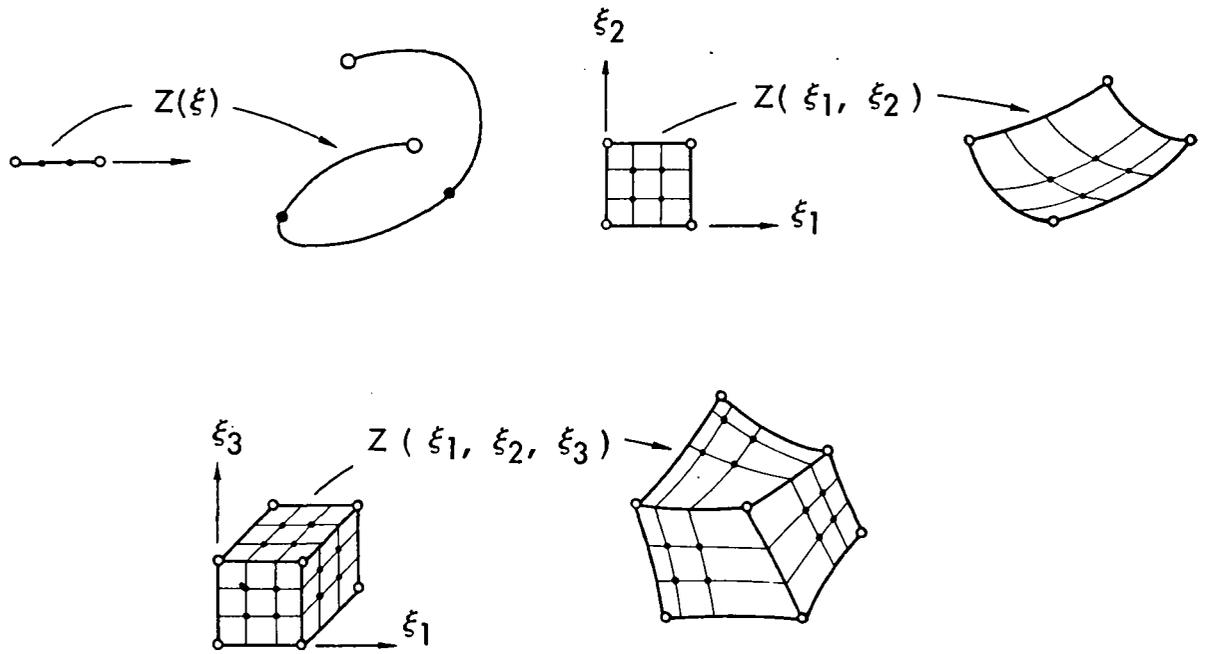


Figure 1. Parametric Cubic Mappings of Line, Surface and Volume Models

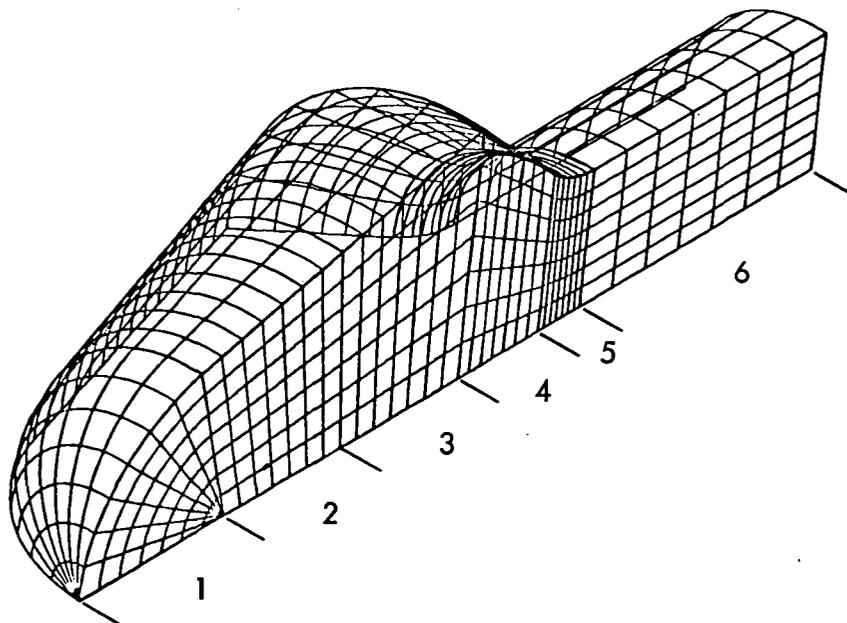


Figure 2. Degenerate Pentahedra and Tetrahedra Hyperpatches

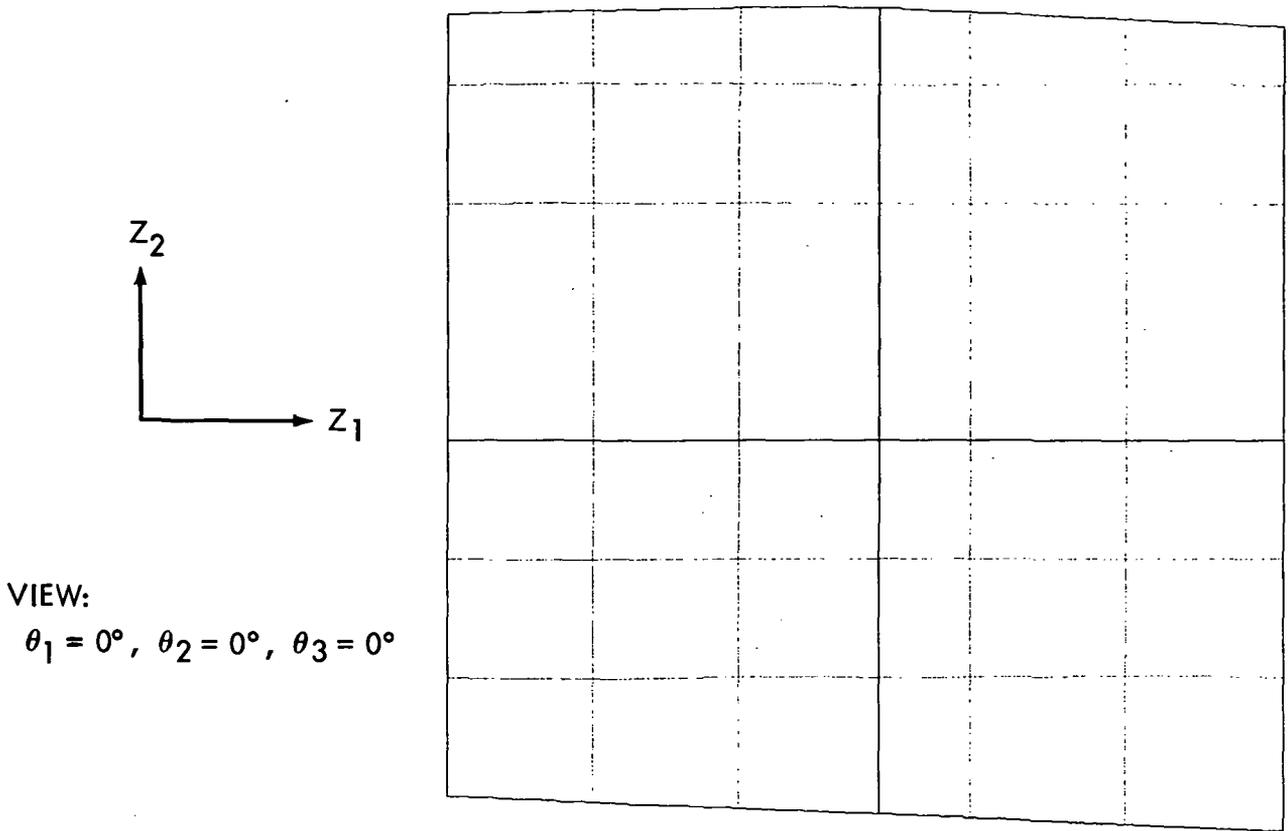


Figure 3. Compressor Fan Blade Planform

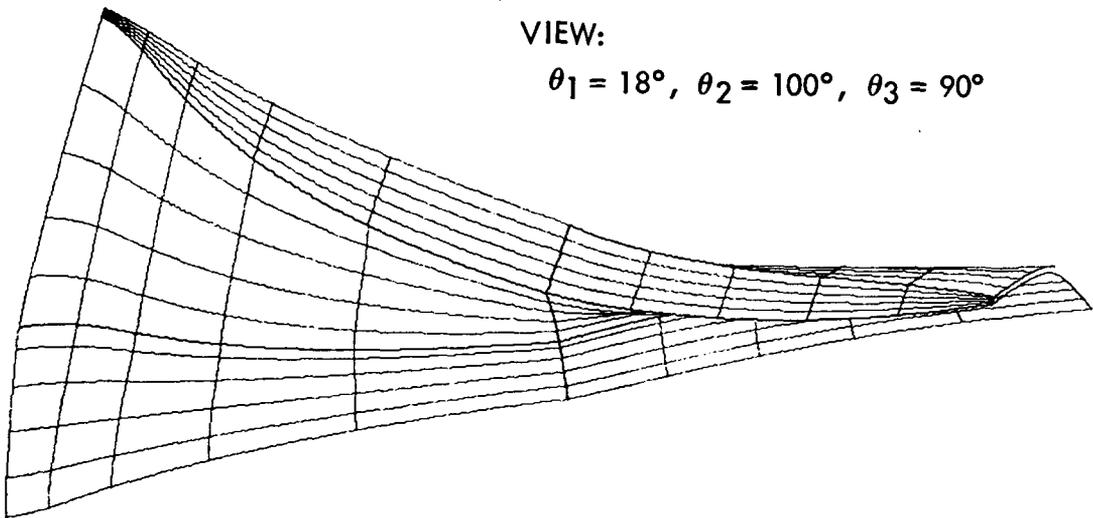


Figure 4. Four Hyperpatch Model of Compressor Fan Blade

Input Data Card: PFEG - Element property generator

Description: Generates property cards for elements obtained by subdivision.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PFEG	LPH-ID	DLPH1	DLPH2	DLPH3		DLPH-7	+P
PFEG	P5	12	5						+P1
+P	PID1	NP1	PID2	NP2	PIDM	NPM	
+P1	14	ALL							

<u>Field</u>	<u>Contents</u>
LPH-ID	Line, patch or hyperpatch identification number that defines geometric region.
DLPH-I	Data line, patch or hyperpatch that defines property I over the region.
PID-I	Property card identification number. This card must agree with the element type defined by the CFEG card.
NP-I	Number of sequential elements that have properties defined by PID-I.

Figure 5. Element Property Generator Card

Input Data Card: CFEG - Element Connectivity Generator

Description: Generates connectivity data for elements obtained by subdivision of a geometric shape.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CFEG	LPH-ID	DG1	DG2	DG3	DG4	N1	N2	N3	+C
CFEG	H10	3	6	8	4	5	5	3	+C1
+C	ETYPE	DG5	DG6	DG7	DG8	E1	G1		
+C1	CHEX1	13	12	15	10	100	S10		

Field

Contents

- LPH-ID** Line, patch or hyperpatch identification number that defines geometry.
- DG-ID** Data generator grid point identification numbers for corners of geometric shape.
- N1** Number of subdivisions in each parametric direction.
- ETYPE** NASTRAN element type to be generated.
- E1** Identification number for first element in the set generated by this card. If preceded by an S, the number points to a SET for user definition of all element identification numbers.
- G1** Identification number for first grid point in the set generated by this card. If preceded by an S, the number points to a SET for user definition of all grid point identification numbers. If blank, the first grid point number will be determined from all CFEG cards.

Figure 6. Element Connectivity Generator Card

Input Data Card: PMATL Laminate Property Generator

Description: Generates composite laminate properties from ply properties and stacking sequence for constant ply properties.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PMATL	ML-ID	MTYPE	MP-ID	T	NSYM	θ_1	θ_2	θ_3	+P1
PMATL	8	MAT2	3	.007	-2	0	45	-45	+1
+P1	θ_4	θ_M					
+1	0								

Field

Contents

- ML-ID Laminate material identification number.
- MTYPE NASTRAN material property card type to be generated.
- MP-ID Ply material identification number.
- T Ply-thickness
- NSYM Number of ply symmetry or asymmetry groups; ($\theta_1, \theta_2, \dots, \theta_M, \theta_N$) counting the primary group as one. If the input number is negative, each group reverses the previous groups sequence.
- $\theta_1, 2, \dots, M$ Ply orientation in the primary group. Each θ is the angle from the laminate material frame to the ply material frame for that ply.

Figure 7. Laminate Property Generator Card

Input Data Card: PMATGL General Laminate Property Generator

Description: Generates composite laminate properties from ply properties and stacking sequence for general plies.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PMATGL	ML-ID	MTYPE	NSYM	MP1	T1	θ_1	MP2	T2	+P1
PMATGL	5	MAT3	3	2	.007	0	3	.914	+1
+P1	θ_2	MP3	T3	θ_3	MP4	T4	θ_4	..	+P2
+1	60	3	.014	-60	2	.007	0		

<u>Field</u>	<u>Contents</u>
ML-ID	Laminate material identification number.
MTYPE	NASTRAN material property card type to be generated.
NSYM	Number of ply symmetry or asymmetry groups; ($\theta_1, \theta_2, \dots, \theta_M$) _N counting the primary group as one. If the input number is negative, each group reverses the previous groups sequence.
MP1, 2, . . . , M	Ply material identification numbers.
T1, 2, . . . , M	Ply-thickness data.
$\theta_1, 2, \dots, M$	Ply orientation in the primary group. Each θ is the angle from the laminate material frame to the ply material frame for that ply.

Figure 8. General Laminate Property Generator Card

Input Data Card: PMATX Composite property generator

Description: Generates effective composite properties from phase material properties and general microstructural parameters.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PMATX	MX-ID	MTYPE	MODEL		X	X	X	X	+P1
PMATX	7	MAT 3							+1
+P1	MID 1	P1	P2	P3	P4	θ1	θ2	θ3	+P2
+1	15	0.6				0	90	0	+2

Field

Contents

- MX-ID Composite material identification number.
- MTYPE NASTRAN material property card type to be generated.
- MODEL Composite material property model to be used to compute effective properties. The default model is a rule-of-mixtures.
- M1, 2, · · M Material property identification number for phase material I.
- P1, 2, 3, 4 Scalar parameters required by selected property model such as volume fraction, porosity, etc., with up to four allowed per phase.
- θ1, 2, 3 Up to three angles required to rotate composite material frame to the material frame for phase I.

Figure 9. Composite Property Generator Card

DEVELOPMENT OF AN AUTOMATED MULTI-STAGE
MODAL SYNTHESIS SYSTEM FOR NASTRAN

D. N. Herting and R. L. Hoesly
Universal Analytics, Inc.
Playa Del Rey, California

SUMMARY

A mode synthesis development to be scheduled in the NASTRAN multi-level substructuring system for general dynamics applications is described. The method combines the better features of several state of the art mode synthesis techniques, yet is general enough to provide for any arbitrary combination of boundary degrees of freedom and normal mode boundary conditions. Normal modes or complex eigenvectors may be used in the definition of a structure component which may be combined with other components of any type. Combination structures fabricated from component modes may be processed as normal substructures, including further multi-stage mode synthesis reductions. Included are discussions of the user control of the system and advantages in actual application.

INTRODUCTION

Component mode synthesis shares with substructure analysis the concept of dividing a structure into separate components, reducing the order of the component matrices, and combining the resultant matrices through the displacement equalities at points in common. Because of these similar operations, the NASTRAN automated multi-stage substructuring system (AMSS), which was developed by Universal Analytics, Inc. (UAI), provides an ideal framework for a mode synthesis system in NASTRAN.

The complex computational tasks of identifying characteristics of each component, joining these components to form a final full model, and managing the associated data has been completely automated using the NASTRAN substructuring capability. The design of the NASTRAN AMSS (Automated Multi-Stage Substructuring) system basically provides an open-ended data base management capability in which substructure data is stored and retrieved by reference to arbitrary, user-defined BCD substructure names. Current provisions allow for multi-stage combining of substructures, performing Guyan reductions, solving the resulting system, and recovering the solution data for the original basic substructure components. Also included are many available data maintenance options such as the ability to perform "dry" runs, delete erroneous data, and transmit data to permanent storage or to other computers.

In the UAI mode synthesis procedure, all existing options for combining, reducing, and solving substructures are also available for the new type of component which uses mode displacements as degrees of freedom. In effect, modal synthesis is an alternate to the AMSS REDUCE operation in which the order of structural matrices is reduced before connecting to other structures. It was obvious to us that many advantages would automatically accrue if the existing COMBINE code could be used for component mode synthesis. However, this could only occur if the existing matrix row and column indexing method in the COMBINE module were used. The present NASTRAN AMSS system uses the actual grid point displacement components as degrees of freedom at the connected grid points. The unified component mode formulation is consistent with this requirement.

Because actual grid point displacements on the boundary are retained as degrees of freedom, the structures defined with modes may be connected to any other type of substructure. The resulting combination structure also may be processed as a normal substructure, using further REDUCE, COMBINE, and/or higher stages of mode synthesis operations. A major advantage of this method is that, in the solution phase of the analysis, boundary conditions and loads may be applied directly to the retained grid point degrees of freedom.

The complex mode synthesis procedure is very similar to the real normal modes process. However, the matrices may be complex and non-symmetric in general. The resulting reduced substructure may be combined with other substructures of the same type, with conventional substructures, or with real modal reduced substructures. Naturally, because of existing NASTRAN restrictions, the result may only be processed with operations allowing complex arithmetic such as the combine operation and frequency response analysis.

In the following development we will summarize the extensive number of existing major variations in the component mode synthesis method, followed by a brief introduction to the theory and rationale behind the new method. This is followed by an example of the input required by NASTRAN for controlling the operation.

BACKGROUND

The use of structural modes as generalized degrees of freedom in dynamic models originated in the analog computer field where structures are combined with aeroelastic and control system models. The first applications to digital computers were simple extensions of the analog techniques. This so-called classical approach proved both highly restrictive and limited in accuracy. Many different approaches have been developed in recent years having increased accuracy and more generality in solving large-order structure dynamics problems.

Although the current methods used in component mode synthesis vary considerably in both approach and application, they may be grouped into two distinct categories. The first category contains all of the methods using a Rayleigh-Ritz approach in which the component degrees of freedom represent the deflections of normal modes and static deflection shapes. The second category contains methods in which the component degrees of freedom are the actual

boundary point displacements and the modal coordinates. Here, the classical method has been improved by adding flexibility coefficients to the matrices to account for the effects of a truncated set of modes. The two basis categories and their variations are described below followed by a discussion of their relevance to NASTRAN and the automated substructuring system.

The first Rayleigh-Ritz component mode synthesis application to digital computers was devised by Hurty (Ref. 3) in which "constraint" point modes and "attach" point modes, obtained by applying unit static loads, were introduced to represent the static deformation shapes of the structure. Bamford (Ref. 5) simplified the method by using unit displacements to define the static modes. Numerous variations followed (Refs. 6 - 8) in which the displacements at both the interior and the boundary grid points were defined using combinations of different shapes. Unfortunately, the process used by these methods to connect the component structures together becomes unwieldy because each interconnected degree of freedom generates a constraint equation coupling all of the various degrees of freedom. These constraint equations generate matrices of order equal to all of the degrees of freedom in the combination structure and require several operations. Other difficulties arise in the solution of the combined structure because the original grid point displacements are no longer available to apply loads or boundary constraints. These conditions can be specified only on the basic substructures.

A different approach was developed from the classical modal formulations used in analog computer systems. MacNeal (Ref. 9) published a method in which the "residual flexibility" of the structure was calculated to correct for the missing static effects which occur when a truncated set of normal modes is used. This method has a distinct advantage in that the actual displacements of the boundary points are retained as degrees of freedom, thereby simplifying the combination process. Rubin (Ref. 10) extended the method to include the deflections occurring from rigid body accelerations. Rubin's method provides a superior method for calculating the resultant displacements of the interior points. However, the method is severely restricted in that only unconstrained (free-free) normal modes may be used, and the formulation is quite cumbersome, requiring several matrix inversions to obtain solutions.

An interesting variation of the Rayleigh-Ritz approach was developed by Craig and Bampton (Ref. 11) in which the actual boundary grid point displacements were retained as degrees of freedom, sharing the simplified connection of component structures of References 9 and 10. Unfortunately, all of the boundary degrees of freedom must be constrained when obtaining the normal modes. These types of modes are generally of little practical use to the analyst who may wish to verify the component modes with test results.

Hasselmann and Kaplan (Ref. 12) extended the modal synthesis method to include complex eigenvalues of structures with damping or added non-conservative effects. This process contained two stages. In the first stage, the real normal modes were used with a Bamford method to simply reduce and convert components. In the second stage, the Craig/Bampton method was extended to use the complex eigenvalues and eigenvectors in each component and to combine the components. For solution of the resulting equations, the complex eigenvalues of the combin-

ation structure are obtained to form a first-order differential equation. Careful analysis of the Hasselman paper will point out an omission of the effects of so-called left-hand eigenvalues which occur with nonsymmetric matrices. The transformations indicated may be singular in some cases (NASTRAN Demonstration Problem 10-1 for example). Other problems associated with the method are that because velocities are used as degrees of freedom, substructures formulated from complex modes may not be attached to other types of components and the existing NASTRAN dynamics procedures may not be used. However, aside from these problems, the method shows that complex modal synthesis formulations are feasible.

For application to NASTRAN it is essential to extract normal modes or complex eigenvectors with an arbitrary user-chosen set of fixed degrees of freedom on the component structure. This would require that a general version of the Rayleigh-Ritz approach be used. However, present versions of this approach would require an algorithm for combining components together which is quite different than the existing COMBINE procedure in the NASTRAN automated substructuring system. Further effort would be required to provide the capability of changing fixed boundaries in the solution phase. The program would be required to generate a multi-point constraint in order to fix a single degree of freedom. The Craig/Bampton or the MacNeal/Rubin approaches appear very attractive from this standpoint but, as stated before, they are restricted to using only specific boundaries for normal modes.

Because of the problems discussed above, a considerable effort has been spent seeking a better method of modal synthesis formulation to operate efficiently within the NASTRAN environment. As a result, UAI has developed a unique and unified method which combines the generality of the Rayleigh-Ritz method, with the simplicity of direct connections, and the accuracy of the Rubin method. The UAI component mode synthesis approach provides several additional improvements over all existing methods, and perhaps of equal importance, conveniently interfaces with the NASTRAN automated substructuring system.

THEORETICAL FORMULATION

This section briefly outlines the development of the unified component mode synthesis method developed by UAI for both real and complex eigenvectors. The specific procedures for implementation into NASTRAN are described in a subsequent section.

The goal of this development was to generate a method compatible with the NASTRAN multi-stage substructuring system. In substructuring, a wide range of user options and conveniences exist, particularly in the methods of combining and reducing substructures. With a compatible system, the structures could be reduced to their modal coordinates, combined with other reduced or non-reduced substructures, and the result could be used for subsequent substructure processes.

The basic relationship between the displacement coordinate sets used in the Rayleigh-Ritz methods is:

$$\{u\} = [\bar{G}]\{p\} + [\phi]\{\xi\} \quad (1)$$

where $\{u\}$ is the vector of absolute displacements at the grid points of a structure

$\{p\}$ is the vector of relative static displacements corresponding to the generalized displacement static shapes

$[\bar{G}]$ is determined from the steady state properties of the substructure

$[\phi]$ is the matrix of normal mode shapes for the unconnected substructure

$\{\xi\}$ is the vector of modal coordinates

The unknown displacements $\{u\}$ satisfy the matrix equation

$$[M]\{\dot{u}\} + [B]\{\ddot{u}\} + [K]\{u\} = \{F\} \quad (2)$$

where $[M]$, $[B]$, and $[K]$ are the mass, damping, and stiffness matrices respectively for the substructure and $\{F\}$ is the vector of applied loads.

We will assume that the modes $[\phi]$ are obtained for an arbitrary set of boundaries. Points having fixed boundaries will simply have zero values in the matrix $[\phi]$. If the displacements are subdivided into interior (u_i) and boundary (u_b) degrees of freedom, Eq. (1) may be written in partitioned form:

$$\begin{Bmatrix} u_b \\ u_i \end{Bmatrix} = \begin{bmatrix} I & 0 & \phi_b \\ G & H_f & \phi_i \end{bmatrix} \begin{Bmatrix} p_b \\ p_o \\ \xi \end{Bmatrix} \quad (3)$$

where $\{p_b\}$ are the displacement coordinates for the static boundary point modes

$[G]$ is the transformation as used in the Guyan reduction process,

$$\text{i.e., } - [K_{ii}^{-1} K_{ib}]$$

$\{p_o\}$ are accelerations of a set of free-body motions

$[H_f]$ represents the inertia relief deflections of the interior due

to unit free body accelerations, i.e., $[K_{ii}^{-1}][M_{ii}D_i + M_{ib}D_b]$
 $(D_i$ and D_b are partitions of the freebody acceleration shape)

$\{\xi\}$ are the generalized displacements of the modal coordinates

Rather than combine the right-hand matrices into a single transformation (as in other methods), the displacements p_b will be eliminated. Solving the upper half of Eq. 3 for p_b and substituting into the lower half, we obtain the equation:

$$\{u_i\} = [G]\{u_b\} + [H_f]\{p_o\} + [\phi_i - G\phi_b]\{\xi\} \quad (4)$$

From Eq. 4, an alternate transformation matrix may be generated in the form:

$$\begin{Bmatrix} u_b \\ u_i \end{Bmatrix} = \begin{bmatrix} I & 0 & 0 \\ G & H_f & \phi_i - G\phi_b \end{bmatrix} \begin{Bmatrix} u_b \\ p_o \\ \xi \end{Bmatrix} \quad (5)$$

or

$$\{u_g\} = [H_{gh}]\{u_h\} \quad (6)$$

where

$$\{u_h\} = \begin{Bmatrix} u_b \\ p_o \\ \xi \end{Bmatrix} \quad (7)$$

The deflection shapes of the component structure due to unit values of the three types of coordinates are illustrated in Fig. 1. A much more rigorous derivation of Eq. 6 has been developed and will be published in the NASTRAN Theoretical Manual. The above development merely illustrates the approach.

Note that when the number of modes is zero and the inertia relief effects are ignored, the transformation matrix given in Eq. (6) is the same as Guyan reduction or matrix condensation transformation. When modes exist, they add information as to the dynamics relative to the static deformations. Free body modes and redundant constraint information are contained in the $[G]$ transformation.

Using the Galerkin principal the stiffness, mass, and loud damping matrices may be transformed using Eq. (6). The loads are transformed by the equation:

$$\{F_h\} = [H_{gh}]^T\{F_g\} \quad (8)$$

The mass matrix is transformed into a full matrix having the equation:

$$[M_{hh}] = [H_{gh}]^T [M] [H_{gh}] \quad (9)$$

The damping matrix would be transformed in a similar manner. However, the stiffness matrix takes the form:

$$K_{hh} = \begin{bmatrix} \bar{K}_{bb} & 0 \\ 0 & K_{pp} \end{bmatrix} \quad (10)$$

where \bar{K}_{bb} is exactly the same as the stiffness obtained from a Guyan reduction. Note that the zero off-diagonal partitions cause static decoupling of the modal coordinates.

After a solution is obtained, the displacements in the original set including the interior points may be recovered using Eq. (6). However, Rubin (Ref. 10) has shown that a mode acceleration technique will result in improved accuracy and may be applied resulting in the following equations:

$$\{u_i\} = [G]\{u_b\} + [K_{ii}]^{-1}\{\bar{p}_i\} \quad (11)$$

where

$$\{\bar{p}_i\} = \{p_i\} - [M_{ii}]\{\ddot{u}_i\} - [B_{ii}]\{\dot{u}_i\} \quad (12)$$

The benefits and efficiencies of the above method are described following the next section which develops the complex mode synthesis variation of the method.

Features of the Method

In general, the formulation described above offers the following beneficial characteristics:

1. The primary difference between the above method and conventional Rayleigh-Ritz method is that the displacements of the boundary points, u_a , are explicitly contained in the system. Each substructure is reduced independently of the others. Any resulting substructure may be combined with other substructures using existing COMBINE logic. Furthermore, the combination may be further combined or reduced - either with modes or by static reduction.

2. The real frequencies, modal masses, and modal stiffnesses of the transformed mass and stiffness matrices (Eqs. (9) and (10)) will be identical to the untransformed system (Eq. (2)) if the constrained boundaries are the same. The undamped eigenvectors of the transformed system are:

$$[\phi_h] = \begin{bmatrix} \phi_b \\ \hline 0 \\ \hline I \end{bmatrix} \quad (13)$$

where each column of the matrix is a modal vector. The validity of the mode shape may be proven by substituting into Eq. (6). The modal stiffness and mass properties may be obtained by pre- and post-multiplying the stiffness and mass matrices in Eqs. (9) and (10) by the vectors in Eq. (13).

3. The dynamic properties of the interior points, u_i , are replaced by the properties of the inertia relief shapes and normal modes $[\phi]$. The modal generalized displacements, ξ , are coupled to the system through the mass matrix only. This allows the use of a very small set of modal coordinates. The modes will be excited only by accelerations of the boundary points and loads on the interior points.
4. The static properties of the transformed structure at the boundary points are maintained independently of the number of modes selected or the boundary conditions used for obtaining the modes. When the number of selected modes is zero, the method is identical to the Guyan reduction or matrix condensation method.
5. Note that no explicit requirement is imposed that the normal modes be orthogonal. The mode shapes ϕ may be obtained from other sources such as experiments or a separate analysis.
6. The eigenvectors may be obtained using any user-selected fixed or free temporary set of fixed boundaries. These constraints are automatically removed from the resultant system.
7. Redundant modes shapes will be automatically detected and removed from the system. An example is when the normal mode contains no inertia effects from the interior points. The interior displacements will then be a defined from statics as:

$$\{\phi_i\} = [G]\{\phi_b\} \quad (14)$$

and the corresponding column of the transformation $[H_{gh}]$ calculated in Eq. (6) will be zero. A test on each column of the matrix is performed to check for errors.

8. The method will produce results equivalent to other methods when the corresponding restrictions are applied. Eliminating the inertia relief effects and fixing only boundary points for the normal modes will simulate the Hurty (Ref. 3) results. In addition, fixing all of the boundary points will result in exactly the same matrices as Craig/Bampton (Ref. 11). Test cases have proven that the use of free-free modes will duplicate Rubin's results (Ref. 10).

In summary, the above method will generate a stiffness and mass matrix corresponding to both boundary grid points and the normal modes of substructure. In essence, the modal degrees of freedom replace the interior degrees of freedom, without effecting the static stiffness properties of the boundary points.

This procedure may be performed for any substructure having stiffness and mass properties. The reduced substructure may be connected to another substructure by existing methods directly through the boundary degrees of freedom. The modal degrees of freedom do not enter into the connection process.

Complex Mode Synthesis

Much like normal mode synthesis, the use of complex eigenvectors to replace structure or normal mode displacements may be used to reduce the order of the dynamic matrices. Complex eigenvectors will provide an improved accuracy/size factor for problems in which the damping or added nonstructure effects produce first-order effects in the fundamental structure motions. This method will not be as effective as the normal modes method in cases where small, uniform damping coefficient effects are present, nor will it be efficient when the nonstructure effects may be isolated to a small number of coordinates such as a control system.

Several differences in the characteristics of the basic matrix equations of motion must be considered for complex mode synthesis when using a general second-order system as presently available in NASTRAN. These are:

1. The matrices may be complex and/or nonsymmetric, requiring different computational operations. This will effect both computer core requirements and execution times.
2. The same complex eigenvectors may occur twice for two different eigenvalues. This occurs frequently when real viscous damping is used and the eigenvalues occur as pairs of roots.
3. When nonsymmetric matrices occur, such as in control system problems or when coriollis inertial terms are present, the left eigenvectors will be different from the conventional right-side vectors. These eigenvectors are defined by the transposed mass, damping, and stiffness matrices, and are important for transforming loads.

The basic approach used in real component mode synthesis is applied to the complex eigenvector synthesis. With this formulation, the reduced structure may be connected to other substructures at the boundary points, u_b , with the existing NASTRAN COMBINE operation. Complex nodes of the combined structures may then be computed with another complex eigenvalue analysis.

Alternate methods which produce first-order differential matrix equations, as described in Reference 12, require the use of velocities in the matrix equations. For combining substructures, these velocity terms must also be processed in the connection procedure which requires all connected substructures to be similarly developed. This restriction would handicap the structure analyst who wishes to perform a complex formulation on only one of his component substructures.

NASTRAN IMPLEMENTATION

The proposed capabilities are being implemented in Level 16 NASTRAN within the existing logical flow path of the automated multi-stage substructuring system (AMSS). The real and complex mode synthesis operations are controlled by the substructure command deck, and may be requested at any stage of the substructuring process. The transient and frequency response capabilities are implemented directly within the existing NASTRAN rigid formats. The substructuring system will provide the system matrices and loads. Existing NASTRAN modules will perform the major numerical processes.

Currently in NASTRAN a substructure is defined by the degrees of freedom representing displacements at selected grid points. Substructures may be connected together through the COMBINE step at "boundary points." The connections are obtained either manually, through input data, or automatically, by allowing the program to scan the geometry for coincident points on different substructures. Either basic substructures or combined structures may be reduced using the static matrix condensation procedure, REDUCE. The new modal reduction method, MREDUCE, will be an alternate to the REDUCE operation, with similar inputs and output, as shown in Figure 2. However, the degrees of freedom defining the matrices will include selected modal displacement coordinates.

The inputs required from the user for the MREDUCE operation are:

1. A list of the boundary degrees of freedom to be attached to other structures, constrained, or directly loaded in subsequent stages of formulation and analysis.
2. A list of degrees of freedom to be constrained or supported for the extraction of modes for the modal REDUCE operation on the substructure.
3. The method, range of frequencies, and number of eigenvalues to be retained in the model.

For structures with large damping effects, the modal reduce command is CREDUCE. User inputs are similar to those listed above for the MREDUCE command.

CONCLUSIONS

The method described above will be a powerful tool in the dynamics analysis of large order NASTRAN problems. Although the correct multilevel substructure system provides for static matrix condensation of substructure components, the number and location of the retained degrees of freedom must frequently be estimated, resulting in solutions of unknown accuracy. The mode synthesis method will relieve the analyst of the guesswork and allow tight control over the size of the matrices in a dynamic formulation by allowing the specification of a range of frequencies to be used in the formulation corresponding to the range of frequencies expected in the solution.

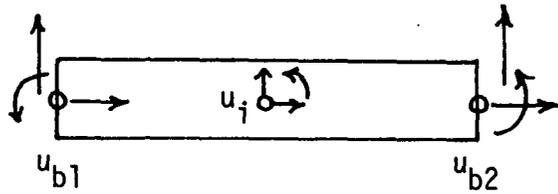
Although results from the new mode synthesis system in NASTRAM were not available at the time of the submission of the paper, simple tests have been performed as a stand-alone program. The results obtained have exactly duplicated other published results when the same mode fixed points and boundary points were used. This was expected since the theory shows that the shape functions used to represent the structure motion are identical except for the different method of combining the shapes.

In addition to duplicating the results of other methods, the new approval will allow mode synthesis procedures that have been impossible or difficult to control in the past. For instance, a matrix condensation may be performed on a combination structure formulated from component modes and boundary grial points. This would allow the elimination of extraneous boundary points which were required for connection to other structures. Another example would be the use of different types of fixities for extracting modes from the different components. Free-free, cantilever, and redundantly constrained modes from the different structure components may be connected together automatically. The user may select the type of boundary condition that best represents the motion of the component in the combined solution structure.

REFERENCES

1. Weidman, Deene J., "NASTRAN Status and Plans," NASTRAN Users' Experiences, NASA TM X-3275, 1975, pp. 1-7.
2. Field, E.I., Herting, D.N., Herendeen, D.L., and Hoesly, R.L., "The Automated Multi-Stage Substructuring System in NASTRAN," NASTRAN Users' Experiences, NASA TM X-3275, 1975, pp. 571-592.
3. Hurty, W.C., "Dynamic Analysis of Structural Systems Using Component Modes," AIAA Journal, Vol. 3, No. 4, April 1965, pp. 678-685.

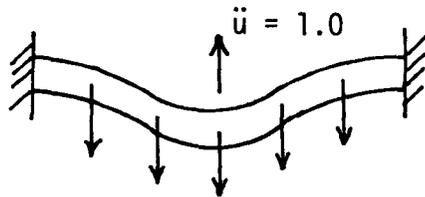
4. Bamford, R.M., "A Modal Combination Program for Dynamic Analysis of Structures," NASA TM 33-290, Jet Propulsion Laboratory, 1966.
5. Bamford, R.M., Wada, B.K., et al., "Dynamic Analysis of Large Structural Systems," Synthesis of Vibrating Systems, Nov. 1971, ASME booklet, pp. 57-71.
6. Bajan, R.L. and Feng, C.C., "Free Vibration Analysis by the Modal Substitution Method," American Astronautics Society Symposium, Paper No. 68-8-1, July 1968.
7. Benfield, W.A. and Hruda, F.R., "Vibration Analysis of Structures by Component Substitution," AIAA Journal, Vol. 9, July 1971, pp. 1255-1261.
8. Hintz, R.M., "Analytical Methods in Component Modal Synthesis," AIAA Journal, Vol. 13, Aug. 1975, pp. 1007-1016.
9. MacNeal, R.H., "A Hybrid Method of Component Mode Synthesis," Computers and Structures, Vol. 1, December 1971, pp. 581-601.
10. Rubin, S., "Improved Component-Mode Representation for Structural Dynamic Analysis," AIAA Journal, Vol. 12, August 1975, pp. 995-1006.
11. Craig, R.R. and Bampton, M.D.D., "Coupling of Substructures for Dynamic Analysis," AIAA Journal, Vol. 6, No. 7, July 1968, pp. 1313-1319.
12. Hasselman, T.K., and Kaplan, A., "Dynamic Analysis of Large Systems by Complex Mode Synthesis," J. Dynamic Systems, Measurement, and Control, September 1974, pp. 327-333.
13. The NASTRAN Theoretical Manual, MacNeal, R.J., ed., NASA SP-221(01), Chapter 14, "Representation of Part of a Structure by its Vibration Modes," April 1972.
14. NASTRAN Demonstration Problem Manual, NASA SP-224, "Rigid Format No. 10, Complex Eigenvalue Analysis - Modal Formulation," September 1970, pp. 10.1-1.
15. Giescke, R.K., "Analysis of Nonlinear Structures via Mode Synthesis," NASTRAN Users' Experiences, NASA TM X-3278, 1975, pp. 341-360.



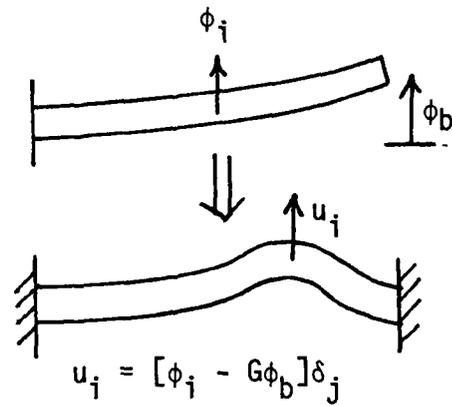
a) Component Substructure



b) Static Deflections, u_b



c) Inertia Relief, δ_o



d) Modified Normal Modes

Figure 1. Generalized coordinate shape functions for component mode synthesis.

Typical Command Data Sequence	Definition
<pre> MREDUCE BIGONE NAME = SMALL1 BOUNDARY = 101 FIXED = 55 METHOD = 10 RANGE = 0.0, 500.0 NMAX = 20 MRECOVER BIGONE PRINT BIGONE (Repeat for other substructures) COMBINE SMALL1, SMALL2, FLAP Etc. : </pre>	<pre> (Modal reduce on substructure 'BIGONE') (Name of result is 'SMALL1') (Boundary degrees of freedom defined on BDYS Bulk Data) (Degrees of freedom fixed during mode extraction defined BDYS Bulk Data) (Eigenvalue method defined by EIGR Bulk Data card) (Limits range of frequencies used) (Use lowest 20 modes) (Recover normal modes for printout) (Normal substructure operations) </pre>

Figure 2. Substructure command data example for modal synthesis.

ADDITION OF RIGID ELEMENTS TO NASTRAN

P. R. Pamidi
Computer Sciences Corporation
Hampton, Virginia

and

J. D. Cronkhite
Bell Helicopter Textron
Fort Worth, Texas

SUMMARY

Four rigid elements, namely, a rigid rod element (CRIGDR) and three rigid body elements (CRIGD1, CRIGD2, and CRIGD3), have recently been added to NASTRAN and will be available in the next public release of the program (Level 17.0). In this paper, the theoretical formulation, the bulk data information and the programming details pertaining to these elements are presented. Also, the use of these elements in practical and realistic problems is illustrated by employing them in the solution of two helicopter structural analysis problems.

INTRODUCTION

The multipoint constraint feature available in NASTRAN provides the capability for specifying linear relationships among various components of motion. It is very general in nature and is widely used in practice. The important requirement of this feature is that the user has to supply explicitly the coefficients of the constraint equations. This may not be a drawback in most cases, but it makes it particularly inconvenient and difficult to use this feature for representing rigid elements and rigid bodies since the computation of the required coefficients in many practical problems is often quite cumbersome.

In order to avoid this difficulty and inconvenience, four rigid elements, namely, a rigid rod element (CRIGDR) and three rigid body elements (CRIGD1, CRIGD2, and CRIGD3), have been added to NASTRAN and will be available in the next public release of the program (Level 17.0). The use of these elements results in the automatic generation of the required coefficients from the connection data. The user is thus relieved of the burden of specifying these coefficients via the multipoint constraint equations.

THEORETICAL FORMULATION

The CRIGDR Element

The CRIGDR element represents a connection between two grid points that is rigid in extension-compression.

Let A and B be two grid points connected by a CRIGDR element and let u_{A_1} , u_{A_2} , and u_{A_3} represent the translational components of motion (in the basic coordinate system) at these points respectively. Let l_1 , l_2 , and l_3 be the direction cosines (with respect to the basic coordinate system) of the line joining A to B. Then, since the distance between the points A and B remains unchanged, we have the condition

$$(u_{A_1} - u_{B_1})l_1 + (u_{A_2} - u_{B_2})l_2 + (u_{A_3} - u_{B_3})l_3 = 0 \quad , \quad (1)$$

or, in matrix form,

$$[l_1 \ l_2 \ l_3] \begin{Bmatrix} u_{A_1} \\ u_{A_2} \\ u_{A_3} \end{Bmatrix} = [l_1 \ l_2 \ l_3] \begin{Bmatrix} u_{B_1} \\ u_{B_2} \\ u_{B_3} \end{Bmatrix} \quad . \quad (2)$$

Let u'_{A_1} , u'_{A_2} , u'_{A_3} and u'_{B_1} , u'_{B_2} , u'_{B_3} be the translational components of motion at A and B in their respective local displacement coordinate systems. These are related to the motion in the basic coordinate system by the equations

$$\begin{Bmatrix} u_{A_1} \\ u_{A_2} \\ u_{A_3} \end{Bmatrix} = [T_A] \begin{Bmatrix} u'_{A_1} \\ u'_{A_2} \\ u'_{A_3} \end{Bmatrix} \quad , \quad (3)$$

and

$$\begin{Bmatrix} u_{B_1} \\ u_{B_2} \\ u_{B_3} \end{Bmatrix} = [T_B] \begin{Bmatrix} u'_{B_1} \\ u'_{B_2} \\ u'_{B_3} \end{Bmatrix} \quad , \quad (4)$$

where $[T_A]$ and $[T_B]$ are (3 x 3) transformation matrices from the respective local displacement coordinate systems to the basic coordinate system.

Substituting Equations (3) and (4) in Equation (2), we get

$$[\lambda_1 \ \lambda_2 \ \lambda_3][T_A] \begin{Bmatrix} u'_{A_1} \\ u'_{A_2} \\ u'_{A_3} \end{Bmatrix} = [\lambda_1 \ \lambda_2 \ \lambda_3][T_B] \begin{Bmatrix} u'_{B_1} \\ u'_{B_2} \\ u'_{B_3} \end{Bmatrix} \quad (5)$$

The above equation can be rewritten as

$$[\lambda_{A_1} \ \lambda_{A_2} \ \lambda_{A_3}] \begin{Bmatrix} u'_{A_1} \\ u'_{A_2} \\ u'_{A_3} \end{Bmatrix} = [\lambda_{B_1} \ \lambda_{B_2} \ \lambda_{B_3}] \begin{Bmatrix} u'_{B_1} \\ u'_{B_2} \\ u'_{B_3} \end{Bmatrix}, \quad (6)$$

where λ_{A_1} , λ_{A_2} , λ_{A_3} and λ_{B_1} , λ_{B_2} , λ_{B_3} represent the modified direction cosines and are given by

$$[\lambda_{A_1} \ \lambda_{A_2} \ \lambda_{A_3}] = [\lambda_1 \ \lambda_2 \ \lambda_3][T_A] \quad (7)$$

and

$$[\lambda_{B_1} \ \lambda_{B_2} \ \lambda_{B_3}] = [\lambda_1 \ \lambda_2 \ \lambda_3][T_B] \quad (8)$$

Equation (6) is the single equation of constraint that represents a rigid rod element connection between the grid points A and B. Note that only the three translational components of motion at each of the two points are involved in this equation. The rotations at the points are not involved. Any one of the six translational components may be specified as the dependent degree of freedom in a CRIGDR element. The other five translational components are considered as reference degrees of freedom. This is summarized in Table 1.

If Equation (6) is to be valid, it is necessary that the grid points A and B be not coincident. If they are, the direction cosines λ_1 , λ_2 , and λ_3 will be undefined. The program checks for this condition.

Equation (6) will also not be meaningful if the direction of motion defined by the dependent translational degree of freedom is perpendicular (or nearly perpendicular) to the rod element because, in that case, the corresponding modified direction cosine will be zero (or nearly zero). The program checks for this condition also.

The CRIGD1 and CRIGD2 Elements

The CRIGD1 and CRIGD2 elements are similar in that they both involve a single reference grid point and one or more dependent grid points. The CRIGD1 element is the simpler and defines a rigid element connection in which all six degrees of freedom of each of the dependent grid points are coupled to all six degrees of freedom of the reference grid point. The CRIGD2 element is more general and defines a rigid element connection in which selected degrees of freedom of the dependent grid points are coupled to all six degrees of freedom of the reference grid point.

Consider a dependent grid point A that is rigidly coupled by means of a CRIGD1 or CRIGD2 element to a reference grid point B. The motion $\{u_A\}$ at the point A is related to the motion $\{u_B\}$ at the point B by the equation

$$\begin{Bmatrix} u_{A1} \\ u_{A2} \\ u_{A3} \\ u_{A4} \\ u_{A5} \\ u_{A6} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & (z_B - z_A) & -(y_B - y_A) \\ 0 & 1 & 0 & -(z_B - z_A) & 0 & (x_B - x_A) \\ 0 & 0 & 1 & (y_B - y_A) & -(x_B - x_A) & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} u_{B1} \\ u_{B2} \\ u_{B3} \\ u_{B4} \\ u_{B5} \\ u_{B6} \end{Bmatrix}, \quad (9)$$

where the motions are in the basic coordinate system and x_A, y_A, z_A and x_B, y_B, z_B are the basic coordinates of the points A and B respectively.

Using relations similar to Equations (3) and (4), Equation (9) can be expressed in terms of the motion in the local displacement coordinate systems of A and B by

$$\begin{Bmatrix} u'_{A1} \\ u'_{A2} \\ u'_{A3} \\ u'_{A4} \\ u'_{A5} \\ u'_{A6} \end{Bmatrix} = \begin{bmatrix} T_A^T & 0 \\ 0 & T_A^T \end{bmatrix} \begin{bmatrix} I & \begin{matrix} 0 & \bar{z} & -\bar{y} \\ -\bar{z} & 0 & \bar{x} \\ \bar{y} & -\bar{x} & 0 \end{matrix} \\ 0 & I \end{bmatrix} \begin{bmatrix} T_B & 0 \\ 0 & T_B \end{bmatrix} \begin{Bmatrix} u'_{B1} \\ u'_{B2} \\ u'_{B3} \\ u'_{B4} \\ u'_{B5} \\ u'_{B6} \end{Bmatrix}, \quad (10)$$

where T_A and T_B are (3×3) transformation matrices from the local displacement coordinate systems to the basic coordinate system and $\bar{x} = x_B - x_A$, $\bar{y} = y_B - y_A$ and $\bar{z} = z_B - z_A$. I is a (3×3) unit matrix.

Equation (10) can be written in compact form as

$$\{u'_A\} = [G]_{AB} \{u'_B\}, \quad (11)$$

where $[G]_{AB}$ is a (6×6) matrix. Each row of this $[G]_{AB}$ matrix corresponds to a dependent degree of freedom of grid point A, and each column corresponds to a reference degree of freedom of grid point B. Each element of this matrix represents a coefficient that corresponds to the coupling of a particular dependent degree of freedom of grid point A with a particular reference degree of freedom of grid point B.

Equation (11) defines a set of six linear equations of constraint that mathematically represent the rigid coupling of dependent grid point A to reference grid point B. In the case of a CRIGD1 element, six equations of constraint are generated for each of the specified dependent grid points. In the case of a CRIGD2 element, the equations generated correspond to those rows of $[G]_{AB}$ that represent the specified dependent degrees of freedom of grid point A.

Let m be the total number of dependent degrees of freedom specified on a CRIGD1 or CRIGD2 element. Then, by combining Equations (11) for all of the dependent grid points, we get m linear equations of constraint represented in matrix form by

$$\{u'\} = [G]_B \{u'_B\}, \quad (12)$$

where $\{u'\}$ is an $(m \times 1)$ vector of dependent degrees of freedom (in global coordinate system) and $[G]_B$ is an $(m \times 6)$ matrix that represents the rigid coupling

of the m dependent degrees of freedom to the six degrees of freedom of reference grid point B. Note that, in the case of a CRIGD1 element, m is equal to six times the number of dependent grid points specified for the element. The above results are summarized in Table 1.

The CRIGD3 Element

The CRIGD3 element is the most general rigid element and defines a rigid connection in which selected degrees of freedom of the dependent grid points are coupled to six selected reference degrees of freedom. The six reference degrees of freedom can be selected at one or more (up to six) reference grid points, but they should together be capable of fully describing rigid body motion. In other words, the six reference degrees of freedom should be so selected that they together represent six independent components of motion. The program checks for this condition since otherwise it leads to the inversion of a singular matrix.

Let B be one of the (up to six) reference grid points in a CRIGD3 element and let m be the total number of dependent degrees of freedom specified on the element. Then, just as in the case of a CRIGD1 or CRIGD2 element, the m equations of constraint can be expressed in terms of the motion of grid point B by the matrix equation

$$\{u'\} = [G]_B \{u'_B\} \quad , \quad (13)$$

which is a re-statement of Equation (12). Note, however, in this case that the six degrees of freedom of grid point B will not, in general, all be the required six reference degrees of freedom. Hence, Equation (13) does not give the required constraint equations.

Let u'_{R_1} , u'_{R_2} , u'_{R_3} , u'_{R_4} , u'_{R_5} , and u'_{R_6} be the six specified reference degrees of freedom (at least some of which will be the degrees of freedom of grid point B) in the global coordinate system. Then, these six degrees of freedom are related to the motion of grid point B by the matrix equation

$$\{u'_R\} = [G]_{RB} \{u'_B\} \quad , \quad (14)$$

which is similar to Equation (13) and where $[G]_{RB}$ is a (6 x 6) matrix.

Equation (14) can be re-written as

$$\{u'_B\} = [G]_{RB}^{-1} \{u'_R\} \quad . \quad (15)$$

Note that $[G]_{RB}^{-1}$ will not exist if the six specified reference degrees of freedom do not together define six independent components of motion. The program checks for this condition.

Substituting Equation (15) in Equation (13), we get

$$\{u'\} = [G]_B [G]_{RB}^{-1} \{u'_R\} \quad . \quad (16)$$

The above matrix equation gives the required equations of constraint for a CRIGD3 element. This is summarized in Table 1.

USE OF RIGID ELEMENTS

The bulk data card descriptions of all the four rigid elements discussed above are given in the Appendix. Note that, on all the rigid element connection cards, the user specifies the degrees of freedom that belong to the dependent set. This specification is implicit on the CRIGD1 card and explicit on the others. It is also important to note that a dependent degree of freedom appearing in a rigid element may not appear as dependent in any other rigid element or on a MPC card nor may it be constrained in any other manner.

When using many rigid elements and multipoint constraints, the user will often find it useful to turn on DIAG's 21 and 22 in the Executive Control Deck (Reference 1) so that he can check the various NASTRAN sets to which all the degrees of freedom in the model belong.

PROGRAMMING DETAILS

In the program, the constraint equations for the rigid elements are generated in a new routine called CRIGGP which is in module GP4. This routine computes the required constraint equations for all rigid elements in a model by means of Equation (6) (for all CRIGDR elements), Equation (12) (for all CRIGD1 and CRIGD2 Elements), and Equation (16) (for all CRIGD3 elements). Module GP4 then combines these constraint equations for all rigid elements with the multipoint constraint equations supplied by the user to obtain the resultant constraint equations for the model as a whole. Once the rigid elements and the multipoint constraint data are processed and the resultant constraint equations obtained by module GP4, no distinction is made subsequently between those constraint equations that are due to rigid elements and those that are due to multipoint constraint data.

EXAMPLES

In order to fully test the capability and usefulness of the Level 17.0 rigid elements discussed above in practical and realistic problems, they were employed in the solution of two problems dealing with the structural analysis of the AH-1G helicopter. Example 1 is the analysis of the AH-1G main rotor pylon on rigid body fuselage which is shown in Figure 1; the model used for the rigid body fuselage is shown in Figure 2. Example 2 is the analysis of the AH-1G built-up frame which is shown in Figure 3. Details of these two models are discussed in Reference 2.

Normal mode analyses of the above two models were performed on a pre-Level 17.0 version of NASTRAN on the IBM 370/168 computer at Bell Helicopter Textron. The total number of degrees of freedom considered in the analysis set or a-set (Reference 3) was 41 for the model in Example 1 and 236 for the model in Example 2. The complete data for the two examples can be obtained from the authors.

The above analyses were also performed on the MacNeal-Schwendler Corporation (MSC) version of NASTRAN by suitably replacing the Level 17.0 rigid elements by the RR0D, RBAR, RBE1, and RBE2 rigid elements available in MSC/NASTRAN (see Reference 4 for a discussion of these elements).

The results of the analyses for significant modes are presented in Tables 2 and 3. As can be seen, there is excellent agreement between the results obtained by using the Level 17.0 rigid elements and the MSC rigid elements. The slight differences in some of the answers are apparently due to the different precision and manner in which the two versions of the program handle some of their internal computations.

SUMMARY AND CONCLUSIONS

Four rigid elements, namely, a rigid rod element (CRIGDR) and three rigid body elements (CRIGD1, CRIGD2, and CRIGD3), have recently been added to NASTRAN and will be available in the next public release of the program (Level 17.0). In this paper, the theoretical formulation, the bulk data information and the programming details pertaining to these elements have been presented. Also, the use of these elements in practical and realistic problems has been illustrated by employing them in the solution of two helicopter structural analysis problems.

The rigid element capability added to NASTRAN represents yet another enhancement in its usefulness. This feature is sure to find wide application in a variety of practical structural analysis problems.

REFERENCES

1. The NASTRAN User's Manual, (Level 16.0), NASA SP-222(03), March 1976.
2. Cronkhite, J. D., "Development, Documentation and Correlation of a NASTRAN Vibration Model of the AH-1G Helicopter Airframe", NASTRAN: Users' Experiences, NASA TM X-3428, October 1976, pp. 273-294.
3. The NASTRAN Theoretical Manual, (Level 16.0), NASA SP-221(03), March 1976.
4. MSC/NASTRAN Application Manual, The MacNeal-Schwendler Corporation, Los Angeles, California, 1974, Section 2.10.

APPENDIX

Input Data Card CRIGDR Rigid Rod Element Connection

Description: Defines a rod element that is rigid in extension-compression.

Format and Example:

	1	2	3	4	5	6	7	8	9	10
CRIGDR	EID	G	G1	C1	EID	G	G1	C1		
CRIGDR	104	5	9	3	302	12	4	2		

Field

Contents

EID	Element identification number (Integer > 0)
G	Identification number of the reference grid point (Integer > 0)
G1	Identification number of the dependent grid point (Integer > 0; G1 ≠ G)
C1	Dependent translational degree of freedom of grid point G1 (1 ≤ Integer ≤ 3)

- Remarks:
1. Element identification numbers must be unique with respect to all other element identification numbers.
 2. Only one reference grid point and only one dependent grid point are allowed per element. The two points may not be coincident.
 3. The direction represented by the dependent translational degree of freedom of the dependent grid point may not be perpendicular or nearly perpendicular to the element.
 4. One or two CRIGDR elements may be defined on a single card.
 5. Dependent degrees of freedom defined in CRIGDR elements may not appear on OMIT, OMIT1, SPC, SPC1 or SUPORT cards nor may they be redundantly implied on ASET or ASET1 cards. They also may not appear as dependent degrees of freedom in CRIGD1, CRIGD2, or CRIGD3 elements or on MPC cards.
 6. Rigid elements are not allowed in heat transfer analysis.
 7. For a discussion of rigid elements, see Section 3.5.6 of the Theoretical Manual.

Input Data Card CRIGD1 Rigid Element Connection

Description: Defines a rigid element in which all six degrees of freedom of each of the dependent grid points are coupled to all six degrees of freedom of the reference grid point.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CRIGD1	EID	IG	G1	G2	G3	G4	G5	G6	abc
CRIGD1	101	15	18	43	9	26	35	41	123
+bc	G7	G8	etc.						
+23	8	63							

<u>Field</u>	<u>Contents</u>
EID	Element identification number (Integer > 0)
IG	Identification number of the reference grid point (Integer > 0)
G1, G2, etc.	Identification numbers of the dependent grid points (Integer > 0)

- Remarks:**
1. Element identification numbers must be unique with respect to all other element identification numbers.
 2. Only one reference grid point is allowed per element. It must appear before any of the dependent grid points.
 3. Any number of dependent grid points may be specified.
 4. Dependent degrees of freedom defined (implicitly) in a CRIGD1 element may not appear on ØMIT, ØMIT1, SPC, SPC1 or SUPØRT cards nor may they be redundantly implied on ASET or ASET1 cards. They also may not appear as dependent degrees of freedom in CRIGD2, CRIGD3, or CRIGDR elements or on MPC cards.
 5. Rigid elements are not allowed in heat transfer analysis.
 6. For a discussion of rigid elements, see Section 3.5.6 of the Theoretical Manual.

Input Data Card

CRIGD2

Rigid Element Connection

Description: Defines a rigid element in which selected degrees of freedom of the dependent grid points are coupled to all six degrees of freedom of the reference grid point.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CRIGD2	EID	IG	G1	C1	G2	C2	G3	C3	abc
CRIGD2	102	20	9	12	45	123	53	135	123
+bc	G4	C4	etc.						
+23	27	456							

FieldContents

EID	Element identification number (Integer > 0)
IG	Identification number of the reference grid point (Integer > 0)
Gi	Identification numbers of the dependent grid points (Integer > 0)
Ci	List of selected degrees of freedom associated with the preceding dependent grid point (any of the digits 1-6 with no imbedded blanks)

- Remarks:
1. Element identification numbers must be unique with respect to all other element identification numbers.
 2. Only one reference grid point is allowed per element. It must appear before the dependent grid point data.
 3. Any number of dependent grid points may be specified.
 4. Dependent degrees of freedom defined in a CRIGD2 element may not appear on OMIT, OMIT1, SPC, SPC1 or SUPORT cards nor may they be redundantly implied on ASET or ASET1 cards. They also may not appear as dependent degrees of freedom in CRIGD1, CRIGD3, or CRIGDR elements or on MPC cards.
 5. Rigid elements are not allowed in heat transfer analysis.
 6. For a discussion of rigid elements, see Section 3.5.6 of the Theoretical Manual.

CRIGD3 (Cont.)

7. Rigid elements are not allowed in heat transfer analysis.
8. For a discussion of rigid elements, see Section 3.5.6 of the Theoretical Manual.

Table 1. Summary of Constraint Equations Generated for Rigid Elements

Name of element	Number of constraint equations generated (per element)	Number of ref. grid points (per element)	Number of ref. degrees of freedom involved (per element)	Constraint equations obtained from
CRIGDR	1	1	5	Equation (6)
CRIGD1	6 x (Number of dependent grid points)	1	6	Equation (12)
CRIGD2	Number of dependent degrees of freedom	1	6	Equation (12)
CRIGD3	Number of dependent degrees of freedom	1 to 6	6	Equation (16)

Table 2. Results for AH-1G Main Rotor Pylon on Rigid Body Fuselage (Example 1)

Mode No.	Natural frequencies (Hz)		Generalized masses (lb-sec. ² /in.)*	
	Using Level 17.0 rigid elements	Using MSC rigid elements	Using Level 17.0 rigid elements	Using MSC rigid elements
1	0.0	0.0	2.308850E+01	2.308852E+01
2	0.0	0.0	2.308850E+01	2.308852E+01
3	0.0	0.0	2.308852E+01	2.308852E+01
4	0.0	0.0	4.745212E+00	4.745212E+00
5	0.0	0.0	2.199127E+01	2.199127E+01
6	0.0	0.0	3.051502E+03	3.051502E+03
7	2.987344E+00	2.987597E+00	3.058784E+00	3.058784E+00
8	3.372945E+00	3.372946E+00	6.502025E+00	6.502025E+00
9	2.447569E+01	2.447569E+01	8.486220E-01	8.486219E-01
10	2.682217E+01	2.682217E+01	8.414585E-01	8.414580E-01
11	6.154906E+01	6.154906E+01	5.886275E-01	5.886274E-01
12	7.034309E+01	7.034311E+01	4.855811E-01	4.855810E-01
13	1.133579E+02	1.133579E+02	3.867739E-01	3.867738E-01
14	1.174531E+02	1.174531E+02	3.940392E-01	3.940392E-01
15	1.646036E+02	1.646037E+02	1.257550E+00	1.257550E+00

*The numbers in the generalized mass columns can be converted to the SI units (Kg) by multiplying by the factor 1.751268E+02.

Table 3. Results for AH-1G Built-Up Airframe (Example 2)

Mode No.	Natural frequencies (Hz)		Generalized masses (lb-sec. ² /in.)*	
	Using Level 17.0 rigid elements	Using MSC rigid elements	Using Level 17.0 rigid elements	Using MSC rigid elements
1	0.0	0.0	1.191102E+00	1.191094E+00
2	0.0	0.0	2.016126E+00	2.016142E+00
3	0.0	0.0	5.916871E+00	5.916818E+00
4	0.0	0.0	6.565869E+00	6.565207E+00
5	0.0	0.0	8.682934E+00	8.682921E+00
6	0.0	0.0	9.696571E+00	9.696051E+00
7	2.986722E+00	2.987587E+00	3.993563E+00	3.993563E+00
8	3.852448E+00	3.852509E+00	3.155181E+00	3.155213E+00
9	7.110725E+00	7.113102E+00	4.354592E-01	4.354487E-01
10	7.957870E+00	7.966046E+00	6.122239E-01	6.121598E-01
11	1.457138E+01	1.457089E+01	2.720798E-01	2.720449E-01
12	1.599263E+01	1.599451E+01	1.332268E+00	1.305253E+00
13	1.719148E+01	1.719003E+01	8.436403E-01	8.467465E-01
14	1.777260E+01	1.778244E+01	5.767109E-01	5.751327E-01
15	1.916400E+01	1.917462E+01	5.818658E-01	5.893239E-01
16	1.983231E+01	1.983275E+01	3.220435E-01	3.220390E-01
17	2.160742E+01	2.161783E+01	1.155260E+00	1.159738E+00
18	2.342937E+01	2.342992E+01	2.507004E-01	2.507226E-01
19	2.512769E+01	2.513480E+01	6.590332E-01	6.238537E-01
20	2.555725E+01	2.556792E+01	3.233123E-01	3.294395E-01
21	2.650203E+01	2.651083E+01	1.169636E+00	1.198108E+00
22	2.706607E+01	2.705106E+01	4.820067E-01	4.687015E-01
23	2.909711E+01	2.909882E+01	2.214295E-01	2.216114E-01

*The numbers in the generalized mass columns can be converted to the SI units (Kg) by multiplying by the factor 1.751268E+02.

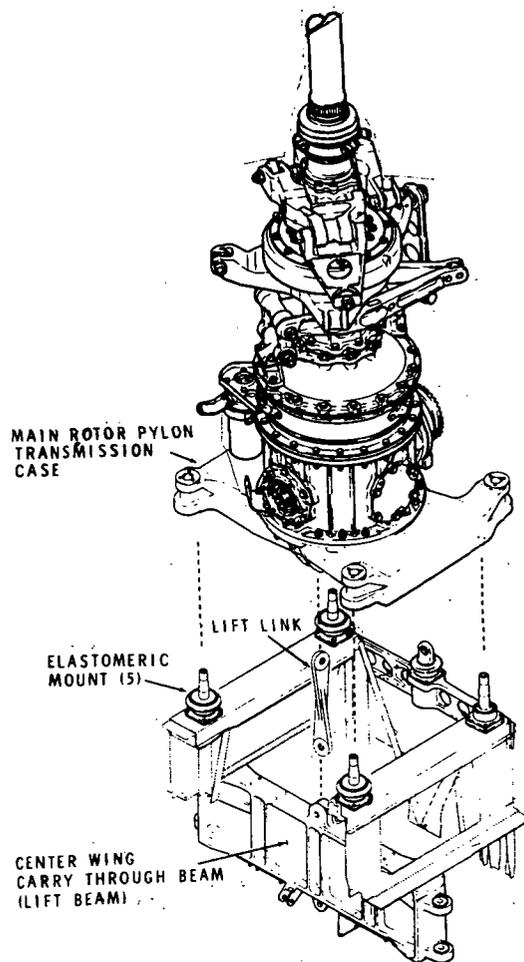


Figure 1. AH-1G Main Rotor Pylon on Rigid Body Fuselage (Example 1)

RIGID ELEMENTS

TYPE	LETTER/LINE DESIGNATION
CRIGD1	C, D, E
CRIGD2	F, ---
CRIGD3	—
CRIGDR	A

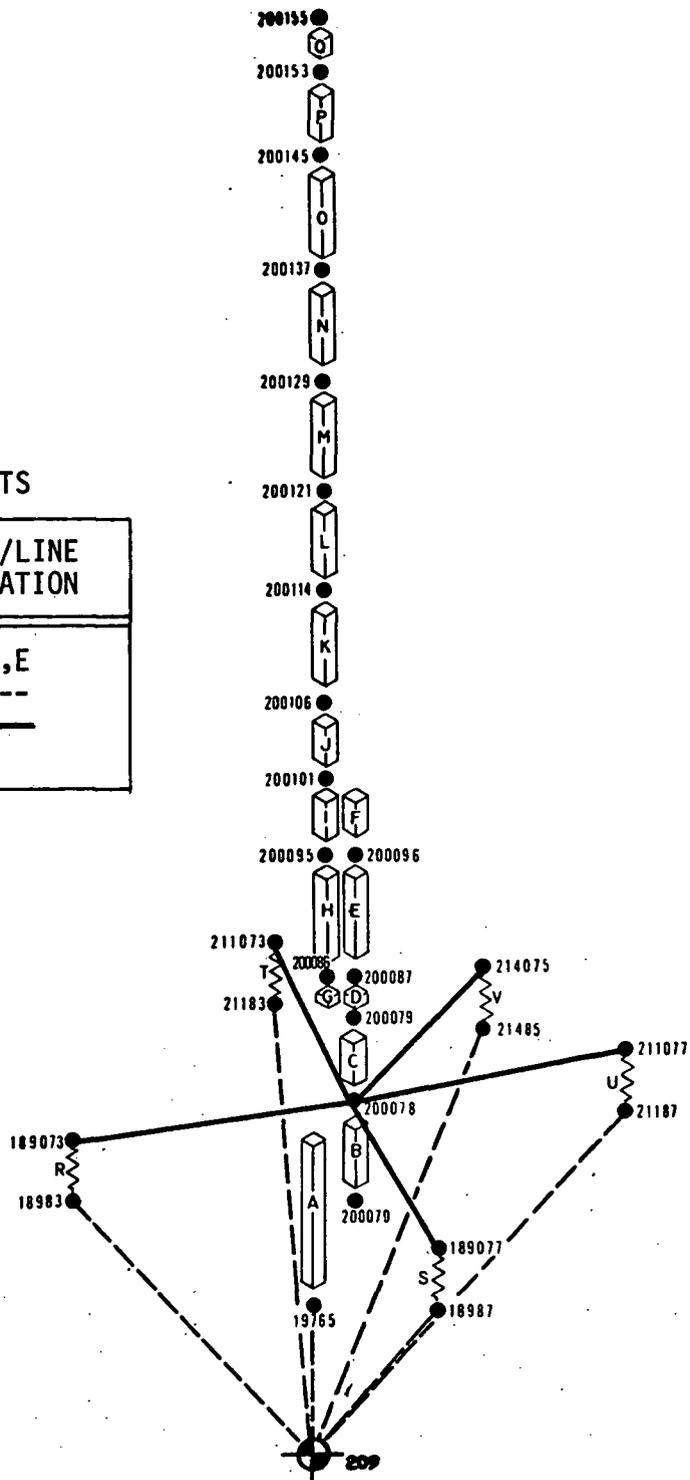


Figure 2. Finite Element Model for Rigid Body Fuselage shown in Figure 1 (Example 1)

RIGID ELEMENTS

TYPE	NO. USED
CRIGD1	2
CRIGD2	21
CRIGD3	1
CRIGDR	20

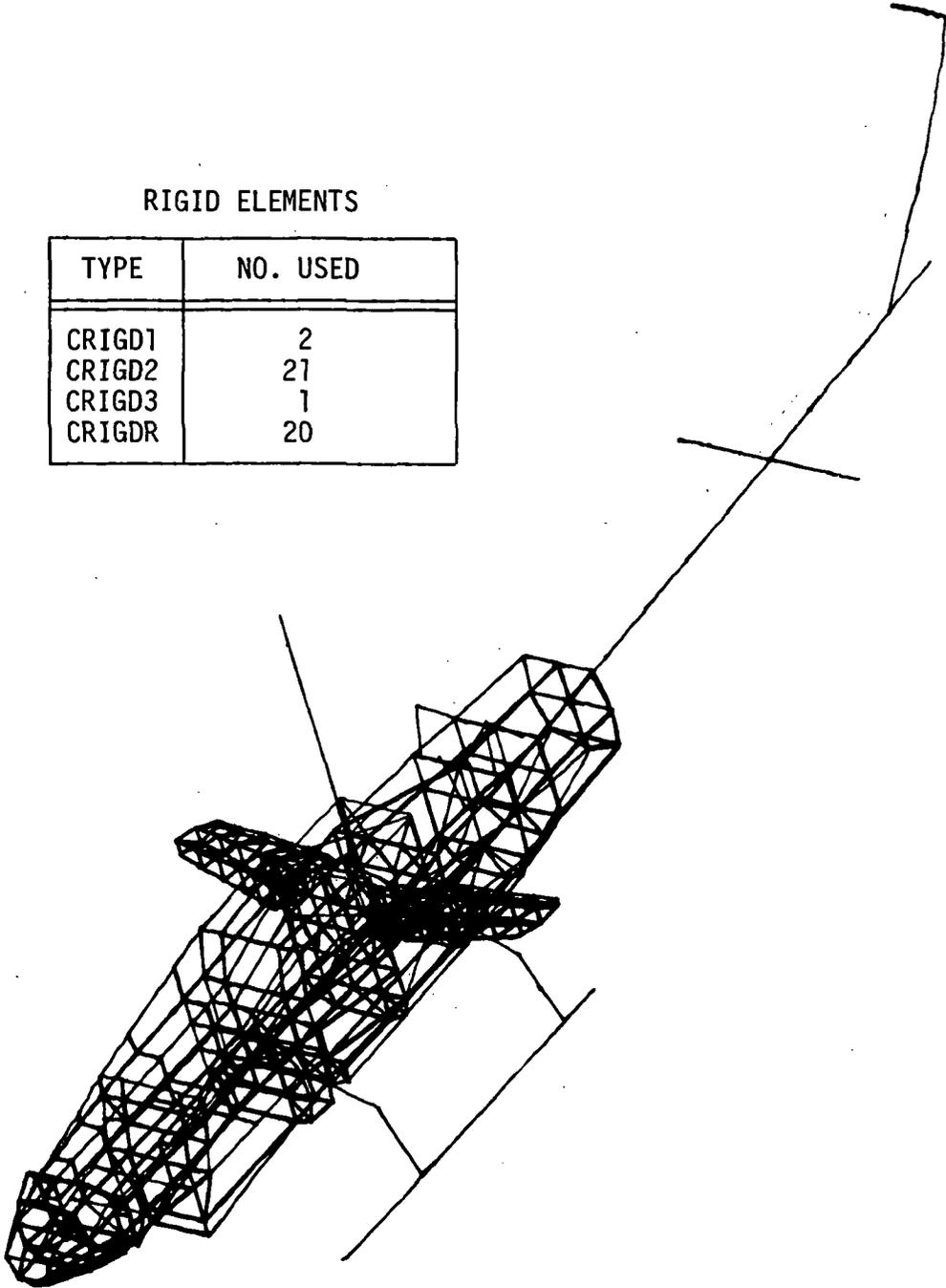


Figure 3. AH-1G Built-Up Airframe (Example 2)

USE OF NEW MATRIX ASSEMBLER FOR DIFFERENTIAL STIFFNESS MATRICES

R. Narayanaswami* and J. G. Cole**

Rockwell International

ABSTRACT

The assembling of the individual element differential stiffness matrices into the global differential stiffness matrix is currently performed in NASTRAN Level 16.0 on a grid-point-by-grid-point basis. This procedure is inefficient since the element differential stiffness matrices have to be formed as many times as there are grid points in the element. In this paper, an improved procedure for forming the global differential stiffness matrices is described. The procedure consists of modifying the existing element matrix generator module to receive the additional data blocks necessary for the evaluation of the element differential stiffness matrix. The element differential stiffness matrix is then used in the new matrix assembler to form the global differential stiffness matrix.

This procedure is used to solve two problems in differential-stiffness-related Rigid Formats 4 and 5. As expected, the improved procedure shows reduced run times in both the buckling-related NASTRAN problems.

* NASTRAN Project Engineer.

** Member of technical staff.

1. Report No. NASA CP -2018	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle SIXTH NASTRAN USERS' COLLOQUIUM		5. Report Date October 1977	
		6. Performing Organization Code	
7. Author(s)		8. Performing Organization Report No. E-9303	
		10. Work Unit No.	
9. Performing Organization Name and Address		11. Contract or Grant No.	
		13. Type of Report and Period Covered Conference Publication	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code	
		15. Supplementary Notes	
16. Abstract <p>This publication comprises papers prepared at the Sixth NASTRAN Users' Colloquium held at the NASA Lewis Research Center, October 4-6, 1977. The authors discuss the general application of finite element methodology and the specific application of NASTRAN to a wide variety of static and dynamic structural problems. Specifically, the papers focus on the following topics: Fluids and Thermal Applications, NASTRAN Programming, Substructuring Methods, Unique New Applications, Large Deformation Approaches, General Auxiliary Programs, Specific Applications, and New Capabilities.</p>			
17. Key Words (Suggested by Author(s)) NASTRAN; Structures; Finite element; Structural analysis; Nonlinear analysis; Flow problems; Substructuring; NASTRAN Colloquium		18. Distribution Statement Unclassified - unlimited STAR Category 39	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 489	22. Price* A21

* For sale by the National Technical Information Service, Springfield, Virginia 22161

National Aeronautics and
Space Administration

Washington, D.C.
20546

Official Business
Penalty for Private Use, \$300

SPECIAL FOURTH CLASS MAIL
BOOK

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451



10 1 082677 S19649DU
THE CONGRESS
SENATE COMMITTEE ON SCI & TRANSP
SUBCOMMITTEE ON SCIENCE TECH & SPACE
ATTN: MR JAMES GEHRIG
WASHINGTON DC 20510

NASA

POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Return
