AD-A095 146 MARC ANALYSIS RESEARCH CORP PALO ALTO CA F/G 13/13
KNOWLEDGE BASED CONSULTATION FOR FINITE ELEMENT STRUCTURAL ANAL--ETC(U)
MAY 80 J M RIVLIN, M B HSU, P V MARCAL F33615-78-C-3206
UNCLASSIFIED AFWAL-TR-80-3069 NL

1 OF 2

AD A095146

AFWAL-TR-80-3069

KNOWLEDGE BASED CONSULTATION FOR FINITE ELEMENT STRUCTURAL ANALYSIS

J. M. RIVLIN
M. B. HSU
P. V. MARCAL

MARC ANALYSIS RESEARCH CORPORATION
260 SHERIDAN AVENUE, SUITE 200
PALO ALTO, CALIFORNIA 94306

MAY 1980

TECHNICAL REPORT AFWAL-TR-80-3069
Final Report for Period October 1977 — May 1980

Approved for public release; distribution unlimited.

FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

NARENDRA S. KHOT,
Project Engineer

FREDERICK A. PICCHIONI, Lt Col, USAF
Chief, Analysis & Optimization Branch

FOR THE COMMANDER

RALPH L. KUSTER, JR., Col, USAF
Chief, Structures & Dynamics Div.

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/FIBR, W-PAFB, OH 45433 to help us maintain a current mailing list".

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE/56780/19 January 1981 — 200

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFWAL-TR-80-3069
2. GOVT ACCESSION NO.: AD-A095 146
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): Knowledge Based Consultation for Finite Element Structural Analysis.
5. TYPE OF REPORT & PERIOD COVERED: FINAL REPORT, October 1977 – May 1980.
6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): J. M. Rivlin, M. B. Hsu, P. V. Marcal
8. CONTRACT OR GRANT NUMBER(s): F33615-78-C-3206
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Flight Dynamics Laboratory (FIBRA), AF Wright Aeronautical Laboratories (AFSC), Wright-Patterson Air Force Base OH 45433
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Project 2401, Task 2401-02, Work Unit 2401-02-01
11. CONTROLLING OFFICE NAME AND ADDRESS:
12. REPORT DATE: May 1980
13. NUMBER OF PAGES: 114
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):
15. SECURITY CLASS. (of this report): UNCLASSIFIED
15a. DECLASSIFICATION DOWNGRADING SCHEDULE:
16. DISTRIBUTION STATEMENT (of this Report): Approved for public release; distribution unlimited
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):
18. SUPPLEMENTARY NOTES:
19. KEY WORDS (Continue on reverse side if necessary and identify by block number):
   - Knowledge Based Consultation
   - Artificial Intelligence
   - Finite Element Program
   - Tutorial

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In recent years, techniques of 'heuristic programming' have been applied to various fields of science and engineering. An attempt has been made, during the current investigation, to develop a knowledge based consultation system and to establish a finite element structural analysis knowledge base for the use of the MARC program, a general purpose finite element analysis program. Backward chaining logic is adopted into the current system and the inference engine is written in Fortran. The knowledge base deals with information on pre-analysis, and post-analysis stages of a structural analysis problem. Rules in the

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) knowledge base are executed through the use of user input keywords. The consultation system is currently operational on the Prime 400 Computer System.

FOREWORD

This report describes the work performed by MARC Analysis Research Corporation, Palo Alto, California. The work was sponsored by the Flight Dynamics Laboratory, Air Force Wright Aeronautical Laboratories, Wright-Patterson Air Force Base, Ohio, under contract F33615-78-C3206.

The work was initiated under Project 2401, "Structures and Dynamics," Task 24010201, "Design and Analysis Methods for Flight Vehicles." The work was administered by Dr. N. S. Khot, Project Engineer of the Structures and Dynamics Division (FIBR).

The contract work was performed between October 1977 and May 1980.

The technical report was released by the Authors in May 1980.

TABLE OF CONTENTS

List of Illustrations

List of Tables

I. INTRODUCTION

In recent years. techniques of 'Heuristic Programming' have been applied to various fields of science and engineering. The DENDRAL program [1] can be used for processing the results from mass and nuclear spectrometry and for the determination of the chemical nature of a substance that was tested. A group involved in Artificial Intelligence at Stanford University developed the program MYCIN [2], for clinical consultation of diseases that require treatment by penicillin and its derivatives. Studies have also been reported to apply knowledge engineering to the field of structural analysis [3], [4]. Results from previous investigations clearly indicate the advantage and usefulness of heuristic programming techniques in dealing with scientific and engineering problems.

The application of finite element methods to structural analysis has long been recognized. A general purpose finite element program such as the MARC program [5] can provide numerical results to a variety of structural problems involving material and geometric non-linearities, thermal effects and anisotropic material properties. A simple description of the MARC program is given in Appendix B.

During this investigation, an attempt has been made to develop a knowledge based consultation system and to establish a finite element structural analysis knowledge base for the use of the MARC program. The backward chaining logic is adopted into the current consultation system and the inference engine in the system is written in Fortran. The use of Fortran allows the incorporation of an interface to existing software. The finite element structural analysis knowledge base is developed for dealing with information on pre-analysis, analysis, and post-analysis stages of a structural analysis problem. Knowledge based rules are established for the determination of the type of analysis; preparation of input file to the MARC program; and examination of the structural responses. All the rules are executed through the use of keywords. Since the MARC program is a general purpose finite element analysis program, a great number of analysis capabilities are available in the program. During this investigation, however, only a few types of analyses and a number of selected elements were chosen for the knowledge base. Expansion of current knowledge base to include additional types of analyses and other elements can be easily achieved in a future investigation.

In the following sections, the knowledge based consultation system is discussed in Section II and the knowledge base for finite element structural analysis is described in Section III. Application of this consultation system to structural problems is demonstrated in Section IV while an attempt to arrive at some conclusions is given in Section V. In addition, a list of references is compiled in Section VI. Finally, the knowledge based rules and their associated questions are shown in Appendix A and a simple description of the MARC program is given in Appendix B. A list of conversations held during consultations is included in Appendix C.

II. KNOWLEDGE BASED CONSULTATION SYSTEM

This section outlines the method used to represent a consultant's knowledge of his area of specialization. It is assumed that the consultant or expert has a thorough knowledge of his area of expertise, but that his knowledge in the area of computing is limited. One of the primary motivating criteria used in selecting the knowledge representation was that a non-programmer can easily and directly implement knowledge without the intervention of computing staff.

1. The Inference Engine and Rules

The basic backward chaining logic, characteristic to Artificial Intelligence, approaching the problem of knowledge representation was adopted into the current system. The inference engine in the system is written in Fortran. This has one major advantage. It allows the incorporation of an interface to a wealth of existing software. In addition, use of Fortran also allows the addition of sophisticated evaluation routines on which to base decision making by the consultant. This, of course, requires some work on the part of programming personnel, however, frequently such analysis programs are already available and can be easily interfaced to the inference engine. These features allow a natural growth and specialization of the inference engine to a variety of tasks.

The inference engine accepts all information in the form of rules. Rules have the form:

Rule #
Premise
Conclusion.

The Rule number is a unique number used to identify a rule. The value of a rule number does not imply execution order. A properly written rule should contain a kernel of knowledge which is true at any time during the execution of the rule set in which it is contained. This means that the premise should solely dictate the validity of the conclusion. In addition, the author of the rules should avoid dependence in rule execution order to produce the desired result.

The modularity of the rule representation makes the sets exceedingly easy to modify. It is usually possible to introduce a new concept into a rule set by simply adding the new criteria required to draw the desired conclusion. This is possible because the execution order of the rules is not specified by the author of the rules, it is implied by the content of the rules themselves.

Once a rule has been chosen for processing it is processed in the following fashion:

a. The premises of the rule (IF elements) are evaluated in the order in which they appear in the rule. The evaluation of an IF statement may lead to one of three possible results. The statement is true, false or unknown. If the statement is true, evaluation proceeds with the next IF statement. If the statement is false or unknown evaluation of current rule is terminated.

b. If the statement is true, the conclusion portion of the rule is executed. The conclusion may include the asking of questions, an internal conclusion, or a variety of other tasks. It is expected that permissable actions will be enhanced to suit a variety of situations.

c. The unknown condition results when one of the names referenced in the premise does not have a value associated with it. When the unknown condition occurs the name of the variable causing the condition is saved for back chaining.

The inference engine operates in a goal directed fashion. This means that all activity is directed towards concluding a value for an initially unknown keyword. Backward chaining is then performed from a rule in which the desired keyword is referenced as an assigned variable in the conclusion. Such a rule may be referred to as a goal rule. If the rule is true, then the search is over and the keyword achieves a role. If the rule is false, then another goal rule is found. If the rule is unknown, then the variable causing the unknown condition is used as a directive. This variable name is found as a conclusion variable in another rule. That rule is then executed. Execution proceeds in a directed fashion until a rule containing the initial keyword is found to be true. Figure 1 shows the execution algorithm graphically.

2. Command Descriptions

a. Logic Statements:

RULE, IF, CONCLUDE.

Logic statements are statements which control the execution logic of the rule set. Conventional programming languages allow the user to explicitly define which section of the program is to be executed next. This is the primary difference between the inference engine and conventional programming languages. All execution information is contained in the three available logic statements.

```
Example:  RULE 29
          IF WEIGHT IS 'LOW'
          CONCLUDE MAN IS 'HUNGRY'
```

(1) RULE:

RULE (NUMBER)

The RULE statement defines a logically related group of statements. Every RULE must contain at least one conclusion. The conclusion is the only means the inference engine has of determining that a rule pertains to a particular topic. In the above example the RULE 29 will never be executed unless a value of 'MAN' is required to achieve a goal. The RULE statement serves to delineate both the end of the previous rule and the beginning of a new rule. IT is also possible to specify a rule with no number. In this instance the rule will be executed once before any other rules are executed. This provides a means for performing initialization.

(2) IF:

IF (expression) (operator) (expression)

The IF statement serves to indicate a premise of the rule. There may be multiple IF statements preceeding the actions and conclusions of a single rule. Any variable which is determined to be unknown during the evaluation of a rule will be used to back chain to another rule containing an assignment to the unknown variable in its conclusion.

(3) CONCLUDE:

CONCLUDE (variable) (assignment operator) (expression)

Conclusion statements are the basis for the selection of a rule for execution. A rule may be executed whenever one of the conclusions of the rule contains the variable that was determined as unknown in the premise of the previously executed rule. All activation of a rule is done on the basis of the conclusion statements of the rule. For this reason every RULE must contain at least one conclusion or it will never be activated.

b. Syntactic Constructs:

(1) Variables:

A variable name may contain up to 14 characters. Numerics and operator symbols may not be used in a variable name. Variables do not need to be declared as in a conventional programming language. The type of a variable is determined by the type of data which is assigned to that variable. Variables may be arrays or lists of values. Arrays of up to three dimensions are supported. Any array of more than one dimension must be dimensioned by means of a 'DIM' statement. Variables may also contain lists. This is indicated by the "ADD_LIST" statement.

(2) Constants:

A constant may be one of three forms: a floating point number, an integer number or a character string. These are indicated by the following example:

(a) 2.33

(b) 233

(c) 'two hundred thirty three'

(3) Expressions:

An expression is formed by operators, variables and constants.

No parenthesis may be used in an expression other than those used to identify array references.

Example:

```
(a)  'A' + 'B' + 'CDE'

                this expression is the same as 'ABCDE'

(b)  A = 2.47
     A + 3

                the expression is the same as 5.47

(c)  A(1) = 2.47
     A(1) + 3

                the expression is the same as 5.47
```

Expressions are always evaluated in a strict left to right fashion.

(4) Conditional Operators:

Conditional operators are the method of specifying logical relationships between expressions. The possible conditional operators are:

EQ,NE,LT,GT,LE,GE,IS_IN,IS,IS_NOT,IS_NOT_IN,
ENDING_IS,ENDING_ISNT

The definitions of these conditional operators are given below.

EQ, IS - truth occurs when the expression on the left is equal to the expression on the right.

NE, IS_NOT - truth occurs when the expression on the left is not equal to the expression on the right.

LT - truth occurs when the expression on the left is less than the expression on the right.

GT - truth occurs when the expression on the left is greater than the expression on the right.

LE - truth occurs when the expression on the left is less than or equal to the expression on the right.

GE - Truth occurs when the expression on the left is greater than or equal to the expression on the right.

IS_IN - truth occurs when the expression on the left is contained in the list on the right.

IS_NOT_IN - truth occurs when the expression on the left is not contained in the list on the right.

ENDING_IS - truth occurs when the ending of the expression is on the left, which must evaluate to a character string, is the same as the expression on the right.

ENDING_ISNT - truth occurs when the ending of the expression on the left, which must evaluate to a character string, is the same as the expresson on the right.

(5) Assignment Operators:

Assignment operators may be used in "CONCLUDE" and "ACTION" statements. They always specify assignment to the variable on the left. The possible assignment operators are: ADD_LIST,REMOVE_LIST,=,IS.

ADD_LIST - the expression on the right is added as a new element to the list on the left.

REMOVE_LIST - the expression on the right is searched in the list on the left and removed from that list if it exists.

=,IS - the expression on the right is evaluated and assigned to the variable on the left.

(6) Operators:

Operators connect the variables in an expression. The possible operators are: *,-,+,/

*	(numerics only)	- multiplication
-	(numerics)	- subtraction
-	(character strings)	- removal of first occurence of character string on the right from expression on the left
+	(numerics)	- addition
+	(character strings)	- concatenation
/	(numerics only)	- division

(7) Additional Action Statements:

The following statements allow the rules to perform useful functions, however, they in no way affect the flow of execution in the way that the CONCLUDE statement does.

ACTION·

ACTION (variable) (assignment operator) (expression)

The action statement is exactly the same as the conclude statement, except that variables referenced are not used to key off the execution of the rule in which the statement occurs.

ASK:

ASK ('question')(# times)(insert)(# variables)(variables) This inference engine maintains a question file. This file is specified at start up time of the program. The ASK feature allows the rules to ask questions which are specified in detail in another file.

Question - specifies the question name to be used.

#times - the number of times to ask the specified question.

insert - the character string or number to be inserted in the question being asked.

#variables - the number of variables which will receive the information typed in by the user. This number must be the same as the number of fields specified in the question definition file.

variables - the names of the variables to receive the data typed in by the user. If the #times is greater than 1, then the variable will be an array with successive answers appearing in successive elements.

DIM:

DIM (variable) (N1,N2,N3)

The DIM statement dimensions multi-dimensional arrays. N1, N2 and N3 are the maximum bounds of the dimensions. Arrays may have maximum of three dimensions. The DIM statement should only appear in initialization rules. It is only necessary to dimension two-and three-dimensional arrays.

OPEN:

OPEN ('filename') (unit) (record length)

The OPEN statement opens a file of the specified filename on the specified unit number. Unit may be 1, 2 or 3. The record length is negative if an output file is being opened and positive if an input file is being specified. In either case record length is specified in characters.

CLOSE:

CLOSE (unit)

The CLOSE statement closes the file open on the specified unit.

MERGE:

MERGE ('file1') ('file2') ('file3') ('output file')

The MERGE statement merges from 1 to 3 files into the specified output file.

WRITE:

WRITE (unit) (variable1) (variable2) ... : (from to by)

The WRITE statement allows the rules to write to a file. The specified unit must be opened with OPEN command. Variable 1, Variable 2, etc. are arrays containing the data to be written. From to and by form a loop specifying the array elements to write. Each increment of this loop creates 'record' in the output file. The FORMAT for the write is specified on the line following the WRITE statement. It is specified as a standard Fortran FORMAT statement.

Example:

```
CONCLUDE A(1) = 3
CONCLUDE A(2) = 4
CONCLUDE B(1) = 5.1
CONCLUDE B(2) = 5.2

WRITE 1 A B : 1 2 1
FORMAT (I5,F4.2)
```

The result is:

```
3 5.1
4 5.2
```

3. Inference Engine Internals

The internals of the inference engine may be thought of as being divided into two sections. These are the utility module and the logic module. The utility module implements all of the functions necessary to perform the actions of rules. This includes such things as string conversions. file handling, parsing etc. There is also a hash coded data retrieval system associated with this module. All variable names and their respective data are stored in the hash coded data-base. The utility module consists of approximately 80 routines. The login of the inference engine constitutes the second module. This module consists of the routines which actually make the decisions as to which rule to process and actually pass and process the rules. This of course is heavily dependent on the utility module for most of its functionality.

The following is a brief description of the primary routines involved in the logic process and their functionalities.

Routine	Functionality
CONCLUD	Performs conclusions
ERROR	Central error handling routine
EVAL	Evaluates expressions
BRAINS	Controls execution flow
INRULE	Reads in rules
PROC	Processes a rule
QUERY	Asks user when all else fails
QUEST	Processes a premise
STKRUL	Finds rules relevant to keyword

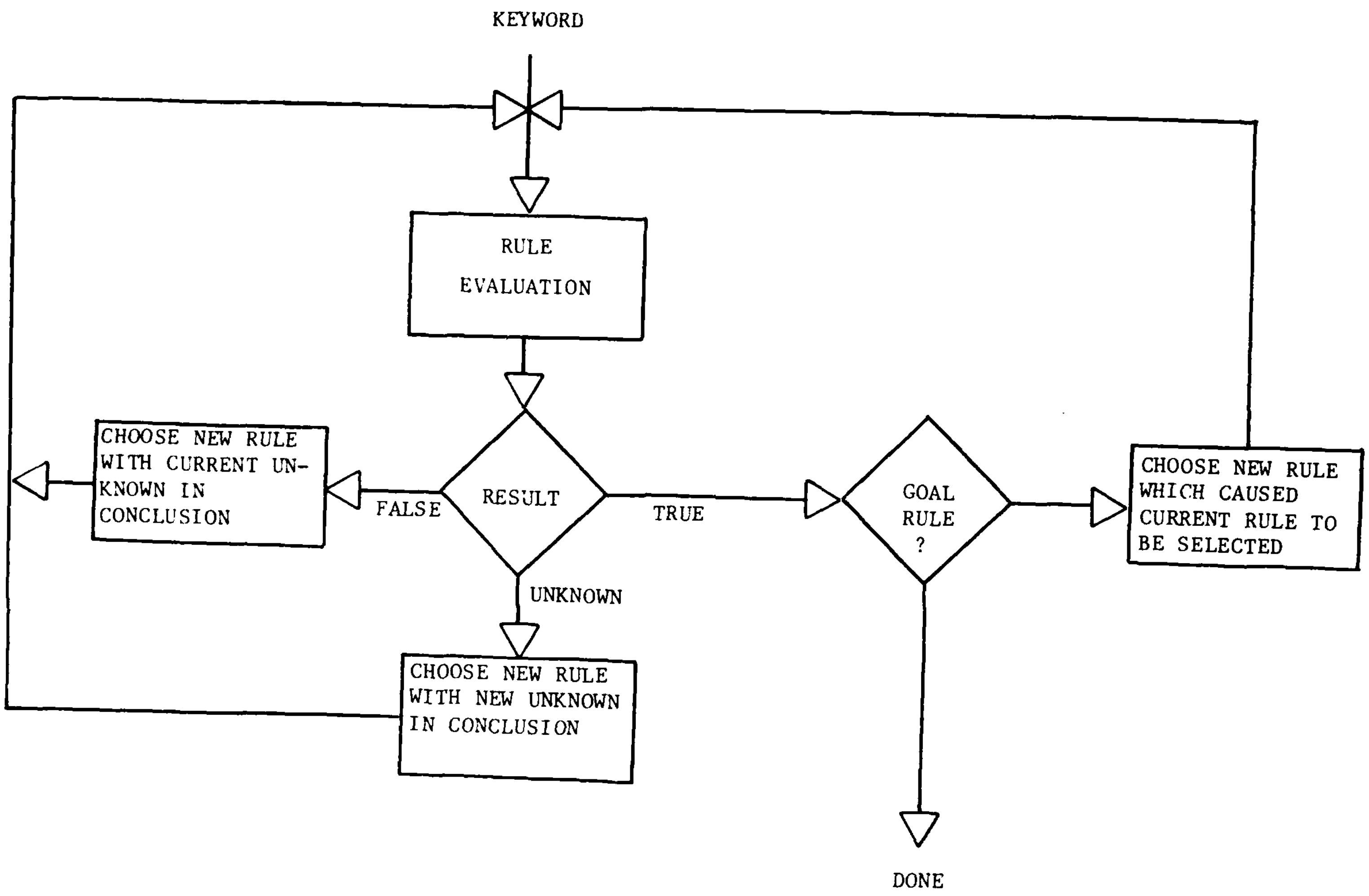


Figure 1 Algorithm of Rule Execution

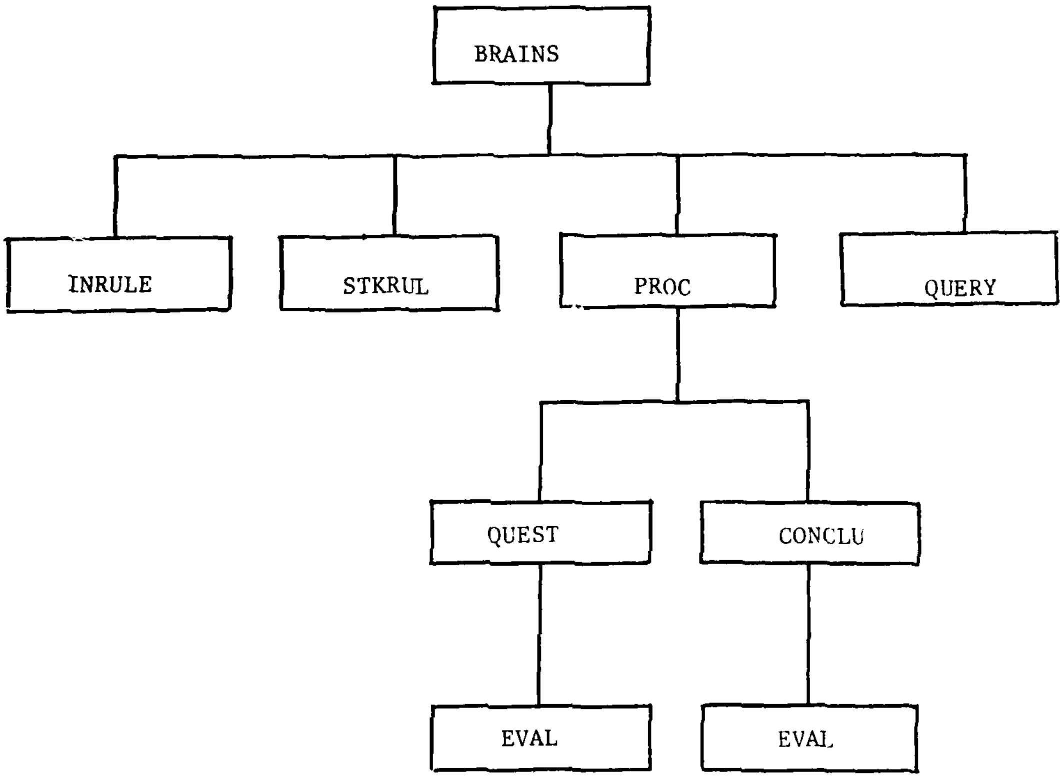


Figure 2 Flow Diagram of the Consultation System

III. KNOWLEDGE BASE FOR FINITE ELEMENT STRUCTURAL ANALYSIS

In the previous section a knowledge based consultation system has been discussed in great detail. The system asks for a 'keyword' from the user and searches through the rules previously defined in a knowledge data base for an answer. If a conclusion can be drawn from the rules stored in the knowledge base, the system will respond to the user's question (keyword) with a positive answer. Otherwise, the system will indicate that currently it is not possible for the system to answer the user's question for lack of expertise in his question. It is obvious that the knowledge base must be constantly updated because new knowledge is required for dealing with new situations.

The application of finite element methods to structural analysis has long been recognized. During this investigation, an attempt has been made to establish a finite element structural analysis knowledge base for the purpose of demonstrating the use of the current knowledge based consultation system for finite element structural analysis. As an initial attempt, this knowledge base is limited to the use of MARC - a general purpose finite element program [5]. Extension of this knowledge base to include other programs is possible in future investigations. In this section, a description of this finite element structural analysis knowledge base is presented.

1. <u>Finite Element Structural Analysis Information</u>

In a structural analysis, there are at least three principal stages where questions must be answered and decisions must be made by engineers. These stages generally include: (1) pre-analysis, (2) analysis and (3) post-analysis. A summary of necessary information associated with these stages is given below.

a. Pre-Analysis Stage

Prior to performing a structural analysis, the type of analysis and the method of analysis are two most important questions to be answered by an analyst. Since the finite element method has been chosen for the method of analysis in the current investigation, only the type of analysis remains to be determined. Generally, the type of analysis of a structure can be adequately defined by the knowledge of: (a) geometrical configuration of the structure, (b) loading environment, (c) material behavior and (d) time dependency of the structural response. With sufficient knowledge of the geometry, loading, material and expected response of the structure, a specific type of analysis (i.e. static, dynamic, linear-elastic, elastic-plastic, small-displacement, large-displacement, etc.) can be reasonably recommended by an expert consultant. As a result, this set of information is the very first group of data stored in the current finite element structural analysis knowledge base.

b. Analysis Stage

The information in the analysis stage consists of input data of the structural problem to be analyzed and the response (output) of the structural system subjected to given loading conditions. The data is generally numerical data or in the form of alphanumerics. Within the context of finite element analysis, the analyst must usually supply the following information: (a) finite element mesh, in the form of element connectivity and nodal coordinates, (b) material properties (i.e. modulus of elasticity Poisson's ratio, yield stress, etc.), (c) loadings (i.e. mechanical, thermal, point-load, distributed-load, etc.) and displacement boundary conditions, and (d) transient conditions. On the basis of this information, an input-file can be generated for numerical analysis using a finite element structural analysis program (e.g. MARC). The execution of the finite element structural analysis program yields the desired response of the structural system. In the current finite element structural analysis knowledge base this set of information is stored numerically, and transformed into the input format of the MARC program for execution.

c. Post-Analysis Stage

The results of a finite element structural analysis consist of nodal and element quantities. The values associated with nodal points are generally displacements, velocities, accelerations and temperatures while the element quantities are often stresses, strains and temperatures. This set of information can be carefully studied for adequacy of the analysis and for the recommendation of further analysis. Examination of the analysis results is also included in the current finite element structural analysis knowledge base.

2. Knowledge Based Rules and Key-Words

Since the current knowledge based consultation system has been developed on the basis of internal search of knowledge based rules and user input key-words, a set of knowledge based rules and key-words has been established for the finite element structural analysis. A list of the rules is given in Appendix A and all the key-words are listed in Table 1. During a consultation, the user must input the key-words as listed in Table 1 in order to activate the knowledge based rules for recommendations to his problem. No conclusion will be drawn by the consultation system if an inappropriate key-word is used. The key-words must be used one at a time and the order of the key-words may, or may not, be important.

The first group of knowledge based rules are prepared for acquiring alphabetic or numerical information from the user for the understanding of his structural problem by asking the user a general question. As an example in RULE 201 and RULE 202 the information sought for are 'DIMENSIONALITY' of the structure and the total number of elements 'NUMEL' in the mesh, respectively.

```
RULE 201
ASK 'MARCINO1' 1 0 1 DIMENSIONALITY
  .
  .
  .
CONCLUDE GENERAL_ONE IS 'GIVEN'
RULE 202
ASK 'MARCIN31' 1 0 1 NUMEL
  .
  .
  .
CONCLUDE GENERAL_TWO IS 'GIVEN'
```

The key-words to be used for activating these rules are GENERAL_ONE and GENERAL_TWO and the questions are stored on files 'MARCINO1' and 'MARCIN31', respectively. Further details are given in Appendix A.

```
FILE 'MARCINO1'
MARCINO1GENERAL
OC15
1PLEASE ENTER DIMENSIONALITY OF YOUR
1STRUCTURE: ONE, TWO OR THREE

FILE 'MARCIN31'
MARCIN31GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF ELEMENTS IN THE MESH
```

In answering these questions, the user must input either ONE, TWO, or THREE for DIMENSIONALITY and a numerical value (say 10, 10 elements in the mesh) for NUMEL.

The next group of knowledge based rules are intended for decision making on the basis of available information. For example in RULE 1 the GEOMETRY of the structure is concluded to be ONE_DIMENSIONAL_BEAM, from the available information that the DIMENSIONALITY of the structure is ONE and the overall THICKNESS of the structure is THIN_SLENDER. In addition, MARC Element No. 5 (See Appendix B) is chosen (JTYPE = 5) for the finite element mesh and the GEOMETRY_CHECK is concluded to be OK.

```
RULE 1
IF DIMENSIONALITY IS 'ONE'
IF THICKNESS IS 'THIN_SLENDER'
CONCLUDE JTYPE = 5
WRITE 0 :
FORMAT ('MARC ELEMENT NO 5 SELECTED FOR THE ANALYSIS')
CONCLUDE GEOMETRY_CHECK IS 'OK'
CONCLUDE GEOMETRY IS 'ONE_DIMENSIONAL_BEAM'
```

For the execution of RULE 1, the key-word is either GEOMETRY or GEOMETRY_CHECK.

The third group of rules are developed for creating an input file for the execution of the MARC Program. An example is shown in RULE 203.

```
RULE 203
OPEN 'INPUT1' 1  80
WRITE 0 :
FORMAT ('SET DEFAULTS MAXALL ETC')
CONCLUDE MAXALL  = 10000
CONCLUDE NUMBC   = 200
CONCLUDE MPRMAX    10
CONCLUDE MESHR   = 5
CONCLUDE MCRD    = 3
CONCLUDE NBINC   = 100
CONCLUDE FACTOR  = 0.1
CONCLUDE NWKSLP  = 1
WRITE 0 :
FORMAT ('WRITE TITLE CARD ON INPUT1')
WRITE 1 :
FORMAT ('TITLE      TEST INPUT FOR AI')
WRITE 1 :
FORMAT ('PRIME,5500024,')
WRITE 0 :
FORMAT ('WRITE SIZING CARD ON INPUT1')
WRITE 1 MAXALL NUMEL NUMNP NUMBC MPRMAX JTYPE : 1 1 1
FORMAT ('SIZING   ',I10,5I5)
CONCLUDE P_CARD_ONE IS 'GIVEN'
```

In RULE 203 file 'INPUT1' is opened first for the storage of input data, and then default values (MAXALL = 10000, NUMBC = 200, etc) are pre-set. P_card images (See Appendix B) of MARC input data (partial) are created and temporarily stored on file 'INPUT1' for final merge. It can be seen that the key-word for the execution of RULE 203 is P_CARD_ONE.

After temporary files 'INPUT1', 'INPUT2' and 'INPUT3' being created for P_CARD M_card and L_card images of MARC input data, a merge of these three files into a final input file 'INPUT' takes place. The merge operation is carried out by the execution of RULE 216, using the key-word INPUT_FILE.

```
RULE 216
CLOSE 1
CLOSE 2
CLOSE 3
MERGE 'INPUT1' 'INPUT2' 'INPUT3' 'INPUT'
CONCLUDE INPUT_FILE IS 'GIVEN'
```

Additional rules have been written for the review of MARC results and for the recommendation of reanalysis. RULE 251 and RULE 252 are two examples. The key-words for these two rules are RESULT_ONE, RESULT_TWO, respectively.

```
RULE 251
OPEN 'USDATA' 1 40
READ 1 IR : 1 1 1
FORMAT (15)
READ 1 VMAX : 1 9 1
FORMAT (E15.4)
WRITE O :
FORMAT ('READ SUMMARY OF MARC RESULTS FROM USDATA')
CONCLUDE AA = VMAX(1)
CONCLUDE AB = VMAX(2)
CONCLUDE AC = VMAX(3)
CONCLUDE AD = VMAX(4)
CONCLUDE AE = VMAX(5)
CONCLUDE AF = VMAX(6)
CONCLUDE AG = VMAX(7)
CONCLUDE AH = VMAX(8)
CONCLUDE AI = VMAX(9)
CONCLUDE RESULT_ONE IS 'GIVEN'
```

```
RULE 252
WRITE O IF : 1 1 1
FORMAT (RESULTS FOR INCREMENT NO     ',15)
WRITE O AA : 1 1 1
FORMAT ('MAX XX STRAIN IS     ',E15.4)
WRITE O AB : 1 1 1
FORMAT ('MAX YY STRAIN IS     ',E15.4)
WRITE O AC : 1 1 1
FORMAT ('MAX XY STRAIN IS     ',E15.4)
WRITE O AD : 1 1 1
FORMAT ('MAX EQUIV. PLASTIC STRAIN IS     ',E15.4)
WRITE O AE : 1 1 1
FORMAT ('MAX EQUIV. CREEP   STRAIN IS     ',F15.4)
WRITE O AF : 1 1 1
FORMAT ('MAX XX STRAIN IS     ',E15.4)
WRITE O AG : 1 1 1
FORMAT ('MAX YY STRESS IS     ',F15.4)
WRITE O AH : 1 1 1
FORMAT ('MAX XY STRESS IS     ',E15.4)
WRITE O AI : 1 1 1
FORMAT ('MAX EQUIV. STRESS         IS     ',E15.4)
CONCLUDE RESULT_TWO IS 'GIVEN'
```

3. Reasoning Steps

As mentioned in Section 1, the pre-analysis stage information dictates the type of structural analysis to be performed and the analysis stage information provides input to a finite element analysis program for numerical results. In the current investigation, this information is transferred from the user to the knowledge based consultation system through the execution of knowledge rules activated by key-words. Based on the available information, decisions are generally made directly from the user's input, or from previously derived conclusions. As an example of dealing with the pre-analysis stage information assume that the answers provided by the user to the questions in RULE 201 are:

```
Dimensionality of the stucture            is TWO
Overall thickness of the structure        is THIN_FLAT
Type of loading                           is THERMO_MECHANICAL
Time dependency of mechanical loads       is INDEPENDENT
Temperature level of the structure        is HIGH
Stress-strain relation of the material    is NON-LINEAR
```

Then the following conclusions will be drawn by the consultation system directly from user information.

a. RULE 101
   MARC ELEMENT NO 3 SELECTED FOR THE ANALYSIS
   GEOMETRY IS 'TWO_D_PLANE_STRESS'

b. RULE 13
   LOADING IS 'ELEVATED_THERMO_MECHANICAL'

c. RULE 19
   MATERIAL IS 'NON_LINEAR_STRESS_STRAIN'

On the basis of the above conclusions. further decisions are made by the consultation system with regard to the type of analysis:

d. RULE 23
   ANALYSIS_TWO IS 'ELASTIC_PLASTIC'

e. RULE 26
   ANALYSIS_THREE IS 'CREEP'

Consequently, a two-dimensional plane-stress, elastic-plastic-creep analysis is recommended to the user for his structural analysis problem by the consultation system. It is obvious from the above example that no numerical input is required for the pre-analysis stage.

In dealing with the analysis-stage information, which is essentially a transformation of a physical model (real structure) into a numerical model (input data) for finite element analysis, the process is usually tedious and cumbersome. The user is always burdened with a large quantity of numerical data and various input formats. The current consultation system only requires necessary information about the structure system from the user, and makes the decisions for the user to comply with the MARC program input requirements. Since the current consultation system does not contain automated capabilities such as mesh generators and built-in material properties, the user must input this group of information through various key-words. Consequently, the user must supply first a number of counters (e.g. number of elements, number of nodes, number of boundary conditions, etc.) and then input additional information sets associated with these counters. An example of this process is that the user defines the number of nodal points (NUMNP) in the mesh by executing RULE 202 using key-word GENERAL_TWO.

```
RULE 202
  .
  .
  .
ASK 'MARCIN32' 1 0 1 NUMNP
  .
  .
```

and then input nodal point coordinates for NUMNP times through the execution of RULE 222 USING KEY-WORD COORDINATES.

```
RULE 222
ASK 'COORDINA' NUMNP 0 4 IDNODE XA XB XC
CONCLUDE COORDINATES IS 'GIVEN'
```

The nodal coordinates X , Y , Z , $i = 1,2,\ldots,$NUMNP are stored in arrays XA, XB and XC, respectively. The acquisition of nodal point coordinates for NUMNP times is decided by the consultation system. Obviously, RULE 202 must be executed before the execution of RULE 222 because the variable NUMNP is being used in RULE 222 and must be defined.

The reasoning steps discussed in this section are by no means complex ones. However, these simple decision making procedures serve well to illustrate the use of the knowledge based consultation system for finite element structural analysis.

TABLE 1 KEY-WORD LIST

GENERAL_ONE	GENERAL_TWO	GEOMETRY
LOADING	MATERIAL	ANALYSIS_ONE
ANALYSIS-TWO	ANALYSIS_THREE	ANALYSIS_FOUR
ANALYSIS_FIVE	CONNECTIVITY	COORDINATES
FMT_THICKNESS	THICKNESS_RANGE	BASE_PROPERTY
MATERIAL_RANGE	BOUNDARY_C	POINT_LOAD
DIST_LOAD	WORK_HARDENING	TEMP_EFFECTS
P_CARD_ONE	P_CARD_TWO	P_CARD_THREE
P_CARD_FOUR	M_CARD_ONE	M_CARD_TWO
M_CARD_THREE	L_CARD_ONE	GEOMETRY_CHECK
LOADING_CHECK	MATERIAL_CHECK	ANALYSIS_CHECK
KEY_WORD_LIST	INPUT_FILE	P_CARD_FIVE
L_CARD_TWO	L_CARD_THREE	L_CARD_FOUR
L_CARD_FIVE	RESULT_ONE	RESULT_TWO
RESULT_THREE		

IV. EXAMPLE CONSULTATIONS

For the purpose of demonstrating the application of the current consultation system to structural analysis, a two-dimensional plane-stress problem was analyzed using the MARC program. In this section, a description of the problem and the procedure of using the consultation system for structural analysis are discussed. A list of the conversations held during the consultation is given in Appendix C.

1. Description of the Structure

As shown in Fig. 3, the structure is a two-dimensional rectangular thin plate with a circular hole located at the center of the plate. It is subjected to a uniformly distributed tensile load along the major axial direction of the plate. The material properties of the plate are assumed to be: Modulus of Elasticity = 30 x 10 psi, Poison's Ratio = 0.3, Yield stress = 20,000 psi, and the plate thickness is 1.0 inch. Due to the symmetry conditions existing in the problem, only one-quarter of the structure is needed for finite element analysis. A finite element mesh is also shown in Fig. 3 where both the x- and y- axis are lines of symmetry. The mesh consists of 8 elements and 15 nodal points.

2. Pre-analysis and Analysis Stages (Initial Consultation)

Fig. 4 shows a schematic of the consultation procedure. The system initially asks for general information about the structure for the determination of the type of analysis to be performed and then seeks additional information about the finite element model for the preparation of the MARC input file. A summary of this group of information is as follows:

a. General Information

- The structure is two-dimensional.
- The appearance of the structure is thin and flat.
- The loading is mechanical and is independent of time. It is a uniformly distributed load.
- The temperature environment is room temperature.
- The structure consists of one material.
- The thickness of the plate is uniform.
- The structure has two lines of symmetry (x- and y- axis).

b. Finite Element Model

- Element No. and Element Connectivity

Element No.	Element Connectivity			
1	1	2	5	4
2	2	3	6	5
3	4	5	8	7
4	5	6	9	8
5	7	8	11	10
6	8	9	12	11
7	10	11	14	13
8	11	12	15	14

- Node No. and Nodal Coordinates

1	3.0	0.0
2	4.5	0.0
3	6.0	0.0
4	2.772	1.148
5	4.386	2.574
6	6.0	4.0
7	2.121	2.121
8	4.06	5.06
9	6.0	8.0
10	1.148	2.772
11	2.074	5.386
12	3.0	8.0
13	0.0	3.0
14	0.0	5.5
15	0.0	8.0

- Plate Thickness

  1.0

- Material Properties

  Modulus of Elasticity = 30 x 10

  Poisson's Ratio = 0.3

  Yield Stress = 20,000

- Boundary Conditions (at lines of symmetry)

  U = 0 at nodes 13, 14, 15 (Horizontal Displacement)

  V = 0 at nodes 1, 2, 3 (Vertical Displacement)

- Loading

  Traction = 10000 (tension)

  Associated Elements 6, 8

  Face Identification 6 (see Appendix B)

A MARC input file, generated by the consultation system, is shown in Table 2. This input file is compatible with the analysis recommended by the consultation system: a two-dimensional, plane-stress, linear-elastic analysis using MARC element number 3.

3. Execution of the MARC Program

During this investigation, the execution of the MARC program and the summarization of MARC results have been carried out separately from the consultation system. This approach is not always necessary but has certain advantages. An off-line execution of the MARC program frees the consultation system and the user from waiting for the program execution. The system can thus be used for further consultations. After the execution of the MARC program, a summary of MARC results (maximum stresses and strains) is written on a file called 'USDATA'. This file is reviewed by the consultation system for making additional recommendations.

4. Post-Analysis Stage (Second Consultation)

During a post-analysis stage, the consultation system compares stresses and strains with given criterions such as yielding of the material and strain limit requirements set by design codes. A reanalysis is recommended if the built-in limits are violated. It can be seen from Table 2 that the maximum equivalent stress (29,820 psi) in the plate does exceed the yield stress of the material (20,000 psi). The system, therefore, recommends a reanalysis to include the effect of plastic flow in the plate.

5. Reanalysis (Third Consultation)

The procedure of setting up a reanalysis is a repeat of the steps 2, 3, and 4. In the current example, the only additional information required for the reanalysis is:

- The stress-strain relation of the material is non-linear.

The reanalysis results show that the built-in inelastic strain limit (5%) has not been violated. Therefore, a further analysis is not required. As a result, the consultation is completed. MARC input data for the reanalysis is given in Table 4 and a summary of the reanalysis results is shown in Table 5.

TABLE 2   MARC INPUT DATA (SAMPLE PROBLEM)

```
TITLE        TEST INPUT FOR AI
PRIME,5500024.
SIZING           10000     8    15   200    10     3
ALL POINTS
END
CONNECTIVI
    8    5
    1    3    1    2    5    4    0    0    0    0
    2    3    2    3    6    5    0    0    0    0
    3    3    4    5    8    7    0    0    0    0
    4    3    5    6    9    8    0    0    0    0
    5    3    7    8   11   10    0    0    0    0
    6    3    8    9   12   11    0    0    0    0
    7    3   10   11   14   13    0    0    0    0
    8    3   11   12   15   14    0    0    0    0
COORDINATE
    3   15    5
    1   3.00000   0.00000   0.00000
    2   4.50000   0.00000   0.00000
    3   6.00000   0.00000   0.00000
    4   2.77200   1.14800   0.00000
    5   4.38600   2.57400   0.00000
    6   6.00000   4.00000   0.00000
    7   2.12100   2.12100   0.00000
    8   4.06000   5.06000   0.00000
    9   6.00000   8.00000   0.00000
   10   1.14800   2.77200   0.00000
   11   2.07400   5.38600   0.00000
   12   3.00000   8.00000   0.00000
   13   0.00000   3.00000   0.00000
   14   0.00000   5.50000   0.00000
   15   0.00000   8.00000   0.00000
PROPERTY
    1
 0.300E 08 0.300E 00 0.000E 00 0.000E 00 0.000E 00 0.200E 05 0.200E 05
    1    8
GEOMETRY
    1
   1.00000   0.00000   0.00000
    1    8
BOUNDARY C
    6
    1    1    2    2 0.000E 00
    2    2    2    2 0.000E 00
    3    3    2    2 0.000E 00
   13   13    1    1 0.000E 00
   14   14    1    1 0.000E 00
   15   15    1    1 0.000E 00
```

TABLE 2 MARC INPUT DATA (SAMPLE PROBLEM)
(Continued)

```
TRACTIONS
     1    1
     2    6
     6    8
-10000.0000
    15   0.00000    0.00000    0.00000
CONTROL
   100
 0.100E 00
POST
     9
     1
     2
     3
    27
    37
    11
    12
    13
    17
END OPTION
```

TABLE 3 SUMMARY OF MARC RESULTS (SAMPLE PROBLEM)

RESULTS FOR INCREMENT NO		
MAX XX STRAIN IS	-0.3556E-03	
MAX YY STRAIN IS	0.1010E-02	
MAX XY STRAIN IS	-0.4184E-03	
MAX EQUIV. PLASTIC STRAIN IS		0.0000E 00
MAX EQUIV. CREEP STRAIN IS		0.0000E 00
MAX XX STRESS IS	-0.1000E 05	
MAX YY STRESS IS	0.3157E 05	
MAX XY STRESS IS	-0.4828E 04	
MAX EQUIV. STRESS IS		0.2982E 05

TABLE 4   MARC INPUT DATA (REANALYSIS)

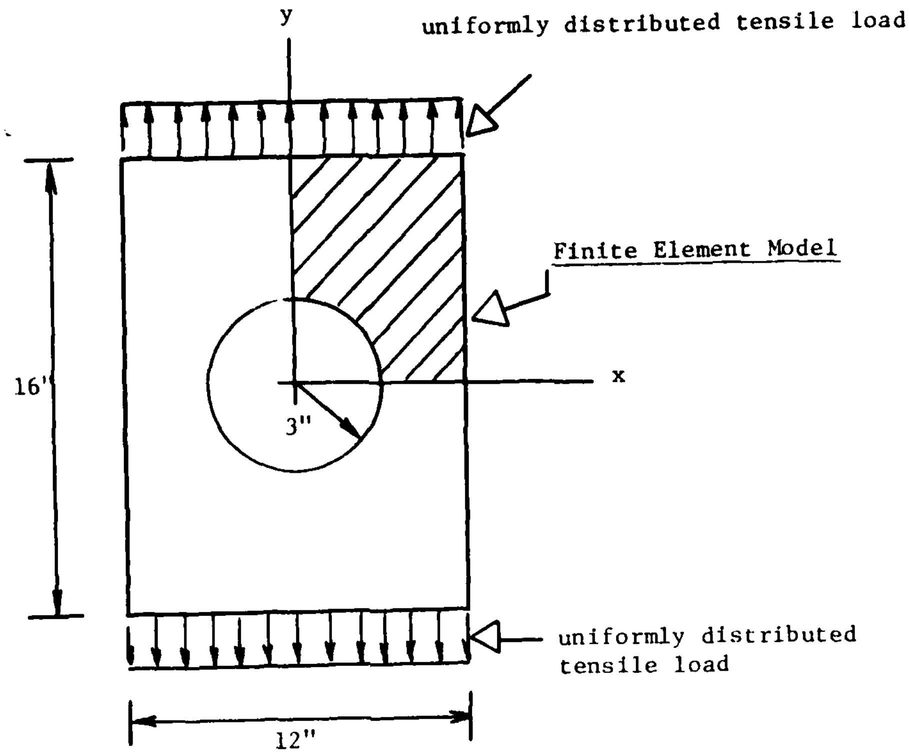
```
TITLE        TEST INPUT FOR AI
PRIME,5500024,
SIZING           10000    8   15  200   10    3
ALL POINTS
SCALE
END
CONNECTIVI
    8    5
    1    3    1    2    5    4    0    0    0    0
    2    3    2    3    6    5    0    0    0    0
    3    3    4    5    8    7    0    0    0    0
    4    3    5    6    9    8    0    0    0    0
    5    3    7    8   11   10    0    0    0    0
    6    3    8    9   12   11    0    0    0    0
    7    3   10   11   14   13    0    0    0    0
    8    3   11   12   15   14    0    0    0    0
COORDINATE
    3   15    5
    1  3.00000   0.00000   0.00000
    2  4.50000   0.00000   0.00000
    3  6.00000   0.00000   0.00000
    4  2.77200   1.14800   0.00000
    5  4.38600   2.57400   0.00000
    6  6.00000   4.00000   0.00000
    7  2.12100   2.12100   0.00000
    8  4.06000   5.06000   0.00000
    9  6.00000   8.00000   0.00000
   10  1.14800   2.77200   0.00000
   11  2.07400   5.38600   0.00000
   12  3.00000   8.00000   0.00000
   13  0.00000   3.00000   0.00000
   14  0.00000   5.50000   0.00000
   15  0.00000   8.00000   0.00000
PROPERTY
    1
 0.300E 08 0.300E 00 0.000E 00 0.000E 00 0.000E 00 0.200E 05 0.200E 05
    1    8
GEOMETRY
    1
   1.00000   0.00000   0.00000
    1    8
BOUNDARY C
    6
    1    1    2    2 0.000E 00
    2    2    2    2 0.000E 00
    3    3    2    2 0.000E 00
   13   13    1    1 0.000E 00
   14   14    1    1 0.000E 00
   15   15    1    1 0.000E 00
```

TABLE 4 MARC INPUT DATA (REANALYSIS)
(Continued)

```
TRACTIONS
    1    1
    2    6
    6    8
-10000.0000
   15  0.00000   0.00000   0.00000
CONTROL
  100
 0.100E 00
POST
    9
    1
    2
    3
   27
   37
   11
   12
   13
   17
END OPTION
PROPORTIONAL INCREMENT
    0     0.1000E 00
AUTO LOAD
    5
CONTINUE
```

TABLE 5 SUMMARY OF MARC RESULTS (REANALYSIS)

RESULTS FOR INCREMENT NO	6	
MAX XX STRAIN IS	-0.4191E-03	
MAX YY STRAIN IS	0.1683E-02	
MAX XY STRAIN IS	-0.7739E-03	
MAX EQUIV. PLASTIC STRAIN IS		0.1025E.02
MAX EQUIV. CREEP STRAIN IS		0.0000E 00
MAX XX STRESS IS	-0.1211E 05	
MAX YY STRESS IS	0.2236E 05	
MAX XY STRESS IS	0.4743E 04	
MAX EQUIV. STRESS IS		0.2002E 05



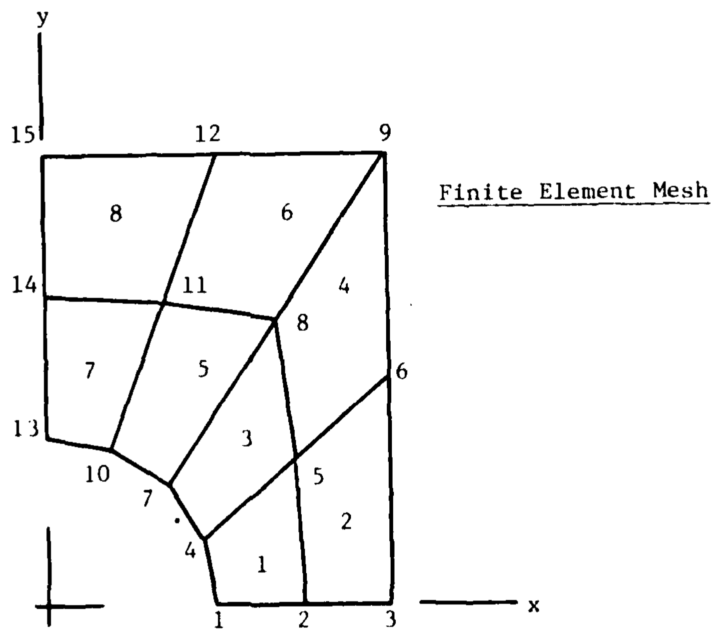


Figure 3 Plate With Hole (Sample Problem)

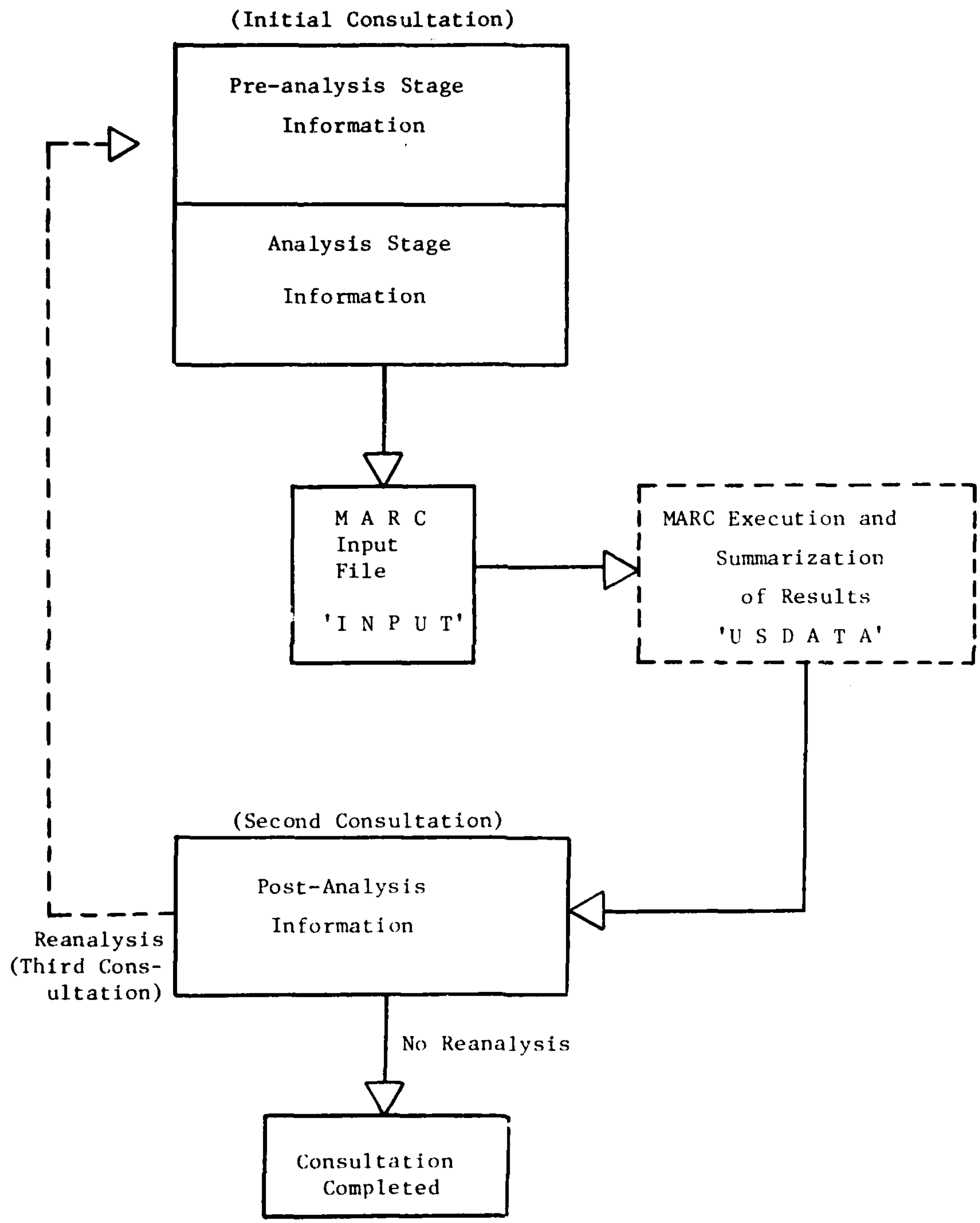


Figure 4 Schematic of Example Consultations

V. DISCUSSION AND CONCLUSIONS

The consultation system discussed in previous sections is a reasonably well designed system. The flexibility of the inference engine in the system allows the incorporation of an interface to existing software, and the modularity of the rule representation makes the rule sets easy to modify. With the help of a library of commands in the system, a rule set which represents an area of expertise can be conveniently established.

The system provides a simplified approach to the use of a general purpose finite element analysis program. It asks for information about the structure, loading, material, etc., directly from an analyst through sessions of consultations. The questions to be asked by the system during a consultation can be activated using a finite number of keywords. As a further refinement, these keywords can be pre-specified for different classes of problems. The introduction of keywords to the current system greatly reduces the user's work to find out various input formats from numerous pages of information in a Program User's Manual. In addition, the free format input available in the system further helps the user to minimize his input effort.

The current system can help an analyst to perform a reasonable structural analysis with minimized engineering effort. It recommends the type of analysis to be performed; reviews analysis results; and suggests, if necessary, additional reanalysis. As demonstrated in Section IV, after a simple question-answer session between a user and the consultation system, the necessary information for a structural analysis problem has been collected by the system. It not only makes recommendations to the user for the type of analysis to be performed, but also creates for the user an input file for the execution of the MARC program. Later, in a second consultation, after the completion of the MARC execution, the system reviews the MARC results, checks against built-in stress and strain limits, and makes additional recommendations with regard to the necessity of a reanalysis. Finally, after reviewing the results of the reanalysis, the system recommends that no further analysis is required. The advantage of using a consultation system to economize the engineer effort for a structural analysis is evident.

The current knowledge base contains a limited amount of information about structural configurations and loading environments. Recommendations given by the consultation system are, therefore, restricted to a few analyses and certain elements of the MARC program. The analyses are: elastic, elastic-plastic, creep, dynamic and large displacement. The elements are: two-node straight beam (Element No. 5), four-node quadrilaterals (Elements No 3 and No. 11), and eight-node brick (Element No. 7). Expansion of current knowledge base to include additional analyses and more elements can be easily achieved in future investigations. In addition, the inclusion of automated data generators into the system will certainly help to minimize the input efforts.

The current system behaves well. However, a continuous growth of the system is highly dependent on both the improvement of the inference engine and the expansion of the knowledge base.

REFERENCES

1. Feigenbaum. E. A., Buchanan, B. G. and Lederberg, J., "On Generality and Problem Solving: A Cast Study Using the DENDRAL Program," Machine Intelligence 6, Edinburgh University Press. 1971.

2. Shortliffe, E., "Computer-based Medical Consultations: MYCIN", New York, Elsevier, 1976.

3. Bennett, J., Creary, L., Englemore, R., and Melosh, R. J., 'SACON: A Knowledge-Based Consultant for Structural Analysis' Report No. STAN-CS-78-699, Computer Science Department, Stanford University, September, 1978.

4. Melosh, R. J., Berke, L., Marcal, P. V., "Structural Analysis Consultation Using Aritificial Intelligence", the Symposium on Future Trends in Computerized Structural Analysis and Synthesis", Joint Institute for Advancement of Flight Sciences and the National Aeronautics Space Administration, October 30, 1978.

5. MARC General Purpose Finite Element Program, User Manuals, MARC Analysis Research Corporation.

APPENDIX A

Knowledge Based Rules

For

Finite Element Structural Analysis

In this appendix, a list of knowledge based rules for finite element structural analysis and a list of questions associated with these rules are given. These rules and questions were developed for dealing with all three stages (pre-analysis, analysis and post-analysis) of information in a finite element structural analysis. The analysis is assumed to be carried out using the MARC program [5].

In current study. although the format of a knowledge based rule is uniquely defined,

```
RULE NUMBER
(PREMISE)
CONCLUSION
```

the variations of premises in a rule can provide different functions. As an illustration, various functions of the rules are discussed below:

1. Standard Evaluation

This type of rule draws conclusion on the basis of 'TRUE' or 'FALSE' of the premises in the rule. An example is RULE 1:

```
RULE 1
IF DIMENSIONALITY IS 'ONE'
IF THICKNESS IS 'THIN_SLENDER'
CONCLUDE JTYPE = 5
WRITE O :
FORMAT ('MARC ELEMENT NO 5 SELECTED FOR THE ANALYSIS')
CONCLUDE GEOMETRY_CHECK IS 'OK'
CONCLUDE GEOMETRY IS 'ONE_DIMENSIONAL_BEAM'
```

The three conclusions in RULE 1 can be successfully arrived at only if the variables DIMENSIONALITY and THICKNESS are defined as ONE and THIN_SLENDER, respectively.

2. Acquisition of Input Data

This type of rule asks the user a specific question for a piece of specific information. The user is obliged to answer the question by inputting appropriate alphabetical or numerical data. RULE 201 and RULE 202 illustrate this application.

```
RULE 201
ASK 'MARCIN01' 1 0 1 DIMENSIONALITY
CONCLUDE GENERAL_ONE IS 'GIVEN'

RULE 202
ASK 'MARCIN31' 1 0 1 NUMEL
CONCLUDE GENERAL_TWO IS 'GIVEN'
```

The variables to be defined in these rules are DIMENSIONALITY in RULE 201 and NUMEL in RULE 202. An alphabetical data and a numerical data must be supplied by the user for the variables DIMENSIONALITY and NUMEL, respectively. The questions to be answered by the user are stored in files 'MARCIN01' and 'MARCIN31' in the following form:

```
FILE 'MARCIN01'
MARCIN01GENERAL
0C15
1PLEASE ENTER DIMENSIONALITY OF YOUR
2STRUCTURE : ONE, TWO OR THREE
2THE VALUE IS ONE, TWO OR THREE

FILE 'MARCIN31'
MARCIN31GENERAL
0I
1PLEASE ENTER TOTAL NUMBER OF ELEMENTS IN THE MESH
2THE VALUE IS AN INTEGER
```

We note that the variable DIMENSIONALITY is defined as a character field (C15) and the variable NUMEL is defined as an integer field of free format (I). The integers 1 0 1 appeared in RULE 201 and RULE 202 serve as counters for the question and the variable, respectively.

IF the input data is in the form of arrays, then the counters must be modified for accommodating the situation. An example is shown in RULE 222 where the coordinates of all the nodal points in the mesh must be supplied by the user.

```
RULE 222
ASK 'COORDINA' NUMNP 0 4 IDNODE XA XB XC
CONCLUDE COORDINATES IS 'GIVEN'
```

It can be seen that the question to be answered by the user is stored in file 'COORDINA', and the nodal coordinates X , Y , Z , i = 1,2...., NUMNP are stored in variable arrays XA, XB, XC, respectively. The array IDNODE stores the nodal point identifier while the integers NUMNP 0 4 serve as counters for the question and the arrays, respectively.

The contents on file 'COORDINA' is shown below:

```
COORDINANODE
OI R R RO
1PLEASE ENTER NODAL COORDINATES AS FOLLOWS :
1NODAL NUMBER FOLLOWED BY
1  TWO NODAL COORDINATES FOR NODAL POINT OF TWO
1  DMENSONAL STRUCTURES OR
1  THREE NODAL COORDINATES FOR NODAL POINT OF THREE
1  DIMENSIONAL STRUCTURES
2THE VALUES ARE · ONE INTEGER FOR NODAL NUMBER
2FOLLOWED BY TWO REAL NUMBERS FOR NODAL POINT OF
2TWO DIMENSIONAL STRUCTURES OR. THREE REAL NUMBERS
2FOR NODAL POINT OF THREE DIMENSIONAL STRUCTURES
```

It asks for one integer (I) and three real numbers from the user NUMNP times. Both the integer and the real numbers are of free format and the third real number (Z) is optional (required for three-dimensional structures only).

3. File Manipulations

This type of rule is developed for I/O operations which are necessary for file manipulations. RULE 203 illustrates the creation of an input file and RULE 216 demonstrates the merge of a number of files into one final file.

```
RULE 203
OPEN 'INPUT1' 1  -80
WRITE 0 :
FORMAT ('SET DEFAULTS MAXALL ETC')
CONCLUDE MAXALL  = 10000
CONCLUDE NUMBC   = 200
CONCLUDE MPRMAX  = 10
CONCLUDE MESHR   = 5
CONCLUDE NCRD    = 3
CONCLUDE NBINC   = 100
CONCLUDE FACTOR  = 0.1
CONCLUDE NWKSLP  = 1
WRITE 0 :
FORMAT ('WRITE TITLE CARD ON INPUT1')
WRITE 1 :
FORMAT ('TITLE       TEST INPUT FOR AI')
WRITE 1 :
FORMAT ('PRIME,5500024,')
WRITE 0 :
FORMAT ('WRITE SIZING CARD ON INPUT1')
WRITE 1 MAXALL NUMEL NUUMNP NUMBC MPRMAX JTYPE : 1 1 1
FORMAT ('SIZING    ',I10,5I5)
CONCLUDE P_CARD[ONE IS 'GIVEN'

RULE 216
CLOSE 1
CLOSE 2
CLOSE 3
MERGE INPUT1 INPUT2 INPUT3 INPUT
CONCLUDE INPUT_FILE IS 'GIVEN'
```

In addition, Rule 251 reads the summarized MARC results from a file 'USDATA' and Rule 252 shows a summary (maximum strains and stresses) of the analysis to the user for his review.

```
RULE 251
OPEN 'USDATA' 1 40
READ 1 IR : 1 1 1
FORMAT (I5)
READ 1 VMAX : 1 9 1
FORMAT (E15.4)
WRITE O :
FORMAT ('READ SUMMARY OF MARC RESULTS FROM USDATA')
CONCLUDE AA = VMAX(1)
CONCLUDE AB = VMAX(2)
CONCLUDE AC = VMAX(3)
CONCLUDE AD = VMAX(4)
CONCLUDE AE = VMAX(5)
CONCLUDE AF = VMAX(6)
CONCLUDE AG = VMAX(7)
CONCLUDE AH = VMAX(8)
CONCLUDE AI = VMAX(9)
CONCLUDE RESULT_ONE IS 'GIVEN'
RULE 252
WRITE O IR : 1 1 1
FORMAT ('RESULTS FOR INCREMENT NO     ',I5)
WRITE O AA : 1 1 1
FORMAT ('MAX XX STRAIN IS     ',E15.4)
WRITE O AB : 1 1 1
FORMAT ('MAX YY STRAIN IS     ',E15.4)
WRITE O AC : 1 1 1
FORMAT ('MAX XY STRAIN IS     ',E15.4)
WRITE O AD : 1 1 1
FORMAT ('MAX EQUIV. PLASTIC STRAIN IS    ',E15.4)
WRITE O AE : 1 1 1
FORMAT ('MAX EQUIV. CREEP   STRAIN IS    ',E15.4)
WRITE O AF : 1 1 1
FORMAT ('MAX XX STRESS IS     ',E15.4)
WRITE O AG : 1 1 1
FORMAT ('MAX YY STRESS IS     ',E15.4)
WRITE O AH : 1 1 1
FORMAT ('MAX XY STRESS IS     ',E15.4)
WRITE O AI : 1 1 1
FORMAT ('MAX EQUIV. STRESS          IS    ',E15.4)
CONCLUDE RESULT_TWO IS 'GIVEN'
```

LIST OF RULES

```
RULE 301
WRITE O :
FORMAT ('THE FOLLOWING IS A KEY_WORD LIST')
WRITE O :
FORMAT ('GENERAL_ONE          GENERAL_TWO          GEOMETRY')
WRITE O :
FORMAT ('LOADING              MATERIAL             ANALYSIS_ONE')
WRITE O :
FORMAT ('ANALYSIS_TWO         ANALYSIS_THREE       ANALYSIS_FOUR')
WRITE O :
FORMAT ('ANALYSIS_FIVE        CONNECTIVITY         COORDINATES')
WRITE O :
FORMAT ('EMT_THICKNESS        THICKNESS_RANGE      BASE_PROPERTY')
WRITE O :
FORMAT ('MATERIAL_RANGE       BOUNDARY_C           POINT_LOAD')
WRITE O :
FORMAT ('DIST_LOAD            WORK_HARDENING       TEMP_EFFECTS')
WRITE O :
FORMAT ('P_CARD_ONE           P_CARD_TWO           P_CARD_THREE')
WRITE O :
FORMAT ('P_CARD_FOUR          M_CARD_ONE           M_CARD_TWO')
WRITE O :
FORMAT ('M_CARD_THREE         L_CARD_ONE           GEOMETRY_CHECK')
WRITE O :
FORMAT ('LOADING_CHECK        MATERIAL_CHECK       ANALYSIS_CHECK')
WRITE O :
FORMAT ('KEY_WORD_LIST        INPUT_FILE           P_CARD_FIVE')
WRITE O :
FORMAT ('L_CARD_TWO           L_CARD_THREE         L_CARD_FOUR')
WRITE O :
FORMAT ('L_CARD_FIVE')
CONCLUDE KEY_WORD_LIST IS 'DISPLACED'
RULE 201
ASK 'MARCINO1' 1 O 1 DIMENSIONALITY
ASK 'MARCINO2' 1 O 1 THICKNESS
ASK 'MARCINO3' 1 O 1 TYPE
ASK 'MARCINO4' 1 O 1 TIME
ASK 'MARCINO5' 1 O 1 TEMPERATURE
ASK 'MARCINO8' 1 O 1 STRESS_STRAIN
CONCLUDE GENERAL_ONE IS 'GIVEN'
RULE 202
ASK 'MARCIN31' 1 O 1 NUMEL
ASK 'MARCIN32' 1 O 1 NUMNP
ASK 'MARCIN38' 1 O 1 NBTHIC
ASK 'MARCIN39' 1 O 1 NTHRNG
ASK 'MARCIN33' 1 O 1 NUMMAT
ASK 'MARCIN34' 1 O 1 NBWKHD
ASK 'MARCIN35' 1 O 1 NBTEMP
ASK 'MARCIN36' 1 O 1 NMARNG
ASK 'MARCIN37' 1 O 1 NBBNDY
ASK 'MARCIN40' 1 O 1 NBPOIN
ASK 'MARCIN41' 1 O 1 NBDIST
ASK 'MARCIN42' 1 O 1 CPTIME
ASK 'MARCIN43' 1 O 1 DTIME
CONCLUDE GENERAL_TWO IS 'GIVEN'
```

```
RULE 203
OPEN 'INPUT1' 1 -80
WRITE 0 :
FORMAT ('SET DEFAULTS MAXALL ETC')
CONCLUDE MAXALL = 10000
CONCLUDE NUMBC  = 200
CONCLUDE MPRMAX = 10
CONCLUDE MESHR  = 5
CONCLUDE NCRD   = 3
CONCLUDE NBINC  = 100
CONCLUDE FACTOR = 0.1
CONCLUDE NWKSLP = 1
CONCLUDE NPOST  = 12
CONCLUDE NEA    = 1
CONCLUDE NEB    = 2
CONCLUDE NEC    = 3
CONCLUDE NED    = 4
CONCLUDE NEE    = 5
CONCLUDE NEF    = 6
CONCLUDE NSA    = 11
CONCLUDE NSB    = 12
CONCLUDE NSC    = 13
CONCLUDE NSD    = 14
CONCLUDE NSE    = 15
CONCLUDE NSF    = 16
CONCLUDE MAXCYC = 0
CONCLUDE PRATIO = 0.1
CONCLUDE NBAUTO = 10
WRITE 0 :
FORMAT ('WRITE TITLE CARD ON INPUT1')
WRITE 1 :
FORMAT ('TITLE       TEST INPUT FOR AI')
WRITE 1 :
FORMAT ('PRIME,5500024,')
WRITE 0 :
FORMAT ('WRITE SIZING CARD ON INPUT1')
WRITE 1 MAXALL NUMEL NUMNP NUMBC MPRMAX JTYPE : 1 1 1
FORMAT ('SIZING    ',I10,5I5)
WRITE 0 :
FORMAT ('WRITE ALL POINTS CARD ON INPUT1')
WRITE 1 :
FORMAT ('ALL POINTS')
CONCLUDE P_CARD_ONE IS 'GIVEN'
RULE 206
IF TYPE IS 'MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE CREEP CARD ON INPUT1')
WRITE 1 :
FORMAT ('CREEP')
CONCLUDE P_CARD_TWO IS 'GIVEN'
```

```
RULE 207
IF TYPE IS 'THERMAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE CREEP CARD ON INPUT1')
WRITE 1 :
FORMAT ('CREEP')
CONCLUDE P_CARD_TWO IS 'GIVEN'
RULE 208
IF TYPE IS 'THERMO_MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE CREEP CARD ON INPUT1')
WRITE 1 :
FORMAT ('CREEP')
CONCLUDE P_CARD_TWO IS 'GIVEN'
RULE 209
IF TYPE IS 'MECHANICAL'
IF TIME IS 'DEPENDENT'
WRITE 0 :
FORMAT ('WRITE DYNAMIC CARD ON INPUT1')
WRITE 1 :
FORMAT ('DYNAMIC,2,')
CONCLUDE P_CARD_THREE IS 'GIVEN'
RULE 210
IF THICKNESS IS 'THIN_SLENDER'
WRITE 0 :
FORMAT ('WRITE LARGE DISP CARD ON INPUT1')
WRITE 1 :
FORMAT ('LARGE DISP')
CONCLUDE P_CARD_FOUR IS 'GIVEN'
RULE 214
IF STRESS_STRAIN IS 'NON_LINEAR'
IF TIME IS 'INDEPENDENT'
IF TYPE IS 'MECHANICAL'
IF THICKNESS IS_NOT 'THIN_SLENDER'
WRITE 0 :
FORMAT ('WRITE SCALE CARD ON INPUT1')
WRITE 1 :
FORMAT ('SCALE')
CONCLUDE P_CARD_FIVE IS 'GIVEN'
```

```
RULE 211
IF STRESS_STRAIN IS 'NON_LINEAR'
WRITE 0 :
FORMAT ('WRITE WORK HARD CARD ON INPUT2')
WRITE 2 NWKSLP WKHRDA WKHRDB : 1 1 1
FORMAT ('WORK HARD'/I5/2E15.5)
CONCLUDE M_CARD_TWO IS 'GIVEN'
RULE 212
IF TYPE IS 'THERMAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE CREEP CARD ON INPUT2')
WRITE 2 :
FORMAT ('CREEP')
WRITE 2 :
FORMAT ('    0   -1    0    0       1.075E-26'/'  5.5')
CONCLUDE M_CARD_THREE IS 'GIVEN'
RULE 213
IF TYPE IS 'THERMO_MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE CREEP CARD ON INPUT2')
WRITE 2 :
FORMAT ('CREEP')
WRITE 2 :
FORMAT ('    0   -1    0    0       1.075E-26'/'  5.5')
CONCLUDE M_CARD_THREE IS 'GIVEN'
RULE 220
IF TYPE IS 'MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE CREEP CARD ON INPUT2')
WRITE 2 :
FORMAT ('CREEP')
WRITE 2 :
FORMAT ('    0   -1    0    0       1.075E-26'/'  5.5')
CONCLUDE M_CARD_THREE IS 'GIVEN'
```

```
RULE 204
OPEN 'INPUT2' 2 -80
WRITE 0 :
FORMAT ('WRITE END CARD ON INPUT2')
WRITE 2 :
FORMAT ('END')
WRITE 0 :
FORMAT ('WRITE CONNECTIVITY CARD ON INPUT2')
WRITE 2 :
FORMAT ('CONNECTIVI')
WRITE 2 NUMEL MESHR   : 1 1 1
FORMAT (2I5)
WRITE 2 IDELMT LTYPE IA IB IC ID IE IG IH II : 1 NUMEL 1
FORMAT (10I5)
WRITE 0 :
FORMAT ('WRITE COORDINATE CARD ON INPUT2')
WRITE 2 :
FORMAT ('COORDINATE')
WRITE 2 NCRD NUMNP MESHR : 1 1 1
FORMAT (3I5)
WRITE 2 IDNODE XA XB XC : 1 NUMNP 1
FORMAT (I5,3F10.5)
WRITE 0 :
FORMAT ('WRITE PROPERTY CARD ON INPUT2')
WRITE 2 :
FORMAT ('PROPERTY  ')
WRITE 2 NMARNG : 1 1 1
FORMAT (I5)
WRITE 2 YOUNGS POISON DENSIT COEFFI BASTMP YIELDS YIELDS : 1 1 1
FORMAT (7E10.3)
WRITE 2 IMRNGA IMRNGB : 1 1 1
FORMAT (2I5)
WRITE 0 :
FORMAT ('WRITE GEOMETRY CARD ON INPUT2')
WRITE 2 :
FORMAT ('GEOMETRY  ')
WRITE 2 NTHRNG : 1 1 1
FORMAT (I5)
WRITE 2 TA TB TC : 1 1 1
FORMAT (3F10.5)
WRITE 2 ITRNGA ITRNGB : 1 1 1
FORMAT (2I5)
WRITE 0 :
FORMAT ('WRITE BOUNDARY CONDITION CARD ON INPUT2')
WRITE 2 :
FORMAT ('BOUNDARY C')
WRITE 2 NBBNDY : 1 1 1
FORMAT (I5)
WRITE 2 IBRNGA IBRNGB INDDOF INDDOF SPDISP : 1 NBBNDY 1
FORMAT (4I5,E10.3)
WRITE 0 :
```

```
FORMAT ('WRITE TRACTION CARD ON INPUT2')
WRITE 2 :
FORMAT ('TRACTIONS ')
WRITE 2 NBPOIN NBDIST : 1 1 1
FORMAT (2I5)
WRITE 2 NELDIS IDFACE ILRNGB ILRNGC DISTFA : 1 NBDIST 1
FORMAT (2I5/2I5/F10.5)
WRITE 2 ILRNGA POINFA POINFB POINFC : 1 NBPOIN 1
FORMAT (I5,3F10.5)
WRITE 0 :
FORMAT ('WRITE CONTROL CARD ON INPUT2')
WRITE 2 :
FORMAT ('CONTROL   ')
WRITE 2 NBINC : 1 1 1
FORMAT (I5)
WRITE 2 FACTOR : 1 1 1
FORMAT (E10.3)
WRITE 0 :
FORMAT ('WRITE POST CARD ON INPUT2')
WRITE 2 NPOST NEA NEB NEC NED NEE NEF NSA NSB NSC NSD NSE NSF : 1 1 1
FORMAT ('POST'/I5/I5/I5/I5/I5/I5/I5/I5/I5/I5/I5/I5/I5)
CONCLUDE M_CARD_ONE IS 'GIVEN'
*
RULE 205
OPEN 'INPUT3' 3 -80
WRITE 0 :
FORMAT ('WRITE END OPTION CARD ON INPUT3')
WRITE 3 :
FORMAT ('END OPTION')
CONCLUDE L_CARD_ONE IS 'GIVEN'
RULE 215
IF STRESS_STRAIN IS 'NON_LINEAR'
IF TIME IS 'INDEPENDENT'
IF TYPE IS 'MECHANICAL'
WRITE 0 :
FORMAT ('WRITE PROPORTIONAL INC CARD ON INPUT3')
WRITE 0 :
FORMAT ('WRITE AUTO LOAD CARD ON INPUT3')
WRITE 3 MINCYC PRATIO : 1 1 1
FORMAT ('PROPORTIONAL INCREMENT'/I5,E15.4)
WRITE 3 NBAUTO : 1 1 1
FORMAT ('AUTO LOAD'/I5/'CONTINUE')
CONCLUDE L_CARD_TWO IS 'GIVEN'
RULE 200
IF TYPE IS 'MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE AUTO CREEP CARD ON INPUT3')
WRITE 3 CPTIME DTIME : 1 1 1
FORMAT ('AUTO CREEP'/2E15.4/'CONTINUE')
CONCLUDE L_CARD_THREE IS 'GIVEN'
```

```
RULE 217
IF TYPE IS 'THERMAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE AUTO CREEP CARD ON INPUT3')
WRITE 3 CPTIME DTIME : 1 1 1
FORMAT ('AUTO CREEP'/2E15.4/'CONTINUE')
CONCLUDE L_CARD_THREE IS 'GIVEN'
RULE 218
IF TYPE IS 'THERMO_MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
WRITE 0 :
FORMAT ('WRITE AUTO CREEP CARD ON INPUT3')
WRITE 3 CPTIME DTIME : 1 1 1
FORMAT ('AUTO CREEP'/2E15.4/'CONTINUE')
CONCLUDE L_CARD_THREE IS 'GIVEN'
RULE 219
IF TYPE IS 'MECHANICAL'
IF TIME IS 'DEPENDENT'
WRITE 0 :
FORMAT ('WRITE DYNAMIC CHANGE CARD ON INPUT3')
WRITE 3 CPTIME DTIME : 1 1 1
FORMAT ('DYNEMIC CHANGE'/2E15.4/'CONTINUE')
CONCLUDE L_CARD_FOUR IS 'GIVEN'
RULE 225
IF THICKNESS IS 'THIN_SLENDER'
IF STRESS_STRAIN IS 'LINEAR'
WRITE 0 :
FORMAT ('WRITE PROPORTIONAL INC CARD ON INPUT3')
WRITE 0 :
FORMAT ('WRITE AUTO LOAD CARD ON INPUT3')
WRITE 3 MINCYC PRATIO : 1 1 1
FORMAT ('PROPORTIONAL INCREMENT'/I5,E15.4)
WRITE 3 NBAUTO : 1 1 1
FORMAT ('AUTO LOAD'/I5/'CONTINUE')
CONCLUDE L_CARD FIVE IS 'GIVEN'
RULE 216
CLOSE 1
CLOSE 2
CLOSE 3
MERGE 'INPUT1' 'INPUT2' 'INPUT3' 'INPUT'
CONCLUDE INPUT_FILE IS 'GIVEN'
*
```

```
RULE 1
IF DIMENSIONALITY IS 'ONE'
IF THICKNESS IS 'THIN_SLENDER'
CONCLUDE JTYPE = 5
WRITE O :
FORMAT ('MARC ELEMENT NO 5 SELECTED FOR THE ANALYSIS')
CONCLUDE GEOMETRY_CHECK IS 'OK'
CONCLUDE GEOMETRY IS 'ONE_DIMENSIONAL_BEAM'
RULE 2
IF DIMENSIONALITY IS 'TWO'
IF THICKNESS IS 'THIN_SLENDER'
CONCLUDE JTYPE = 5
WRITE O :
FORMAT ('MARC ELEMENT NO 5 SELECTED FOR THE ANALYSIS')
CONCLUDE GEOMETRY_CHECK IS 'OK'
CONCLUDE GEOMETRY IS 'TWO_DIMENSIONAL_BEAM'
RULE 3
IF DIMENSIONALITY IS 'TWO'
IF THICKNESS IS_NOT 'THIN_FLAT'
CONCLUDE JTYPE = 11
WRITE O :
FORMAT ('MARC ELEMENT NO 11 SELECTED FOR THE ANALYSIS')
CONCLUDE GEOMETRY_CHECK IS 'OK'
CONCLUDE GEOMETRY IS 'TWO_D_PLANE_STRAIN'
RULE 101
IF DIMENSIONALITY IS 'TWO'
IF THICKNESS IS 'THIN_FLAT'
CONCLUDE JTYPE = 3
WRITE O :
FORMAT ('MARC ELEMENT NO 3 SELECTED FOR THE ANALYSIS')
CONCLUDE GEOMETRY_CHECK IS 'OK'
CONCLUDE GEOMETRY IS 'TWO_D_PLANE_STRESS'
RULE 4
IF DIMENSIONALITY IS 'THREE'
IF THICKNESS IS 'THIN_SLENDER'
WRITE O :
FORMAT ('THIS TYPE OF GEOMETRY CURRENTLY NOT ACCEPTABLE')
CONCLUDE GEOMETRY IS 'THREE_DIMENSIONAL_BEAM'
RULE 5
IF DIMENSIONALITY IS 'THREE'
IF THICKNESS IS 'THIN_CURVED'
WRITE O :
FORMAT ('THIS TYPE OF GEOMETRY CURRENTLY NOT ACCEPTABLE')
CONCLUDE GEOMETRY IS 'THREE_DIMENSIONAL_SHELL'
```

```
RULE 6
IF DIMENSIONALITY IS 'THREE'
IF THICKNESS IS 'THIN_FLAT'
WRITE 0 :
FORMAT ('THIS TYPE OF GEOMETRY CURRENTLY NOT ACCEPTABLE')
CONCLUDE GEOMETRY IS 'THREE_DIMENSIONAL_PLATE'
RULE 7
IF DIMENSIONALITY IS 'THREE'
IF THICKNESS IS 'THICK'
CONCLUDE JTYPE = 7
WRITE 0 :
FORMAT('MARC ELEMENT NO 7 SELECTED FOR THE ANALYSIS')
CONCLUDE GEOMETRY_CHECK IS 'OK'
CONCLUDE GEOMETRY IS 'THREE_DIMENSIONAL_SOLID'
RULE 102
IF GEOMETRY IS 'UNKNOWN'
WRITE 0 :
FORMAT ('THIS TYPE OF GEOMETRY CURRENTLY NOT ACCEPTABLE')
CONCLUDE GEOMETRY_CHECK IS 'NOT_OK'
RULE 8
IF TYPE IS 'MECHANICAL'
IF TIME IS 'DEPENDENT'
CONCLUDE LOADING_CHECK IS 'OK'
CONCLUDE LOADING IS 'TIME_MECHANICAL'
RULE 9
IF TYPE IS 'MECHANICAL'
IF TIME IS 'INDEPENDENT'
CONCLUDE LOADING_CHECK IS 'OK'
CONCLUDE LOADING IS 'MECHANICAL'
RULE 10
IF TYPE IS 'THERMAL'
IF TEMPERATURE IS 'LOW'
IF TIME IS 'INDEPENDENT'
CONCLUDE LOADING_CHECK IS 'OK'
CONCLUDE LOADING IS 'THERMAL'
RULE 11
IF TYPE IS 'THERMAL'
IF TEMPERATURE IS 'HIGH'
IF TIME IS 'INDEPENDENT'
CONCLUDE LOADING_CHECK IS 'OK'
CONCLUDE LOADING IS 'ELEVATED_THERMAL'
RULE 12
IF TYPE IS 'THERMO_MECHANICAL'
IF TEMPERATURE IS 'LOW'
IF TIME IS 'INDEPENDENT'
CONCLUDE LOADING_CHECK IS 'OK'
CONCLUDE LOADING IS 'THERMO_MECHANICAL'
```

```
RULE 13
IF TYPE IS 'THERMO_MECHANICAL'
IF TEMPERATURE IS 'HIGH'
IF TIME IS 'INDEPENDENT'
CONCLUDE LOADING_CHECK IS 'OK'
CONCLUDE LOADING IS 'ELEVATED_THERMO_MECHANICAL'
RULE 14
IF TYPE IS 'THERMAL'
IF TIME IS 'DEPENDENT'
WRITE O :
FORMAT ('THIS TYPE OF LOADING CURRENTLY NOT ACCEPTABLE')
CONCLUDE LOADING IS 'UNKNOWN'
RULE 15
IF TYPE IS 'THERMO_MECHANICAL'
IF TIME IS 'DEPENDENT'
WRITE O :
FORMAT ('THIS TYPE OF LOADING CURRENTLY NOT ACCEPTABLE')
CONCLUDE LOADING IS 'UNKNOWN'
RULE 103
IF LOADING IS 'UNKNOWN'
WRITE O :
FORMAT ('THIS TYPE OF LOADING CURRENTLY NOT ACCEPTABLE')
CONCLUDE LOADING_CHECK IS 'NOT OK'
RULE 18
IF STRESS_STRAIN IS 'LINEAR'
CONCLUDE MATERIAL_CHECK IS 'OK'
CONCLUDE MATERIAL IS 'LINEAR_STRESS_STRAIN'
RULE 19
IF STRESS_STRAIN IS 'NON_LINEAR'
CONCLUDE MATERIAL_CHECK IS 'OK'
CONCLUDE MATERIAL IS 'NON_LINEAR_STRESS_STRAIN'
RULE 104
IF MATERIAL IS 'UNKNOWN'
WRITE O :
FORMAT ('THIS TYPE OF MATERIAL CURRENTLY NOT ACCEPTABLE')
CONCLUDE MATERIAL_CHECK IS 'NOT OK'
RULE 105
IF ANALYSIS_ONE IS 'UNKNOWN'
CONCLUDE ANALYSIS_CHECK IS 'ANALYSIS IS NOT ELASTIC'
RULE 106
IF ANALYSIS_TWO IS 'UNKNOWN'
CONCLUDE ANALYSIS_CHECK IS 'ANALYSIS IS NOT ELASTIC_PLASTIC'
```

```
RULE 107
IF ANALYSIS_THREE IS 'UNKNOWN'
CONCLUDE ANALYSIS_CHECK IS 'ANALYSIS IS NOT CREEP'
RULE 108
IF ANALYSIS_FOUR IS 'UNKNOWN'
CONCLUDE ANALYSIS_CHECK IS 'ANALYSIS IS NOT DYNAMIC'
RULE 109
IF ANALYSIS_FIVE IS 'UNKNOWN'
CONCLUDE ANALYSIS_CHECK IS 'ANALYSIS IS NOT LARGE_DISPLACEMENT')
RULE 22
IF STRESS_STRAIN IS 'LINEAR'
CONCLUDE ANALYSIS_CHECK IS 'OK'
CONCLUDE ANALYSIS_ONE IS 'ELASTIC'
RULE 23
IF STRESS_STRAIN IS 'NON_LINEAR'
CONCLUDE ANALYSIS_CHECK IS 'OK'
CONCLUDE ANALYSIS_TWO IS 'ELASTIC_PLASTIC'
RULE 24
IF TYPE IS 'MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
CONCLUDE ANALYSIS_CHECK IS 'OK'
CONCLUDE ANALYSIS_THREE IS 'CREEP'
RULE 25
IF TYPE IS 'THERMAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
CONCLUDE ANALYSIS_CHECK IS 'OK'
CONCLUDE ANALYSIS_THREE IS 'CREEP'
RULE 26
IF TYPE IS 'THERMO_MECHANICAL'
IF TIME IS 'INDEPENDENT'
IF TEMPERATURE IS 'HIGH'
CONCLUDE ANALYSIS_CHECK IS 'OK'
CONCLUDE ANALYSIS_THREE IS 'CREEP'
RULE 27
IF TYPE IS 'MECHANICAL'
IF TIME IS 'DEPENDENT'
CONCLUDE ANALYSIS_CHECK IS 'OK'
CONCLUDE ANALYSIS_FOUR IS 'DYNAMIC'
RULE 28
IF THICKNESS IS 'THIN_SLENDER'
CONCLUDE ANALYSIS_CECK IS 'OK'
CONCLUDE ANALYSIS_FIVE IS 'LARGE_DISPLACEMENT'
RULE 221
ASK 'CONNECTI' NUMEL O 10 IDELMT LTYPE IA IB IC ID IE IG IH II
CONCLUDE CONNECTIVITY IS 'GIVEN'
RULE 222
ASK 'COORDINA' NUMNP O 4 IDNODE XA XB XC
CONCLUDE COORDINATES IS 'GIVEN'
```

```
RULE 223
ASK 'GEOMELMT' NBTHIC 0 3 TA TB TC
CONCLUDE ELMT_THICKNESS IS 'GIVEN'
RULE 224
ASK 'GEO_RANG' NTHRNG 0 3 IDTHIC ITRNGA ITRNGB
CONCLUDE THICKNESS_RANGE IS 'GIVEN'
RULE 241
ASK 'YOUNG' NUMMAT 0 2 IDMATA YOUNGS
ASK 'POISSON' NUMMAT 0 2 IDMATB POISON
ASK 'DENSITY' NUMMAT 0 2 IDMATC DENSIT
ASK 'COEFF' NUMMAT 0 2 IDMATD COEFFI
ASK 'BASETEMP' NUMMAT 0 2 IDMATE BASTMP
ASK 'YIELD' NUMMAT 0 2 IDMATF YIELDS
CONCLUDE BASE_PROPERTY IS 'GIVEN'
RULE 242
ASK 'WORKHARD' NBWKHD 0 3 IDMATG WKHRDA WKHRDB
CONCLUDE WORK_HARDENING IS 'GIVEN'
RULE 243
ASK 'YOUNTEMP' NBTEMP 0 3 IDMATH FTYOUN TTYOUN
ASK 'POISTEMP' NBTEMP 0 3 IDMATI FTPOIS TTPOIS
ASK 'COEFTEMP' NBTEMP 0 3 IDMATJ FTCOEF TTCOEF
ASK 'YIELTEMP' NBTEMP 0 3 IDMATK FTYIEL TTYIEL
CONCLUDE TEMP_EFFECTS IS 'GIVEN'
RULE 244
ASK 'MAT_RANG' NMARNG 0 3 IDMATL IMRNGA IMRNGB
CONCLUDE MATERIAL_RANGE IS 'GIVEN'
RULE 261
ASK 'BOUNDCON' NBBNDY 0 4 IBRNGA IBRNGB INDDOF SPDISP
CONCLUDE BOUNDARY_C IS 'GIVEN'
RULE 271
ASK 'POINLOAD' NBPOIN 0 4 ILRNGA POINFA POINFB POINFC
CONCLUDE POINT_LOAD IS 'GIVEN'
RULE 272
ASK 'DISTLOAD' NBDIST 0 5 NELDIS ILRNGB ILRNGC IDFACE DISTFA
CONCLUDE DIST_LOAD IS 'GIVEN'
RULE 251
OPEN 'USDATA' 1 40
READ 1 IR : 1 1 1
FORMAT (I5)
READ 1 VMAX : 1 9 1
FORMAT (E15.4)
WRITE 0 :
FORMAT ('READ SUMMARY OF MARC RESULTS FROM USDATA')
CONCLUDE SIGY = 20050.0
CONCLUDE ELIMIT = 0.05
CONCLUDE AA = VMAX(1)
CONCLUDE AB = VMAX(2)
CONCLUDE AC = VMAX(3)
CONCLUDE AD = VMAX(4)
CONCLUDE AE = VMAX(5)
CONCLUDE AF = VMAX(6)
CONCLUDE AG = VMAX(7)
CONCLUDE AH = VMAX(8)
CONCLUDE AI = VMAX(9)
CONCLUDE RESULT_ONE IS 'GIVEN'
```

```
RULE 252
WRITE O IR : 1 1 1
FORMAT ('RESULTS FOR INCREMENT NO     ',I5)
WRITE O AA : 1 1 1
FORMAT ('MAX XX STRAIN IS     ',E15.4)
WRITE O AB : 1 1 1
FORMAT ('MAX YY STRAIN IS     ',E15.4)
WRITE O AC : 1 1 1
FORMAT ('MAX XY STRAIN IS     ',E15.4)
WRITE O AD : 1 1 1
FORMAT ('MAX EQUIV. PLASTIC STRAIN IS    ',E15.4)
WRITE O AE : 1 1 1
FORMAT ('MAX EQUIV. CREEP   STRAIN IS    ',E15.4)
WRITE O AF : 1 1 1
FORMAT ('MAX XX STRESS IS     ',E15.4)
WRITE O AG : 1 1 1
FORMAT ('MAX YY STRESS IS     ',E15.4)
WRITE O AH : 1 1 1
FORMAT ('MAX XY STRESS IS     ',E15.4)
WRITE O AI : 1 1 1
FORMAT ('MAX EQUIV. STRESS          IS    ',E15.4)
CONCLUDE RESULT_TWO IS 'GIVEN'
RULE 253
IF 'SIGY' LT 'AI'
WRITE O :
FORMAT ('MAX EQUIV STRESS EXCEEDS YIELD')
WRITE O AI SIGY : 1 1 1
FORMAT ('MAX EQUIV STRESS AND YIELD STRESS ARE',2E15.4)
WRITE O :
FORMAT ('AN ELASTIC-PLASTIC ANALYSIS IS RECOMMENDED')
CONCLUDE RESULT_THREE IS 'GIVEN'
RULE 254
IF 'AD' LT 'ELIMIT'
WRITE O :
FORMAT ('MAX PLASTIC STRAIN SMALLER THAN STRAIN LIMIT 0.05')
WRITE O  AD ELIMIT : 1 1 1
FORMAT ('MAX PLASTIC STRAIN AND STRAIN LIMIT ARE',2E15.4)
WRITE O :
FORMAT ('NO FURTHER ANALYSIS IS RECOMMENDED')
CONCLUDE RESULT_FOUR IS 'GIVEN'
```

LIST OF QUESTIONS ASSOCIATED WITH RULES

```
MARCINO1GENERAL
OC15
1PLEASE ENTER DIMENSIONALITY OF YOUR
1STRUCTURE : ONE, TWO OR THREE
2THE VALUE IS ONE, TWO OR THREE
MARCINO2GENERAL
OC15
1PLEASE ENTER OVER_ALL_THICKNESS OF YOUR
1STRUCTURE: THICK,THIN_FLAT,THIN_CURVED, OR
1THIN_SLENDER
2THE VALUE IS THICK,THIN_FLAT,THIN_CURVED, OR
2THIN_SLENDER
MARCINO3GENERAL
OC20
1PLEASE ENTER TYPE OF LOADINGS : MECHANICAL,
1THERMAL, OR THERMO_MECHANICAL
2THE VALUE IS MECHANICAL, THERMAL, OR
THERMO_MECHANICAL
MARCINO4GENERAL
OC20
1PLEASE ENTER TIME DEPENDENCY OF MECHANICAL LOADINGS :
1DEPENDENT, OR INDEPENDENT
2THE VALUE IS DEPENDENT OR INDEPENDENT
MARCINO5GENERAL
OC20
1PLEASE ENTER TEMPERATURE LEVEL OF YOUR STRUCTURE
1 : HIGH OR LOW
2THE VALUE IS HIGH OR LOW
MARCINO6GENERAL
OC20
1PLEASE ENTER TYPE_OF_MATERIAL OF YOUR STRUCTURE
1: METAL OR NON_METAL
2THE VALUE IS METAL OR NON_METAL
MARCINO7GENERAL
OC20
1PLEASE ENTER DUCTILITY OF YOUR MATERIAL :
1BRITTLE OR DUCTILE
2THE VALUE IS BRITTLE OR DUCTILE
MARCINO8GENERAL
OC20
1PLEASE ENTER STRESS_STRAIN RELATION OF THE
1MATERIAL : LINEAR OR NON_LINEAR
MARCIN31GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF ELEMENTS IN THE MESH
2THE VALUE IS AN INTEGER
MARCIN32GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF NODES IN THE MESH
2THE VALUE IS AN INTEGER
```

```
MARCIN33GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF DIFFERENT MATERIALS IN
1THE MESH
2THE VALUE IS AN INTEGER
MARCIN34GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF POINTS IN WORK_HARDENING
1SLOPE FUNCTIONS FOR ALL MATERIALS
2THE VALUE IS AN INTEGER
MARCIN35GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF POINTS IN TEMPERATURE
1DEPENDENT MATERIAL FUNCTIONS FOR ALL MATERIALS
1(MAX. VALUE AMONG ALL PROPERTIES)
2THE VALUE IS AN INTEGER
MARCIN36GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF ELEMENT_RANGE PAIRS
1FOR ALL MATERIALS IN THE MESH
2THE VALUE IS AN INTEGER
MARCIN37GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF SPECIFIED NODAL POINT
1DEGREES_OF_FREEDOMS IN THE MESH
2THE VALUE IS AN INTEGER
MARCIN38GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF DIFFERENT ELEMENT
1THICKNESSES IN THE MESH
2THE VALUE IS AN INTEGER
MARCIN39GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF ELEMENT_RANGE PAIRS
1OF ALL ELEMENT THICKNESSES IN THE MESH
2THE VALUE IS AN INTEGER
MARCIN40GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF POINT LOADS (NODE_RANGE
1PAIRS) IN THE MESH
2THE VALUE IS AN INTEGER
MARCIN41GENERAL
OI
1PLEASE ENTER TOTAL NUMBER OF DISTRIBUTED LOADS (ELEMENT_
1RANGE PAIRS) IN THE MESH
2THE VALUE IS AN INTEGER
```

```
MARCIN42GENERAL
OR
1PLEASE ENTER TRANSIENT TIME FOR DYNAMIC OR CREEP ANALYSIS
2THE VALUE IS A REAL NUMBER
MARCIN43GENERAL
OR
1PLEASE ENTER TIME INCREMENT FOR DYNAMIC OF CREEP ANALYSIS
2THE VALUE IF A REAL NUMBER
CONNECTIELMT
OI I I I IO IO IO IO IO IO
1PLEASE ENTER ELEMENT CONNECTIVITY AS FOLLOWS:
1ELEMENT NUMBER AND ELEMENT TYPE  FOLLOWED BY
1  TWO NODAL NUMBERS FOR TWO_NODE BEAM AND AXISYMMETRIC
1  SHELL ELEMENTS OR,
1  FOUR NODAL NUMBERS FOR FOUR_NODE PLANE STRESS,PLANE
1  STRAIN, AND AXISYMMETRIC ELEMENTS OR,
1  EIGHT NODAL NUMBERS FOR EIGHT_NODE THREE DIMENSIONAL
1  BRICK ELEMENTS
2THE VALUES ARE : ONE INTEGER FOR ELEMENT NUMBER FOLLOWED BY
2ONE INTEGER FOR ELEMENT TYPE AND
2  TWO INTEGERS FOR BEAM AND AXISYMMETRIC SHELL
2ELEMENTS OR, FOUR INTEGERS FOR PLANE STRESS, PLANE STRAIN,
2AND AXISYMMETRIC ELEMENTS OR, EIGHT INTEGERS FOR THREE
2DIMENSIONAL BRICK ELEMENTS
COORDINANODE
OI R R RO
1PLEASE ENTER NODAL COORDINATES AS FOLLOWS :
1NODAL NUMBER FOLLOWED BY
1  TWO NODAL COORDINATES FOR NODAL POINT OF TWO
1  DMENSIONAL STRUCTURES OR,
1  THREE NODAL COORDINATES FOR NODAL POINT OF THREE
1  DIMENSIONAL STRUCTURES
2THE VALUES ARE : ONE INTEGER FOR NODAL NUMBER
2FOLLOWED BY TWO REAL NUMBERS FOR NODAL POINT OF
2TWO DIMENSIONAL STRUCTURES OR, THREE REAL NUMBERS
2FOR NODAL POINT OF THREE DIMENSIONAL STRUCTURES
GEOMELMTTHICKNES
OR RO RO
1PLEASE ENTER GEOMETRY DATA OR, ELEMENT THICKNESS AS
1FOLLOWS :
1  GEOMETRY DATA FOR TWO_NODE BEAM AND AXISYMMETRIC SHELL
1  ELEMENTS OR,
1  ELEMENT THICKNESS FOR PLANE STRESS AND PLANE STRAIN
1  ELEMENTS
2THE VALUES ARE : TWO REAL NUMBERS FOR TWO_NODE BEAM
2AND AXISYMMETRIC SHELL ELEMENTS OR, ONE REAL NUMBER
2FOR PLANE STRESS AND PLANE STRAIN ELEMENTS
GEO_RANGEL_RANGE
OI I I
1PLEASE ENTER ELEMENT THICKNESS IDENTIFICATION AND
1ELEMENT_RANGE PAIR FOR GIVEN THICKNESS
2THE VALUES ARE THREE INTEGERS
```

```
YOUNG   MATERIAL
OI R
1PLEASE ENTER MATERIAL IDENTIFICATION AND YOUNGS_
1MODULUS OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND A REAL NUMBER
POISSON MATERIAL
OI R
1PLEASE ENTER MATERIAL IDENTIFICATION AND POISSONS_
1RATIO OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND A REAL NUMBER
DENSITY MATERIAL
OI R
1PLEASE ENTER MATERIAL IDENTIFICATION AND DENSITY
1OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND A REAL NUMBER
COEFF   MATERIAL
OI R
1PLEASE ENTER MATERIAL IDENTIFICATION AND COEFF_
1OF_THERMAL_EXPANSION OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND A REAL NUMBER
BASETEMPMATERIAL
OI R
1PLEASE ENTER MATERIAL IDENTIFICATION AND BASE
1(STRESS_FREE) TEMPERATURE OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND A REAL NUMBER
YIELD   MATERIAL
OI R
1PLEASE ENTER MATERIAL IDENTIFICATION AND YIELD_
1STRESS OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND A REAL NUMBER
WORKHARDSIG-EPSP
OI R R
1PLEASE ENTER MATERIAL IDENTIFICATION AND EFFECTIVE_
1STRESS VS EFFECTIVE_PLASTIC_STRAIN OF ALL MATERIALS
2THER VALUES ARE ONE INTEGER AND TWO REAL NUMBERS
YOUNTEMPYOUNG- T
OI R R
1PLEASE ENTER MATERIAL IDENTIFICATION AND YOUNGS_
1MODULUS VS TEMPERATURE OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND TWO REAL NUMBERS
POISTEMPPOISS -T
OI R R
1PLEASE ENTER MATERIAL IDENTIFICATION AND POISSONS_
1RATIO VS TEMPERATURE OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND TWO REAL NUMBERS
COEFTEMPCOEFF -T
OI R R
1PLEASE ENTER MATERIAL IDENTIFICATION AND COEFF_
1OF THERMAL_EXPANSION VS TEMPERATURE OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND TWO REAL NUMBERS
```

```
YIELTEMPYIELD -T
OI R R
1PLEASE ENTER MATERIAL IDENTIFCATION AND YIELD_
1STRESS VS TEMPERATURE OF ALL MATERIALS
2THE VALUES ARE ONE INTEGER AND TWO REAL NUMBERS
MAT_RANGEL RANGE
OI I I
1PLEASE ENTER MATERIAL IDENTIFICATION AND ELEMENT_
1RANGE PAIRS OF ALL MATERIALS
2THE VALUES ARE THREE INTEGERS
BOUNDCONBOUNDARY
OI I I RO
1PLEASE ENTER NODAL POINT RANGE, DEGREES_OF_FREEDOM,
1AND SPECIFIED DISPLACEMENT(OPTIONAL,DEFAULT IS 0.0)
1
2THE VALUES ARE THREE INTEGERS AND ONE OPTIONAL REAL
2NUMBER
POINLOADNODE
OI R R R
1PLEASE ENTER NODE NO.,POINT LOAD IN FIRST DIRECTION,
1POINT LOAD IN SECOND DIRECTION AND POINT LOAD IN THIRD DIRECTION
2THE VALUES ARE TWO INTEGERS AND THREE REAL NUMBERS
DISTLOADELMT
OI I I I R
1PLEASE ENTER NUMBER OF ELEMENTS,
1ELEMENT RANGE,FACE IDENTIFICATION OF THE
1DISTRIBUTED LOAD AND DISTRIBUTED LOAD
2THE VALUES ARE FOUR INTEGERS AND ONE REAL NUMBER
```

APPENDIX B

The MARC Program

And

MARC Elements 3, 5, 7, 11

The MARC program is a general purpose finite element structural analysis program [5]. Possible uses of the MARC system include the following: Linear elastic analysis of two- and three-dimensional solids, shells, and beams; and applications where nonlinear material and geometric effects are needed for geometric modeling. Both the linear and nonlinear analysis can be carried out in both the static and dynamic regimes for stress analysis, and also for heat transfer (diffusion) analysis. Mesh generators, graphics, and post-processing aid the user in the preparation of input and the interpretation of results.

The program derives its broad applicability by allowing the user to select from three comprehensive libraries for elements, materials and structural procedures, respectively. The Element Library contains over 60 elements, which allow the user to describe any geometry that may be encountered. The Material Library contains over 35 different material models covering the behavior of most engineering materials through its various modules. This allows simulation of particular physical phenomena, such as temperature cycling, buckling, dynamic transient, etc. The Structural Procedures Library contains about 15 structural procedures.

MARC allows the user to combine any number of components from each of the three libraries and, in doing so, puts the tools to solve almost any structural mechanics problem at the disposal of the user.

The input data for MARC is organized into three basic groups. These groups form a natural subdivision of the data. Each group is then subdivided into various optional blocks of input data. The optional blocks of data within each group have been organized to minimize the input of unnecessary data.

The main idea is to enable the user to specify only the data for the optional blocks that he needs to define his problem. The various blocks of input are referred to here as optional in the sense that many have built-in default values which may be used and is not meant to imply that they are optional in all cases. The input data is divided into the following three groups:

PARAMETER CARDS (P_CARD)

This group of cards is used to allocate the necessary working space for the problem and to set up initial switches which control the flow of the program through the desired analysis. This set of input cards is terminated with an END card.

MODEL DEFINITION CARDS (M_CARD)

This set of data cards is used to read in the initial loading, geometry and material data of the model. It also provides nodal point data such as boundary conditions. In general, the initial model data is provided in this group and control, restart and plot options may also be specified here for further program processing. This data provides the program with the necessary information for determination of an initial elastic solution (zero increment solution in program terminology). This group of cards is terminated with an END OPTION card.

LOAD INCREMENTATION CARDS (L_CARD)

This group of cards provide the load incrementation and control of the program after the initial elastic analysis. The group also includes blocks which allow changes in the initial model specifications. Each set of control cards is terminated with a CONTINUE card. This card sends the program back for another increment or series of increments if the auto incrementation features are requested.

Figure B-1 shows the input deck set up and keywords for P_CARD, M_CARD and L_CARD are listed in Tables B-1, B-2 and B-3, respectively.

TABLE B-1 P_CARD KEYWORDS

KEY WORD	KEY WORD	KEY WORD
ACCUM BUC	FOLLOW FOR	PRINT
ALIAS	FORCDT	QUALIFY
ALL POINTS	FOURIER	R-P FLOW
ANISOTROPI	HEAT	RESTART
APPBC	HYPOELAS	SAVE ELEM
BEAM SECT	INPUT TAPE	SCALE
BUCKLE	ISTRESS	SHELL SECT
CENT	J-INT	SHELL TRAN
COMBINED	KINEMATIC	SIZING
CRACKING	LARGE DISP	SPRINGS
CREEP	LOAD COR	STATE VARS
DIST LOADS	LUMP	STOP
DYNAMIC	MASS POINT	TAPES
ECS	MATERIAL	THERMAL
EDIT	MOHRC	TIE
ELAS FOUND	MOONEY	TITLE
ELASTIC	MESH PLOT	TRANSFORM
ELSTO	NO LOADCOR	UPDATE
END	NOTES	VECSTO
FILMS	ORNL	VECSTO ELAS
FLU LOAD	POST	
FLUXES		

TABLE B-2 M_CARD KEYWORDS

KEY WORD	KEY WORD	KEY WORD
BC FILL	INIT FILS	PRINT CHOICE
BC GENER	INIT GENER	PROPERTY
BOUNDARY CONDITIONS	INITIAL CONDITIONS	REAUTO
CASE COMBIN	INITIAL STATE	REGEN
CHANGE STATE	J-INTEGRAL	RENUMBER
CONN FILL	MASS FILL	RESTART
CONN GENER	MASS GENER	ROTATION A
CONNECTIVITY	MASSES	SERVO LINK
CONTROL	MESH 2D	SHELL TRANSFORMATION
COORDINATES	BLOCKS	SPRINGS
CRACKING	DEFINE	STRAIN RATE
CREEP	BOUNDARY	TEMPERATURE EFFECTS
DAMPING	CONSTRAINT	THERMAL LOADS
END OPTION	MAPPER	TIE FILL
FLUID SOLID	MERGE	TIE GENER
FILMS	SYMMETRY	TRAC FILL
FLUX FILL	SPECIFIED MODES	TRAC GENER
FLUX GENER	GENERATE	TRACTIONS
FLUXES	NODE CIRCLE	TRANS FILL
FORCDT	NODE FILL	TRANS GENER
FOUNDATION	NODE GENER	TRANSFORMATION
FOURIER	NODE MERGE	TYING
FXORD	OPTIMIZE	UFCONN
GEOMETRY	POST	UFXORD
		WORK HARD

TABLE B-3 L_CARD KEYWORDS

KEY WORD
AUTO CREEP
AUTO LOAD
AUTO THERM
BOUNDARY CHANGE
BUCKLE
CHANGE STATE
CONTINUE
CREEP EXT
CREEP INCREMENT
DYNAMIC CHANGE
FLUXES
FLUX FILL
FLUX GENER
MODEL SHAPE
PRINT CHOICE
PROPORTIONAL INCREMENT
THERMAL LOADS
TRAC FILL
TRAC GENER
TRACTIONS
TYING CHANGE

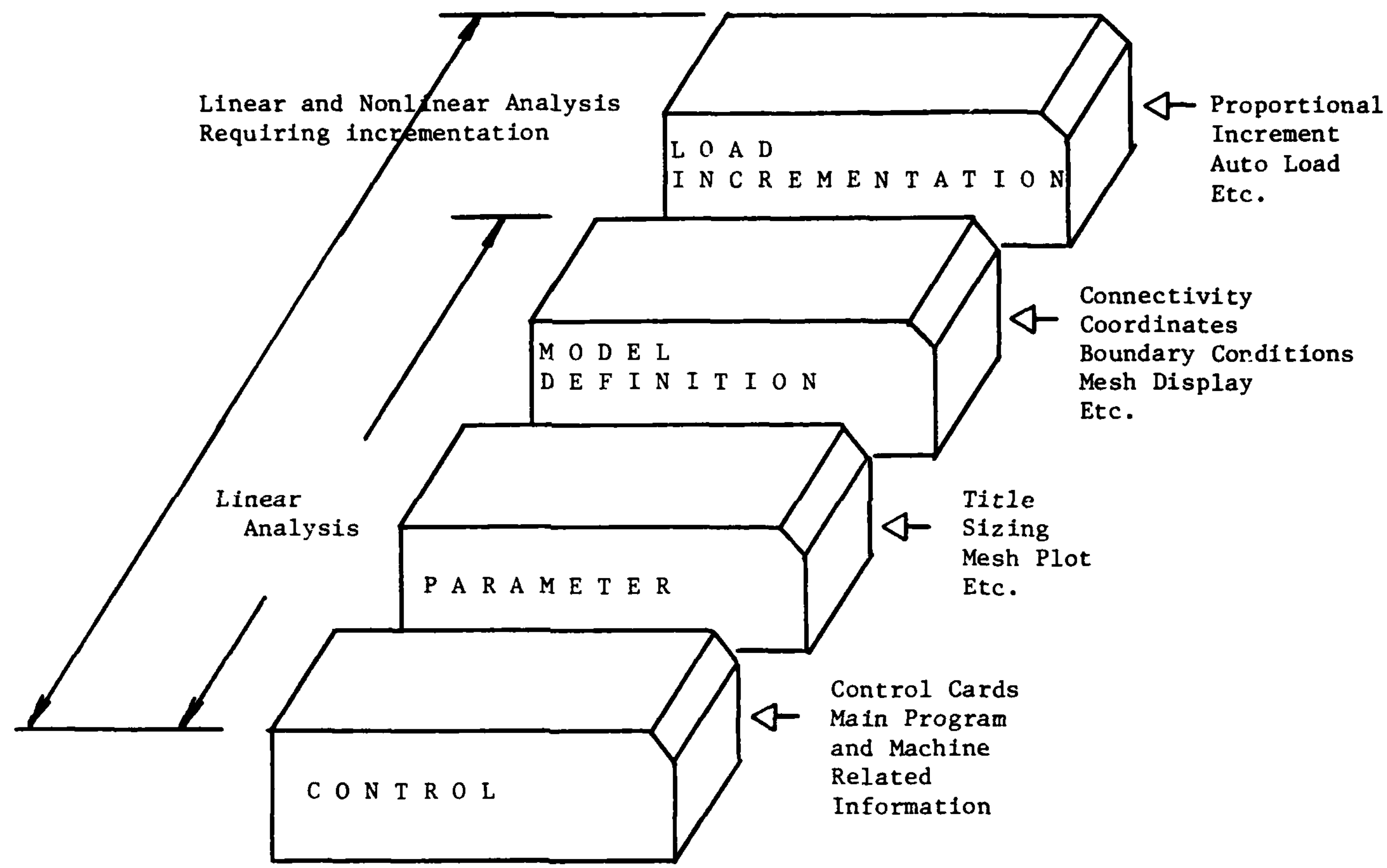


Figure B-1 MARC Input Deck

BEAM COLUMN -- ELEMENT 5

This element (Figure B-2) is a straight, two-node, rectangular-section beam-column element using linear interpolation parallel to its length, cubic interpolation in the normal direction. (Element 16 is a full cubic beam that is more accurate for many applications).

The degrees of freedom are the u and v displacements, and the right-handed rotation at the two end points of the element. The element stiffness is formed by numerical integration: the strain-stress transformation is formed by a Simpson rule integration using 11 points through the thickness of the element. The stiffness is formed by Gaussian integration along the length of the element using 3 points.

QUICK REFERENCE

- TYPE 5

  Two-dimensional, rectangular-section, beam-column.

- CONNECTIVITY

  Two nodes per element.

- GEOMETRY

  A rectangular section is assumed. The height is input in the first data field (EGEOM1).

  The cross-sectional area in the second data field (EGEOM2)

  The third data field is not used.

- COORDINATES

  Two coordinates in the global x and y directions.

- DEGREES OF FREEDOM

  1 = u displacement

  2 = v displacement

  3 = right-handed rotation

- TRACTIONS

  Pressure is assumed positive in the local y direction. Since this is the only type of distributed load for this element, no specification of load type need be made.

- OUTPUT OF STRAINS

  Two generalized strains, membrane stretching and bending.

- OUTPUT OF STRESSES

  Stresses at each integration point or the centroid. Each of these points will have points equally spaced through its thickness. The stress will be output at each representative point.

  The first point is on the positive normal (positive local y) face, the last on the negative face.

- TRANSFORMATION

  Standard transformation will transform first two global degrees of freedom to local degrees of freedom.

- TYING

  No standard tying available, use UFORMS.

- OUTPUT POINTS

  Centroid or three Gaussian integration points if ALL POINTS used.

- MESH GENERATOR

  None available.

- SECTION STRESS INTEGRATION

  Integration through the thickness is performed numerically using Simpson's rule. Use the SHELL SECT parameter card to define the number of integration points -- this number must be odd. 3 points are enough for linear material response -- 7 points for simple plasticity or creep analysis; 11 points for complex plasticity or creep (e.g., dynamic plasticity). The default is 11 points.

- UPDATING LAGRANGE PROCEDURE AND FINITE STRAIN PLASTICITY

  Capability is not available -- use Element 16 instead.

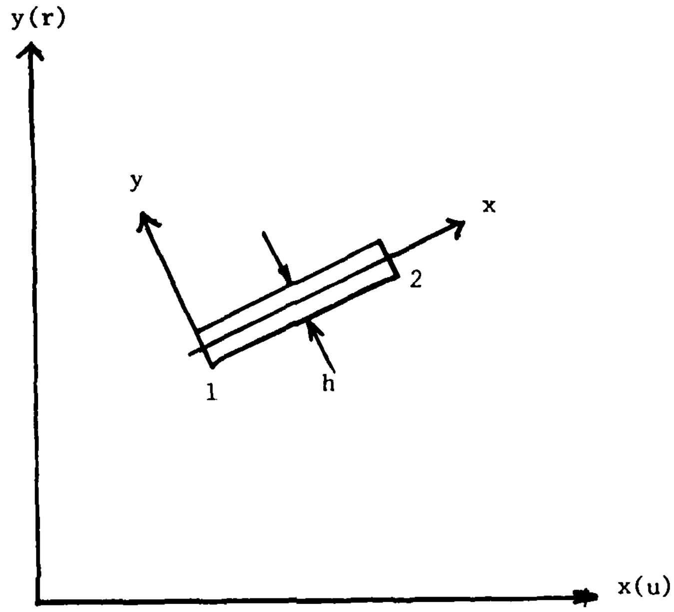


Figure B-2 Beam-Column Element

PLANE STRESS QUADRILATERAL -- ELEMENT 3

This element (Figure B-3) is a four-node isoparametric arbitrary quadrilateral element (Element 26 is the better element, use it if possible). The element is formed by a mapping from the (x-y) plane to a (g-h) plane. Both the mapping and the displacement assumption take the typical form:

$$x = a_0 + a_1 g + a_2 h + a_3 gh$$

$$u = b_0 + b_1 g + b_2 h + b_3 gh$$

and so forth.

The stiffness of the element is formed by numerical integration using four points as shown in Figure B-3. See Reference 3 in Volume A, Section 7, for further details on this element.

QUICK REFERENCE

- TYPE 3

  Plane stress quadrilateral.

- CONNECTIVITY

  Node numbering must follow right-handed convention.

- GEOMETRY

  The thickness is stored in the first data field (EGEOM1). The other two data fields (EGEOM2 and EGEOM3) are not used for this element. Default thickness is 1.

- COORDINATES

  Two global coordinates x and y.

  DEGREES OF FREEDOM

  1 = u (displacement in the global x direction)

  2 = v (displacement in the global y direction)

- TRACTIONS

  Load types for distributed loads are defined as follows:

Load Type	Description
0	Uniform pressure distributed on 1-2 face of the element.
1	Uniform body force per unit volume in first coordinate direction.
2	Uniform body force by unit volume in second coordinate direction.
3	Non-uniform pressure on 1-2 face of the element, magnitude supplied through subroutine FORCEM.
4	Non-uniform body force per unit volume in first coordinate direction, magnitude supplied through subroutine FORCEM.
5	Non-uniform body force per unit volume in second coordinate direction, magnitude supplied through subroutine FORCEM.
6	Uniform pressure on 2-3 face of the element.
7	Non-uniform pressure on 2-3 face of the element, magnitude supplied through subroutine FORCEM.
8	Uniform pressure on 3-4 face of the element.
9	Non-uniform pressure on 3-4 face of the element, magnitude supplied through subroutine FORCEM.
10	Uniform pressure on 4-1 face of the element.
11	Non-uniform pressure on 4-1 face of the element, magnitude supplied through subroutine FORCEM.

All pressures are positive when directed into the element. In addition, point loads may be applied at the nodes.

. OUTPUT OF STRAINS

Output of strain at the centroid of the element is:

$1 = E_{xx}$

$2 = E_{yy}$

$3 = E_{xy}$

- OUTPUT OF STRESSES

  Output of stress is the same as for the strain.

- TRANSFORMATION

  The two global degrees of freedom may be transformed to local coordinates.

- TYING

  User UFORMS subroutine.

- OUTPUT POINTS

  Output is available at the centroid or with the ALL POINTS option, at the four numerical integration points shown in Figure B-3.

- MESH GENERATOR

  Use MESH2D option.

- UPDATING LAGRANGE PROCEDURE AND FINITE STRAIN PLASTICITY

  Capability is available - output of stress and strain in global coordinate directions. Thickness will be updated.

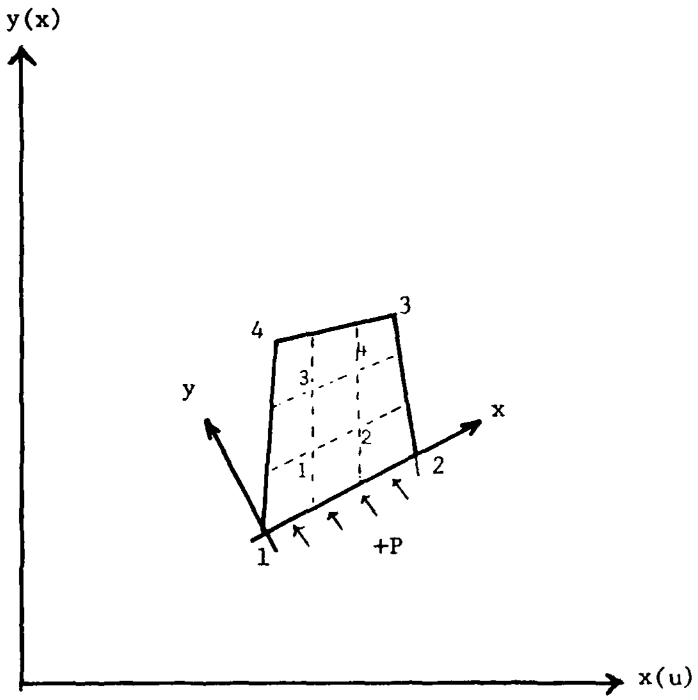


Figure B-3   Plane Stress Quadrilateral

ARBITRARY QUADRILATERAL PLANE STRAIN - ELEMENT 11

This is the same formulation as element 3 and 10, written for plane-strain (Element 27 is the better element, use it if possible).

QUICK REFERENCE

- TYPE 11

  Plane strain quadrilateral.

- CONNECTIVITY

  Four nodes per element. Node numbering follows right-handed convention (counterclockwise).

- GEOMETRY

  The thickness is entered in the first data field (EGEOM1). Default the unit thickness. If a nonzero value is entered in the second data field (EGEOM2) the volume strain will be constant throughout the element. That is particularly useful for analysis of approximately incompressible materials and for analysis of structures in the fully plastic range. It is also recommended for creep problems in which it is attempted to obtain the steady solution. For details, see Volume G, Reference XXV.

- COORDINATES

  Two coordinates in the global x and y directions.

- DEGREES OF FREEDOM

  Global displacement degrees of freedom:

  1 = z displacement (along symmetry axis)

  2 = radial displacement

- TRACTIONS

  Load types for distributed loads as follows:

Load Type	Description
0	Uniform pressure distributed on 1-2 face of the element.
1	Uniform body force per unit volume in first coordinate direction.
2	Uniform body force by unit volume in second coordinate direction.
3	Non-uniform pressure on 1-2 face of the element, magnitude supplied through subroutine FORCEM.
4	Non-uniform body force per unit volume in first coordinate direction, magnitude supplied through subroutine FORCEM.
5	Non-uniform body force per unit volume in second coordinate direction, magnitude supplied through subroutine FORCEM.
6	Uniform pressure on 2-3 face of the element.
7	Non-uniform pressure on 2-3 face of the element, magnitude supplied through subroutine FORCEM.
8	Uniform pressure on 3-4 face of the element.
9	Non-uniform pressure on 3-4 face of the element, magnitude supplied through subroutine FORCEM.
10	Uniform pressure on 4-1 face of the element.
11	Non-uniform pressure on 4-1 face of the element, magnitude supplied through subroutine FORCEM.

All pressures are positive when directed into the element. In addition, point loads may be applied at the nodes.

- OUTPUT OF STRAINS

Output of strains at the centroid of the element in global coordinates is:

1 = E_{zz}

2 = E_{rr}

3 = $E_{\theta\theta}$

4 = E_{rz}

- OUTPUT OF STRESSES

  Same as for strain.

- TRANSFORMATION

  Two global degrees of freedom may be transformed into local coordinates.

- TYING

  May be tied to axisymmetric shell type 1 using standard tying type 23.

- OUTPUT POINTS

  Output is available at the centroid or at the 4 Gaussian points shown in Figure B-4.

- MESH GENERATOR

  MESH2D Option.

- UPDATING LAGRANGE PROCEDURE AND FINITE STRAIN PLASTICITY

  Capability is available - stress and strain output in global coordinate directions. Reduced volume strain integration recommended. (See GEOMETRY).

- DEGREES OF FREEDOM

  Global displacement degrees of freedom:

  1 = u direction

  2 = v direction

- TRACTIONS

  Load types for distributed loads as follows·

AD-A095 146 MARC ANALYSIS RESEARCH CORP PALO ALTO CA F/G 13/13
KNOWLEDGE BASED CONSULTATION FOR FINITE ELEMENT STRUCTURAL ANAL--ETC(U)
MAY 80 J M RIVLIN, M B HSU, P V MARCAL F33615-78-C-3206
UNCLASSIFIED AFWAL-TR-80-3069 NL

2 OF 2
AD A
095146

END
DATE
FILMED
3-81
DTIC

Load Type	Description
0	Uniform pressure distributed on 1-2 face of the element.
1	Uniform body force per unit volume in first coordinate direction.
2	Uniform body force by unit volume in second coordinate direction.
3	Non-uniform pressure on 1-2 face of the element, magnitude supplied through subroutine FORCEM.
4	Non-uniform body force per unit volume in first coordinate direction, magnitude supplied through subroutine FORCEM.
5	Non-uniform body force per unit volume in second coordinate direction, magnitude supplied through subroutine FORCEM.
6	Uniform pressure on 2-3 face of the element.
7	Non-uniform pressure on 2-3 face of the element. magnitude supplied through subroutine FORCEM.
8	Uniform pressure on 3-4 face of the element.
9	Non-uniform pressure on 3-4 face of the element, magnitude supplied through subroutine FORCEM.
10	Uniform pressure on 4-1 face of the element.
11	Non-uniform pressure on 4-1 face of the element, magnitude supplied through subroutine FORCEM.

All pressures are positive when directed into the element. In addition, point loads may be applied at the nodes.

- OUTPUT OF STRAINS

Output of strain at the centroid of the element is:

$1 = E_{xx}$

$2 = E_{yy}$

$3 = E_{zz} = 0$

$4 = E_{xy}$

- OUTPUT OF STRESSES

Same as for strain.

- TRANSFORMATION

The two global degrees of freedom may be transformed to local degrees of freedom.

- TYING

User subroutine UFORMS.

- OUTPUT POINTS

Output is available at the centroid, or with the ALL POINT Option, at the 4 Gaussian points shown in Figure B-4.

- MESH GENERATOR

MESH2D Option.

- UPDATING LAGRANGE PROCEDURE AND FINITE STRAIN PLASTICITY

Capability is available - stress and strain output in global coordinate direction. Reduced volume strain integration recommended. (See GEOMETRY).

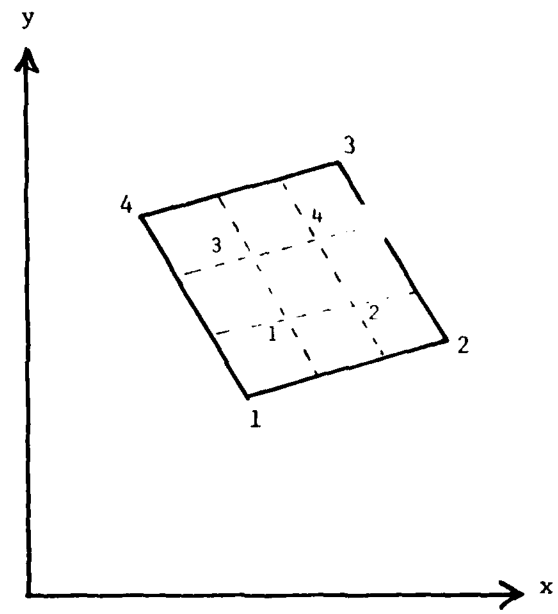


Figure B-4 Gaussian Integration Points for Element Type 11

THREE-DIMENSIONAL ARBITRARILY DISTORTED CUBE - ELEMENT 7

This is an *eight-node isoparametric element* (*Figure* B-5). Element 21 is a better element. Use it if possible. The element is based on the following type of displacement assumption and mapping into a cube in the (g-h-r) space:

$$x = a_0 + a_1 g + a_2 h + a_3 r + a_4 gh + a_5 hr + a_6 gr + a_7 ghr$$

$$u = b_0 + b_1 g + b_2 h + b_3 r + b_4 gh + b_5 hr + b_6 gr + b_7 ghr$$

The 24 generalized displacements are related to the u-v-w displacements (in global coordinates) at the eight corners of the distorted cube. The strain-displacement relations for this element are written for finite strain, large displacement analysis. The stiffness of the element is formed by numerical integration using eight points defined in the (g-h-r) space.

See Reference 6, Volume A, Section 7, for further details.

QUICK REFERENCE

- TYPE 7

  Eight-node, three-dimensional, first-order isoparametric element (arbitrarily distorted cube).

- CONNECTIVITY

  Eight nodes per element. Node numbering must follow the scheme below:

  Nodes 1, 2, 3 and 4 are corners of one face, given in counterclockwise order when viewed from inside the element. Node 5 is the same edge as node 1, node 6 as node 2, node 7 as node 3, and node 8 as node 4.

- GEOMETRY

  If a nonzero value is entered in the second data field (EGEOM2), the volume strain will be constant throughout the element. This is particularly useful for analysis of approximately incompressible materials, and for analysis of structures in the fully plastic range. And it is also recommended for creep problems in which it is attempted to obtain the steady state soltion. For details see Volume G, Reference XXV.

- COORDINATES

  Three coordinates in the global x, y and z directions.

- DEGREES OF FREEDOM

Three global degrees of freedom u, v and w per node.

- TRACTIONS

Distributed loads chosen by value of IBODY as:

IBODY = 0	Uniform pressure on 1-2-3-4 face.
IBODY = 1	Non-uniform pressure on 1-2-3-4 face with magnitude supplied by user subroutine FORCEM.
IBODY = 2	Uniform body force per unit volume in -z direction.
IBODY = 3	Non-uniform body force per unit volume (e.g., centrifugal force) with magnitude and direction supplied through user subroutine FORCEM.
IBODY = 4	Uniform pressure on 6-5-7-8 face.
IBODY = 5	Non-uniform presure on 6-5-8-7 face (FORCEM).
IBODY = 6	Uniform pressure on 2-1-5-6 face.
IBODY = 7	Non-uniform pressure on 2-1-5-6 face (FORCEM).
IBODY = 8	Uniform pressure on 3-2-6-7 face.
IBODY = 9	Non-uniform pressure on 3-2-6-7 face (FORCEM).
IBODY = 10	Uniform pressure on 4-3-7-8 face.
IBODY = 11	Non-uniform pressure on 4-3-7-8 face (FORCEM).
IBODY = 12	Uniform pressure on 1-4-8-5 face.
IBODY = 13	Non-uniform pressure on 1-4-8-5 face (FORCEM).

Pressure forces are positive into element face. Note that for use with this element, FORCEM has the following form:

```
SUBROUTINE FORCEM (P,XYZ,DIR,NN,N)
DIMENSION XYZ (3), DIR(3)

     User Coding

RETURN
END
```

Where:

XYZ(1), XYZ(2), XYZ(3)	are the (x,y,z) coordinates of the integration point (supplied by the program).
DIR(1), DIR(2), DIR(3)	are the (x,y,z) components defining the direction of the force. This array must be be defined by the user for IBODY=3, but otherwise, ignored.
P	is the magnitude of the force (per unit volume) for IBODY=3, or the magnitude of the non-uniform surface pressure that is defined by the user.
NN	is the integration point number on the surface to which the load is applied (1 through 4), or in volume (1 through 8).
N	is the element number.

This subroutine will be called once per integration point when flagged. The magnitude of load defined by TRACTIONS will be ignored and the FORCEM value will be used instead. For non-uniform surface pressures, values need only be supplied for the four integration points on the face of application. Note that XYZ and N must not be changed.

- OUTPUT OF STRESSES

Output of stress is the same as for strain.

- TRANSFORMATION

Standard transformation of three global degrees of freedom to local degrees of freedom.

- SHELL TRANSFORMATION

  The shell transformation option type 2 may be used to permit easier application of point loads, moments and/or boundary conditions on a node. For a description of the transformation type, see Volume A, Page A2.12-1. Note that if the FOLLOW FORCE option is invoked, the transformations will be based on the updated configuration of the element.

- TYING

  No special tying available.

- OUTPUT POINTS

  Centroid or the eight integration points as shown in Figure B-6.

- MESH GENERATOR

  MARCMESH3D with optimizer and hidden lines removed plotting.

- CAUTION

  As in all three-dimensional analysis a large nodal bandwidth results in long computing times. Use the optimizers either in MARCMESH3D or MARC-OPT as much as possible.

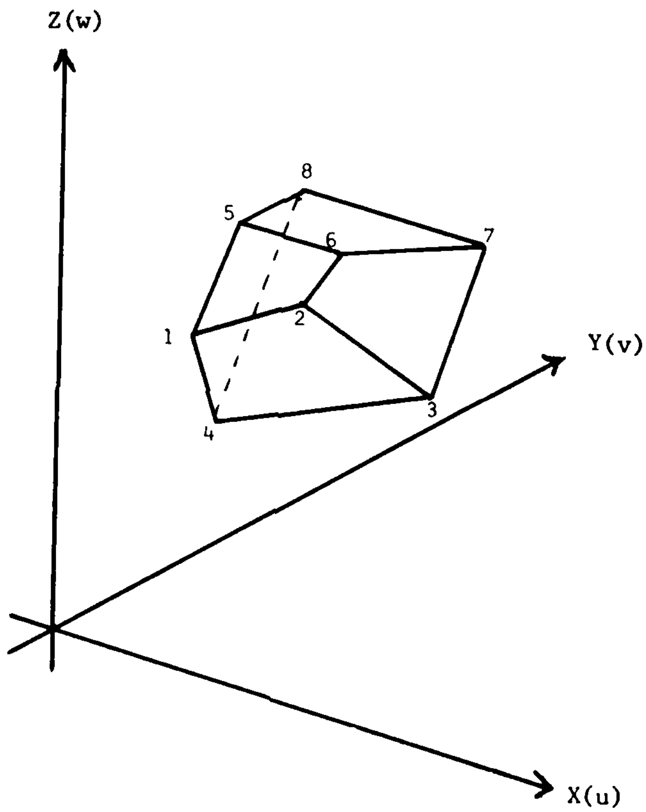


Figure B-5 Arbitrarily Distorted Cube

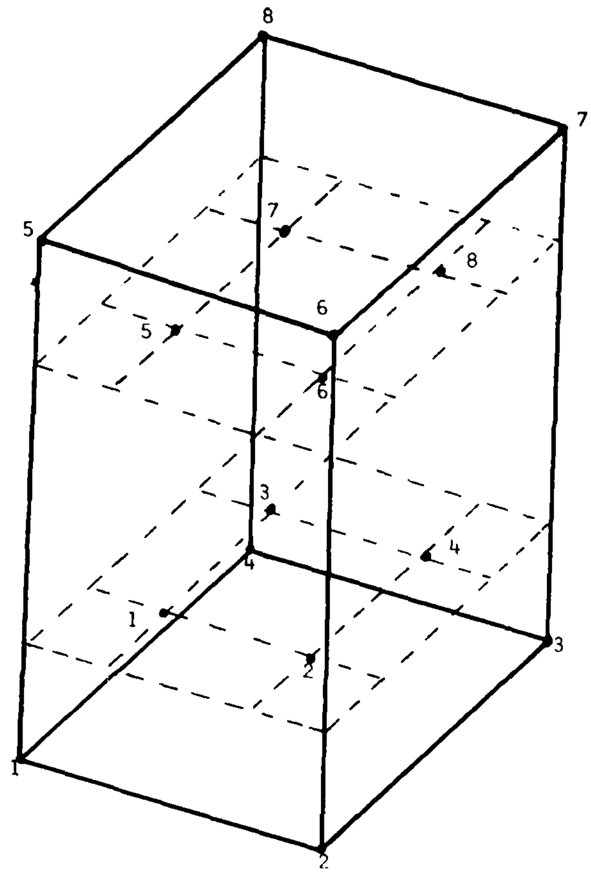


Figure B-6   8-Point Gauss Integration Scheme for Element 7

APPENDIX C

List of Conversations

During a Consultation

For the purpose of demonstrating the use of current consultation system for structural analysis, a two-dimensional structure has been analyzed as discussed in Section IV. The conversations held during the consultation are compiled and listed in this Appendix.

```
     SEG JMR>AI>#AI
GO
ENTER DATA BASE FILE NAME :
RESTORE CHECKPOINT?
ENTER RULE FILE NAME:  MBHTEST2
INITIALIZING.......
ENTER KEYWORD:  GENERAL ONE
 PLEASE ENTER DIMENSIONALITY OF YOUR
 STRUCTURE : ONE, TWO OR THREE
GENERAL 1: TWO
 PLEASE ENTER OVER_ALL_THICKNESS OF YOUR
 STRUCTURE: THICK,THIN_FLAT,THIN_CURVED, OR
 THIN_SLENDER
GENERAL 1: THIN_FLAT
 PLEASE ENTER TYPE OF LOADINGS : MECHANICAL,
 THERMAL, OR THERMO_MECHANICAL
GENERAL 1: MECHANICAL
 PLEASE ENTER TIME DEPENDENCY OF MECHANICAL LOADINGS :
 DEPENDENT, OR INDEPENDENT
GENERAL 1: INDEPENDENT
 PLEASE ENTER TEMPERATURE LEVEL OF YOUR STRUCTURE :
 HIGH OR LOW
GENERAL 1: LOW
 PLEASE ENTER STRESS_STRAIN RELATION OF THE
 MATERIAL : LINEAR OR NON_LINEAR
GENERAL 1: LINEAR
RULE #  201 PROCESSED - RESULT WAS TRUE
GENERAL_ONE     IS GIVEN
ENTER KEYWORD: GEOMETRY
MARC ELEMENT NO 3 SELECTED FOR THE ANALYSIS
RULE #  101 PROCESSED - RESULT WAS TRUE
GEOMETRY        IS TWO_D_PLANE_STRESS
ENTER KEYWORD: MATERIAL
RULE #   18 PROCESSED - RESULT WAS TRUE
MATERIAL        IS LINEAR_STRESS_STRAIN
ENTER KEYWORD:  LOADING
RULE #    9 PROCESSED - RESULT WAS TRUE
LOADING          IS MECHANICAL
```

```
ENTER KEYWORD: GENERAL_TWO
 PLEASE ENTER TOTAL NUMBER OF ELEMENTS IN THE MESH
GENERAL 1· 8
 PLESE ENTER TOTAL NUMBER OF NODES IN THE MESH
GENERAL 1: 15
 PLEASE ENTER TOTAL NUMBER OF DIFFERENT ELEMENT
 THICKNESSES IN THE MESH
GENERAL 1: 1
PLEASE ENTER TOTAL NUMBER OF ELEMENT_RANGE PAIRS
 OF ALL ELEMENT THICKNESSES IN THE MESH
GENERAL 1: 1
 PLEASE ENTER TOTAL NUMBER OF DIFFERENT MATERIALS IN
 THE MESH
GENERAL 1: 1
 PLEASE ENTER TOTAL NUMBER OF POINTS IN WORK_HARDENING
 SLOPE FUNCTIONS FOR ALL MATRIALS
GENERAL 1: 0
 PLEASE ENTER TOTAL NUMBER OF POINTS IN TEMPERATURE
 DEPENDENT MATERIAL FUNCTIONS FOR ALL MATERIALS
 (MAX. VALUE AMONG ALL PROPERTIES)
GENERAL 1: 0
 PLEASE ENTER TOTAL NUMBER OF ELEMENT_RANGE PAIRS
 FOR ALL MATERIALS IN THE MESH
GENERAL 1: 1
 PLEASE ENTER TOTAL NUMBER OF SPECIFIED NODAL POINT
 DEGREES OF FREEDOM IN THE MESH
GENERAL 1: 6
 PLEASE ENTER TOTAL NUMBER OF POINT LOADS (NODE RANGE
 PAIRS) IN THE MESH
GENERAL 1 : 1
 PLEASE ENTER TOTAL NUMBER OF DISTRIBUTED LOADS (ELEMENT
 RANGE PAIRS) IN THE MESH
GENERAL 1: 1
 PLEASE ENTER TRANSIENT TIME FOR DYNAMIC OR CREEP ANALYSIS
GENERAL 1: 0
 PLEASE ENTER TIME INCREMENT FOR DYNAMIC OF CREEP ANALYSIS
GENERAL 1· 0
RULE #  202 PROCESSED - RESULT WAS TRUE
GENERAL_TWO      IS GIVEN
```

```
ENTER KEYWORD: CONNECTIVITY
 PLEASE ENTER ELEMENT CONNECTIVITY AS FOLLOWS:
 ELEMENT NUMBER AND ELEMENT TYPE FOLLOWED BY
   TWO NODAL NUMBERS FOR TWO_NODE BEAM AND AXISYMMETRIC
   SHELL ELEMENTS OR,
   FOUR NODAL NUMBERS FOR FOUR_NODE PLANE STRESS, PLANE
   STRAIN, AND AXISYMMETRIC ELEMENTS OR,
   EIGHT NODAL NUMBERS FOR EIGHT_NODE THREE DIMENSIONAL
   BRICK ELEMENTS
ELMT 1: 1 3 1 2 5 4
ELMT 2: 2 3 2 3 6 5
ELMT 3: 3 3 4 5 8 7
ELMT 4: 4 3 5 6 9 8
ELMT 5: 5 3 7 8 11 10
ELMT 6: 6 3 8 9 12 11
ELMT 7: 7 3 10 11 14 13
ELMT 8: 8 3 11 12 15 14
RULE #    221 PROCESSED - RESULT WAS TRUE
CONNECTIVITY      IS GIVEN
ENTER KEYWORD: COORDINATES
 PLEASE ENTER NODAL COORDINATES AS FOLLOWS :
 NODAL NUMBER FOLLOWED BY
   TWO NODAL COORDINATES FOR NODAL POINT OF TWO
   DIMENSIONAL STRUCTURES OR,
   THREE NODAL COORDINATES FOR NODAL POINT OF THREE
   DIMENSIONAL STRUCTURES
NODE 1: 1 3. 0.
NODE 2: 2 4.5 0.
NODE 3: 3 6. 0.
NODE 4: 4 2.772 1.148
NODE 5: 5 4.386 2.574
NODE 6: 6 6. 4.
NODE 7: 7 2.121 2.121
NODE 8: 8 4.06 5.06
NODE 9: 9 6. 8.
NODE 10: 10 1.148 2.772
NODE 11: 11 2.074 5.386
NODE 12: 12 3. 8.
NODE 13: 13 0. 3.
NODE 14: 14 0. 5.5
NODE 15: 15 0. 8.
RULE #    222 PROCESSED - RESULT WAS TRUE
COORDINATES       IS GIVEN
```

```
ENTER KEYWORD: BASE_PROPERTY
 PLEASE ENTER MATERIAL IDENTIFICATION AND YOUNGS_
 MODULUS OF ALL MATERIALS
MATERIAL 1: 1 30.E6
 PLEASE ENTER MATERIAL IDENTIFICATION AND POISSONS
 RATIO OF ALL MATERIALS
MATERIAL 1: 1 0.3
 PLEASE ENTER MATERIAL IDENTIFICATION AND DENSITY
 OF ALL MATERIALS
MATERIAL 1: 1 0.
 PLEASE ENTER MATERIAL IDENTIFICATION AND COEFF_
 OF_THERMAL_EXPANSION OF ALL MATERIALS
MATERIAL 1: 1 0.
 PLEASE ENTER MATERIAL IDENTIFICATION AND BASE
 (STRESS_FREE) TEMPERATURE OF ALL MATERIALS
MATERIAL 1: 1 0.
 PLEASE ENTER MATERIAL IDENTIFICATION AND YIELD_
 STRESS OF ALL MATERIALS
MATERIAL 1: 1 20000.
RULE #   241 PROCESSED - RESULT WAS TRUE
BASE PROPERTY    IS GIVEN
ENTER KEYWORD: MATERIAL RANGE
 PLEASE ENTER MATERIAL IDENTIFICATION AND ELEMENT_
 RANGE PAIRS OF ALL MATERIALS
EL RANGE 1: 1 1 8
RULE #   244 PROCESSED - RESULT WAS TRUE
MATERIAL_RANGE    IS GIVEN
ENTER KEYWORD: ELMT_THICKNESS
 PLEASE ENTER GEOMETRY DATA OR, ELEMENT THICKNESS AS
 FOLLOWS:
   GEOMETRY DATA FOR TWO_NODE BEAM AND AXISYMMETRIC SHELL
   ELEMENTS OR,
   ELEMENT THICKNESS FOR PLANE STRESS AND PLANE STRAIN
   ELEMENTS
THICKNESS 1: 1.
RULE #    223 PROCESSED - RESULT WAS TRUE
ELMN_THICKNESS    IS GIVEN
ENTER KEYWORD: THICKNESS_RANGE
 PLEASE ENTER ELEMENT THICKNESS IDENTIFICATION AND
 ELEMENT_RANGE PAIR FOR GIVEN THICKNESS
EL RANGE 1: 1 1 8
RULE #   224 PROCESSED - RESULT WAS TRUE
THICKNESS_RANGE  IS GIVEN
```

```
ENTER KEYWORD: BOUNDARY_C
 PLEASE ENTER NODAL POINT RANGE, DEGRESS_OF_FREEDOM.
 AND SPECIFIED DISPLACEMENT (OPTIONAL, DEFAULT IS 0.0)

BOUNDARY 1: 1 1 2 0.
BOUNDARY 2: 2 2 2 0.
BOUNDARY 3: 3 3 2 0.
BOUNDARY 4: 13 13 2"1 0.
BOUNDARY 5: 14 14 1 0.
BOUNDARY 6: 15 15 1 0.
RULE #   261 PROCESSED - RESULT WAS TRUE
BOUNDARY_C        IS GIVEN
ENTER KEYWORD: POINT_LOAD
 PLEASE ENTER NODE NO., POINT LOAD IN FIRST DIRECTION.
 POINT LOAD IN SECOND DIRECTION AND POINT LOAD IN THIRD DIRECTION
NODE 1: 15 0. 0. 0.
RULE #   271 PROCESSED - RESULT WAS TRUE
POINT_LOAD        IS GIVEN
ENTER KEYWORD: DIST_LOAD
 PLEASE ENTER NUMBER OF ELEMENTS,
 ELEMENT RANGE, FACE IDENTIFICATION OF THE
 DISTRIBUTED LOAD AND DISTRIBUTED LOAD
ELMT 1: 2 6 8 6 -10000.
RULE #   272 PROCESSED - RESULT WAS TRUE
DIST_LOAD         IS GIVEN
ENTER KEYWORD: ANALYSIS_ONE
RULE #    22 PROCESSED - RESULT WAS TRUE
ANALYSIS_ONE      IS ELASTIC
ENTER KEYWORD: ANALYSIS_TWO
CANT DO ANYTHING
ENTER KEYWORD: ANALYSIS_THREE
CANT DO ANYTHING
ENTER KEYWORD: ANALYSIS_FOUR
CANT DO ANYTHING
ENTER KEYWORD: ANALYSIS_FIVE
CANT DO ANYTHING
ENTER KEYWORD: P_CARD_ONE
SET DEFAULTS MAXALL ETC
WRITE TITLE CARD ON INPUT1
WRITE SIZING CARD ON INPUT1
WRITE ALL POINTS CARD ON INPUT1
RULE #   203 PROCESSED - RESULT WAS TRUE
P_CARD_ONE        IS GIVEN
```

```
ENTER KEYWORD: P_CARD_TWO
CANT DO ANYTHING
ENTER KEYWORD: P_CARD_THREE
CANT DO ANYTHING
ENTER KEYWORD: P_CARD_FOUR
CANT DO ANYTHING
ENTER KEYWORD: P_CARD_FIVE
CANT DO ANYTHING
ENTER KEYWORD: M_CARD_ONE
WRITE END CARD ON INPUT2
WRITE CONNECTIVITY CARD ON INPUT2
WRITE COORDINATE CARD ON INPUT2
WRITE PROPERTY CARD ON INPUT2
WRITE GEOMETRY CARD ON INPUT2
WRITE BOUNDARY CONDITION CARD ON INPUT2
WRITE TRACTION CARDD ON INPUT2
WRITE CONTROL CARD ON INPUT2
WRITE POST CARD ON INPUT2
RULE #    204 PROCESSED - RESULT WAS TRUE
M_CARD_ONE         IS GIVEN
ENTER KEYWORD: M_CARD_TWO
CANT DO ANYTHING
ENTER KEYWORD: M_CARD_THREE
CANT DO ANYTHING
ENTER KEYWORD: M_CARD_FOUR
CANT DO ANYTHING
ENTER KEYWORD: L_CARD_ONE
WRITE END OPTION CARD ON INPUT3
RULE #    205 PROCESSED - RESULT WAS TRUE
L_CARD_ONE         IS GIVEN
ENTER KEYWORD: L_CARD_TWO
CANT DO ANYTHING
ENTER KEYWORD: L_CARD_THREE
CANT DO ANYTHING
ENTER KEYWORD: L_CARD_FOUR
CANT DO ANYTHING
ENTER KEYWORD: L_CARD_FIVE
CANT DO ANYTHING
ENTER KEYWORD: INPUT_FILE
RULE #    216 PROCESSED - RESULT WAS TRUE
INPUT_FILE         IS GIVEN
```

```
ENTER KEYWORD: RESULT_ONE
READ SUMMARY OF MARC RESULTS FROM USDATA
RULE #   251 PROCESSED _ RESULT WAS TRUE
RESULT_ONE         IS GIVEN
ENTER KEYWORD: RESULT_TWO
RESULTS FOR INCREMENT NO         1
MAX XX STRAIN IS          -0.3556E-03
MAX YY STRAIN IS          -0.1010E-02
MAX XY STRAIN IS          -0.4184E-03
MAX EQUIV. PLASTIC STRAIN IS          0.0000E 00
MAX EQUIV. CREEP   STRAIN IS          0.0000E 00
MAX XX STRESS IS          -0.1000E 05
MAX YY STRESS IS          -0.3157E 05
MAX XY STRESS IS          -0.4828E 04
MAX EQUIV. STRESS             IS          0.2982E 05
RULE #   252 PROCESSED - RESULT WAS TRUE
ENTER KEYWORD:
```


☆U.S.Government Printing Office: 1980 — 757-002/316

DATE
ILMED
-8