LS-INGRID: A Pre-Processor
And Three-Dimensional Mesh Generator
For The Programs LS-DYNA,
LS-NIKE3D
And TOPAZ3D

Version 3.5

Livermore Software Technology Corporation
2876 Waverley Way
Livermore, CA  94550

August 1998
# TABLE OF CONTENTS

ABSTRACT ....................................................................................................................... I.1
PREFACE........................................................................................................................ I.1

1. LS-INGRID BASICS ........................................................................................................... 1.1
   1.1 The Parser................................................................................................................ 1.1
   1.2 Command File Format............................................................................................ 1.2
   1.3 The Calculator ........................................................................................................ 1.2
   1.4 Built in Variables .................................................................................................... 1.3
   1.5 Basic Arithmetic Options ...................................................................................... 1.4
   1.6 Logical Operations .................................................................................................. 1.4
   1.7 Functions ................................................................................................................. 1.4
   1.8 Options ................................................................................................................... 1.5
   1.9 Directives ................................................................................................................ 1.6

2. Control Commands........................................................................................................... 2.1

3. IDEAS Part.................................................................................................................. 3.1
   3.1 Options and Functions............................................................................................. 3.1

4. PATRAN Part.............................................................................................................. 4.1
   4.1 Options and Functions............................................................................................. 4.1

5. NASTRAN Part............................................................................................................. 5.1
   5.1 Options and Functions............................................................................................. 5.1

6. Standard Part .............................................................................................................. 6.1
   6.1 DEFINITIONS ............................................................................................................. 6.1
   6.2 Index Progression .................................................................................................... 6.10
   6.3 Part Commands and Functions ............................................................................. 6.11

7. Beam Part .................................................................................................................... 7.1
   7.1 Options and Functions............................................................................................. 7.3

8. Old Data Part .............................................................................................................. 8.1
   8.1 Options and Functions............................................................................................. 8.5

9. MAZE Part .................................................................................................................. 9.1
   9.1 Required Part Data .................................................................................................. 9.1
   9.2 Options and Functions............................................................................................. 9.3
   9.3 Functions ................................................................................................................ 9.5

10. EDIT Part.................................................................................................................. 10.1
    10.1 Options and Functions......................................................................................... 10.1

11. DYNA3D Part ............................................................................................................. 11.1
    11.1 Options and Functions......................................................................................... 11.1

12. Loads and Boundary Conditions.................................................................................. 12.1
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.</td>
<td>Interactive Commands</td>
<td>13.1</td>
</tr>
<tr>
<td>13.1</td>
<td>Exploded View Commands</td>
<td>13.10</td>
</tr>
<tr>
<td>13.2</td>
<td>TAURUS/Post-Processing Commands</td>
<td>13.11</td>
</tr>
<tr>
<td>14.</td>
<td>Two-Dimensional Line Definitions</td>
<td>14.1</td>
</tr>
<tr>
<td>15.</td>
<td>Three-Dimensional Line Definitions</td>
<td>15.1</td>
</tr>
<tr>
<td>16.</td>
<td>Surface Definitions</td>
<td>16.1</td>
</tr>
<tr>
<td>17.</td>
<td>Volume Definitions</td>
<td>17.1</td>
</tr>
<tr>
<td>18.</td>
<td>Coordinate Transformations</td>
<td>18.1</td>
</tr>
<tr>
<td>19.</td>
<td>LS-DYNA2D Commands and Materials</td>
<td>19.1</td>
</tr>
<tr>
<td>19.1</td>
<td>LS-DYNA2D Material Input</td>
<td>19.4</td>
</tr>
<tr>
<td>19.2</td>
<td>General Material Options</td>
<td>19.4</td>
</tr>
<tr>
<td>20.</td>
<td>LS-DYNA3D Commands and Materials</td>
<td>20.1</td>
</tr>
<tr>
<td>20.1</td>
<td>LS-DYNA3D Material Input</td>
<td>20.13</td>
</tr>
<tr>
<td>20.2</td>
<td>General Material Options</td>
<td>20.13</td>
</tr>
<tr>
<td>22.</td>
<td>LS-NIKE2D Commands and Materials</td>
<td>22.1</td>
</tr>
<tr>
<td>22.1</td>
<td>LS-NIKE2D Material Input</td>
<td>22.4</td>
</tr>
<tr>
<td>23.</td>
<td>LS-NIKE3D Commands and Materials</td>
<td>23.1</td>
</tr>
<tr>
<td>23.1</td>
<td>LS-NIKE3D Material Input</td>
<td>23.4</td>
</tr>
<tr>
<td>24.</td>
<td>TOPAZ Commands and Materials</td>
<td>24.1</td>
</tr>
<tr>
<td>24.1</td>
<td>TOPAZ Material Input</td>
<td>24.3</td>
</tr>
</tbody>
</table>

**ACKNOWLEDGMENTS** | ACK.1 |

**REFERENCES** | REF.1 |
**LS-INGRID**: A Pre-Processor and Three Dimensional Mesh Generator for the Programs LS-DYNA, LS-NIKE3D and TOPAZ3D.

**ABSTRACT**

LS-INGRID is a general purpose pre-processor for the programs, LS-NIKE2D[1], LS-NIKE3D[2], LS-DYNA2D[3], LS-DYNA3D[4], TOPAZ2D[5] and TOPAZ3D[6]. It can be used as a simple translator to convert various databases to these programs. In addition, it is a general purpose three-dimensional mesh generator with considerable capability to deal with complex geometries and allows for parametric geometric modeling.

**PREFACE**

LS-INGRID is an alternative mesh generator for finite element modeling which is principally intended as "research" program or one that focuses on various capabilities and techniques which are not addressed by commercial mesh generators. As a general purpose mesh generator, the capabilities are fairly complete with a wide range of geometric capabilities. An extensive parametric modeling capability is also support. LS-INGRID is most effective in combination with NIKExx and DYNAxx. A considerable amount of effort has gone into making LS-INGRID support virtually every feature of these programs (an almost impossible task given the rate that LS-DYNA3D improves). Although the usage of LS-INGRID can seem somewhat combersome relative to more traditional mesh generation schemes, the productivity of users performing parametric modeling tasks with LS-INGRID can much higher in some cases.

Unlike most general purpose mesh generators, LS-INGRID was developed by the Author for the sole purpose of aiding them in their routine analysis tasks. The original code was developed to assist in the preparation of SAP5 models at the University of Tennessee beginning in 1978. The 1978 program was loosely based on index space ideas obtained from the INGEN[7] program which was developed at Los Alamos National Laboratory. In 1981, the author moved to Lawrence Livermore National Laboratory. INGRID developments continued at LLNL on the side because LLNL was committed to the development of MAZE3D, but did not have any supported three-dimensional mesh generator. In 1985, the MAZE3D effort was finally abandoned and INGRID became the principal mesh generator of LLNL by default. At this time, the program was merged with the MAZE[8] program to produce a version similar to the current LS-INGRID.

After 1985, development work continued at SPARTA with a steady evolution and the
addition of the parametric modeling capabilities. LS-INGRID is the latest version supported by LSTC. Development on LS-INGRID is continuing in the directions which proved most popular in the past. The emphasis will continue to be providing a general purpose capability focused on NIKExx and DYNAxx with much work being done to support advanced modeling capabilities which are not found in any other program.
1. LS-INGRID BASICS

The LS-INGRID input file is an ASCII datafile which contains a complete description of the analysis. The commands are input using a parser which is simple and efficient, but also has a considerable amount of flexibility for dealing with complex situations.

1.1 THE PARSER

The parser basically takes a stream of blank delimited character strings and number and decodes them for the program. The character strings are for commands or parameters and are arbitrary in length. Normally, only the first four characters are significant. Deviations from this rule are described in the documentation. Numbers can be input in a variety of formats ranging from simple integers to floating point numbers specified with an "E" format. If an error is detected in the decoding of a number, the user will be notified.

All character input for commands or numbers is automatically converted to lower case for processing. Thus case selection can be performed strictly for the purpose of enhancing readability. Any number of commands and numbers can be placed on a single line of input with the only constraint being the 80 character input line limit. In the commands description, upper case characters or characters enclosed within quotes are commands which are to be typed exactly as lower (ignoring case). Lower case items represent variables which require input.

Comments may be included by using a "c" anywhere in the input followed by a blank and the comment. If the comment does not begin in column 1, then the "c" must be preceded by a blank. Blocks of input lines can be commented by preceding the block with the character "{" and ending it with "}".

Although items are normally blank delimited, commas can also be used to separate items. Two commas which are separated by blanks are treated as having the number 0 between them. Lists of numbers or character strings are input and terminated normally by a ";". This ";" does not necessarily need a blank between it and the last item. If the list is a list of numbers, then the list can be terminated by simply beginning the next command and eliminating the semicolon.

A function calculator is also built into the parser to permit advanced programming techniques to be used. This calculator is invoked by placing the calculator command within two square brackets. If the parser is expecting a character string, then the function will be processed without any other effect on the command stream. If a number is expected, then the calculator will send whatever value it calculates to the program. See Section 1.3 for a detailed description of the calculator functions.
1.2 COMMAND FILE FORMAT

The LS-INGRID input file "ingredi" has a relatively free flowing input format with few restrictions, some of which are:

- Define an item before using it; e.g. a line definition must occur before applying it to a part.
- Materials data and code execution options cannot be input until a code output option has been selected.
- Some commands have order dependent effects, e.g. rotating local coordinate systems successively about different axes.
- Many items which have names in LS-INGRID are assigned numbers for the analysis program. These numbers are assigned sequentially starting from one based on the order of first occurrence of names.

The form of ingredi is as follows:

Title line (format is 80al)
Control commands (Section 2)
Part definition (Section 3 - 8)
Control commands
Part definition
.
.
.
END

1.3 The Calculator

The calculator is used to insert expressions into LS-INGRID input descriptions and is particularly useful for developing parametric models. When used in conjunction with the "include" command, it is possible to write programs for individual parts which can then be assembled into larger models. The calculator capabilities are invoked by inserting an expression anywhere in the input between two square brackets (e.g. [5*sin(30)]). If at that point in the input, LS-INGRID is expecting an integer or a floating point number, then the expression is evaluated and the results passed to LS-INGRID as either the nearest integer or floating point number. If a character string is expected, then the expression is evaluated and skipped over as if it were just a comment.
Within the calculator variables may be created and they will remain in effect until the program completes. Thus the expression \([\text{length}=5*5]\) would store 25.0 into a variable named "length" and return 25.0 to LS-INGRID if a number is expected. This variable could be recalled later by \([\text{length}]\).

Separate from the variable capability is a function capability. The function capability stores an expression which may consist of variables and other functions into a particular name for future evaluations. An example follows:

\[
[a=1] \quad [b=2] \quad [c=1] \quad \text{Set some variables so this won't evaluate improperly.}
\]

\[
[\text{def root1}(a,b,c)=(-b+\sqrt{b^2-4*a*c})/2*a] \quad \text{Define the function.}
\]

\[
[\text{root1}(2,1,0)] \quad \text{Evaluate the function.}
\]

The general form of the calculator's capabilities is as follows:

\[
[\text{option name} = \text{expression}]
\]

Following is a summary of the calculator capabilities:

### 1.4 Built in Variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>pi</td>
<td>p</td>
</tr>
<tr>
<td>e</td>
<td>e</td>
</tr>
<tr>
<td>#</td>
<td>Result of last operation</td>
</tr>
<tr>
<td>nnnode</td>
<td>The current node number is set to nnnode when outside of a part. Until the first part is complete, nnnode is zero.</td>
</tr>
<tr>
<td>nbeam</td>
<td>The current beam element number is set to nbeam outside of a part. Until the first part is complete, nbeam is zero.</td>
</tr>
<tr>
<td>nbrick</td>
<td>The current brick element number is set to nbrick outside of a part. Until the first part is complete, nbrick is zero. nbrick includes both regular bricks and 8-node shell elements.</td>
</tr>
<tr>
<td>nshell</td>
<td>The current shell element number is set to nshell outside of a part. Until the first part is complete, nshell is zero.</td>
</tr>
<tr>
<td>npart</td>
<td>This variable is set inside parts and is set to the current part number.</td>
</tr>
</tbody>
</table>
1.5 Basic Arithmetic Options

<table>
<thead>
<tr>
<th>Operator</th>
<th>Purpose</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Addition</td>
<td>3+4</td>
</tr>
<tr>
<td>-</td>
<td>Subtraction</td>
<td>4-1</td>
</tr>
<tr>
<td>*</td>
<td>Multiplication</td>
<td>5.5*7.6</td>
</tr>
<tr>
<td>/</td>
<td>Division</td>
<td>7.5/length</td>
</tr>
<tr>
<td>^</td>
<td>Exponentiation</td>
<td>10^3</td>
</tr>
<tr>
<td>%</td>
<td>Modulo arithmetic</td>
<td>5%2</td>
</tr>
</tbody>
</table>

1.6 Logical Operations

The result of a logical operator is 1.0 if true and 0.0 if false. These may be used either as expressions or as part of "if-then-else-endif" constructs.

<table>
<thead>
<tr>
<th>Function</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>==</td>
<td>Equal to</td>
</tr>
<tr>
<td>!=</td>
<td>Not equal to</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>&amp;&amp;</td>
<td>Logical and</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>!</td>
<td>Negation</td>
</tr>
</tbody>
</table>

1.7 Functions

The angles in the following trigonometric functions are all in radians by default. This can be controlled by the "deg" and "rad" options listed in the "options" section below.

<table>
<thead>
<tr>
<th>Function</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>sin(angle)</td>
<td>Trigonometric sine.</td>
</tr>
<tr>
<td>cos(angle)</td>
<td>Trigonometric cosine.</td>
</tr>
<tr>
<td>tan(angle)</td>
<td>Trigonometric tangent.</td>
</tr>
<tr>
<td>asin(x)</td>
<td>Inverse trigonometric sine.</td>
</tr>
<tr>
<td>acos(x)</td>
<td>Inverse of trigonometric cosine.</td>
</tr>
<tr>
<td>atan(x)</td>
<td>Inverse of trigonometric tangent.</td>
</tr>
<tr>
<td>atan2(y,x)</td>
<td>Two argument inverse tangent.</td>
</tr>
<tr>
<td>sinh(x)</td>
<td>Hyperbolic sine.</td>
</tr>
<tr>
<td>cosh(x)</td>
<td>Hyperbolic cosine.</td>
</tr>
<tr>
<td>tanh(x)</td>
<td>Hyperbolic tangent.</td>
</tr>
<tr>
<td>exp(x)</td>
<td>Exponential.</td>
</tr>
</tbody>
</table>
### 1.5 LS-INGRID BASICS

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ln(x)</td>
<td>Natural logarithm.</td>
</tr>
<tr>
<td>ln2(x)</td>
<td>Logarithm base 2.</td>
</tr>
<tr>
<td>log(x)</td>
<td>Logarithm base 10.</td>
</tr>
<tr>
<td>min(x₁,x₂,...)</td>
<td>Minimum of arbitrary number of parameters.</td>
</tr>
<tr>
<td>max(x₁,x₂,...)</td>
<td>Maximum of arbitrary number of parameters.</td>
</tr>
<tr>
<td>gcd(x₁,x₂,...)</td>
<td>Greatest common denominator.</td>
</tr>
<tr>
<td>lcm(x₁,x₂,...)</td>
<td>Least common multiple.</td>
</tr>
<tr>
<td>asa(angle,side,angle)</td>
<td>Evaluate the triangle and return largest angle.</td>
</tr>
<tr>
<td>ass(angle,side,side)</td>
<td>Evaluate the triangle and return largest angle.</td>
</tr>
<tr>
<td>sas(side,angle,side)</td>
<td>Evaluate the triangle and return largest angle.</td>
</tr>
<tr>
<td>sss(side,side,side)</td>
<td>Evaluate the triangle and return largest angle.</td>
</tr>
<tr>
<td>rnd</td>
<td>Return a random number.</td>
</tr>
<tr>
<td>rnd2</td>
<td>Return a random number but do not update the seed.</td>
</tr>
</tbody>
</table>

### 1.8 OPTIONS

<table>
<thead>
<tr>
<th>Function</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>help</td>
<td>Print the help message.</td>
</tr>
<tr>
<td>help subject</td>
<td>Print help for any of the calculator functions or options.</td>
</tr>
<tr>
<td>def name=expression</td>
<td>Define a function &quot;name&quot;. Any time name is encountered in future expressions, it will be recursively evaluated.</td>
</tr>
<tr>
<td>save filename</td>
<td>Save all variables to file &quot;filename&quot;.</td>
</tr>
<tr>
<td>load filename</td>
<td>Load variables from file &quot;filename&quot;.</td>
</tr>
<tr>
<td>quit</td>
<td>Exit calculator (this will shut down LS-INGRID).</td>
</tr>
<tr>
<td>rad</td>
<td>All angles for trigonometric functions are assumed to be defined in radians (default).</td>
</tr>
<tr>
<td>deg</td>
<td>All angles for trigonometric functions are assumed to be defined in degrees.</td>
</tr>
<tr>
<td>list</td>
<td>List current active variables.</td>
</tr>
<tr>
<td>flist</td>
<td>List current definitions of functions.</td>
</tr>
<tr>
<td>root(cₙ,...,c₁,c₀)</td>
<td>Determine the roots of the nth degree polynomial with coefficients c₀ through cₙ.</td>
</tr>
<tr>
<td>factor(x)</td>
<td>Factor x into prime coefficients.</td>
</tr>
<tr>
<td>integral(e₁,e₂,f,v)</td>
<td>Determine the integral of the function f with respect to the variable v. The limits are from e₁ to e₂ which may be expressions. A Romberg integration rule is used.</td>
</tr>
<tr>
<td>degree n</td>
<td>The degree of Romberg integration for the &quot;integral&quot; command is n (default=4). Simpson's rule corresponds to n=1 and the trapezoidal rule is n=0.</td>
</tr>
<tr>
<td>solve (f₁,f₂,...,fₙ) for (x₁=v₁+d₁,...,xₙ=vₙ+dₙ)</td>
<td>Solve a system of nonlinear equations. The equations are previously defined functions, f₁ through fₙ. Variables x₁ through xₙ must be listed and the calculator will attempt to determine them. Optional inputs include vᵢ and dᵢ. The initial starting guess is vᵢ and the initial increments for iterations are dᵢ.</td>
</tr>
</tbody>
</table>
maxits $n$  Set the maximum number of iterations for the "solve" command to $n$ (default=30).

tol $t$  Set the convergence tolerance for the "solve" command to $t$ (default=1e-6).

display  The display command is a brute force method for improving results of a divergent solve command. Results are displayed after every iteration.

if expr1 then expr2 else expr3 endif

If expression 1 is true than evaluate expression 2. Otherwise, evaluate expression 3.
1.9 DIRECTIVES

LS-INGRID provides directives to control the flow of logic in command file descriptions. Directives begin in the first column of a line and no other commands are allowed on the same line as the directive. This capability is patterned similar to the preprocessor used in the C-programming language.

<table>
<thead>
<tr>
<th>DIRECTIVE</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>#ELSE</td>
<td>This is for #IF - #ELSE - #ENDIF constructs.</td>
</tr>
<tr>
<td>#ELSEIF expression</td>
<td>Perform conditional execution of the following input lines as part of an #IF - #ELSEIF - #ENDIF construct. The results of expression should be either true (1) or false (0).</td>
</tr>
<tr>
<td>#ENDIF</td>
<td>This signifies the end of an #IF - #ENDIF block.</td>
</tr>
<tr>
<td>#ENDMACRO</td>
<td>End definition of macro initiated by #MACRO.</td>
</tr>
<tr>
<td>#IF expression</td>
<td>Conditionally execute the following lines of input. The results of expression should be either true (1) or false (0).</td>
</tr>
<tr>
<td>#INCLUDE name</td>
<td>Begin execution of commands in file name. When a #RETURN or an end-of-file is encountered, control is returned to the original file.</td>
</tr>
<tr>
<td>#MACRO name</td>
<td>Begin definition of macro name. The definition ends when an #ENDMACRO is encountered.</td>
</tr>
<tr>
<td>#RETURN</td>
<td>Cease reading input from the current input file and return control back to the next higher level file. See also #INCLUDE.</td>
</tr>
</tbody>
</table>
2. Control Commands

Control commands are optional and can be input in any order. They must not be placed inside a part. The following control commands are available:

**ARRI lc** Generate arrival times for pressure surfaces associated with load curve lc. Arrival times are generated by assuming that the loads are caused by a wave. This wave starts from a three-dimensional point, line, or surface and travels with a velocity. The arrival time is the time required for the wave to travel from the source to an individual pressure segment.

**Options:**

- **CG p** Pressure cannot exceed p.
- **CL p** Pressure cannot be less than p. (This option is selected with \( p = 0.0 \) when "COSI" is executed).
- **COSINE** The pressure varies as a function of the angle between the pressure segments normal and the direction of the wave from the source.
- **DECAY d type [a]** The pressure wave decays as a function of the distance from the source. The distance at which the scale factor for the input pressure equals 1.0 is d. The type of decay is specified by type.
  - type = "R": relationship is \( 1.0/R \)
  - type = "R2": relationship is \( 1.0/R^2 \)
  - type = "R3": relationship is \( 1.0/R^3 \)
  - type = "CONSTANT": no decay
  - type = "EXP": relationship is \( 1.0/R^a \)
- **LINE p_x p_y p_z v_x v_y v_z** The source is a line. \((p_x, p_y, p_z)\) is any point on the line and \((v_x, v_y, v_z)\) is any vector along the line.
- **PLANE p_x p_y p_z v_x v_y v_z** The source is a plane. \((p_x, p_y, p_z)\) is any point on the plane and \((v_x, v_y, v_z)\) is any vector normal to the plane.
- **POINT p_x p_y p_z** The source is a point located at \((p_x, p_y, p_z)\).
- **TOFF dt** Add dt to the arrival time.
- **VELO vel** The wave travels with velocity vel.
- **;** Terminate this command.
**BATCH**

LS-INGRID is to operate in batch mode. The interactive commands are placed at the end of the LS-INGRID model description so that they can be read automatically. A graphics device will still be requested since one of the batch output devices may be desired. If no graphics are needed use the NOPL command.

**BELT**

This section defines the properties of seat belt systems, but possibly has other applications. The seat belt capability is supported in LS-920 and later. A detailed description is included in the LS-920 manual.

**SLIPRING** name

Define a slipring.

Slip rings provide for a continuous feeding of material through a pulley. One node for the slip ring is fixed to a support structure. The slip ring logic works with seatbelt elements. Two seatbelt elements must also be identified which touch the slip ring. The friction coefficient, \( f \), determines the resisting force to the belt being pulled through the slipring.

\[
\text{FRIC} f \quad \text{Friction coefficient for material sliding through the slip ring.}
\]

; End of slipring definition.

**RETRACTOR** name

Define a retractor.

Retractor elements simulate the normal function of retractor systems for seat belts within an automobile.

**DELAY** \( dt \) Time delay for retractor operation.

**FEDL** feed_length

**LCL** lcl Load curve for loading.

**LCU** lcu Load curve for unloading.

**PULL** pullout Amount of pull-out between time delay ending and retractor locking.

**SENSOR** name Sensor for triggering retractor. At least one must be specified and no more than four.

; End of retractor definition.

**PRETENSIONER** name

Define pretensioner.

During an automobile accident, pretensioners are frequently employed to automatically increase the tension on a seatbelt. Both pyrotechnic and spring type systems are supported. Usually, a sensor triggers the event.

**PYROTECHNIC**

Use a pyrotechnic pretensioner.

**LCP** lcp Load curve for pretensioner.

**RETR** name Retractor name effected.

**TIME** \( t \) Time between sensor triggering and pretensioner acting.
The pretensioner consists of a preloaded spring.  
**DELAY** $dt$  
The time between sensor triggering and pretensioner acting.  
**SPRING** $ispd$  
Spring element number.  

The pretensioner consists of a lock spring which is removed.  
**DELAY** $dt$  
The time between sensor triggering and pretensioner acting.  
**SPRING** $ispd$  
Spring element number.  

The distance between nodes is locked.  

Pretensioner is activated by one to four sensors.  
**SENSOR** name  
End of pretensioner definition.  

A variety of sensor systems are incorporated into automobiles to sense the onset of a crash. The accelerometers are simply used for saving output to an ASCII file. The other sensors are used to initiate the retractors and pretensioners.  

The sensor is an accelerometer.  
**ACCE** $a$  
The acceleration is measured in the $x$-direction.  
**X**  
The acceleration is measured in the $y$-direction.  
**Y**  
The acceleration is measured in the $z$-direction.  
**Z**  
The sensor is triggered if $a$ is exceeded for duration $dt$.  
**TIME** $dt$  

The sensor triggers based on the retractor pullout rate.  
**RETR** name  
Retractor name.  
**RATE** $r$  
Pullout rate.  
**TIME** $t$  
Time over which rate of pull-out must be exceeded.  

The sensor triggers after time $t$.  
**TIME** $t$  

The sensor triggers based on the distance between two nodes.  
**DIST**  
**DMAX** $d_{\text{max}}$  
Maximum distance.  
**DMIN** $d_{\text{min}}$  
Minimum distance.  

End of Sensor definition.  

End of BELT command.  

Define control volume $n$. (MVMA/DYNA3D, LS-910 and later.)  
**CNV** $n$  

Options:
DAMP $d$
Set airbag damping constant to $d$.

MATE $m_1 \ldots m_n$;
The airbag consists of material subset $m_1 \ldots m_n$.

PSCA psca
Pressure scale factor used for converting pressures calculated by the thermodynamic control volume to pressures which will be applied to the finite element model. (default=1.0)

REVERSE
Reverse normals.

TYPE $m$
Control volume is of type $m$. Input for type $m$ control volume begins immediately.

VINI vini
Initial filled volume. (default = 0.0)

VSCA vsca
Scale factor for converting calculated volume to volume used for thermodynamic calculations. (default = 1.0)

Type 1:
The pressure-volume relationship is of the form:

$$\text{Pressure} = p_0 \times \frac{s}{\text{Relative volume}}$$

P0 $p_0$
Initial pressure.

SCAL $s$
Scale factor.

;
Terminate control volume input.

Type 3:
The pressure-volume relationship is of the form:

$$\dot{n}_{\text{out}} = A \sqrt{2 \rho \mu} \sqrt{\frac{\gamma}{\gamma - 1} \left( Q \frac{\gamma}{\gamma - Q} \frac{(\gamma + 1)}{\gamma} \right)}$$

where

$$Q = f(p_e, p)$$

$$\gamma = f(c_p, c_v)$$

$$p = f((\gamma - 1)E, V)$$

Options:

CV $c_v$
Heat capacity at constant volume.

CP $c_p$
Heat capacity at constant pressure.

TIN $t$
Input gas temperature.

LCM lcm
Load curve defining input mass flow rate.

MU $m$
Shape factor for exit area. If $m$ is negative, then $|m|$ is the number of a load curve which defines the shape factor as a function of pressure.
A $a$

Exit Area. If $a$ is less than zero, then $|a|$ is the number of a load curve which defines the area as a function of pressure.

$PE \ p_e$

Ambient pressure.

$RHO \ r$

Ambient density.

$GRAV \ g$

Gravitational constant. If the ambient density is defined in units of weight per volume, then the actual gravitational constant must be used. Otherwise, $g$ is set to 1.

$\;

Terminate control volume input.

Type 4:

Type 4 applies a constant internal pressure scaled by $s$ until a point in time. A load curve is used to cause a change in behavior at some point in time. When the change occurs, the volume of the control volume is first calculated and used to initialize an adiabatic gas relationship.

$PINT \ p_{int}$

Interior pressure.

$LC \ lc$

Load curve.

$SCAL \ s$

Scale factor for pressure.

$PE \ p_e$

Ambient pressure.

$RHO \ r$

Density of gas when initialized.

$GAMM \ g$

Ratio of specific heats.

$\;

Terminate control volume input.

Type 5:

Type 5 is an implementation of the Wang-Nefske airbag model.

$CV \ c_v$

Heat capacity at constant volume.

$CP \ c_p$

Heat capacity at constant pressure.

$TIN \ t$

Input gas temperature.

$LCM \ lcm$

Load curve defining input mass flow rate.

$C23 \ c_{23}$

Shape factor for exit hole.

$A23 \ a_{23}$

Exit hole area.

$CP23 \ c_{23}^e$

Shape factor for exit porosity.

$AP23 \ a_{23}^e$

Exit hole porosity.

$PEXT \ p_e$

Ambient pressure.

$RHO \ r$

Ambient density.

$GRAV \ g$

Gravitational constant.

$VOLT \ v$

Optional tank volume.

$LCOUT \ lc$

Optional load curve specifying exit flow as a function of pressure.

$PINI \ p_0$

Optional initial overpressure (gauge).

$PPOP \ p_{pop}$

Optional pressure where a plug is assumed to pop and venting begins.

$\;

Terminate control volume input.

$COOR \ nc \ data$

Input $nc$ global coordinate systems. Global coordinate systems remain in effect until reset using this command. Coordinate system data is
CONTROL COMMANDS

CSCA $s$
Scale all nodal coordinates by $s$.

CSYM
Define cyclic symmetry interface.

Options:

\textbf{AXIS $p_x p_y p_z$}
The vector which orients the axis for rotational cyclic symmetry is $(p_x,p_y,p_z)$.

; Terminate the CSYM command.

DEFAULT dir
The default directory for finding include files is dir. Currently, this is only supported on CONVEX computers.

DETP mat
Define detonation point for material mat. If mat is zero then all materials are detonated.

Options:

\textbf{LNPT $p_x p_y p_z q_x q_y q_z n$}
Generate $n$ equally spaced detonation points on the line from $(p_x,p_y,p_z)$ to $(q_x,q_y,q_z)$.

\textbf{POINT $p_x p_y p_z$}
Detonate the point $(p_x,p_y,p_z)$.

\textbf{TIME $t$}
Lighting time for detonation point.

; Terminate this command.

DN2D $i j$
Output generated is compatible with LS-DYNA2D $i$ and $j$ flags specifying which 3-D coordinates correspond to the LS-DYNA2D $r$ and $z$ coordinates. $i$ and $j$ can have values "x", "y", or "z". This activates additional commands which are described in \textit{LS-DYNA2D Options and Materials}.

DN3D
Output is generated for LS-DYNA3D. This activates additional control commands which are described in \textit{LS-DYNA3D Options and Materials}.

DS $n$
Input digitized 3-D surface number $n$. Digitized surfaces consist of a surface defined by triangles. This is not smooth for coarse meshes; however, 10,000 and more triangles are common in defining surfaces to achieve...
reasonably accurate definitions. This command consist of "DS n" optionally followed by a coordinate transformation and then one of the digitized surface types is input to complete the command.

Option:

MOVE data

Move the surface definition by data. data is described in the section on Coordinate Transformations.

Type 1: GRID n_i n_j

The surface is defined by a logically regular set of points in three dimensions. n_i*n_j points must be input in the following order. x_11 y_11 z_11 ... x_i y_i z_i 

Type 2: FEM m n

The surface is a grid of finite element quadrilaterals. It has m nodes and n elements.

Type 3: CONT n_cont

The surface is defined by n_cont contours that each have an arbitrary number of points.

Options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XLE x</td>
<td>X-coordinate of leading edge.</td>
</tr>
<tr>
<td>YLE y</td>
<td>Y-coordinate of leading edge.</td>
</tr>
<tr>
<td>ZLE z</td>
<td>Z-coordinate of leading edge.</td>
</tr>
<tr>
<td>CHORD scal</td>
<td>Scale factor for the chord length.</td>
</tr>
<tr>
<td>FNU n_u FNL n_1</td>
<td>The contour is defined by n_u points on the upper surface followed by n_1 points on the lower surface.</td>
</tr>
<tr>
<td>XF xf</td>
<td>Contour points are in the plane X=xf.</td>
</tr>
<tr>
<td>YF yf</td>
<td>Contour points are in the plane Y=yf.</td>
</tr>
<tr>
<td>ZF zf</td>
<td>Contour points are in the plane Z=zf.</td>
</tr>
<tr>
<td>L3D l</td>
<td>Use three-dimensional line definition l. The number of points on the contour is the number of points used to define the line definition.</td>
</tr>
<tr>
<td>L3E l n</td>
<td>Use three-dimensional line definition l with n equal spaced points.</td>
</tr>
</tbody>
</table>

Data:

<table>
<thead>
<tr>
<th>n</th>
<th>Number of points on contour. (Input only if nu=n1=0).</th>
</tr>
</thead>
<tbody>
<tr>
<td>x_1 y_1 z_1 ... x_n y_n z_n</td>
<td>Contour coordinates. If XF was used, do not input any X coordinates, and similarly for YF and ZF.</td>
</tr>
</tbody>
</table>
CONTROL COMMANDS

Type 4: **FUNC** $n_s$ $n_t$ $x$ $y$ $z$

Define a parametric surface using parameters $s$ and $t$. The number of points for making a grid in the $s$ direction is $n_s$ and the number of points in the $t$ direction is $n_t$. $s$ and $t$ are assumed to range from 0 to 1. $x$, $y$ and $z$ are input as functions of $s$ and $t$.

Type 5: **PROJ** $m$ offset

The current digitized surface is formed by taking digitized surface $m$ and projecting $m$ a distance "offset" in the normal direction.

*Options:*

- **XSYM** $x$sym $x$tol
  - Digitized surface $m$ is symmetric about $X=x$sym. This forces normal components of points within $x$tol of the symmetry plane to be in the $Y$-$Z$ plane only.

- **YSYM** $y$sym $y$tol
  - Digitized surface $m$ is symmetric about $Y=y$sym. This forces normal components of points within $y$tol of the symmetry plane to be in the $X$-$Z$ plane only.

- **ZSYM** $z$sym $z$tol
  - Digitized surface $m$ is symmetric about $Z=z$sym. This forces normal components of points within $z$tol of the symmetry plane to be in the $X$-$Y$ plane only.

*;*

End digitized surface definition.

**END**

Terminate the model description.

**FDEF** $n$ $v_1$ $v_2$ $v_3$ $v_4$ $v_5$ $v_6$ $v_7$ $v_8$ $v_9$

Define fold plane number $n$. Fold planes are used later in the interactive phase to generate folded models of meshes such as airbags. The nine parameters have the following meanings:

$v_1 = x$ or $y$ position of fold relative to the unfolded mesh.

$v_2 = x$ or $y$ position of fold relative to the folded mesh.

$v_3 = \text{positive / negative fold flag. If } v_3=+1, \text{ then the folded portion lies along the positive coordinate. If } v_3=-1, \text{ then the folded portion lies along the negative coordinate.}$

$v_4 = \text{fold thickness desired.}$
$v_5$ = Up / down flag. If $v_5$=+1, then the material is folded onto the top of the mesh. If $v_5$=-1, then the material is folded under the mesh.

$v_6$ = Direction. The fold plane is normal to the $X$-axis if $v_6$=0 and normal to the $Y$-axis if $v_6$=1.

$v_7$ = Fold logic flag. The thin fold logic is used if $v_7$ = 0 and the thick fold logic is used if $v_7$=1.

$v_8$ = Scale factor. This can be used to scale the normal thickness offsets at the fold point. This will increase the separation between layers, but possibly cause unacceptable mesh distortion.

$v_9$ = Fold radius. This will take precedence over the internally computed fold radius.

See also FOLD and PFOLD under Interactive Commands.

**FIGN** $f$ expr

When performing fold definition $f$, ignore nodes with undeformed coordinates that result in expr being true. (e.g. fign 6 $[y\leq0]$).

**FLEX** name

Begin definition of flexure-torsion or cardan joints (LS-920 and later.)

*Options:*

**CARDAN**

This joint is a cardan joint.

**FLEXION**

This joint is a flexion-torsion joint.

**LC1** lc1

First torque-twist load curve.

**LC2** lc2

Second torque-twist load curve.

**LC3** lc3

Third torque-twist load curve.

**LC4** lc4

Fourth torque-twist load curve.

**LC5** lc5

Fifth torque-twist load curve.

**LC6** lc6

Sixth torque-twist load curve.

**MATM** $m$

Define master material.

**MATS** $m$

Define slave material.

**MSYS** $s$

Define master side local system.
PHIF \( s \)  
First angle friction.

PHIS \( s \)  
First angle stiffness.

PSIF \( s \)  
Third angle friction.

PSIS \( s \)  
Third angle stiffness.

SSYS \( s \)  
Define slave side local system.

STOPA- \( s \)  
Negative stop for first angle.

STOPA+ \( s \)  
Positive stop for first angle.

STOPB- \( s \)  
Negative stop for second angle.

STOPB+ \( s \)  
Positive stop for second angle.

STOPC- \( s \)  
Negative stop for third angle.

STOPC+ \( s \)  
Positive stop for third angle.

THEF \( s \)  
Second angle friction.

THES \( s \)  
Second angle stiffness.

FMOV \( f \) data  
After performing fold definition \( f \), affected nodes are moved by the transformation described in data (see Coordinate Transformations).

FOPT \( f \) options  
Input additional parameters for airbag folding.

Options:

L3D  
Fold about 3-D line definition \( ldnum \).

ANGLE\( \theta \)  
Fold the material \( \theta \) degrees.

SCALE scale  
The folded section will become thicker by the factor \( scale \).

;  
End of FOPT command.

FSYM \( m x y z n_x n_y n_z s_f \)  
Define failing symmetry plane \( m \). \((x,y,z)\) is any point on the plane and \((n_x,n_y,n_z)\) is any normal vector. Solid element faces are slaved to the symmetry plane and failure occurs when the normal stress exceeds \( s_f \).

GEOC igeo mat  
Geometric contact entity definition. A geometric contact entity is an analytical surface type which can be attached to a rigid body of
Options:

COUPLE type $n$

The contact is between a CAL3D/MADYMO coupled rigid body and a deformable body. The rigid body type is either "ELLIPSE" or "PLANE" and $n$ is the number of the shape in either CAL3D or MADYMO.

FRIC $f$

Set friction coefficient to $f$.

INSIDE

MATE $m_1 \ldots m_n$;

The slaved mesh is the material subset $m_1 \ldots m_n$.

MOVE $n$

Move the entity using the global transformation number $n$.

OUTSIDE

PNLT $p$

Penalty $p$.

QUAD $q$

Quadrature rule.

$\begin{align*}
q=0: & \text{ Nodes only.} \\
q=1: & \text{ Element centers.} \\
q=2: & \text{ 2«2 quadrature on segments.}
\end{align*}$

SD $n$

Use surface definition $n$. Valid surface types include planes, ellipsoids and spheres.

; Terminate this command.

GMI $n$

Increment the default material number by $n$ for each global copy of a part. This number is initially set to zero.

INCLUDE fname

Include the information in file fname in the command stream. The INCLUDE command can perform to 20 levels deep.

JD $j$

Begin joint definition for joint $j$. Diagrams of the types of joints are shown in Figure 2-1. Nodes are assigned to joint definitions within parts.

Options:

SJ

Spherical joint.

RJ

Revolute joint.

CJ

Cylindrical joint.

PJ

Planar joint.

UJ

Universal joint.

TJ

Translational joint.

PNLT $p$

Joint penalty.

NC icode

This joint is a simple nodal constraint. The common translational degrees of freedom are specified by icode:

$\begin{align*}
=1: & \text{ Rigid Massless Beam (LS-902 and later)}
\end{align*}$
= 0: Rigid Massless Truss (LS-902 and later)
=1: X
=2: Y
=3: Z
=4: X and Y
=5: Y and Z
=6: Z and X
=7: X, Y, and Z

**RC icode**

This joint is a simple nodal constraint. The common rotational degrees of freedom are specified by icode:

-0: none
=1: X
=2: Y
=3: Z
=4: X and Y
=5: Y and Z
=6: Z and X
=7: X, Y, and Z

**REPE n**

Repeat the current joint definition for a total of $n$ joints.

; Terminate this command.
Figure 2-1. Joint definitions.
**L3D** \( n \) data

Begin definition of three-dimensional line \( n \). If line \( n \) has been previously defined, this command has the effect of destroying the old definition. See *Three Dimensional Line Definitions* for a description of the data for this command.

**LABELS**

Define offsets for node, element and other item numbering. This applies to meshes which are imported after this command.

*Options:*

- **ELEMENT** \( m \) Offset element labels by \( m \).
- **MAT** \( m \) Offset material labels by \( m \).
- **NODE** \( m \) Offset node labels by \( m \).

; End of LABELS command.

**LCD** \( n \) \( m \) \( t_1 \) \( f_1 \) ... \( t_m \) \( f_m \)

Define load curve \( n \) with \( m \) pairs of time function points.

**LCDF** \( n \) \( m \) \( f \) \( t_1 \) \( t_n \)

Define load curve \( n \) with \( m \) pairs of time function points. \( f \) is input as an analytical function of variable \( t \) which ranges from \( t_1 \) to \( t_n \). Thus to input one period of a sine wave with 100 points:

\[
\text{LCDF} 1 100 \sin(t) [0] [2\pi]
\]

**LD** \( n \) data

Begin definition of two-dimensional line \( n \). If line \( n \) has been previously defined, this command has the effect of destroying the old definition. See *Two Dimensional Line Definitions* for a description of the data for this command.

**LEV** \( n \)

Define part transformation sequence \( n \). This defines a series of operations which can be performed on groups of parts.

*Options:*

- **ADD** \( m \) Add the list of transformations in sequence number \( m \) to the current sequence.
- **COOR** \( n \) data Add \( n \) coordinate transformations to the current sequence. The data for this command is described in the section, *Coordinate Transformations.*
**LS-INGRID CONTROL COMMANDS**

**CYLI**
Perform a cylindrical coordinate transformation.

**PROD** \(i \ j\)
Form the product of sequence \(i\) with sequence \(j\).
If sequence \(i\) has \(l\) transformations, and sequence \(j\) has \(m\) transformations, then this option produces \(l \times m\) transformations and adds them to the current sequence.

**REPE** \(l_1 \ l_2 \ldots\)
Copy parts in global coordinate systems \(l_1, l_2, \ldots\).

**SPHE**
Perform a spherical coordinate transformation.

**;**
Terminate this command.

**LMI** \(n\)
Increment the default material number by \(n\) for each local copy of a part. This number is initially set to zero.

**LSYS** name
Define local system name for single point constraints, etc.

Options:

**PLANE** \(p_x \ p_y \ p_z \ r_x \ r_y \ r_z\)
The local \(X\)-axis is parallel to \((p_x, p_y, p_z)\) and \((r_x, r_y, r_z)\) is a vector in the \(XY\)-plane.

\(2 \ c_x \ c_y \ c_z \ p_x \ p_y \ p_z \ r_x \ r_y \ r_z\)
The center of the local system is \((c_x, c_y, c_z)\). \((p_x, p_y, p_z)\) is a point along the local \(X\)-axis and \((r_x, r_y, r_z)\) is a point in the \(XY\)-plane.

**NODE** \(n_1 \ n_2 \ n_3\)
The center of the local system is node \(n_1\). Node \(n_2\) is a point along the local \(X\)-axis and node \(n_3\) is a point in the \(XY\)-plane.

**;**
Terminate LSYS command.

**MAT** \(n\) data
Code-dependent material data can be input. See the chapter on the specific computer program for input related to the MAT command.

**MATE** \(m\)
The default material name for the following parts is set to \(m\). This name is initially set to 1.

**MAZT** tol
Set the MAZE tolerance to tol. This is used for a variety of two-dimensional line definitions and the MAZE parts.

**MDBC** \(m\) lc amp \(f_x \ f_y \ f_z\)
Material displacement boundary condition. This command is used only for rigid body materials in DYNA3D. The load curve number is lc, amp is the scale factor and \((f_x, f_y, f_z)\) is in the load direction.
MKDS
Make a binary database of digitized 3-D surfaces. Digitized surfaces are generated using the DS command and they are read back in using the RDDS command. This command is primarily intended to allow fast reinitialization during restarts of LS-INGRID.

MFBC mat lc amp f_x f_y f_z
Apply force to rigid body material mat. The force is scale by load curve lc and factor amp and is in direction (f_x,f_y,f_z).

MVBC m lc amp f_x f_y f_z
Material velocity boundary condition. This command is used only for rigid body materials in DYNA3D. The load curve number is lc, amp is the scale factor and (f_x,f_y,f_z) is in the load direction.

NFG name
Define nodal force group name.

Options:

LSYS name
The nodal force group is defined relative to local system name. (default=global)

; End of nodal force groups.

NIP m_1 m_2
Non-interacting pairs of materials. This is used to determine lists of noninteracting segments for use by FACET to determine radiation view factors.

NK2D i j
Output generated is compatible with LS-NIKE2D. i and j flags specifying which 3-D coordinates correspond to the LS-NIKE2D r and z coordinates. i and j can have values "x", "y", or "z". This command activates additional control commands which are described in LS-NIKE2D Options and Materials.

NK3D
Output is generated for LS-NIKE3D. This command activates additional control commands which are described in LS-NIKE3D Options and Materials.

NOPL
Do not perform plotting. This command suppresses the normal prompting for a graphics device and is useful in combination with the BATCH command.

NOTE
Input a note to be included into the output file. Example:
NOTE "Copyright 1985"

**NSMOOT**\(h\) \(n\) Perform \(n\) smoothing operations on surfaces when using the standard part. The default is zero since this can be costly and is usually necessary only for complex free form surfaces.

**ORV** \(n\) \{options\} Input orientation vector \(n\).

*Options:*

- **PLANE** \(v_x\) \(v_y\) \(v_z\) The spring/damper distances are measured in the plane defined by normal vector \((v_x,v_y,v_z)\).
- **VECTOR** \(v_x\) \(v_y\) \(v_z\) The spring/damper distances are measured along the vector defined by \((v_x,v_y,v_z)\).

**PAUSE** Execute a FORTRAN pause statement.

**PLANE** \(nplane\) Input \(nplane\) plane definition. These planes are for applying boundary conditions only. Do not try to use this command more than once in the same input file.

Repeat the following information for each plane:

- \(p_x\) \(p_y\) \(p_z\) Global coordinates of any point on the plane.
- \(q_x\) \(q_y\) \(q_z\) Any vector normal to the plane.

*Tolerance* All nodes within a distance less than tolerance from the plane are included in the definition. If the tolerance is negative, no nodes will be found. The "SW" command in the standard part can also be used to include nodes in the definition.

*Options:*

- **CYLI** \(radius\) \(len\) The stonewall is a cylindrical surface. The radius is \(radius\) and the length is \(len\). If \(len=0\), then an infinite cylinder is assumed. (LS-910 and later).
- **FRIC** \(m\) Specify stonewall friction properties.
  - \(m=0\): Frictionless sliding occurs.
  - \(0<m<1\): \(m\) is colomnb friction coefficient (LS-910 and later).
  - \(m=1\): No tangential motion allowed during contact.
- **LCD** \(lc\) \(v_x\) \(v_y\) \(v_z\) Load curve \(lc\) specifies the displacement history of the stone wall in the direction \((v_x,v_y,v_z)\).
**CONTROL COMMANDS**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCV <em>lc vx vy vz</em></td>
<td>Load curve <em>lc</em> specifies the velocity history of the stone wall in the direction (<em>vx</em>,<em>vy</em>,<em>vz</em>).</td>
</tr>
<tr>
<td>MASS <em>mass</em></td>
<td>The stonewall has mass <em>mass</em>. (LS-910 and later).</td>
</tr>
<tr>
<td>OVERRIDE <em>i</em></td>
<td>If a node is also on plane <em>i</em>, then this plane takes precedence.</td>
</tr>
<tr>
<td>PLANE <em>a_x a_y a_z alen blen</em></td>
<td>The stonewall is a finite plane. (<em>a_x</em>,<em>a_y</em>,<em>a_z</em>) is a vector which specifies an in-plane <em>a</em>-axis. The <em>b</em>-axis is determined from the cross-product of the <em>a</em>-axis with the normal vector. <em>alen</em> is the extent of the plane along the <em>a</em>-axis and <em>blen</em> is the extent along the <em>b</em>-axis. (LS-910 and later).</td>
</tr>
<tr>
<td>PRISM <em>a_x a_y a_z alen blen clen</em></td>
<td>The stonewall is a prism. (<em>a_x</em>,<em>a_y</em>,<em>a_z</em>) is a vector which specifies an in-plane <em>a</em>-axis. The <em>b</em>-axis is determined from the cross-product of the <em>a</em>-axis with the normal vector. <em>alen</em> is the extent of the plane along the <em>a</em>-axis and <em>blen</em> is the extent along the <em>b</em>-axis. <em>clen</em> is the extent along the normal axis. (LS-910 and later).</td>
</tr>
<tr>
<td>SPHE <em>radius</em></td>
<td>The stonewall is a spherical surface. The radius is <em>radius</em>. (LS-910 and later).</td>
</tr>
<tr>
<td>VELOCITY <em>v</em></td>
<td>The stonewall has an initial velocity <em>v</em> normal to the surface. (LS-910 and later).</td>
</tr>
</tbody>
</table>

One of the following three options is required to terminate the plane definition:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASYM</td>
<td>Asymmetric boundary conditions are applied to the nodes.</td>
</tr>
<tr>
<td>STONE or SW</td>
<td>The boundary condition is a stonewall.</td>
</tr>
<tr>
<td>SYMM</td>
<td>Symmetric boundary conditions are applied to the nodes.</td>
</tr>
<tr>
<td>PPLV</td>
<td>Eliminate the part transformation sequence at the top of the stack. See also &quot;PPLV&quot; and &quot;LEV&quot; in this section.</td>
</tr>
<tr>
<td>PRINT <em>v</em></td>
<td>Echo the value of <em>v</em> to the terminal. This is primarily used with the calculator functions to verify calculations.</td>
</tr>
<tr>
<td>PSCALE <em>m1 m2</em></td>
<td>Scale properties. Materials from <em>m1</em> to <em>m2</em> are treated by this command. If <em>m1</em> and <em>m2</em> are numbers, than standard numeric comparisons are used to determine if materials are within the range. Otherwise, string comparisons are used.</td>
</tr>
<tr>
<td>SECTION <em>scale</em></td>
<td>Scale all section properties by <em>scale</em>. This allows for a general unit conversion on section</td>
</tr>
</tbody>
</table>
properties.

; Terminate PSSCALE command.

**PSLV** $n$

Begin performing part transformation sequence $n$ on all following parts. This remains in effect until a PPLV command is given. A stack is used for performing transformation sequences. PSLV adds a transformation sequence to the top of the stack and PPLV eliminates the top sequence on the stack.

**QUAD**

Turn on generation of elements with quadratic shape functions in standard part. This causes 8-node shells and 20-node bricks to be generated.

**RBMG** $m_1$ $m_2$

Merge rigid body $m_1$ to rigid body $m_2$. $m_1$ is the master and $m_2$ is the slave. For a group of merged rigid bodies, there can be only one master. (DYNA3D only.)

**READ**

Read external database.

*Options:*

**NDIV** $n$

Number of subdivisions for internal NURB surface processing. (Default=1) Setting this to 2 or 3 can improve the reliability of intersections calculated from NURB surfaces, however, costs and memory requirements will increase roughly proportional to the square of this number.

**NURB** *name*

Read a NURB surface database in file *name*. This ends the READ command.

**SC03** *name*

Read a SC03 database in file *name*. This ends the READ command.

**ROTATION** $p_x$ $p_y$ $p_z$ $v_x$ $v_y$ $v_z$ $w$

Assign an initial rigid body rotation to all parts defined after this command. ($p_x$, $p_y$, $p_z$) is any point on the axis of rotation and ($v_x$, $v_y$, $v_z$) defines the axis direction. The angular velocity is $w$.

**RVBC** $m$ lc idof amp $f_x$ $f_y$ $f_z$

Rigid body velocity boundary condition. This command is used only for rigid body materials in DYNA3D. The load curve number is lc, amp is the scale factor and ($f_x$, $f_y$, $f_z$) is in the load direction. idof can be:

- 1: X-translational degree-of-freedom
- 2: Y-translational degree-of-freedom
- 3: Z-translational degree-of-freedom
=4: translational velocity in direction of vector \((f_x, f_y, f_z)\).

=5: X-rotational degree-of-freedom

=6: Y-rotational degree-of-freedom

=7: Z-rotational degree-of-freedom

=8: rotational velocity in direction of vector \((f_x, f_y, f_z)\).

=9: \(Y\) and \(Z\) degrees-of-freedom for node rotating about the global \(X\)-axis

=10: \(Z\) and \(X\) degrees-of-freedom for node rotating about the global \(Y\)-axis

=11: \(X\) and \(Y\) degrees-of-freedom for node rotating about the global \(Z\)-axis

\textbf{SD} \(n\) \textit{data}

Begin definition of surface \(n\). If surface \(n\) has been previously defined, this command has the effect of destroying the old definition. See \textit{Surface Definitions} for a description of the additional input for this command.

\textbf{SDMV} \(s_1 \ s_2\) \textit{data}

Move surface definitions \(s_1\) through \(s_2\). \textit{Data} is described in the section on \textit{Coordinate Transformations}.

\textbf{SI} \textit{islide}

Define sliding interface \(\textit{islide}\). These options apply to both slide surfaces and slide lines.

\textbf{Options:}

\textbf{A3}
Select contact interface type a3 (LS-920).
Contact type a3 is insensitive to orientation of the contact segments.

\textbf{A5}
Select contact interface type a5 (LS-920).
Contact type a5 is insensitive to orientation of the contact segments.

\textbf{A13}
Select contact interface type a13 (LS-920). This model is a single surface method which is principally used for inflating folded airbags.

\textbf{BIRTH} \(t\)
Birth time for interface (LS-910).

\textbf{BOND} \(t\)
GA slideline option: Bond shear modulus.

\textbf{BOXM} \(x_m \ x_y \ y_m \ y_x \ z_m \ z_x\)
Define box for master side of sliding interfaces (LS-910, VECDYNA).

\textbf{BOXS} \(x_m \ x_y \ y_m \ y_x \ z_m \ z_x\)
Define box for slave side of sliding interfaces. (LS-910, VECDYNA)

\textbf{COMP} \(t\)
GA slideline option: compressive strength of concrete.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAMP (d)</td>
<td>Damping coefficient (percent of critical) (LS-910).</td>
</tr>
<tr>
<td>DEATH (t)</td>
<td>Death time for interface (LS-910).</td>
</tr>
<tr>
<td>DNIS</td>
<td>Discrete nodes impacting surface.</td>
</tr>
<tr>
<td>DNTS</td>
<td>Discrete nodes tied to surface.</td>
</tr>
<tr>
<td>DUMMY</td>
<td>Dummy slide surface. This option can be used to allow distinct but coincident nodes.</td>
</tr>
<tr>
<td>FAIL (e)</td>
<td>Tied slide surface with failure when volume weighted strain exceeds (e) (LS-DYNA3D).</td>
</tr>
<tr>
<td>FD (fd)</td>
<td>Dynamic friction coefficient.</td>
</tr>
<tr>
<td>FE (d)</td>
<td>Exponential decay coefficient.</td>
</tr>
<tr>
<td>FFN (f)</td>
<td>Normal failure force.</td>
</tr>
<tr>
<td>FFNE (f)</td>
<td>Normal failure exponent.</td>
</tr>
<tr>
<td>FFS (f)</td>
<td>Shear failure force.</td>
</tr>
<tr>
<td>FFSE (f)</td>
<td>Shear failure exponent.</td>
</tr>
<tr>
<td>FRIC (f)</td>
<td>Set static and dynamic friction to (f). (default=0)</td>
</tr>
<tr>
<td>FS (fs)</td>
<td>Static friction coefficient.</td>
</tr>
<tr>
<td>GA</td>
<td>Select General Atomic's 1-D rebar slideline.</td>
</tr>
<tr>
<td>HDMG (t)</td>
<td>GA slideline option: Exponent in damage curve.</td>
</tr>
<tr>
<td>LCV</td>
<td>Load curve for force-penetration in types 19 and 20 contact. (LS-920 and later).</td>
</tr>
<tr>
<td>LS</td>
<td>Turn on limited search flag. (Default is off).</td>
</tr>
<tr>
<td>MATERIAL MAST (m_1\ m_2\ \ldots)</td>
<td>The master side of the interface consists of material subset (m_1, m_2, \ldots) (VEC/DYNA3D, LS-920 and later).</td>
</tr>
<tr>
<td>MATERIAL SLAV (m_1\ m_2\ \ldots)</td>
<td>The slave side of the interface consists of material subset (m_1, m_2, \ldots) (VEC/DYNA3D, LS-920 and later).</td>
</tr>
<tr>
<td>MAXS (t)</td>
<td>GA slideline option: maximum shear displacement.</td>
</tr>
<tr>
<td>MERGE</td>
<td>Coincident nodes are merged.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>MSCA $s$</td>
<td>Scale factor for master thicknesses. (LS-910 and later).</td>
</tr>
<tr>
<td>MTHI $t$</td>
<td>Master side thickness. (LS-910 and later).</td>
</tr>
<tr>
<td>NFAIL $fs$</td>
<td>Normal failure stress.</td>
</tr>
<tr>
<td>NOMERGE</td>
<td>Coincident nodes are not merged.</td>
</tr>
<tr>
<td>NSWS</td>
<td>Nodes spot welded to surface.</td>
</tr>
<tr>
<td>PNLM $p$</td>
<td>Change master side penalty to $p$.</td>
</tr>
<tr>
<td>PNLS $p$</td>
<td>Change slave side penalty to $p$.</td>
</tr>
<tr>
<td>PNLT $p$</td>
<td>Change penalty to $p$.</td>
</tr>
<tr>
<td>RADIUS $t$</td>
<td>GA slideline option: Radius of rebar.</td>
</tr>
<tr>
<td>SETS</td>
<td>Shell edge tied to shell surface.</td>
</tr>
<tr>
<td>SFAIL $fs$</td>
<td>Shear failure stress.</td>
</tr>
<tr>
<td>SINGLE</td>
<td>Single sided slide surface.</td>
</tr>
<tr>
<td>SL</td>
<td>Sliding only.</td>
</tr>
<tr>
<td>SSCA $s$</td>
<td>Scale factor for slave thicknesses. (LS-910 and later).</td>
</tr>
<tr>
<td>STHI $t$</td>
<td>Slave side thickness. (LS-910 and later).</td>
</tr>
<tr>
<td>SV</td>
<td>Sliding with voids (default).</td>
</tr>
<tr>
<td>T10</td>
<td>Type 10 interface.</td>
</tr>
<tr>
<td>T11</td>
<td>Type 11 interface. This is the box/material limited automatic contact for shells in LS-910 and later. It is the single surface airbag contact for MVMA/DYNA3D.</td>
</tr>
<tr>
<td>T12</td>
<td>Type 12 interface. Automatic contact for shells. (LS-910 and later).</td>
</tr>
<tr>
<td>T13</td>
<td>Type 13 interface. (LS-920 and later. Converts to the similar type 11 in MVMA/DYNA3D).</td>
</tr>
<tr>
<td>T14</td>
<td>Type 14 interface. Surface to surface eroding contact. (LS-920 and later).</td>
</tr>
<tr>
<td>T15</td>
<td>Type 15 interface. Node to surface eroding contact. (LS-920 and later).</td>
</tr>
<tr>
<td>T16</td>
<td>Type 16 interface. Single surface eroding</td>
</tr>
</tbody>
</table>
contact. (LS-920 and later).

**T17**
Type 17 interface. Surface to surface symmetric/asymmetric constraint method. (LS-920 and later).

**T18**
Type 18 interface. Taylor and Flanagan contact force calculation technique from PRONTO3D. (LS-920 and later).

**T19**
Type 19 interface. Rigid body to rigid body with specified force deflection curve. (LS-920 and later.)

**T20**
Type 20 interface. Node to rigid body with specified force deflection curve. (LS-920 and later.)

**TBI**
Tie-break interface.

**TCRS**
Thermal contact resistance is $r$.

**TIED**
Tied slide surface

**VFRI**
Viscous friction coefficient is $v$.

`;` Terminate this slide surface definition.

**SPD**
Input definition for spring/damper.

*Options:*

**ROTA**
The spring/damper is rotary and not translational.

*Options (the following options end the SPD command):*

**LE** $e$
Define a linear elastic spring with stiffness $e$. (force/displacement).

**LV** $d$
Define a linear damper with damping constant $d$. (force/velocity).

**IEP** $e$ $t$ $y$
Define an elastic-plastic spring with stiffness $e$ (force/displacement), tangent stiffness $t$ (force/displacement), and yield $y$ (force).

**NE** $l$
Define a nonlinear spring using load curve $l$. $l$ represents force versus displacement.

**NV** $l$
Define a nonlinear damper using load curve $l$. $l$ represents force versus velocity.
**CONTROL COMMANDS**

**GN** \( l_1 l_{ul} b Y_t Y_c \)  
Define a general nonlinear spring. The spring loads along load curve \( l_1 \) and unloads along \( l_{ul} \) with hardening parameter \( b \). The initial yield in tension is \( Y_t \) and \( Y_c \) for compression. (LS-910 and later).

**VE** \( K_0 K_i b T_c F_c \) \( \text{iop} \)  
Three-parameter Maxwell viscoelastic spring. \( K_0 \) is the short time stiffness, \( K_i \) is the long time stiffness with decay parameter \( b \). \( T_c \) is a cutoff time and \( F_c \) is the force after cutoff. \( \text{iop} \) is zero for an incremental treatment and nonzero for a continuous treatment. (LS-910 and later).

**TCO** \( l \) \( K_{ul} \) \( \text{flag} \)  
Inelastic tension or compression only. The spring loads along load curve \( l \). \( K_{ul} \) is an optional unloading stiffness and \( \text{flag} \) is -1.0 for tension only and +1.0 for compression only. (LS-910 and later).

**SLVM** \( m_1 m_2 \)  
This command applies to the DYNA3D coupling with CAL3D or MADYMO3D. Deformable materials can be identified as being slaved to rigid bodies which are coupled to CAL3D or MADYMO3D. During the DYNA3D initialization, the deformable materials will be repositioned to reflect the shifting to global coordinates performed by CAL3D or MADYMO3D. The master rigid body is material \( m_1 \) and the slaved deformable material is \( m_2 \). (LS-920 and later.)

**STOL** \( t \)  
Set the tolerance for surface intersections to \( t \). (Default=1.0e-6.)

**STOP**  
Execute a FORTRAN stop statement.

**SYNTAX**  
Command for redefining the syntax of various part options.

**Options:**

**REGION**  
Set syntax for <Region> in part definitions.

**Options:**

**STANDARD**  
Set syntax for <Region> in standard part.

**Options:**

**STANDARD**  
Use the standard syntax for <Region> in the
standard part.

; Terminate SYNTAX command.

MAZE Set syntax for <Region> in the MAZE part.

Options:

STANDARD The syntax for the <MRegion> is according to this manual and the 1985 INGRID manual from LLNL.

PD The syntax for the <MRegion> has 6 indices according to the modification to INGRID by LLNL after 1986.

; Terminate SYNTAX command.

OLD Set syntax for <Region> in the OLD, BEAM and other low level input parts.

Options:

STANDARD Use the standard definition of <Region>. This assumes <Region> = imin imax, unless the first item encountered is an expression in brackets. For an expression, it will only read one parameter.

1 Use a one parameter definition of <Region>. <Region> = imin=imax normally identifies single nodes and elements only, but can also identify a range with an expression.

2 Use a two parameter definition of <Region>. This does not allow an expression to specify the range, but is necessary for the following:

B [nodebeg] [nodeend] 111000.

This is because the standard method would see [nodebeg] and convert to a one parameter method.

; Terminate SYNTAX command.

SYSEND Terminate existing subsystem definition.

SYSTEM name Begin definition of subsystem name. This remains in effect until a SYSEND or another SYSTEM command is encountered or another
CONTROL COMMANDS

T12

This command must be typed just prior to the use of the MAZE part. The third side, \( L_3 \), of the next part will have exactly two times as many elements as side \( L_1 \). The transition is accomplished with quadrilateral elements. This command does not apply to triangular parts.

T13

This command must be typed just prior to the use of the MAZE part. The third side, \( L_3 \), of the next part will have exactly three times as many elements as side \( L_1 \). The transition is accomplished with quadrilateral elements. This command does not apply to triangular parts.

TEMP \( t \)

All parts defined after this command have initial temperature \( t \). This remains in effect until reset with another "Temp" command. This can be overridden with an individual part. \( t \) can be a single number or it can be an expression of the form \( t(x,y,z) \). This allows nodes to be assigned temperatures based on an analytical expression of a temperature distribution based on the nodal coordinates.

THIC \( t \)

The default thickness for shells is \( t \).

TIND mat \( I_{xx} I_{xy} I_{xz} I_{yy} I_{yz} I_{zz} \)

Specify inertia tensor.

TINE mat \( I_{xx} I_{yy} I_{zz} I_{xy} I_{yz} I_{zx} \)

Specify inertia tensor.

TIVE mat \( v_x v_y v_z w_x w_y w_z \)

Initial velocities (global), translational and rotational

TMCG mat \( c_x c_y c_z \)

Specify center of gravity.

TMM \( n t \)

The total mass of material \( n \) is \( t \). The density of the material is determined by dividing the total mass of the material by the calculated volume.

TMSM \( m s_1 s_2 \ldots ; \)

The inertial properties which are input for material \( m \) include the masses of deformable materials \( s_1, s_2, \ldots \). The properties of \( m \) are computed such that the total mass properties of \( m, s_1, s_2 \ldots \) is equal to the input values.

TMVP mat (transformation)

Move center of gravity and inertias. Transformation refers to the section, "Coordinate Transformations."

TRACER

Define tracer particles for material.
Options:

**LNPT** \( p_x \ p_y \ p_z \ q_x \ q_y \ q_z \ n \) Generate \( n \) equally spaced tracer particles on the line from \((p_x, p_y, p_z)\) to \((q_x, q_y, q_z)\).

**MATERIAL** \( t \) The tracer particle is fixed to a material point.

**POINT** \( p_x \ p_y \ p_z \) Define a tracer particle starting at point \((p_x, p_y, p_z)\).

**SPACE** \( t \) The tracer particle is fixed in space.

**TIME** \( t \) Activation time for tracer particle.

**;** Terminate this command.

**TRANS** This command must be typed just prior to a MAZE part and changes the command such that \( k+m \) elements are generated along sides \( L_1 \) and \( L_2 \) and \( m \) elements are generated along sides \( L_3 \) and \( L_4 \). This command does not apply to triangular parts or parts with variable zoning.

**TZ2D** \( i \ j \) Output generated is compatible with TOPAZ2D. \( i \) and \( j \) flags specifying which 3-D coordinates correspond to the TOPAZ2D \( r \) and \( z \) coordinates. \( i \) and \( j \) can have values "x", "y", or "z". This command activates additional control commands which are described in *TOPAZ Options and Materials*.

**TZ3D** Output is generated for TOPAZ3D. A FACET input deck will also be created if necessary. This command activates additional control commands which are described in *TOPAZ Options and Materials*.

**WRITE** \( \text{format } v_1 \ v_2 \ldots ; \) Issue a Fortran write statement. variables \( v_1, v_2, \ldots \) are written to standard out and \text{format} is the Fortran format statement. Example:

```fortran
WRITE "'(1=','e13.5)" [i] ;
```

**XOFF** \( d_x \) Global X-offset.

**XSCA** \( s \) Scale all X-coordinates.

**VD** \( n \ \text{data} \) Begin definition of volume \( n \). If volume \( n \) has been previously defined, this command has the effect of destroying the old definition. *Volume Definitions* describes the data for this command.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOFF $d_y$</td>
<td>Global Y-offset.</td>
</tr>
<tr>
<td>YSCA $s$</td>
<td>Scale all Y-coordinates.</td>
</tr>
<tr>
<td>VELOCITY $v_x$ $v_y$ $v_z$</td>
<td>Assign initial rigid body velocity ($v_x$, $v_y$, $v_z$) to all parts defined after this command. $v_x$, $v_y$ and $v_z$ can be functions of (x,y,z) to allow initial velocity distributions.</td>
</tr>
<tr>
<td>ZOFF $d_z$</td>
<td>Global Z-offset.</td>
</tr>
<tr>
<td>ZSCA $s$</td>
<td>Scale all Z-coordinates.</td>
</tr>
</tbody>
</table>
3. IDEAS Part

The IDEAS part provides for importing SDRC/IDEAS neutral files into LS-INGRID. The form of the part is as follows:

```
IDEA filename
<optional functions>
END
```

filename is the name of the IDEAS neutral file.

3.1 OPTIONS AND FUNCTIONS

Functions require the ability to identify groups of nodes and elements in a part and assign various properties. These have the general form of:

```
Keyword <region> function data
```

Where `<region>` is a part specific description of where the function is to be applied. For the current part, the nodes or elements through either node or element numbers or through analytical expressions. As an example:

```
SI [mat==2] 1 M C   Elements of material 2 are assigned to C   the master side of contact interface 1.
```

Variables available for function application are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y z</td>
<td>Part local coordinates of node or element center.</td>
</tr>
<tr>
<td>xg yg zg</td>
<td>Global coordinates of node or element center.</td>
</tr>
<tr>
<td>node</td>
<td>Node number.</td>
</tr>
<tr>
<td>mat</td>
<td>Material number.</td>
</tr>
<tr>
<td>elem</td>
<td>Element number.</td>
</tr>
</tbody>
</table>

The following options are allowed in any order. Additional functions can be applied and are described in the section on Loads and Boundary Conditions.

**COOR** nc data

Input nc local coordinate systems. Coordinate system data is described in detail in the section on Coordinate Transformations.

**CYLI**

Nodes are converted from cylindrical to rectangular coordinates. The equations for this transformation are:

\[ X = R \cos \theta \]
\[ Y = R \sin \theta \]
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LREP</strong> $l_1$ $l_2$ ... $l_n$</td>
<td>Repeat command. This command makes copies of the part in each of the <strong>local</strong> coordinate systems $l_1$ to $l_n$. If the coordinate system number is zero, the part is repeated with no transformation.</td>
</tr>
<tr>
<td><strong>MATE</strong> matnum</td>
<td>The part has material number matnum.</td>
</tr>
<tr>
<td><strong>REPE</strong> $l_1$ $l_2$ ... $l_n$</td>
<td>Repeat command. This command makes copies of the part in each of the <strong>global</strong> coordinate systems $l_1$ to $l_n$. If the coordinate system number is zero, the part is repeated with no transformation.</td>
</tr>
<tr>
<td><strong>ROTA</strong> $p_x$ $p_y$ $p_z$ $v_x$ $v_y$ $v_z$ $w$</td>
<td>Assign an initial rigid body rotation to the part. ($p_x$, $p_y$, $p_z$) is any point on the axis of rotation and ($v_x$, $v_y$, $v_z$) defines the axis direction. The angular velocity is $w$ in radians per second.</td>
</tr>
<tr>
<td><strong>SPHE</strong></td>
<td>Nodes are converted from spherical to rectangular coordinates. The equations for this transformation are: $X = R \cos \theta \sin \phi$ $Y = R \sin \theta \sin \phi$ $Z = R \cos \phi$</td>
</tr>
<tr>
<td><strong>TEMP</strong> $t$</td>
<td>The initial temperature of this part is $t$ and it can be expressed as a function of $x$, $y$, $z$ coordinates.</td>
</tr>
<tr>
<td><strong>THIC</strong> thic</td>
<td>Plates have the thickness thic for this part.</td>
</tr>
<tr>
<td><strong>VELO</strong> $v_x$ $v_y$ $v_z$</td>
<td>Assign initial rigid body velocity to all nodes within this parts. ($V_x$, $V_y$, $V_z$) is the global velocity vector. ($V_x$, $V_y$, $V_z$) can be expressed as a function of $x$, $y$, $z$ coordinates.</td>
</tr>
</tbody>
</table>
4. PATRAN Part

The PATRAN part provides for importing PATRAN neutral files into LS-INGRID. The form of the part is as follows:

```
PATRAN filename
<optional functions>
END
```

filename is the name of the PATRAN neutral file.

4.1 OPTIONS AND FUNCTIONS

Functions require the ability to identify groups of nodes and elements in a part and assign various properties. These have the general form of

```
Keyword <region> function data
```

Where `<region>` is a part specific description of where the function is to be applied. For the current part, the nodes or elements through either node or element numbers or through analytical expressions. As an example:

```
SI [mat==2] 1 M
```

Elements of material 2 are assigned to the master side of contact interface 1.

Variables available for function application are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y z</td>
<td>Part local coordinates of node or element center.</td>
</tr>
<tr>
<td>xg yg zg</td>
<td>Global coordinates of node or element center.</td>
</tr>
<tr>
<td>node</td>
<td>Node number.</td>
</tr>
<tr>
<td>mat</td>
<td>Material number.</td>
</tr>
<tr>
<td>elem</td>
<td>Element number.</td>
</tr>
</tbody>
</table>

The following options are allowed in any order. Additional functions can be applied and are described in the section on LOADS AND BOUNDARY CONDITIONS.

COOR nc data

Input nc local coordinate systems. Coordinate system data is described in detail in the section on COORDINATE TRANSFORMATIONS.

CYLI

Nodes are converted from cylindrical to rectangular coordinates. The equations for this transformation are:

\[ X = R \cos \theta \]
\[ Y = R \sin \theta \]
**LREP** $l_1 l_2 ... l_n$

Repeat command. This command makes copies of the part in each of the **local** coordinate systems $l_1$ to $l_n$. If the coordinate system number is zero, the part is repeated with no transformation.

**MATE** matnum

The part has material number matnum.

**REPE** $l_1 l_2 ... l_n$

Repeat command. This command makes copies of the part in each of the **global** coordinate systems $l_1$ to $l_n$. If the coordinate system number is zero, the part is repeated with no transformation.

**ROTA** $p_x p_y p_z v_x v_y v_z \omega$

Assign an initial rigid body rotation to the part. ($p_x$, $p_y$, $p_z$) is any point on the axis of rotation and ($v_x$, $v_y$, $v_z$) defines the axis direction. The angular velocity is $\omega$ in radians per second.

**SPHE**

Nodes are converted from spherical to rectangular coordinates. The equations for this transformation are:

\[
\begin{align*}
X &= R \cos \theta \sin \phi \\
Y &= R \sin \theta \sin \phi \\
Z &= R \cos \phi
\end{align*}
\]

**TEMP** $t$

The initial temperature of this part is $t$ and it can be expressed as a function of $x$, $y$, $z$ coordinates.

**THIC** thic

Plates have the thickness thic for this part.

**VELO** $v_x v_y v_z$

Assign initial rigid body velocity to all nodes within this part. ($v_x$, $v_y$, $v_z$) is the global velocity vector. ($v_x$, $v_y$, $v_z$) can be expressed as a function of $x$, $y$, $z$ coordinates.
5. NASTRAN Part

The NASTRAN part provides for importing NASTRAN input files into LS-INGRID. The form of the part is as follows:

NASTRAN filename
<optional functions>
END

filename is the name of the NASTRAN input file.

5.1 OPTIONS AND FUNCTIONS

Functions require the ability to identify groups of nodes and elements in a part and assign various properties. These have the general form of:

Keyword <region> function data

Where <region> is a part specific description of where the function is to be applied. For the current part, the nodes or elements through either node or element numbers or through analytical expressions. As an example:

SI [mat==2] 1 M       C  Elements of material 2 are assigned to
                     C  the master side of contact interface 1.

Variables available for function application are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y z</td>
<td>Part local coordinates of node or element center.</td>
</tr>
<tr>
<td>xg yg zg</td>
<td>Global coordinates of node or element center.</td>
</tr>
<tr>
<td>node</td>
<td>Node number.</td>
</tr>
<tr>
<td>mat</td>
<td>Material number.</td>
</tr>
<tr>
<td>elem</td>
<td>Element number.</td>
</tr>
</tbody>
</table>

The following options are allowed in any order. Additional functions can be applied and are described in the section on Loads and Boundary Conditions.

COOR nc data  Input nc local coordinate systems. Coordinate system data is described in detail in the section on Coordinate Transformations.
Nodes are converted from cylindrical to rectangular coordinates. The equations for this transformation are:
\[ X = R \cos \theta \]
\[ Y = R \sin \theta \]

Repeat command. This command makes copies of the part in each of the local coordinate systems \( l_1 \) to \( l_n \). If the coordinate system number is zero, the part is repeated with no transformation.

The part has material number matnum.

Repeat command. This command makes copies of the part in each of the global coordinate systems \( l_1 \) to \( l_n \). If the coordinate system number is zero, the part is repeated with no transformation.

Assign an initial rigid body rotation to the part. \((p_x, p_y, p_z)\) is any point on the axis of rotation and \((v_x, v_y, v_z)\) defines the axis direction. The angular velocity is \( w \) in radians per second.

Nodes are converted from spherical to rectangular coordinates. The equations for this transformation are:
\[ X = R \cos \theta \sin \phi \]
\[ Y = R \sin \theta \sin \phi \]
\[ Z = R \cos \phi \]

The initial temperature of this part is \( t \) and it can be expressed as a function of \( x, y, z \) coordinates.

Plates have the thickness thic for this part.

Assign initial rigid body velocity to all nodes within this parts. \((V_x, V_y, V_z)\) is the global velocity vector. \((V_x, V_y, V_z)\) can be expressed as a function of \( x, y, z \) coordinates.
Notes:

1. The following NASTRAN keywords are supported:

   - CBAR
   - CELAS2
   - CONM2
   - CORD1S
   - CORD2S
   - CTETRA
   - GRAV
   - MPC
   - PLOAD2
   - PSOLID
   - SPC

   - CBEAM
   - CHEXA
   - CORD1C
   - CORD2C
   - CPENTA
   - CTRIA3
   - GRID
   - PBAR
   - PLOAD4
   - RBE2
   - SPC1

   - CDAMP
   - CMASS2
   - CORD1R
   - CORD2R
   - CQUAD4
   - FORCE
   - MAT1
   - PBEAM
   - PSHELL
   - SPC

2. The following keywords from MSC/DYNA are also supported:

   - DYMAT24
   - MATRIG

3. To preserve the arbitrary node, element and material numbering of NASTRAN input files, use the ARBITRARY command. (See *LS-DYNA3D Commands and Materials*.)

4. The material properties from the NASTRAN input are generally not used for LS-DYNAxx calculations. To assign properties from LS-INGRID, the materials may be defined either before or after the NASTRAN part. The LS-INGRID material ID's are input the same as those of the NASTRAN input. If the LS-INGRID materials are defined prior to the NASTAN model input, but the sections are not input, then LS-INGRID will try to use the section property data from the NASTRAN input. Section properties may be scaled using the global command: PSCALE.

5. Node, element and material ID's can be shifted using the global command, LABELS.
6. Standard Part

6.1 DEFINITIONS

Index Space: An index space is a three-dimensional discrete coordinate system with integer values greater than or equal to 1 in each of the three directions. The three discrete coordinates are labeled, \( I \), \( J \), and \( K \) axes respectively. Each point in the index space \((i, j, k)\), represents a nodal point. Elements are defined as groups of adjacent nodes in the index space.

Region: A region is any rectangular or cubic block of nodes. A region is usually defined by a block in an index space.

Part: A part is a collection of regions which can be grouped and generated conveniently in an index space. Beginning users will typically use one region per part while more experienced users will be able to group numerous regions together into complex parts.

Model: The final model is a collection of parts. Each part has its own index space and is independent of other parts. Parts are connected together either by global coincident node removal, slide surfaces, or other constraints.

The standard part in LS-INGRID is based on a three-dimensional index space which is commonly used for finite difference mesh generation. Although this can be somewhat awkward for finite element meshes, proper usage technique and some enhancements have made this quite effective for certain geometries including some that are difficult for standard finite element mesh generators. The principal enhancement to the three-dimensional index space is an additional type of index notion, the "Index Progression". Index progressions provide a concise and simple method for describing complex structures, and are used to input data to LS-INGRID. The following is a detailed description of the index space notion and the index progression. This information provides the user with the concepts necessary to use LS-INGRID effectively.

Index Space

Node generation in LS-INGRID is done by a mapping from Index space onto the object of interest as is shown in Figure 6-1. Each region of the object is referenced by a set of six indices; \((I_{MIN}, J_{MIN}, K_{MIN})\) specify the minimum indices for a region in the index space and \((I_{MAX}, J_{MAX}, K_{MAX})\) specify the maximum indices. For a solid region, all eight corner nodes are defined by combinations of minimum and maximum indices. Table 6-1 lists the indices of the vertices in the example of Figure 6-1. We assume that any set of three indices, \((I, J, K)\), defines a region in space.

If \(K_{MIN}\) is set equal to \(K_{MAX}\), the resulting region is a plane of constant \(K\) as shown...
in Figure 6-2a. Similarly, a plane of constant $I$ is defined when $\text{IMIN}$ is set equal to $\text{IMAX}$ and a plane of constant $J$ for $\text{JMIN}$ equal to $\text{JMAX}$. A line in the index space is defined by holding two indices constant while the third index varies as shown in Figure 6-2b.

Figure 6-1. Mapping from index space to object space.
Figure 6-2a. Planes in index space.

Figure 6-2b. Lines in index space.
TABLE 6-1. Indices associated with the vertices of a region.

<table>
<thead>
<tr>
<th>Node</th>
<th>Indices</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>(1, 1, 1)</td>
<td>(IMIN, JMIN, KMIN)</td>
</tr>
<tr>
<td>B</td>
<td>(5, 1, 1)</td>
<td>(IMAX, JMAX, KMAX)</td>
</tr>
<tr>
<td>C</td>
<td>(1, 1, 2)</td>
<td>(IMIN, JMIN, KMIN)</td>
</tr>
<tr>
<td>D</td>
<td>(5, 1, 2)</td>
<td>(IMAX, JMAX, KMAX)</td>
</tr>
<tr>
<td>E</td>
<td>(1, 5, 1)</td>
<td>(IMIN, JMIN, KMIN)</td>
</tr>
<tr>
<td>F</td>
<td>(5, 5, 1)</td>
<td>(IMAX, JMAX, KMAX)</td>
</tr>
<tr>
<td>G</td>
<td>(1, 5, 2)</td>
<td>(IMIN, JMIN, KMIN)</td>
</tr>
<tr>
<td>H</td>
<td>(5, 5, 1)</td>
<td>(IMAX, JMAX, KMAX)</td>
</tr>
</tbody>
</table>

An index space is defined as the set of all possible indices < IMAX, < JMAX, 1 < KMAX. If an index is zero, then it varies over all possible indices. Thus, the indices (3, 0, 2) defines a line which extends across the index space, and (0, 0, 2) defines a plane which divides the index space into two regions. (0, 0, 0) defines the entire index space.

Index Progressions

Index progressions were developed to facilitate the defining of multiple regions in index space. Rather than specifying the minimum and maximum indices for a region, one simply specifies the progression in indices along the I, J, and K directions respectively. For example, the region (2, 7, 6, 8, 9, 10) is represented as the progression (2, 8); (7, 9); (6, 10). If there is a region adjacent to (2, 7, 6, 8, 9, 20) such as (2, 7, 2, 8, 9, 6), the two regions are defined together by a new progression (2, 8); (7, 9); (2, 6, 10). To define the four solids regions shown in Figure 6-3a requires the progression (3, 5, 7); (2, 4, 6); (1, 4).

Index progressions for planes are defined in a similar manner. The index which remains constant throughout a plane is indicated by a negative sign so the plane (2, 5, 2, 7, 8) is represented as (-2); (5, 7); (5, 8).

In Figure 6-3b there are eight planes which can be represented by the progression (3, 5, 7); (2, 4, 6); (-1, -4). The savings by this notation is apparent since specifying separately the eight regions in Figure 6-3b requires 49 number where as the index progression requires only 8 numbers.
Figure 6-3. Index progressions for planes and solids.
Another addition to the index progression notation is the zero index. The two solids regions shown in Figure 6-4 could be represented as an index progression except that they are not connected. In this case, a zero index is used along the I direction to indicate that the structure is discontinuous. This gives the progression (2, 4, 0, 6, 8); (3, 7); (4, 5). Plane regions can be separated by the zero index in a manner similar to solid regions.

More complicated regions can be represented by combining index progressions. An example of this is in Figure 6-5. The open box could be represented by two index progressions (-2, -5); (1, 7); (3, 5); and (2, 5); (1, 7); (-3, -5), but they can also be combined to give (-2, -5); (1, 7); (-3, -5). Figure 6-6 shows several more structures and their index progression representation.

LS-INGRID uses the index progression notation to set up regions in the index space which are to be mapped onto the object of interest. This notation has the advantage that it requires little input data and with less than 20 indices can represent thousands of configurations in index space. In practice, not all configurations in index space can be defined by an index progression so a command is added to allow deletion of regions in the index space. The delete command along with the index progression is enough to produce almost any conceivable region in the index space and is used as the central part of LS-INGRID's mesh generation.
Figure 6-4. Separated solid regions.

Figure 6-5. Open Box.
(a) Intersecting Plates

\((2, -4, 6); (2, -4, 6); (-3, -7)\)

(b) Cube in a Box

\((-2, 4, 6, -8); (-2, 4, 6, -8); (3, 5)\)

(c) Examples of Region Deletion

\((-2, -6, -10); (3, 7); (-2, -4)\)

Deleted Regions:

\((2, 3, 2, 6, 7, 2)\) and \((6, 3, 4, 10, 7, 4)\)

(d) Planes and Solids with Gaps

\((2, 4, 0, 6, -8); (2, 4, 0, 6, 8); (-2, 4, 6, -8)\)

Figure 6-6. Examples of index progressions.
Each part definition consists of the following data:

**START**
Index progression
Part control commands and functions
Loads and Boundary Conditions

**END**

"START" signifies the beginning of a part definition and is require as the first card in each part. Part control commands affect properties of the mesh. Following is a list of the default properties for a part.

- 4 node plate elements
- 8 node solid element
- rectangular coordinates
- material property = 1
- plate thickness = 0.0

The dimension of the index space along with all plane and solid regions are defined by the index progression. Function cards manipulate the mesh defined by the index progression and an "END" signifies the end of a part. Following are some important definitions in addition to those previously given.

**Index Space**

The set of all indices defined by an index progression. For example, the progression (2, 3, -5, 10), (4, 5), (2, 6) defines the index space $2^{|I|}10, 4^{|J|}5, 2^{|K|}6$.

**Reduced Index Space**

The reduced index space references positions in an index progression. The point, $I, J, K$ in the reduced index space refers to the point in the index progression defined by the $I$th integer in the $I$-progression, the $J$th integer in the $J$-progression and the $K$th integer in the $K$-progression. For the progression (2, 3, -5, 10), (4, 5), (2, 6) the relationship between the reduced index space and the index space is shown in Table 6-3.

Unless otherwise noted, all points and regions are defined in the reduced index space. Since the reduced index space is independent of the actual values of the index progression, the mesh can be refined or contracted only by changing the index progression.
Table 6-3. Comparison of the Reduced Index Space and the Index Space for the Index Progression (2, 3, -5, 10), (4, 5), (2, 6)

<table>
<thead>
<tr>
<th>Reduced Index Space</th>
<th>Index Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 1, 1</td>
<td>2, 4, 2</td>
</tr>
<tr>
<td>1, 1, 2</td>
<td>2, 4, 6</td>
</tr>
<tr>
<td>1, 2, 1</td>
<td>2, 5, 2</td>
</tr>
<tr>
<td>2, 2, 2</td>
<td>3, 5, 6</td>
</tr>
<tr>
<td>3, 1, 2</td>
<td>5, 4, 6</td>
</tr>
<tr>
<td>4, 1, 1</td>
<td>10, 4, 2</td>
</tr>
</tbody>
</table>

6.2 Index Progression

Each part must have an index progression. The following input is required.

\[ i_1 \ i_2 \ldots \ i_i \]  \quad \text{Progression in } I \text{ direction}
\[ j_1 \ j_2 \ldots \ j_i \ ; \]  \quad \text{Progression in } J \text{ direction}
\[ k_1 \ k_2 \ldots \ k_i \]  \quad \text{Progression in } K \text{ direction}
\[ x_1 \ x_2 \ldots \ x_i \]  \quad \text{Initial } X\text{-coordinates}
\[ y_1 \ y_2 \ldots \ y_i \]  \quad \text{Initial } Y\text{-coordinates}
\[ z_1 \ z_2 \ldots \ z_i \]  \quad \text{Initial } Z\text{-coordinates}
6.3 Part Commands and Functions

All functions have the following form:

Keyword - index specification - parameters

Index specifications have three types which are abbreviated as <Point>, <Region>, or <Index Progression>. All index specifications are applied in the reduced index space. The input is defined as follows:

<Point>
Input for <point> consists only of the three indices (i,j,k). If any index is input as zero, then the index varies from the smallest to the largest possible value.

<Region>
The function locates the region defined by \((i_m, j_m, k_m, i_x, j_x, k_x)\). If \(i_m, j_m, \) or \(k_m\) is input as zero, the zero index is given the minimum possible value. If \(i_x, j_x, \) or \(k_x\) is input as zero, the zero index is set to the maximum possible value.

<Index Progression>
This is used to define multiple regions according to \(i_1, i_2, i_3, ... ; j_1, j_2, ... ; k_1, k_2, ... ;\)

Functions which use <Region> or <Index Progression> for index specification can be repeated and shifted to other parts of the index space. The general form of these commands is as follows:

Keyword - <Region> or <Index Progression> parameters - first offset - parameters - second offset - parameters.

The offset information is as follows:

+ - or +o
Either a "+" or a "+o" is required as the first information for the offset. If "+" is used then the offset occurs from the region defined by the last offset. If "+O" is used then the offset is relative to the region defined by <Region> or <Index Progression>.

One and only one of the following commands must be input following "+" or "+o".
I $di$  
Increment $I$ indices by $di$.

J $dj$  
Increment $J$ indices by $dj$.

K $dk$  
Increment $K$ indices by $dk$.

IJ $di$ $dj$  
Increment $I$ and $J$ indices by $di$ and $dj$.

JK $dj$ $dk$  
Increment $J$ and $K$ indices by $dj$ and $dk$.

KI $dk$ $di$  
Increment $K$ and $I$ indices by $dk$ and $di$.

IJK $di$ $dj$ $dk$  
Increment $I$, $J$, and $K$ indices by $di$, $dj$, and $dk$.

SIJ  
Switch $I$ indices with $J$ indices.

SJK  
Switch $J$ indices with $K$ indices.

SKI  
Switch $K$ indices with $I$ indices.

A <Region>  
Form a curved edge between nodes A and B. The region is a line in the reduced index space.

ityp  
Flag specifying type of curve.

=1: A parabola through point P1 (See Figure 6-7).
=2: A circular arc through point P1.
=3: A circular arc with center P2.

x  
$X$-coordinate of point P1 or P2

y  
$Y$-coordinate of point P1 or P2

z  
$Z$-coordinate of point P1 or P2

r  
Radius

If the radius is non-zero for a circular arc with center P2, then nodes A and B (See Figure 6-7) are moved radially from P2 until they are a distance equal to the radius from P2. An arc is formed through the nodes at their final location.
Figure 6-7. Curved boundaries.

**A, AE <Region>**

Arc keyword. The region is a plane or a solid in the reduced index space with an arbitrary length.

idir Flag specifying axis of rotation in the index space.

= "I": I-axis is axis of rotation
= "J": J-axis is axis of rotation
= "K": K-axis is axis of rotation

r Radius

For any plane normal to the axis of rotation such as ABCD in Figure 6-8, a point 0 on the axis of rotation is located in the center of the plane. If the radius of the cylinder is not zero, then the points A, B, C and D are moved radially from 0 until they are a distance, R, from point 0. Curved boundaries are then formed for the segments AB, BD, AC and CD using center.0. This is done for each plane normal to the axis of rotation in the reduced index space.
**AC, ACE** <Region>  

<Region> is a surface in the index space.

- **idir**  
  Flag specifying axis of rotation in the index space.  
  - "I": I-axis is axis of rotation  
  - "J": J-axis is axis of rotation  
  - "K": K-axis is axis of rotation

- **p_x p_y p_z**  
  Any point on axis of cylinder (See Figure 6-9).

- **r**  
  Radius of the cylinder.

- **q_x q_y q_z**  
  Any vector parallel to the axis of the cylinder.
Perform automatic smoothing of edges and surfaces which represent continuous surface definitions.

**BG**

Beam generation command.

The BG command permits beam elements to be defined within parts defined using an index space. If only beam elements are desired for the part, then all of the shell and solid elements can be deleted.

**Options:**

- **MT** *m*  
  Beams have material number *m*.

- **SC** *n*  
  Beams have section number *n*.

- **NGEN** *n*  
  Generate *n* beams from point 1 to point 2.

- **N1** <Point>  
  Point 1 is located at <Point>.

- **P1** *p x* *p y* *p z*  
  Point 1 is located at (*p x*, *p y*, *p z*).

- **N2** <Point>  
  Point 2 is located at <Point>.

- **P2** *p x* *p y* *p z*  
  Point 2 is located at (*p x*, *p y*, *p z*).

- **B1** *n*  
  Set boundary code for point 1. *n* is a six digit binary number which specifies degrees of freedom which are to be constrained. Numbering the digits from left to right, they affect the following degree-of-freedom.
1st digit:  $x$-displacement
   $= 0$: free
   $= 1$: fixed
2nd digit:  $y$-displacement
3rd digit:  $z$-displacement
4th digit:  $x$-rotation
5th digit:  $y$-rotation
6th digit:  $z$-rotation

\textbf{V2} $v_x \ v_y \ v_z$

Point 2 is offset from point 1 by the vector $(v_x, v_y, v_z)$.

\textbf{B2} $n$

Set boundary code for point 2. $n$ is has the same meaning as for the "B1" option in this command.

\textbf{NO} <Point>

The point defining the orientation of the local 2-axis is located at <Point>.

\textbf{PO} $p_x \ p_y \ p_z$

The point defining the orientation of the local 2-axis is located at $(p_x, p_y, p_z)$.

\textbf{VO} $v_x \ v_y \ v_z$

The local 2-axis is defined by vector $(v_x, v_y, v_z)$.

\textbf{CO} $p_x \ p_y \ p_z$

Same as "PO" except the point is in cylindrical coordinates.

\textbf{SO} $p_x \ p_y \ p_z$

Same as "PO" except the point is in spherical coordinates.

;  

\textbf{BIAS} $x_0 \ y_0 \ z_0 \ v_x \ v_y \ v_z \ r_0 \ r_1 \ f$

Bias mesh. This command is experimental. $(x_0, y_0, z_0, v_x, v_y, v_z)$ represents a line towards which the elements are biased. A transition distance is defined beginning at $r_0$ and ending at $r_1$ and $f$ is a factor for adjacent element scaling.

\textbf{COOR} nc data

Input nc local coordinate systems. Coordinate system data is described in detail in the section on \textit{Coordinate Transformations}.

\textbf{CPL} <Region>

Center points along line. If <Region> is a line, then this command forces elements to be equally spaced from the beginning point to the ending point. If <Region> is a surface or a volume then the command is subdivided into lines in the direction specified by the direction flag.

dir Direction flag (do not input if <Region> is a line).
   ="I": Equal space along $I$-index
   ="J": Equal space along $J$-index
   ="K": Equal space along $K$-index

\textbf{CYLI}

Nodes are converted from cylindrical to rectangular coordinates. The equations for this
transformation are:
\[ X = R \cos \theta \]
\[ Y = R \sin \theta \]

**D** <Region> or

Region deletion keyword.

**DI** <Index Progression>

**EQSP**

Equal space along arc. This applies to the "AC" and "A" functions.

**FIND** <Point> exp1 exp2 exp3 exp4

The FIND command places the generated coordinates of <Point> into the variables [cenx] [ceny] [cenz] and the node number into [node]. Four expressions must be input as part of this command.

*Example:*

```
FIND 1 2 1 [bp3x=cenx] [bp3y=ceny] [bp3z=cenz] [bp3n=node]
```

**I, J or K** <Point>

Specify independent variable for the function.
- "I": coordinates vary as a function of the I-index.
- "J": coordinates vary as a function of the J-index.
- "K": coordinates vary as a function of the K-index.

**dir2**

Flag specifying which coordinate is modified.
- "X": X-coordinate is modified.
- "Y": Y-coordinate is modified.
- "Z": Z-coordinate is modified.

**c1 c2 c3 ...**

New progression of coordinates along index dir1.

**INT** <Region> s1 s2

Nodes within <Region> lie on the intersection of surface s1 and s2. Surfaces are defined using the "SD" command in the control section. These commands will be generated automatically if two "SF" commands result in an intersection surface in the index space.

**LORI** v_x v_y v_z

Specify local axis for orthotropic shell elements. The vector in the local part system is (v_x, v_y, v_z). The vector (v_x, v_y, v_z) may be specified as a function of the local x,y,z coordinates. For example:
LORI [-y] [x] 0

**LREP** \( l_1 \ l_2 \ ... \ l_n \)

Repeat part command. This command makes copies of the part in each of the local coordinate systems \( l_1 \) to \( l_n \). If the coordinate system number is zero, the part is repeated with no transformation.

**MA** or **MB**

Point functions. These commands are used to modify 1, 2, or 3 coordinates of groups of nodes.

- **<POINT>**
  - For "MA" only.

- **<REGION>**
  - For "MB" only.

\( n \)

Flag indicating which coordinates to change.
- "X": \( x \) coordinate is changed
- "Y": \( y \) coordinate
- "Z": \( z \) coordinate
- "XY": \( x \) and \( y \) coordinates
- "XZ": \( x \) and \( z \) coordinates
- "YZ": \( y \) and \( z \) coordinates
- "XYZ": \( x \), \( y \), and \( z \) coordinates

\( d_x \ d_y \ d_z \)

New coordinates. Only the coordinates required by flag \( n \) need to be input. The new coordinates are added to the old coordinates.

**MATE** \( m \)

The part has material number \( m \).

**MS** <Region>

Apply multiple surface equations to <Region>. This command permits the identification of parallel index planes for the purpose of applying surface equations. The function of this command is similar to the "SF" command; however, this command can result in considerable reduction in input for many common cases.

\( \text{idir} \)

<Region> is divided into a series of parallel planes normal to the axis in index space specified by idir.
- "I": \( I \)-axis
- "J": \( J \)-axis
- "K": \( K \)-axis

Next, one surface equation must be input for each of the index planes in <Region> normal to the specified axis. One of the following options may be used.
Option 1:

\[ sf_1 \]
Data for first surface equation (See Surface Definitions).

\[ sf_2 \]
Data for second surface equation.

Option 2:

**PPX, PPY, or PPZ**
Parallel planes normal to \( x, y, \) or \( z \) axes respectively.

\[ u_1 \ u_2 \ u_3 \ldots \]
The point along the specified axis where the planes intercept. One value must be input for each plane.

Option 3:

**CNSP**

\[ p_x \ p_y \ p_z \]
Center of the spheres.

\[ r_1 \ r_2 \ r_3 \ldots \]
Radii.

Option 4:

**CNCY**

\[ p_x \ p_y \ p_z \]
Any point on the axis of the cylinder.

\[ v_x \ v_y \ v_z \]
Any vector parallel to the axis.

\[ r_1 \ r_2 \ r_3 \ldots \]
Radii.

Option 5:

**PON, POX, POY, POZ**
Planes offset normal or in the \( x, y, \) or \( z \) direction, respectively.

\[ p_x \ p_y \ p_z \]
Any point on the plane.

\[ q_x \ q_y \ q_z \]
Any vector normal to the plane.

\[ o_1 \ o_2 \ o_3 \ldots \]
Offsets in the requested direction.

MT \<Region>\ or

MTI \<Index Progression>\ Signifies material command.

mat Material number.

MTV \( m \ n \) All elements contained within volume definition
m are assigned material number n.

**OR** <Region> Specify orientation of local axes relative to the index space. This is necessary when orthotropic materials are used and/or if 8-node shells are requested.

\[ l_1 \]
Local r axis.

- "I": I-axis
- "J": J-axis
- "K": K-axis

\[ l_2 \]
Local s axis.

- "I": I-axis
- "J": J-axis
- "K": K-axis

**ORDER** d1 d2 d3 Order of writing nodes in index space (d = "I", "J", or "K").

**PA.i.Commands**: PA; or **PB.i.Commands**: PB; Point functions. These commands are used to modify 1, 2, or 3 coordinates of groups of nodes.

**<POINT>** For "PA" only.

**<REGION>** For "PB" only.

\[ n \]
Flag indicating which coordinates to change.

- "X": x coordinate is changed
- "Y": y coordinate
- "Z": z coordinate
- "XY": x and y coordinates
- "XZ": x and z coordinates
- "YZ": y and z coordinates
- "XYZ": x, y, and z coordinates

\[ d_x \ d_y \ d_z \]
New coordinates. Only the coordinates required by flag n need to be input. The old coordinates are replaced by the new coordinates.

**REPE** l1 l2 ...ln Repeat command. This command makes copies of the part in each of the global coordinate systems l1 to ln. If the coordinate system number is zero, the part is repeated with no transformation.

**RES** <Region> Use unequal element spacing.

\[ idir \]
Direction of sides to be operated on in <Region>.

- "I": I-direction
The ratio of the length of one element side to the next element side as the $I$, $J$, or $K$ index increases is $r$.

**REST name**

Restore the nodal coordinates of the existing standard part from file "name". For complicated parts, this can save considerable amounts of computing. See also the SAVE command.

**ROTATION $p_x \ p_y \ p_z \ v_x \ v_y \ v_z \ w$**

Assign an initial rigid body rotation to all parts defined after this command. ($p_x, p_y, p_z$) is any point on the axis of rotation and ($v_x, v_y, v_z$) defines the axis direction. The angular velocity is $w$ in radians per time unit.

**RR <Region>**

Rotate region.

**SAVE name**

Save the nodal coordinates of the existing standard part to file "name". For complicated parts, this can save considerable amounts of computing by using the REST command.

**SF <Region> ityp -or- SFI <Index Progression> ityp**

Surface command. This command allows for the exact equation specification for 3-D surfaces. The command operates by moving nodes from an initial location to the closest point on the surface. Intersections of surfaces in the index space are detected and calculated. Since intersections are rarely unique the user must define initial coordinates which are near the final configuration using the initial coordinates and/or point functions. This is often necessary for LS-INGRID to converge to the correct geometry.

If a part is generated in cylindrical coordinates the surfaces are still assumed to be in rectangular coordinates. This permits non-axisymmetric surfaces to be generated on primarily axisymmetric parts.

ityp= "SD n". If itype = "SD" then the surface is defined using the command "SD" in the control section. The surface name must be input to complete this option.
ityp\o(=,/) "SD". ityp refers to an option in Surface Definitions. See Surface Definitions for the remaining input.

**SFE** <Region> dir ityp -or-

Surface command. These commands are similar to the and SFI commands. The primary difference is that only edges of blocks oriented in direction dir (dir = "I", "J" or "K") are projected onto the surface rather than all nodes within the region.

ityp= "SD n". If itype = "SD" then the surface is defined using the command "SD" in the control section. The surface name must be input to complete this option.

**SFEI** <Index Progression> dir ityp

ityp\o(=,/) "SD". ityp refers to an option in Surface Definitions. See Surface Definitions for the remaining input.

**SFV** <Region> -or-

Surface command. These commands are similar to the SF and SFI commands. The primary difference is that only vertices of blocks are projected to the nearest point on a surface rather than all nodes within a region.

ityp= "SD n". If itype = "SD" then the surface is defined using the command "SD" in the control section. The surface name must be input to complete this option.

**SFVI** <Index Progression>

ityp\o(=,/) "SD". ityp refers to an option in Surface Definitions. See Surface Definitions for the remaining input.

**SPHE**

Nodes are converted from spherical to rectangular coordinates. The equations for this transformation are:

\[ X = R \cos \theta \sin \varphi \]
\[ Y = R \sin \theta \sin \varphi \]
\[ Z = R \cos \varphi \]

**TEMP** \( t \)

The initial temperature of this part is \( t \) and it can be expressed as a function of \( x, y, z \) coordinates.

**THIC** \( t \)

Plates have the thickness \( t \) for this part. The thickness, \( t \), may be specified as a function of the part local coordinates to permit thickness distributions.
**TRI2** $t$

All quadrilateral shell elements in this part will be converted to triangular shells. The attached pressure segments, contact segments, etc. will remain as quadrilaterals.

**TRIA** $t$

All quadrilateral shell elements in this part will be converted to triangular shells. The attached pressure segments, contact segments, etc. will also be converted to triangles.

**VELOCITY** $v_x \ v_y \ v_z$

Assign initial rigid body velocity $(V_x, V_y, V_z)$ to all parts defined after this command. $(V_x, V_y, V_z)$ can be expressed as a function of $x, y, z$ coordinates to allow for velocity distributions.

**VTSP**

Equal space along chord. This applies to the "AC" and "A" functions.
7. Beam Part

Beam generation in LS-INGRID is performed by a special part. The data in the part is as follows:

**BEAM**
Local nodal point input.
0 (zero)
Element generation commands
0 (zero)
Optional functions
END

**Local Node Point Input**

Important vertices are listed in this section. All points in this section are assigned node numbers in the global system and output even if there are no elements connected. Nodes are input sequentially and assigned local node numbers starting from one. These numbers are used later for generating elements.

<table>
<thead>
<tr>
<th><strong>ctype</strong></th>
<th>Coordinate transformation to be performed on nodal coordinates.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&quot;RT&quot;: rectangular coordinates (no transformation)</td>
</tr>
<tr>
<td></td>
<td>&quot;CY&quot;: cylindrical coordinates</td>
</tr>
<tr>
<td></td>
<td>&quot;SP&quot;: spherical coordinates</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>n</strong></th>
<th>$n$ is a six digit binary number which specifies degrees of freedom to be constrained. Numbering the digits from left to right they affect the following degrees of freedom.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1st digit: $x$-displacement</td>
</tr>
<tr>
<td></td>
<td>=0 free</td>
</tr>
<tr>
<td></td>
<td>=1 fixed</td>
</tr>
<tr>
<td></td>
<td>2nd digit: $y$-displacement</td>
</tr>
<tr>
<td></td>
<td>3rd digit: $z$-displacement</td>
</tr>
<tr>
<td></td>
<td>4th digit: $x$-rotation</td>
</tr>
<tr>
<td></td>
<td>5th digit: $y$-rotation</td>
</tr>
<tr>
<td></td>
<td>6th digit: $z$-rotation</td>
</tr>
</tbody>
</table>

| **$x_1$ $y_1$ $z_1$** | Nodal coordinates. |
Element Generation Commands

- **is**: First local node number in a beam sequence.
- **if**: Last local node number in a beam sequence.
- **nel**: Number of elements to be generated from is to if.
- **mat**: Material number for the beams.
- **isect**: Section property number for the beams.
- **normal**: Third local node for defining the orientation of the beams. Note: this node can be moved by the "REPEAT" command and is not necessarily in global coordinates.

### 7.1 OPTIONS AND FUNCTIONS

Functions require the ability to identify groups of nodes and elements in a part and assign various properties. These have the general form of

Keyword <region>  function data

Where <region> is a part specific description of where the function is to be applied. For the current part, the nodes or elements through either node or element numbers or through analytical expressions. As an example:

\[ \text{SI } \text{[mat==2]} \text{ 1 M} \quad \text{C} \quad \text{Elements of material 2 are assigned to the master side of contact interface 1.} \]

Variables available for function application are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y z</td>
<td>Part local coordinates of node or element center.</td>
</tr>
<tr>
<td>xg yg zg</td>
<td>Global coordinates of node or element center.</td>
</tr>
<tr>
<td>node</td>
<td>Node number.</td>
</tr>
<tr>
<td>mat</td>
<td>Material number.</td>
</tr>
<tr>
<td>elem</td>
<td>Element number.</td>
</tr>
</tbody>
</table>

The following options are allowed in any order. Additional functions can be applied and are described in the section on *Loads and Boundary Conditions*.

**COOR nc data**

Input nc global coordinate systems. Coordinate system data is described in detail in the section on *Coordinate Transformations*. 

---

7.2
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| **CYLI** | Nodes are converted from cylindrical to rectangular coordinates. The equations for this transformation are:  
  \[ X = R \cos \theta \]  
  \[ Y = R \sin \theta \] |
| **LREP** \( l_1 l_2 \ldots l_n \) | Repeat command. This command makes copies of the part in each of the local coordinate systems \( l_1 \) to \( l_n \). If the coordinate system number is zero, the part is repeated with no transformation. |
| **REPE** \( l_1 l_2 \ldots l_n \) | Repeat command. This command makes copies of the part in each of the global coordinate systems \( l_1 \) to \( l_n \). If the coordinate system number is zero, the part is repeated with no transformation. |
| **ROTA** \( p_x p_y p_z v_x v_y v_z w \) | Assign an initial rigid body rotation to the part. \((p_x, p_y, p_z)\) is any point on the axis of rotation and \((v_x, v_y, v_z)\) defines the axis direction. The angular velocity is \( w \) in radians per time unit. |
| **SPHE** | Nodes are converted from spherical to rectangular coordinates. The equations for this transformation are:  
  \[ X = R \cos \theta \sin \phi \]  
  \[ Y = R \sin \theta \sin \phi \]  
  \[ Z = R \cos \phi \] |
| **TEMP** \( t \) | The initial temperature of this part is \( t \) and it can be expressed as a function of \( x, y, z \) coordinates. |
| **VELO** \( v_x v_y v_z \) | Assign an initial rigid body velocity to all nodes within this part. \( V_x, V_y, V_z \) is the global velocity vector and it can be expressed as a function of \( x, y, z \) coordinates. |
8. Old Data Part

This part permits the user to input and manipulate models which were generated by other mesh generators. It can also be used to take old finite element models and update them. The data in the part is as follows:

```
OLD
Commands
END
```

The "commands" include the input nodes and elements in the form of tables. These tables may be either in free format or formatted. Before a table is input, a list of keywords is input which tells what the columns correspond to. After the data is input, the part may then be moved or otherwise modified before inclusion with the rest of the LS-INGRID model.

**NODES** \(n\)  \(n\) nodal points are input.

*Options:*

- **BCND**  LS-NIKE3D displacement boundary codes are input.
- **BCNR**  LS-NIKE3D rotational boundary codes are input.
- **BCSP**  SAP boundary codes are input.
- **DUMMY**  Read and ignore this item. (Must be a number).
- **FORM** \(f\)  Nodal points are read using format \(f\). \(f\) is a character string up to 80 characters long which has the correct FORTRAN format. All items must be read in floating point format. No more than one node point can be specified on a card. If this option is not used then nodal point data is input free format.
- **INCLUDE**  Nodes are read from file. This option terminates the NODES command and reads the nodes.
- **K**  Node point increment \(k\) is input.
- **NUMBER**  Node numbers are to be read. If this option is not used, then node numbers are assigned sequentially.
- **T**  Temperature.
| X      | X-coordinate.          |
| Y      | Y-coordinate.          |
| Z      | Z-coordinate.          |
| ;      | Terminate options and read the nodal points. This is done automatically if an include file is specified. |

**BEAMS** \( n \)

\( n \) beam elements are input.

*Options:*

**FORM** \( f \)

Beam elements are read using format \( f \). \( f \) is a character string up to 80 characters long which has the correct FORTRAN format. All items must be read in floating point format. No more than one element can be specified on a card. If this option is not used then nodal point data is input free format.

**NUMBER**

Element numbers are to be read. If this option is not used, then element numbers are assigned sequentially.

**K**

Element increment \( K \) is input.

**MATERIAL**

Material numbers are input.

**SECTION**

Section property numbers are input.

**INCLUDE**

Beam elements are read from file. This option terminates the BEAMS command and reads the beam elements.

**NODES**

Three nodes are input (first node, second node, and node defining local two axis).

\( N1 \)

First node.

\( N2 \)

Second node.

\( N3 \)

Node defining local two axis.

**DUMMY**

Read and ignore this item. (Must be a number).

; 

Terminate options and read the element data.

**SHELLS** \( n \)

\( n \) shell elements are input.

*Options:*

**FORM** \( f \)

Elements are read using format \( f \). \( f \) is a
character string up to 80 characters long which has the correct FORTRAN format. All items must be read in floating point format. No more than one element can be specified on a card. If this option is not used then nodal point data is input free format.

**NUMBER**

Element numbers are to be read. If this option is not used, then element numbers are assigned sequentially.

**K**

Element increment $K$ is input.

**MATERIAL**

Material numbers are input.

**THICKNESS**

Thickness of element.

**INCLUDE**

Shell elements are read from file. This option terminates the SHELLS command and reads the shells.

**NODES**

Four nodes are input.

**N1**

Node 1.

**N2**

Node 2.

**N3**

Node 3.

**N4**

Node 4.

**DUMMY**

Read and ignore this item.

**;**

Terminate options and read the element data.

**BRICKS $n$**

$n$ brick elements are input.

**Options:**

**FORM $f$**

Elements are read using format $f$. $f$ is a character string up to 80 characters long which has the correct FORTRAN format. All items must be read in floating point format. No more than one element can be specified on a card. If this option is not used then nodal point data is input free format.

**NUMBER**

Element numbers are to be read. If this option is not used, then element numbers are assigned sequentially.

**K**

Element increment $K$ is input.

**MATERIAL**

Material numbers are input.
<table>
<thead>
<tr>
<th>INCLUDE</th>
<th>Brick elements are read from <em>file</em>. This option terminates the BRICKS command and reads the brick elements.</th>
</tr>
</thead>
<tbody>
<tr>
<td>NODES</td>
<td>Input 8 node numbers.</td>
</tr>
<tr>
<td>N1</td>
<td>Node 1.</td>
</tr>
<tr>
<td>N2</td>
<td>Node 2.</td>
</tr>
<tr>
<td>N3</td>
<td>Node 3.</td>
</tr>
<tr>
<td>N4</td>
<td>Node 4.</td>
</tr>
<tr>
<td>N5</td>
<td>Node 5.</td>
</tr>
<tr>
<td>N6</td>
<td>Node 6.</td>
</tr>
<tr>
<td>N7</td>
<td>Node 7.</td>
</tr>
<tr>
<td>N8</td>
<td>Node 8.</td>
</tr>
<tr>
<td>DUMMY</td>
<td>Read and ignore this item.</td>
</tr>
<tr>
<td>;</td>
<td>Terminate option and read the element data.</td>
</tr>
</tbody>
</table>
8.1 Options and Functions

Functions require the ability to identify groups of nodes and elements in a part and assign various properties. These have the general form of

Keyword <region>  function data

Where <region> is a part specific description of where the function is to be applied. For the current part, the nodes or elements through either node or element numbers or through analytical expressions. As an example:

\[
\text{SI [mat==2] 1 M} \quad \text{C Elements of material 2 are assigned to C the master side of contact interface 1.}
\]

Variables available for function application are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y z</td>
<td>Part local coordinates of node or element center.</td>
</tr>
<tr>
<td>xg yg zg</td>
<td>Global coordinates of node or element center.</td>
</tr>
<tr>
<td>node</td>
<td>Node number.</td>
</tr>
<tr>
<td>mat</td>
<td>Material number.</td>
</tr>
<tr>
<td>elem</td>
<td>Element number.</td>
</tr>
</tbody>
</table>

The following options are allowed in any order. Additional functions can be applied and are described in the section on Loads and Boundary Conditions.

**COOR** nc data

Input nc local coordinate systems. Coordinate system data is described in detail in the section on Coordinate Transformations.

**CYLI**

Nodes are converted from cylindrical to rectangular coordinates. The equations are:

\[
X = R \cos \theta \\
Y = R \sin \theta
\]

**LREP** \(l_1 \ l_2 \ldots \ l_n\)

Repeat command. This command makes copies of the part in each of the local coordinate systems \(l_1\) to \(l_n\). If the coordinate system number is zero, the part is repeated with no transformation.

**MATE** matnum

The part has material number matnum.

**REDUCE**

Eliminate unattached nodes which are input in this part.
Repeat command. This command makes copies of the part in each of the global coordinate systems $l_1$ to $l_n$. If the coordinate system number is zero, the part is repeated with no transformation.

Assign an initial rigid body rotation to the part. $(p_x, p_y, p_z)$ is any point on the axis of rotation and $(v_x, v_y, v_z)$ defines the axis direction. The angular velocity is $w$ in radians per second.

Nodes are converted from spherical to rectangular coordinates. The equations are:

\[
\begin{align*}
X &= R \cos \theta \sin \phi \\
Y &= R \sin \theta \sin \phi \\
Z &= R \cos \phi
\end{align*}
\]

The initial temperature of this part is $t$ and it can be expressed as a function of $x, y, z$ coordinates.

Plates have the thickness $thic$ for this part.

Assign initial rigid body velocity to all nodes within this part. $(V_x, V_y, V_z)$ is the global velocity vector. $(V_x, V_y, V_z)$ can be expressed as a function of $x, y, z$ coordinates.
9. MAZE Part

MAZE parts provide simple methods for generating two dimensional cross sections. These sections can then be used as shell elements or as 3-D solids using drag mesh operations. The data in the part is as follows:

\begin{verbatim}
PART
  Required part data (9.1)
  Optional part control commands (9.2)
  Optional functions (9.3)
END
\end{verbatim}

9.1 Required Part Data

Each MAZE part requires a set of line definitions followed by a material number and mesh density information. There are many possible methods for describing MAZE parts as:

\[ L_1 \ L_2 \ L_3 \ L_4 \ \text{mt} \ k \ m \]
Define four sided region edges consisting of the intersection lines \( L_1, L_2, L_3, \) and \( L_4 \). This region will have material name \( \text{mt} \) and will be subdivided in a \( k \times m \) element mesh with \( k \) elements lying along edges \( L_1 \) and \( L_3 \) and \( m \) elements lying along edges \( L_2 \) and \( L_4 \). Edges must be listed in a counterclockwise order. If \( k \) or \( m \) are zero, the number of elements are assumed to be one less than the number of points in lines \( L_1 \) or \( L_2 \), respectively. Points defining the lines then become nodal coordinates.

\[ L_1 \ L_2 \ L_3 \ \text{mt} \ k \ m \]
Define three sided region having edges consisting of the intersecting lines \( L_1, L_2, \) and \( L_3 \). This region will have material name \( \text{mt} \) and will be subdivided into \( m \times (2k+m) \) elements with \( k+m \) elements along edges \( L_1 \) and \( L_2 \) and \( 2m \) elements along edge \( L_3 \). Edges must be listed in a counterclockwise order.

\[ L_1 \ L_2 \ L_3 \ 0 \ \text{mt} \ k \ m \]
Define three sided region having edges consisting of the intersecting lines \( L_1, L_2, \) and \( L_3 \). This region will have material name \( \text{mt} \) and will be subdivided into \( m \times (2k+m) \) elements with \( k+m \) elements along edges \( L_1 \) and \( L_2 \) and \( 2m \) elements along edge \( L_3 \). Edges must be listed in a counterclockwise order.
Define four sided region as described above but with variable zoning. Parameters \( r_1 \) and \( r_2 \) are the ratios of the first segment length to the last segment length along edges 1,3, and 2,4 respectively.

Define four sided region as described above but with variable zoning. Parameters \( r_1 \) to \( r_4 \) are the ratios of the first segment length to the last segment length along edges 1 to 4, respectively.

Define four sided region as described above but with a specified number of elements between consecutive points defining the lines whose line numbers are proceeded by a minus sign. For this option to work properly, the first intersection point must lie either on the first and second point of the line being subdivided. The total number of points used to define the line is equal to \( p \). If desired, not all subdivisions need to be defined. For example, if it is desired to specify the number of subdivisions between the first three points of the first line type. The other segments are equally spaced over the balance of the line.
9.2 Options and Functions

The following part control commands are allowed.

**COOR nc data**

Input nc local coordinate systems. Coordinate system data is described in detail in the section on Coordinate Transformations.

**DRAG**

Perform a drag mesh operation to make solid elements from plane elements.

*Options:*

**MOVE n data**

Form n layers of solid elements by moving the original plane elements to the new location specified by data. Data is described in detail in Coordinate Transformations.

**ROTA n p x p y p z q x q y q z α**

Form n layers of solid elements by rotating the original plane elements about an axis. \((p_x, p_y, p_z)\) is any point on the axis of rotation and \((q_x, q_y, q_z)\) is a vector parallel to the axis. The angle of rotation in degrees is \(α\).

**RES r**

The ratio of one element length to the next is \(r\). This applies only to the previous drag operation.

**LREP l 1 l 2 ... l n**

Repeat command. This command makes copies of the part in each of the local coordinate systems \(l_1\) to \(l_n\). If the coordinate system number is zero, the part is repeated with no transformation.

**REPE l 1 l 2 ... l n**

Repeat command. This command makes copies of the part in each of the global coordinate systems \(l_1\) to \(l_n\). If the coordinate system number is zero, the part is repeated with no transformation.

**ROTA p x p y p z w x w y w z**

Assign an initial rigid body rotation to the part. \((p_x, p_y, p_z)\) is any point on the axis of rotation and \((w_x, w_y, w_z)\) is the rotation vector in radians per second.

**SPIN n θ**

Perform spin operation. The number of layers of nodes is \(n\) and the total angle of the part is \(q\) in degrees.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STACK $n q$</td>
<td>Perform stack operation. The number of layers of nodes is $n$ and the total length of the part is $\theta$.</td>
</tr>
<tr>
<td>TEMP $t$</td>
<td>The initial temperature of this part is $t$ and it can be expressed as a function of $x$, $y$, $z$ coordinates.</td>
</tr>
<tr>
<td>THICK $t$</td>
<td>Plates have the thickness $t$ for this part.</td>
</tr>
<tr>
<td>TRI2 $t$</td>
<td>All quadrilateral shell elements in this part will be converted to triangular shells. The attached pressure segments, contact segments, etc. will remain as quadrilaterals.</td>
</tr>
<tr>
<td>TRIA $t$</td>
<td>All quadrilateral shell elements in this part will be converted to triangular shells. The attached pressure segments, contact segments, etc. will also be converted to triangles.</td>
</tr>
<tr>
<td>VELO $v_x v_y v_z$</td>
<td>Assign initial rigid body velocity to all nodes within this part. $(V_x, V_y, V_z)$ is the global velocity vector and it can be expressed as a function of $x$, $y$, $z$ coordinates.</td>
</tr>
</tbody>
</table>
9.3 Functions

All MAZE part functions have the following form:

**Keyword** - index specification - parameters

MAZE parts have one type of index specification which is abbreviated as <Mregion>. The input for this index specification is as follows:

\[ c_1 \quad c_2 \quad p_1 \quad p_2 \]

Four indices can identify any vertex, edge, or surface in the MAZE part. Each MAZE part has either 3 or 4 corners. The first corner is the intersection of the first line and the last line that makes up the part. The second corner is the intersection of the first and second lines. Further corners are defined similarly around the part. The part also has several planes including the original cross section and one more plane for each drag operation. The first corner node reference by <Mregion> is \( c_1 \) and the last corner is \( c_2 \). The first plane is \( p_1 \) and the last plane is \( p_2 \). If \( c_1 \) or \( c_2 \) is zero, they take on the minimum and maximum corner numbers respectively. Similarly, if \( p_1 \) or \( p_2 \) is zero, they are assigned the minimum and maximum plane numbers, respectively.
10. EDIT Part

The EDIT part allows loads and boundary conditions to be applied to previously defined parts. It also provides for the performing of system assembly operation from subsystems using system joint commands. The general form of this part is:

EDIT filename
<optional functions>
END

10.1 OPTIONS AND FUNCTIONS

Functions require the ability to identify groups of nodes and elements in a part and assign various properties. These have the general form of

Keyword <region> function data

Where <region> is a part specific description of where the function is to be applied. For the current part, the nodes or elements through either node or element numbers or through analytical expressions. As an example:

SI [mat==2] 1 M C Elements of material 2 are assigned to
C the master side of contact interface 1.

Variables available for function application are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y z</td>
<td>Part local coordinates of node or element center.</td>
</tr>
<tr>
<td>xg yg zg</td>
<td>Global coordinates of node or element center.</td>
</tr>
<tr>
<td>node</td>
<td>Node number.</td>
</tr>
<tr>
<td>mat</td>
<td>Material number.</td>
</tr>
<tr>
<td>elem</td>
<td>Element number.</td>
</tr>
</tbody>
</table>

The following options are allowed in any order. Additional functions can be applied and are described in the section on Loads and Boundary Conditions.

COOR nc data Input nc local coordinate systems. Coordinate system data is described in detail in the section on Coordinate Transformations.

CYLI Nodes are converted from cylindrical to
rectangular coordinates. The equations for this transformation are:
\[ X = R \cos \theta \]
\[ Y = R \sin \theta \]

**LREP** \(l_1 \ l_2 \ ... \ l_n\)  Repeat command. This command makes copies of the part in each of the **local** coordinate systems \(l_1\) to \(l_n\). If the coordinate system number is zero, the part is repeated with no transformation.

**MATE** matnum  The part has material number matnum.

**REPE** \(l_1 \ l_2 \ ... \ l_n\)  Repeat command. This command makes copies of the part in each of the **global** coordinate systems \(l_1\) to \(l_n\). If the coordinate system number is zero, the part is repeated with no transformation.

**ROTA** \(px \ py \ pz \ vx \ vy \ vz \ w\)  Assign an initial rigid body rotation to the part. \((px, py, pz)\) is any point on the axis of rotation and \((vx, vy, vz)\) defines the axis direction. The angular velocity is \(w\) in radians per second.

**SPHE**  Nodes are converted from be to rectangular coordinates. The equations for these transformation are:
\[ X = R \cos \theta \sin \phi \]
\[ Y = R \sin \theta \sin \phi \]
\[ Z = R \cos \phi \]

**TEMP** \(t\)  The initial temperature of this part is \(t\) and it can be expressed as a function of \(x, y, z\) coordinates.

**THIC** thic  Plates have the thickness thic for this part.

**VELO** \(v_x \ v_y \ v_z\)  Assign initial rigid body velocity to all nodes within this parts. \((V_x, V_y, V_z)\) is the global velocity vector. \((V_x, V_y, V_z)\) can be expressed as a function of \(x, y, z\) coordinates.
11. DYNA3D Part

The DYNA3D part provides for importing existing DYNA3D input files into LS-INGRID. The form of the part is as follows:

For DYNA3D or LS-DYNA3D input files, use:

```
DYNA3D filename
<optional functions>
END
```

For VEC-DYNA3D input files, use:

```
VECDYNA filename
<optional functions>
END
```

filename is the name of the DYNA3D input file.

11.1 OPTIONS AND FUNCTIONS

Functions require the ability to identify groups of nodes and elements in a part and assign various properties. These have the general form of

```
Keyword <region> function data
```

Where <region> is a part specific description of where the function is to be applied. For the current part, the nodes or elements through either node or element numbers or through analytical expressions. As an example:

```
SI [mat==2] 1 M
```

Elements of material 2 are assigned to the master side of contact interface 1.

Variables available for function application are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y z</td>
<td>Part local coordinates of node or element center.</td>
</tr>
<tr>
<td>xg yg zg</td>
<td>Global coordinates of node or element center.</td>
</tr>
<tr>
<td>node</td>
<td>Node number.</td>
</tr>
<tr>
<td>mat</td>
<td>Material number.</td>
</tr>
<tr>
<td>elem</td>
<td>Element number.</td>
</tr>
</tbody>
</table>

The following options are allowed in any order. Additional functions can be applied and are described in the section on Loads and Boundary Conditions.

```
COOR nc data
```

Input nc local coordinate systems. Coordinate
system data is described in detail in the section on Coordinate Transformations.

**CYLI**

Nodes are converted from cylindrical to rectangular coordinates. The equations for this transformation are:

\[
X = R \cos \theta \\
Y = R \sin \theta
\]

**LREP \( l_1 \ l_2 \ \ldots \ l_n \)**

Repeat command. This command makes copies of the part in each of the local coordinate systems \( l_1 \) to \( l_n \). If the coordinate system number is zero, the part is repeated with no transformation.

**MATE \( \text{matnum} \)**

The part has material number \( \text{matnum} \).

**REPE \( l_1 \ l_2 \ \ldots \ l_n \)**

Repeat command. This command makes copies of the part in each of the global coordinate systems \( l_1 \) to \( l_n \). If the coordinate system number is zero, the part is repeated with no transformation.

**ROTA \( p_x \ p_y \ p_z \ v_x \ v_y \ v_z \ w \)**

Assign an initial rigid body rotation to the part. \((p_x, p_y, p_z)\) is any point on the axis of rotation and \((v_x, v_y, v_z)\) defines the axis direction. The angular velocity is \( w \) in radians per second.

**SPHE**

Nodes are converted from spherical to rectangular coordinates. The equations for this transformation are:

\[
X = R \cos \theta \sin \phi \\
Y = R \sin \theta \sin \phi \\
Z = R \cos \phi
\]

**TEMP \( t \)**

The initial temperature of this part is \( t \) and it can be expressed as a function of \( x, y, z \) coordinates.

**THIC \( \text{thic} \)**

Plates have the thickness \( \text{thic} \) for this part.

**VELO \( v_x \ v_y \ v_z \)**

Assign initial rigid body velocity to all nodes within this parts. \((V_x, V_y, V_z)\) is the global velocity vector. \((V_x, V_y, V_z)\) can be expressed as a function of \( x, y, z \) coordinates.
12. Loads and Boundary Conditions

Loads and boundary can be applied as optional functions within any of the previously describe parts. The syntax of <Region> can be slightly different depending on which part it is applied in. Refer to the appropriate part for a description of the meaning of <Region>. Alternately, <Region> may be an expression. The local part coordinates for a node are stored as internal variables, $x$, $y$ and $z$. The current global coordinates of the same node are $x_g$, $y_g$ and $z_g$. The node number is stored as variable node. Thus, to create slave nodes for sliding interface 6, the following command may be used:

```
SFC [(node<=55)&&(x>5.0)] 6
```

**ACC** <Region> lc amp $f_x f_y f_z$  
Acceleration boundary condition. The load curve number is lc. amp is a scale factor and $(f_x, f_y, f_z)$ indicates the load direction.

**ACCI** <Index Progression> lc amp $f_x f_y f_z$
Definition of accelerometer acc_name. The local node numbers are 1 through 3.

**ACCE** acc_name local_node

**Options:**

**N** <Point>
The local node is defined by <Point>.

**P** $p_x p_y p_z m$
The local node is at point $(p_x, p_y, p_z)$ in the local coordinate system. $m$ is the rigid body number which is attached to the node.

**B** $n$
The local joint node has boundary constraint $n$. $n$ is a six digit binary number which specifies degrees of freedom to be constrained. Numbering digits from left to right they affect the following degrees of freedom.

1st digit: $x$-displacement
   =0 free
   =1 fixed
2nd digit: $y$-displacement
3rd digit: $z$-displacement
4th digit: $x$-rotation
5th digit: $y$-rotation
6th digit: $z$-rotation

```
;
```

**B** <Region> code
Boundary condition keyword. Code is a six digit binary number which specifies degrees of freedom to be constrained. Numbering the digits from left to right they affect the following
LOADS AND BOUNDARY CONDITIONS

degrees of freedom
1st digit: x-displacement
   =0: free
   =1: fixed
2nd digit: y-displacement
3rd digit: z-displacement
4th digit: x-rotation
5th digit: y-rotation
6th digit: z-rotation

BELT type belt_name local_node

Seat belt definition. This command identifies local node number local_node for item belt_name. Values for type are as follows:
   =RETRACTOR: Retractor definition.
   =SENSOR: Sensor definition.
   =SLIPRING: Slipring definition.

Options:

N <Point>
The local node is defined by <Point>.

P p_x p_y p_z m
The local node is at point (p_x,p_y,p_z) in the local coordinate system. m is the rigid body number which is attached to the node.

B n
The local node has boundary constraint n. n is a six digit binary number which specifies degrees of freedom to be constrained. Numbering digits from left to right they affect the following degrees of freedom.
   1st digit: x-displacement
   =0: free
   =1: fixed
   2nd digit: y-displacement
   3rd digit: z-displacement
   4th digit: x-rotation
   5th digit: y-rotation
   6th digit: z-rotation

; Terminate BELT command.

CNV <Region> icv p_x p_y p_z -or-
CNVI <Index Progression> icv p_x p_y p_z
Control volume. This command defines segments for control volume number icv. The segments are facing towards point (p_x,p_y,p_z).
(MVMA/DYNA3D, LS-910 and later).

CSE <Point> n
Identify elements for LS-DYNA3D cross-section resultant force calculations on interface n.

PO i j k
Use the element offset from <Point> by (i,j,k).

RA <Region>
Ignore <Point> and grab the elements identified
RO im jm km ix jx kx

;  

CSN <Region> n

Identify nodes for LS-DYNA3D cross-section resultant force calculations on interface n.

CSY <Region> side -or- CSYI <Index Progression> side

Cyclic symmetry interface nodes. side can be MASTER or SLAVE. (VEC/DYNA3D, LS-920 and later).

CV <Region> lc1 h lc2 Tinfa -or- CVI <Index Progression> lc1 h lc2 Tinfa

Convection boundary condition for surface segments. lc1 is the load curve for the convection coefficient with scale factor h. lc2 is the load curve for the ambient temperature with scale factor Tin. a is the exponent in the equation q = h(T-Tinf)^a.

CVL <Region> lc1 h lc2 Tinfa

Convection boundary condition for edge segments.

EDR i j k n

Identify elements for deletion during restart number n. (i,j,k) are the minimum indices of the region in which elements are to be deleted.

Options:

RO im jm km ix jx kx

The elements to be deleted are the absolute indices im < i < iX, jm < j < jX, km < k < kX, but offset by (i,j,k).

;  

Terminates this function.

EPB <Point>

Element Print Blocks.

PO i j k

Use the element offset from <Point> by (i,j,k).

RO im jm km ix jx kx

Use the block of elements offset from <Point>.

Terminates this command.

FC <Region> lc amp fx fy fz -or- FCI <Index Progression> lc amp fx fy fz

Point force. The load curve number is lc. amp is a scale factor and (fx, fy, fz) indicates the load direction.

FD <Region> lc amp fx fy fz -or- FDI <Index Progression> lc amp fx fy fz

Displacement boundary condition. The load curve number is lc. amp is a scale factor and (fx, fy, fz) indicates the load direction.
**FIND** <Point> expr1 expr2 expr3 expr4  

The FIND command places the generated coordinates of <Point> into the variables [cenx] [ceny] [cenz] and the node number into [node]. Example:

\[
\text{FIND 1 2 1 [bp3x=cenx] [bp3y=ceny] [bp3z=cenz] [bp3n=node]}
\]

**FL** <Region> lc scal  
**FLI** <Index Progression> lc scal  

Flux boundary condition.

**FN** <Region> e_{fail}  

All nodes within <Region> are failure nodes and will fail at strain $e_{\text{fail}}$. Additional nodes are created, shell elements, pressure surfaces, and slide surfaces are renumbered to permit independent motion of adjacent elements (LS-DYNA3D).

Failure nodes are a simple method for allowing fracture. Each adjacent element has completely independent nodes. Groups of nodes are initially constrained to move together. When the average strain of adjacent elements reaches the failure strain, the constraint is eliminated and the elements separate.

**FRV** <Region> lc amp $v_x$ $v_y$ $v_z$  

Set rotational velocity boundary conditions.

**FT** <Region> lc T  
**FTI** <Index Progression> lc T  

Set temperature boundary condition to T and scale by load curve lc.

**FTB** <Region> lc T Tbase  
**FTBI** <Index Progression> lc T Tbase  

Set temperature boundary condition to T and scale by load curve lc. The scaling is:

\[(T-T_{\text{base}}) \times f(lc,\text{time}).\]

**FV** <Region> lc amp $f_x$ $f_y$ $f_z$  
**FVI** <Region> lc amp $f_x$ $f_y$ $f_z$  

Velocity boundary condition. The load curve number is lc. amp is a scale factor and $(f_x,f_y,f_z)$ indicates the load direction.

**GEOC** <Region> igeo  

Geometric contact slave nodes. All identified nodes are slaved to geometric contact entity igeo. (LS-910 and later.)

**IN** <Region> name  
**INI** <Index Progression> name  

Define nodes associated with component interface name. (LS-920 and later.)

**IS** <Region> name  
**ISI** <Index Progression> name  

Define segments associated with component interface name. (LS-920 and later.)

**JOY** <Region>  

Identify JOY interface nodes. (JOY is an
JOYI <Index Progression>  Eulerian hydrocode which exists only at Lawrence Livermore National Laboratory.

JT  Joint command.

\( jn \)  Joint definition number.

\( l \)  Local node number (See Figure 2-1).

Options:

- **N** <Point>  The local joint node, \( l \), is defined by <Point> in the index space.

- **P** \( p_x p_y p_z m \)  The local joint node, \( l \), is at point \((p_x,p_y,p_z)\) in the local coordinate system. \( m \) is the rigid body number which is attached to the node.

- **INC** \( i \)  Increment \( jn \) by \( i \) for each copy of the part. (default = 1).

- **B** \( n \)  The local joint node has boundary constraint \( n \). \( n \) is a six digit binary number which specifies degrees of freedom to be constrained. Numbering digits from left to right they affect the following degrees of freedom:
  - 1st digit: \( x \)-displacement
    - =0: free
    - =1: fixed
  - 2nd digit: \( y \)-displacement
  - 3rd digit: \( z \)-displacement
  - 4th digit: \( x \)-rotation
  - 5th digit: \( y \)-rotation
  - 6th digit: \( z \)-rotation

;  Terminate joint command.

MKI, MKI+, MKI-  Surfaces in <Region> or <Index Progression> are assigned marked surface number \( m \).

MK, MK+, MK-  <Region> \( m p_x p_y p_z \)  For MK+ and MKI+ the surface points toward this point. For MK- and MKI- the surface points away from this point. Do not input \((p_x,p_y,p_z)\) for MK or MKI.

MKL <Region> \( m \)  Identify marked line number \( m \).

N+ <Region> \( p_x p_y p_z \)  -or-  Shell normal orientation command. \((p_x,p_y,p_z)\) is a vector along the element normal vector.

NI+ <Region> \( p_x p_y p_z \)

N- <Region> \( p_x p_y p_z \)  -or-  Shell normal orientation command. \((p_x,p_y,p_z)\) is a vector reverse to the element normal vector.

NI- <Region> \( p_x p_y p_z \)
LOADS AND BOUNDARY CONDITIONS

NPB <Point>

Nodal Print Block.

Options:

PO i j k

Use the node offset from <Point> by (i,j,k).

RO i_m j_m k_m i_x j_x k_x

Use the block of nodes offset for <Point>.

;

Terminate this command.

NFG <Region> name

or-

NFGI <Index Progression> name

Identify nodal force group name.

NRB <Region> -or-

NRBI <Index Progression>

Identify non-reflecting boundaries.

ORV

Define orientation vectors. The orientation vectors are used to specify which axis is to be used for determining the effects of springs and dampers. This particular option is used when two nodes are required for specifying an axis slaved to a body.

jn

Orientation vector number.

l

Local node number (either 1 or 2).

Options:

N <Point>

The local orientation vector node, l, is defined by <Point> in the index space.

P p_x p_y p_z m

The local orientation vector node, l, is at point (p_x,p_y,p_z) in the local coordinate system. m is the rigid body number which is attached to the node.

INC i

Increment jn by i for each copy of the part. (default = 1).

B n

The local orientation vector node has boundary constraint n. n is a six digit binary number which specifies degree of freedom to be constrained. Numbering digits from left to right they affect the following degrees of freedom.

1st digit: x-displacement
=0: free
=1: fixed

2nd digit: y-displacement

3rd digit: z-displacement

4th digit: x-rotation

5th digit: y-rotation

6th digit: z-rotation

;

Terminate ORV command.
<table>
<thead>
<tr>
<th>LS-INGRID</th>
<th>LOADS AND BOUNDARY CONDITIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PM</strong> &lt;Region&gt; ( m )</td>
<td>All nodes within &lt;Region&gt; have mass ( m ) attached to them.</td>
</tr>
<tr>
<td><strong>PR</strong> &lt;Region&gt; ( -or- )</td>
<td>Signifies pressure load command for surface segments.</td>
</tr>
<tr>
<td><strong>PRI</strong> &lt;Index Progression&gt;</td>
<td>Load curve or load curve number.</td>
</tr>
<tr>
<td>( lc )</td>
<td>Pressure magnitude. Spatial variations may be obtained by inputting ( p ) as a function of global coordinates ((x,y,z)).</td>
</tr>
<tr>
<td>( a_x \ a_y \ a_z )</td>
<td>((a_x,a_y,a_z)) is a point in the local coordinate system toward which the pressure acts. By specifying ((a_x,a_y,a_z)) LS-INGRID knows in which direction the pressure is acting and numbers the pressure card node accordingly.</td>
</tr>
<tr>
<td><strong>PRL</strong> &lt;Region&gt;</td>
<td>Signifies pressure load command for edge segments.</td>
</tr>
<tr>
<td>( lc )</td>
<td>Load curve or load curve number.</td>
</tr>
<tr>
<td>( p )</td>
<td>Pressure magnitude.</td>
</tr>
<tr>
<td>( a_x \ a_y \ a_z )</td>
<td>((a_x,a_y,a_z)) is a point in the local coordinate system toward which the pressure acts. By specifying ((a_x,a_y,a_z)) LS-INGRID knows in which direction the pressure is acting and numbers the pressure card node accordingly.</td>
</tr>
<tr>
<td><strong>RB</strong> &lt;Region&gt; ( lc1 \ f lc2 ( t_{inf} ) ( -or- )</td>
<td>Radiation boundary condition.</td>
</tr>
<tr>
<td><strong>RBI</strong> &lt;Index Progression&gt; ( lc1 \ f lc2 ( t_{inf} )</td>
<td>Assign nodes to rigid body node set \textit{set name}.</td>
</tr>
<tr>
<td><strong>RBN</strong> &lt;Region&gt; ( set_name )</td>
<td>Assign nodes to rigid body node set \textit{set name}.</td>
</tr>
<tr>
<td><strong>RE</strong> &lt;Region&gt; ( lc1 \ T ) ( flag \ ( -or- )</td>
<td>Define a radiation enclosure. ( lc1 ) is a load curve for emissivity. The temperature of the segment is ( T ) if ( lc1=0 ). flag is &quot;YES&quot; if this surface is to be included in obstructing surface calculations and &quot;NO&quot; otherwise.</td>
</tr>
<tr>
<td><strong>RE+</strong> &lt;Region&gt; ( lc1 \ T ) ( flag \ ( p_x \ p_y \ p_z ) ( -or- )</td>
<td>Define a radiation enclosure. ( lc1 ) is a load curve for emissivity. The temperature of the segment is ( T ) if ( lc1=0 ). flag is &quot;YES&quot; if this surface is to be included in obstructing surface calculations and &quot;NO&quot; otherwise.</td>
</tr>
<tr>
<td><strong>RE-</strong> &lt;Region&gt; ( lc1 \ T ) ( flag \ ( p_x \ p_y \ p_z )</td>
<td>Define a radiation enclosure. ( lc1 ) is a load curve for emissivity. The temperature of the segment is ( T ) if ( lc1=0 ). flag is &quot;YES&quot; if this surface is to be included in obstructing surface calculations and &quot;NO&quot; otherwise.</td>
</tr>
</tbody>
</table>
and "NO" otherwise. \((p_x,p_y,p_z)\) is a point in the local coordinate system toward which the radiation occurs (RE+), or facing the opposite direction (RE-).

**RXN <Region> mat -or- RXNI <Index Progression> mat**

Extra nodes for rigid body of material mat.

**SBI <Region>**

Define a shell/brick interface. <Region> must be a point or a line in the index space.
- \(="b"\): We are identifying nodes on the brick side of the interface.
- \(="s"\): We are identifying nodes on the shell side of the interface.

**idir**

Nodes on an interface are in a line parallel to axis idir.

- \(=I\): I-axis
- \(=J\): J-axis
- \(=K\): K-axis

**ioff1 ioff2**

Increments for determining the nodes to be selected along direction dir.

**SC <Region> idir{options}**

ALE smoothing constraints. idir Smooth constraints are generated along the line defined by axis idir.

- \(=I\): I-axis
- \(=J\): J-axis
- \(=K\): K-axis

**Options:**

1. **1A i j k**

The first point of the smoothing constraint is located at absolute indices (i,j,k).

2. **1R i j k**

The first point of the smoothing constraint is located at absolute indices (i,j,k).

3. **2A i j k**

The last point of the smoothing constraint is located at absolute indices (i,j,k).

4. **2R i j k**

The last point of the smoothing constraint is located at absolute indices (i,j,k).

5. **PRE**

Constraints are applied before ALE iterative smoothing is done. (The default requires that the constraints be performed after the smoothing is done.)

6. **;**

Terminate smoothing constraint command.
**LS-INGRID LOADS AND BOUNDARY CONDITIONS**

**SFC <Region> n**
Identify slave nodes for sliding interface *n*. This is used for interfaces which involve nodes impacting surfaces or to make more precise distinctions between master and slave sides for the merging algorithms.

**SI <Region> -or- SII <Index Progression>**
Identify sliding interfaces.

islid
Sliding interface number

mslid
Master/slave flag

*"M":* master surface  
*"S":* slave surface

**SI+ <Region> -or- SII+ <Index Progression>**
Identify sliding interface.

islid1
Sliding interface number

mslid
Master/slave flag

*"M":* master surface  
*"S":* slave surface

*p x p y p z*
A point in the local coordinate system toward which the sliding interface faces.

**SI- <Region> -or- SII- <Index Progression>**
Signifies sliding interface command.

islid1
Sliding interface number.

mslid
Master/slave flag

*"M":* master surface  
*"S":* slave surface

*p x p y p z*
A point in the local coordinate system which the sliding interface faces away from.

**SL <Region> n isid**
Define nodes on slide line *n*. <Region> should be a line in the index space. *isid* is either "master" or "slave". This command is sometimes useful in conjunction with SI to fix node tolerance problems.

**SPC <Region> name xyzxyz -or-**
Single point constraints to plane *name*. *xyzxyz* is a binary number which is zero for an unconstrained degree-of-freedom and 1 for a constrained degree-of-freedom. The left three digits are for the translational dof's and the right three are for the rotational.

**SPDP <Region> {options} n1 isid**
Define springs or dampers on all nodes within...
<Region>. They behave according to spring/damper definition \( n \). \( \text{isd} \) is used to force nodes to be on opposite side of the definition. \( \text{isd} = "m" \) for the master side and "s" for the slave side. {options} are as follows:

**ORV \( n \)**
This spring/damper acts along orientation vector \( n \).

**POFF**
Turn element printing off (default).

**PON**
Turn element printing on.

**SCAL \( s \)**
The spring/damper force is scaled by \( s \).

**SW <Region> \( n \) -or- SWI <Index Progression> \( n \)**
Slave nodes to stonewall number \( n \).

**SYSJ**
Joint command.

\( jn \)
Joint definition name.

\( l \)
Local node number. Nodes 1 through 3 define the local system for the master side of a joint. Nodes 4 through 6 define the slave side of a joint.

**Options:**

**INC \( i \)**
Increment \( jn \) by \( i \) for each copy of the part. (default = 1).

**B \( n \)**
The local joint node has boundary constraint \( n \). \( n \) is a six digit binary number which specifies degrees of freedom to be constrained. Numbering digits from left to right they affect the following degrees of freedom.

1st digit: \( x \)-displacement  
   =0: free  
   =1: fixed  
2nd digit: \( y \)-displacement  
3rd digit: \( z \)-displacement  
4th digit: \( x \)-rotation  
5th digit: \( y \)-rotation  
6th digit: \( z \)-rotation

**N <Point>**
The local joint node, \( l \), is defined by <Point> in the index space.

**P \( p_x p_y p_z m \)**
The local joint node, \( l \), is at point \((p_x, p_y, p_z)\) in the local coordinate system. \( m \) is the rigid body number which is attached to the node.
**SYSTEM name**

The local joint node, \( l \), is defined for system name. (Default is the current active system.)

; Terminate joint command.

**TH** <Region> -or- **THI** <Index Progression> thick

Specify thickness command.

**TM** <Region> \( t \) -or- **TMI** <Index Progression> \( t \)

Set initial temperature (TOPAZ) or steady state temperature (DYNA/NIKE) to \( t \).

**TN** <Region> \( n \) -or- **TNI** <Index Progression> \( n \)

Set thickness number to \( n \).

**VE** <Region>

Set edge visibility on for outline and phantom edge plotting. <Region> must be a line in the reduced index space.
13. Interactive Commands

After the model is generated, LS-INGRID enters the interactive graphics phase of the program. The $x$-axis in screen coordinates is fixed relative to the screen and extends horizontally to the viewer’s right. The $y$-axis is positive up. The $z$-axis extends out of the screen towards the viewer.

The following commands are allowed in this phase.

- **AJNP** $p_x p_y p_z$  
  Print the nodal point which is nearest to point $(p_x, p_y, p_z)$.

- **AM** $m_1 m_2 ...$  
  Add materials $m_1, m_2, ...$ (by number) to the active list.

- **AMN** $m_1 m_2 ...$  
  Add materials $m_1, m_2, ...$ (by name) to the active list.

- **AP** $p_1 p_2 ...$;  
  Add parts $p_1, p_2, ...$ to the active list.

- **ARROW**  
  Toggle arrow plotting on or off. This allows the direction of the tool path to be visualized.

- **ASCII**  
  Read ASCII tracer particle file.

- **BPTOL** $p_1 p_2 t$  
  The tolerance to be used when merging part $p_1$ to $p_2$ is $t$.

- **CCEN**  
  Select the center of the picture using the mouse.

- **CCOL** $i r g b$  
  Change the (red, green, blue) values of color number $i$ to $(r, g, b)$.

- **CENT**  
  Moments and products of inertia are determined relative to the centroid and global axes (default).

- **CHUE** $m h$  
  Change hue of material $m$ to $h$.

- **CONT**  
  Finish model generation and format the output file.

- **CSAT** $m s$  
  Change saturation of material $m$ to $s$.

- **Dx**  
  Move down distance $x$ relative to the structure.

- **DI ACCE**  
  Display accelerometers.

- **DI BELT**  
  Display seat belts.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DI CV</strong></td>
<td>Display convection boundary condition. (surface segments).</td>
</tr>
<tr>
<td><strong>DI CVL</strong></td>
<td>Display convection boundary condition. (edge segments).</td>
</tr>
<tr>
<td><strong>DI D lc</strong></td>
<td>Display forced displacements associated with load case (or load curve) lc.</td>
</tr>
<tr>
<td><strong>DI DETP</strong></td>
<td>Display detonation points.</td>
</tr>
<tr>
<td><strong>DI DS n1 n2... ;</strong></td>
<td>Display digitized surface definitions n1, n2...</td>
</tr>
<tr>
<td><strong>DI DSAD n1 n2 ... ;</strong></td>
<td>Display digitized surfaces and add definitions n1, n2 ... to the display list.</td>
</tr>
<tr>
<td><strong>DI DSRM n1 n2 ... ;</strong></td>
<td>Display digitized surfaces and remove definitions n1, n2 ... from the display list.</td>
</tr>
<tr>
<td><strong>DI DX</strong></td>
<td>Display X-translational boundary conditions.</td>
</tr>
<tr>
<td><strong>DI DY</strong></td>
<td>Display Y-translational boundary conditions.</td>
</tr>
<tr>
<td><strong>DI DZ</strong></td>
<td>Display Z-translational boundary conditions.</td>
</tr>
<tr>
<td><strong>DI EDR n</strong></td>
<td>Display elements to be deleted on restart n.</td>
</tr>
<tr>
<td><strong>DI EPB</strong></td>
<td>Display element print blocks.</td>
</tr>
<tr>
<td><strong>DI F lc</strong></td>
<td>Display point loads associated with load case (or load curve) lc.</td>
</tr>
<tr>
<td><strong>DI FL</strong></td>
<td>Display flux boundary conditions; (edge segments).</td>
</tr>
<tr>
<td><strong>DI FLUX</strong></td>
<td>Display flux boundary condition; (surface segments).</td>
</tr>
<tr>
<td><strong>DI FSYM</strong></td>
<td>Display failing symmetry planes.</td>
</tr>
<tr>
<td><strong>DI INTF name</strong></td>
<td>Display component substructure name.</td>
</tr>
<tr>
<td><strong>DI JOY</strong></td>
<td>Display joy interface nodes.</td>
</tr>
<tr>
<td><strong>DI JTS</strong></td>
<td>Display joints.</td>
</tr>
<tr>
<td><strong>DI L3D</strong></td>
<td>Display three-dimensional line definitions.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| **DI LAX** code | Display local axes.  
  - code = "R": local R-axis  
  - code = "S": local S-axis  
  - code = "T": local T-axis  
  - code = "RS": local RS-axes  
  - code = "ST": local ST-axes  
  - code = "TR": local TR-axes  
  - code = "RST": local RST-axes  
  - code = "CORI": local composite angles |
| **DI M** \(m_1 \ m_2 \ldots\) | Materials \(m_1, m_2, \ldots\) are to be highlighted during plotting. |
| **DI MCG** \(m\) | Display mass properties of individual materials. |
| **DI MK** \(m\) | Display marked surface \(m\). |
| **DI NCV** | Display NURB curves. |
| **DI NFG** | Display nodal force groups. |
| **DI NRB** | Display non-reflecting boundaries. |
| **DI NPB** | Display nodal print blocks. |
| **DI NSF** | Display NURB surfaces. |
| **DI NV** | Display shell element normal vectors. |
| **DI ORV** | Display orientation vectors. |
| **DI OUTL** | Display free edges of shells. |
| **DI P** \(p_1 \ p_2 \ldots\) | Parts \(p_1, p_2, \ldots\) are to be highlighted during plotting. |
| **DI PL** \(lc\) | Display pressure surfaces associated with load case (or load curve) \(lc\); (edge segments). |
| **DI PM** | Display point masses. |
| **DI PR** \(lc\) | Display pressure surfaces associated with load case (or load curve) \(lc\); (surface segments). |
| **DI PV** \(n\) | Display tool path \(n\). |
| **DI RB** | Display radiation boundary conditions. (surface segments). |
| **DI RBL** | Display radiation boundary conditions; (edge segments). |
| **DI RBN** | Display nodal rigid bodies. (LS-910 and later). |
Interactive Commands

**DI RE**  
Display radiation enclosure; (surface segments).

**DI REL**  
Display radiation enclosure. (edge segments)

**DI RX**  
Display X-rotational boundary conditions.

**DI RXN m**  
Display extra nodes slaved to rigid body material m.

**DI RY**  
Display Y-rotational boundary conditions.

**DI RZ**  
Display Z-rotational boundary conditions.

**DI SBI**  
Display shell/brick interfaces.

**DI SFC islid mslid**  
Display nodes which are part of sliding interface definition islid.  
mslid = "M": display master side.  
mslid = "S": display slave side.  
mslid = "B": display both sides.

**DI SI islid mslid**  
Display slide surface islid; (surface segments).  
mslid = "M": display master side  
mslid = "S": display slave side  
mslid = "B": display both sides

**DI SL n isid**  
Display slide line n; (edge segments).  
mslid = "M": display master side  
mslid = "S": display slave side  
mslid = "B": display both sides.

**DI SPC**  
Display single point constraints.

**DI SPD**  
Display springs and dampers.

**DI SW s**  
Display stonewall s.

**DI SY isym**  
Display symmetry plane isym.

**DI SYSJ isym**  
Display symmetry plane isym.

**DI TB**  
Display temperature boundary conditions.

**DI TI**  
Display temperature initial conditions.

**DI TRACER**  
Display tracer particles.

**DI VB lc**  
Display velocity boundary conditions associated with load curve lc.

**DI VECT c**  
Display vectors of component c.  
\( c = IV \): initial velocity  
\( c = IR \): initial rotational velocity  
\( c = IDV \): initial distortional velocity
c = IRB: initial rigid body velocity
\( c = V \): current velocity
\( c = DV \): current distortional velocity
\( c = RB \): current rigid body velocity
\( c = A \): current acceleration

**DI WARP** \( ang \)  
Display shells with warp angles that exceed \( ang \).

**DIAD**  
This can be used in any of the above commands instead of DI. If this is used, then the display request is in addition to the previous ones rather than replacing them.

**DICOL 1**  
Following DI and DIAD options are to be performed using color number 1. Valid numbers for l are 1 through 7.

**DIOFF**  
Turn off display options.

**DM** \( m_1 \) \( m_2 \) ..., Delete materials \( m_1 \), \( m_2 \), ... (by number) from active display list.

**DMN** \( m_1 \) \( m_2 \) ..., Delete materials \( m_1 \), \( m_2 \), ... (by name) from active display list.

**DMEM**  
Dump memory allocations statistics.

**DRAW**  
Draw the mesh. All mesh lines are plotted.

**DSV**  
View three-dimensional digitized surfaces.

**DSVS** \( d_1 \), \( d_2 \), ...\( d_n \);  
View digitized surfaces \( d_1 \), \( d_2 \) ...

**DSAD** \( d_1 \), \( d_2 \), ...\( d_n \);  
Add digitized surfaces \( d_1 \), \( d_2 \) ... to the active display list.

**DSRM** \( d_1 \), \( d_2 \), ...\( d_n \);  
Remove digitized surfaces \( d_1 \), \( d_2 \) ... from the active display list.

**ELPLT** on/off  
Turn element number plotting on or off. The default is off.

**EXIT**  
Exit LS-INGRID now.

**FOLD** \( n \) \( d \)  
Airbag folding. All nodes of the model are included in the fold operation. Fold definitions 1 through \( n \) are applied in ascending order. \( d \) is an optional thickness which can be used to increase the fold thicknesses. The maximum of \( d \) and the fold definition specified thickness is used.

**FRAME**  
Reference frame with tick marks plotted
GRID
Displays will be overlaid by a grid of orthogonal lines. (Two dimensional plots only).

INFO
Print information on the mesh size.

L x
Move left a distance x relative to the structure.

L3V
View three dimensional digitized surfaces.

L3VS l1, l2, ...ln ;
View all three-dimensional lines l1, l2, ...

LCV n
View load curve n.

LIGHT px py pz
Locate the light source for continuous color plots at (px,py,pz).

LMIN l
Set minimum luminosity for continuous color plots to l.

LSIZE
On/off switch for printing extent of active three-dimensional line definitions during plotting.

LV
Display all two-dimensional line definitions.

LVI m n
Display lines m to n.

LVS l1, l2, ... ln ;
Display lines l1, l2, ...

M m1 m2 ..., Display materials m1, m2, ... (by number).

MCOL
Color plots based on element materials. (See also PCOL).

MMASS
This is the same as TMASS except that the calculation is only performed for the active materials.

MN m1 m2 ..., Display materials m1, m2, ... (by name).

MPLT
Plot mass properties on screen.

MSIZ
Print the extent of the current active material subset.

NCV ;
View all NURB curves.

NCV d1, d2, ... dn ;
View NURB curves d1, d2 ...

NCAD d1, d2, ... dn ;
Add NURB curves d1, d2 ... to the active display list.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NCRM</strong></td>
<td>$d_1, d_2, ... d_n$; Remove NURB curves $d_1, d_2$ ... from the active display list.</td>
</tr>
<tr>
<td><strong>NDPLT</strong></td>
<td>on/off; Turn node number plotting on or off. The default is off.</td>
</tr>
<tr>
<td><strong>NOFRAME</strong></td>
<td>No reference frame is plotted.</td>
</tr>
<tr>
<td><strong>NOGRID</strong></td>
<td>Displays will not be overlaid by a grid of orthogonal lines. (Default).</td>
</tr>
<tr>
<td><strong>NSET</strong></td>
<td>$n x y z$; Set the coordinates of node $n$ to $(x, y, z)$.</td>
</tr>
<tr>
<td><strong>NSV</strong></td>
<td>$d_1, d_2, ... d_n$; View all NURB surfaces.</td>
</tr>
<tr>
<td><strong>NSV</strong></td>
<td>$d_1, d_2, ... d_n$; View NURB surfaces $d_1, d_2$ ...</td>
</tr>
<tr>
<td><strong>NSAD</strong></td>
<td>$d_1, d_2, ... d_n$; Add NURB surfaces $d_1, d_2$ ... to the active display list.</td>
</tr>
<tr>
<td><strong>NSRM</strong></td>
<td>$d_1, d_2, ... d_n$; Remove NURB surfaces $d_1, d_2$ ... from the active display list.</td>
</tr>
<tr>
<td><strong>OVERLAY</strong></td>
<td>$n x y z$; Stop screen erasing of previous picture so that the next picture is overlaid.</td>
</tr>
<tr>
<td><strong>P</strong></td>
<td>$p_1 p_2 ...$; Display parts $p_1, p_2, ...$ To display all parts simply type &quot;$P ;&quot;. The &quot;;&quot; is also optional so that the command &quot;$P 1 VIEW&quot; would show part one on the screen.</td>
</tr>
<tr>
<td><strong>PCHK</strong></td>
<td>Turn on checking of penetrations in the single surface contact algorithms. Currently, this is only designed to work with the airbag folding capability. Repeating this command will turn the option off. Penetrations are graphically displayed.</td>
</tr>
<tr>
<td><strong>PCOL</strong></td>
<td>Color plots based on element parts. This is the default. (See also <strong>MCOL</strong>).</td>
</tr>
<tr>
<td><strong>PFOLD</strong></td>
<td>$n d p_1 p_2$; This is the same as the FOLD command except that only parts $p_1$ through $p_2$ are treated rather than the entire mesh.</td>
</tr>
<tr>
<td><strong>PINF</strong></td>
<td>Print information on each part.</td>
</tr>
<tr>
<td><strong>PMASS</strong></td>
<td>This is the same as TMASS except that the calculation is only performed for the active parts.</td>
</tr>
<tr>
<td><strong>POOR</strong></td>
<td>Poor man's hidden line algorithm.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| PRINT \( v \) | Echo the value of \( v \) back to the terminal. This is most frequently used with the calculator program. (e.g. "PRINT \([\sqrt{27}] \times 24.3\)"
| PSRGB | Create a RGB Postscript file.
| PTOL \( n \) \( t \) | Set the tolerance for part \( n \) to \( t \). See also "\( T \)" and "\( TP \)".
| PV | View tool paths.
| PVS \( p_1 \) \( p_2 \ldots \) \( p_n \); | View tool paths \( p_1, p_2, \ldots \)
| QUIT | Quit LS-INGRID now.
| R \( x \) | Move right a distance \( x \) relative to the structure.
| REDUCE | Eliminate exterior faces which have become interior faces due to the tolerance command.
| REFP \( r_x \) \( r_y \) \( r_z \) | Moments and products of inertia are determined relative to the point \( (r_x, r_y, r_z) \) and global axes.
| RESO \( ires \) | Set the Z-buffer resolution to \( ires \) for the VIEW command. \( ires \) is limited to one of 256, 512, 1024, 2048, 4096, 8192. (The default is 1024).
| REST | Restore all rotations to their initial settings.

**Note:** The local coordinates are fixed to the model and rotate as the model rotates.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| RLX \( \theta \) | Rotate the body \( \theta \) degrees about the local \( x \)-axis.
| RLY \( \theta \) | Rotate the body \( \theta \) degrees about the local \( y \)-axis.
| RLZ \( \theta \) | Rotate the body \( \theta \) degrees about the local \( z \)-axis.
| RM \( m_1 \) \( m_2 \ldots \) , | Remove materials \( m_1, m_2, \ldots \) (by number) from the active list.
| RMN \( m_1 \) \( m_2 \ldots \) , | Remove materials \( m_1, m_2, \ldots \) (by name) from the active list.
| RP \( p_1 \) \( p_2 \ldots \); | Remove parts \( p_1, p_2, \ldots \) from the active list.
| RX \( \theta \) | Rotate body \( \theta \) degrees about the \( x \)-axis in the screen coordinates. A positive rotation is counterclockwise.
| RY \( \theta \) | Rotate body \( \theta \) degrees about the \( y \)-axis in the screen coordinates. A positive rotation is counterclockwise.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RZ ( \theta )</td>
<td>Rotate body ( \theta ) degrees about the ( z )-axis in the screen coordinates. A positive rotation is counterclockwise.</td>
</tr>
<tr>
<td>SCALE ( s )</td>
<td>Multiply the mesh size by ( s ). Default is 1.0.</td>
</tr>
<tr>
<td>SCOL</td>
<td>Color plots based on system name; (see also MCOL and PCOL).</td>
</tr>
<tr>
<td>SEAL name</td>
<td>Seal airbag edges which are marked with name.</td>
</tr>
<tr>
<td>SEAL CIRCLE</td>
<td>Seal the airbag periphery. The airbag mesh is assumed to be circular in the ( x-y ) plane and centered along the ( z )-axis at ( z=0 ); (default).</td>
</tr>
<tr>
<td>SEAL OFF</td>
<td>Turn off airbag sealing options.</td>
</tr>
<tr>
<td>SEAL OUTLINE</td>
<td>Seal the free edges of an airbag.</td>
</tr>
<tr>
<td>SHRINK ( s )</td>
<td>Shrink individual elements by ( s ) when plotting. This is used to see if there are any holes in the mesh.</td>
</tr>
<tr>
<td>SIZE</td>
<td>Print the range of coordinates in the current active part list.</td>
</tr>
<tr>
<td>STOP</td>
<td>Exit the program immediately.</td>
</tr>
<tr>
<td>T tol</td>
<td>Remove duplicate nodes within a distance tol. This command will not eliminate coincident nodes on opposite sides of slide surfaces or joints.</td>
</tr>
<tr>
<td>TMASS</td>
<td>Calculate the total mass of the model. Mass densities must be input using the MAT command. This command also calculates kinetic energy, linear momentum, volume, moments of inertia, and the centroid.</td>
</tr>
<tr>
<td>TP tol</td>
<td>Remove duplicate nodes within a distance tol and print the number of nodes merged between any two parts.</td>
</tr>
<tr>
<td>TRIAD on/off</td>
<td>Show the coordinate system triad on the screen when doing three-dimensional plots. The default is on.</td>
</tr>
<tr>
<td>TRPT</td>
<td>On/off flag for printing timing statistics from plot commands.</td>
</tr>
<tr>
<td>TTIME</td>
<td>Plot time histories of tracer particles. (See also ASCII TRACER.) ( \text{comp} ) is one of the following components: ( \sigma_{xx} )</td>
</tr>
</tbody>
</table>
= SY --> $\sigma_{yy}$
= SZ --> $\sigma_{zz}$
= SXY --> $\sigma_{xy}$
= SYZ --> $\sigma_{yz}$
= SZX --> $\sigma_{zx}$
= P --> Pressure
= EFP --> Effective Plastic Strain
= RHO --> $\rho$
= RVOL --> Relative Volume

**TV \( n \)**
Select graphics device \( n \). Available graphics devices are dependent on the installation. When typing this option, LS-INGRID will prompt the user for the correct device and provide a list of available devices.

**U \( x \)**
Move up a distance \( x \) relative to the structure.

**UPDATE**
Re-read the LS-INGRID input deck and return to the interactive phase for continued plotting.

**VEOS \( n \ V_1 \ V_2 \)**
View equation of state for material \( n \) from relative volume \( V_1 \) to relative volume \( V_2 \).

**VIEW or G**
View the mesh. An algorithm based on a z-buffer method is used for hidden line processing.

**WBGR**
Write the boundary grid into the LS-INGRID output file. For a solid grid, all internal polygons are removed and the external polygons are written out.

**WBIF**
Write the boundary information file, grfinfo for use by POST.

**WRDB**
Write reduced TAURUS database. This option only writes out surface polygons.

**WTDB**
Write TAURUS database.

**ZIN**
Zoom in on the picture by selecting the upper and lower corners with the mouse.

**ZOUT**
Zoom out on the picture by selecting the upper and lower corners with the mouse.
13.1 Exploded View Commands

Exploded view commands permit collections of parts or materials to be moved from their generated locations. Exploding a model will affect the graphics and mass property calculations, but will not affect the output computational model. Exploding a model with respect to parts will only affect the TMASS and PMASS commands, while exploding with respect to materials will only affect the MMASS command.

**MEXP**
Exploded views are performed with respect to materials. This command is automatically invoked by all other material explode commands.

**MLOC** data
Set position of material subset to the position specified in data. Data is described in the section on *Coordinate Transformations*.

**MMOV** data
Shift the position of material subset by the transformation specified in data. Data is described in the section on *Coordinate Transformations*.

**MSEL** $m_1 \, m_2 \, ...$
Select material subset $m_1, m_2, ...$; for explode operations.

**NEXP**
Turn off exploded view option.

**PEXP**
Exploded views are performed with respect to parts. This command is automatically invoked by all other part explode commands.

**PLOC** data
Set position of part subset to the position specified in data. Data is described in the section on *Coordinate Transformations*.

**PMOV** data
Shift the position of part subset by the transformation specified in data. Data is described in the section on *Coordinate Transformations*.

**PSEL** $p_1, p_2, ...$
Select part subset $p_1, p_2, ...$; for explode operations.
13.2 TAUROUS/Post-Processing Commands

The post-processing phase of LS-INGRID allows for the generated models to be displayed in their deformed configurations with a variety of boundary conditions or other graphical information superimposed. Some time history facilities are also included.

**TAURUS file**
Read TAUROUS database file.

**ALAB label**
Set abscissa label.

**ALAB OFF**
Use default abscissa label.

**ASCL scale**
Set abscissa scale factor.

**ASET min max**
Set abscissa range.

**DSF \(d\)**
Set displacement scale factor to \(d\); (default=1.0).

**DTS \(s_1 \ s_2\)**
Delete states \(s_1\) through \(s_2\).

**GTIM \(E\ \text{comp}\)**
Plot global component \(E\ \text{comp}\). Components are defined in table 13.2.1.

**LS**
Last state.

**LTS**
List time states.

**NTIME \(E\ \text{comp}\ \text{n_1 \ n_2}\)**
Plot nodal component \(E\ \text{comp}\) for nodes \(n_1, n_2, \ldots\). Components are defined in table 13.2.2.

**NTIME \(E\ \text{comp}\)**
Plot nodal component \(E\ \text{comp}\) for the previously used nodal list. Components are defined in table 13.2.2.

**OLAB label**
Set ordinate label.

**OLAB OFF**
Use default ordinate label.

**OSCL scale**
Set ordinate scale factor.

**OSET min max**
Set ordinate range.

**REP \(s_1 \ s_2 \ k\)**
Execute the next plot command from states \(s_1\) to \(s_2\) by increment state increment \(k\).

**S \(n\)**
Select state number \(n\).

**S 0**
Plot the generated geometry.

**SI \(n\)**
Increment the state number by \(n\).
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDEF $n$</td>
<td>The undeformed state is number $n$; (default=1).</td>
</tr>
</tbody>
</table>
**TABLE 13.2.1**  
**GLOBAL TIME HISTORY COMPONENTS**

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENERGY</td>
<td>Plot the total, kinetic and internal energy.</td>
</tr>
<tr>
<td>MOMENTUM</td>
<td>Plot the momentum vector.</td>
</tr>
<tr>
<td>XVEL</td>
<td>$X$-momentum/total mass.</td>
</tr>
<tr>
<td>YVEL</td>
<td>$Y$-momentum/total mass.</td>
</tr>
<tr>
<td>ZVEL</td>
<td>$Z$-momentum/total mass.</td>
</tr>
<tr>
<td>INTERNAL</td>
<td>Internal energy.</td>
</tr>
<tr>
<td>KINETIC</td>
<td>Kinetic energy.</td>
</tr>
<tr>
<td>TOTAL</td>
<td>Total energy.</td>
</tr>
</tbody>
</table>

**TABLE 13.2.2**  
**VECTOR PLOT COMPONENTS**

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Current acceleration.</td>
</tr>
<tr>
<td>DV</td>
<td>Current distortional velocity.</td>
</tr>
<tr>
<td>IDV</td>
<td>Initial distortional velocity.</td>
</tr>
<tr>
<td>IR</td>
<td>Initial rotational velocity.</td>
</tr>
<tr>
<td>IRB</td>
<td>Initial rigid body velocity.</td>
</tr>
<tr>
<td>IV</td>
<td>Initial velocity.</td>
</tr>
<tr>
<td>RB</td>
<td>Current rigid body velocity.</td>
</tr>
<tr>
<td>V</td>
<td>Current velocity.</td>
</tr>
</tbody>
</table>
### Table 13.2.3

**Nodal Time History Components**

| Rectangular coordinates | | Cylindrical coordinates | | Spherical coordinates | | Special components |
|-------------------------|-----------------|------------------------|----------------------|-----------------|-----------------|
| X                       | X-position      | CR                     | Radial-position      | SR               | Radial-position |
| Y                       | Y-position      | CT                     | Circumferential-position | ST              | Theta-position |
| Z                       | Z-position      | CZ                     | Axial-position       | SP               | Phi-position    |
| AX                      | X-acceleration  | CAR                    | Radial-acceleration  | SAR              | Radial-acceleration |
| AY                      | Y-acceleration  | CAT                    | Circumferential-acceleration | SAT             | Theta-acceleration |
| AZ                      | Z-acceleration  | CAZ                    | Axial-acceleration   | SAP              | Phi-acceleration |
| DX                      | X-displacement  | CDR                    | Radial-displacement  | SDR              | Radial-displacement |
| DY                      | Y-displacement  | CDT                    | Circumferential-displacement | STT             | Theta-displacement |
| DZ                      | Z-displacement  | CDZ                    | Axial-displacement   | SPD              | Phi-displacement |
| VX                       | X-velocity      | CVR                    | Radial-velocity      | SVR              | Radial-velocity |
| VY                       | Y-velocity      | CVT                    | Circumferential-velocity | SVT             | Theta-velocity  |
| VZ                       | Z-velocity      | CVZ                    | Axial-velocity       | SVP              | Phi-velocity    |

| TEMP                     | Temperature     |
| TIME                     | Time            |
| TOTP                     | Total position  |
| TOTA                     | Total acceleration |
| TOTD                     | Total displacement |
| TOTV                     | Total velocity  |
14. Two-Dimensional Line Definitions

Two-dimensional line definitions are lists of \( r,z \) \((x,y)\) points which form a piecewise linear curve. Each line definitions has a number.

**LAD** \( r_c z_c t \) Define a circular arc centered at point \((r_c,z_c)\), beginning at the last point defined and sweeping through \( t \) degrees. Positive \( t \) is assumed to be counterclockwise.

**LADD** \( l_1 \, s \, l_2 \, t \) Define line definition \( l \) as a linear combination of line definitions \( l_1 \) and \( l_2 \). \( l = s \cdot l_1 + t \cdot l_2 \).

**LADV** \( ldr \, dz \) Add vector \((dr,dz)\) to line definition \( l \).

**LAP** \( r_1 \, z_1 \, r_c \, z_c \) Define a circular arc by specifying points. The arc is assumed to begin at the last point defined and to end at \((r_1,z_1)\). Point \((r_c,z_c)\) lies at the center of the arc.

**LAR** \( r_1 \, z_1 \, R \) Define a circular arc by specifying radius. An arc of radius \( R \) is assumed to begin at the last point defined and to end at \((r_1,z_1)\). If \( R \) is positive, the center of the arc lies to the left as one moves from the last point defined to \((r_1,z_1)\). If \( R \) is negative, the center of the arc will be to the right.

**LAT** \( r_1 \, z_1 \, r_2 \, z_2 \, R \) Define a circular arc of radius \( R \) tangent to the last line segment and a line segment joining point \((r_1,z_1)\) to point \((r_2,z_2)\). These line segments will be automatically extended or truncated at the tangency point.

**LCC** \( n \, r_c \, z_c \, t_1 \, t_2 \, r_1 \ldots r_n \) Define \( n \) lines consisting of circular arcs centered at point \((r_c,z_c)\) that sweep from angle \( t_1 \) to \( t_2 \). \( r_1, r_2, \ldots, r_n \) are the radii of the next \( n \) lines. Line numbers are assigned by LS-INGRID beginning with the next available number.

**LD** \( n \) Begin line definition \( n \).
Define an elliptical arc by the semi-major and semi-minor axes, \(a\) and \(b\) respectively, the center point \((r_c, z_c)\), and a sweep from angle \(t_1\) to angle \(t_2\) as measured from the major axis. Angle \(p\) is the angle between the major axis and the \(r\)-axis. A circular arc is generated by setting \(a=b\). Positive angles represent counterclockwise rotations.

Define a line definition using expression \(r(s)\) and \(z(s)\) where \(0 \leq s \leq 1\). The number of points to be used is \(n\).

The current line definition is formed by interpolation between line definition \(l_1\) and line definition \(l_2\). The equation is \(l = s*l_1 + (1-s)*l_2\).

Define a line segment for line \(n\) by offsetting a segment of line \(m\) such that the first point of the new segment begins at \((r_1, z_1)\) and the last point terminates at \((r_2, z_2)\).

Define a line segment for line \(n\) by offsetting the entire line \(m\) a distance \(d\). Positive \(d\) offsets the line segments to the left as one moves along line \(m\) in the direction that was originally defined. Negative \(d\) offsets the segment to the right.

The line definition consist of \(n\) points.

Define point for line \(n\) at the intersection point of lines \(l_1\) and \(l_2\).

Define a circular arc of radius \(R\) beginning at the last point defined and tangent to a line segment joining point \((r_1, z_1)\) to point \((r_2, z_2)\). This line segment will be extended or truncated to begin at the tangency point.

Define a line segment beginning at the last point defined and terminating at the tangency point on an arc of radius \(R\) centered at \((r_c, z_c)\). The first tangency point encountered as the arc is generated by a counterclockwise rotation from the \(r\)-axis will become the end point. If \(R\) is given as a negative number, a clockwise rotation from the \(r\)-axis will determine the first tangency point.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LRL</strong> ( n \ r_c \ z_c \ L \ t_1 \ ... \ t_n )</td>
<td>Define ( n ) lines consisting of radial lines of length ( L ) originating at point ((r_c, z_c)) and oriented at angles ( t_1 \ ... \ t_n ). Positive angles are measured counterclockwise from ( r )-axis.</td>
</tr>
<tr>
<td><strong>LROT</strong> ( l \ t )</td>
<td>Rotate line definition ( l ) about the origin ( t ) degrees.</td>
</tr>
<tr>
<td><strong>LSCA</strong> ( l \ s )</td>
<td>Scale line definition ( l ) by ( s ).</td>
</tr>
<tr>
<td><strong>LSCR</strong> ( l \ s )</td>
<td>Scale ( r )-coordinates of line definition ( l ) by ( s ).</td>
</tr>
<tr>
<td><strong>LSCZ</strong> ( l \ s )</td>
<td>Scale ( z )-coordinates of line definition ( l ) by ( s ).</td>
</tr>
<tr>
<td><strong>LSTL</strong> ( m \ d_r \ d_z )</td>
<td>Define a line segment for line ( n ) by translating line ( m ) an increment ((d_r, d_z)).</td>
</tr>
<tr>
<td><strong>LT</strong> ( n \ d_r \ d_z )</td>
<td>Translate line ( n ) by the increment ((d_r, d_z)).</td>
</tr>
<tr>
<td><strong>LTAS</strong> ( r_1 \ z_1 \ rot \ r_2 \ z_2 \ R )</td>
<td>Define a line segment tangent to a circular arc centered at point ((r_1, z_1)), beginning at the last point defined, and sweeping counterclockwise if ( rot=1 ) and clockwise if ( rot=-1 ). The line segment terminates at its tangency point on a second arc of radius ( R ) centered at ((r_2, z_2)). The first tangency point encountered as the second arc is generated by a counterclockwise rotation from the ( r )-axis will become the end point. If ( R ) is given as a negative number; a clockwise rotation from the ( r )-axis will determine the tangency point.</td>
</tr>
<tr>
<td><strong>LTBC</strong> ( m \ t \ dt \ s \ r_1 \ r_2 \ ... \ r_m )</td>
<td>Define a line segment for line ( n ) with tab cell data. Tab cell data is often used in drafting programs, and consist of ( m ) radii each ( dt ) degrees apart starting at angle ( t ). Each radius is scaled by ( s ). Positive angles represent counterclockwise rotations.</td>
</tr>
<tr>
<td><strong>LTBO</strong> ( m_1 \ d_1 \ m_2 \ d_2 \ ... \ m_m \ d_m )</td>
<td>Define a line segment for line ( n ) by offsetting last line defined with the &quot;LTBC&quot; or &quot;LTBO&quot; command. The radii of the first ( m_1 ) points are offset ( d_1 ), the next ( m_2 ) by ( d_2 ), and so on. Note that ( m=m_1 + m_2 + \ldots + m_m ) where ( m ) comes from the last &quot;LTBC&quot; command.</td>
</tr>
<tr>
<td><strong>LTP</strong> ( r \ z \ R )</td>
<td>Define a circular arc of radius ( R ) tangent to the last line segment and terminating at point ((r, z)). The last line segment will be automatically extended or truncated to the tangency point.</td>
</tr>
<tr>
<td><strong>LVC</strong> ( t \ l )</td>
<td>Define a line segment vector of length ( l ).</td>
</tr>
</tbody>
</table>
or

LVC \( r_1 \ z_1 \ t \ l \)

LVC \( r_2 \ z_2 \ t \ -l \)

oriented at \( t \) degrees (positive counterclockwise from the \( r \)-axis). If this is the first command in a new line, the origin \((r,z)\) must be given (second or third forms). A negative \( l \) indicates that the second point is defined, i.e., that the vector points towards the first point.

**RLN**

Read next line definition in operational input file and assign the next available line number.

**RLNS**

Read all line definitions in operational input file and assign the next available line numbers.
15. Three-Dimensional Line Definitions

Three-dimensional line definitions are lists of \(x, y, z\) points which form a piecewise linear curve. Each line definition has a name which is a character string with up to eight characters.

**AVGN** \(l_1 l_2 \ldots l_n\) ;

Average \(n\) line definitions.

**BLEN** \(s_1 s_2 s_3 s_4 s_5 p\)

Determine a line interpolated between surface definition \(s_1\) and \(s_2\) by a ratio \(p\). \(s_3\) and \(s_4\) determine the end points of the line and the line lies on \(s_5\). Convergence can be improved by using the following: \(PO\) for \((s_1, s_3, s_5)\), \(P1\) for \((s_1, s_4, s_5)\), \(P2\) for \((s_2, s_4, s_5)\), \(P3\) for \((s_2, s_3, s_5)\). Refer to Figure 14-1.

**COMP** \(l_1 l_2 \ldots l_n\) ;

Form a single line definition by placing line definitions \(l_1 l_2 \ldots l_n\) end-to-end.

**COPY** \(n\) data

Move line definition \(n\) using the transformation defined by data. Input for data is described in the section on Coordinate Transformations.

**L2D** \(n\) data

Turn two-dimensional line definition \(n\) into a three dimensional line definition. The line definition is initially assumed to lie in the \(x-y\) plane and can be moved anywhere in space using data which is described in Coordinate Transformations.

**L3D** \(n\)

Begin line definition \(n\).

**LAD** \(x_c y_c a\)

Form an arc about a \(z\)-vector located at \((x_c, y_c)\) beginning at the last point defined and sweeping through \(a\) degrees.

**LADD** \(l_1 s_1 l_2 s_2\)

Make a linear combination: \(l_{\text{new}} = l_1 * s_1 + l_2 * s_2\).

**LBCX** \(l r f\)

Ball-correct line definition \(l\) with a cylinder parallel to the \(x\)-axis with radius \(r\). If the correction is to the left then \(f\) is 'left', otherwise \(f\) is 'right'.

**LBCY** \(l r f\)

Ball-correct line definition \(l\) with a cylinder parallel to the \(y\)-axis with radius \(r\). If the correction is to the left then \(f\) is 'left', otherwise \(f\) is 'right'.

**LBCZ** \(l r f\)

Ball-correct line definition \(l\) with a cylinder
THREE-DIMENSIONAL LINE DEFINITIONS

parallel to the z-axis with radius r. If the correction is to the left then f is 'left', otherwise f is 'right'.

**LBCV l r v_x v_y v_z**

Ball-correct line definition l with a cylinder parallel to the vector, (v_x, v_y, v_z), with radius r. If the correction is to the left then f is 'left', otherwise f is 'right'.

**LCUT l opt dist**

Cut line definition l with a plane normal to axis opt (opt = X, Y or Z) at a distance of dist from the origin. The results are stored into calculator variables (l3cenx, l3ceny, l3cenz). They may then be accessed and used as necessary.

**LEXP x(s) y(s) z(s) n**

Define a line definition using expression x(s), y(s) and z(s) where 0 ≤ s ≤ 1. The number of points to be used is n.

**LFOR l opt vd sd dir**

Form line definition l. If a point on l is inside (opt=IN) or outside (opt=OUT) of surface definition vd, then it is projected onto surface definition sd. The projection is constrained by dir:

- dir=0: project to nearest point.
- dir=1: project along X-axis
- dir=2: project along Y-axis
- dir=3: project along Z-axis

**LINT l_1 l_2 s**

Form a line by linear interpolation between l_1 and l_2 with parameter s.

**LLCM l_1 s_1 ... l_n s_n ;**

Form a linear combination of n lines where

\[ l_{\text{new}}(r) = l_1(r) * s_1 + ... + l_n(r) * s_n. \]

**LP n x_1 y_1 z_1 ... x_n y_n z_n**

The line definition consists of n points.

**LPN n p_x p_y p_z**

The next point on the line definition is at (p_x, p_y, p_z) but n equal spaced points in a straight line are added to make up this segment.

**LPRJ l s**

Project line definition l onto surface definition s.

**LREV**

Reverse the direction of the line. Additions to the line definition will occur at the beginning of the line rather than the end of the line.

**LRNX l r**

Sharp corners on line definitions l are rounded by a cylinder parallel to the x-axis with radius r. The coordinates of the center of rotation of the last round are returned to calculator variables (l3cenx, l3ceny, l3cenz) and the last angle of sweep.
is returned to l3angle.

**LRNY l r**

Sharp corners on line definitions $l$ are rounded by a cylinder parallel to the $y$-axis with radius $r$. The coordinates of the center of rotation of the last round are returned to calculator variables (l3cenx,l3ceny,l3cenz) and the last angle of sweep is returned to l3angle.

**LRNZ l r**

Sharp corners on line definitions $l$ are rounded by a cylinder parallel to the $z$-axis with radius $r$. The coordinates of the center of rotation of the last round are returned to calculator variables (l3cenx,l3ceny,l3cenz) and the last angle of sweep is returned to l3angle.

**LRNV l r v_x v_y v_z**

Sharp corners on line definition $l$ are rounded by a cylinder parallel to the vector, $(v_x, v_y, v_z)$, with radius $r$. The coordinates of the center of rotation of the last round are returned to calculator variables (l3cenx,l3ceny,l3cenz) and the last angle of sweep is returned to l3angle.

**LROT p_x p_y p_z v_x v_y v_z w**

Form an arc by taking the last point and rotating it an angle $w$ (in degrees) about the axis defined by point $(p_x,p_y,p_z)$ and orientation vector $(v_x,v_y,v_z)$.

**LVT d**

Add a vector tangent to the last line segment with length $d$.

**LVTB d**

Add a vector tangent to the first line segment with length $d$.

**P0 x_0 y_0 z_0**

Set point P0 for intersection.

**P1 x_1 y_1 z_1**

Set point P1 for intersection determination on the next command.

**P2 x_2 y_2 z_2**

Set point P2 for intersection determination on the next command.

**P3 x_3 y_3 z_3**

Set point P3 for intersection determination on the next command.

**PINT s_1 s_2 s_3**

The next point on the line is at the intersection of $s_1$, $s_2$ and $s_3$. P0 can be used to improve convergence. The results of the projection are returned to the calculator variables (l3cenx,l3ceny,l3cenz).

**PPRJ p_x p_y p_z surf**

The next point on the line definition is formed
by projecting \((p_x, p_y, p_z)\) to the nearest point on surface surf.

**SINT \(s_1 s_2 s_3 s_4\)**

Determine the curve formed by the intersection of \(s_1\) and \(s_2\) beginning at \(s_3\) and terminating at \(s_4\). If this is not the first point on the line, then \(s_3\) is not input and LS-INGRID assumes that the last point defined lies on the intersection of \(s_1\) and \(s_2\). The convergence can be improved by using P0 for \((s_1, s_2, s_3)\) and P1 for \((s_2, s_3, s_4)\).
16. Surface Definitions

This section describes options for defining three-dimensional curved surfaces. Analytical representations of the surfaces are stored, if possible, so that exact projections can be made.

**BLND** $d_1 l_1 d_2 l_2$

The surface is blended between line definition $l_1$ and line definition $l_2$. Initially the line definitions are in the $x$-$z$ plane at $y=0$. Line definitions are moved by $d_1$ and $d_2$ which are described in Coordinate Transformations.

**CN2P** $p_x p_y p_z v_x v_y v_z r_1 z_1 r_2 z_2$

Define a conical surface by specifying the axis and two points. $(p_x, p_y, p_z)$ is a point on the axis and $(v_x, v_y, v_z)$ is a vector which orients the axis. Radial and axial positions relative to the center point are $(r_1, z_1)$ and $(r_2, z_2)$.

**CONE** $p_x p_y p_z v_x v_y v_z r \theta$

Define a conical surface by specifying an axis, a radius and an angle. $(p_x, p_y, p_z)$ is a point on the axis where the cone has radius $r$ and $(v_x, v_y, v_z)$ is a vector along the axis. The angle of the cone relative to the positive axis is $\theta$.

**CP** data $l$

Form an infinite surface from line definition $l$. Initially the $r$-coordinate of the line definition is the $x$-coordinate of the part and the $z$-coordinate of the line definition is the $y$-coordinate of the part. The surface is the same curve in any $x$-$y$ plane along the part's $z$-axis. Coordinate Transformations describes data which can be used to move the surface anywhere in space.

**CR** $p_x p_y p_z v_x v_y v_z l$

Spin two-dimensional line definition, $l$, about an axis to form a line. $(p_x, p_y, p_z)$ is the center point on the axis and $(v_x, v_y, v_z)$ is a vector which orients the axis.

**CRX, CRY, or CRZ** $l$

Spin two-dimensional line definition, $l$, about the $X$, $Y$, or $Z$ axes, respectively.

**CYLI** $p_x p_y p_z v_x v_y v_z r$

Define a cylindrical surface with a point on the axis at $(p_x, p_y, p_z)$, an orientation vector $(v_x, v_y, v_z)$, and radius $r$.

**ER** $p_x p_y p_z v_x v_y v_z r_1 r_2$

Define an elliptical surface revolved about an
SURFACE DEFINITIONS

axis. \((p_x p_y p_z)\) is the center point, \((v_x, v_y, v_z)\) is a vector which orients the ellipse. The radius in the plane normal to the axis of rotation is \(r_1\) and the intercept along the axis of rotation is at +/- \(r_2\).

**GELN a b c n data**  Define a general ellipsoid with the equation: 
\[(x/a)^n+(y/b)^n+(z/c)^n=1\]

The ellipsoid can be positioned anywhere in space with data which is described in the section on Coordinate Transformations.

**GELS a b c data**  Define an ellipsoid with the equation: 
\[(x/a)^2+(y/b)^2+(z/c)^2=1\]

The ellipsoid can be positioned anywhere in space with data which is described in the section on Coordinate Transformations.

**GS n**  Use general 3-D digitized surface number \(n\).

**GS1 n**  Use lower side of general 3-D surface.

**GS2 n**  Use upper side of general 3-D surface.

**GSM data n**  Digitized surface \(n\) is moved by data which is described in Coordinate Transformations.

**GSN offset n**  Use digitized surface \(n\) but offset the surface by offset in the normal direction.

**L3 l**  The surface is actually just three-dimensional line definition \(l\).

**L3P l v_x v_y v_z**  Three-dimensional line definition \(l\) is projected along \((v_x, v_y, v_z)\) to form a surface.

**L3R l r**  The surface is a circular tube of radius \(r\) about three-dimensional line definition \(l\).

**L3S p_x p_y p_z v_x v_y v_z l**  Spin three-dimensional line definition \(l\) about the axis defined by point \((p_x p_y p_z)\) and orientation vector \((v_x, v_y, v_z)\).

**NCV data n**  NURB curve defined by entity number \(n\) in the NURB geometry database is used. This curve is moved by data which is described in Coordinate Transformations.

**NSF data n**  NURB surface defined by entity number \(n\) in the NURB geometry database is used. This surface
LS-INGRID SURFACE DEFINITIONS

NSFN data \( n \)  
NURB surface defined by entity name \( n \) in the SC03 geometry database is used. This surface is moved by data which is described in Coordinate Transformations.

PL3 \( f_1 \) p1x p1y p1z  
\( f_2 \) p2x p2y p2z  
\( f_3 \) p3x p3y p3z offset  
Plane Defined by three points. \( f_1, f_2 \) and \( f_3 \) specify the coordinate system which is RT for rectangular, CY for cylindrical, or SP for spherical. \( P_1, P_2, \) and \( P_3 \) must be three non-collinear points in the plane. \( f_2 \) and \( f_3 \) can also be V to indicate that the input points are vectors relative to \( P_1 \). The surface definition is offset from the three points by the distance offset.

PLAN \( p_x \) p\( y \) p\( z \) v\( x \) v\( y \) v\( z \)  
Define a plane. \((p_x p_y p_z)\) is any point on the plane and \((v_x v_y v_z)\) is a normal vector.

POLY \( p_x \) p\( y \) p\( z \) v\( x \) v\( y \) v\( z \) \( n \) \( a_0 a_1 ... a_n \)  
Define a surface as a planar polynomial which is then spun about an axis. \((p_x p_y p_z)\) is a point on the axis of rotation and \((v_x v_y v_z)\) is a vector which orients the axis of rotation. The polynomial is of degree \( n \) with coefficients \( a_0, a_1, ... a_n \).

PR \( p_x \) p\( y \) p\( z \) v\( x \) v\( y \) v\( z \) r\( 1 \) t\( 1 \) r\( 2 \) t\( 2 \) r\( 3 \) t\( 3 \)  
Parabolic surface of revolution. \((p_x p_y p_z)\) is a point on the axis of revolution and \((v_x v_y v_z)\) is a vector orienting the axis of revolution. \((r_1, t_1)\), \((r_2, t_2)\) and \((r_3, t_3)\) are radial and axial positions of three points which are fit with a parabola.

SP \( p_x \) p\( y \) p\( z \) r  
Define a sphere with center \((p_x p_y p_z)\) and radius \( r \).

T1 \( l_1 \) l\( 2 \) l\( 3 \) l\( 4 \) l\( 5 \)  
This is a special purpose surface. \( l_1, l_2, l_3, l_4 \) and \( l_5 \) refer to two-dimensional line definitions. The surface is axisymmetric about the \( Z \)-axis and performs \( Z \)-projections only. The equation for the surface is as follows: 
\[ z = l_3(r) + (l_2(r) - l_1(r)) \times (1 + l_4(q) \times l_5(r)) + l_1(r) \]

T2 \( n \) opt  
Project along an axis onto digitized surface definition \( n \). Values for opt are as follows:  
=MINX: project along the \( X \)-axis to the minimum \( X \)-intercept.  
=MAXX: project along the \( X \)-axis to the maximum \( X \)-intercept.  
=MINY: project along the \( Y \)-axis to the minimum \( Y \)-intercept.
=MAXY: project along the Y-axis to the maximum Y-intercept.
=MINZ: project along the Z-axis to the minimum Z-intercept.
=MAXZ: project along the Z-axis to the maximum Z-intercept.

**TS** $p_x p_y p_z q_x q_y q_z r_1 t_1 r_2$

Define a torus. $(p_x, p_y, p_z)$ is a point on the primary axis of rotation and $(v_x, v_y, v_z)$ is a vector which orients this axis. $r_1$ is the radius to the secondary axis, $t_1$ is an axial offset relative to $(p_x, p_y, p_z)$ and $r_2$ is the radius from the secondary axis to the torus surface.

**TS2P** $p_x p_y p_z v_x v_y v_z r_1 z_1 r_2 z_2 r_3$

Define a torus with two points on the surface. $(p_x, p_y, p_z)$ is a point on the primary axis of rotation and $(v_x, v_y, v_z)$ is a vector which orients this axis. $r_3$ is the radius of the surface from the secondary axis. If $r_3 > 0$ then the secondary axis lies to the left as one moves from $(r_1, z_1)$ to $(r_2, z_2)$. Otherwise, the axis is to the right.
17. Volume Definitions

This section documents the available solid geometric objects. Solid objects are used by the "VD" command in the control section. The following types are available.

**CR** $p_x p_y p_z v_x v_y v_z l$

Form a solid by spinning two-dimensional line definition $l$ about the axis defined by point $(p_x,p_y,p_z)$ and orientation vector $(v_x,v_y,v_z)$.

**CYF** $p_x p_y p_z v_x v_y v_z r t_{min} t_{max}$

Define a cylinder of radius $r$ and axis defined by point $(p_x,p_y,p_z)$ and orientation vector $(v_x,v_y,v_z)$. The cylinder extends along the axis from $t_{min}$ to $t_{max}$.

**CYLI** $p_x p_y p_z v_x v_y v_z r$

Define a cylinder of radius $r$ and axis defined by point $(p_x,p_y,p_z)$ and orientation vector $(v_x,v_y,v_z)$. The cylinder has infinite length.

**DS** $n$

Digitized surface $n$ is a closed surface which defines a volume.

**RECT** $n x_{min} x_{max} y_{min} y_{max} z_{min} z_{max}$

Define a rectangular solid with $x_{min} \leq x \leq x_{max}$, $y_{min} \leq y \leq y_{max}$ and $z_{min} \leq z \leq z_{max}$. This can be positioned anywhere in space using global coordinate transformation number $n$.

**SD** $n t$

The surface is defined by surface definition $n$ and thickness $t$.

**SPHE** $p_x p_y p_z r$

Define a sphere of radius $r$ and centered at $(p_x,p_y,p_z)$.

**TRIA** $n x_1 y_1 x_2 y_2 x_3 y_3 z_{min} z_{max}$

The solid is a triangular section in the X-Y plane which runs from $z_{min}$ to $z_{max}$ in the Z-direction. $(x_1,y_1)$, $(x_2,y_2)$, and $(x_3,y_3)$ are the three corner points. This can be moved anywhere in space using global coordinate transformation $n$. 

---

17.1
18. Coordinate Transformations

Option 1:

For Option 1, three nodal points must be input. Figure 18-1 shows the orientation of the local coordinate system defined by the three points.

\[ f_1 \]
Flag describing coordinate type for point 1
- "RT": rectangular coordinates
- "CY": cylindrical coordinates \((R, \Theta, Z)\)
- "SP": spherical coordinates \((R, \Theta, \Psi)\)

\[ P_{1x} P_{1y} P_{1z} \]
Coordinates for point 1.

\[ f_2 \]
Flag describing coordinate type for point 2
- "RT": rectangular coordinates
- "CY": cylindrical coordinates \((R, \Theta, Z)\)
- "SP": spherical coordinates \((R, \Theta, \Psi)\)
- "V": point 2 is offset from point 1 by the vector.

\[ P_{2x} P_{2y} P_{2z} \]
Coordinates or vector for point 2

\[ f_3 \]
Flag describing coordinate type for point 3.
- "RT": rectangular coordinates
- "CY": cylindrical coordinates \((R, \Theta, Z)\)
- "SP": spherical coordinates \((R, \Theta, \Psi)\)
- "V": point 3 is offset from point 1 by the vector.

\[ P_{3x} P_{3y} P_{3z} \]
Coordinates or vector for point 3.
Figure 18-1. Coordinate Transformations.
Option 2:

Option 2 allows the following commands in any order.

- **CSCA** \(s\) Scale coordinates by \(s\).
- **D1** Save the current offset position and perform rotations relative to this point.
- **D2** Restore the offset position.
- **L** Copy the previous transformation and begin defining the next system.
- **MATRIX** \(a_{11} a_{21} a_{31} a_{12} a_{22} a_{32} a_{31} a_{32} a_{33}\) Set the transformation to the input 3x3 matrix.
- **MX** \(\Delta x\) Move \(\Delta x\) in the \(x\)-direction.
- **MY** \(\Delta y\) Move \(\Delta y\) in the \(y\)-direction.
- **MZ** \(\Delta z\) Move \(\Delta z\) in the \(z\)-direction.
- **REPE** \(n\) Repeat the current transformation \(n\) times.
- **RX** \(\theta\) Rotate \(\theta\) degrees about the \(X\)-axis.
- **RY** \(\theta\) Rotate \(\theta\) degrees about the \(Y\)-axis.
- **RZ** \(\theta\) Rotate \(\theta\) degrees about the \(Z\)-axis.
- **RXY** Reflect about the \(XY\) plane.
- **RYZ** Reflect about the \(YZ\) plane.
- **RZX** Reflect about the \(ZX\) plane.
- **SAVE** \(n\) The sequence of coordinate transformations is generated starting from coordinate system \(n\) when using the "REPE" command.
- **SCALE** \(s\) Scale coordinates by \(s\).
- **V** \(\Delta x\ \Delta y\ \Delta z\) Move \(\Delta x\), \(\Delta y\), and \(\Delta z\).
- **XSCA** \(s\) Scale \(X\)-coordinates.
- **YSCA** \(s\) Scale \(Y\)-coordinates.
- **ZSCA** \(s\) Scale \(Z\)-coordinates.
- `;` Terminate Option 2.
19. LS-DYNA2D Commands and Materials

Analysis options are code dependent. They can be set either in the control section of the LS-INGRID input file or in the graphics phase. These commands become active when LS-DYNA2D output is selected with the DN2D command (see Control Commands).

BRODE

Define Brode function parameters.

Options:

- **YLD** yld Yield (Ktons)
- **HEIGHT** h Height of burst
- **XBO** x Coordinates of Brode origin (space, time) in LS-INGRID units.
- **YBO** y
- **ZBO** z
- **TBO** t
- **CL** cl Conversion factor - ft. to DYNA length units (default = meters).
- **CT** ct Conversion factor - ms to DYNA time units (default = seconds).
- **CP** cp Conversion factor - psi to DYNA pressure units (default = Pascals).

; Terminate Brode function input.

Note: If "RANG" "COEF", and "GFUN" are specified, a "modified" Brode function will be used in DYNA; otherwise, straight Brode is used.

- **RANG** r1 ... r5 Range values for Brode function.
- **COEF** c1 ... c8 Coefficient values for Brode function.
- **GFUN** g1 ... g7 GFUNC values for Brode Function.

The Brode function is applied to pressure surfaces with load curve number -1.

DBQT i Change default bulk viscosity type from 1 to i:
   =1: standard LS-DYNA2D
   =2: Richards-Wilkins

DHGQ Qh Change default hourglass viscosity from .10 to Qh.

DHQT i Change default hourglass viscosity type from 1 to i:
=1: standard LS-DYNA2D  
=2: rotational  
=3: Flanagan-Belytschko viscous form.  
=4: Hancock

**DQL** $Q_l$  
Change default linear bulk viscosity from .06 to $Q_l$.

**DQQ** $Q_q$  
Change default quadratic bulk viscosity from 1.5 to $Q_q$.

**GEOM**  
Select geometry type.  
=AXIS: axisymmetric (default)  
=PLAN: plane strain

**GRAV** $g_x g_y g_z$  
Gravity acceleration vector.

**ITSS** $t_o$  
Initial time step size. This is optional input for LS-DYNA3D. If $t_o$ is zero, LS-DYNA3D picks the initial time step size.

**PLTI** $Dt$  
Node and element data dump interval for TAURUS post-processing.

**PRTI** $Dt$  
Node and element data dump interval for high speed printer.

**RDMT** $m$  
Delete material $m$. This applies to the restart number selected by the RNUM command.

**RDSI** $s$  
Delete sliding interface $s$. This applies to the restart number selected by the RNUM command.

**REST** name  
Set the family name for restart input file generation to name.

**REZO** $t_1 t_2 Dt$  
Periodic rezones begin at time $t_1$ and end at time $t_2$. Rezones are performed after every time interval of $Dt$.

**RHVC** $h$  
The default hourglass viscosity for restart is set to $h$. This applies to the restart number selected by the RNUM command.

**RLBV** $l$  
The default linear bulk viscosity for restart is set to $l$. This applies to the restart number selected by the RNUM command.

**RNUM** $n$  
Restart commands apply to restart number $n$.

**RPLT** $t$  
The plot interval for restart is set to $t$. This applies to the restart number selected by the RNUM command.
**RPRT** \( t \)  
The print interval for restart is set to \( t \). This applies to the restart number selected by the RNUM command.

**RQBV** \( q \)  
The default quadratic bulk viscosity for restart is set to \( q \). This applies to the restart number selected by the RNUM command.

**RTERM** \( t \)  
The termination time for this restart is \( t \). This applies to the restart number selected by the RNUM command.

**RTSF** \( s \)  
The time step scale factor for restart is set to \( s \). This applies to the restart number selected by the RNUM command.

**SBRF** \( n \)  
Number of time steps between restart dumps is \( n \).

**SFSI** \( s \)  
Sliding interface scale factor (default = 1.0).

**TEO** \( i \)  
Thermal effects option

\( =0 \): no thermal effects.

\( =N \): nodal temperatures are defined in input and are scaled according to a time function. \( N \) is the load curve number.

\( =-1 \): each time step a new temperature state is read from a disk file. The time word at the beginning of each temperature state is ignored.

\( =-2 \): each time step a temperature state is interpolated from the temperature state in a disk file. Therefore the time words at the beginning of each temperature state are used.

\( =-3 \): the disk file containing temperatures has only one state. The initial state is assumed to be zero.

**TERM** \( t \)  
Terminate dynamic time integration at time \( t \).

**TINV** \( n \)  
Number of time steps between dumps of reaction history blocks.

**TSSF** \( s \)  
Scale factor on time step size.
19.1 **LS-DYNA2D MATERIAL INPUT**

LS-DYNA2D material input is possible after the DN2D command is input (see *Control Commands*). The form of this input is: `MAT n TYPE m {options specific to material type m} {general material options} ENDMAT`. *n* is a material name which is assigned an input number. Therefore, the materials should be defined in order before any additional use of materials is made.

19.2 **GENERAL MATERIAL OPTIONS**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BQL $Q_l$</td>
<td>Change linear bulk viscosity for .06 to $Q_l$.</td>
</tr>
<tr>
<td>BQQ $Q_q$</td>
<td>Change quadratic bulk viscosity from 1.5 to $Q_q$.</td>
</tr>
</tbody>
</table>
| BQT $i$ | Change bulk viscosity type from 1 to $i$:  
=1: standard LS-DYNA2D  
=2: Richards-Wilkins |
| EOS $eost$ | Begin defining equation-of-state type $eost$ for the current material definition. Each equation-of-state is terminated by the ENDEOS command. |
| HEAD | Replace default heading (typed on the next line). |
| HGQ $Q_h$ | Change value of hourglass viscosity from .10 to $Q_h$. |
| HGQT | Change value of hourglass viscosity type from 1 to $i$:  
=1: standard LS-DYNA2D  
=2: rotational  
=3: Flanagan-Belytschko viscous form  
=4: Hancock |
| IMGL | Initialize material for gravity loads. |
| MAT $m$ | Begin material definition $m$. Each material definition is terminated by the ENDMAT command. |
| RO $m$ | Density (required - no default). |
| TYPE $n$ | The current material is of type $n$. |
Material Type 1 (Elastic)

Default heading: Material Type #1 (Elastic)

Input any two of the following.

\[
\begin{array}{lcl}
\text{BULK } K & \text{Bulk modulus} \\
\text{E } E & \text{Young’s modulus} \\
\text{G } G & \text{Shear modulus} \\
\text{PR } \nu & \text{Poisson’s ratio}
\end{array}
\]
Material Type 2 (Orthotropic Elastic)

\[ C = T^T C_L T, \]

Where \( T \) is a transformation matrix and \( C_L \) is the constitutive matrix defined in terms of the material constants of the orthogonal material axes, \( a, b, \) and \( c \). The inverse of \( C_L \) is defined as

\[ C^{-1}_L = \begin{bmatrix}
\frac{1}{E_a} & -\nu_{ba} & -\nu_{ca} & 0 & 0 & 0 \\
-\nu_{ab} & \frac{1}{E_b} & -\nu_{cb} & 0 & 0 & 0 \\
-\nu_{ac} & -\nu_{bc} & \frac{1}{E_c} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{G_{ab}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{G_{bc}} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{G_{ca}}
\end{bmatrix} \]
Note that $\frac{\nu_{ab}}{\nu_{ba}} = \frac{\nu_{ca}}{\nu_{ac}} = \frac{\nu_{bc}}{\nu_{cb}}$. 

\[ \frac{\nu_{ab}}{\nu_{ba}} = \frac{\nu_{ca}}{\nu_{ac}} = \frac{\nu_{bc}}{\nu_{cb}}. \]
Figure 19-1. Options for determining principal materials axes: (a) AOPT = 0.0, (b) AOPT = 1.0, and (c) AOPT = 2.0.
Material Type 3 (Kinematic/Isotropic Elastic/Plastic)

Default heading: Material Type #3 (Elastic-Plastic)

Input any two of the following:

- **BULK K** Bulk modulus
- **E E** Young's modulus
- **G G** Shear modulus
- **PR ν** Poisson's ratio

Additional Options:

- **SIGY σ_y** Yield stress
- **ETAN E_t** Hardening modulus
- **BETA β'** Hardening parameter, 0 ≤ β' ≤ 1

Isotropic, kinematic, or a combination of isotropic and kinematic hardening may be specified by varying β' between 0 and 1. For β' equal to 0 and 1, respectively kinematic and isotropic hardening are obtained as shown in Figure 19-2. Effective stress is defined in terms of the deviatoric stress tensor, sij as:

\[
\bar{\sigma} = \left( \frac{3}{2} S_{ij} S_{ij} \right)^{\frac{1}{2}}
\]

where,

\[
S_{ij} = \sigma_{ij} - \frac{1}{3} \sigma_{kk} \delta_{ij}
\]

and effective plastic strain by:

\[
\bar{\varepsilon}^P = \int_0^t \varepsilon_{ij}^p d \varepsilon_{ij}^p
\]

where t denotes time and

\[
\varepsilon_{ij}^p = \left( \frac{2}{3} \varepsilon_{ij}^p \varepsilon_{ij}^p \right)^{\frac{1}{2}}
\]
Figure 19-2. Elastic-plastic behavior with isotropic and kinematic hardening where $l_0$ and $l$ are undeformed and deformed length of uniaxial tension specimen.
Material Type 4 (Thermo-Elastic-Plastic)

Default heading: Material Type 4 (Thermo-Elastic-Plastic)

- **NPTS** $n$  Number of temperature values for which material constants are defined.
- **TEMP** $T_1 T_2 ... T_n$  Temperatures
- **E** $E_1 E_2 ... E_n$  Young's moduli
- **PR** $\nu_1 \nu_2 ... \nu_n$  Poisson's ratios
- **ALPHA** $\alpha_1 \alpha_2 ... \alpha_n$  Coefficients of thermal expansion.
- **SIGY** $\sigma_y 1 \sigma_y 2 ... \sigma_y n$  Yield stresses
- **ETAN** $E_t 1 E_t 2 ... E_t n$  Tangent moduli

Material Type 5 (Soil and Crushable Foam)

Default heading: Material Type 5 (Soil and Crushable Foam)

Input any two of the following:

- **BULK** $K$  Bulk modulus
- **E** $E$  Young's modulus
- **G** $G$  Shear modulus
- **PR** $\nu$  Poisson's ratio

Additional Options:

- **AO** $a_0$  Yield function constant.
- **A1** $a_1$  Yield function constant.
- **A2** $a_2$  Yield function constant.
- **PC** $P_c$  Pressure cutoff for tensile fracture.
- **NPTS** $n$  Number of points in volumetric strain versus pressure curve ($n \leq 10$).
- **VS** $\varepsilon_v 1 \varepsilon_v 2 ... \varepsilon_v n$  Volumetric strain values.
- **P** $p_1 p_2 ... p_n$  Pressures corresponding to volumetric strain values.

The deviatoric yield function, $\phi$, is described in terms of the second invariant $J_2$.

$$J_2 = \frac{1}{2} s_j s_j$$
Pressure, \( p \), and constants \( a_0, a_1, \) and \( a_2 \) as:

\[
\phi = J_2 - \left[ a_0 + a_1 p + a_2 p^2 \right].
\]

On the yield surface, \( J_2 = 1/3 (\sigma_y)^{1/2} \), where \( \sigma_y \) is the yield stress, i.e.,

\[
\sigma_y = \left[ 3 (a_0 + a_1 p + a_2 p^2) \right]^{1/2}
\]

For elastic-perfectly plastic behavior \( a_1 = a_2 = 0 \), and \((3a_0)^{1/2}\) defines the yield strength.

The volumetric strain is given by the natural logarithm of the relative volume \( V \). If the pressure drops below the cutoff value, PC, then it is reset to that value.

---

**Figure 19-3.** Volumetric strain versus pressure curve for soil and crushable foam model.
Material Type 6 (Viscoelastic)

\[
\begin{align*}
G & \quad G_0 \quad \text{Short term shear modulus.} \\
G & \quad G' \quad \text{Long term shear modulus.} \\
K & \quad K \quad \text{Bulk modulus.} \\
BETA & \quad \beta \quad \text{Decay constant.}
\end{align*}
\]

The shear relaxation behavior is described by:

\[
G(t) = G + (G_0 - G') e^{-\beta t}
\]

A Jaumann rate formulation is used:

\[
\overline{\sigma}_{ij} = 2 \int_0^t G(t - \tau) D_{ij}(\tau) d\tau
\]

where the prime denotes the deviatoric part of the stress rate, \(\overline{\sigma}_{ij}\), and the strain rate \(D_{ij}\).

Material Type 7 (Blatz-Ko Rubber)

Default heading: Material Type #7 (Rubber)

\[
G \quad G_m \quad \text{Shear modulus}
\]

The second Piola-Kirchhoff stress is computed as

\[
S_{ij} = \mu \left( \frac{1}{V} C_{ij} - V^{-\frac{1}{2}} \nu \delta_{ij} \right)
\]

where \(V\) is the relative volume, \(C_{ij}\) is the right Cauchy-Green strain tensor, and \(\nu\) is the Poisson's ratio which is set to .463 internally. This stress measure is transformed to the Cauchy stress, \(\sigma_{ij}\), according to the relationship

\[
\sigma_{ij} = V^{-1} F_{ik} F_{jl} S_{lk}
\]

where \(F_{ij}\) is the deformation gradient tensor.

Material Type 8 (High Explosive Burn)
Default heading: Material Type #8 (High Explosive Burn)

\[ D \] Detonation velocity.

\[ PCJ \] Chapman-Jouget pressure.

This material model requires an equation-of-state.

**Material Type 9 (Null Material)**

Default heading: Material Type #9 (Null Material)

\[ PC \text{ pc} \] Pressure cutoff.

\[ MU \mu \] Viscosity.

The null material must be used with an equation-of-state. Pressure cutoff is negative in tension. A viscous stress of the form

\[ \sigma_{ij} = \mu \dot{\varepsilon}^{Y}_{ij} \]

is computed for nonzero \( \mu \) where \( \dot{\varepsilon}^{Y}_{ij} \) is the deviatoric strain rate.

**Material Type 10 (Isotropic-Elastic-Plastic-Hydrodynamic)**

Default heading: Material Type #10 (Isotropic-Elastic-Plastic-Hydrodynamic)

\[ G \text{ G} \] Shear modulus

\[ SIGY \sigma_{y} \] Yield strength

\[ EH E_{h} \] Plastic hardening modulus

\[ PC p_{c} \text{ or } -s_{f} \] Pressure cutoff

\[ A1 a_{1} \] Yield function constant

\[ A2 a_{2} \] Yield function constant

\[ NPTS n \] Number of points in yield stress-effective plastic strain curve or yield stress-pressure curve.

\[ ES \sigma_{y1} \sigma_{y2} \ldots \sigma_{yn} \] Yield stress

\[ EPS \varepsilon_{p1} \varepsilon_{p2} \ldots \varepsilon_{pn} \] Effective plastic strain

\[ P p_{1} p_{2} \ldots p_{n} \] Pressure

\[ FS s \] Failure strain
If the yield stress-plastic strain curve is not defined and if $a_1 = a_2 = 0$, the bilinear stress-strain curve shown in Figure 19-2 is obtained with $b = 1$. The yield strength is calculated as

$$\sigma_y = \sigma_0 + E_h \varepsilon^p$$

The quantity $E_h$ is the plastic hardening modulus defined in terms of Young’s modulus, $E$, and the tangent modulus, $E_t$, as follows

$$E_h = \frac{E_t E}{E - E_t}$$

If Cards 5-8 are used, a curve like that shown in Figure 3.4 may be defined. Effective stress is defined in terms of the deviatoric stress tensor, $s_{ij}$, as:

$$\bar{\sigma} = \left( \frac{3}{2} s_{ij} s_{ij} \right)^{1/2}$$

and effective plastic strain by:

$$\varepsilon^p = \int_0^t \left( \frac{2}{3} D_{ij}^p D_{ij}^p \right)^{1/2} dt$$

where $t$ denotes time and $D_{ij}^p$ is the plastic component of the rate of deformation tensor. In this case the plastic hardening modulus on Card 3 is ignored and the yield stress is given as

$$\sigma_y = f(\varepsilon^p)'$$

where the value for $f(\varepsilon^p)'$ is found by interpolation from the data curve.
Piecewise linear curve defining the yield stress versus effective plastic strain. A nonzero yield stress is defined when the plastic strain is zero.

Figure 19-4. Effective stress versus effective plastic strain curve.

Material Type 11 (Temperature Dependent Elastic-Plastic Hydrodynamic)

Default heading: Material Type #11 (Temperature Dependent Elastic-Plastic Hydrodynamic)

\[ G G_0 \]
Shear modulus.

\[ \text{SIGO } \sigma_o \]
See equations below.

\[ \text{BETA } \beta \]

\[ \text{N } n \]

\[ \text{GAMA } g_i \]

\[ \text{SIGM } \sigma_m \]

\[ B b \]

\[ \text{BP } b' \]

\[ H h \]

\[ F f \]

\[ A A \]

\[ \text{TO } T_{mo} \]

\[ \text{GAMO } \gamma_0 \]

\[ \text{SA } a \]
PC \text{ pmin or -sf}
ECO \text{ EC}_0 \quad \text{Cold compression energy coefficients (optional)}
EC1 \text{ EC}_1
EC2 \text{ EC}_2
EC3 \text{ EC}_3
EC4 \text{ EC}_4
EC5 \text{ EC}_5
EC6 \text{ EC}_6
EC7 \text{ EC}_7
EC8 \text{ EC}_8
EC9 \text{ EC}_9

If cold compression energy coefficients are not input, then LS-DYNA2D will calculate them based on the equation-of-state.

**SPALL type**

Spall type
- \text{=0: default set to "2.0"}
- \text{=1: } p \geq p_{\text{min}}\text{ p}_{\text{min}}
- \text{=2: } \text{if } \sigma_{\text{max}} \geq \sigma_{f} \text{ element spalls and tension}
- \text{p} < 0, \text{ is never allowed; } \sigma_{\text{max}} = \text{maximum principal stress.}
- \text{=3: } \text{if } p < p_{\text{min}} \text{ element spalls and tension}
- \text{p}<0, \text{ is never allowed.}
- \text{=4: failure strain}

Users who have an interest in this mode are encouraged to study the paper by Steinberg and Guinan [9] which provides the theoretical basis. Another useful reference is the KOVEC user's manual [10].

In terms of the foregoing input parameters, we define the shear modulus, \( G \), before the material melts as:

\[
G = G_0 \left[ 1 + b p V^3 - h \left( \frac{E_i - E_c}{3R'} - 300 \right) \right] e^{-\frac{fE_i}{E_m - E_i}}
\]

where \( p \) is the pressure, \( V \) is the relative volume, \( E_c \) is the cold compression energy:

\[
\bar{\varepsilon} p = \int_0^t (\frac{2}{3} D_{ij} p D_{ij} p)^{\frac{1}{2}} \, dt
\]

\[ x = 1 - V \]

and \( E_m \) is the melting energy:

\[
E_m (x) = E_c (x) + 3R'T_m (x)
\]
which is in terms of the melting temperature $T_m(x)$:

$$T_m(x) = \frac{T_{m0} \exp (2a \chi)}{\sqrt{2(y_0-a^{-\frac{1}{3}})}}$$

and the melting temperature at $r = r_0$, $T_{mo}$.

In the above equation, $R'$ is defined by

$$R' = \frac{R \rho}{A}$$

where $R$ is the gas constant and $A$ is the atomic weight. If $R'$ is not defined, LS-DYNA2D computes it with $R$ in the cm-gram-microsecond system of units.

The yield strength $\sigma_y$ is given by:

$$\sigma_y = \sigma'_0 \left[ 1 + b' \rho \sqrt{h} - h \left( \frac{E_i - E_c}{3R'} - 300 \right) \right] e^{-\frac{E_i}{3R'} - E_i}$$

if $E_m$ exceeds $E_i$. Here, $\sigma'_0$ is given by:

$$\sigma_y = \sigma'_0 \left[ 1 + \beta \left( \gamma_i + \epsilon_i^{\beta} \right)^n \right]$$

where $\epsilon_i$ is the initial plastic strain. Whenever $\sigma'_0$ exceeds $\sigma_m$, $\sigma'_0$ is set equal to $\sigma_m$. After the material melts, $\sigma_y$ and $G$ are set to zero.

If the coefficients $E_{CO}...EC9$ are not defined above, LS-DYNA2D will fit the cold compression energy to the ten term polynomial expansion:

$$E_c = \sum_{i=0}^{9} E_{C_i} \eta^i$$

where $E_{C_i}$ is the $i$th coefficient and $\eta = \rho/\rho_0 - 1$. The least square method is used to perform the fit.

**Material Type 12 (Johnson/Cook Plasticity Model)**

**G G**

Shear modulus.

**A A**

See equation (1).

**B B**

See equation (1).

**N n**

See equation (1).

**R r**

See equation (1).
The Johnson/Cook model is described in reference [11]. This includes strain rate hardening, thermal softening, and a complex damage model. The equations describing the flow stress versus effective plastic strain and failure strain are as follows:

\[ \sigma_y = \left( A + B \bar{\varepsilon}^p \right) \left( l + c \ln \dot{\varepsilon}^p \right) \left( l - T^* m \right) \]

where \( A, B, C, n, \) and \( m \) are input constants,

\( \bar{\varepsilon}^p \) effective plastic strain

\( \dot{\varepsilon}^p = \frac{\dot{\varepsilon}^p}{\dot{\varepsilon}_0} \) effective plastic strain rate for \( \dot{\varepsilon}_0 = 1 \text{ s}^{-1} \)

\( T^* = T_r / T_m \) homologous temperature

Constants for a variety of materials are also provided in [11].

Due to the nonlinearity in the dependence of flow stress on plastic strain, an accurate value of the flow stress requires iteration for the increment in plastic strain. However, by using a Taylor series expansion with linearization about the current time, we can solve for \( \sigma_y \) with sufficient accuracy to avoid iteration.

The strain at fracture is given by

\[ \varepsilon^f = \left[ D_1 + D_2 \exp D_3 \sigma^* \right] \left( l + D_4 \ln \dot{\varepsilon}^p \right) \left( l + D_5 T^* \right) \]

where \( \sigma^* \) is the ratio of pressure divided by effective stress:
Fracture occurs when the damage parameter

\[ D = \sum \frac{\Delta \overline{\varepsilon}^p}{\varepsilon^T} \]

reaches the value of 1.

**Material Type 13 (Power Law Plasticity)**

Input any two of the following:

- **BULK K** Bulk modulus.
- **E E** Young’s modulus.
- **G G** Shear modulus.
- **PR ν** Poisson’s ratio.

**Additional Options:**

- **K k** See equation below.
- **M m** See equation below.

Elastoplastic behavior with isotropic hardening is provided by this model. The yield stress, \( \sigma_y \), is a function of plastic strain and obeys the equation:

\[ \sigma_y = k \left( \varepsilon_e + \overline{\varepsilon}^p \right)^n \]

\( \varepsilon_e \) is the elastic strain to yield and where \( \overline{\varepsilon}^p \) is the effective plastic strain. ***missing***

**Material Type 16 (Pseudo Tensor Geological Model)**

Default heading: Material Type #16 (Pseudo Tensor Geological Model)

- **G G** Shear modulus (constant Shear modulus model).
LS-INGRID  

**LS-DYNA2D COMMANDS AND MATERIALS**

**PR** \( \nu \)  
Poisson's ratio (constant Poisson's ratio model).

**SIGF** \( \text{sigf} \)  
Tensile cutoff. (Maximum principal stress for failure.)

**A0** \( a_0 \)  
Cohesion.

**A1** \( a_1 \)  
Yield function constant.

**A2** \( a_2 \)  
Yield function constant.

**A0F** \( a_{0f} \)  
Cohesion for failed material.

**A1F** \( a_{1f} \)  
Pressure hardening coefficient for failed material.

**B1** \( b_1 \)  
Damage scaling factor.

**PER** \( \rho \)  
Percent reinforcement.

**ER** \( E_r \)  
Elastic modulus for reinforcement.

**PR** \( \nu_r \)  
Poisson's ratio for reinforcement.

**SIGY** \( \sigma_y \)  
Initial yield strength.

**ETAN** \( E_t \)  
Tangent modulus.

**LCP** \( l_{c_1} \)  
Load curve giving rate sensitivity for principal material.

**LCR** \( l_{c_2} \)  
Load curve giving rate sensitivity for reinforcement.

**NPTS** \( n \)  
Number of points in yield stress-effective plastic strain curve or yield stress-pressure curve; \((n \leq 16)\).

**ES** \( \sigma_1 \sigma_2 \ldots \sigma_n \)  
Yield stress.

**EPS** \( \varepsilon_{p1} \varepsilon_{p2} \ldots \varepsilon_{p3} \)  
Effective plastic strain.

**P** \( p_1 \ p_2 \ldots \ p_n \)  
Pressure.

See the LS-DYNA2D manual for a description of this model.

---

**Material Type 25 (Inviscid Two Invariant Geologic Cap Model)**

**G** \( G \)  
Shear Modulus.

**K** \( K \)  
Bulk Modulus.

**ALPHA** \( \alpha \)  
\( \alpha \).

**BETA** \( \beta \)  
\( \beta \).

**GAMMA** \( \gamma \)  
\( \gamma \).

**THETA** \( \theta \)  
\( \theta \).
**R R**

**D D**

**X0 X0**

**CC C**

**T T**

**NPLOT nplot**

Save the following variable for plotting in ORION:

- 1: k
- 2: X
- 3: $e^p$
- 4: $J_1$
- 5: $(J_2)^{1/2}$
- 6: $(J_2|_L)^{1/2}$
- 7: $(J_2|_{tr})^{1/2}$
- 8: MTYPE
- 9: number of iterations

**LTYPE ltype**

Variable ltype.

- 1: soil/concrete (cap contracts)
- 2: rock (cap doesn't contract)

Figure 19-5. The yield surface of the two-invariant cap model in pressure/$J_2$-deviator space. $f_1 = 0$, $f_2 = 0$, and $f_3 = 0$, denote the failure envelope, the hardening cap surface, and the tension cut-off surface, respectively.

The shaded area in Figure 19-5 is the “compressive corner regions”.

---

20.22
$$f_1(\sigma) := \| \mathbf{s} \| - F_e \left( J_1 \right), \quad \text{for} \quad -T \leq J_1 < \kappa$$

$$f_2(\sigma, \kappa) := \| \mathbf{s} \| - F_c \left( J_1, \kappa \right), \quad \text{for} \quad \kappa \leq J_1 < X(\kappa)$$

$$f_3(\sigma) := -T - J_1, \quad \text{for} \quad -T \leq J_1 < \kappa$$

where $J_1 := \text{tr} \sigma$, $\| \mathbf{s} \| := \sqrt{\mathbf{s}^T \mathbf{s}}$, with $\mathbf{s} := \sigma - \frac{1}{3} \left( \text{tr} \sigma \right) \mathbf{1}$.

In addition, $T > 0$ is a material constant referred to as the tension cutoff. Note that the following standard conventions in soil mechanics, we have assumed compression and compaction positive. Functional forms for $F_e$ and $F_c$ used are:

$$F_e \left( J_1 \right) := \left[ \alpha - \lambda \exp \left( -\beta J_1 \right) + \theta J_1 \right]$$

$$F_c \left( J_1, \kappa \right) := \sqrt{F_e^2(\kappa) - \frac{\left[ J_1 - \kappa \right]^2}{R^2}},$$

where $a > 0$, $\lambda > 0$, $\beta > 0$, $\theta > 0$, and $R > 0$ are material parameters. In addition, $X(\kappa)$ is a function of the hardening parameter $\kappa$ defined as

$$X(\kappa) := \kappa + R F_e(\kappa).$$

$$\dot{\mathbf{h}}(\kappa) := \max \left[ \frac{\mathbf{h}'(\kappa) \text{tr} \dot{\mathbf{e}}^p}{F_e(\kappa)}, \frac{d}{dt} \dot{\mathbf{e}}^p \right],$$

where

$$\dot{\mathbf{h}}(\kappa) := W \left[ 1 - \exp \left( -D \left( X(\kappa) - X_0 \right) \right) \right]$$

otherwise
20. LS-DYNA3D Commands and Materials

Analysis options are code dependent. They can be set either in the control section of the LS-INGRID input file or in the graphics phase. These commands become active when LS-DYNA3D output is selected with the DN3D command (see Control Commands).

**ARBITRARY**

Node and element numbering is arbitrary. (LS-902 and later).

**BRODE**

Define Brode function parameters.

*Options:*

- **YLD** yld
  Yield (Ktons)
- **HEIGHT** h
  Height of burst
- **XBO** x
  Coordinates of Brode origin (space, time) in LS-INGRID units.
- **YBO** y
- **ZBO** z
- **TBO** t
- **CL** cl
  Conversion factor - ft. to DYNA length units (default = meters).
- **CT** ct
  Conversion factor - ms to DYNA time units (default = seconds).
- **CP** cp
  Conversion factor - psi to DYNA pressure units (default = Pascals).
- **;**
  Terminate Brode function input.

Note: If "RANG" "COEF", and "GFUN" are specified, a "modified" Brode function will be used in DYNA; otherwise, straight Brode is used.

- **RANG** r1 ... r5
  Range values for Brode function.
- **COEF** c1 ... c8
  Coefficient values for Brode function.
- **GFUN** g1 ... g7
  GFUNC values for Brode Function.

The Brode function is applied to pressure surfaces with load curve number -1.

**BRUL** n

Begin definition of user specified integration rule for beams number n.
Options:

NPTS $n s_1 t_1 w_1 <m_1> \ldots$ Input $n$ integration points with the parametric coordinate, $(s,t)$ and the weight, $w$. This terminates the rule.

BUPD opt Flag for updating coordinates of reference node for beam elements. Values of opt are "on" or "off". (LS-910 and later).

CUNI length time force Unit conversion factors for coupling between LS-DYNA3D and CAL3D or MADYMO3D. (LS-910 and later).

D2R $m_1 m_2$ Convert material $m_1$ from deformable to rigid. If $m_2$ is 0, then this is an independent rigid body. Otherwise, $m_2$ is the master rigid body material. If a restart file definition has been initiated, then this command applies to the restart. Otherwise, it applies to the main DYNA3D input. (LS-920 and later.)

D3HSP Additional output options for the D3HSP and message files.

Options:

DEBUG opt Option for producing debug output on calculation progress in the message file. Values for opt are either "on" or "off". (LS-910 and later).

ECHO opt Additional suppression options for printout. (LS-910 and later).

=0: all data is printed.
=1: nodal printing is suppressed.
=2: element printing is suppressed.
=3: both node and element printing are suppressed.

IKEDIT $n$ Number of time steps between writing global statistics data to D3HSP file. (Default = 1000.)

SUPP opt Performs suppression of output echo. Values for opt are either "on" or "off". (LS-910 and later).

TSTEP opt Print flag for element time step sizes on first cycle. Values for opt are either "on" or "off". (LS-910 and later).

; Terminate the D3HSP command.
DBQT $i$
Change default bulk viscosity type from 1 to $i$:
- $i=1$: standard LS-DYNA3D

DELT $\Delta t$
Set time step for mass scaled calculations to $\Delta t$.
Note that this is an advanced option. Normally
LS-DYNA3D sets the time step. Study the
mass scaling option in LS-DYNA3D before
using this option. (LS-910 and later).

DHGQ $Qh$
Set default hourglass viscosity from .10 to $Qh$.

DHQT $i$
Set default hourglass viscosity type from 1 to $i$:
- $i=1$: standard LS-DYNA3D
- $i=2$: Flanagan-Belytschko viscous form
- $i=3$: Flanagan-Belytschko viscous form with
  exact volume integration.
- $i=4$: Flanagan-Belytschko stiffness form
- $i=5$: full Flanagan-Belytschko stiffness form
  with exact volume integration.

DQL $Ql$
Set default linear bulk viscosity for .06 to $Ql$.

DQQ $Qq$
Set default quadratic bulk viscosity from 1.5 to $Qq$.

DROPTS
Select dynamic relaxation options. There are
three separate methods in LS-910 and later for
performing dynamic relaxation. The first uses the
SYSD or LCDAMP commands and is designated
"Method A". This is the recommended method.
The second is designated "Method B" and
requires the options below which must be set with
care. The third is a method due to Papadrakakis
and is designated "Method C". Only one method
should be used at a time to avoid confusion.

Options:

**DRFCTR** $d$
Damping factor expressed as $V_{n+1}=d*V_n$. This
should be set with care based on the formulas in
the DYNA3D Course Notes. (Method B)

**DRTOL** tol
Tolerance on distortional kinetic energy for
determining convergence. (Method B)

**NRCYCK** $n$
Number of time steps between convergence
checks. (Method B)

**TSSFDR** tssfdr
Time step scale factor during dynamic relaxation.
(Metho B)

**DRTERM** $t$
Termination time for dynamic relaxation
simulation should convergence not be obtained.
LS-DYNA3D COMMANDS AND MATERIALS

FLUID
Set ALE and Eulerian options. (VECALE, LS-930 and later.)

Options:

ADVECTION opt
Set the advection formulation.

\( opt = 1 \): first order. (SALE Method)
\( opt = 2 \): second order. (Benson HIS)
\( opt = 3 \): second order. (Van-Leer)

ALE
The element formulation is Arbitrary Lagrangian-Eulerian.

EULERIAN
The element formulation is Eulerian.

LAGRANGIAN
The element formulation is Lagrangian (default).

MAT m
These options apply to material m. (The default is that the specified fluid options apply globally to the model.)

NCYCLES n
The number of cycles between smoothing and advection (ALE) or smoothing (Eulerian) is n.

RELAX1 r_1
Weight for simple average relaxation method.

RELAX2 r_2
Weight for Kikuchi relaxation method.

RELAX3 r_3
Weight for isoparametric relaxation method.

RELAX4 r_4
Weight for equipotential relaxation method.

START t_s
Start time for ALE.

STOP t_e
Stop time for ALE.

; End of dynamic relaxation options.

FRES m_1 m_2 ... ; Perform a full restart. Materials m_1, m_2, ... are to be remapped. If "FRES ;" is input, then all materials will be remapped.
LS-INGRID

LS-DYNA3D COMMANDS AND MATERIALS

GMPRT

Input general printing option intervals. (LS-910 and later).

Options:

- ABSTAT t: Airbag statistics.
- AVSFLT t: AVS filter.
- BCOU t: Boundary condition forces.
- BELT t: Seat belt output file.
- DEFGEO t: Smug animator file.
- DEFORC t: Discrete element.
- ELOUT t: Element data.
- GEFORC t: Geometric entity resultants.
- GLSTAT t: Global data.
- JOINTS t: Joint file.
- MATSUM t: Material energies.
- MOVIE t: Movie-BYU output file.
- MPGS t: MPGS output.
- NCFORCE t: Nodal interface forces.
- NFG t: Nodal force groups.
- NODOUT t: Nodal point data.
- RBOU t: Rigid body acceleration output.
- RCFORC t: Resultant interface forces.
- RWFORC t: Rigid wall forces.
- SECFORCE t: Section forces.
- SIDB t: Sliding interface database.
- SPCFORC t: SPC reaction forces.
- SWFORC t: Nodal constant resultants.
- TRACER
- VARIABLE typ icomp: Components for ASCII state output. typ can be AVS, MOVIE or MPGS. The component number is icomp.

; Terminate this command.

GRAV g x g y g z

Gravity acceleration vector.

HGENERGY on/off

Option for computing hourglass energy dissipation. (Default=Off.) (LS-910 and later).

IARB on/off

Selection for material input method. If on, then the material input is broken into separate constitutive model, equation-of-state and section property sections. LS-INGRID can convert from one method to another during generation. The last method selected applies to the output file. (Default=Off.) (LS-910 and later).

IRDMS on/off

Turn on deformable to rigid switching. (LS-920 and later.)

ITSS t o

Initial time step size. This is optional input for LS-DYNA3D. If t o is zero, LS-DYNA3D picks the initial time step size.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCDAMP lc</td>
<td>System damping is specified by load curve lc. (LS-902 and later).</td>
</tr>
<tr>
<td>LCGX lcgx</td>
<td>Load curve number for X-body load. (default=1).</td>
</tr>
<tr>
<td>LCGY lcgy</td>
<td>Load curve number for Y-body load. (default=1).</td>
</tr>
<tr>
<td>LCGZ lcgz</td>
<td>Load curve number for Z-body load. (default=1).</td>
</tr>
<tr>
<td>LCRX lcrx</td>
<td>Load curve number for X-centrifugal load. (default=1).</td>
</tr>
<tr>
<td>LCRY lcry</td>
<td>Load curve number for Y-centrifugal load. (default=1).</td>
</tr>
<tr>
<td>LCRZ lcrz</td>
<td>Load curve number for Z-centrifugal load. (default=1).</td>
</tr>
<tr>
<td>LCMA lc</td>
<td>lc is a load curve which specifies the maximum time step as a function of time.</td>
</tr>
<tr>
<td>MVMA</td>
<td>Output is generated which is compatible with MVMA-DYNA3D.</td>
</tr>
<tr>
<td>NCPU n</td>
<td>Use n CPU’s for parallel processing. (LS-920 and later.)</td>
</tr>
<tr>
<td>NEWC</td>
<td>Use new contact formulations. (LS-902, VEC/DYNA3D). This turns on the eroding contact in VEC/DYNA3D.</td>
</tr>
<tr>
<td>NSTEP n</td>
<td>The number of time steps for mass scaled calculations is n. Note that this is an advanced option and normally LS-DYNA3D sets the time step. Study the mass scaling option in LS-DYNA3D before using this option. (LS-910 and later).</td>
</tr>
<tr>
<td>OPIFS n</td>
<td>Output interval for interface file.</td>
</tr>
<tr>
<td>PASS opt</td>
<td>Option for sorting parallel assembly of the right hand side. Values for opt are &quot;on&quot; or &quot;off&quot;. (LS-920 and later).</td>
</tr>
<tr>
<td>PERCENT n</td>
<td>Maximum allowable change in total energy in percent.</td>
</tr>
<tr>
<td>PLTI Δt</td>
<td>Node and element data dump interval for TAURUS post-processing.</td>
</tr>
<tr>
<td>PRTI Δt</td>
<td>Node and element data dump interval for high</td>
</tr>
</tbody>
</table>
speed printer.

**PSPO** iopt  
Plane stress iteration flag.  
=1: iterative plasticity with 3 secant iterations (default).  
=2: full iterative plasticity.  
=3: radial return non-iterative plasticity.  
(quick and very dirty.)

**R2D** $m_1$ $m_2$  
Convert material $m_1$ from rigid to deformable. If a restart file definition has been initiated, then this command applies to the restart. Otherwise, it applies to the main DYNA3D input. (LS-920 and later.)

**RDENERGY** on/off  
Option for computing stone wall energy dissipation. (Default=Off.) (LS-910 and later).

**RDMT** $m$  
Delete material $m$. This applies to the restart number selected by the RNUM command.

**RDSI** $s$  
Delete sliding interface $s$. This applies to the restart number selected by the RNUM command.

**REIN** $i$  
Hughes-Liu shell normal initialization count.  
$i = -2$: unique nodal fibers per Hughes-Liu  
$= -1$: compute normals each cycle (default).  
$= 1$: compute on restart.  
$= n$: compute on restart and every nth cycle.

**REST** name  
Set the family name for restart input file generation to name.

**RHVC** $h$  
The default hourglass viscosity for restart is set to $h$. This applies to the restart number selected by the RNUM command.

**RIRDMS** on/off  
Turn on rigid to deformable switching. (LS-920 and later.)

**RLBV** $l$  
The default linear bulk viscosity for restart is set to $l$. This applies to the restart number selected by the RNUM command.

**RNUM** $n$  
Restart commands apply to restart number $n$.

**RPLT** $t$  
The plot interval for restart is set to $t$. This applies to the restart number selected by the RNUM command.

**RPRT** $t$  
The print interval for restart is set to $t$. This applies to the restart number selected by the RNUM command.
**RQBV** \( q \)  
The default quadratic bulk viscosity for restart is set to \( q \). This applies to the restart number selected by the RNUM command.

**RTERM** \( t \) 
The termination time for this restart is \( t \). This applies to the restart number selected by the RNUM command.

**RTSF** \( s \)  
The time step scale factor for restart is set to \( s \). This applies to the restart number selected by the RNUM command.

**RWPNAL** \( p \)  
Scale factor for rigid body nodes impacting rigid walls. If \( p=0.0 \), then this capability is ignored.

**SBRF** \( n \)  
Number of time steps between restart dumps is \( n \).

**SEQUENTIAL**  
Use sequential node, element and material numbering. (Default)

**SFSI** \( s \)  
Sliding interface scale factor (default = 0.1).

**SIOPT**  
Additional sliding interface options. (LS-910 and later).

**Options:**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ENER</strong> opt</td>
<td>Option for determining sliding interface energy dissipation. Values for opt are &quot;on&quot; and &quot;off&quot;.</td>
</tr>
<tr>
<td><strong>CHECK</strong> opt</td>
<td>Option for performing initial penetration checks on contact interfaces. Values for opt are &quot;on&quot; and &quot;off&quot;.</td>
</tr>
</tbody>
</table>
| **OFFSET** \( n \) | Set shell thickness offset option to \( n \):  
  \( =0 \): thickness is not considered in two surface contacts.  
  \( =1 \): thickness is considered but rigid bodies are excluded.  
  \( =2 \): thickness is considered including rigid bodies. |
| **ORIE** opt | Option for automatically reorienting normals of shell contact segments during initialization. Values for opt are "on" and "off". |
| **PSOPT** \( n \) | Penalty stiffness option:  
  \( =1 \): use minimum of master segment and slave node (default).  
  \( =2 \): use master segment stiffness (old way).  
  \( =3 \): use slave node value.  
  \( =4 \): use slave node value area or mass |
weighted.
=5: same as 4 but inversly proportional to the shell thickness.

**THIN opt**  
Option for including thinning of shells in thickness offsets. Values for opt are "on" and "off".

```
;
```

Terminate SIOPT command.

**SRUL n**  
Begin definition of user specified integration rule for shell number n.

**Options:**

**MATE**  
Include optional material selection.  
(default=off)

**NPTS n t₁ w₁ <m₁> ... tₙ wₙ <mₙ>**  
Input n integration points with the parametric coordinate, t, the weight, w and the optional material number, m. This terminates the rule.

**STYP s**  
Default shell formulation type  
s = HUGHES: use Hughes-Liu shell formulation (default).  
= BELYSCHKO: use Belytschko-Lin-Tsay shell theory.

**SWENERGY on/off**  
Option for computing stone wall energy dissipation.  (Default=Off.)  (LS-910 and later).

**SYSD d**  
System damping constant d.  
(MVMA/DYNA3D, VEC/DYNA3D, LS-902 and later.)

**TAURUS**  
Additional output options for the D3PLOT, D3IFF and D3THDT files.

**Options:**

**AVER opt**  
Output averaged accelerations from velocities in file "nodout" and the time history database file, "d3thdt".  (LS-910 and later).

**CMSO opt**  
Composite material stress output option. Values for opt are "global" and "local".  (LS-910 and later).

**DRDB opt**  
Produce a separate TAURUS database for the dynamic relaxation option. Values for opt are "on" or "off".  (LS-910 and later).

**IFDT Dt**  
Output interval for interface force database. If zero, the default is the same as for complete state dumps.
INT4 \( n \)  Number of additional integration point history variables written to the TAURUS database for shell elements.

INT8 \( n \)  Number of additional integration point history variables written to the TAURUS database for solid elements.

SEPARATE opt  Option for separating D3PLOT file into one state per family output member. Values for opt are "on" or "off". (LS-910 and later).

TINT \( n \)  Number of through thickness integration points written to TAURUS database. (default=3).

;  Terminate TAURUS command options.

TCYCLE \( n \)  The termination cycle is \( n \). (LS-910 and later).

TEO \( i \)  Thermal effects option

=0: no thermal effects.

=N: nodal temperatures are defined in input and are scaled according to a time function. \( N \) is the load curve number.

=-1: each time step a new temperature state is read from a disk file. The time word at the beginning of each temperature state is ignored.

=-2: at each time step a temperature state is interpolated from the temperature state in a disk file. Therefore the time words at the beginning of each temperature state is used.

=-3: the disk file containing temperatures has only one state. The initial state is assumed to be zero.

TERM \( t \)  Terminate dynamic time integration at time \( t \).

TINV \( n \)  Number of time steps between dumps of reaction history blocks.

TSLIMIT \( \Delta t \)  The minimum time step for shell elements of type 3, 18, 19 and 24 cannot go below \( \Delta t \). To enforce this condition, the element stiffness is artificially softened. This is useful when pretty pictures are more important than good results.

TSORT opt  Sort triangular elements to treat degenerate quadrilateral elements with the \( C_0 \) triangular shell formulation. Values for opt are "on" or "off". (LS910 and later).
**LS-INGRID**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSSF $s$</td>
<td>Scale factor on time step size.</td>
</tr>
<tr>
<td>TUPD</td>
<td>Modify shell thickness based on membrane strains (default does not modify shell thickness).</td>
</tr>
<tr>
<td>V90</td>
<td>Output is compatible with LS-DYNA3D version 902.</td>
</tr>
<tr>
<td>V91</td>
<td>Output is compatible with LS-DYNA3D version 910.</td>
</tr>
<tr>
<td>V92</td>
<td>Output is compatible with LS-DYNA3D version 920.</td>
</tr>
<tr>
<td>V93</td>
<td>Output is compatible with LS-DYNA3D version 930. This produces the LS-DYNA3D keyword based input.</td>
</tr>
<tr>
<td>VEC</td>
<td>Output is compatible with VEC-DYNA3D.</td>
</tr>
<tr>
<td>VEC92</td>
<td>Output is compatible with VECALE.</td>
</tr>
<tr>
<td>WARP $\text{ang}$</td>
<td>Shell element warpage angle in degrees. If a warpage greater than this angle is found, a warning message is printed. (default=20.0) (LS-902 and later).</td>
</tr>
<tr>
<td>WEDGE</td>
<td>Normally, LS-INGRID does not allow the generation of wedge elements. This command turns on the support for 6-node and 4-node solid elements.</td>
</tr>
</tbody>
</table>

**20.1 LS-DYNA3D Material Input**

LS-DYNA3D material input is possible after the DN3D command is input (see Control Commands). The form of this input is: MAT $n$ TYPE $m$ {options specific to material type $m$} {general material options} ENDMAT. $n$ is a material name which is assigned a number as input. Therefore, the materials should be defined in order before any additional use of materials is made.

**20.2 General Material Options**

**ANGLES $\beta_1 \beta_2 \ldots \beta_n$**

Input angles for laminated materials. $n$ is the number of integration points; thus, this command cannot be used until after the QUAD command has been used to specify the number of integration points for the current material.
BEAM

This material is defined for two node beam elements only.

BFORM s

Beam formulation type

s = "HUGH": Hughes-Liu
s = "BELY": Belytschko-Schwer
s = "TRUS": Truss

BQL Q_l

Change linear bulk viscosity for .06 to Q_l.

BQQ Q_q

Change quadratic bulk viscosity from 1.5 to Q_q.

BRFORM s

Brick element formulation type (LS-920 and later):

s = 1: standard single point brick
s = 2: fully integrated brick element

BQT i

Change bulk viscosity type from 1 to i:

=1: standard LS-DYNA3D (not much choice)

CAREH a

Cross sectional area for Belytscko-Schwer beam.

EOS eost

Begin defining equation-of-state type eost for the current material definition. Each equation-of-state is terminated by the ENDEOS command.

HEAD

Replace default heading (typed on the next line).

HGQ Q_h

Change hourglass viscosity from .10 to Q_h.

HGQT i

Change type of hourglass viscosity from 1 to i:

=1: standard LS-DYNA3D
=2: Flanagan-Belytschko viscous form
=3: Flanagan-Belytschko viscous form with exact volume integration.
=4: Flanagan-Belytschko stiffness form
=5: full Flanagan-Belytschko stiffness form with exact volume integration.

IMGL

Initialize material for gravity loads.

IRR Irr

Moment of area along r-axis for Belytschko-Schwer beam.

IRULE GAUSS

Use gauss quadrature (default).

IRULE TRAPEZOIDAL

Use trapezoidal integration.

IRULE USER n

Use user defined integration rule n.

ISS Iss

Area moment of inertia along s-axis for
Belytschko-Schwer beam.  

**ITT** \( I_{tt} \)

Area moment of inertia along \( t \)-axis for Belytschko-Schwer beam.

**LTMN**

The local \( t \)-axis for thick shell elements of this material is the shortest direction through the brick.

**LTMX**

The local \( t \)-axis for thick shell elements of this material is the longest direction through the brick.

**MAT** \( m \)

Begin material definition \( m \). Each material definition is terminated by the ENDMAT command.

**MDMP** \( lc \) scale

Apply mass weighted damping to material mat. The magnitude is scale which is multiplied by load curve \( lc \). (LS-920 and later.)

**QUADRATURE** \( i \)

Select quadrature rule \( i \).

For beams the rules are:

\( i = 1 \): truss element  
\( i = 2 \): 2 X 2 Gauss (default)  
\( i = 3 \): 3 X 3 Gauss  
\( i = 4 \): 3 X 3 Lobatto  
\( i = 5 \): 4 X 4 Gauss

For four node shells the rules are:

\( i = 1 \): membrane element  
\( i = 2 \): 2 point Gauss (default)  
\( i = 3 \): 3 point Gauss  
\( i = 4 \): 4 point Gauss  
\( i = 5 \): 5 point Gauss

For eight node thick shells the rules are:

\( i = 1 \): membrane element  
\( i = 2 \): 2 point Gauss (default)  
\( i = 3 \): 3 point Gauss  
\( i = 4 \): 4 point Gauss  
\( i = 5 \): 5 point Gauss

**RAYD** \( b \)

Rayleigh (stiffness proportional) damping coefficient. (LS-920 and later.)

**REPOSITION**

Reposition deformable materials which are positioned relative to CAL3D/MADYMO3D bodies at initialization time. (LS-920 and later)

**Options:**

**ELLIPSE** \( n \)

Slave to MADYMO3D ellipse \( n \).
PLANE n  Slave to MADYMO3D plane n.

SEGMENT n  Slave to CAL3D segment n.

SYSTEM n  Slave to MADYMO3D system n.

RO m  Density (required - no default).

SAREA a  Shear area for Belytschko-Schwer beam.

SFORM s  Shell formulation type
          s = "HUGH": Hughes-Liu.
          s = "BELY": Belytschko-Lin-Tsay
          s = "BCZ": BCIZ triangular shell
          s = "C0T": $C_0$ triangular shell.
          s = "MEMB": B-L-T membrane.
          s = "SRHL": S/R Hughes-Liu
          s = "CRHL": Corotational Hughes-Liu.
          s = "YASE": Engelmann-Whirley's
           "YASE" shell. (Not recommended.)

SHELL  This material is defined for four node shell elements only.

SLOC s  Factor specifying offset of the local s axis.
          =-1: reference surface is at bottom plane of shell.
          =0: reference surface is at center plane of shell
          =1: reference surface is at upper plane of shell

STHICK thick  The default thickness along the element local s axis is thick (beams and shell).

TLOC t  Factor specifying offset of the local t axis.
          =-1: reference surface is at bottom plane of shell.
          =0: reference surface is at center plane of shell.
          =1: reference surface is at upper plane of shell.

TSHELL  This material is defined for thick (8-node solid) shell elements only.

TTHICK thick  The default thickness along the element local t axis is thick (beams only).

TYPE n  The current material is of type n.
Material Type 1 (Elastic)

Default heading: Material Type #1 (Elastic)

Input any two of the following.

- **BULK** \( K \)  \hspace{2cm} \text{Bulk modulus.}
- **E** \( E \)  \hspace{2cm} \text{Young's modulus.}
- **G** \( G \)  \hspace{2cm} \text{Shear modulus.}
- **PR** \( \nu \)  \hspace{2cm} \text{Poisson's ratio.}

Material Type 2 (Orthotropic Elastic)

- **EA** \( E_a \)  \hspace{2cm} \text{See constitutive matrix below.}
- **EB** \( E_b \)
- **EC** \( E_c \)
- **PRBA** \( \nu_{ba} \)
- **PRCA** \( \nu_{ca} \)
- **PRCB** \( \nu_{cb} \)
- **GAB** \( G_{ab} \)
- **GBC** \( G_{bc} \)
- **GCA** \( G_{ca} \)
- **AOPT** \( \text{aopt} \) \hspace{2cm} \text{Material axes option (Figure 20-1).}
  
  \( =0.0 \): locally orthotropic with materials axes by element nodes \( n_1, n_2, \) and \( n_4 \), (see Figure 20-1).
  
  \( =1.0 \): locally orthotropic with materials axes by a point in space and global location of element center.
  
  \( =2.0 \): globally orthotropic with materials axes determined by vectors defined below.
  
  \( =3.0 \): SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

- **XP** \( x_p \)  \hspace{2cm} \text{Define for AOPT = 1.}
- **YP** \( y_p \)  \hspace{2cm} \text{Define for AOPT = 1.}
The material law that relates stresses to strains is defined as:

\[ C = \mathbf{T}^T \mathbf{C}_L \mathbf{T}, \]

where \( \mathbf{T} \) is a transformation matrix, and \( \mathbf{C}_L \) is the constitutive matrix defined in terms of the material constants of the orthogonal material axes, \( a, b, \) and \( c \). The inverse of \( \mathbf{C}_L \) is defined as

\[
\mathbf{C}_L^{-1} = \begin{bmatrix}
\frac{1}{E_a} & \frac{v_{ba}}{E_a} & \frac{v_{ca}}{E_a} & 0 & 0 & 0 \\
\frac{v_{ab}}{E_b} & \frac{1}{E_b} & \frac{v_{cb}}{E_b} & 0 & 0 & 0 \\
\frac{v_{ac}}{E_c} & \frac{v_{bc}}{E_c} & \frac{1}{E_c} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{G_{ab}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{G_{bc}} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{G_{ca}}
\end{bmatrix}
\]

Note that \( \frac{v_{ab}}{E_a} = \frac{v_{ba}}{E_b}, \frac{v_{ca}}{E_c} = \frac{v_{ac}}{E_a}, \frac{v_{bc}}{E_c} = \frac{v_{bc}}{E_b} \).
Figure 20-1. Options for determining principal materials axes: (a) AOPT = 0.0, (b) AOPT = 1.0, and (c) AOPT = 2.0, (d) AOPT = 3.0.
Material Type 3 (Kinematic/Isotropic Elastic/Plastic)

Default heading: Material Type #3 (Elastic-Plastic)

Input any two of the following:

- **BULK K**
  - Bulk modulus.
- **E E**
  - Young's modulus.
- **G G**
  - Shear modulus.
- **PR ν**
  - Poisson's ratio.

Additional Options:

- **SIGY σ_y**
  - Yield stress
- **ETAN E_t**
  - Hardening modulus
- **BETA β'**
  - Hardening parameter, $0 \leq β' \leq 1$
- **SC c**
  - Strain rate parameter, C
- **SP p**
  - Strain rate parameter, $p$

Strain rate is accounted for using the Cowper and Symonds model which scales the yield stress with the factor

$$1 + \left( \frac{\dot{\gamma}}{C} \right)^p$$

where $\dot{\gamma}$ is the strain rate.

Isotropic, kinematic, or a combination of isotropic and kinematic hardening may be specified by varying $β'$ between 0 and 1. For $β' = 0$ and 1, respectively kinematic and isotropic hardening are obtained as shown in Figure 20-2. Effective stress is defined in terms of the deviatoric stress tensor, $S_{ij}$ as:

$$\bar{\sigma} = \left( \frac{3}{2} S_{ij} S_{ij} \right)^{\frac{1}{2}}$$

where,

$$S_{ij} = \sigma_{ij} - \frac{1}{3} \sigma_{kk} \delta_{ij}$$

and effective plastic strain by:

$$\bar{\varepsilon}^p = \int_0^t \dot{\varepsilon}^p$$

where $t$ denotes time and
\[
\varepsilon_p = \left( \frac{2}{3} \varepsilon_{ij}^p \varepsilon_{ij}^p \right)^{\frac{1}{2}}
\]

For isotropic hardening (\(\beta' = 1\)) material model 12 requires less storage and is more efficient.

Figure 20-2. Elastic-plastic behavior with isotropic and kinematic hardening where \(l_0\) and \(l\) are undeformed and deformed length of uniaxial tension specimen.
Material Type 4 (Thermo-Elastic-Plastic)

Default heading: Material Type 4 (Thermo-Elastic-Plastic)

- **NPTS** \( n \)  
  Number of temperature values for which material constants are defined.

- **TEMP** \( T_1 \ T_2 \ldots T_n \)  
  Temperatures.

- **E** \( E_1 \ E_2 \ldots E_n \)  
  Young's moduli.

- **PR** \( \nu_1 \ \nu_2 \ldots \nu_n \)  
  Poisson's ratios.

- **ALPHA** \( \alpha_1 \ \alpha_2 \ldots \alpha_n \)  
  Coefficients of thermal expansion.

- **SIGY** \( \sigma_{y1} \ \sigma_{y2} \ldots \sigma_{yn} \)  
  Yield stresses.

- **ETAN** \( E_{t1} \ E_{t2} \ldots E_{tn} \)  
  Tangent moduli.

Material Type 5 (Soil and Crushable Foam)

Default heading: Material Type 5 (Soil and Crushable Foam)

Input any two of the following:

- **BULK** \( K \)  
  Bulk modulus.

- **E** \( E \)  
  Young's modulus.

- **G** \( G \)  
  Shear modulus.

- **PR** \( \nu \)  
  Poisson's ratio.

Additional Options:

- **AO** \( a_0 \)  
  Yield function constant

- **A1** \( a_1 \)  
  Yield function constant

- **A2** \( a_2 \)  
  Yield function constant

- **PC** \( P_c \)  
  Pressure cutoff for tensile fracture

- **UL** \( u_{opt} \)  
  Unloading option
  
  - =0: volumetric crushing
  - =1: loading and unloading are the same.
  - =2: hysteretic behavior.

- **ULD** \( d \)  
  Unloading distance for option 2 above.

- **NPTS** \( n \)  
  Number of points in volumetric strain versus pressure curve \((n \leq 10)\).

- **VS** \( \varepsilon_{v1} \ \varepsilon_{v2} \ldots \varepsilon_{vn} \)  
  Volumetric strain values

- **P** \( p_1 \ p_2 \ldots p_n \)  
  Pressures corresponding to volumetric strain values

The deviatoric, perfectly yield function, \( \phi \), is described in terms of the second invariant \( J_2 \).
\[ J_2 = \frac{1}{2} s_{ij} s_{ij} \]

Pressure, \( p \), and constants \( a_0, a_1, \) and \( a_2 \) as:

\[ \phi = J_2 - \left[ a_0 + a_1 p + a_2 p^2 \right]. \]

On the yield surface, \( J_2 = \frac{1}{3} \sigma_y^2 \), where \( \sigma_y \) is the yield stress, i.e.,

\[ \sigma_y = \left[ 3 \left( a_0 + a_1 p + a_2 p^2 \right) \right]^{1/2} \]

For elastic-perfectly plastic behavior \( a_1 = a_2 = 0 \), and \( (3a_0)^{1/2} \) defines the yield strength.

The volumetric strain is given by the natural logarithm of the relative volume \( V \). If the pressure drops below the cutoff value, \( PC \), then it is reset to that value.

![Volumetric strain versus pressure curve for soil and crushable foam model.](image_url)

Figure 20-3. Volumetric strain versus pressure curve for soil and crushable foam model.
Material Type 6 (Viscoelastic)

- **G** $G_0$  
  Short term shear modulus.

- **GI** $G_\text{-}$  
  Long term shear modulus.

- **K** $K$  
  Bulk modulus.

- **BETA** $\beta$  
  Decay constant.

The shear relaxation behavior is described by:

$$G(t) = G_0 + (G_0 - G_\text{-}) e^{-\beta t}$$

A Jaumann rate formulation is used:

$$\sigma'_{ij} = 2 \int_0^t G(t-\tau) \mathcal{D}_{ij}(\tau) d\tau$$

where the prime denotes the deviatoric part of the stress rate, $\sigma'_{ij}$, and the strain rate $\mathcal{D}_{ij}$.

Material Type 7 (Blatz-Ko Rubber)

Default heading: Material Type #7 (Rubber)

- **G** $G$  
  Shear modulus.

The second Piola-Kirchhoff stress is computed as

$$S_{ij} = \mu \left(\frac{1}{V} C_{ij} - V^{-1} \nabla \cdot \mathbf{u} \right)$$

where $V$ is the relative volume, $C_{ij}$ is the right Cauchy-Green strain tensor, and $n$ is the Poisson's ratio which is set to .463 internally. This stress measure is transformed to the Cauchy stress, $s_{ij}$, according to the relationship

$$s_{ij} = V^{-1} F_{ik} F_{jl} S_{lk}$$

where $F_{ij}$ is the deformation gradient tensor.
Material Type 8 (High Explosive Burn)

Default heading: Material Type #8 (High Explosive Burn)

\[ \textbf{D} \quad \text{Detonation velocity.} \]
\[ \textbf{PCJ} \quad \text{PCJ Chapman-Jouget pressure.} \]

This material model requires an equation-of-state.

Material Type 9 (Null Material)

Default heading: Material Type #9 (Null Material)

\[ \textbf{PC} \quad \text{Pressure cutoff.} \]
\[ \textbf{MU} \quad \text{\( \mu \).} \]

The null material must be used with an equation-of-state. Pressure cutoff is negative in tension. A viscous stress of the form

\[ \sigma_{ij} = \mu \dot{\varepsilon}_{ij} \]

is computed for nonzero \( m \) where \( \dot{\varepsilon}_{ij} \) is the deviatoric strain rate.

Material Type 10 (Isotropic-Elastic-Plastic-Hydrodynamic)

Default heading: Material Type #10 (Isotropic-Elastic-Plastic-Hydrodynamic)

\[ \textbf{G} \quad \text{Shear modulus} \]
\[ \textbf{SIGY} \quad \sigma_y \quad \text{Yield strength} \]
\[ \textbf{EH} \quad E_h \quad \text{Plastic hardening modulus} \]
\[ \textbf{PC} \quad p_c \quad \text{or} \quad -\sigma_f \quad \text{Pressure cutoff} \]
\[ = 0: \quad \text{cutoff of} \quad -\sigma_f \quad \text{is assumed} \]
\[ \textbf{A1} \quad a_1 \quad \text{Yield function constant} \]
\[ \textbf{A2} \quad a_2 \quad \text{Yield function constant} \]
\[ \textbf{NPTS} \quad n \quad \text{Number of points in yield stress-effective plastic strain curve or yield stress-pressure curve.} \]
\[ \textbf{ES} \quad \sigma_{y1} \quad \sigma_{y2} \ldots \sigma_{yn} \quad \text{Yield stress} \]
If the yield stress-plastic strain curve is not defined and if $a_1 = a_2 = 0$, the bilinear stress-strain curve shown in Figure 20-2 is obtained with $b = 1$. The yield strength is calculated as

$$\sigma_y = \sigma_0 + E_h \varepsilon_p$$

where $p$ is the pressure. The quantity $E_h$ is the plastic hardening modulus defined in terms of Young’s modulus, $E$, and the tangent modulus, $E_t$, as follows

$$E_h = \frac{E_t}{E - E_t}$$

If the yield stress-plastic strain (pressure) curve is defined, a curve like that shown in Figure 20-4 may be defined. In this latter case, the yield stress and plastic hardening modulus, $a_1$ and $a_2$ are ignored.

Effective stress is defined in terms of the deviatoric stress tensor, $s_{ij}$, as:

$$\bar{\sigma} = \left( \frac{3}{2} s_{ij} s_{ij} \right)^{\frac{1}{2}}$$

and effective plastic strain by:

$$\bar{\varepsilon}^p = \int_0^t \left( \frac{2}{3} D_{ij}^P D_{ij}^P \right)^{\frac{1}{2}} dt$$

where $t$ denotes time and $D_{ij}^P$ is the plastic component of the rate of deformation tensor. Yield stress may be defined as a function of plastic strain or pressure but not both.
Piecewise linear curve defining the yield stress versus effective plastic strain. A nonzero yield stress is defined when the plastic strain is zero.

Figure 20-4. Effective stress versus effective plastic strain curve.

Material Type 11 (Temperature Dependent Elastic-Plastic Hydrodynamic)

Default heading: Material Type #11 (Temperature Dependent Elastic-Plastic Hydrodynamic)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>G G₀</td>
<td>Shear modulus.</td>
</tr>
<tr>
<td>SIGO σ₀</td>
<td>See equations below.</td>
</tr>
<tr>
<td>BETA β</td>
<td></td>
</tr>
<tr>
<td>N n</td>
<td></td>
</tr>
<tr>
<td>GAMA γᵢ</td>
<td></td>
</tr>
<tr>
<td>SIGM σₘ</td>
<td></td>
</tr>
<tr>
<td>B b</td>
<td></td>
</tr>
<tr>
<td>BP b'</td>
<td></td>
</tr>
<tr>
<td>H h</td>
<td></td>
</tr>
<tr>
<td>F f</td>
<td></td>
</tr>
<tr>
<td>A a</td>
<td></td>
</tr>
<tr>
<td>TO Tₘ₀</td>
<td></td>
</tr>
<tr>
<td>GAMO g₀</td>
<td></td>
</tr>
<tr>
<td>SA a</td>
<td></td>
</tr>
</tbody>
</table>
PC pmin or -s
ECO EC0
EC1 EC1
EC2 EC2
EC3 EC3
EC4 EC4
EC5 EC5
EC6 EC6
EC7 EC7
EC8 EC8
EC9 EC9

Cold compression energy coefficients (optional)

If cold compression energy coefficients are not input, then LS-DYNA3D will calculate them based on the equation-of-state.

SPALL type

Spall type

-0: default set to "2.0"
-1: \( p \geq p_{min} \)
-2: if \( s_{max} \geq s_f \) element spalls and tension \( p < 0 \), is never allowed; \( s_{max} \) = maximum principal stress.
-3: if \( p < p_{min} \) element spalls and tension \( p < 0 \), is never allowed.
-4: failure strain

Users who have an interest in this mode are encouraged to study the paper by Steinberg and Guinan [9] which provides the theoretical basis. Another useful reference is the KOVEC user's manual [10].

In terms of the foregoing input parameters, we define the shear modulus, \( G \), before the material melts as:

\[
G = G_0 \left[ 1 + b p V^{3/2} \left( \frac{E_i - E_c}{3 R'} - 300 \right) \right]^{- \frac{f E_i}{E_m - E_i}}
\]

where \( p \) is the pressure, \( V \) is the relative volume, \( E_c \) is the cold compression energy:

\[
\bar{\varepsilon}^p = \int_0^t \left( \frac{2}{3} D_{ij}^p D_{ij}^p \right)^{1/2} dt
\]

\[ x = 1 - V \]

and \( E_m \) is the melting energy:

\[
E_m (x) = E_c (x) + 3R'T_m (x)
\]
which is in terms of the melting temperature $T_m(x)$:

$$T_m(x) = \frac{T_{m0} \exp{(2ax)}}{\sqrt{2(y_0-a^{-\frac{1}{3}})}}$$

and the melting temperature at $r = r_0$, $T_{m0}$.

In the above equation, $R'$ is defined by

$$R' = \frac{R}{A}$$

where $R$ is the gas constant and $A$ is the atomic weight. If $R'$ is not defined, LS-DYNA3D computes it with $R$ in the cm-gram-microsecond system of units.

The yield strength $\sigma_y$ is given by:

$$\sigma_y = \sigma_0' \left[ 1 + b'p \sqrt[3]{h} - h \left( \frac{E_i - E_c}{3R'} - 300 \right) \right] e^{-\frac{fE_i}{E_m - E_i}}$$

if $E_m$ exceeds $E_i$. Here, $\sigma_0'$ is given by:

$$\sigma_0' = \sigma_0' \left[ 1 + \beta \gamma_i + e^{-p} \right]$$

where $\gamma_i$ is the initial plastic strain. Whenever $\sigma_0'$ exceeds $\sigma_m$, $\sigma_0'$ is set equal to $\sigma_m$. After the material melts, $\sigma_y$ and $G$ are set to zero.

If the coefficients $EC0,...,EC9$ are not defined above, LS-DYNA3D will fit the cold compression energy to the ten term polynomial expansion:

$$E_c = \sum_{i=0}^{9} Ec_i \eta^i$$

where $Ec_i$ is the $i$th coefficient and $h=r/r_0-1$. A least square method is used to perform the fit.

**Material Type 12 (Isotropic-Elastic-Plastic)**

Default heading: Material Type #12 (Isotropic-Elastic-Plastic)

Input any two of the following:

- **BULK K** Bulk modulus.
- **E E** Young's modulus.
- **G G** Shear modulus.
Poisson's ratio.

Addition Options:

- **SIGY** $\sigma_y$
  - Yield strength.
- **EH** $E_h$
  - Hardening modulus.

Pressure is integrated in time

$$\dot{p} = -K \frac{\gamma}{V}$$

where $V$ is the relative volume. This model is recommended for brick elements but not for shell elements since it is not too accurate.

**Material Type 13 (Elastic-Plastic with Failure Model)**

Input any two of the following:

- **BULK** $K$
  - Bulk modulus.
- **E** $E$
  - Young's modulus.
- **G** $G$
  - Shear modulus.
- **PR** $\nu$
  - Poisson's ratio.

Additioan Options:

- **SIGY** $\sigma_y$
  - Yield strength.
- **EH** $E_h$
  - Hardening modulus.
- **FS** $\epsilon_f$
  - Failure strain.
- **FP** $p_f$
  - Failure pressure ($\leq 0.0$).

When the effective plastic strain reaches the failure strain or when the pressure reaches the failure pressure, the material loses its ability to carry tension and the deviatoric stresses are set to zero, i.e., the material behaves like a fluid.
Material Type 14 (Soil and Crushable Foam with Failure Model)

The input for this model is the same as for material type 5; however, when the pressure reaches the failure pressure, the element loses its ability to carry tension.

Material Type 15 (Johnson/Cook Plasticity Model)

\( G \)  
Shear modulus.

\( A \)  
See equation (1).

\( B \)  
See equation (1).

\( N \)  
See equation (1).

\( R \)  
See equation (1).

\( M \)  
See equation (1).

\( TM \)  
Melt temperature

\( TO \)  
Room temperature.

\( EPSO \)  
Effective plastic strain rate.

\( HCP \)  
Specific heat.

\( PC \)  
Pressure cutoff (pc < 0.0).

\( D1 \)  
See equation (2).

\( D2 \)  
See equation (2).

\( D3 \)  
See equation (2).

\( D4 \)  
See equation (2).

\( D5 \)  
See equation (2).

\( IT \)  
Iteration options:

  \( =0 \): no iterations.

  \( =1 \): LS-DYNA3D iterates to determine a more accurate point on the stress-strain curve.

The Johnson/Cook model is described in reference [11]. This model includes strain rate hardening, thermal softening, and has a complex damage model. The equations describing the flow stress vs. effective plastic strain and failure strain are as follows:

\[
\sigma_y = \left( A + B \varepsilon^p \right) \left( 1 + C \ln \varepsilon^p \right) \left( 1 - T^m \right)
\]

where \( A, B, C, n, \) and \( m \) are input constants,

\( \varepsilon^p \) effective plastic strain
\[ \dot{\varepsilon}^p = \frac{\varepsilon^p}{\varepsilon^0} \] effective plastic strain rate for \( \varepsilon^0 = 1 \text{ s}^{-1} \)

\[ T^* = \frac{T_r}{T_m} \] homologous temperature

Constants for a variety of materials are also provided in [11].

Due to the nonlinearity in the dependence of flow stress on plastic strain, an accurate value of the flow stress requires iteration for the increment in plastic strain. However, by using a Taylor series expansion with linearization about the current time, we can solve for \( s_y \) with sufficient accuracy to avoid iteration.

The strain at fracture is given by

\[ \varepsilon^f = \left[ D_1 + D_2 \exp D_3 \sigma^* \right] \left[ l + D_4 \ln \varepsilon^* \right] \left[ l + D_5 T^* \right] \]

where \( s^* \) is the ratio of pressure divided by effective stress:

\[ \sigma^* = \frac{p}{\sigma_{\text{eff}}} \]

Fracture occurs when the damage parameter

\[ D = \sum \frac{\Delta \varepsilon^p}{\varepsilon^f} \]

reaches the value of 1.

**Material Type 16 (Pseudo Tensor Geological Model)**

Default heading: Material Type #16 (Pseudo Tensor Geological Model)

- **G G** Shear modulus (constant Shear modulus model).
- **PR v** Poisson's ratio (constant Poisson's ratio model).
- **SIGF sigf** Tensile cutoff. (Maximum principal stress for failure.)
- **A0 a_0** Cohesion.
- **A1 a_1** Yield function constant.
- **A2 a_2** Yield function constant.
- **A0F a_{0f}** Cohesion for failed material.
A1F $a_{1f}$  Pressure hardening coefficient for failed material.

B1 $b_1$  Damage scaling factor.

PER $p$  Percent reinforcement.

ER $E_r$  Elastic modulus for reinforcement.

PR $\nu_r$  Poisson's ratio for reinforcement.

SIGY $\sigma_y$  Initial yield strength.

ETAN $E_t$  Tangent modulus.

LCP $l_{c1}$  Load curve giving rate sensitivity for principal material.

LCR $l_{c2}$  Load curve giving rate sensitivity for reinforcement.

NPTS $n$  Number of points in yield stress-effective plastic strain curve or yield stress-pressure curve; $(n \leq 16)$.

ES $\sigma_1 \sigma_2 ... \sigma_n$  Yield stress.

EPS $\varepsilon_{p1} \varepsilon_{p2} ... \varepsilon_{p3}$  Effective plastic strain.

P $p_1 p_2 ... p_n$  Pressure.

See the LS-DYNA3D manual for a description of this model.

**Material Type 17 (Elastic Plastic with Failure Model)**

Input any two of the following:

BULK $K$  Bulk modulus.

E $E$  Young's modulus.

G $G$  Shear modulus.

PR $\nu$  Poisson's ratio.

Additional Options:

SIGY $\sigma_y$  Yield strength.

EH $E_h$  Plastic hardening modulus.

FS $\varepsilon_f$  Failure stress.

Model 17 can fail in two ways. In hydrostatic tension, the element will fail when the failure stress is exceeded. The element will then allow hydrostatic compressive loads only.
If the effective stress exceeds the failure stress, the element will form a fracture plane and retain part of its strength.

**Material Type 18 (Power Law Plasticity)**

Input any two of the following:

- **BULK K** Bulk modulus.
- **E E** Young's modulus.
- **G G** Shear modulus.
- **PR ν** Poisson's ratio.

Additional **Options**:

- **K k** See equation below
- **M m** See equation below.
- **SC c** Strain rate parameter, $C$.
- **SP p** Strain rate parameter, $p$.

***missing***

Elastoplastic behavior with isotropic hardening is provided by this model. The yield stress, $\sigma_y$, is a function of plastic strain and obeys the equation:

$$\sigma_y = k (\varepsilon_e + \varepsilon_p^n)$$

$\varepsilon_e$ is the elastic strain to yield and where $\varepsilon_p$ is the effective plastic strain. The strain-rate parameters are defined in material type 3.

**Material Type 19 (Strain Rate Sensitive Plasticity)**

Input any two of the following:

- **BULK K** Bulk modulus.
- **E E** Young's modulus.
- **G G** Shear modulus.
- **PR ν** Poisson's ratio.

Additional **Options**:
In this model, a load curve is used to describe the yield strength, \( s_0 \), as a function of effective strain rate,

\[
\dot{\varepsilon}^e = \left( \frac{2}{3} \dot{\varepsilon}^{\prime}_{ij} \dot{\varepsilon}^{\prime}_{ij} \right)^{\frac{1}{2}}
\]

and the prime denotes the deviatoric component. The yield stress is defined as

\[
\sigma_y = \sigma_0 \left( \dot{\varepsilon}^e \right) + E_h \varepsilon^p
\]

where \( \varepsilon^p \) is the effective plastic strain and \( E_h \) is given by

\[
E_h = \frac{E - E_t}{E - E_t}
\]

**Material Type 20 (Rigid Body)**

All elements with the same material number become a single rigid body if the material is type 20 whether the elements are connected or not. Density and two independent material strength constants are required to establish penalties for contact surfaces and joints.

Input any two of the following:

- **BULK** \( K \)  
  Bulk modulus.
- **E** \( E \)  
  Young's modulus.
- **G** \( G \)  
  Shear modulus.
- **PR** \( \nu \)  
  Poisson's ratio.

Additional **Options**:

- **DEFG**  
  The rigid body is defined in the global system used by CAL3D/MADYMO3D. (LS-920)
**LS-DYNA3D COMMANDS AND MATERIALS**

**DEFL**
The rigid body is defined in the local system used by CAL3D/MADYMO3D. (LS-920)

**ELLIPSE m**
The rigid body is slaved to MADYMO3D ellipsoid m. (LS-920)

**MESH**
Generate a mesh for the CAL3D/MADYMO3D coupled rigid body. (LS-920)

**MSLAV m**
The rigid body is slaved to CAL3D rigid body number m. (LS-920)

**PLANE m**
The rigid body is slaved to MADYMO3D plane m. (LS-920)

**SYSTEM n**
The rigid body is slaved to MADYMO3D system n. (LS-920)

**VDA**
The rigid body is characterized by a VDA surface geometry. (LS-920)

**AVEC a_x a_y a_z**
Define the vector a for the rigid body local system.

**VVEC v_x v_y v_z**
Define the vector v for the rigid body local system.

---

**Material Type 21 (Thermal Orthotropic)**

**EA E_a**

See constitutive matrix for material 2.

**EB E_b**

**EC E_c**

**PRBA \( \nu_{ba} \)**

**PRCA \( \nu_{ca} \)**

**PRCB \( \nu_{cb} \)**

**GAB G_{ab}**

**GBC G_{bc}**

**GCA G_{ca}**

**AA \( \alpha_a \)**

\( \alpha_a \)

**AB \( \alpha_b \)**

\( \alpha_b \)

**AC \( \alpha_c \)**

\( \alpha_c \)

**AOPT aopt**

Material axes option (Figure 20-1).

=0.0: locally orthotropic with materials axes determined by element nodes \( n_1, n_2, \) and \( n_4 \), (see Figure 20-1).

=1.0: locally orthotropic with materials axes...
determined by a point in space and global location of element center.

=2.0: globally orthotropic with materials axes determined by vectors defined below.

=3.0: SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

\[ \begin{align*}
\text{XP} & \quad \text{Define for } AOPT = 1. \\
\text{YP} & \quad \text{Define for } AOPT = 1. \\
\text{ZP} & \quad \text{Define for } AOPT = 1. \\
\text{A1} & \quad \text{Define for } AOPT = 2. \\
\text{A2} & \quad \text{Define for } AOPT = 2. \\
\text{A3} & \quad \text{Define for } AOPT = 2. \\
\text{D1} & \quad \text{Define for } AOPT = 2. \\
\text{D2} & \quad \text{Define for } AOPT = 2. \\
\text{D3} & \quad \text{Define for } AOPT = 2. \\
\text{V1} & \quad \text{Define for } AOPT = 3. \\
\text{V2} & \quad \text{Define for } AOPT = 3. \\
\text{V3} & \quad \text{Define for } AOPT = 3.
\end{align*} \]

**Material Type 22 (Orthotropic Damage Model)**

\[ \begin{align*}
\text{EA} & \quad E_a \\
\text{EB} & \quad E_b \\
\text{EC} & \quad E_c \\
\text{PRBA} & \quad \nu_{ba} \\
\text{PRCA} & \quad \nu_{ca} \\
\text{PRCB} & \quad \nu_{cb} \\
\text{GAB} & \quad G_{ab} \\
\text{GBC} & \quad G_{bc} \\
\text{GCA} & \quad G_{ca} \\
\text{K} & \quad K_f \\
\text{SC} & \quad S_c \\
\text{XT} & \quad x_t \\
\text{YT} & \quad y_t \\
\text{YC} & \quad y_c \\
\text{ALPH} & \quad \alpha
\end{align*} \]

See constitutive matrix below.

Bulk modulus of failed material.

Shear strength, \( ab \) plane.

Longitudinal tensile strength, \( a \)-axis.

Transverse tensile strength, \( b \)-axis.

Transverse compressive strength.

Non-linear shear stress parameter.
Material axes option (Figure 20-1).

=0.0: locally orthotropic with material axes determined by element nodes \( n_1, n_2, \) and \( n_4 \), (see Figure 20-1).

=1.0: locally orthotropic with material axes determined by a point in space and global location of element center.

=2.0: globally orthotropic with material axes determined by vectors defined below.

=3.0: SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

\[
\begin{align*}
\text{XP } x_p & \quad \text{Define for AOPT } = 1. \\
\text{YP } y_p & \quad \text{Define for AOPT } = 1. \\
\text{ZP } z_p & \quad \text{Define for AOPT } = 1. \\
\text{A1 } a_1 & \quad \text{Define for AOPT } = 2. \\
\text{A2 } a_2 & \quad \text{Define for AOPT } = 2. \\
\text{A3 } a_3 & \quad \text{Define for AOPT } = 2. \\
\text{D1 } d_1 & \quad \text{Define for AOPT } = 2. \\
\text{D2 } d_2 & \quad \text{Define for AOPT } = 2. \\
\text{D3 } d_3 & \quad \text{Define for AOPT } = 2. \\
\text{V1 } v_1 & \quad \text{Define for AOPT } = 3. \\
\text{V2 } v_2 & \quad \text{Define for AOPT } = 3. \\
\text{V3 } v_3 & \quad \text{Define for AOPT } = 3.
\end{align*}
\]

**Material Type 23 (Thermal Orthotropic with Curves)**

\[
\begin{align*}
\text{NPTS } \text{npts} & \quad \text{Number of points. (1<NPTS<50).} \\
\text{EA } (E_a)_1 \ldots (E_a)_n & \\
\text{EB } (E_b)_1 \ldots (E_b)_n & \\
\text{EC } (E_c)_1 \ldots (E_c)_n & \\
\text{PRBA } (\nu_{ba})_1 \ldots (\nu_{ba})_n & \\
\text{PRCA } (\nu_{ca})_1 \ldots (\nu_{ca})_n & \\
\text{PRCB } (\nu_{cb})_1 \ldots (\nu_{cb})_n & \\
\text{AA } (\alpha_a)_1 \ldots (\alpha_a)_n & \\
\text{AB } (\alpha_b)_1 \ldots (\alpha_b)_n & \\
\text{AC } (\alpha_c)_1 \ldots (\alpha_c)_n &
\end{align*}
\]
GAB \((G_{ab})_1 \ldots (G_{ab})_n\)
GBC \((G_{bc})_1 \ldots (G_{bc})_n\)
GCA \((G_{ca})_1 \ldots (G_{ca})_n\)

**AOPT** \(\text{aopt}\)

Material axes option (Figure 20-1).

- \(0.0\): locally orthotropic with materials axes determined by element nodes \(n_1, n_2,\) and \(n_4\), (see Figure 20-1).

- \(1.0\): locally orthotropic with materials axes determined by a point in space and global location of element center.

- \(2.0\): globally orthotropic with materials axes determined by vectors defined below.

- \(3.0\): SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

**XP** \(x_p\)
Define for **AOPT** = 1.

**YP** \(y_p\)
Define for **AOPT** = 1.

**ZP** \(z_p\)
Define for **AOPT** = 1.

**A1** \(a_1\)
Define for **AOPT** = 2.

**A2** \(a_2\)
Define for **AOPT** = 2.

**A3** \(a_3\)
Define for **AOPT** = 2.

**D1** \(d_1\)
Define for **AOPT** = 2.

**D2** \(d_2\)
Define for **AOPT** = 2.

**D3** \(d_3\)
Define for **AOPT** = 2.

**V1** \(v_1\)
Define for **AOPT** = 3.

**V2** \(v_2\)
Define for **AOPT** = 3.

**V3** \(v_3\)
Define for **AOPT** = 3.

**Material Type 24 (Elastic Plastic with Failure)**

Input any two of the following:

**BULK** \(K\)  
Bulk modulus.

**E** \(E\)  
Young's modulus.

**G** \(G\)  
Shear modulus.

**PR** \(\nu\)  
Poisson's ratio.
Additional Options:

**SIGY** \( s_y \)  
Yield strength.

**ETAN** \( E_t \)  
Hardening modulus

**NPTS** \( n \)  
Number of points in effective stress-effective plastic strain curve. Note that the first point on this curve must be \( e_1 = 0.0 \) and \( s_1 = \) yield stress.

**ES** \( \sigma_1 \sigma_2 \ldots \sigma_n \)  
Effective stress.

**EPS** \( \varepsilon_1 \varepsilon_2 \ldots \varepsilon_n \)  
Effective plastic strain.

**TDEL** \( \Delta t \)  
Minimum time step. (This is for automatic element deletion).

**FAIL** \( \varepsilon_f \)  
Failure strain

**CSR** \( lc \)  
Load curve which describes strain-rate effects.

Strain rate is accounted for using the Cowper and Symonds model which scales the yield stress with the factor

\[
1 + \left( \frac{\dot{\varepsilon}}{C} \right)^\alpha
\]

where \( \dot{\varepsilon} \) is the strain rate. For complete generality a load curve may be input instead. This latter option is quite expensive.

A curve similar to that shown in Figure 3.4 is expected. A load curve may be used with an arbitrary number of points if eight is not sufficient. The cost is roughly the same for either approach.

---

**Material Type 25 (Inviscid Two Invariant Geologic Cap Model)**

**G** \( G \)  
Shear Modulus.

**K** \( K \)  
Bulk Modulus.

**ALPHA** \( \alpha \)  
\( \alpha \).

**BETA** \( \beta \)  
\( \beta \).

**GAMMA** \( \gamma \)  
\( \gamma \).

**THETA** \( \theta \)  
\( \theta \).

**R** \( R \)  
\( R \).

**D** \( D \)  
\( D \).

**X0** \( X_0 \)  
\( X_0 \).

**CC** \( C \)  
\( C \).

**T** \( T \)  
Tension cutoff.

**NPLOT** \( nplot \)  
Save the following variable for plotting in TAUROS:

\[
=1: \ k
=2: \ X
\]
=3:  $e_1$
=4:  $J_1$
=5:  $(J_2)^{1/2}$
=6:  $(J_2|^L)^{1/2}$
=7:  $(J_2|^T)^{1/2}$
=8:  MTYPE
=9:  number of iterations

**LTYPE ltype**

Variable ltype.
- 1: soil/concrete (cap contracts)
- 2: rock (cap doesn't contract)

For details of this model, please refer to the LS-DYNA3D User's Manual.

### Material Type 26 ( Metallic Honeycomb )

Model 26 provides a method for modeling the crushing of an anisotropic material which eventually compresses to a solid, isotropic mass. This model is valid for brick elements only. For more details, see the LS-DYNA3D manual.

- **E E** Young's modulus for fully compressed state.
- **PR ν** Poisson's ratio for fully compressed state.
- **SIGY σ_y** Yield stress for fully compressed state.
- **VF V_f** Relative volume at which the material is fully compacted.
- **LCA lca** Load curve for sigma-$aa$ versus either relative volume or volumetric strain.
- **LCB lcb** Load curve for sigma-$bb$ versus either relative volume or volumetric strain.
- **LCC lcc** Load curve for sigma-$cc$ versus either relative volume or volumetric strain.
- **LCS lcs** Load curve for shear stress versus either relative volume or volumetric strain.
- **EAAU E_{aa}$** Elastic modulus $E_{aa}$ in uncompressed configuration.
- **EBBU E_{bb}$** Elastic modulus $E_{bb}$ in uncompressed configuration.
- **ECCU E_{cc}$** Elastic modulus $E_{cc}$ in uncompressed configuration.
- **GABU G_{ab}$** Elastic shear modulus $G_{ab}$ in uncompressed configuration.
- **GBCU G_{bc}$** Elastic shear modulus $G_{bc}$ in uncompressed configuration.
- **GCAU G_{ca}$** Elastic shear modulus $G_{ca}$ in uncompressed configuration.
- **LCAB lcab** Load curve number for $s_{ab}$ versus either relative
**LCBC lcbc**  
Load curve number for \( s_{bc} \) versus either relative volume or volumetric strain. (default: \( lcab=lcs \))

**LCCA lcca**  
Load curve number for \( s_{ca} \) versus either relative volume or volumetric strain. (default: \( lcca=lcs \))

**AOPT aopt**  
Material axes option (Figure 20-1).  
=0.0: locally orthotropic with materials axes by determined element nodes \( n_1, n_2, \) and \( n_4 \), (see Figure 20-1).  

=1.0: locally orthotropic with materials axes determined by a point in space and global location of element center.  

=2.0: globally orthotropic with materials axes determined by vectors defined below.

**XP xp**  
Define for AOPT = 1.

**YP yp**  
Define for AOPT = 1.

**ZP zp**  
Define for AOPT = 1.

**A1 a1**  
Define for AOPT = 2.

**A2 a2**  
Define for AOPT = 2.

**A3 a3**  
Define for AOPT = 2.

**D1 d1**  
Define for AOPT = 2.

**D2 d2**  
Define for AOPT = 2.

**D3 d3**  
Define for AOPT = 2.

**Material Type 27 (Compressible Mooney-Rivlin Rubber)**

This material model provides an alternative to the Blatz-Ko rubber model. The implementation is due to Maker [12].

**A A**  
Constant A.

**B B**  
Constant B.

**PR ν**  
Poisson's ratio.

The strain energy density function is defined as:

\[
W = A(I-3)+B(II-3)+C(III^{-2}-1)+D(III-1)^2
\]

where
$C = 0.5A + B.$

\[ D = \frac{A(5\nu - 2) + B(11\nu - 5)}{2(1 - 2\nu)} \]

\(\nu\) = Poisson's ratio.

\(2(A + B) = G\) = shear modulus of linear elasticity.

I, II, III are invariants of the right Cauchy-Green Tensor \(C\).

**Material Type 28 (Resultant Plasticity)**

Default heading: Material Type #28 (Resultant Plasticity)

This model is available for the Belytschko-Schwer beam and the Belytschko-Tsay shell and is still under development. For beams the treatment is elastic-perfectly plastic, but for shell elements isotropic hardening is approximately modeled.

Input any two of the following:

- **BULK** \(K\) = Bulk modulus.
- **E** \(E\) = Young's modulus.
- **G** \(G\) = Shear modulus.
- **PR** \(\nu\) = Poisson's ratio.

Additional *Options*:

- **SIGY** \(\sigma_y\) = Yield strength.
- **ET** \(E_t\) = Hardening modulus (shells only).

**Material Type 29 (Force Limited Resultant Formulation)**

This model is valid for the Belytschko beam element only. Experimentally obtained force-deflection curves may be used to model buckling and plastic behavior. See the LS-DYNA3D manual for more details.

Input any two of the following:
BULK $K$  
Bulk modulus.

$E$  $E$  
Young's modulus.

$G$  $G$  
Shear modulus.

$PR$  $v$  
Poisson's ratio.

Additional Options:

$R$  $R_1$  $lc_1$ ...  $R_n$  $lc_n$ ;  
Applied moments for force deflection curves.  
$n \leq 8$.

$LPS1$  $lps1$  
Load curve for plastic moment versus rotation at node 1 in $s$-direction.

$SFS1$  $sfs1$  
Scale factor for plastic moment versus rotation curve at node 1 in $s$-direction.

$LPS2$  $lps2$  
Load curve for plastic moment versus rotation at node 2 in $s$-direction.

$SFS2$  $sfs2$  
Scale factor for plastic moment versus rotation curve at node 2 in $s$-direction.

$LPT1$  $lpr1$  
Load curve for plastic moment versus rotation at node 1 in $t$-direction.

$SFT1$  $sft1$  
Scale factor for plastic moment versus rotation curve at node 1 in $t$-direction.

$LPT2$  $lpr2$  
Load curve for plastic moment versus rotation at node 2 in $t$-direction.

$SFT2$  $sft2$  
Scale factor for plastic moment versus rotation curve at node 2 in $t$-direction.

**Material Type 30 (Closed-Form Update Shell Plasticity)**

Default heading: Material Type #30 (Closed-Form Update Shell Plasticity)

Input any two of the following:

BULK $K$  
Bulk modulus.

$E$  $E$  
Young's modulus.

$G$  $G$  
Shear modulus.

$PR$  $v$  
Poisson's ratio.
Additional Options:

**SIGY** $\sigma_y$  
Yield strength.

**ET** $E_t$  
Hardening modulus.

This model is available for the Belytschko-Schwer beam and the Belyschko-Tsay shell and is still under development. For beams, the treatment is elastic-perfectly plastic, but for shell elements, isotropic hardening is approximately modeled.

**Material Type 31 (Frazer-Nash Rubber Model)**

This model implements a hyperelastic constitutive law described in [13].

\[
\begin{align*}
C_{001} & C_{001} \\
C_{010} & C_{010} \\
C_{020} & C_{020} \\
C_{100} & C_{100} \\
C_{101} & C_{101} \\
C_{110} & C_{110} \\
C_{200} & C_{200} \\
C_{210} & C_{210} \\
C_{300} & C_{300} \\
C_{400} & C_{400} \\
\text{LIMIT } l & \\
& \text{Limit option} \\
& = 0.0: \text{ stop if strain limits are exceeded.} \\
& 10.0: \text{ continue if strain limits are exceeded.} \\
\text{EMAX } \varepsilon_{\text{max}} & \text{ Maximum strain limit.} \\
\text{EMIN } \varepsilon_{\text{min}} & \text{ Minimum strain limit.}
\end{align*}
\]

The strain energy function, $U$, is defined in terms of the input constants as:

\[
U = C_{100} I_1 + C_{200} I_1^2 + C_{300} I_1^3 + C_{400} I_1^4 + C_{010} I_2 + \\
C_{020} I_2^2 + C_{110} I_1 I_2 + C_{210} I_1^2 I_2 + C_{001} I_3 + C_{101} I_1 I_3
\]

The derivative of $U$ with respect to a component of strain gives the corresponding component of stress:
\[ S_{ij} = \frac{\partial U}{\partial E_{ij}} = 2 \frac{\partial U}{\partial C_{ij}} \]

where \( S_{ij}, E_{ij}, \) and \( C_{ij} \) are the second Piola-Kirchhoff stress tensor, the Green-St. Venant strain tensor, and the right Cauchy-Green deformation tensor, respectively.

**Material Type 32 (Laminated Glass Model)**

- **EG** \( E_g \) Young's modulus for glass.
- **PRG** \( \nu_g \) Poisson's ratio for glass.
- **SYG** \( (\sigma_y)_g \) Yield stress for glass.
- **ETG** \( (E_t)_g \) Hardening modulus for glass.
- **FSG** \( (\epsilon_f)_g \) Failure strain.
- **EP** \( E_p \) Young's modulus for polymer.
- **PRP** \( \nu_p \) Poisson's ratio for polymer.
- **SYP** \( (\sigma_y)_p \) Yield stress for polymer.
- **ETP** \( (E_t)_p \) Hardening modulus for polymer.
- **IOPT** \( f_1 \ldots f_n \) Integration point options.

\( f_i = 0: \text{glass}. \)
\( f_i = 1: \text{polymer}. \)

Isotropic hardening is assumed. The material to which the glass is bonded is assumed to stretch plastically without failure. A user defined integration rule is required which specifies the thickness of the layers making up the glass. There must be the same number of parameters for the IOPT command as integration points.

**Material Type 34 (Fabric)**

The fabric material is similar to the orthotropic composite model (22). It is designed to allow a fabric to be modeled as layers of orthotropic material. The principal characteristic of a fabric material is that it does not support compressive stresses. This is because it is usually modeled with elements that are at least an order-of-magnitude wider than the thickness of the material. This model is still somewhat experimental and model 22 is frequently substituted.

- **EA** \( E_a \) See constitutive matrix below.
- **EB** \( E_b \)
EC \( E_c \)
PRBA \( \nu_{ba} \)
PRCA \( \nu_{ca} \)
PRCB \( \nu_{cb} \)
GAB \( G_{ab} \)
GBC \( G_{bc} \)
GCA \( G_{ca} \)
CSF \( csf \) Compressive modulus scale factor.
TSF \( tsf \) Tensile modulus scale factor.
EXP \( exp \) Exponent.
CSEF \( f \) Compressive stress elimination flag.

=0: use the variable modulus method.
=1: truncate stresses (recommended).

The material law that relates stresses to strains is defined as:

\[
\mathbf{C} = \mathbf{T}^T \mathbf{C}_L \mathbf{T},
\]

where \( \mathbf{T} \) is a transformation matrix, and \( \mathbf{C}_L \) is the constitutive matrix defined in terms of the material constants of the orthogonal material axes, a, b, and c. The inverse of \( \mathbf{C}_L \) is defined as

\[
\mathbf{C}_L^{-1} = \begin{bmatrix}
\frac{1}{E_a} & -\nu_{ba} & -\nu_{ca} & 0 & 0 & 0 \\
-\nu_{ab} & \frac{1}{E_b} & -\nu_{cb} & 0 & 0 & 0 \\
-\nu_{ac} & -\nu_{bc} & \frac{1}{E_c} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{G_{ab}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{G_{bc}} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{G_{ca}}
\end{bmatrix}
\]

Note that \( \frac{\nu_{ab}}{E_a} = \frac{\nu_{ba}}{E_b}, \frac{\nu_{ca}}{E_c} = \frac{\nu_{ac}}{E_a}, \frac{\nu_{cb}}{E_c} = \frac{\nu_{bc}}{E_b} \).
AOPT aopt  
Material axes option (Figure 20-1).
=0.0: locally orthotropic with materials axes determined by element nodes $n_1$, $n_2$, and $n_4$, (see Figure 20-1).
=1.0: locally orthotropic with materials axes determined by a point in space and global location of element center.
=2.0: globally orthotropic with materials axes determined by vectors defined below.
=3.0: SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

- $x_p$: Define for AOPT = 1.
- $y_p$: Define for AOPT = 1.
- $z_p$: Define for AOPT = 1.
- $a_1$: Define for AOPT = 2.
- $a_2$: Define for AOPT = 2.
- $a_3$: Define for AOPT = 2.
- $d_1$: Define for AOPT = 2.
- $d_2$: Define for AOPT = 2.
- $d_3$: Define for AOPT = 2.
- $v_1$: Define for AOPT = 3.
- $v_2$: Define for AOPT = 3.
- $v_3$: Define for AOPT = 3.

**Material Type 35 (Kinematic/Isotropic Elastic-Plastic Green-Naghdi Rate)**

Default heading: Material Type #35 (Green-Naghdi Rate Plasticity)

Input any two of the following:

- **BULK K**: Bulk modulus.
- **E E**: Young's modulus.
- **G G**: Shear modulus.
- **PR v**: Poisson's ratio.

**Additional Options:**

- **SIGY σ_y**: Yield strength
- **ET E_t**: Hardening modulus
**BSA** $\beta' \quad$ Hardening parameter, $0 \leq \beta' \leq 1$

**SC** $c \quad$ Strain rate parameter, $C$

**SP** $p \quad$ Strain rate parameter, $p$

This model is available only for brick elements and is similar to model 3 but uses the Green-Naghdi Rate formulation rather than the Jaumann rate.

**Material Type #37 (Transversely Anisotropic Elastic-Plastic)**

Default heading: Material Type #37 (Transversely Anisotropic Elastic-Plastic)

Input any two of the following:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BULK $K$</td>
<td>Bulk modulus.</td>
</tr>
<tr>
<td>$E$ $E$</td>
<td>Young's modulus.</td>
</tr>
<tr>
<td>$G$ $G$</td>
<td>Shear modulus.</td>
</tr>
<tr>
<td>PR $\nu$</td>
<td>Poisson's ratio.</td>
</tr>
</tbody>
</table>

Additional Options:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ET $E_t$</td>
<td>Hardening modulus</td>
</tr>
<tr>
<td>LCSS $lc$</td>
<td>Load curve number for stress-strain curve.</td>
</tr>
<tr>
<td>R $R$</td>
<td>Anisotropic hardening parameter, $R$.</td>
</tr>
<tr>
<td>SIGY $\sigma_y$</td>
<td>Yield strength</td>
</tr>
</tbody>
</table>

This model is only available for shell elements and is intended for modeling sheet metal forming processes. This is a degenerate form of Hill's model which assumes similar in-plane flow characteristics in all directions but different through-thickness effects. See the LS-DYNA3D manual for more details.

**Material Type 41-50 (User Defined Material Models)**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NPTS npts</td>
<td>Number of material parameters.</td>
</tr>
<tr>
<td>PARAM parameter 1 ... parameter</td>
<td>Material parameters.</td>
</tr>
<tr>
<td>AOPT aopt</td>
<td>Material axes option (Figure 20-1).</td>
</tr>
</tbody>
</table>
=0.0: locally orthotropic with materials axes determined by element nodes \( n_1, n_2, \) and \( n_4 \), (see Figure 20-1).

=1.0: locally orthotropic with materials axes determined by a point in space and global location of element center.

=2.0: globally orthotropic with materials axes determined by vectors defined below.

=3.0: SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

\[
\begin{align*}
\text{XP } & x_p \\
\text{YP } & y_p \\
\text{ZP } & z_p \\
\text{A1 } & a_1 \\
\text{A2 } & a_2 \\
\text{A3 } & a_3 \\
\text{D1 } & d_1 \\
\text{D2 } & d_2 \\
\text{D3 } & d_3 \\
\text{V1 } & v_1 \\
\text{V2 } & v_2 \\
\text{V3 } & v_3 \\
\end{align*}
\]

Define for AOPT = 1.

\[
\begin{align*}
\text{BULK } & K \\
\text{E } & E \\
\text{G } & G \\
\text{PR } & v \\
\end{align*}
\]

Bulk modulus.

Young's modulus.

Shear modulus.

Poisson's ratio.

**Material Type 51 (Temperature and Rate Dependent Plasticity)**

Input any two of the following:

\[
\begin{align*}
\text{BULK } & K \\
\text{E } & E \\
\text{G } & G \\
\text{PR } & v \\
\end{align*}
\]

Bulk modulus.

Young's modulus.

Shear modulus.

Poisson's ratio.

Additional Options:

\[
\begin{align*}
\text{T } & T \\
\text{HC } & HC \\
\text{COEF } & C_1 \ldots C_{18} \\
\text{ALPHA } & \alpha_1 \alpha_2 \alpha_4 \alpha_5 \alpha_6 \\
\end{align*}
\]

Initial Temperature.

Heat generation coefficient.

Model Coefficients.

Initial value of internal state variables.
Material Type 52 (Sandia's Damage Model)

Input any two of the following:

- **BULK K** Bulk modulus.
- **E E** Young's modulus.
- **G G** Shear modulus.
- **PR v** Poisson's ratio.

Additional Options:

- **T T** Initial Temperature.
- **HC HC** Heat generation coefficient.
- **COEF C_1 \ldots C_{18}** Model Coefficients.
- **ALPHA \alpha_1 \alpha_2 \alpha_4 \alpha_5 \alpha_6** Initial value of internal state variables.
- **NEXP n** Exponent in damage evolution
- **D0 D0** Initial damage (porosity).

Material Type 53 (Low Density Closed Cell Polyurethane Foam)

Options:

- **E E** Young's modulus.
- **GAM0 \gamma_0** Initial volumetric strain.
- **P0 p_0** Initial foam pressure.
- **PA a** a.
- **PB b** b.
- **PC c** c.
- **PHI \phi** Ratio of foam to polymer density.
See the LS-DYNA3D manual for a description of this model.

**Material Type 54 and 55 (Composite Damage Model)**

Material 54 uses the Chang matrix failure criterion (the same as model 22). Material 55 uses the Tsay-Wu criterion. These models are for LS-920 and later.

\[
\begin{align*}
EA & \quad E_a \\
EB & \quad E_b \\
EC & \quad E_c \\
PRBA & \quad \nu_{ba} \\
PRCA & \quad \nu_{ca} \\
PRCB & \quad \nu_{cb} \\
GAB & \quad G_{ab} \\
GBC & \quad G_{bc} \\
GCA & \quad G_{ca} \\
FBRT & \quad fbrt \\
SOFT & \quad soft \\
K & \quad K_f \\
SC & \quad S_c \\
XT & \quad x_t \\
YT & \quad y_t \\
YC & \quad y_c \\
ALPH & \quad \alpha \\
TFAIL & \quad t_f \\
AOPT & \quad aopt
\end{align*}
\]

- **See constitutive matrix below.**

- Softening for fiber tensile strength
  - =0.0: fiber rupture with tension cutoff.
  - >0.0: stress=fbrt, \( X_c \) after failure.

- Softening reduction factor for material strength in crashfront elements (default=1.0)

- Bulk modulus of failed material.

- Shear strength, \( ab \) plane.

- Longitudinal tensile strength, \( a \)-axis.

- Transverse tensile strength, \( b \)-axis.

- Transverse compressive strength.

- Non-linear shear stress parameter.

- Time step for automatic element deletion.

Material axes option (Figure 20-1).

- =0.0: locally orthotropic with materials axes determined by element nodes \( n_1, n_2, \) and \( n_4 \), (see Figure 20-1).

- =1.0: locally orthotropic with materials axes determined by a point in space and global location of element center.

- =2.0: globally orthotropic with materials axes determined by vectors defined below.

- =3.0: SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.
Material Type 57 (Low Density Urethane Foam)

This model is for LS-920 and later.

Options:

- **BETA** $\beta$: Decay constant.
- **E** $E$: Young's modulus.
- **LC** $l$: Load curve number of nominal stress versus strain.
- **TENSION** $t$: Tension cut-off stress.
- **UNLOAD** $d$: Hysteretic unloading factor between 0 and 1 (Default=1, i.e. no energy dissipation).

See the LS-DYNA3D manual for a description of this model.

Material Type 59 (Composite Failure Model - Plasticity Based)

This model is for LS-920 and later.

- **EA** $E_a$: See constitutive matrix below.
- **EB** $E_b$
- **EC** $E_c$
- **PRBA** $\nu_{ba}$
- **PRCA** $\nu_{ca}$
- **PRCB** $\nu_{cb}$
- **GAB** $G_{ab}$
GBC \( G_{bc} \)
GCA \( G_{ca} \)

**FBRT** fb\( r \)t

Softening for fiber tensile strength

\( =0.0 \): fiber rupture with tension cutoff.

\( >0.0 \): stress=fb\( r \)t, \( X_c \) after failure.

**SOFT** soft

Softening reduction factor for material strength in crashfront elements (default=1.0)

**SF** sf

Softening factor. (default=0.0).

**SR** sr

Reduction factor. (default=0.447).

**K** \( K_f \)

Bulk modulus of failed material.

**SC** \( S_c \)

Shear strength, \( ab \) plane.

**XT** \( x_t \)

Longitudinal tensile strength, \( a \)-axis.

**YT** \( y_t \)

Transverse tensile strength, \( b \)-axis.

**YC** \( y_c \)

Transverse compressive strength.

**ALPH** \( \alpha \)

Non-linear shear stress parameter.

**TFAIL** \( t_f \)

Time step for automatic element deletion.

**AOPT** aopt

Material axes option (Figure 20-1).

\( =0.0 \): locally orthotropic with materials axes determined by element nodes \( n_1 \), \( n_2 \), and \( n_4 \), (see Figure 20-1).

\( =1.0 \): locally orthotropic with materials axes determined by a point in space and global location of element center.

\( =2.0 \): globally orthotropic with materials axes determined by vectors defined below.

\( =3.0 \): SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

**XP** \( x_p \)

Define for AOPT = 1.

**YP** \( y_p \)

Define for AOPT = 1.

**ZP** \( z_p \)

Define for AOPT = 1.

**A1** \( a_1 \)

Define for AOPT = 2.

**A2** \( a_2 \)

Define for AOPT = 2.

**A3** \( a_3 \)

Define for AOPT = 2.

**D1** \( d_1 \)

Define for AOPT = 2.

**D2** \( d_2 \)

Define for AOPT = 2.

**D3** \( d_3 \)

Define for AOPT = 2.

**V1** \( v_1 \)

Define for AOPT = 3.

**V2** \( v_2 \)

Define for AOPT = 3.

**V3** \( v_3 \)

Define for AOPT = 3.
Material Type 60 (Elastic with Viscosity)

This model is for LS-910 and later.

Input any two of the following:

- **BULK K**  
  Bulk modulus.
- **E E**  
  Young’s modulus.
- **G G**  
  Shear modulus.
- **PR ν**  
  Poisson’s ratio.

Additional Options:

- **NPTS npts**  
  Number of points. (npts≤8). (Default=1).
- **T T₁ ... Tₙ**  
  Temperatures. (input only if npts>1.)
- **VC ν₁ ... νₙ**  
  Viscosity coefficients (at least one is input.)

Material Type 64 (Simple Creep Model)

This model is for LS-930 and later.

Input any two of the following:

- **BULK K**  
  Bulk modulus.
- **E E**  
  Young’s modulus.
- **G G**  
  Shear modulus.
- **PR ν**  
  Poisson’s ratio.

Additional Options:

- **EI ei**  
  Value for ei.
- **K k**  
  Value for k.
- **LCK lck**  
  Load curve for k.
- **LCM lcm**  
  Load curve for m.
- **M m**  
  Value for m.
- **N n**  
  Value for n.
Material Type Belt

This is a special material which applies to beam elements only. When material type belt is specified, beams are converted to the special seat belt element in LS-920 and later. Dummy beam elements are output to LS-DYNA3D also for viewing as null materials.

LCL lcl Load curve for loading.
LCU lcu Load curve for unloading.
RO r Mass per unit length.
MINIMUM l Minimum allowable length. (This is used to determine the minimum element size before an element is passed through a slip ring.)

Example:

MAT 56 TYPE BELT LCL 24 LCU 24 RO [0.100/386.4]
MINIMUM 0.2 BEAM ENDMAT
21. Equations-of-State

Equations-of-state are required by certain LS-DYNA2D and LS-DYNA3D material models. They provide a relationship between pressure, relative volume, and temperature (or internal energy) which is used in place of a bulk modulus. Equations-of-state are needed when significant volume changes occur during a deformation process. They are attached to a material model and the general form of the input is:

```
MAT i TYPE j {material options} ENDMAT
EOS k {equation-of-state options} ENDEOS
```

This will define material $i$ as being of type $j$ and having equation-of-state characteristics of type $k$.

**Equation-of-State Form 1 (Linear Polynomial)**

Default heading: Equation-of-State Form 1 (Linear Polynomial)

```
CO C0
C1 C1
C2 C2
C3 C3
C4 C4
C5 C5
C6 C6
E0 E0
V0 V0
ENDEOS
```

Where

```
E = \frac{\rho}{\rho_0} - 1
```

CO $C_0$ See equation below
C1 $C_1$
C2 $C_2$
C3 $C_3$
C4 $C_4$
C5 $C_5$
C6 $C_6$
E0 $E_0$ Initial internal energy
V0 $V_0$ Initial relative volume
ENDEOS End equation-of-state definition.

The linear polynomial equation-of-state is linear in internal energy. The pressure is given by:

$$P = C_0 + C_1\mu + C_2\mu^2 + C_3\mu^3 + \left(C_4 + C_5\mu + C_6\mu^2\right)E,$$

where terms $C_2\mu^2$ and $C_6\mu^2$ are set to zero if $\mu<0$, $\mu=\rho/\rho_0-1$ and $\rho/\rho_0$, is the ratio of current density to the initial density.
Equation-of-State Form 2 (JWL)

Default heading: Equation-of-State Form 2 (JWL High Explosive)

A A See equation below.
B B
R1 R1
R2 R2
OMEGA ω
E0 E0 Initial internal energy
V0 V0 Initial relative volume
ENDEOS End equation-of-state definition.

The JWL equation-of-state defines the pressure as

\[ p = A \left( 1 - \frac{\omega}{R_1 V} \right) e^{-R_1 V} + B \left( 1 - \frac{\omega}{R_2 V} \right) e^{-R_2 V} + \frac{\omega E}{V}, \]

and is usually used for detonation products of high explosives.

Equation-of-State Form 3 (Sack)

Default heading: Equation-of-State Form 3 (Sack Tuesday High Explosive)

A1 A1 See equation below
A2 A2
A3 A3
B1 B1
B2 B2
E0 E0 Initial internal energy
V0 V0 Initial relative volume
ENDEOS End equation-of-state definition.

The Sack equation-of-state defines the pressure as

\[ p = \frac{A_3}{V} A_1 e^{-A_2 V} \left( 1 - \frac{B_1}{V} \right) + \frac{B_2}{V} E. \]
and is used for detonation products of high explosives.

**Equation-of-State Form 4 (Gruneisen)**

Default heading: Equation-of-State Form 4 (Gruneisen)

See equation below.

SP $C$

S1 $S_1$

S2 $S_2$

S3 $S_3$

GAMMA $\gamma_0$

SA $a$

E0 $E_0$ Initial internal energy.

V0 $V_0$ Initial relative volume.

ENDEOS End equation-of-state definition.

The Gruneisen equation-of-state with cubic shock velocity-particle velocity defines pressure for compressed materials as

$$p = \rho_0 C^2 \mu \left[ 1 + \left( 1 - \frac{\gamma_0}{2} \right) \mu - \frac{a}{2} \mu^2 \right] \times \left( 1 - \left( S_1 - 1 \right) \mu - S_2 \frac{\mu^2}{\mu+1} - S_3 \frac{\mu^3}{(\mu+1)} \right)^2 + \left( \gamma_0 + a \mu \right) E.$$ 

and for expanded materials as

$$p = \rho_0 C^2 \mu + \left( \gamma_0 + a \mu \right) E.$$ 

where $C$ is the intercept of the $u_s-u_p$ curve, $S_1$, $S_2$, and $S_3$ are the coefficients of the slope of the $u_s-u_p$ curve, $\gamma_0$ is the Gruneisen gamma; and $a$ is the first order volume correction to $\gamma_0$ and $\mu = \frac{\rho}{\rho_0} - 1$. 
Equation-of-State Form 5 (Ratio of Polynomials)

Default heading: Equation-of-State Form 5 (Ratio of Polynomials)

\[
\begin{align*}
A10 & A_{10} \\
A11 & A_{11} \\
A12 & A_{12} \\
A13 & A_{13} \\
A20 & A_{20} \\
A21 & A_{21} \\
A22 & A_{22} \\
A23 & A_{23} \\
A30 & A_{30} \\
A31 & A_{31} \\
A32 & A_{32} \\
A33 & A_{33} \\
A40 & A_{40} \\
A41 & A_{41} \\
A42 & A_{42} \\
A43 & A_{43} \\
A50 & A_{50} \\
A51 & A_{51} \\
A52 & A_{52} \\
A53 & A_{53} \\
A60 & A_{60} \\
A61 & A_{61} \\
A62 & A_{62} \\
A63 & A_{63} \\
A70 & A_{70} \\
A71 & A_{71} \\
A72 & A_{72} \\
A73 & A_{73} \\
\text{ALPHA} & a \\
\text{BETA} & b \\
A14 & A_{14} \\
A24 & A_{24} \\
\text{COEF A10..A24} & \\
\text{E0} & E_0 \\
\text{V0} & V_0 \\
\text{ENDEOS} & \\
\end{align*}
\]

List the 32 above coefficients in the same order as they appear.

\text{E0} \quad \text{Initial internal energy}

\text{V0} \quad \text{Initial relative volume}

\text{ENDEOS} \quad \text{End equation-of-state definition.}
The ratio of polynomials equation-of-state defines the pressure as

\[ p = \frac{F_1 + F_2 E + F_3 E^2 + F_4 E^3}{F_5 + F_6 E + F_7 E^2} \left(1 + \alpha \mu \right) \]

where

\[ F_i = \sum_{j=0}^{n} A_{ij} \mu^j \quad n = 4 \text{ if } i < 3 \]

\[ \mu = \frac{\rho}{\rho_0} - 1 \quad n = 3 \text{ if } i \cdot 3 \]

In expanded zones \( F_1 \) is replaced by \( F' \). By setting coefficient \( A_{10} = 1.0 \), the delta-phase pressure modeling for this material will be initiated. The code will reset it to 0.0 after setting flags.

---

**Equation-of-State Form 6 (Linear Polynomial With Energy Leak)**

Default heading: Equation-of-State Form 6 (Linear Polynomial with Energy Leak)

- **C1** \( C_1 \) See Equation-of-State Form 1.
- **C2** \( C_2 \)
- **C3** \( C_3 \)
- **C4** \( C_4 \)
- **C5** \( C_5 \)
- **C6** \( C_6 \)
- **E0** \( E_0 \) Initial internal energy.
- **V0** \( V_0 \) Initial relative volume.
- **CN** \( cn \) Curve number of time history that gives energy deposition rate.
- **ENDEOS** End equation-of-state definition.
Equation-of-State Form 7 (Ignition and Growth of Reaction in HE)

Default heading: Equation-of-State Form 7 (Ignition and Growth of Reaction in High Explosive)

\[ \text{AP} A_p \quad \text{See equations below} \]
\[ \text{BP} B_p \quad \text{See equations below} \]
\[ \text{RP} R_{1p} \quad \text{See equations below} \]
\[ \text{R2P} R_{2p} \quad \text{See equations below} \]
\[ G \quad \text{Second ignition coefficient} \]
\[ \text{WPCP} w_p C_p \quad \text{See equations below} \]
\[ \text{AE} A_e \quad \text{See equations below} \]
\[ \text{BE} B_e \quad \text{See equations below} \]
\[ \text{WECE} w_e C_e \quad \text{See equations below} \]
\[ \text{R1E} R_{1e} \quad \text{See equations below} \]
\[ \text{R2E} R_{2e} \quad \text{See equations below} \]
\[ \text{FCRIT} \quad \text{Critical fraction reached} \]
\[ I \quad \text{First ignition coefficient} \]
\[ H \quad \text{Growth coefficient} \]
\[ Z \quad \text{Pressure exponent} \]
\[ X \quad \text{See equations below} \]
\[ Y \quad \text{See equations below} \]
\[ \text{CP} \quad C_p \quad \text{Heat capacity of reaction products} \]
\[ \text{CE} \quad C_e \quad \text{Heat capacity of unreacted HE} \]
\[ M \quad m \quad \text{(generally} = 0) \]
\[ T0 \quad T_0 \quad \text{Initial temperature (°K)} \]
\[ E0 \quad E_0 \quad \text{Initial internal energy} \]
\[ \text{ENDEOS} \quad \text{End equation-of-state definition.} \]

A JWL equation-of-state defines the pressure in the unreacted HE as

\[ P_e = A_e \left( 1 - \frac{\omega_e}{R_{1e} V_e} \right) e^{-R_{1e} V_e} + B_e \left( 1 - \frac{\omega_e}{R_{2e} V_e} \right) e^{-R_{2e} V_e} + \frac{\omega E_e}{V_e} \]

where \( V_e \) is the relative volume, \( E_e \) is the internal energy, and the constants \( A_e, B_e, w_e, R_{1e}, \) and \( R_{2e} \) are input constants. Similarly the pressure in the reaction products is defined by another JWL form

\[ P_p = A_p \left( 1 - \frac{\omega_p}{R_{1p} V_p} \right) e^{-R_{1p} V_p} + B_e \left( 1 - \frac{\omega_p}{R_{2p} V_p} \right) e^{-R_{2p} V_p} + \frac{\omega E_p}{V_p} \]

The mixture of unreacted explosive and reaction products is defined by the fraction reacted \( F \) (\( F=0 \) " no reaction, \( F=1 \) " complete conversion from explosive to products. The pressures and temperatures are assumed to be in equilibrium and the volumes are assumed to be additive.)
\[ V = (1-F)V_e + FV_p \]

The rate of reaction is

\[ \frac{\partial F}{\partial t} = I (\text{FCRIT} - F)^\gamma \left( V_e^{-1} - 1 \right)^3 \left[ 1 + G \left( V_e^{-1} - 1 \right) \right] + H (1 - F)^\gamma F^x P^z (V_p^{-1} - 1)^m \]

where \( I, G, H, x, y, z, \) and \( m \) (generally \( m=0 \)) are input constants.

The JWL equations of state and the reaction rates have been fitted to one- and two-dimensional shock initiation and detonation data for four explosives: PBX-9404, RX-03-BB, PETN, and cast TNT. The details of the calculational method are described by Cochran and Chan [14]. The detailed one-dimensional calculations and parameters for the four explosives are given by Lee and Tarver [15].

**Equation-of-State Form 8 (Tabulated-Compaction)**

Default heading: Equation-of-State Form 8 (Tabulated-Compaction)

- **NPTS \( n \)**: Number of points in tabulated curves.
- **LNV \( e_{V_1}, e_{V_2}, ..., e_{V_n} \)**: Volumetric strain points, \( e_{V_i} = \ln(V_i) \).
- **PC \( C_1, C_2, ..., C_n \)**: Points on the curve for \( C(e_{V}) \).
- **PT \( T_1, T_2, ..., T_n \)**: Points on the curve for \( T(e_{V}) \).
- **KU \( K_1, K_2, ..., K_n \)**: Points on the curve for the unloading bulk modulus.
- **GAMMA \( \gamma \)**: See equation below.
- **E0 \( E_0 \)**: Initial internal energy.
- **V0 \( V_0 \)**: Initial relative volume.
- **ENDEOS**

End equation-of-state definition.

The tabulated compaction model is linear in internal energy. Pressure is defined by

\[ p = C(e_{V}) + \gamma T(e_{V}) E \]

in the loading phase. The volumetric strain \( e_{V} \) is given by the natural logarithm of the relative volume. Unloading occurs along the unloading bulk modulus to the pressure cutoff. Reloading always follows the unloading path to the point where unloading began, and continues on the loading path. See Figure 21-1. Up to 10 points and as few as 2 may be used when defining the tabulated function, LS-DYNA2D/3D will extrapolate to find the pressure.
if necessary.

Figure 21-1. Pressure versus volumetric strain curve for equation-of-state form 8 with compaction. In the compacted states the bulk unloading modulus depend on the peak volumetric strain.

Equation-of-State Form 9 (Tabulated)

Default heading: Equation-of-State Form 9 (Tabulated)

- **NPTS** \( n \)  
  Number of points in tabulated curves.

- **LNV** \( e V_1, e V_2, e V_n \)  
  Volumetric strain points, \( e V_i = ln(V_i) \).

- **PC** \( C_1, C_2, ..., C_n \)  
  Points on the curve for \( C(e V) \).

- **PT** \( T_1, T_2, ..., T_n \)  
  Points on the curve for \( T(e V) \).

- **GAMMA** \( g \)  
  See equation below.
The tabulated compaction model is linear in internal energy. Pressure is defined by

\[ P = C (\varepsilon_V) + \gamma T (\varepsilon_V) E \]

in the loading phase. The volumetric strain \( \varepsilon_V \) is given by the natural logarithm of the relative volume. Unloading occurs along the unloading bulk modulus to the pressure cutoff. Reloading always follows the unloading path to the point where unloading began, and continues on the loading path. See Figure 21-1. Up to 10 points and as few as 2 may be used when defining the tabulated function, LS-DYNA2D/3D will extrapolate to find the pressure if necessary.
22. LS-NIKE2D Commands and Materials

Analysis options are code dependent. They can be set either in the control section of the LS-INGRID input file or in the graphics phase. These commands become active when LS-NIKE2D output is selected with the NK2D command.

**ANAL** \( n \)  
Analysis type  
"STAT": static analysis (default).  
"DYN": direct time integration.  
"DYNS": direct time integration with static initialization.  
"EIGE": eigenvalue extraction.

**BWMO** \( n \)  
Bandwidth minimization option.  
"ON": perform minimization in analysis code (default).  
"OFF": don't minimize bandwidth.

**DCTOL** tol  
Convergence tolerance on displacements. LS-NIKE2D defaults to 0.001.

**DELT** \( Dt \)  
Time step size for LS-NIKE2D.

**DTMAX** \( D \)  
Maximum step size permitted. If SSO = "AUTO" the default is set by LS-NIKE2D.

**DTMN** \( d \)  
Minimum step size permitted. If SSO = "AUTO" the default is set by LS-NIKE2D.

**ECTOL** tol  
Convergence tolerance on energy. LS-NIKE2D defaults to 0.01.

**GEOM** \( sn \)  
Node and element data dump interval for high speed printer.  
"PLAN"  Plane strain  
"STRE"  Plane stress  
"AXIS"  Axisymmetric

**GRAV** \( gx \ y \ g_z \)  
Gravity acceleration vector. The gravitational field is scaled in time by load curve one.

**GSTIF** on/off  
Geometric stiffness option. The default is off and generally gives the best results.

**IPLT** \( n \)  
Node and element data dump interval for TAURUS post-processing.

**LST** tol  
Line search tolerance.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSRF</td>
<td>Maximum number of stiffness reformations per time step. LS-NIKE2D defaults to the recommended value of 15.</td>
</tr>
<tr>
<td>NBEI</td>
<td>The number of time steps between equilibrium iterations.</td>
</tr>
<tr>
<td>NBSR</td>
<td>The number of time steps between stiffness matrix reformation.</td>
</tr>
<tr>
<td>NEIG</td>
<td>Number of eigenvectors. This option turns on the subspace iteration eigenvalue/eigenvector solution method and overrides all other solution options. Eigenvectors are mass normalized and written into the graphics database. The time word corresponds to the frequency in radians/units of time.</td>
</tr>
<tr>
<td>NIBSR</td>
<td>Maximum number of equilibrium iterations permitted between stiffness matrix reformations. LS-NIKE2D defaults to the recommended value of 10.</td>
</tr>
<tr>
<td>NIP1</td>
<td>First Newmark integration parameter.</td>
</tr>
<tr>
<td>NIP2</td>
<td>Second Newmark integration parameter.</td>
</tr>
<tr>
<td>NSMD</td>
<td>Nonlinear solution method.</td>
</tr>
<tr>
<td></td>
<td>&quot;BFGS&quot;: BFGS (default)</td>
</tr>
<tr>
<td></td>
<td>&quot;BROY&quot;: Broyden's</td>
</tr>
<tr>
<td></td>
<td>&quot;MODN&quot;: modified Newton</td>
</tr>
<tr>
<td>NSTEP</td>
<td>Number of desired time steps.</td>
</tr>
<tr>
<td>RFTS</td>
<td>Reduction factor for tangential stiffness. This is used for modeling the stick condition due to friction in the penalty formulation of contact.</td>
</tr>
<tr>
<td>SBRF</td>
<td>Number of time steps between restart file generations. If zero, LS-NIKE2D writes a restart file as it terminates.</td>
</tr>
<tr>
<td>SHIFT</td>
<td>Shift frequency in hertz. This option works with</td>
</tr>
</tbody>
</table>
the eigenvalue/eigenvector solution method. Using this option, LS-NIKE2D will find the NEIG eigenvalues nearest to \( w \). If the model has rigid body modes, a negative value for \( w \) should be used to make the run stable. If \( w \) is exactly the same value as an eigenvalue the system becomes singular.

<table>
<thead>
<tr>
<th>SSIT</th>
<th>( s )</th>
<th>Slide surface insertion tolerance.</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSO</td>
<td>( u )</td>
<td>Step size option.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;AUTO&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;MANUAL&quot;</td>
</tr>
<tr>
<td>SSOO</td>
<td>( n )</td>
<td>Optimal number of iterations per step.</td>
</tr>
<tr>
<td>TEO</td>
<td>( i )</td>
<td>Thermal effects option</td>
</tr>
<tr>
<td></td>
<td></td>
<td>=0: no thermal effects.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>=N: nodal temperatures are defined in input and are scaled according to a time function. N is the load curve number.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>=-1: each time step a new temperature state is read from a disk file. The time word at the beginning of each temperature state is ignored.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>=-2: each time step a temperature state is interpolated from the temperature state in a disk file. Therefore the time words at the beginning of each temperature state are used.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>=-3: the disk file containing temperatures has only one state. The initial state is assumed to be zero.</td>
</tr>
<tr>
<td>TERM</td>
<td>( t )</td>
<td>Terminate dynamic time integration at time ( t ). The dynamic time step size will be computed if this command is used instead of the &quot;DELT&quot; command.</td>
</tr>
</tbody>
</table>
22.1 LS-NIKE2D Material Input

LS-NIKE2D material input is possible after the NK2D command has been input (see Control Commands). The form of this input is: MAT \( n \) TYPE \( m \) \{options specific to material type \( m \}\} \{general material options\} ENDMAT. \( n \) is a material name which is assigned a number in the order that they occur in the input. Therefore, the materials should be defined in order before any additional use of materials is made.

**Material Type 1 (Elastic)**

Default heading: Material Type #1 (Elastic)

Input any two of the following.

- **BULK** \( K \) Bulk modulus.
- **E** \( E \) Young's modulus.
- **G** \( G \) Shear modulus.
- **PR** \( n \) Poisson's ratio.

**Material Type 2 (Orthotropic Elastic)**

- **EA** \( E_a \) See constitutive matrix below.
- **EB** \( E_b \)
- **EC** \( E_c \)
- **PRBA** \( u_{ba} \)
- **PRCA** \( u_{ca} \)
- **PRCB** \( u_{cb} \)
- **GAB** \( G_{ab} \)
- **AOPT** \( aopt \) Material axes option (Figure 22-1).

  - \( =0.0 \): locally orthotropic with materials axes by \( y \) value specified on each element card and element nodes \( n_1 \) and \( n_2 \), (see Figure 22-1).

  - \( =1.0 \): locally orthotropic with materials axes by a point in space and global location of element center.

  - \( =2.0 \): globally orthotropic with materials axes determined by \( y_G \).

- **RP** \( r_p \) Define for \( AOPT = 1 \).
- **ZP** \( z_p \) Define for \( AOPT = 1 \).
- **PSIG** \( y_G \) Define for \( AOPT = 2 \).
The material law that relates stresses to strains is defined as:

\[
\mathbf{C} = \mathbf{T}^T \mathbf{C}_L \mathbf{T},
\]

Where \( \mathbf{T} \) is a transformation matrix, and \( \mathbf{C}_L \) is the constitutive matrix defined in terms of the material constants of the orthogonal material axes, \( a \), \( b \), and \( c \). The inverse of \( \mathbf{C}_L \) is defined as

\[
\mathbf{C}_L^{-1} = \begin{bmatrix}
\frac{1}{E_a} & -\frac{v_{ba}}{E_b} & -\frac{v_{ba}}{E_c} & 0 & 0 & 0 \\
-\frac{v_{ab}}{E_a} & \frac{1}{E_b} & -\frac{v_{ab}}{E_c} & 0 & 0 & 0 \\
-\frac{v_{ac}}{E_a} & -\frac{v_{bc}}{E_b} & \frac{1}{E_c} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{G_{ab}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{G_{bc}} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{G_{ca}}
\end{bmatrix}
\]

Note that \( \frac{v_{ab}}{E_a} = \frac{v_{ba}}{E_b}, \frac{v_{ca}}{E_c} = \frac{v_{ac}}{E_a}, \frac{v_{cb}}{E_c} = \frac{v_{bc}}{E_b} \).
Figure 22-1. Options for determining principal materials axes: (a) AOPT = 0.0, (b) AOPT = 1.0, and (c) AOPT = 2.0.
Material Type 3 (Kinematic/Isotropic Elastic/Plastic)

Default heading: Material Type #3 (Elastic-Plastic)

Input any two of the following:

- **BULK K**: Bulk modulus.
- **E E**: Young's modulus.
- **G G**: Shear modulus.
- **PR n**: Poisson's ratio.

Additional Options

- **SIGY sy**: Yield strength.
- **ETAN Et**: Hardening modulus.
- **BETA b’**: Hardening parameter, \(0 \leq b' \leq 1\)
- **NPTS n**: Number of points on stress-effective plastic strain curve.
- **ES s1y, s2y, ..., sny**: Effective stress.
- **EPS ep1, ep2, ..., epn**: Effective plastic strain.

Isotropic, kinematic, or a combination of isotropic and kinematic hardening may be specified by varying \(b'\) between 0 and 1. For \(b'\) equal to 0 and 1, respectively, kinematic and isotropic hardening are obtained as shown in Figure 22-2. Effective stress is defined in terms of the deviatoric stress tensor, \(s_{ij}\) as:

\[
\sigma = \left( \frac{3}{2} S_{ij} S_{ij} \right)^{1/2}
\]

where,

\[
S_{ij} = \sigma_{ij} - \frac{1}{3} \sigma_{kk} \delta_{ij}
\]

and effective plastic strain by:

\[
\bar{\varepsilon}^p = \int_0^t \varepsilon^p dt
\]

where \(t\) denotes time and

\[
\varepsilon^p = \left( \frac{2}{3} \frac{d\varepsilon_{ij}^p d\varepsilon_{ij}^p}{d\gamma_{ij}} \right)^{1/2}
\]
Figure 22-2. Elastic-plastic behavior with isotropic and kinematic hardening where $l_0$ and $l$ are undeformed and deformed length of uniaxial tension specimen.
**Material Type 4 (Thermo-Elastic-Plastic)**

Default heading: Material Type 4 (Thermo-Elastic-Plastic)

- **NPTS** \( n \)  
  Number of temperature values for which material constants are defined.
- **TEMP** \( T_1 \ T_2 \ldots T_n \)  
  Temperatures.
- **E** \( E_1 \ E_2 \ldots E_n \)  
  Young’s moduli.
- **PR** \( u_1 \ u_2 \ldots u_n \)  
  Poisson’s ratios.
- **ALPHA** \( a_1 \ a_2 \ldots a_n \)  
  Coefficients of thermal expansion.
- **SIGY** \( \sigma_{y1} \ \sigma_{y2} \ldots \sigma_{yn} \)  
  Yield stresses.
- **ETAN** \( E_{t1} \ E_{t2} \ldots E_{tn} \)  
  Tangent moduli.

**Material Type 5 (Soil and Crushable Foam)**

Default heading: Material Type 5 (Soil and Crushable Foam)

Input any two of the following:

- **BULK** \( K \)  
  Bulk modulus.
- **E** \( E \)  
  Young’s modulus.
- **G** \( G \)  
  Shear modulus.
- **PR** \( \nu \)  
  Poisson’s ratio.

Additional *Options*:

- **AO** \( n \)  
  Yield function constant
- **A1** \( a_1 \)  
  Yield function constant
- **A2** \( a_2 \)  
  Yield function constant
- **PC** \( P_c \)  
  Pressure cutoff for tensile fracture
- **UL** \( u_{opt} \)  
  Unloading option
  
  - =0: volumetric crushing
  - =1: no volumetric crushing
- **NPTS** \( n \)  
  Number of points in volumetric strain versus pressure curve (\( n \leq 10 \)).
- **VS** \( e_{v1} \ e_{v2} \ldots e_{vn} \)  
  Volumetric strain values
- **P** \( p_1 \ p_2 \ldots p_n \)  
  Pressures corresponding to volumetric strain values

The deviatoric yield function, \( \phi \), is described in terms of the second invariant \( J_2 \).

\[
J_2 = \frac{1}{2} \mathbf{s}_j \cdot \mathbf{s}_j
\]
Pressure, \( p \), and constants \( a_0 \), \( a_1 \), and \( a_2 \) as:

\[
\phi = J_2 - \left[ a_0 + a_1 p + a_2 p^2 \right].
\]

On the yield surface, \( J_2 = \frac{1}{3}(\sigma_y)^{1/2} \), where \( \sigma_y \) is the yield stress, i.e.,

\[
\sigma_y = \left[ \frac{1}{3} (a_0 + a_1 p + a_2 p^2) \right]^{1/2}
\]

For elastic-perfectly plastic behavior \( a_1 = a_2 = 0 \), and \((3a_0)^{1/2}\) defines the yield strength. The volumetric strain is given by the natural logarithm of the relative volume \( V \). If the pressure drops below the cutoff value, \( PC \), then it is reset to that value.

![Figure 22-3. Volumetric strain versus pressure curve for soil and crushable foam model.](image-url)
Material Type 6 (Viscoelastic)

\[ G(t) = G + (G_0 - G) e^{-\beta t} \]

A Jaumann rate formulation is used:

\[ \sigma_{ij}^\prime = 2 \int_0^t G(t - \tau) D_{ij}(\tau) d\tau \]

where the prime denotes the deviatoric part of the stress rate, \( \sigma_{ij}^\prime \), and the strain rate \( D_{ij} \).

Material Type 7 (Thermal Orthotropic Elastic)

Default heading: Material Type #7 (Thermal Orthotropic Elastic)

\[ EA E_a \]
\[ EB E_b \]
\[ EC E_c \]
\[ PRBA u_{ba} \]
\[ PRCA u_{ca} \]
\[ PRCB u_{cb} \]
\[ ALPA a_a \]
\[ ALPB a_b \]
\[ ALPC a_c \]
\[ GAB G_{ab} \]
\[ AOPT \text{ aopt} \]

Where:

- \( EA \) is the Young's modulus along axis a.
- \( EB \) is the Young's modulus along axis b.
- \( EC \) is the Young's modulus along axis c.
- \( PRBA \) is the thermal expansion coefficient along axis a.
- \( PRCA \) is the thermal expansion coefficient along axis c.
- \( GAB \) is the shear modulus.
- \( AOPT \) is the material axis option (Figure 22-1). 

- \( =0.0 \): locally orthotropic with materials axes determined by element nodes \( n_1, n_2, \) and \( n_4 \), (see Figure 22-1).

- \( =1.0 \): locally orthotropic with materials axes determined by a point in space and global location of element center.
Material Type 8 (Thermo-Elastic-Creep)

Default heading: Material Type #8 (Thermo-Elastic-Creep)

- **NPTS** $n$ Number of temperature values for which material constants are defined.
- **TEMP** $T_1 T_2 ... T_n$ Temperatures
- **G** $G_1 G_2 ... G_n$ Shear moduli.
- **K** $K_1 K_2 ... K_n$ Bulk moduli.
- **ALPHA** $a_1 a_2 ... a_n$ Coefficients of thermal expansion.
- **A** $a_1 a_2 ... a_n$ Creep parameters.
- **B** $b_1 b_2 ... b_n$ Creep parameters.

In this model, $G$, is the shear modulus and the instantaneous creep is given by a power law of the form

$$
S_{ij} = \mu \left( \frac{1}{V} C_{ij} - V^{-1/2} \nu \delta_{ij} \right)
$$

where $a$ and $b$ are functions of temperature. This model was developed and provided for LS-NIKE2D by R. D. Krieg of Sandia National Laboratories.

Material Type 9 (Blatz-Ko Rubber)

Default heading: Material Type #9 (Rubber)

- **G** $m$ Shear modulus

The second Piola-Kirchhoff stress is computed as

$$
S_{ij} = \mu \left( \frac{1}{V} C_{ij} - V^{-1/2} \nu \delta_{ij} \right)
$$

where $V$ is the relative volume, $C_{ij}$ is the right Cauchy-Green strain tensor, and $n$ is the
Poisson's ratio which is set to .463 internally. This stress measure is transformed to the Cauchy stress, \( s_{ij} \), according to the relationship

\[
\sigma_{ij} = V^{-1} F_{ik} F_{jl} S_{lk}
\]

where \( F_{ij} \) is the deformation gradient tensor.

**Material Type 10 (Power Law Plasticity)**

Input any two of the following:

- **BULK** \( K \) Bulk modulus
- **E** \( E \) Young's modulus
- **G** \( G \) Shear modulus
- **PR** \( n \) Poisson's ratio

Additional Options:

- **K** \( k \) See equation below
- **M** \( m \) See equation below
- **FC** \( f \) Failure criteria.
  - 1: Mohr-Coulomb.
  - 2: Drucker-Prager
  - 3: check both
- **MPS** \( s_{\text{max}} \) Maximum principal stress (optional).
- **MSS** \( t_{\text{max}} \) Maximum shear stress (optional).
- **LC** \( l_c \) Optional failure curve number.

The stress-strain curve for this model is based on the following equation:

\[
\sigma_y = k(\varepsilon_e + \bar{\varepsilon}^P)^n
\]
**Material Type 12 (Power Law Thermo Plasticity)**

- **NPTS** $n$  
  Number of temperature points ($\ell$8).
- **T** $T_1, T_2, ... T_n$  
  Temperatures
- **E** $E_1, E_2, ... E_n$  
  Young's moduli
- **PR** $u_1, u_2, ... u_n$  
  Poisson's ratios
- **K** $k_1, k_2, ... k_n$  
  See equation below
- **M** $m_1, m_2, ... m_n$  
  See equation below

The stress-strain curve for this model is based on the following equation:

$$\sigma_y = k(\varepsilon_e + \varepsilon_p)^n$$

**Material Type 22 (Frazer-Nash Rubber Model)**

This model implements a hyperelastic constitutive law described in [13].

- **C001** C001
- **C010** C010
- **C020** C020
- **C100** C100
- **C101** C101
- **C110** C110
- **C200** C200
- **C210** C210
- **C300** C300
- **C400** C400

The strain energy function, $U$, is defined in terms of the input constants as:

$$U = C_{100}I_1 + C_{200}I_1^2 + C_{300}I_1^3 + C_{400}I_1^4 + C_{010}I_2 +$$

$$+ C_{020}I_2^2 + C_{110}I_1I_2 + C_{210}I_1^2I_2 + C_{001}I_3 + C_{101}I_1I_3$$

The derivative of $U$ with respect to a component of strain gives the corresponding component of stress:

$$S_{ij} = \frac{\partial U}{\partial E_{ij}} = 2 \frac{\partial U}{\partial C_{ij}}$$
where, $S_{ij}$, $E_{ij}$ and $C_{ij}$ are the second Piola-Kirchhoff stress tensor, the Green-St. Venant strain tensor, and the right Cauchy-Green deformation tensor, respectively.
## 23. LS-NIKE3D Commands and Materials

Analysis options are code dependent. They can be set either in the control section of the LS-INGRID input file or in the graphics phase. These commands become active when LS-NIKE3D output is selected with the NK3D command.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANAL $n$</td>
<td>Analysis type</td>
</tr>
<tr>
<td>BWMO $n$</td>
<td>Bandwidth minimization option.</td>
</tr>
<tr>
<td>DCTOL $\text{tol}$</td>
<td>Convergence tolerance on displacements. LS-NIKE3D defaults to 0.001.</td>
</tr>
<tr>
<td>DELT $Dt$</td>
<td>Time step size for LS-NIKE3D.</td>
</tr>
<tr>
<td>DTMAX $D$</td>
<td>Maximum step size permitted. If SSO = &quot;AUTO&quot; the default is set by LS-NIKE3D.</td>
</tr>
<tr>
<td>DTMN $d$</td>
<td>Minimum step size permitted. If SSO = &quot;AUTO&quot; the default is set by LS-NIKE3D.</td>
</tr>
<tr>
<td>ECTOL $\text{tol}$</td>
<td>Convergence tolerance on energy. LS-NIKE3D defaults to 0.01.</td>
</tr>
<tr>
<td>GRAV $g_x g_y g_z$</td>
<td>Gravity acceleration vector. The gravitational field is scaled in time by load curve one.</td>
</tr>
<tr>
<td>GSTIF $\text{on/off}$</td>
<td>Geometric stiffness option. The default is off and generally gives the best results.</td>
</tr>
<tr>
<td>IPLT $n$</td>
<td>Node and element data dump interval for TAURUS post-processing.</td>
</tr>
<tr>
<td>LST $\text{tol}$</td>
<td>Line search tolerance.</td>
</tr>
<tr>
<td>MSRF $n$</td>
<td>Maximum number of stiffness reformations per time step. LS-NIKE3D defaults to the recommended value of 15.</td>
</tr>
<tr>
<td>NBEI $n$</td>
<td>The number of time steps between equilibrium iterations.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>NBSR</strong> $n$</td>
<td>The number of time steps between stiffness matrix reformation.</td>
</tr>
<tr>
<td><strong>NEIG</strong> $n$</td>
<td>Number of eigenvectors. This option turns on the subspace iteration eigenvalue/eigenvector solution method and overrides all other solution options. Eigenvectors are mass normalized and written into the graphics database. The time word corresponds to the frequency in radians/units of time.</td>
</tr>
<tr>
<td><strong>NIBSR</strong> $n$</td>
<td>Maximum number of equilibrium iterations permitted between stiffness matrix reformation. LS-NIKE3D defaults to the recommended value of 10.</td>
</tr>
<tr>
<td><strong>NIP1</strong> $s$</td>
<td>First Newmark integration parameter.</td>
</tr>
<tr>
<td><strong>NIP2</strong> $s$</td>
<td>Second Newmark integration parameter.</td>
</tr>
</tbody>
</table>
| **NSMD** $n$ | Nonlinear solution method.  
  - "BFGS": BFGS (default)  
  - "BROY": Broyden's  
  - "MODN": modified Newton  

To obtain a linear elastic solution, NBSR and NBEI should be larger than the number of time steps in the problem. |
| **NSTEP** $n$ | Number of desired time steps. |
| **RFTS** $r$ | Reduction factor for tangential stiffness. This is used for modeling the stick condition due to friction in the penalty formulation of contact. |
| **SBRF** $n$ | Number of time steps between restart file generation. If zero, LS-NIKE3D writes a restart file as it terminates. |
| **SHIFT** $w$ | Shift frequency in hertz. This option works with the eigenvalue/eigenvector solution method. Using this option, NIKE will find the NEIG eigenvalues nearest to $w$. If the model has rigid body modes, a negative value for $w$ should be used to make the run stable. If $w$ is exactly the same value as an eigenvalue the system becomes singular. |
### LS-INGRID LS-NIKE3D COMMANDS AND MATERIALS

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSIT</td>
<td>$s$ Slide surface insertion tolerance</td>
</tr>
<tr>
<td>SSO</td>
<td>$u$ Step size option.</td>
</tr>
<tr>
<td></td>
<td>&quot;AUTO&quot;</td>
</tr>
<tr>
<td></td>
<td>&quot;MANUAL&quot;</td>
</tr>
<tr>
<td>SSOO</td>
<td>$n$ Optimal number of iterations per step.</td>
</tr>
<tr>
<td>TEO</td>
<td>$i$ Thermal effects option</td>
</tr>
<tr>
<td></td>
<td>$= 0$: no thermal effects.</td>
</tr>
<tr>
<td></td>
<td>$=N$: nodal temperatures are defined in input and are scaled according to a time function.</td>
</tr>
<tr>
<td></td>
<td>$N$ is the load curve number.</td>
</tr>
<tr>
<td></td>
<td>$=-1$: at each time step a new temperature state is read from a disk file. The time word at the beginning of each temperature state is ignored.</td>
</tr>
<tr>
<td></td>
<td>$=-2$: at each time step a temperature state is interpolated from the temperature state in a disk file. Therefore the time words at the beginning of each temperature state is used.</td>
</tr>
<tr>
<td></td>
<td>$=-3$: the disk file containing temperatures has only one state. The initial state is assumed to be zero.</td>
</tr>
<tr>
<td>TERM</td>
<td>$t$ Terminate dynamic time integration at time $t$.</td>
</tr>
<tr>
<td></td>
<td>The dynamic time step size will be computed if this command is used instead of the &quot;DELT&quot; command.</td>
</tr>
</tbody>
</table>

### 23.1 LS-NIKE3D Material Input

LS-NIKE3D material input is possible after the NK3D command has been input (see Control Commands). The form of this input is: MAT $n$ TYPE $m$ {options specific to material type $m$} {general material options} ENDMAT. $n$ is a material name which is assigned a number in the order that they occur in the input. Therefore, the materials should be defined in order before any additional use of materials is made.
Material Type 1 (Elastic)

Default heading: Material Type #1 (Elastic)

Input any two of the following.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BULK K</td>
<td>Bulk modulus.</td>
</tr>
<tr>
<td>E E</td>
<td>Young's modulus.</td>
</tr>
<tr>
<td>G G</td>
<td>Shear modulus.</td>
</tr>
<tr>
<td>PR n</td>
<td>Poisson's ratio.</td>
</tr>
</tbody>
</table>

Material Type 2 (Orthotropic Elastic)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>EA E_a</td>
<td>See matrix</td>
<td></td>
</tr>
<tr>
<td>EB E_b</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EC E_c</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRBA uba</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRCA uca</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRCB ucb</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAB G_ab</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GBC G_bc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GCA G_ca</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AOPT aopt</td>
<td>Material axes option (Figure 23-1).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>=0.0: locally orthotropic with materials axes by element nodes n_1, n_2, and n_4, (see Figure 23-1).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>=1.0: locally orthotropic with materials axes by a point in space and global location of element center.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>=2.0: globally orthotropic with materials axes determined by vectors defined below.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>=3.0: SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>XP x_p</td>
<td>Define for AOPT = 1.</td>
</tr>
<tr>
<td>YP y_p</td>
<td>Define for AOPT = 1.</td>
</tr>
<tr>
<td>ZP z_p</td>
<td>Define for AOPT = 1.</td>
</tr>
<tr>
<td>A1 a_1</td>
<td>Define for AOPT = 2.</td>
</tr>
<tr>
<td>A2 a_2</td>
<td>Define for AOPT = 2.</td>
</tr>
<tr>
<td>A3 a_3</td>
<td>Define for AOPT = 2.</td>
</tr>
<tr>
<td>D1 d_1</td>
<td>Define for AOPT = 2.</td>
</tr>
</tbody>
</table>
D2 \( d_2 \) Define for AOPT = 2.

D3 \( d_3 \) Define for AOPT = 2.

V1 \( v_1 \) Define for AOPT = 3.

V2 \( v_2 \) Define for AOPT = 3.

V3 \( v_3 \) Define for AOPT = 3.

The material law that relates stresses to strains is defined as:

\[
C = T^T C_L T,
\]

Where \( \omega(T, \cdot) \) is a transformation matrix, and \( \omega(C, \cdot)_L \) is the constitutive matrix defined in terms of the material constants of the orthogonal material axes, \( a, b, \) and \( c \). The inverse of \( \omega(C, \cdot)_L \) is defined as

\[
C^{-1} = \begin{bmatrix}
\frac{1}{E_a} & -\frac{v_{ba}}{E_b} & -\frac{v_{ca}}{E_c} & 0 & 0 & 0 \\
-\frac{v_{ab}}{E_a} & \frac{1}{E_b} & \frac{1}{E_c} & 0 & 0 & 0 \\
-\frac{v_{ac}}{E_a} & -\frac{v_{bc}}{E_b} & \frac{1}{E_c} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{G_{ab}} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1}{G_{bc}} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{G_{ca}}
\end{bmatrix}
\]

Note that \( \frac{v_{ab}}{E_a} = \frac{v_{ba}}{E_b}, \frac{v_{ca}}{E_c} = \frac{v_{ac}}{E_a}, \frac{v_{bc}}{E_c} = \frac{v_{bc}}{E_b}. \)
Figure 23-1. Options for determining principal materials axes: (a) AOPT = 0.0, (b) AOPT = 1.0, and (c) AOPT = 2.0.
Material Type 3 (Kinematic/Isotropic Elastic/Plastic)

Default heading: Material Type #3 (Elastic-Plastic)

Input any two of the following:

- **BULK K**  
  Bulk modulus
- **E E**  
  Young's modulus
- **G G**  
  Shear modulus
- **PR n**  
  Poisson's ratio

Additional Options:

- **SIGY s_y**  
  Yield stress
- **ETAN E_t**  
  Hardening modulus
- **BETA b'**  
  Hardening parameter, $0 \leq b' \leq 1$
- **NPTS n**  
  Number of points on stress-effective plastic strain curve.
- **ES s_y1 s_y2 ... s_yn**  
  Effective stress.
- **EPS e_p1 e_p2 ... e_pn**  
  Effective plastic strain.

Isotropic, kinematic, or a combination of isotropic and kinematic hardening may be specified by varying $b'$ between 0 and 1. For $b'$ equal to 0 and 1, respectively kinematic and isotropic hardening are obtained as shown in Figure 23-2. Effective stress is defined in terms of the deviatoric stress tensor, $S_{ij}$ as:

$$\bar{\sigma} = \left( \frac{3}{2} S_{ij} S_{ij} \right)^{\frac{1}{2}}$$

where,

$$S_{ij} = \sigma_{ij} - \frac{1}{3} \sigma_{kk} \delta_{ij}$$

and effective plastic strain by:

$$\bar{\varepsilon}^p = \int_0^t \bar{\varepsilon}^p$$

where $t$ denotes time and

$$d\bar{\varepsilon}^p = \left( \frac{2}{3} d\varepsilon_{ij}^p d\varepsilon_{ij}^p \right)^{\frac{1}{2}}$$
Figure 23-2. Elastic-plastic behavior with isotropic and kinematic hardening where \( l_0 \) and \( l \) are undeformed and deformed length of uniaxial tension specimen.
Material Type 4 (Thermo-Elastic-Plastic)

Default heading: Material Type 4 (Thermo-Elastic-Plastic)

NPTS \( n \)  
Number of temperature values for which material constants are defined.

TEMP \( T_1, T_2, \ldots, T_n \)  
Temperatures

\( E \) \( E_1, E_2, \ldots, E_n \)  
Young's moduli

\( PR \) \( u_1, u_2, \ldots, u_n \)  
Poisson's ratios

\( ALPHA \) \( a_1, a_2, \ldots, a_n \)  
Coefficients of thermal expansion.

\( SIGY \) \( s_y_1, s_y_2, \ldots, s_y_n \)  
Yield stresses

\( ETAN \) \( E_t_1, E_t_2, \ldots, E_t_n \)  
Tangent moduli

Material Type 5 (Soil and Crushable Foam)

Default heading: Material Type 5 (Soil and Crushable Foam)

Input any two of the following:

BULK \( K \)  
Bulk modulus

\( E \) \( E \)  
Young's modulus

\( G \) \( G \)  
Shear modulus

\( PR \) \( n \)  
Poisson's ratio

Additional Options:

AO \( a_0 \)  
Yield function constant.

A1 \( a_1 \)  
Yield function constant

A2 \( a_2 \)  
Yield function constant

PC \( P_c \)  
Pressure cutoff for tensile fracture

UL \( u_{opt} \)  
Unloading option

\( =0: \) volumetric crushing
\( =1: \) loading and unloading are the same.

NPTS \( n \)  
Number of points in volumetric strain versus pressure curve (\( n\leq10 \)).

VS \( e_{v_1}, e_{v_2}, \ldots, e_{v_n} \)  
Volumetric strain values

\( P \) \( p_1, p_2, \ldots, p_n \)  
Pressures corresponding to volumetric strain values

The deviatoric yield function, \( \phi \), is described in terms of the second invariant \( J_2 \).
\[ J_2 = \frac{1}{2} s_{ij} s_{ij} \]

Pressure, \( p \), and constants \( a_0, a_1, \) and \( a_2 \) as:

\[ \phi = J_2 - \left[ a_0 + a_1 p + a_2 p^2 \right]. \]

On the yield surface, \( J_2 = 1/3(s_y)^{1/2} \), where \( s_y \) is the yield stress, i.e.,

\[ \sigma_y = \left[ 3 \left( a_0 + a_1 p + a_2 p^2 \right) \right]^{1/2}. \]

For elastic-perfectly plastic behavior \( a_1 = a_2 = 0 \), and \( (3a_0)^{1/2} \) defines the yield strength. The volumetric strain is given by the natural logarithm of the relative volume \( V \). If the pressure drops below the cutoff value, \( PC \), then it is reset to that value.

Figure 23-3. Volumetric strain versus pressure curve for soil and crushable foam model.
Material Type 6 (Viscoelastic)

\[ G \quad G_0 \] Short term shear modulus.
\[ GI \quad G' \] Long term shear modulus.
\[ K \quad K \] Bulk modulus.
\[ \text{BETA} \quad \beta \quad \text{Decay constant.} \]

The shear relaxation behavior is described by:

\[ G(t) = G + (G_0 - G') e^{-\beta t} \]

A Jaumann rate formulation is used:

\[ \frac{\nabla}{t} \sigma_{ij} = 2 \int_0^t G(t-\tau) D_{ij}^r(\tau) d\tau \]

where the prime denotes the deviatoric part of the stress rate, \( \nabla \sigma_{ij} \), and the strain rate \( D_{ij} \).

Material Type 7 (Thermal Orthotropic Elastic)

Default heading: Material Type #7 (Thermal Orthotropic Elastic)

\[ EA \quad E_a \] See constitutive matrix below.
\[ EB \quad E_b \]
\[ EC \quad E_c \]
\[ PRBA \quad \nu_{ba} \]
\[ PRCA \quad \nu_{ca} \]
\[ PRCB \quad \nu_{cb} \]
\[ ALPA \quad \alpha_a \] Thermal expansion coefficient along material axis a.
\[ ALPB \quad \alpha_b \] Thermal expansion coefficient along material axis b.
\[ ALPC \quad \alpha_c \] Thermal expansion coefficient along material axis c.
\[ GAB \quad G_{ab} \]
\[ AOPT \quad aopt \quad \text{Material axes option (Figure 22-1).} \]

=0.0: locally orthotropic with materials axes determined by element nodes \( n_1, n_2, \) and \( n_4 \),
(see Figure 22-1).
=1.0: locally orthotropic with materials axes determined by a point in space and global location of element center.

=2.0: globally orthotropic with materials axes determined by $\psi_G$.

$\text{RP } r_p$ Define for AOPT = 1.

$\text{ZP } z_p$ Define for AOPT = 1.

$\text{PSIG } \psi_G$ In radians, define for AOPT = 2.

**Material Type 8 (Thermo-Elastic-Creep)**

Default heading: Material Type #8 (Thermo-Elastic-Creep)

- **NPTS** $n$ Number of temperature values for which material constants are defined.
- **TEMP** $T_1 \ T_2 ... T_n$ Temperatures
- **G** $G_1 \ G_2 ... G_n$ Shear moduli
- **K** $K_1 \ K_2 ... K_n$ Bulk moduli
- **ALPHA** $a_1 \ a_2 ... a_n$ Coefficients of thermal expansion
- **A** $a_1 \ a_2 ... a_n$ Creep parameters
- **B** $b_1 \ b_2 ... b_n$ Creep parameters

In this model, $G$, is the shear modulus and the instantaneous creep is given by a power law of the form

$$ a \mu \ t^b $$

where $a$ and $b$ are functions of temperature. This model was developed and provided for LS-NIKE3D by R. D. Krieg of Sandia National Laboratories.

**Material Type 9 (Blatz-Ko Rubber)**

Default heading: Material Type #9 (Rubber)

- **G** $\mu$ Shear modulus
The second Piola-Kirchhoff stress is computed as

\[ \sigma_{ij} = V^{-1} F_{ik} F_{jl} S_{kl} \]

where \( V \) is the relative volume, \( C_{ij} \) is the right Cauchy-Green strain tensor, and \( n \) is the Poisson's ratio which is set to .463 internally. This stress measure is transformed to the Cauchy stress, \( s_{ij} \), according to the relationship

\[ s_{ij} = F_{ij} \sigma_{ij} \]

where \( F_{ij} \) is the deformation gradient tensor.

**Material Type 10 (Power Law Thermo Plasticity)**

| NPTS n  | Number of temperature points (£8). |
| T T \_1 T \_2 \ldots T \_n | Temperatures |
| E E \_1 E \_2 \ldots E \_n | Young's moduli |
| PR u \_1 u \_2 \ldots u \_n | Poisson's ratios |
| K k \_1 \ldots k \_n | See equation below |
| M m \_1 \ldots m \_n | See equation below |

The stress-strain curve for this model is based on the following equation:

\[ \sigma_y = k(\varepsilon_e + \varepsilon^p)^n \]

**Material Type 11 (Compressible Mooney-Rivlin Rubber)**

This material model provides an alternative to the Blatz-Ko rubber model. The implementation is due to Maker [12].

| A A  | Constant A. |
| B B  | Constant B. |
| PR n | Poisson's ratio. |

The strain energy density function is defined as:

\[ W = A(I-3) + B(II-3) + C(III^{-2}-1) + D(III-1)^2 \]
where
\[ C = 0.5A + B. \]
\[ D = \frac{A(5V - 2) + B(11V - 5)}{2(1 - 2V)} \]

\( n = \text{Poisson's ratio.} \)

\( 2(A + B) = G = \text{shear modulus of linear elasticity.} \)

I, II, III are invariants of the right Cauchy-Green Tensor.

**Material Type 20 (Rigid Body)**

All elements with the same material number become a single rigid body if the material is type 20 whether the elements are connected or not. Density and two independent material strength constants are required to establish penalties for contact surfaces and joints.

Input any two of the following:

- **BULK K** Bulk modulus.
- **E E** Young's modulus.
- **G G** Shear modulus.
- **PR n** Poisson's ratio.

Additional Options:

- **LC lc** Load curve number for displacement control.
- **SCALE scale** Scale factor.
- **VVEC v_x v_y v_z** Define the vector \( \mathbf{v} \) for the direction cosines.

**Material Type 23 (Thermal Orthotropic with Curves)**

- **NPTS npts** Number of points (1 < NPTS < 50).
- **EA (E_a)_1 ... (E_a)_n**
- **EB (E_b)_1 ... (E_b)_n**
- **EC (E_c)_1 ... (E_c)_n**
- **PRBA (n_{ba})_1 ... (n_{ba})_n**
- **PRCA (n_{ca})_1 ... (n_{ca})_n**
PRCB \((n_{cb})_1 \ldots (n_{cb})_n\)

AA \((a_a)_1 \ldots (a_a)_n\)

AB \((a_b)_1 \ldots (a_b)_n\)

AC \((a_c)_1 \ldots (a_c)_n\)

GAB \((G_{ab})_1 \ldots (G_{ab})_n\)

GBC \((G_{bc})_1 \ldots (G_{bc})_n\)

GCA \((G_{ca})_1 \ldots (G_{ca})_n\)

\textbf{AOPT} \(aopt\)

Material axes option (Figure 23-1).

\(=0.0\): locally orthotropic with materials axes determined by element nodes \(n_1, n_2,\) and \(n_4\), (see Figure 23-1).

\(=1.0\): locally orthotropic with materials axes determined by a point in space and global location of element center.

\(=2.0\): globally orthotropic with materials axes determined by vectors defined below.

\(=3.0\): SHELL ELEMENTS ONLY: The material axis is locally orthotropic with material axes determined by a vector in the plane of the shell and the shell normal.

\textbf{XP} \(x_p\)

Define for \(AOPT = 1\).

\textbf{YP} \(y_p\)

Define for \(AOPT = 1\).

\textbf{ZP} \(z_p\)

Define for \(AOPT = 1\).

\textbf{A1} \(a_1\)

Define for \(AOPT = 2\).

\textbf{A2} \(a_2\)

Define for \(AOPT = 2\).

\textbf{A3} \(a_3\)

Define for \(AOPT = 2\).

\textbf{D1} \(d_1\)

Define for \(AOPT = 2\).

\textbf{D2} \(d_2\)

Define for \(AOPT = 2\).

\textbf{D3} \(d_3\)

Define for \(AOPT = 2\).

\textbf{V1} \(v_1\)

Define for \(AOPT = 3\).

\textbf{V2} \(v_2\)

Define for \(AOPT = 3\).

\textbf{V3} \(v_3\)

Define for \(AOPT = 3\).
### 24. TOPAZ Commands and Materials

Analysis options are code dependent. They can be set either in the control section of the LS-INGRID input file or in the interactive phase. These commands become active when TOPAZ2D or TOPAZ3D output is selected with the TZ2D or TZ3D commands respectively.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| BWMO    | Bandwidth minimization option.  
|         | -="ON": minimize bandwidth (default).  
|         | -="OFF": don't minimize bandwidth. |
| DCMX    | Desired maximum temperature change in each time step above which the time step will decrease. |
| DCTOL   | Convergence tolerance for equilibrium iterations (default = 0.0001). |
| DELT    | Time step size for fixed time step and initial time step for variable time step. |
| DTMAX   | Maximum time step size. |
| DTMIN   | Minimum time step size. |
| FLUX    | Nodal heat flux calculations  
|         | -="ON": perform calculations  
|         | -="OFF": don't perform calculations (default). |
| IPLT    | Number of time steps between output of graphics database. |
| IPRT    | Number of time steps between output printouts. |
| IUNIT   | Temperature units  
|         | -="DIME": dimensionless  
|         | -="CENT": centigrade  
|         | -="FAHR": fahrenheit  
|         | -="KELV": Kelvin  
<p>|         | -=&quot;RANK&quot;: Rankine |
| LINEAR  | Problem is linear. |
| MFTS    | Modification factor for increasing/decreasing time step. |
| MRDI    | Maximum number of radiosity iterations. |
| MSRF    | Maximum number of conductance matrix reformations per time step (default = 10). |</p>
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NBEI $n$</td>
<td>The number of time steps between equilibrium iterations (default = 1).</td>
</tr>
<tr>
<td>NBSR $n$</td>
<td>The number of time steps between conductance matrix reformation (default = 1).</td>
</tr>
<tr>
<td>NIBSR $n$</td>
<td>Maximum number of equilibrium iterations permitted per conductance matrix reformation.</td>
</tr>
<tr>
<td>NIP1 $x$</td>
<td>First Newmark integration parameter. (default = 0.5). $1.0$: fully implicit</td>
</tr>
<tr>
<td>NONLINEAR</td>
<td>Problem is non-linear.</td>
</tr>
<tr>
<td>NSSD $n$</td>
<td>Number of surface subdivision for radiation view factor calculation (default = 5).</td>
</tr>
</tbody>
</table>
| PHASE $n$ | Phase change flag  
  $"ON"$: perform phase change calculation  
  $"OFF"$: no phase change calculation (default). |
| RADI $n$ | Radiation calculation type  
  $"VIEW"$: view factors  
  $"EXCH"$: exchange factors |
| RBAND $m$ $n$ | Radiation bands. The number of wavelength break points is $m$ and the number of curves is $n$.  
  $l_1$ $l_2$...$l_m$  
  $(E_1)_l$ $(E_2)_l$ ...$(E_m)_l$  
  Emissivities for curve $l$  
  $(E_1)_n$ $(E_2)_n$ ...$(E_m)_n$  
  Emissivities for curve $n$ |
| RCTOL $s$ | Radiosity convergence tolerance. |
| RELAX $r$ | Relaxation parameter (default = 1). |
| SBC $s$ | Stefan-Boltzmann constant. |
| SBRF $n$ | Number of time steps between restart dumps. |
| SOLUTION STEADY | Analysis is steady state. |
| SOLUTION TRANSIENT | Analysis is transient. |
| STEP $n$ | Time step code  
  $"FIXE"$: fixed time step  
  $"VARI"$: variable time step. |
| TERM $t$ | Final problem time. |
24.1 TOPAZ Material Input

TOPAZ material input is possible after the TZ2D or TZ3D command has been input (see Control Commands). The form of this input is: MAT n TYPE m {options specific to material type m} {general material options} ENDMAT. \( n \) is a material name which is assigned a number in the order that they occur in the input. Therefore, the materials should be defined in order before any additional use of materials is made.

Material Type #1 (Isotropic)

- **RO r** Density
- **TLHA t** Temperature at which latent heat is absorbed or released.
- **LH h** Latent heat.
- **TGC lc** Thermal generation rate curve number.
- **TGM r** Thermal generation rate multiplier.
- **HCP c** Heat capacity.
- **K k** Thermal conductivity
- **ENDMAT** End this material model.

Material Type 2 (Orthotropic)

- **RO r** Density
- **TLHA t** Temperature at which latent heat is absorbed or released.
- **LH h** Latent heat.
- **TGC lc** Thermal generation rate curve number.
- **TGM r** Thermal generation rate multiplier.
- **HCP c** Heat capacity.
- **K1 k1** Thermal conductivity in local 1 direction.
- **K2 k2** Thermal conductivity in local 2 direction.
- **K3 k3** Thermal conductivity in local 3 direction.
- **ENDMAT** End this material model.
**Material Type 3 (Isotropic Temperature Dependent)**

- **RO r**  
  Density
- **TLHA t**  
  Temperature at which latent heat is absorbed or released.
- **LH h**  
  Latent heat.
- **TGC lc**  
  Thermal generation rate curve number.
- **TGM r**  
  Thermal generation rate multiplier.
- **NPTS n**  
  Number of temperature points.
- **TEMP T_1 T_2...T_n**  
  Temperatures.
- **CP C_1 C_2...C_n**  
  Heat capacities
- **K K_1 K_2...K_n**  
  Thermal conductivities.
- **ENDMAT**  
  End this material model.

**Material Type 4 (Orthotropic Temperature Dependent)**

- **RO r**  
  Density
- **TLHA t**  
  Temperature at which latent heat is absorbed or released.
- **LH h**  
  Latent heat.
- **TGC lc**  
  Thermal generation rate curve number.
- **TGM r**  
  Thermal generation rate multiplier.
- **NPTS n**  
  Number of temperature points.
- **TEMP T_1 T_2...T_n**  
  Temperatures.
- **CP C_1 C_2...C_n**  
  Heat capacities
- **K1 K_1 (K_1)_2...(K_1)_n**  
  Thermal conductivities in local 1 direction.
- **K2 K_2 (K_2)_2...(K_2)_n**  
  Thermal conductivities in local 2 direction.
- **K3 K_3 (K_3)_2...(K_3)_n**  
  Thermal conductivities in local 3 direction.
- **ENDMAT**  
  End this material model.
ACKNOWLEDGMENTS

Any work of this magnitude obviously was influenced by a large number of people who cannot possibly be given proper credit. The authors very much appreciate the all of the inputs whether positive or hostile which have aided in this work. The original work of Bill Cook on INGEN influenced LS-INGRID considerably. Special thanks must be given to Russ Rosinsky for his patience in finding bugs, recommending new capabilities and proof reading this manual. Steve Sackett, Greg Kay, and Tracy Glover also helped in providing sample problems, new ideas, and uncovering bugs.

Developments at SPARTA benefited from David Lichtblau's and Brian Wainscott's work on the calculator program. Sophie Tsui and Dawn Greayer made contributions to the materials processing portion and Bill Campbell provided some useful ideas for surface intersection algorithms. Sharon Kiefer made some important contributions in debugging and improving the manufacturing capabilities. The NURB curve and surface algorithms were developed by Alan Winslow.

Eunice Hinkle, Nikki Falco, Valli A. James and Debbie Aiken all suffered through the preparation of various versions of this manual. The University of Tennessee, Lawrence Livermore National Laboratory and SPARTA, Inc. all generously provided computer resources to allow LS-INGRID to be developed.
REFERENCES

Index

Acceleration Boundary Condition
Applying 95
Defining 95
Displaying 107

Advection Formulation
DYNA3D 169

Airbag
DYNA3D Output 170
Folding 18, 20, 111, 113
Inflator Model 13
Penetration Check 113
Sealing 114

Single Surface Contact Algorithm 31

ALE Formulation
DYNA3D 170

ALE Smoothing
Applying 101

Analytical Contact 20
Angular Velocity
Initial 29, 68

Anisotropic Plasticity 226
Arbitrary Numbering
DYNA3D 167

Arrival Time
Pressure Load 11
Automatic Time Step
NIKE2D 255
TOPAZ 295

AVS
DYNA3D 170

Axisymmetry
NIKE2D 253

Bandwidth Minimization
NIKE2D 253
NIKE3D 273
TOPAZ 294

Batch Operation 12, 26

BCIZ Shell
DYNA3D 179

Beam
Area 177
Element Generation 63
Formulation 177
Moment of Area 178
Quadrature 178

Thickness 179

Belytschko-Lin-Tsay Shell
DYNA3D 179

Belytschko-Schwer Beam
DYNA3D 177, 179

Blatz-Ko Rubber
DYNA3D 191
NIKE2D 267

Boltzmann Constant
TOPAZ 295

Boundary Condition
Constraining 95
Displaying 109

Brick Formulation
DYNA3D 177

Brode Function
DYNA3D 167

Bulk Viscosity
DYNA3D 177

C0 Triangle
DYNA3D 179

CAL3D 20, 179

Repositioning Materials 35

Unit Conversions In Coupling 168

Cap Model 214

Cardan Joint
Defining 19

Circular Arc
Standard Part 60, 61, 62

Coincident Node Removal 107

Commands
ACE 62
ACCE 95
AA 208, 211, 292
AB 208, 211, 292
ABSTAT 170
ACC 95
AA 208, 211, 292
AB 208, 211, 292
AC 62, 208, 211, 292
ACCE 13, 95
ACCI 95
ADD 24
ADVECTION 169
AE 61, 248
AJNP 107
ALAB 117
ALE 170
ALPHA 265
ALPB 265
ALPC 265
ALPH 209, 231, 234
ALPHA 148, 163, 187,
214, 228, 229, 245, 261,
266, 283, 287
AM 107
AMN 107
ANAL 253, 273
ANGL 20
ANGLES 177
AO 149, 188, 262, 284
AOPT 144, 182, 208, 210,
212, 216, 224, 227, 232,
235, 243, 262, 278, 284,
293
AP 107, 248
AP23 15
APRI 11
ARRI 107
ASCII 107
ASCII 107
ASCL 117
ASET 117
ASYM 28
atan 8
atan2 8
AVG 125
AVENG 174
AVSFLT 170
AVSFLT 170
AXIS 16
B0 149, 155, 162, 182,
188, 194, 203, 208, 210,
212, 216, 224, 227, 232,
235, 243, 262, 278, 284,

BRICK 177

C0 Triangle
DYNA3D 179

CAL3D 20, 179

Repositioning Materials 35

Unit Conversions In Coupling 168

Cap Model 214

Cardan Joint
Defining 19

Circular Arc
Standard Part 60, 61, 62

Coincident Node Removal 107

Commands
ACE 62
ACCE 95
AA 208, 211, 292
AB 208, 211, 292
ABSTAT 170
ACC 95
AA 208, 211, 292
AB 208, 211, 292
AC 62, 208, 211, 292
ACCE 13, 95
ACCI 95
ADD 24
ADVECTION 169
AE 61, 248
AJNP 107
ALAB 117
ALE 170
ALPHA 265
ALPB 265
ALPC 265
ALPH 209, 231, 234
ALPHA 148, 163, 187,
214, 228, 229, 245, 261,
266, 283, 287
AM 107
AMN 107
ANAL 253, 273
ANGL 20
ANGLES 177
AO 149, 188, 262, 284
AOPT 144, 182, 208, 210,
212, 216, 224, 227, 232,
235, 243, 262, 278, 284,
293
AP 107, 248
AP23 15
APRI 11
ARRI 107
ASCII 107
ASCII 107
ASCL 117
ASET 117
ASYM 28
atan 8
atan2 8
AVG 125
AVENG 174
AVSFLT 170
AVSFLT 170
AXIS 16
B0 149, 155, 162, 182,
188, 194, 203, 208, 210,
212, 216, 224, 227, 232,
235, 243, 262, 278, 284,
<table>
<thead>
<tr>
<th>INDEX</th>
<th>LS-INGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>B 95, 96, 100, 103, 157, 159, 196, 201, 217, 242, 266, 287, 290</td>
<td>227, 232, 235, 278, 293</td>
</tr>
<tr>
<td>BI 63, 162, 203, 243</td>
<td>D2R 168</td>
</tr>
<tr>
<td>B2 64, 243</td>
<td>D3 159, 182, 201, 208, 278, 293</td>
</tr>
<tr>
<td>BATCH 12</td>
<td>D4 159, 201</td>
</tr>
<tr>
<td>BCND 78</td>
<td>D5 159, 201</td>
</tr>
<tr>
<td>BCP 78</td>
<td>DAMP 13, 31</td>
</tr>
<tr>
<td>B1 63, 162, 203, 243</td>
<td>DBQT 139, 168</td>
</tr>
<tr>
<td>BE 248</td>
<td>DCMX 294</td>
</tr>
<tr>
<td>BEAM 73, 177</td>
<td>DCTOL 253, 273, 294</td>
</tr>
<tr>
<td>BEAMS 79</td>
<td>DEATH 31</td>
</tr>
<tr>
<td>BELT 12, 96, 170</td>
<td>DEBUG 168</td>
</tr>
<tr>
<td>BETA 146, 151, 157, 163, 185, 190, 196, 214, 225, 233, 245, 259, 264, 281, 286</td>
<td>DEFAUL 16</td>
</tr>
<tr>
<td>BFORM 177</td>
<td>DEFG 207</td>
</tr>
<tr>
<td>BG 63</td>
<td>DEFGEO 170</td>
</tr>
<tr>
<td>BIAS 64</td>
<td>DEFL 207</td>
</tr>
<tr>
<td>BIRTH 31</td>
<td>DEFORO 170</td>
</tr>
<tr>
<td>BLEN 125</td>
<td>DIEL 12, 13</td>
</tr>
<tr>
<td>BLND 129</td>
<td>DELT 168, 253, 273, 294</td>
</tr>
<tr>
<td>BOND 31</td>
<td>DETHP 16</td>
</tr>
<tr>
<td>BOXM 31</td>
<td>DI 65</td>
</tr>
<tr>
<td>BOXS 31</td>
<td>DI ACCE 107</td>
</tr>
<tr>
<td>BP 157, 196, 248</td>
<td>DI ACG 107</td>
</tr>
<tr>
<td>BPTOL 107</td>
<td>DI BELT 107</td>
</tr>
<tr>
<td>BQL 142, 177</td>
<td>DI CV 108</td>
</tr>
<tr>
<td>BQK 142, 177</td>
<td>DI CYL 108</td>
</tr>
<tr>
<td>BQT 142, 177</td>
<td>DI D 108</td>
</tr>
<tr>
<td>BRFORM 177</td>
<td>DI DETP 108</td>
</tr>
<tr>
<td>BRICKS 81</td>
<td>DI DS 108</td>
</tr>
<tr>
<td>BRODE 139, 167</td>
<td>DI DSMR 108</td>
</tr>
<tr>
<td>BRUL 167</td>
<td>DI DX 108</td>
</tr>
<tr>
<td>BUFD 168</td>
<td>DI DZ 108</td>
</tr>
<tr>
<td>BWMO 253, 273, 294</td>
<td>DI EDR 108</td>
</tr>
<tr>
<td>C001 221, 270</td>
<td>DI EPB 108</td>
</tr>
<tr>
<td>C001 221, 270</td>
<td>DI F 108</td>
</tr>
<tr>
<td>C010 221, 270</td>
<td>DI FL 108</td>
</tr>
<tr>
<td>C020 221, 270</td>
<td>DI FLUX 108</td>
</tr>
<tr>
<td>C020 221, 270</td>
<td>DI FSYM 108</td>
</tr>
<tr>
<td>C1 241, 247</td>
<td>DI INTF 108</td>
</tr>
<tr>
<td>C1 241, 247</td>
<td>DI JOY 108</td>
</tr>
<tr>
<td>C100 221, 270</td>
<td>DI JTS 108</td>
</tr>
<tr>
<td>C100 221, 270</td>
<td>DI L3D 108</td>
</tr>
<tr>
<td>C101 221, 270</td>
<td>DI LAX 108</td>
</tr>
<tr>
<td>C101 221, 270</td>
<td>DI M 108</td>
</tr>
<tr>
<td>C110 221, 270</td>
<td>DI MCG 109</td>
</tr>
<tr>
<td>C12 241, 247</td>
<td>DI MK 109</td>
</tr>
<tr>
<td>C200 221, 270</td>
<td>DI NCV 109</td>
</tr>
<tr>
<td>C210 221, 270</td>
<td>DI NFG 109</td>
</tr>
<tr>
<td>C23 15</td>
<td>DI NPB 109</td>
</tr>
<tr>
<td>C3 241, 247</td>
<td>DI NR 109</td>
</tr>
<tr>
<td>C300 221, 270</td>
<td>DI NSF 109</td>
</tr>
<tr>
<td>C4 241, 247</td>
<td>DI NS 109</td>
</tr>
<tr>
<td>C400 221, 270</td>
<td>DI NV 109</td>
</tr>
<tr>
<td>C5 241, 247</td>
<td>DI ORV 109</td>
</tr>
<tr>
<td>C6 241, 247</td>
<td>DI P 109</td>
</tr>
<tr>
<td>CARDAN 19</td>
<td>DI PL 109</td>
</tr>
<tr>
<td>CAREH 177</td>
<td>DI PM 109</td>
</tr>
<tr>
<td>LS-INGRID</td>
<td>INDEX</td>
</tr>
<tr>
<td>------------</td>
<td>-------</td>
</tr>
<tr>
<td>DI PR 109</td>
<td>199, 204, 205, 206, 207, 261, 281, 283</td>
</tr>
<tr>
<td>DI PV 109</td>
<td>213, 215, 218, 219, 220, 281, 283, 284, 285</td>
</tr>
<tr>
<td>DI RB 109</td>
<td>222, 225, 226, 228, 229, 230, 233, 236, 237, 256, 261, 281, 283, 284, 285</td>
</tr>
<tr>
<td>DI RBN 109</td>
<td>259, 261, 262, 268, 269, 277, 281, 283, 284, 285</td>
</tr>
<tr>
<td>DI RE 109</td>
<td>289, 291, F 157, 196</td>
</tr>
<tr>
<td>DI REL 109</td>
<td>289, 291, F 157, 196</td>
</tr>
<tr>
<td>DI RX 109</td>
<td>E0 241, 242, 243, 244, ETG 222</td>
</tr>
<tr>
<td>DI RXN 109</td>
<td>245, 247, 248, 250, 252, EQU 170</td>
</tr>
<tr>
<td>DI RY 109</td>
<td>EA 144, 182, 208, 209, EQU 170</td>
</tr>
<tr>
<td>DI RZ 109</td>
<td>211, 223, 231, 234, 257, ECRV 248</td>
</tr>
<tr>
<td>DI SBI 109</td>
<td>265, 278, 292, FCRV 206</td>
</tr>
<tr>
<td>DI SFC 109</td>
<td>EAAU 215, FE 31</td>
</tr>
<tr>
<td>DI SI 110</td>
<td>EB 144, 182, 208, 209, FDI 97</td>
</tr>
<tr>
<td>DI SL 110</td>
<td>211, 223, 231, 234, 257, FDI 97</td>
</tr>
<tr>
<td>DI SP 110</td>
<td>265, 278, 292, FDI 97</td>
</tr>
<tr>
<td>DI SPD 110</td>
<td>EBBU 215, FE 31</td>
</tr>
<tr>
<td>DI SW 110</td>
<td>EC 144, 182, 208, 209, FDL 12</td>
</tr>
<tr>
<td>DI SY 110</td>
<td>211, 223, 231, 234, 257, FDL 12</td>
</tr>
<tr>
<td>DI SPSI 110</td>
<td>265, 278, 292, FDL 12</td>
</tr>
<tr>
<td>DI TB 110</td>
<td>EC1 157, 196, FE 31</td>
</tr>
<tr>
<td>DI TI 110</td>
<td>EC2 157, 196, FE 31</td>
</tr>
<tr>
<td>DI TRACER 110</td>
<td>EC3 157, 196, FE 31</td>
</tr>
<tr>
<td>DI VB 110</td>
<td>EC4 157, 196, FE 31</td>
</tr>
<tr>
<td>DI VECT 110</td>
<td>EC5 157, 196, FE 31</td>
</tr>
<tr>
<td>DI WARP 110</td>
<td>EC6 157, 196, FE 31</td>
</tr>
<tr>
<td>DI AD 110</td>
<td>EC7 157, 196, FE 31</td>
</tr>
<tr>
<td>DI COL 110</td>
<td>EC8 157, 196, FE 31</td>
</tr>
<tr>
<td>DI OFF 110</td>
<td>EC9 157, 196, FE 31</td>
</tr>
<tr>
<td>DIST 13</td>
<td>ECCU 215, FE 31</td>
</tr>
<tr>
<td>DISTANCE 13</td>
<td>ECHO 168, FE 31</td>
</tr>
<tr>
<td>DM 110</td>
<td>ECO 157, 196, FN 97</td>
</tr>
<tr>
<td>DMAX 13</td>
<td>ECRV 206, FN 97</td>
</tr>
<tr>
<td>DMEM 111</td>
<td>ECTOL 253, 273, FN 97</td>
</tr>
<tr>
<td>DMIN 13</td>
<td>EDR 97, FEN 17</td>
</tr>
<tr>
<td>DMN 111</td>
<td>EH 155, 194, 198, 199, FN 97</td>
</tr>
<tr>
<td>DN2D 16</td>
<td>204, FNL 17</td>
</tr>
<tr>
<td>DN3D 16</td>
<td>EI 237, FNL 17</td>
</tr>
<tr>
<td>DNIS 31</td>
<td>ELEMENT 23, FNL 17</td>
</tr>
<tr>
<td>DNTS 31</td>
<td>ELLIPSE 179, 207, FNL 17</td>
</tr>
<tr>
<td>DQL 140, 169</td>
<td>ELOUT 170, FNL 17</td>
</tr>
<tr>
<td>DQQ 140, 169</td>
<td>ELPLT 111, FNL 17</td>
</tr>
<tr>
<td>DRAG 87</td>
<td>EMAX 221, FNL 17</td>
</tr>
<tr>
<td>DRAW 111</td>
<td>EMIN 221, FNL 17</td>
</tr>
<tr>
<td>DRDB 174</td>
<td>END 18, 41, 43, 45, FNL 17</td>
</tr>
<tr>
<td>DRFCR 169</td>
<td>ENDEOS 241, 242, 243, FNL 17</td>
</tr>
<tr>
<td>DROPTS 169</td>
<td>244, 245, 247, 250, FNL 17</td>
</tr>
<tr>
<td>DRTERM 169</td>
<td>252, FNL 17</td>
</tr>
<tr>
<td>DRTO 169</td>
<td>ENDMAT 297, 298, 299, FNL 17</td>
</tr>
<tr>
<td>DSV 16, 133</td>
<td>300, FNL 17</td>
</tr>
<tr>
<td>DSAD 111</td>
<td>ENER 173, FNL 17</td>
</tr>
<tr>
<td>DSF 117</td>
<td>ENERGY 118, FNL 17</td>
</tr>
<tr>
<td>DSRM 111</td>
<td>EOS 142, 177, FNL 17</td>
</tr>
<tr>
<td>DSV 111</td>
<td>EPB 97, FNL 17</td>
</tr>
<tr>
<td>DSVS 111</td>
<td>EPS 155, 162, 194, 203, FNL 17</td>
</tr>
<tr>
<td>DT MAX 253, 273, 294</td>
<td>213, 259, 281, FNL 17</td>
</tr>
<tr>
<td>DT MIN 294</td>
<td>EPSO 159, 201, FNL 17</td>
</tr>
<tr>
<td>DTM 253, 273</td>
<td>EQSP 65, FNL 17</td>
</tr>
<tr>
<td>DTS 117</td>
<td>ER 129, 162, 203, FNL 17</td>
</tr>
<tr>
<td>DUMMY 31, 78, 79, 80, 81</td>
<td>ES 155, 162, 194, 203, FNL 17</td>
</tr>
<tr>
<td>DYNA3D 93</td>
<td>213, 259, 281, FNL 17</td>
</tr>
<tr>
<td>E 143, 146, 148, 149, 161, 181, 185, 187, 188, 198, 261, 281, 283</td>
<td>FNL 17</td>
</tr>
<tr>
<td></td>
<td>261, 281, 283, FNL 17</td>
</tr>
<tr>
<td></td>
<td>261, 281, 283, FNL 17</td>
</tr>
<tr>
<td>INDEX</td>
<td>LS-INGRID</td>
</tr>
<tr>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>GAB 144, 182, 208, 209, 211, 223, 231, 234, 257, 265, 278, 292</td>
<td>INT 175</td>
</tr>
<tr>
<td>GABU 215</td>
<td>INT 175</td>
</tr>
<tr>
<td>GAM0 230</td>
<td>INTERNAL 118</td>
</tr>
<tr>
<td>GAMA 157, 196</td>
<td>IOPT 222</td>
</tr>
<tr>
<td>GAMM 15</td>
<td>IPLT 253, 273, 294</td>
</tr>
<tr>
<td>GAMMA 157, 196</td>
<td>IPRT 294</td>
</tr>
<tr>
<td>GAMU 215</td>
<td>IRDM 171</td>
</tr>
<tr>
<td>GBC 182, 208, 209, 211, 223, 231, 234, 278, 292</td>
<td>IRR 178</td>
</tr>
<tr>
<td>GBCU 215</td>
<td>IRULE-TRAPEZOIDAL</td>
</tr>
<tr>
<td>GCA 182, 208, 209, 211, 223, 231, 234, 278, 292</td>
<td>178</td>
</tr>
<tr>
<td>GCAU 215</td>
<td>IRULE-USER 178</td>
</tr>
<tr>
<td>gcd 8</td>
<td>IRULE.GAUSS 178</td>
</tr>
<tr>
<td>GEFORO 170</td>
<td>IS 98</td>
</tr>
<tr>
<td>GELN 129</td>
<td>ISI 98</td>
</tr>
<tr>
<td>GELS 130</td>
<td>ISS 178</td>
</tr>
<tr>
<td>GEOM 20, 98</td>
<td>IT 159, 201</td>
</tr>
<tr>
<td>GEOM 140, 253</td>
<td>ITT 178</td>
</tr>
<tr>
<td>GFUN 139, 167</td>
<td>IUNIT 294</td>
</tr>
<tr>
<td>GI 151, 190, 264, 286</td>
<td>JOINTS 170</td>
</tr>
<tr>
<td>GLSTAT 170</td>
<td>JOY 98</td>
</tr>
<tr>
<td>GMI 21</td>
<td>JOYI 98</td>
</tr>
<tr>
<td>GMPRT 170</td>
<td>JOY 98</td>
</tr>
<tr>
<td>GN 34</td>
<td>JOYI 98</td>
</tr>
<tr>
<td>GRAV 14, 15, 140, 171, 253, 273</td>
<td>K 60</td>
</tr>
<tr>
<td>GRID 17, 111</td>
<td>K 60, 65, 78, 79, 80, 81</td>
</tr>
<tr>
<td>GSI 130</td>
<td>151, 161, 163, 190, 205, 209, 214, 231, 234, 237, 264, 266, 268, 269, 286</td>
</tr>
<tr>
<td>GS1 130</td>
<td>287, 288, 289, 297, 299</td>
</tr>
<tr>
<td>GS2 130</td>
<td>K1 298, 300</td>
</tr>
<tr>
<td>GSM 130</td>
<td>K2 298, 300</td>
</tr>
<tr>
<td>GSN 130</td>
<td>K3 298, 300</td>
</tr>
<tr>
<td>GSTIF 253, 273</td>
<td>KAPPA 228</td>
</tr>
<tr>
<td>GTIME 117</td>
<td>KI 60</td>
</tr>
<tr>
<td>H 157, 196, 248</td>
<td>KINETIC 118</td>
</tr>
<tr>
<td>HC 228, 229</td>
<td>L 111, 137</td>
</tr>
<tr>
<td>HCP 159, 201, 297, 298</td>
<td>L2D 125</td>
</tr>
<tr>
<td>HDMG 32</td>
<td>L3 130</td>
</tr>
<tr>
<td>HEAD 142, 177</td>
<td>L3D 17, 20, 23, 125</td>
</tr>
<tr>
<td>HEIGHT 139, 167</td>
<td>L3E 17</td>
</tr>
<tr>
<td>HENERGY 171</td>
<td>L3P 130</td>
</tr>
<tr>
<td>HGQ 142, 177</td>
<td>L3R 130</td>
</tr>
<tr>
<td>HGQT 142, 177</td>
<td>L3S 130</td>
</tr>
<tr>
<td>H 59, 65, 248</td>
<td>L3V 111</td>
</tr>
<tr>
<td>HARB 171</td>
<td>L3VS 111</td>
</tr>
<tr>
<td>IDEA 41</td>
<td>LABELS 23</td>
</tr>
<tr>
<td>IEP 34</td>
<td>LADD 121, 125</td>
</tr>
<tr>
<td>IFDT 174</td>
<td>LADV 121</td>
</tr>
<tr>
<td>IJ 60</td>
<td>LAGRANGIAN 170</td>
</tr>
<tr>
<td>IJK 60</td>
<td>LAM 121</td>
</tr>
<tr>
<td>IKEDIT 168</td>
<td>LAR 121</td>
</tr>
<tr>
<td>IMGL 142, 178</td>
<td>LAT 121</td>
</tr>
<tr>
<td>IN 98</td>
<td>LBCV 125</td>
</tr>
<tr>
<td>INC 98, 100, 103</td>
<td>LBCX 125</td>
</tr>
<tr>
<td>INCLUDE 21, 78, 79, 80, 81</td>
<td>LBCY 125</td>
</tr>
<tr>
<td>INFO 111</td>
<td>LBCZ 125</td>
</tr>
<tr>
<td>INI 98</td>
<td>LC 15, 233, 268, 291</td>
</tr>
<tr>
<td>INSIDE 20</td>
<td>LC 19, 20</td>
</tr>
<tr>
<td>INT 65</td>
<td>LC 2 19</td>
</tr>
<tr>
<td>INT4 175</td>
<td>LC 3 19</td>
</tr>
<tr>
<td>INT8 175</td>
<td>LC4 19</td>
</tr>
<tr>
<td>INTERNAL 118</td>
<td>LC5 19</td>
</tr>
<tr>
<td>IOPT 222</td>
<td>LC6 19</td>
</tr>
<tr>
<td>IPLT 253, 273, 294</td>
<td>LCA 215</td>
</tr>
<tr>
<td>IPRT 294</td>
<td>LCAB 215</td>
</tr>
<tr>
<td>IRDM 171</td>
<td>LCB 215</td>
</tr>
<tr>
<td>IRR 178</td>
<td>LCC 121, 215</td>
</tr>
<tr>
<td>IRULE-TRAPEZOIDAL</td>
<td>LCCA 215</td>
</tr>
<tr>
<td>178</td>
<td>LCD 23, 27</td>
</tr>
<tr>
<td>IRULE-USER 178</td>
<td>LCDAMP 171</td>
</tr>
<tr>
<td>IRULE.GAUSS 178</td>
<td>LCDF 23</td>
</tr>
<tr>
<td>IS 98</td>
<td>LCGX 171</td>
</tr>
<tr>
<td>ISI 98</td>
<td>LCGY 171</td>
</tr>
<tr>
<td>ISS 178</td>
<td>LCGZ 171</td>
</tr>
<tr>
<td>IT 159, 201</td>
<td>LCK 237</td>
</tr>
<tr>
<td>ITT 178</td>
<td>LCL 12, 238</td>
</tr>
<tr>
<td>IUNIT 294</td>
<td>lcm 8, 14, 15, 237</td>
</tr>
<tr>
<td>JOINTS 170</td>
<td>LCMAX 171</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LCPOUTF 15</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LCP 12, 162, 203</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LCR 162, 203</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LCRX 171</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LCRY 171</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LCRZ 171</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LCSS 226</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LCU 12, 238</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LCUT 126</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LCV 27, 32, 111</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LD 23, 121</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LE 34</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LEP 122</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LEV 24</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LEXP 122, 126</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LFOR 126</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LH 297, 298, 299, 300</td>
</tr>
<tr>
<td>JOY 98</td>
<td>LMIN 111</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>IN 8</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>ln 8</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>ln10 8</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>ln2 8</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LNPT 16, 38</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LNV 250, 252</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LO 122</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LOCK 13</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LOD 122</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LORI 65</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LP 122, 126</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPIL 122</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPN 126</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPRJ 126</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPS1 219</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPS2 219</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPT 122</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPT1 219</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPT2 219</td>
</tr>
<tr>
<td>JOYI 98</td>
<td>LPTA 212</td>
</tr>
</tbody>
</table>
| JOYI 98 | LREIP 41, 43, 46, 65, 75,
<table>
<thead>
<tr>
<th>LS-INGRID</th>
<th>INDEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>82, 87, 92, 94</td>
<td>MK+ 99, NFG 25, 99, 170</td>
</tr>
<tr>
<td>LREV 126</td>
<td>MK- 99, NFGI 99</td>
</tr>
<tr>
<td>LRL 122</td>
<td>MKDS 25, NGEN 63</td>
</tr>
<tr>
<td>LRNV 127</td>
<td>MKI 99, NI+ 99</td>
</tr>
<tr>
<td>LRNX 126</td>
<td>MKI+ 99, NI- 99</td>
</tr>
<tr>
<td>LRNY 126</td>
<td>MKL 99, NIBSR 254, 274, 295</td>
</tr>
<tr>
<td>LRNZ 126</td>
<td>MLOC 116, NIP 25</td>
</tr>
<tr>
<td>LROT 122, 127</td>
<td>M MASS 112, NIP1 254, 274, 295</td>
</tr>
<tr>
<td>LS 32, 117</td>
<td>MMOV 116, NIP2 254, 274</td>
</tr>
<tr>
<td>LSCA 122</td>
<td>MN 112, NK2D 25</td>
</tr>
<tr>
<td>LSCR 123</td>
<td>MOMENTUM 118, NO 64</td>
</tr>
<tr>
<td>LSCZ 123</td>
<td>MOVE 17, 20, 87, NODE 23, 24</td>
</tr>
<tr>
<td>LSIZE 111</td>
<td>MOVIE 170, NODES 78, 79, 80, 81</td>
</tr>
<tr>
<td>LST 253, 273</td>
<td>MPGS 170, NODOUT 170</td>
</tr>
<tr>
<td>LSTL 123</td>
<td>MPLT 112, NONFRAME 112</td>
</tr>
<tr>
<td>LSYS 24, 25</td>
<td>MPS 268, NOGRID 112</td>
</tr>
<tr>
<td>LT 123</td>
<td>MRDI 294, NOMERGE 32</td>
</tr>
<tr>
<td>LTAS 123</td>
<td>MS 66, NONLINEAR 295</td>
</tr>
<tr>
<td>LTBC 123</td>
<td>MSCA 32, NOPL 26</td>
</tr>
<tr>
<td>LTBO 123</td>
<td>MSEL 116, NOTE 26</td>
</tr>
<tr>
<td>LTMN 178</td>
<td>MSIZ 112, NPB 99</td>
</tr>
<tr>
<td>LTMX 178</td>
<td>MSLAV 207, NRB 99</td>
</tr>
<tr>
<td>LTP 123</td>
<td>MSRF 253, 273, 294, NRB1 99</td>
</tr>
<tr>
<td>LTS 117</td>
<td>MSS 268, NCRCYCK 169</td>
</tr>
<tr>
<td>LTYPE 163, 214</td>
<td>MSYS 19, NSAD 112</td>
</tr>
<tr>
<td>LV 34, 111</td>
<td>MT 63, 67, NSET 112</td>
</tr>
<tr>
<td>LVC 123</td>
<td>MTHI 32, NSF 130</td>
</tr>
<tr>
<td>LVI 111</td>
<td>MTI 67, NSFN 130</td>
</tr>
<tr>
<td>LVS 111</td>
<td>MTV 67, NSMD 254, 274</td>
</tr>
<tr>
<td>LVT 127</td>
<td>MU 14, 154, 193, NSMO 26</td>
</tr>
<tr>
<td>LVTB 127</td>
<td>MVBC 25, NSRM 112</td>
</tr>
<tr>
<td>M 112, 159, 161, 201, 205, 237, 248, 268, 269, 288, 289</td>
<td>MVMA 171, NSSD 295</td>
</tr>
<tr>
<td>MA 66</td>
<td>MX 137, NSTEP 172, 254, 274</td>
</tr>
<tr>
<td>MASS 27</td>
<td>MY 137, NSV 112</td>
</tr>
<tr>
<td>MAT 23, 25, 142, 170, 178</td>
<td>MZ 137, NSWS 32</td>
</tr>
<tr>
<td>MATE 13, 20, 25, 42, 44, 46, 66, 82, 92, 94, 174</td>
<td>N 95, 96, 98, 99, 103, 157, NTIME 117</td>
</tr>
<tr>
<td>MATERIAL 38, 79, 80, 81</td>
<td>159, 196, 201, 237, NUMBER 78, 79, 80, 81</td>
</tr>
<tr>
<td>MATERIAL MAST 32</td>
<td>N+ 99, NURB 29</td>
</tr>
<tr>
<td>MATERIAL SLAV 32</td>
<td>N- 99, NV 34</td>
</tr>
<tr>
<td>MATM 19</td>
<td>N1 63, 79, 80, 81, OFFSET 173</td>
</tr>
<tr>
<td>MATRIX 137</td>
<td>N2 63, 79, 80, 81, OLAB 117</td>
</tr>
<tr>
<td>MATS 19</td>
<td>N3 79, 80, 81, OLD 36, 77</td>
</tr>
<tr>
<td>MATSUM 170</td>
<td>N4 80, 81, OMEGA 242</td>
</tr>
<tr>
<td>max 8</td>
<td>NC 21, OPIFS 172</td>
</tr>
<tr>
<td>MAXS 32</td>
<td>NCAD 112, OR 67</td>
</tr>
<tr>
<td>MAZE 36</td>
<td>NCFORCE 170, ORDER 67</td>
</tr>
<tr>
<td>MAZT 25</td>
<td>NCFORCE 170, ORIE 173</td>
</tr>
<tr>
<td>MB 66</td>
<td>NCN 112, ORV 26, 99, 103</td>
</tr>
<tr>
<td>MCOL 112</td>
<td>NCV 112, 130, OSCI 117</td>
</tr>
<tr>
<td>MDBC 25</td>
<td>NCY 112, 130, OSET 117</td>
</tr>
<tr>
<td>MDMP 178</td>
<td>NCYCLICES 170, OUTSIDE 20</td>
</tr>
<tr>
<td>MERGE 32</td>
<td>NDIV 29, OVERRIDE 27</td>
</tr>
<tr>
<td>MESH 207</td>
<td>NDPLT 112, P 95, 96, 98, 100, 103, 112</td>
</tr>
<tr>
<td>MEXP 116</td>
<td>NE 34, 149, 155, 162, 188, 194, 203, 284</td>
</tr>
<tr>
<td>MFBC 25</td>
<td>NEIG 254, 274, P 95, 96, 98, 100, 103, 112</td>
</tr>
<tr>
<td>MFTS 294</td>
<td>NEIG 254, 274, 149, 155, 162, 188, 194, 203, 284</td>
</tr>
<tr>
<td>min 8</td>
<td>NEWC 171, P 95, 96, 98, 100, 103, 112</td>
</tr>
<tr>
<td>MINIMUM 238</td>
<td>NEXP 116, 229, 149, 155, 162, 188, 194, 203, 284</td>
</tr>
<tr>
<td>MK 99</td>
<td>NFAIL 32</td>
</tr>
<tr>
<td>INDEX</td>
<td>LS-INGRID</td>
</tr>
<tr>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>PO 14, 230</td>
<td>PRBA 144, 182, 208, 209,</td>
</tr>
<tr>
<td>P1 63, 127</td>
<td>211, 223, 231, 234, 257,</td>
</tr>
<tr>
<td>P2 63, 127</td>
<td>265, 278, 292</td>
</tr>
<tr>
<td>P3 127</td>
<td>PRCA 144, 182, 208, 209,</td>
</tr>
<tr>
<td>PA 68, 230</td>
<td>211, 223, 231, 234, 257,</td>
</tr>
<tr>
<td>PARAM 227</td>
<td>265, 278, 292</td>
</tr>
<tr>
<td>PART 85</td>
<td>PRCB 144, 182, 208, 209,</td>
</tr>
<tr>
<td>PASS 172</td>
<td>211, 223, 231, 234, 257,</td>
</tr>
<tr>
<td>PATRAN 43</td>
<td>265, 278, 292</td>
</tr>
<tr>
<td>PAUSE 26</td>
<td>PRE 102</td>
</tr>
<tr>
<td>PB 68, 230</td>
<td>PRELOAD 12</td>
</tr>
<tr>
<td>PC 149, 154, 155, 157, 159, 188, 193, 194, 196,</td>
<td>PRINT 28, 113</td>
</tr>
<tr>
<td>201, 230, 252, 262, 284</td>
<td>PRISM 27</td>
</tr>
<tr>
<td>PCHK 112</td>
<td>PRL 100</td>
</tr>
<tr>
<td>PCI 153, 192</td>
<td>PSIS 19</td>
</tr>
<tr>
<td>PCOL 113</td>
<td>PSIC 13</td>
</tr>
<tr>
<td>PD 36</td>
<td>PSSCALE 28</td>
</tr>
<tr>
<td>PE 14, 15</td>
<td>PSEL 116</td>
</tr>
<tr>
<td>PER 162, 203</td>
<td>PSOPT 174</td>
</tr>
<tr>
<td>PERCENT 172</td>
<td>PSR 113</td>
</tr>
<tr>
<td>PEXP 116</td>
<td>PTOL 113</td>
</tr>
<tr>
<td>PEXT 15</td>
<td>PULL 12</td>
</tr>
<tr>
<td>PFOID 113</td>
<td>PV 113</td>
</tr>
<tr>
<td>PHASE 295</td>
<td>PVS 113</td>
</tr>
<tr>
<td>PHI 230</td>
<td>PYROTECHNIC 12</td>
</tr>
<tr>
<td>PHF 19</td>
<td>QUAD 20, 28</td>
</tr>
<tr>
<td>PHIS 19</td>
<td>QUADRATURE 178</td>
</tr>
<tr>
<td>PINF 113</td>
<td>QUIT 113</td>
</tr>
<tr>
<td>PINI 15</td>
<td>R 113, 159, 163, 201, 214,</td>
</tr>
<tr>
<td>PINT 15, 127</td>
<td>219, 226</td>
</tr>
<tr>
<td>PJ 21</td>
<td>R 242</td>
</tr>
<tr>
<td>PL3 130</td>
<td>R1 242</td>
</tr>
<tr>
<td>PLAN 131</td>
<td>R1E 248</td>
</tr>
<tr>
<td>PLANE 11, 24, 26, 27, 179, 207</td>
<td>R1P 248</td>
</tr>
<tr>
<td>PLOC 116</td>
<td>R2 242</td>
</tr>
<tr>
<td>PLTI 140, 172</td>
<td>R2D 172</td>
</tr>
<tr>
<td>PM 100</td>
<td>R 2E 248</td>
</tr>
<tr>
<td>PMASS 113</td>
<td>R2P 248</td>
</tr>
<tr>
<td>PMOV 116</td>
<td>RA 96</td>
</tr>
<tr>
<td>PNL 116</td>
<td>RADI 295</td>
</tr>
<tr>
<td>PLS 32</td>
<td>RADIUS 32</td>
</tr>
<tr>
<td>PNLT 20, 21, 32</td>
<td>RANG 139, 167</td>
</tr>
<tr>
<td>PO 64, 96, 97, 99, 127</td>
<td>RATE 13</td>
</tr>
<tr>
<td>POFF 103</td>
<td>RAYD 178</td>
</tr>
<tr>
<td>POINT 16, 38</td>
<td>RBAND 295</td>
</tr>
<tr>
<td>POLY 131</td>
<td>RBI 100</td>
</tr>
<tr>
<td>PON 103</td>
<td>RBMG 28</td>
</tr>
<tr>
<td>POOR 113</td>
<td>RBN 101</td>
</tr>
<tr>
<td>PLLV 28</td>
<td>RBOUND 170</td>
</tr>
<tr>
<td>POP 16</td>
<td>RC 21</td>
</tr>
<tr>
<td>PPRJ 127</td>
<td>RCFORC 170</td>
</tr>
<tr>
<td>220, 222, 225, 226, 228, 229, 236, 237, 256, 259, 261, 262, 268, 269, 277, 281, 283, 284, 288, 289, 290, 291</td>
<td>RDEENERGY 172</td>
</tr>
<tr>
<td>PO 14, 230</td>
<td>RDSI 140, 172</td>
</tr>
<tr>
<td>P1 63, 127</td>
<td>RE 101</td>
</tr>
<tr>
<td>P2 63, 127</td>
<td>RE+ 101</td>
</tr>
<tr>
<td>P3 127</td>
<td>READ 29</td>
</tr>
<tr>
<td>PA 68, 230</td>
<td>RE5 101</td>
</tr>
<tr>
<td>PARAM 227</td>
<td>RECT 133</td>
</tr>
<tr>
<td>PART 85</td>
<td>REDUCE 82, 113</td>
</tr>
<tr>
<td>PASS 172</td>
<td>REFP 113</td>
</tr>
<tr>
<td>PATRAN 43</td>
<td>REGION 36</td>
</tr>
<tr>
<td>PAUSE 26</td>
<td>REIN 172</td>
</tr>
<tr>
<td>PB 68, 230</td>
<td>RELAX 295</td>
</tr>
<tr>
<td>PC 149, 154, 155, 157, 159, 188, 193, 194, 196, 201, 230, 252, 262, 284</td>
<td>RELAX1 170</td>
</tr>
<tr>
<td>PRELOAD 12</td>
<td>RELAX2 170</td>
</tr>
<tr>
<td>PCI 153, 192</td>
<td>RELAX3 170</td>
</tr>
<tr>
<td>PCOL 113</td>
<td>RELAX4 170</td>
</tr>
<tr>
<td>PD 36</td>
<td>REPO 179</td>
</tr>
<tr>
<td>PE 14, 15</td>
<td>RES 68, 87</td>
</tr>
<tr>
<td>PER 162, 203</td>
<td>RESO 113</td>
</tr>
<tr>
<td>PERCENT 172</td>
<td>REST 68, 113, 140, 172</td>
</tr>
<tr>
<td>PEXP 116</td>
<td>RET 12, 13</td>
</tr>
<tr>
<td>PEXT 15</td>
<td>RETRACT 12</td>
</tr>
<tr>
<td>PFOID 113</td>
<td>REVERSE 13</td>
</tr>
<tr>
<td>PHASE 295</td>
<td>REZO 140</td>
</tr>
<tr>
<td>PHI 230</td>
<td>RFTS 254, 274</td>
</tr>
<tr>
<td>PHF 19</td>
<td>RHO 14, 15</td>
</tr>
<tr>
<td>PHIS 19</td>
<td>RHVC 140, 172</td>
</tr>
<tr>
<td>PINF 113</td>
<td>RIRDMS 172</td>
</tr>
<tr>
<td>PINI 15</td>
<td>RJ 21</td>
</tr>
<tr>
<td>PINT 15, 127</td>
<td>RLBV 140, 173</td>
</tr>
<tr>
<td>PO 64, 96, 97, 99, 127</td>
<td>RLN 123</td>
</tr>
<tr>
<td>POFF 103</td>
<td>RLS 123</td>
</tr>
<tr>
<td>POINT 16, 38</td>
<td>RLY 113</td>
</tr>
<tr>
<td>POLY 131</td>
<td>RM 113</td>
</tr>
<tr>
<td>PON 103</td>
<td>RNM 113</td>
</tr>
<tr>
<td>POOR 113</td>
<td>RND 8</td>
</tr>
<tr>
<td>PLLV 28</td>
<td>N1R 8</td>
</tr>
<tr>
<td>POP 16</td>
<td>RNUM 140, 173</td>
</tr>
<tr>
<td>PPRJ 127</td>
<td>RPLT 140, 173</td>
</tr>
<tr>
<td>PR 100, 131, 143, 146, 148, 149, 161, 162, 181, 185, 187, 188, 198, 199,</td>
<td>RPT 140, 173</td>
</tr>
<tr>
<td>203, 204, 205, 206, 207, 213, 215, 217, 218, 219, 220, 222, 225, 226, 228,</td>
<td>RQBV 140, 173</td>
</tr>
<tr>
<td>229, 236, 237, 256, 259, 261, 262, 268, 269, 277, 281, 283, 284, 288, 289, 290, 291</td>
<td>RR 68</td>
</tr>
<tr>
<td>PO 14, 230</td>
<td>RATION 29, 68</td>
</tr>
<tr>
<td>P1 63, 127</td>
<td>RP 114, 144, 257, 265</td>
</tr>
<tr>
<td>P2 63, 127</td>
<td>RPLT 140, 173</td>
</tr>
<tr>
<td>P3 127</td>
<td>RPFT 140, 173</td>
</tr>
<tr>
<td>PA 68, 230</td>
<td>RQBV 140, 173</td>
</tr>
<tr>
<td>PARAM 227</td>
<td>RR 68</td>
</tr>
<tr>
<td>PART 85</td>
<td>RTERM 141, 173</td>
</tr>
<tr>
<td>PASS 172</td>
<td>RTSF 141, 173</td>
</tr>
<tr>
<td>PATRAN 43</td>
<td>RVBC 29</td>
</tr>
<tr>
<td>PAUSE 26</td>
<td>RWFORC 170</td>
</tr>
<tr>
<td>PB 68, 230</td>
<td>RWPNAL 173</td>
</tr>
<tr>
<td>PC 149, 154, 155, 157,</td>
<td>RX 114, 137</td>
</tr>
<tr>
<td>159, 188, 193, 194, 196, 201, 230, 252, 262, 284</td>
<td>RXN 101</td>
</tr>
<tr>
<td>PCHK 112</td>
<td>RXNI 101</td>
</tr>
<tr>
<td>PCI 153, 192</td>
<td>RXY 137</td>
</tr>
<tr>
<td>PCOL 113</td>
<td>RZ 114, 137</td>
</tr>
<tr>
<td>RZX 137</td>
<td>SII- 102</td>
</tr>
<tr>
<td>S 117</td>
<td>SIIJ 60</td>
</tr>
<tr>
<td>S1 244</td>
<td>sin 8</td>
</tr>
<tr>
<td>S2 244</td>
<td>SINGLE 32</td>
</tr>
<tr>
<td>S3 244</td>
<td>sinh 8</td>
</tr>
<tr>
<td>SA 157, 196, 244</td>
<td>SIOPT 173</td>
</tr>
<tr>
<td>SAREA 179</td>
<td>SIZE 114</td>
</tr>
<tr>
<td>sas 8</td>
<td>SJ 21</td>
</tr>
<tr>
<td>SAVE 68, 137</td>
<td>SJK 60</td>
</tr>
<tr>
<td>SBC 295</td>
<td>SKI 60</td>
</tr>
<tr>
<td>SBI 101</td>
<td>SL 32, 102</td>
</tr>
<tr>
<td>SBRF 141, 173, 254, 274, 295</td>
<td>SLPRING 12</td>
</tr>
<tr>
<td>SC 63, 101, 185, 205, 209, 225, 231, 234</td>
<td>SLOC 179</td>
</tr>
<tr>
<td>SC03 29</td>
<td>SLOVM 35</td>
</tr>
<tr>
<td>SCAL 14, 15, 103</td>
<td>SO 64</td>
</tr>
<tr>
<td>SCALE 114, 137, 291</td>
<td>SOFT 231, 234</td>
</tr>
<tr>
<td>SCOL 114</td>
<td>SOLUTION STEADY 295</td>
</tr>
<tr>
<td>SD 20, 30, 133</td>
<td>SOLUTION</td>
</tr>
<tr>
<td>SDMV 30</td>
<td>TRANSIENT 295</td>
</tr>
<tr>
<td>SEAL 114</td>
<td>SP 131, 185, 205, 225, 244</td>
</tr>
<tr>
<td>SEAL CIRCLE 114</td>
<td>SPACE 38</td>
</tr>
<tr>
<td>SEAL OFF 114</td>
<td>SPALL 157, 196</td>
</tr>
<tr>
<td>SEAL OUTLINE 114</td>
<td>SPH 24, 27, 42, 44, 46, 70, 75, 83, 92, 94, 133</td>
</tr>
<tr>
<td>SECFORCE 170</td>
<td>SPIN 8</td>
</tr>
<tr>
<td>SECTION 28, 79</td>
<td>SPC 102</td>
</tr>
<tr>
<td>SEGMENT 179</td>
<td>SPC 102</td>
</tr>
<tr>
<td>SENSOR 12, 13</td>
<td>SPD 34</td>
</tr>
<tr>
<td>SEPARATE 175</td>
<td>SPD 103</td>
</tr>
<tr>
<td>SEQUENTIAL 173</td>
<td>SPHE 24, 27, 42, 44, 46, 70, 75, 83, 92, 94, 133</td>
</tr>
<tr>
<td>SETS 32</td>
<td>TENSION 233</td>
</tr>
<tr>
<td>SF 69, 234</td>
<td>TGC 297, 298, 299, 300</td>
</tr>
<tr>
<td>SFAIL 32</td>
<td>TFAIL 231, 234</td>
</tr>
<tr>
<td>SFC 102</td>
<td>TGM 297, 298, 299, 300</td>
</tr>
<tr>
<td>SFE 69</td>
<td>TINV 141, 175, 255</td>
</tr>
<tr>
<td>SFEI 69</td>
<td>TIVES 20</td>
</tr>
<tr>
<td>SFI 69</td>
<td>TIVE 37</td>
</tr>
<tr>
<td>SFORM 179</td>
<td>THET 20</td>
</tr>
<tr>
<td>SFS1 219</td>
<td>TIC 37, 42, 44, 46, 70, 76, 83, 87, 92, 94, 148, 187, 261, 266, 283, 287, 299, 300</td>
</tr>
<tr>
<td>SFS2 219</td>
<td>TIC 37, 42, 44, 46, 70, 76, 83, 87, 92, 94, 148, 187, 261, 266, 283, 287, 299, 300</td>
</tr>
<tr>
<td>SFSI 141, 173</td>
<td>TIC 37, 42, 44, 46, 70, 76, 83, 87, 92, 94, 148, 187, 261, 266, 283, 287, 299, 300</td>
</tr>
<tr>
<td>SFT1 219</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SFT2 219</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SFV 69</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SFVI 69</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SHELL 179</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SHELLS 80</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SHIFT 254, 274</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SHRINK 114</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SI 31, 102, 117</td>
<td>TICR 33</td>
</tr>
<tr>
<td>SI 102</td>
<td>TICY 37</td>
</tr>
<tr>
<td>SIDX 170</td>
<td>TICY 37</td>
</tr>
<tr>
<td>SIGF 162, 203</td>
<td>TICY 37</td>
</tr>
<tr>
<td>SIGM 157, 196</td>
<td>TICY 37</td>
</tr>
<tr>
<td>SIGO 157, 196</td>
<td>TICY 37</td>
</tr>
<tr>
<td>SIGY 146, 148, 155, 162, 185, 187, 194, 198, 199, 203, 204, 206, 213, 215, 218, 220, 225, 259, 261, 281, 283</td>
<td>TITY 37</td>
</tr>
<tr>
<td>SII 102</td>
<td>SII 102</td>
</tr>
<tr>
<td>SII+ 102</td>
<td>TITY 37</td>
</tr>
</tbody>
</table>

**IND.7**
<table>
<thead>
<tr>
<th>INDEX</th>
<th>LS-INGRID</th>
</tr>
</thead>
<tbody>
<tr>
<td>TJ 21</td>
<td>VARIABLE 171</td>
</tr>
<tr>
<td>TLHA 297, 298, 299, 300</td>
<td>VC 236</td>
</tr>
<tr>
<td>TLOC 179</td>
<td>VD 38</td>
</tr>
<tr>
<td>TM 104, 159, 201</td>
<td>VDA 207</td>
</tr>
<tr>
<td>TMASS 114</td>
<td>VE 34, 104</td>
</tr>
<tr>
<td>TMCG 37</td>
<td>VEC 176</td>
</tr>
<tr>
<td>TMI 104</td>
<td>VEC92 176</td>
</tr>
<tr>
<td>TMM 37</td>
<td>VECDYNA 93</td>
</tr>
<tr>
<td>TMSM 37</td>
<td>VECTOR 26</td>
</tr>
<tr>
<td>TMVP 38</td>
<td>VELOCITY 27, 39, 70</td>
</tr>
<tr>
<td>TN 104</td>
<td>VEOS 115</td>
</tr>
<tr>
<td>TNI 104</td>
<td>VF 215</td>
</tr>
<tr>
<td>TO 157, 159, 196, 201</td>
<td>VFRI 33</td>
</tr>
<tr>
<td>TOTAL 118</td>
<td>VIEW 115</td>
</tr>
<tr>
<td>TP 114</td>
<td>VARIABLE 171</td>
</tr>
<tr>
<td>TRACER 38</td>
<td>VO 64</td>
</tr>
<tr>
<td>TRACER t</td>
<td>VOLT 15</td>
</tr>
<tr>
<td>particle file..i.Tracer</td>
<td>Particles 171</td>
</tr>
<tr>
<td></td>
<td>VS 149, 188, 262, 284</td>
</tr>
<tr>
<td>TRANS 38</td>
<td>VSCA 14</td>
</tr>
<tr>
<td>TR12 70, 87</td>
<td>VTSP 70</td>
</tr>
<tr>
<td>TRIA 70, 88, 133</td>
<td>VVEC 207, 291</td>
</tr>
<tr>
<td>TRIAD 114</td>
<td>WARP 176</td>
</tr>
<tr>
<td>TRPT 114</td>
<td>WBGR 115</td>
</tr>
<tr>
<td>TS 131</td>
<td>WBIF 115</td>
</tr>
<tr>
<td>TS2P 131</td>
<td>WECE 248</td>
</tr>
<tr>
<td>TSF 223</td>
<td>WEDGE 176</td>
</tr>
<tr>
<td>TSH 119</td>
<td>WPCP 248</td>
</tr>
<tr>
<td>TSLIMIT 175</td>
<td>WRDB 115</td>
</tr>
<tr>
<td>TSORT 175</td>
<td>WRITE 38</td>
</tr>
<tr>
<td>TSS 175</td>
<td>X 13, 78, 248</td>
</tr>
<tr>
<td>TSSFD 169</td>
<td>X0 163, 214</td>
</tr>
<tr>
<td>TSTEP 168</td>
<td>X 13, 78, 248</td>
</tr>
<tr>
<td>TTHICK 179</td>
<td>XBO 139, 167</td>
</tr>
<tr>
<td>TIME 115</td>
<td>XF 17</td>
</tr>
<tr>
<td>TUPD 175</td>
<td>XLE 17</td>
</tr>
<tr>
<td>TV 115</td>
<td>XO 38</td>
</tr>
<tr>
<td>TYPE 13, 142, 180</td>
<td>XP 182, 208, 210, 212, 216, 224, 227, 232, 235, 278, 293</td>
</tr>
<tr>
<td>TZD 38</td>
<td>XSCA 38, 137</td>
</tr>
<tr>
<td>TZD 38</td>
<td>XSYM 18</td>
</tr>
<tr>
<td>U 115</td>
<td>XT 209, 231, 234</td>
</tr>
<tr>
<td>UDEF 117</td>
<td>XVEL 118</td>
</tr>
<tr>
<td>UI 21</td>
<td>Y 13, 78, 248</td>
</tr>
<tr>
<td>UL 188, 262, 284</td>
<td>YBO 139, 167</td>
</tr>
<tr>
<td>ULD 188</td>
<td>YC 209, 231, 234</td>
</tr>
<tr>
<td>UPDATE 115</td>
<td>YF 17</td>
</tr>
<tr>
<td>V 137</td>
<td>YLD 139, 167</td>
</tr>
<tr>
<td>V0 241, 242, 243, 244, 245, 247, 250, 252</td>
<td>YLE 17</td>
</tr>
<tr>
<td>V1 182, 208, 210, 212, 224, 227, 232, 235, 278, 293</td>
<td>YOFF 39</td>
</tr>
<tr>
<td>V2 64, 182, 208, 210, 212, 224, 227, 232, 235, 278, 293</td>
<td>YP 182, 208, 210, 212, 216, 224, 227, 232, 235, 278, 293</td>
</tr>
<tr>
<td>V3 182, 208, 210, 212, 224, 227, 232, 235, 278, 293</td>
<td>YSCA 39, 137</td>
</tr>
<tr>
<td>V90 176</td>
<td>YSYM 18</td>
</tr>
<tr>
<td>V91 176</td>
<td>YT 209, 231, 234</td>
</tr>
<tr>
<td>V92 176</td>
<td>YVEL 118</td>
</tr>
<tr>
<td>V93 176</td>
<td>Z 13, 78, 248</td>
</tr>
</tbody>
</table>

Component Interface
Defining 98

Composite
Angles 177
Damage Model 231
Plasticity Based Damage 234

Contact Interface
Defining 31
Displaying 110
DYNA3D Options 173
Eroding 33, 171
Geometric 20
Rigid Wall 27
Segment Selection 102
Slave Nodes 102

Control Volume
Defining 96
Definition 13
Displaying 107

Convection Boundary
Condition
Applying 97
Displaying 108

Convergence Tolerance
TOPAZ 294

Coordinate transformations 16
Copying Part 24
Creep
DYNA3D 237
NIKE2D 266
NIKE3D 287

Cross Section
Displaying 107
Crushable Foam
DYNA3D 188, 200
NIKE2D 262
NIKE3D 284

Cyclic Symmetry
Defining 16, 96
Displaying 107
Cylindrical Joint 21
Damper 34
Defining 103
Displaying 110
Damping
Material 178
Density
DYNA3D 179
Detonation Point 16
Displaying 108
Digitized Surface
Defining 17
Displaying 111
Saving 25

Directives 9
<table>
<thead>
<tr>
<th>LS-INGRID INDEX</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Discrete Mass</strong></td>
</tr>
<tr>
<td>Defining 100</td>
</tr>
<tr>
<td><strong>Displacement Boundary Condition</strong></td>
</tr>
<tr>
<td>Applying 97</td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td>Rigid Body 25</td>
</tr>
<tr>
<td><strong>Displacement Convergence Tolerance</strong></td>
</tr>
<tr>
<td>NIKE2D 253</td>
</tr>
<tr>
<td>NIKE3D 273</td>
</tr>
<tr>
<td><strong>Display Options</strong></td>
</tr>
<tr>
<td>Color Selection 110</td>
</tr>
<tr>
<td>Overlaying 110</td>
</tr>
<tr>
<td>Removal 110</td>
</tr>
<tr>
<td><strong>Duplicate Node Removal</strong> 113, 114</td>
</tr>
<tr>
<td><strong>DYNA3D</strong></td>
</tr>
<tr>
<td>Airbag Statistics 170</td>
</tr>
<tr>
<td>ASCII Output Files 170</td>
</tr>
<tr>
<td>Beam Integration Rule 167</td>
</tr>
<tr>
<td>Brode Function 167</td>
</tr>
<tr>
<td>Bulk Viscosity 168, 169</td>
</tr>
<tr>
<td>CAL3D Coupling 168</td>
</tr>
<tr>
<td>Centrifugal Load 171</td>
</tr>
<tr>
<td>Comments 26</td>
</tr>
<tr>
<td>Contact Penalty 173</td>
</tr>
<tr>
<td>D3HSP 168</td>
</tr>
<tr>
<td>Damping 174</td>
</tr>
<tr>
<td>Dynamic Relaxation 169</td>
</tr>
<tr>
<td>Full Restart 170</td>
</tr>
<tr>
<td>Gravity Load 171</td>
</tr>
<tr>
<td>Hourglass Control 169</td>
</tr>
<tr>
<td>Hourglass Energy 171</td>
</tr>
<tr>
<td>Importing Files 93</td>
</tr>
<tr>
<td><strong>Initial Time Step</strong> 171</td>
</tr>
<tr>
<td><strong>MADYMO Coupling</strong> 168</td>
</tr>
<tr>
<td>Mass Scaling 169, 172</td>
</tr>
<tr>
<td>Maximum Time Step 171</td>
</tr>
<tr>
<td>Minimum Time Step 175</td>
</tr>
<tr>
<td><strong>Output Control</strong> 168</td>
</tr>
<tr>
<td>Parallel Processing 172</td>
</tr>
<tr>
<td>Plane Stress Plasticity Option 172</td>
</tr>
<tr>
<td>Rayleigh Damping 179</td>
</tr>
<tr>
<td>Rayleigh Damping Energy Dissipation 172</td>
</tr>
<tr>
<td>Restart 173</td>
</tr>
<tr>
<td>Rigid Wall Penalty 173</td>
</tr>
<tr>
<td>Shell Formulation 174</td>
</tr>
<tr>
<td>Shell Integration Rule 174</td>
</tr>
<tr>
<td>Shell Thickness Updates 175</td>
</tr>
<tr>
<td><strong>Stone Wall Energy Dissipation</strong> 174</td>
</tr>
<tr>
<td><strong>System Damping</strong> 171</td>
</tr>
<tr>
<td>TAUROS Database Save Interval 172</td>
</tr>
<tr>
<td>Termination Cycle 175</td>
</tr>
<tr>
<td>Termination Time 175</td>
</tr>
<tr>
<td>Time Step 169</td>
</tr>
<tr>
<td>Time Step Scale Factor 175</td>
</tr>
<tr>
<td><strong>Eigenvalue Extraction</strong></td>
</tr>
<tr>
<td>NIKE2D 254</td>
</tr>
<tr>
<td>NIKE3D 274</td>
</tr>
<tr>
<td><strong>Elastic Material</strong></td>
</tr>
<tr>
<td>DYNA3D 181</td>
</tr>
<tr>
<td><strong>Elasticity</strong></td>
</tr>
<tr>
<td>NIKE2D 256</td>
</tr>
<tr>
<td>Element Delete on Restart Displaying 108</td>
</tr>
<tr>
<td>Element Number Displaying 111</td>
</tr>
<tr>
<td>Shifting 23</td>
</tr>
<tr>
<td><strong>Element Print Block</strong></td>
</tr>
<tr>
<td>Defining 97</td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td>Element Shrink Plots 114</td>
</tr>
<tr>
<td><strong>Emmisivity Curves</strong></td>
</tr>
<tr>
<td>TOPAZ 295</td>
</tr>
<tr>
<td><strong>Energy Convergence Tolerance</strong></td>
</tr>
<tr>
<td>NIKE2D 253</td>
</tr>
<tr>
<td>NIKE3D 273</td>
</tr>
<tr>
<td><strong>Equation-of-State Gruneisen 244</strong></td>
</tr>
<tr>
<td>Ignition and Growth of Reaction 248</td>
</tr>
<tr>
<td>JWL 242</td>
</tr>
<tr>
<td>Linear Polynomial 241</td>
</tr>
<tr>
<td>Linear Polynomial with Energy Leak 247</td>
</tr>
<tr>
<td>Ratio of Polynomials 245</td>
</tr>
<tr>
<td>Sack Tuesday High Explosive 243</td>
</tr>
<tr>
<td>Tabulated 252</td>
</tr>
<tr>
<td>Tabulated Compaction 250</td>
</tr>
<tr>
<td><strong>Equipotential Relaxation</strong> 170</td>
</tr>
<tr>
<td><strong>Eulerian Formulation</strong></td>
</tr>
<tr>
<td>DYNA3D 170</td>
</tr>
<tr>
<td><strong>Explosive</strong></td>
</tr>
<tr>
<td>JWL Burn Model 242</td>
</tr>
<tr>
<td>Material 192</td>
</tr>
<tr>
<td>Reactive Burn Model 249</td>
</tr>
<tr>
<td>Sack Burn Model 243</td>
</tr>
<tr>
<td><strong>Failing Symmetry Plane</strong></td>
</tr>
<tr>
<td>Defining 20</td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td><strong>Failure</strong></td>
</tr>
<tr>
<td>Tied Nodal Group 97</td>
</tr>
<tr>
<td><strong>Fixed Nodes</strong></td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td><strong>Flexion-Torsion Joint</strong></td>
</tr>
<tr>
<td>Defining 19</td>
</tr>
<tr>
<td><strong>Fluid</strong></td>
</tr>
<tr>
<td>DYNA3D 169</td>
</tr>
<tr>
<td><strong>Flux Boundary Condition</strong></td>
</tr>
<tr>
<td>Applying 97</td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td><strong>Foam</strong> 233</td>
</tr>
<tr>
<td><strong>Folding</strong> 20</td>
</tr>
<tr>
<td><strong>Force Load</strong></td>
</tr>
<tr>
<td>Applying 97</td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td>Frazer-Nash Rubber 36</td>
</tr>
<tr>
<td><strong>Gravity Load</strong></td>
</tr>
<tr>
<td><strong>Gravity Control</strong></td>
</tr>
<tr>
<td><strong>Interactive Model Updates</strong> 115</td>
</tr>
<tr>
<td><strong>Intrinsic functions</strong> 8</td>
</tr>
<tr>
<td><strong>Isoparametric Relaxation</strong> 170</td>
</tr>
<tr>
<td><strong>Johnson/Cook Material</strong></td>
</tr>
<tr>
<td><strong>Joint</strong></td>
</tr>
<tr>
<td>Defining 21, 98</td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td><strong>Joints</strong> 170</td>
</tr>
<tr>
<td><strong>JOY Interface Node</strong></td>
</tr>
<tr>
<td><strong>Defining 98</strong></td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td><strong>Kikuchi Relaxation</strong> 170</td>
</tr>
<tr>
<td><strong>Line Definition</strong></td>
</tr>
<tr>
<td>Displaying 111</td>
</tr>
<tr>
<td><strong>Three-dimensional</strong> 23</td>
</tr>
<tr>
<td><strong>Two-dimensional</strong> 23</td>
</tr>
<tr>
<td><strong>Line Search Tolerance</strong></td>
</tr>
<tr>
<td>NIKE2D 253</td>
</tr>
<tr>
<td>NIKE3D 273</td>
</tr>
<tr>
<td><strong>LLNL-INGRID Compatibility</strong> 36</td>
</tr>
<tr>
<td>Load Curve 23</td>
</tr>
<tr>
<td>INDEX</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>Displaying 111</td>
</tr>
<tr>
<td>Local Axes</td>
</tr>
<tr>
<td>Displaying 108</td>
</tr>
<tr>
<td>Specifying 67</td>
</tr>
<tr>
<td>Local System</td>
</tr>
<tr>
<td>Defining 24</td>
</tr>
<tr>
<td>LS-DYNA2D 16</td>
</tr>
<tr>
<td>LS-DYNA3D 16</td>
</tr>
<tr>
<td>LS-DYNA3D Version 902 176</td>
</tr>
<tr>
<td>LS-DYNA3D Version 910 176</td>
</tr>
<tr>
<td>LS-DYNA3D Version 920 176</td>
</tr>
<tr>
<td>LS-DYNA3D Version 930 176</td>
</tr>
<tr>
<td>LS-NIKE3D 26</td>
</tr>
<tr>
<td>MADYMO3D 20, 179</td>
</tr>
<tr>
<td>Repositioning Materials</td>
</tr>
<tr>
<td>35</td>
</tr>
<tr>
<td>Unit Conversions In</td>
</tr>
<tr>
<td>Coupling 168</td>
</tr>
<tr>
<td>Mass Property</td>
</tr>
<tr>
<td>Displaying 109, 112</td>
</tr>
<tr>
<td>Input 37</td>
</tr>
<tr>
<td>Material Subset 112</td>
</tr>
<tr>
<td>Part Subset 113</td>
</tr>
<tr>
<td>Total 114</td>
</tr>
<tr>
<td>Mass Scaling</td>
</tr>
<tr>
<td>DYN3D 172</td>
</tr>
<tr>
<td>Masses 109</td>
</tr>
<tr>
<td>Material</td>
</tr>
<tr>
<td>Data Input 25</td>
</tr>
<tr>
<td>Display 107</td>
</tr>
<tr>
<td>Displaying 110, 112, 113</td>
</tr>
<tr>
<td>Heading 177</td>
</tr>
<tr>
<td>Highlighting 109</td>
</tr>
<tr>
<td>Increments 21</td>
</tr>
<tr>
<td>Label Increment 24</td>
</tr>
<tr>
<td>Maximum Time Step</td>
</tr>
<tr>
<td>NIKE2D 253</td>
</tr>
<tr>
<td>TOPAZ 294</td>
</tr>
<tr>
<td>Maze Part</td>
</tr>
<tr>
<td>Tolerance 25</td>
</tr>
<tr>
<td>Membrane</td>
</tr>
<tr>
<td>DYN3D 179</td>
</tr>
<tr>
<td>Metallic Honeycomb 215</td>
</tr>
<tr>
<td>Minimum Time Step</td>
</tr>
<tr>
<td>NIKE2D 253</td>
</tr>
<tr>
<td>TOPAZ 294</td>
</tr>
<tr>
<td>Mooney-Rivlin Rubber</td>
</tr>
<tr>
<td>NIKE3D 290</td>
</tr>
<tr>
<td>MOVIE.BYU</td>
</tr>
<tr>
<td>DYN3D 170</td>
</tr>
<tr>
<td>MPGS</td>
</tr>
<tr>
<td>DYN3D 170</td>
</tr>
<tr>
<td>MVMA-DYN3D 171</td>
</tr>
<tr>
<td>NASTRAN</td>
</tr>
<tr>
<td>Importing Files 45</td>
</tr>
<tr>
<td>Newmark Integration</td>
</tr>
<tr>
<td>Parameters</td>
</tr>
<tr>
<td>NIKE2D 254</td>
</tr>
<tr>
<td>NIKE3D 274</td>
</tr>
<tr>
<td>TOPAZ 295</td>
</tr>
<tr>
<td>NIKE2D 25</td>
</tr>
<tr>
<td>Plastic Hydrodynamic Material</td>
</tr>
<tr>
<td>Rigid Material</td>
</tr>
</tbody>
</table>
DYNA3D 207
Rigid Wall 27
Rotational Velocity
  Initial 75, 83, 87
Rubber
  DYNA3D 191
SALE Advection 169
Scaling coordinates 16
Screen Movement
  Left 107, 111
  Restoring Original View 113
  Right 113
  Rotation 113, 114
  Scaling 114
SDRC Ideas 41
Seat Belt 12
  Defining 96
  Displaying 107
Seat Belts 170
Section Property
  Scaling 28
Sensor 12
  Defining 96
Shell
  Displaying Free Edge 109
  Displaying Normal Vectors 109
  Displaying Warpage 110
  Formulation 179
  Integration Rule 178
  Orientation 99
  Property Numbers 104
  Quadrature 178
  Reference Fiber 179
  Thicknesses 70, 104
  Triangular 70, 88
  User Integration Rule 178
Shell/Brick Interface
  Displaying 109
Shift Frequency
  NIKE2D 254
Single Point Constraint
  Applying 95, 103
  Displaying 110
Slide Line
  Applying 102
  Displaying 110
Sliding Interface
  Applying 102
  Defining 31
  Displaying 110
  DYNA3D Options 173
  Insertion Tolerance 254, 274
Slipring 12
  Defining 96
SMUG 170
Soil
  NIKE2D 262
  NIKE3D 284
Solution Method
  NIKE2D 253
  Spherical Joint 21
  Spotweld 21
  Spring 34
    Defining 103
    Displaying 110
  Standard Part
    Mesh Smoothing 63
  Steady State Solution
    TOPAZ 295
  Steinberg Material
    DYNA3D 196
  Stone Wall 27
    Displaying 110
    Energy Dissipation 174
    Identifying Slave Nodes 103
  Substructure Interface
    Displaying 108
  Surface
    Applying To Mesh 69
    Definition 30
    Digitized 17
    NURB 29
    Smoothing 26
  Symmetry Plane 27
    Displaying 110
    Failing 20
  System Assembly
    Joint Definition 103
    System Name 37
    Viewing Assembly Points 110
    Viewing Systems 114
  System Damping
    Control volume 13
    DYNA3D 171, 174
  TAUROUS
    Creating Database From INGRID 115
  Temperature Boundary Condition
    Applying 98
    Displaying 110
  Temperature Initial Condition 37, 42, 44, 46, 70, 76, 83, 87, 92, 94
    Displaying 110
  Termination Time
    NIKE2D 255
    TOPAZ 295
  Thermal Effect
    DYNA2D 141
    DYNA3D 175
    NIKE2D 255
    NIKE3D 275
  Thermo-Plastic Material
    DYNA3D 187
  Thermo-Plasticity
    NIKE2D 261, 269
    NIKE3D 283, 289
  Thick Shell
    DYNA3D 179
    Local-t 178
Thickness
  Shell 37
  Three-Dimensional Line Definition
    Displaying 108
  Time Step
    NIKE2D 253
    NIKE3D 273
    TOPAZ 294
  Tolerance
    MAZE Part 25
    Surface Intersections 35
  Tool Path
    Displaying 109
  TOPAZ Materials
    Isotropic 297
    Isotropic Temperature Dependent 299
    Orthotropic Temperature Dependent 300
    TOPAZ2D 38, 294
    TOPAZ3D 38, 294
    Tracer Particles 38, 107
    Transient Solution
      TOPAZ 295
    Translational Joint 21
  Triad
    Plotting 114
  Truss Element
    NIKE3D 177
  Unit Systems
    TOPAZ 294
  Universal Joint 21
  Van Leer Advection 169
  VEC-DYNA3D 176
  Vector
    Displaying 110
  Velocity
    Initial 39, 42, 44, 46, 60, 76, 88, 92, 94
    Velocity Boundary Condition
      Applying 98
      Displaying 110
      Rigid Body 25, 29
  View Factor
    Non-interacting Materials 25
  Visco-Plastic Material
    DYNA3D 206
  Viscoelasticity
    NIKE3D 190, 236
    NIKE3D 286
  Volume Definition 39
  Wang-Nefske Inflator Model 14
  Warpage of Shell
    Displaying 110
  YASE Shell
    DYNA3D 179