True*Grid*®Output Manual

Version 2.1.0

XYZ Scientific Applications, Inc.

December 18, 2001

Table of Contents

I. Output File Format Commands

epb element print block for DYNA3D and LSDYNA

epb *set_name*

where

set_name	name of an element set

npb node print block for DYNA3D and LSDYNA

npb *node_selection*

where *node_selection* can be one of the following:

n *node*	select a node
rt *x y z*	select the node nearest to a point in Cartesian coordinates
cy *rho theta z*	select the node nearest to a point in cylindrical coordinates
sp *rho theta phi*	select the node nearest to a point in spherical coordinates
nset *set_name*	select all nodes in a node set

ndigits number of digits written for coordinates

ndigits *n*

where *n* is the number of digits

Remarks

The default is 7 and the minimum is 5. This applies only to the Ale3d and CFX output options.

abaqus ABAQUS output format

abaqus (no arguments)

Remarks

ABAQUS version 4-8 from Hibbitt, Karlsson, & Sorensen, Inc. This output will require the use of some of the following commands to generate a complete input file to ABAQUS.

title to assign a title to the problem
quadratic to build 2nd order elements
bsd to define beam cross section properties
beam to generate a beam part
block or **cylinder** to generate bricks and shells
mate to specify the default material
mt to assign material numbers
b for nodal constraints
ibm, **jbm**, **kbm** to define rebar within the part
bm to generate a string of beams
rotation to assign rigid body rotation
velocity to assign rigid body velocity
sid to define the property of an interface
lcd (load curves) to define amplitude curves
jd to define numbered shared nodal degrees of freedom
si to specify segments for an interface
temp to set the default global/part temperature
tm to specify part of the initial nodal temperatures
pr to specify pressure faces with load curves
pramp to specify pressure amplitude
dom to specify the domain of pramp
fc to specify concentrated nodal forces with load curves
fd to specify nodal displacements with load curves
fv to specify nodal velocities with load curves
ft to specify nodal temperatures with load curves
acc to specify nodal accelerations with load curves
mom to specify nodal moments with load curves
pm to assign mass to a existing node
npm to create a node and assign a mass to it
nset to assign nodes to named nodal sets
jt to specify nodes shared degrees of freedom
mpc to assign shared nodal (multiple point) constraints for a nodal set
abaqstep to define time/history steps with procedures and conditions
condition in the merge phase to view these properties
stp to merge parts into a single connected model
write to create a abaqus input file after selecting abaqus

ale3d ALE3D output format

ale3d *(no arguments)*

Remarks

ALE3D from the Lawrence Livermore National Laboratory this output will require the use of some of the following commands to generate a complete input file to ALE3D.

ale3d to select the ale3d output option
title to set the problem title
plane to define symmetry planes and stone walls
mate to specify the default material
mt to assign material numbers
b for nodal constraints
lcd and flcd to define function curves
pr to assign pressure loads on faces
pramp to specify pressure amplitude
arri to define shock arrival calculations
dist to define functions used by arri
dom to specify the domain of pramp
sw to specify nodes on a stone wall
rotation to assign rigid body rotation
velocity to assign rigid body velocity
fv to assign boundary velocities
sid to define the property of an interface (types 1-4)
si to specify segments for an interface
ve to specify initial velocities of a region within a part
vhg to assign volume heat generators
temp to assign the default constant temperature
tm to assign initial temperatures within a region
ft to assign prescribed nodal temperatures
fl to assign flux boundary condition surfaces
cv to assign convection boundary condition surfaces
rb to assign radiation boundary condition surfaces
nset to assign nodes to named nodal sets. this used to create the generalized nodal **boundary** conditions. nsetc is required as well.
nsetc to attach parameters to the node set
stp to merge parts into a single connected model
condition in the merge phase to view these properties
write to create an ALE3D input file after selecting ALE3D

ansys ANSYS output format

ansys *(no arguments)*

Remarks

ANSYS from Swanson Analysis Systems, Inc. This output will require the use of some of the following commands to generate a complete input file to ANSYS.

ansys to select the ANSYS output option
ansymats to define material and element types
ansynl to define nonlinear material properties, current sources, and free space properties
ansyopts to select analysis options
title to set the problem title
plane to define symmetry planes
rotation to assign rigid body rotation
velocity to assign rigid body velocity
mate to select a material number for a part
jd to define numbered shared nodal degrees of freedom
spd to define the properties of numbered springs and dampers
bsd to define beam cross section properties
beam to generate a beam part
block or cylinder to generate bricks and shells
mt to select a material for a region of a part
b to constrain nodal degrees of freedom
n to orient shell elements
or to orient local material coordinate systems within the element
ibm, **jbm**, **kbm** to define rebar within the part
fc to specify nodal forces (use 0 for the load curve)
mom to assign nodal moments (use 0 for the load curve)
fd to specify fixed nodal displacements (use 0 for the load curve)
fa to specify fixed nodal rotations
ndl for distributed nodal loads
pr to generate surface pressures (use 0 for the load curve)
pramp to specify pressure amplitude
dom to specify the domain of pramp
th to set shell thickness in a region different from the default
te to define constant nodal temperatures for temperature dependent materials.
vhg to generate heat generating elements (use 0 for the load curve)
cvt to generate convection thermal loads

hfl to generate nodal flux, flow, amps, and heat conditions
mp to set the scalar magnetic potential
v to set the electrostatic potential
pm to assign mass to a existing node
npm to create a node and assign a mass to it
temp to set the default part temperature
nset to assign nodes to named nodal sets
mpc to assign shared nodal (multiple point) constraints for a nodal set
jt to specify nodes shared degrees of freedom
spdp to specify a face to form a surface of springs/dampers
spring to create a numbered spring
bm to generate a string of beams
stp to merge parts into a single connected model
condition in the merge phase to view these properties
write to create a ansys input file after selecting ansys

autodyn AUTODYN-3D output format

autodyn (no arguments)

Remarks

AUTODYN-3D by Century Dynamics Incorporated. The AUTODYN-3D output must be selected before any parts are generated. Use the **supblk** command in the Part Phase to change the default definition of grids.

supblk to combine several blocks on a single part into one grid. Each grid becomes a part in the **True*Grid*®** database. See 461.
mt and **mti** to blank out blocks of the part without deleting them. This makes it possible to build a multiple block part which is then treated as a single grid. This is an alternative to using **supblk** to form multiple grids from one part. Use the material number 2 to identify those blocks within the part which are to be ignored. When the output file is written in the merge phase, the blanked blocks will be written so that AUTODYN will also ignore them. See page 415.

cf3d Convective Flow output format

cf3d (no arguments)

Remarks

Use the **cfc** and **cfci** commands in the part phase and the **cfc** command in the Merge Phase to assign conditions.

cfd-ace CFD-ACE output format

cfd-ace (no arguments)

Remarks

Output option for ACE by CFD Research Corporation. This is a structured output option. It must be selected before any parts are generated. To block out a region in the mesh, use the **mt** command and assign a material number other than 1 to that region. Use the **supblk** command to control the way regions are used to build structured grids. Use the **bb** command to define block interfaces. These commands will create interface commands in the grid.xxx file when the output files are written.

cfx CFX output format

cfx (no arguments)

Remarks

CFX4 by AEA technology. This option must be specified before the parts are generated because the mesh must be generated in the structured form.

> **supblk** to combine several blocks on a single part into one grid. Each grid becomes a part in the **True*Grid*®** database. See 461.
> **bb** to force two faces of two different parts to match. it is also used to identify block interfaces between blocks of the same part. warning - this does not cause the two faces within a single part to match. the matching of faces must be done using other commands in **True*Grid*®**.
> **il** to identify exterior faces of the model as inlets.

ol to identify exterior faces of the model as outlets.
flowint to identify faces or solids for generalized conditions
condition in the merge phase to view these properties.
write to create a cfx4 input file.

dyna3d DYNA3D output format

dyna3d (no arguments)

Remarks

DYNA3D from LLNL November, 1993. The **True*Grid*®** commands used to invoke specific DYNA3D options are listed below.

parameters in the control cards: **dynaopts**
materials: **dynamats**, **mate**, **mt**
equation of state: **dynaeos**
truss and beam element cross section properties: **dynamats**, **bsd**
shell element cross section properties: **dynamats**
thick shell element cross section properties: **dynamats**
beam user defined integration rules: **bind**
shell user defined integration rules: **sind**
nodes: **b**, **plane**
solid elements: **block**, **cylinder**, **or**
beam and truss elements: **block**, **cylinder**, **ibm**, **jbm**, **kbm**, **bm**
shell and membrane elements: **block**, **cylinder**, **n**, **th**
thick shell elements: **block**, **cylinder**, **or**
interface save segment definition: **iss**
nodal single point constraints: **lsys**, **lb**, **sfb**
sliding boundary planes: **plane**, **sfb**
symmetry planes with failure: **plane**, **syf**
node time history blocks: **npb**
element time history blocks: **epb**
gravity stress initialization: **dynaopts** (**gvst**)
brode functions: **dynaopts**
cross section definitions for force output: **csf**
load curves: **lcd**, **flcd**
nodal forces and follower forces: **fc**, **mom**, **ndl**
pressure loads: **pr**, **pramp**, **dom**, **arri**, **dist**
prescribed velocities and accelerations: **fv**, **acc**, **dynamats** (**rbv**)

rigid walls: **sw**
nodal constraints: **mpc**
initial conditions: **rotation**, **velocity**, **ve**
sliding interface definitions: **sid**, **si**
tie-breaking shell slidelines: not supported
tied node sets with failure: **fn**
rigid body merges: **rigbm**
extra nodes with rigid bodies: **jt**
rigid body joints: **jd**, **jt**
prescribed base accelerations: **dynaopts** (**grav**)
prescribed angular velocities: **dynaopts** (**xvel**, **yvel**, **zvel**)
momentum deposition in solid elements: **mdep**
detonation points: **detp**
shell-solid interfaces: not supported
discrete springs, dampers, and masses: **spd**, **spdp**, **spring**, **pm**, **npm**
rigid body inertial properties: not supported
nonreflecting boundary segments: **nr**
temperature input option i: **tepro**
temperature input option ii: **temp**, **te**
one dimensional slidelines: **si**
generalized rayleigh damping: not supported
material initialization for rotational motion: not supported
body force by material: not supported
cvs (madymo/atb) coupling data: not supported

This output will also require the use of some of the following commands to generate a complete input file to DYNA3D.

title to set the problem title
comment to set comments for specific sections of the output
nset to assign nodes to named nodal sets
stp to merge parts into a single connected model
condition in the merge phase to view these properties
write to create a DYNA3D input file after selecting dyna3d

es3d ES3D output format

es3d (no arguments)

Remarks

Electrostatic TOPAZ3D from LLNL.

exodusii Exodus II output format

exodusii (no arguements)

Remarks

This format is defined by Sandia National Laboratories. The manual used for this definition is dated Nov., 1995.

fidap FIDAP output format

fidap (no arguments)

Remarks

FIDAP revision 7.0 from Fluid Dynamics International. This output will require the use of some of the following commands to generate a complete input file to FIDAP.

stp	to merge parts into a single connected model
condition	in the merge phase to view properties.
write	to create a FIDAP input file.

fluent FLUENT output format

fluent no argument

FLUENT version 5 from Fluent, Inc. This option must be specified before the part is generated because the mesh must be generated in structured form. The mesh must be generated using only one part at this time. No regions within the part can be deleted for this option. The face type for the boundaries are set to walls (type 3) and the interior to live (type 2) by default. Use the material specification commands such as **mate** and **mt** to specify the interior cell types. Use the **ol**, **il** and **fbc**/**fbci** commands (under the Boundary sub-menu) to specify outlets, inlets and other boundary conditions, respectively, for boundary cell types (faces). After completing the part, end the part (**endpart**) and enter the Merge Phase (**merge**) to write the output file (**write**) for FLUENT.

2 - interior
3 - wall
4 - pressure-inlet, inlet-vent, intake-fan
5 - pressure-outlet, exhaust-fan, outlet-vent
7 - symmetry
8 - periodic shadow
9 - pressure-far-field
10 - velocity-inlet
12 - periodic
14 - fan, porous-jump, radiator
20 - mass-flow-inlet
24 - interface
36 - outflow
37 - axis

gemini GEMINI output format

gemini (no arguments)

Remarks

Select output for GEMINI from Lawrence Livermore National Laboratory.

gridgen3d GRIDGEN output option

gridgen (no arguments)

Remarks

Creates the gridgen3d volgrid file. Select the GRIDGEN3D output option before generating parts. Then switch to the Merge Phase (**merge**) and write (**write**) the file.

iri IRI output format

iri (no arguments)

This fluids output option creates grid data for the fluids codes developed by Innovative Research Inc. This option is unique because the material identification numbers are included for each cell. Use the **mate**, **mtv**, **mt**, and **mti** commands to specify the materials in the part. Use the **supblk** command to decompose the parts into grids or rely on the default grid decomposition. In either case, each grid will then become a part in the **True*Grid*®** database.

lsdyna LS-DYNA output format

lsdyna *type*

where

type can be:

fixed	for the fixed format
keyword	for the keyword format

Remarks

LS-DYNA 950 by Livermore Software Technology Corporation. This output will require the use of some of the following commands to generate an input file:

acc, **accs**, **accc**, **vacc**, **vaccs**, and **vaccc** to specify nodal accelerations with load curves
arri to define shock arrival calculations
b for nodal constraints
bind to define beam integration rules.
block or **cylinder** to generate bricks and shells
bm to generate a string of beams
bsd to define beam cross section properties.
condition in the merge phase to view these properties
cv to specify boundary convections
detp to create detonation points
dist to define functions used by arri
epb for element time history blocks
fc, **fcs** and **fcc** to assign nodal forces
fd, **fdc**, and **fds** to assign prescribed displacements
fl to specify surface boundary fluxes
fn to created tied nodes w/ failure
ft to assign prescribed nodal temperatures
fv, **fvc**, **fvs**, **fvv**, **fvvc** and **fvvs** to assign prescribed velocities
ibm, **jbm**, **kbm** to define beams within a part
iss to create a component segment interface
jd to define numbered shared nodal degrees of freedom
jt to specify nodes w/ shared degrees of freedom, stop welds, and extra nodes

lb to assign boundary constraints in a local coordinate system (lsys)
lcd and **flcd** to define function curves
ll for linearly interpolated nodal loads
lsdyeos to define equation of state models associated with a material model
lsdyopts to define analysis options
lsdymats to define material models
lsys to define a local coordinate system used by the lb command
mate to set the default material
mdep to specify momentum deposition in solid elements
mom to assign nodal moments (use 0 for the load curve)
mpc to assign shared nodal (multiple point) constraints for a nodal set
mt to assign a material to a region
n to orient shell elements
ndl for nodal distributed loads
npb for nodal time history blocks
npm to create a node and assign a mass to it
nr to specify non-reflecting boundaries
nset to assign nodes to a named nodal sets
or to orient local material coordinate systems within the element
orpt to orient boundary conditions
plane to define symmetry planes, symmetry planes w/ failure, and rigid walls.
pm to specify nodes as mass points
rb to assign radiation boundary condition surfaces
rigid to create nodal rigid bodies
rigbm for rigid body merges
rotation to specify a rigid body rotation
sc to generate ale smoothing constraints
sfb to automatically define local coordinate system nodal constraints.
shso to generate constrained shell in solid
si to specify segments for an interface
sid to define the property of an interface
sind to define shell integration rules
spd, **spdp** and **spring** for discrete springs and dampers
ssf to create variable thickness shells
stp to merge parts into a single connected model
sw to specify nodes on a stone wall
syf for symmetry plane with failure
te to define constant nodal temperatures for temperature dependent materials.
temp to define the default initial nodal temperature
tepro to assign temperature profiles (temperature option i)
th to set the shell thickness
title to set the problem title

tm to assign initial nodal temperatures
trp for tracer particles
velocity to specify a rigid body velocity
vhg to identify solid element heat generation
write to create a LS-DYNA input file after selecting ls-dyna3d

The following cross reference lists the LS-DYNA keyword followed by the **True*Grid*®** commands used to generate the data:

*airbag_option is not supported.
*airbag_interaction is not supported.
*airbag_reference_geometry is not supported.
*ale_smoothing use the **sc** command.
*boundary_convection_option use the **cv** or **cvi** commands.
*boundary_cyclic is not supported.
*boundary_flux_option use the **fl**, **fli**, **efl**, **efli**, **arri**, and **dist** commands.
*boundary_non_reflecting use the **nr** or **nri** commands.
*boundary_prescribed_motion_node use **fv** and its variants **fvi**, fvc, **fvci**, **fvs**, **fvsi**, **acc**, **acci**, **accc**, **accci**, **accs**, **accsi**, **fvv**, **fvvi**, **fvvc**, **fvvci**, **fvvs**, **fvvsi**, **vacc**, **vacci**, **vaccc, vaccci**, **vaccs**, **vaccsi**, **fd**, **fdi**, **fdc**, **fdci**, **fds**, and **fdsi**.
*boundary_prescribed_motion_rigid use **lsdymats** command to define the motion for the rigid body material type.
*boundary_pressure_outflow_option is not supported.
*boundary_radiation_segment use the **rb** and **rbi** commands.
*boundary_sliding_plane use the **plane** or the **sfb** command.
*boundary_spc_node use the **lb** (see also **lsys**) or the **sfb** command.
*boundary_symmetry_failure use the **plane**, **syf**, and **syfi** commands.
*boundary_temperature_option use the **ft** or **fti** commands.
*boundary_usa_surface is not supported.
*constrained_extra_nodes_set use the **jt** command (see also **jd**).
*constrained_generalized_weld_option is not supported.
*constrained_joint_option is not supported.
*constrained_joint_stiffness_option is not supported.
*constrained_linear is not supported.
*constrained_nodal_rigid_body use the **rigid** command requiring the creation of a node set.
*constrained_nodal_rigid_body_inertia use the **rigid** command requiring the creation of a node set.
*constrained_node_set use the **jt** command (also see **jd**) or **mpc** requiring the creation of a node set.
*constrained_rigid_bodies use the **rigbm** command.
*constrained_rigid_body_stoppers is not supported.
*constrained_rivet is not supported.

*constrained_shell_in_solid use the **shso** command.
*constrained_shell_to_solid is not supported.
*constrained_spotweld use the **jt** command (see also **jd**).
*constrained_tie-break is not supported.
*constrained_tied_nodes_failure use the **fn** or **fni** commands.
*contact use the **si** or **sii** commands (see also **sid** and **orpt**).
*contact_entity is not supported.
*contact_1d is not supported.
*control_accuracy use the **lsdyopts** command (define accuracy parameters)
*control_adapstep use the **lsdyopts** command (use update contact interface force)
*control_adaptive use the **lsdyopts** command (activate automatic remeshing).
*control_ale use the **lsdyopts** command (control parameters for ale).
*control_bulk_viscosity use the **lsdyopts** command (global bulk viscosity).
*control_contact use the **lsdyopts** command (change defaults for contact surfaces).
*control_coupling use the **lsdyopts** command (options for coupling).
*control_cpu use the **lsdyopts** command (control cpu time).
*control_dynamic_relaxation use the **lsdyopts** command (controls for dynamic relaxation).
*control_energy use the **lsdyopts** command (controls for energy dissipation).
*control_explosive_shadow use the **lsdyopts** command (control explosive shadow)
*control_hourglass use the **lsdyopts** command (set hourglass control defaults).
*control_implicit_auto use the **lsdyopts** command
(set implicit automatic time step parameters)
*control_implicit_dynamics use the **lsdyopts** command (activiate implicit dynamic analysis)
*control_implicit_general use the **lsdyopts** command (set implicit analysis control parameters)
*control_implicit_linear use the **lsdyopts** command (set implicit linear solver parameters
*control_implicit_nonlinear use the **lsdyopts** command (set implicit nonlinear solver)
*control_implicit_stabilization use the **lsdyopts** command (set artificial stabilization parameters)
*control_output use the **lsdyopts** command (set output display parameters).
*control_parallel use the **lsdyopts** command (control parallel processing usage).
*control_rigid use the **lsdyopts** command (switch the explicit rigid body joint treatment)
*control_shell use the **lsdyopts** command (controls for computing shell response).
*control_solution use the **lsdyopts** command (specify the analysis procedure).
*control_sph use the **lsdyopts** command (set the SPH particle control)
*control_structured use the **lsdyopts** command (option to write a ls-dyna3d structured deck).
*control_subcycle use the **lsdyopts** command (subcycling option).
*control_termination use the **lsdyopts** command (job termination).
*control_thermal_nonlinear use the **lsdyopts** command (control nonlinear hermal analysis).
*control_thermal_solver use the **lsdyopts** command (control thermal analysis).
*control_thermal_timestep use the **lsdyopts** command (control thermal analysis time step).

*control_timestep use the **lsdyopts** command (time step control).
*damping_global is not supported.
*damping_part_mass is not supported.
*damping_part_stiffness is not supported.
*database_option use the **lsdyopts** command (ascii database file creation).
*database_binary_option use the **lsdyopts** command (binary database options).
*database_cross_section_option is not supported.
*database_extent_option use the **lsdyopts** command (avs, mpgs, or movie database options and binary database options).
*database_history_option use the **epb** and **npb** commands.
*database_nodal_force_group is not supported.
*database_spring_forward is not supported.
*database_superplastic_forming use the **lsdyopts** command (superplastic forming files).
*database_tracer use the **trp** command.
*define_box is not supported.
*define_coordinate_nodes is not supported.
*define_coordinate_system use the **lsys** command. Local coordinate systems are also created when the **rigid** command is invoked.
*define_coordinate_vector is created when the **sfb** command is invoked.
*define_curve use **lcd** and **flcd** to define a load curve. A curve is created for an equation of state type 11 or for a density vs. depth load. A load curve is created for gravity stress initialization in the **lsdyopts** command.
*define_sd_orientation is created for spring/damped elements defined with the **spring** and **spdp** commands.
*define_table is not supported.
*define_vector this card is automatically generated when creating prescribed motion.
*deformable_to_rigid is not supported.
*deformable_to_rigid_automatic is not supported.
*deformable_to_rigid_inertia is not supported.
*element_beam_option use **bm**, **ibm**, **ibmi**, **jbm**, **jbmi**, **kbm** and **kbmi** commands.
*element_discrete use the **spdp** and **spring** commands (see **spd**).
*element_mass use the **npm** and **pm** commands.
*element_seatbelt is not supported.
*element_seatbelt_accelerometer is not supported.
*element_seatbelt_pretensioner is not supported.
*element_seatbelt_retractor is not supported.
*element_seatbelt_sensor is not supported.
*element_seatbelt_slipring is not supported.
*element_shell_thickness use the **block** or **cylinder** part command with negative indices to build shell elements. Use the **th** command to define constant thickness in the shell element. Use the **ssf** command to create variable thickness elements.
*element_solid_option use the **block** or **cylinder** part command.

*element_tshell use the **block** or **cylinder** part command. identify the thick shell elements by defining a material in **lsdymats** as a thick shell material. Then use the **mate** or **mt** commands to assign that material to the brick elements which are to be treated as thick shells.
*eos_linear_polynomial use the **lsdyeos** command.
*eos_jwl use the **lsdyeos** command.
*eos_sack_tuesday use the **lsdyeos** command.
*eos_gruneisen use the **lsdyeos** command.
*eos_ratio_of_polynomials use the **lsdyeos** command.
*eos_linear_polynomial_with_energy_leak use the **lsdyeos** command.
*eos_ignition_and_growth_of_reaction_in_he use the **lsdyeos** command.
*eos_tabulated_compaction use the **lsdyeos** command.
*eos_tabulated use the **lsdyeos** command.
*eos_propellant_deflagration use the **lsdyeos** command.
*eos_tensor_pore_collapse use the **lsdyeos** command.
*include is not supported.
*initial_detonation use the **detp** command.
*initial_momentum use the **mdep** command.
*initial_stress_beam is not supported.
*initial_stress_shell is not supported.
*initial_stress_solid is not supported.
*initial_temperature_node use the **temp** command to specify the default temperature. use the **tm** and **tmi** commands to change the temperature for a region of the mesh.
*initial_velocity is not supported.
*initial_velocity_node use the **velocity** or **rotation** commands to assign a rigid body velocity. Use the **ve** and **vei** commands to assign velocities to regions of the mesh.
*initial_velocity_generation is not supported.
*integration_beam use the **bind** command to specify a list of integration points. Use the **bsd** command to define a standard cross section. when a beam element class is selected during the definition of a material (**lsdymats**), select the numbered **bind** or **bsd** definition. Only one of **bind** and **bsd** should be defined for each number.
*integration_shell use the **sind** command. when a shell element class is selected during the definition of a material (**lsdymats**), select the numbered sind definition as the integration rule.
*interface_component_segment use the **iss** and **issi** commands.
*interface_linking_discrete_node_option is not supported.
*interface_linking_segment is not supported.
*interface_linking_edge is not supported.
*interface_joy is not supported.
*interface_springback is not supported.
*load_beam_option is not supported.
*load_body_option is not supported.

*load_body_generalized is not supported.
*load_brode use the **lsdyopts** command.
*load_density_depth use the **lsdyopts** command.
*load_heat_generation_solid use the **vhg** and **vhgi** commands.
*load_node_node use the **fc**, **fci**, **fcs**, **fcsi**, **fcc**, **fcci**, **ll** and **ndl** commands to define directed load actions. Use the **mom** and **momi** commands to define moments. Follower forces are not supported.
*load_rigid_body is not supported.
*load_segment use **pr**, **pri**, **pramp**, **dom**, **arri**, and **dist** commands.
*load_segment_set is not supported.
*load_shell_option is not supported.
*load_superplastic_forming is not supported.
*load_thermal_constant is not supported.
*load_thermal_constant_node use the **te** and **tei** commands.
*load_thermal_load_curve is not supported.
*load_thermal_topaz is not supported.
*load_thermal_variable is not supported.
*load_thermal_variable_node use the **tepro** command.
*mat_option use the **lsdymats** command. the following models are supported:

- add_erosion
- elastic fluid
- orthotropic elastic
- anisotropic elastic
- kinematic/isotropic elastic-plastic
- thermo-elastic-plastic
- soil and crushable foam
- linear viscoelastic
- blatz-ko rubber
- high explosive burn
- null (hydrodynamic w/o deviatoric stress)
- isotropic elastoplastic hydrodynamic
- steinberg-guinan thermal elastoplastic hydrodynamic
- isotropic elastoplastic
- isotropic elastoplastic with failure
- soil and crushable foam with failure
- johnson/cook plasticity
- pseudo tensor concrete/geological model
- elastoplastic with oriented crack
- power law isotropic plasticity
- strain rate dependent plasticity
- rigid
- thermal orthotropic w/ 12 constants

fiber composite w/ damage
temperature dependent thermal orthotropic w/ 12 curves
rate-dependent tabular isotropic plasticity
inviscid, two invariant geologic cap
orthotropic crushable honeycomb
compressible mooney-rivlin hyperelastic rubber
resultant plasticity
force limited resultant formulation for beams
closed-form update plasticity shell
slightly compressible rubber model
laminated glass model
barlat's anisotropic plasticity model
fabric model
kinematic/isotropic elastic-plastic green-naghdi rate model
barlat's 3-parameter plasticity model
transversely anisotropic elastic-plastic
blatz-ko compressible foam
transversely anisotropic elastic-plastic with fld
nonlinear elastic orthotropic material
user-defined material model #41
planar anisotropic plasticity model
user-defined material model #43
user-defined material model #44
user-defined material model #45
user-defined material model #46
user-defined material model #47
strain rate dependent plasticity with size dependent failure
user-defined material model #49
user-defined material model #50
temperature and rate dependent plasticity
sandia's damage model
low density closed cell polyurethane foam
composite damage model (chang matrix failure)
composite damage model (tsay-wu matrix failure)
low density urethane foam
composite damage model - damage mechanics
composite failure model - plasticity based
elastic with viscosity
maxwell/kelvin viscoelastic with maximum strain
viscous foam, ove arup & partners model
crushable foam
strain rate sensitive power-law plasticity

modified zerilli/armstrong
linear stiffness/linear viscous 3d discrete beam
nonlinear stiffness/viscous 3d discrete beam
nonlinear plastic/linear viscous 3d discrete beam
side impact dummy damper, sid damper
hydraulic/gas damper model
cable model
bikhu/dubois foam model

*node use the **block** and **cylinder** automatically generate nodes. The **bm**, **jt**, and **npm** commands also create nodes. Boundary constraints are set using the **b**, **b**i, and **plane** commands.

*part_option is not supported.

*rigidwall_geometric_option is not supported.

*rigidwall_planar use the **plane**, **sw**, and **swi** commands.

*section_beam use the **lsdymats** command. section properties are specified once the element class is selected for the material in question.

*section_discrete use the **spd** command.

*section_seatbelt is not supported.

*section_shell use the **lsdymats** command. section properties are specified once the element class is selected for the material in question.

*section_solid_option use the **lsdymats** command. section properties are specified once the element class is selected for the material in question.

*section_tshell use the **lsdymats** command. section properties are specified once the element class is selected for the material in question.

*set_beam is not supported.

*set_discrete is not supported.

*set_node_list use the **nset** command to define a node set. A node set is automatically created for each rigidwall planar, specified by the **sw** and **swi** commands, for each nodal type contact, constrained node set, constrained spotweld, constrained tied nodes failure, constrained extra nodes set, constrained nodal rigid body inertia.

*set_part_list - a part set is created for gravity stress initialization in the **lsdyopts** command, contact between parts, constrained shell in solid.

*set_segment - sets are created when the symmetry plane with failure is invoked using **plane**, **syf**, and **syfi** commands. Sets are created for segmented interfaces from the **iss** and **issi** commands.

*set_shell_option is not supported.

*set_solid is not supported.

*set_tshell is not supported.

*termination_option is not supported.

*title use the **title** command.

*user_interface_option is not supported.

*user_loading is not supported.

lsnike3d LS-NIKE3D output format

lsnike3d (no arguments)

Remarks

LS-NIKE3D by Livermore Software Technology Corporation. Use some of the following commands to generate a complete input file to LS-NIKE3D.

title: **title**
comments: **comment**
control card parameters: **lsnkopts**
materials: **lsnkmats**, **mt**, **mate**
beam element cross sections: **lsnkmats**, **bsd**
shell element cross sections: **lsnkmats**
user defined integration rules for shells: **sind**
user defined integration rules for beams: **bind**
nodes: **b**, **plane**
beam elements: **block**, **cylinder**, **ibm**, **jbm**, **kbm**, **bm**
shell elements: **block**, **cylinder**, **n**, **th**
solid elements: **block**, **cylinder**, **or**
rigid node and facet deck: **rigid**
discrete element deck: **spd**, **spdp**, **spring**, **pm**, **npm**
constrained nodal pairs: **mpc**, **jt, jd**
1d slideline: not supported
sliding interface definitions: **sid**, **si**
nodal reorder data: not supported
stonewall cards and symmetry planes: **sw**, **plane**
nodal time history blocks: **npb**
element time history blocks: **epb**
load curves: **lcd**, **flcd**
concentrated nodal loads: **fc**, **ndl**, **mom**
temperature profiles: **tepro**
pressure boundary condition cards: **pr**, **pramp**, **dom**, **arri**, **dist**
displacement boundary condition cards: **fd**
body force loads: **lsnkopts**
initial conditions: **rotation**, **velocity**, **ve**, **fv**, **lsnkmats**
aerodynamic drag load deck: not supported
foundation node boundary condition: not supported

marc MARC output format

marc (no arguments)

Remarks

MARC Rev. k.5 from MARC Analysis Research Corp. Use the **bsd** command to define the cross section for the beam. Note that only 3 coordinates are generated per node. If additional coordinates are required, they must be added to the **True*Grid*®** output file before running MARC. Beam elements 13, 14, 25, 31, 52, 76, 79, and 98 are supported. Use the default **bsd** options radius, thick, and bend to avoid defining a beam section in MARC. Geometry cards are automatically generated where they are needed for the beam elements. If a beam has no cross section properties, then it will be reported as such after the MARC input deck is written. Use the **lbsd** and **bsinfo** commands to inquire abut the cross sections. This output will require the use of some of the following commands to generate an input file:

marcmats to select material models
marcopts to select analysis options
title to set the problem title
bsd to define beam cross section properties
beam to generate a beam part
block or cylinder to generate bricks and shells
linear to generate first order elements (beams, shells, and bricks)
spd to define the properties of numbered springs and dampers
ibm, **jbm**, **kbm** to define rebar within the part
n to orient shell elements
or to orient local material coordinate systems within the element
b for nodal constraints
pm to assign mass to a existing node
npm to create a node and assign a mass to it
nset to assign nodes to named nodal sets
spdp to specify a face to form a surface of springs/dampers
spring to create a numbered spring
bm to generate a string of beams
stp to merge parts into a single connected model
condition in the merge phase to view these properties
write to create a marc input file after selecting marc

nastran NASTRAN output format

nastran (no arguments)

Remarks

NASTRAN v 68 from Macneal-Schwendler Corporation. This output will require the use of some of the following commands to generate an input file:

nastmats to select material models and element properties.
bsd to define beam cross section properties
beam to generate a beam part
title to set the problem title
block or **cylinder** to generate bricks and shells (chexa, cpenta, ctetra, cquad4, cquad8, cquadr, ctria3, ctriar, ctria6). Shell elements (cquadi and ctriai), planar shell elements (cquadr, ctriar), and shear panel (cshear) elements are selected through the material definition (nastmats).
linear to specify first order elements in the block or cylinder parts.
quadratic to specify second order elements in the block or cylinder parts.
ibm, **jbm**, **kbm** to define rebar within the part
jd to define numbered shared nodal degrees of freedom
spd to define the properties of numbered springs and dampers
n to orient shell elements
or to orient local material coordinate systems within the element
mt to assign material numbers
th to set constant shell element thicknesses
temp to assign the default constant temperature
te to define constant nodal temperatures for temperature dependent materials.
fc, **fcs**, and **fcc** to assign nodal forces (force)
ndl for distributed nodal loads (force)
ll for linearly interpolated nodal loads (force)
pr to assign pressure loads on faces
pramp to specify pressure amplitude
dom to specify the domain of pramp
b for nodal constraints
jt to specify nodes shared degrees of freedom
mom to assign nodal moments (moment)
pm to assign mass to a existing node
npm to create a node and assign a mass to it
nset to assign nodes to named nodal sets
mpc to assign shared nodal (multiple point) constraints for a nodal set
bm to generate a string of beams

spdp to specify a face to form a surface of springs/dampers
spring to create a numbered spring
t, **tp**, **stp**, **ptol**, and **bptol** to merge parts into a single connected model.
condition in the merge phase to view the above properties of the model.
write to create a Nastran input file after selecting Nastran.

nike3d NIKE3D output format

nike3d (no arguments)
enike3d (no arguments)
nnike3d (no arguments)
fnike3d (no arguments)

Remarks

NIKE3D from LLNL April, 1995. Alternatively, use **nnike3d** for the old nn output format, **enike3d** for the old en output format, or **fnike3d** (same as nike3d) for the new fn output format. These outputs will require the use of some of the following commands to generate a complete input file to NIKE3D.

nikeopts to select analysis options
nikemats to select material models
beam to generate a beam part
block or **cylinder** to generate bricks and shells
title to set the problem title
lcd and **flcd** to define function curves
jd to define numbered shared nodal degrees of freedom
sid to define the property of an interface
bind to define beam integration rules
sind to define shell integration rules
plane to define symmetry planes and stone walls
bsd to define beam cross section properties
spd to define the properties of numbered springs and dampers
mate to specify the default material
ibm, **jbm**, **kbm** to define rebar within the part
n to orient shell elements
or to orient local material coordinate systems within the element
mt to assign material numbers
b for nodal constraints
jt to specify nodes shared degrees of freedom

si to specify segments for an interface
sw to specify nodes on a stone wall
npb for nod print blocks
epb for element print blocks
fc to assign nodal forces
fv to assign prescribed velocities
ndl for distributed nodal loads
pr to assign pressure loads on faces
pramp to specify pressure amplitude
arri to define shock arrival calculations
dist to define functions used by arri
dom to specify the domain of pramp
mom to assign nodal moments
fd to assign fixed displacement boundary condition
tepro to assign temperature profiles
rotation to assign rigid body rotation
velocity to assign rigid body velocity
ve to specify initial velocities of a region within a part
th to set constant shell element thicknesses
spdp to specify a face to form a surface of springs/dampers
spring to create a numbered spring
pm to specify nodes as mass points
npm to create a node and assign a mass to it
nset to assign nodes to named nodal sets
mpc to assign shared nodal (multiple point) constraints for a nodal set
bm to generate a string of beams
stp to merge parts into a single connected model
condition in the merge phase to view these properties
write to create a nike3d input file after selecting nike3d

nekton2d NEKTON2D output format

nekton2d (no arguments)

Remarks

Two dimensional NEKTON 2.85 by Fluent Inc. This output will require the use of some of the following commands to generate a NEKTON2D input file. Select the NEKTON2D output option before generating parts. Then switch to the Merge Phase (**merg**e) and write (**write**) the file.

nekopts	to set switches, output options, and problem parameters
block or **cylinder**	to generate bricks and shells
title	to set the problem title
bv	to specify the inlet velocities
ol	to specify the outlets
condition	in the Merge Phase to view these properties
stp	to merge parts into a single connected model
write	to create a NEKTON input file after selecting NEKTON2D

nekton3d NEKTON3D output format

nekton3d (no arguments)

Remarks

Three dimensional NEKTON 2.85 by Fluent, Inc. This output will require the use of some of the following commands to generate a NEKTON3D input file. Select the NEKTON3D output option before generating parts. Then switch to the Merge Phase (**merge**) and write (**write**) the file.

nekopts	to set switches, output options, and problem parameters
block or **cylinder**	to generate bricks and shells
title	to set the problem title
bv	to specify the inlet velocities
ol	to specify the outlets
condition	in the Merge Phase to view these properties
stp	to merge parts into a single connected model
write	to create a nekton input file after selecting nekton3d

neutral Neutral Output Format

neutral (no arguments)

PATRAN Neutral File format from PDA Engineering. This output will require the use of some of the following commands to generate a neutral file.

patmats to select material models
mate to specify the default material
title to set the problem title

jd to define numbered shared nodal degrees of freedom
mt to assign material numbers
b for nodal constraints
te to assign constant nodal temperatures
pr to assign pressure loads on faces
pramp to specify pressure amplitude
dom to specify the domain of pramp
fc to assign nodal forces
ndl for distributed nodal loads
fd to assign fixed displacement boundary condition
vhg to generate heat generating elements (use 0 for the load curve)
jt to specify nodes shared degrees of freedom
nset to assign nodes to named nodal sets
mpc to assign shared nodal (multiple point) constraints for a nodal set
stp to merge parts into a single connected model
condition in the merge phase to view these properties
write to create a neutral input file after selecting neutral

refleqs REFLEQS output format

refleqs (no arguments)

Remarks

REFLEQS output must be selected before any parts are generated. Only one part should be generated. No regions should be deleted from the part. While in the part phase, use the **reg** and **regi** commands in the boundary menu to specify all boundary conditions. Also use the **por** and **pori** commands in the material menu to specify the porosity of regions. The **setsor** command in the part phase in the radiation menu allows you to specify the source terms on the right-hand side of the transport (conservation equations) for each dependent variable. The **inizone** command assigns initial conditions to zones in the mesh. Use **para** to define iuser and ruser parameters. To define iuser 10, for example, define the parameter iuser10 in **True*Grid*®**. When the part is finished, end the part and go into the merge phase to write the output file. A second binary file will be automatically written which will contain the grid information.

starcd STARCD output format

starcd (no arguments)

Remarks

Output of nodes and bricks for STARCD.

plot3d PLOT3D output format

plot3d (no arguments)

Remarks

Output for 3D multi-domain structured grids. Make this selection before creating any parts.

poly3d Generic output format

poly3d (no arguments)

Remarks

The output file is an ASCII representation of the three-dimensional polygons that comprise the mesh. Hardly any other information is written to the file.

The file format is designed so that you can easily write your own software to read it. The first record of the file is a title; all following data describes the polygons. For each polygon data set, the first record has three integers written with the Fortran format (I1,2I8). The first integer is the number 2, 3, or 4 for the number of nodes forming the polygon. A 2-node polygon forms a string or beam element. A 3-node polygon is a triangle, and a 4-node polygon is a quadrilateral in a plane. The next two integers on the first record are the part and material numbers of the polygon, respectively. The subsequent records give the nodal coordinates, one record for each node. These are written with the Fortran format (3(E14.7,1X)).

tascflow TASCflow output format

tascflow (no arguments)

Remarks

TASCFLOW version 2.2 by ASC.

topaz3d TOPAZ3D output format

topaz3d (no arguments)

Remarks

TOPAZ3D from LLNL. This output will require the use of some of the following commands to generate a complete input file to topaz3d.

tz3dopts to select analysis options
tz3dmats to select material models
title to set the problem title
sid to define the property of an interface
temp to define a default initial temperature
mate to specify the default material
lcd and **flcd** to define function curves for
    for the rate of thermal generation for each material
    for element heat generation
    for temperature boundary conditions
    for flux boundary conditions
    for convection boundary conditions
    for radiation boundary conditions
rband to define wave length breakpoints
emissivity to define emissivity curves
si to specify segments for an interface
vhg to specify heat generators
tm to specify initial temperatures
ft to specify boundary temperatures
fl to specify surface boundary fluxes
arri to define shock arrival calculations
dist to define functions used by arri
cv to specify boundary convections
rb to specify boundary thermal radiations
re to specify radiation enclosures
stp to merge parts into a single connected model
condition in the merge phase to view these properties
write to create a topaz3d input file after selecting topaz3d

verbatim write verbatim to the output file

verbatim
text
endverbatim

The text can be any number of lines of text. Anything on the same line and following the verbatim command is ignored. This text is saved. When the write command is issued, this text is written to the output file if the output file format is keyword driven so that the text can be in any order. This is needed because text is always placed near the beginning of the output file. Some of the formats supported by this feature are **Abaqus**, Ansys, **LSdyna**, **Marc**, and **Nastran**.

viewpoint VIEWPOINT output format

viewpoint (no arguments)

An existing mesh file can be used as geometry to create a new mesh. Typically, a mesh is read into **True*Grid*®** using the **readmesh** command. It is then written out as ViewPoint surface data. The exterior faces of solid elements and all shell elements are written in the ViewPoint format as surfaces, one surface per material. This requires two files. The first is the coordinate data. The second is the polygon connectivity data. Then this data can be read back into **True*Grid*®** using the **vpsd** command, forming surfaces.

II. Analysis Options

True*Grid*® is more than just a grid generator. A **True*Grid*®** output file is the *complete* input file for a simulation code. Much of a simulation code's input data, notably the geometry and mesh, are generic in nature. For control parameters and other such input options which are specific to the simulation code, use the appropriate command from this section. Material models also vary from one simulation code to another; see the section on material specification, page 283.

To issue one of these commands, use its dialogue box in **True*Grid*®**'s graphical user interface. For full details, read the simulation code's manual. It is impossible to reproduce all those manuals in this section. But for the sake of completeness, the command names are given below.

XYZ Scientific Applications is actively adding support of more simulation codes. By the time you get **True*Grid*®**, there may be more available than you will find in this section. We welcome your requests for more output formats!

abaqstep ABAQUS analysis step

abaqstep *abstep step_# proc pr_param pr_opts ; st_opts* ;

where the *proc* can be one of the following:
buckle for *buckle
the *pr_param* are (choose one):
dead for dead loading
live for live loading
there are no *pr_opts:*
ctd for *coupled temperatiure-displacement
the *pr_param* are:
basic tolerance
temperature tolerance
suggested initial time step
total time period for the step
the *pr_opts* are:
explicit for explicit integration
mtol tol for moments tolerance
nocreep for no creep
steady for steady state analysis
cetol *tol* for creep tolerance
deltmx *temp* for maximum temperature change
cetol *tol* for creep tolerance

deltmx *tol* for maximum temperature change
timemin *time* for minimum time increment
timemax *time* for maximum time increment

dynamic for *dynamic
the *pr_param* are:
method which can be one of the following:
explicit for explicit integration
subspace for subspace projection method
implicit tol for implicit integration
time for suggested time increment
time for time period
the *pr_opts* are:
direct for user control of step size (explicit only)
vectors *#_modes* to set number of modes (subspace only)
alpha *alpha* for artificial damping control
haftol *tol* for half-step residual tolerance
initial for no initial accelerations
nohaf for no half-step residual
mtol *tol* for moments tolerance
timemin *time* for minimum time increment
timemax *time* for maximum time increment

frequenc for *frequency
the *pr_param* are:
number_eigenvalues
maximum_frequency
the *pr_opts* are:
shift *frequency_squared* for shift point
nvecs *n* for number of vectors
maxit *n* for number of iterations

geostati for *geostatic
the *pr_param* are:
tol
the *pr_opts* are:
mtol *tol* for moments tolerance

heat for *heat transfer
the *pr_param* are:
temp for temperature tolerance
time for time step
time for time period
time for minimum time increment
the *pr_opts* are:
deltmx *temp* for maximum temperature change

endcon *ss* for steady state ending condition
endcon *period* for periodic
steady for steady state analysis
timmxinc *time* for maximum time increment
temprate *time* for temperature change rate

moddyn for *modal dynamic
the *pr_param* are: *time* for total time
the *pr_opts* are:
initial **yes** for start new dynamic response
initial **no** for use last dynamic response

random for *random responce
the *pr_param* are:
freq for lowest frequency
freq for highest frequency
n for number of points
bias for bias
i for frequency scale
there are no *pr_opts*

response for *response spectrum
the *pr_param* are:
name for name of response spectrum
x for x-direction cosine
y for y-direction cosine
z for z-direction cosine
scale for magnitude
the *pr_opts* are to repeat the following up to 2 times:
name for name of response spectrum
x for x-direction cosine
y for y-direction cosine
z for z-direction cosine
scale for magnitude

soils for *soils
the *pr_param* are:
tol for tolerance
time for initial time step
time for total time period for the step
the *pr_opts* are:
consolid for transient, consolidated analysis
endcon *ss* for end at steady state
endcon *period* for periodic
mtol tol for moments tolerance
utol tol for maximum pore pressure change

tmmninc *time* for minimum time increment
tmmxinc *time* for maximum time increment
presrate *rate* for minimum pore pressure rate of change

static for *static
the *pr_param* are:
tol for tolerance
time for initial time step
time for total time period for the step
the *pr_opts* are:
mtol *tol* for moments tolerance
tmmninc *time* for minimum time increment
tmmxinc *time* for maximum time increment

ssdyn for *steady state dynamic
the *pr_param* are:
freq for lowest frequency
freq for highest frequency
n for number of points
bias for bias
i for frequency scale
there are no *pr_opts*

visco for *visco
the *pr_param* are:
tol for tolerance
time for suggested initial time step
time for total time period for the step
the *pr_opts* are:
cetol tol for maximum creep strain rate
explicit for explicit integration
mtol *tol* for moments tolerance
tmmninc *time* for minimum time increment
tmmxinc *time* for maximum time increment

the procedure definition is followed by *st_opts* which can be:
amplitude *step* for stepped amplitude
amplitude *ramp* for ramped amplitude
cycle *#_inter* for maximum iterations in an increment
inc *#_inter* for maximum increments in a step
linear new for linear analysis with a new stiffness matrix
linear old for linear analysis with old stiffness matrix
monotoni for monotonic
nlgeom for geometric nonlinearity
rottol *tol* for maximum increment of rotation
submax for suppress subdivisions

abdload blc *load_curve_# type*
where type can be
pr for pressure
abcload blc *load_curve_# type*
where *type* can be
fc for concentrated force
mom for concentrated moments
fd for displacement
fv for velocity
acc for acceleration
ft for forced temperature

Remarks

This command is used to define each time/history step. Some parameters can be selected and the required procedure is specified. Pressures, concentrated forces, moments, nodal displacements, nodal velocities, nodal velocities, and nodal temperatures can be selected within the step by load curve number. There are 13 procedures to select from. each procedure has a set of required parameters and additional options.

buckle for *buckle
ctd for *coupled temperature-displacement
dynamic for *dynamic
frequenc for *frequency
geostati for *geostatic
heat for *heat transfer
moddyn for *modal dynamic
random for *random responce
response for *response spectrum
soils for *soils
static for *static
ssdyn for *steady state dynamic
visco for *visco

The procedure definition is followed by step_options which can be:

amplitude *step* for stepped amplitude
amplitude *ramp* for ramped amplitude
cycle *#_inter* for maximum iterations in an increment
inc *#_inter* for maximum increments in a step
linear new for linear analysis with a new stiffness matrix
linear old for linear analysis with old stiffness matrix

monotoni for monotonic
nlgeom for geometric nonlinearity
rottol *tol* for maximum increment of rotation
submax for suppress subdivisions
abdload for face loads
abcload for nodal loads

ansyopts ANSYS analysis option

ansyopts *kan analysis_type kay **key** option cgloc x y z domega fx fy fz omega vx vy vz acel fx fy fz*

where *analysis_type* can be
-1 for heat transfer
0 for static analysis
1 for buckling analysis
2 for modal analysis
3 for full harmonic response
4 nonlinear transient analysis
5 for linear transient dynamic analysis
6 for reduced harmonic response

Remarks

The values of "**key**" and "**option**" depend on the type of analysis to be performed (see the kay command in the ANSYS user's manual or use a dialogue box).

The ANSYS kay commands and options are also built into this command, as well as global accelerations, velocities and coordinates.

dynaopts DYNA3D analysis options

dynaopts *options*

where any or all of the following *options* can be invoked:
iif interval
stsm *factor*
rfpf *0_or_1*

defpf *0_or_1*
edsdf *0_or_1*
gvst a*cceleration direction mat_1 mat_2 ...* ;
density_1 depth_1 density_2 depth_2 ... ;
stat *interval*
ticsf *time*
yldb *yield*
hiteb *distance*
xb0 *x*
yb0 *y*
zb0 *z*
tb0 *time*
lcb1 *load_curve*
lcb2 *load_curve*
clb *factor*
ctb *factor*
cpb *factor*
ticsf *interval*
ngrav *x_acceleration load_curve*
y_acceleration load_curve
z_acceleration load_curve
xvel *x_velocity load_curve*
yvel *y_velocity load_curve*
zvel *z_velocity load_curve*
term *time*
prti *interval*
plti *interval*
nrest *time_step*
nrunr *time_step*
itss *time_step*
pnlt *factor*
iteo *load_curve*
teo *load_curve*
where a nonpositive value for load curve means
0 for no thermal effects
-1 for nodal temperatures in topaz3d generated plot files
-2 for nodal temperatures use temperature option 2
-9999 for nodal temperatures use temperature option 1
tssf *factor*
lcmax *load_curve*
ssdm (no arguments)
snrs *option*

where *option* is the number of time steps between computations or

-2 for unique nodal fibers

-1 to compute normals each time step

1 compute on restarts

stup (no arguments)

sfor *option*

where the *option* can be

hl for hughes-liu

bt for belytschko-tsay

bciz

c0

membrane

yase

tsmin *factor*

itrx *#_iterations*

tolrx *tolerance*

facrx *factor*

scftrx *factor*

stss *option*

where the *option* can be

0 for characteristic length is area/longest side

1 for characteristic length is area/longest diagonal

2 based on bar wave, shortest side, area/longest side

plas *option*

where the *option* can be

1 for iterative plasticity with 3 secant iterations

2 for full iterative plasticity

3 for stress scaling noniterative plasticity

prtflg (no arguments)

drdb (no arguments)

rayl *alpha*

ihq *option*

where the *option* can be

1 for standard dyna3d (viscous form)

2 for flanagan-belytschko (viscous form)

3 for flanagan-belytschko with exact volume integration (viscous form)

4 for stiffness with flanagan-belytschko

5 for stiffness with flanagan-belytschko with exact volume

qh *coefficient*

q1 *coefficient*

Remarks

iif for time interval between writes of the interface segment save file
stsm for minimum time step size for thin shell element using the materials kinematic/isotropic elastic-plastic, strain rate dependent elastic-plastic, or rate dependent tabular isotropic elastic-plastic
stat for number of steps between problem status reports (default 1000)
rfpf to set the reaction force print flag
defpf to set the discrete (lumped parameter) element forces print flag
edsdf to set the element delete/sand database flag
gvst to initialize gravity stress
brode function yield in **ktons**
brode function height of **burst**
model x-coordinate of brode origin
model y-coordinate of brode origin
model z-coordinate of brode origin
model time of brode time origin
brode function arrival versus range load curve relative to brode origin
brode function yield scaling versus time load curve relative to brode
brode function origin divided by the cube root of the yield
brode function conversion factor kft to DYNA3D length units
brode function conversion factor milliseconds to DYNA3D time units
brode function conversion factor psi to DYNA3D pressure units
ticsf for time interval between output cross section forces
ngrav for prescribed base acceleration
xvel for angular velocity about the x-axis
yvel for angular velocity about the y-axis
zvel for angular velocity about the z-axis
term for termination time
prti for time interval between writes of time history node and element print block plot data
plti for time interval between writes of state plot database for all nodes and elements and, optionally, the interface force database containing pressures and shear tractions for all sliding interfaces
nrest for number of time steps between writes of the saved restart files
nrunr for number of time steps between writes of the continuously overwritten running restart file
itss for initial time step size
pnlt for global scale factor for sliding interface penalty stiffness controlling interpenetration and stability (default 0.1)
teo for thermal effects option
tssf for time step scale factor (default is 0.67 for high explosives, 0.90 otherwise)
lcmax for load curve number that limits maximum time step size (optional)
ssdm for write shell strain tensor at inner and outer surface
snrs for hughes-liu shell normal update option

stup for shell thickness change due to membrane straining
sfor for default shell element formulation which can be Hughes-Liu, Belytschko-Tsay, BCIZ, C0, membrane, and YASE
tsmin for reduction factor for initial time step size to determine minimum time step size. when this minimum time step is reached, DYNA3D terminates with a restart dump.
itrx for number of iterations between convergence checks for dynamic relaxation in quasistatic problems (default 250)
tolrx for convergence tolerance for dynamic relaxation option (default 0.0001)
facrx for dynamic relaxation static analysis velocity reduction factor (default 0.995)
scftrx for scale factor for computed time step during dynamic relaxation
stss for alternative methods for approximating the maximum stable time step size for 4-node shell elements
plas for plane stress constitutive integration algorithms for elastoplastic shell material models
prtflg for print element time step sizes on the first cycle
drdb for write the taurus database at every convergence check during dynamic relaxation
rayl for global generalized rayleigh damping mass proportional coefficient
ihq for select an hourglass stabilization method standard DYNA3D (viscous form), flanagan-belytschko (viscous form), flanagan-belytschko with exact volume integration (viscous form), stiffness with flanagan-belytschko, and stiffness with flanagan-belytschko with exact volume
qh for hourglass stabilization coefficient
q1 for quadratic bulk viscosity coefficient for added stability and resolution with shock waves
q2 for linear bulk viscosity coefficient for added stability and resolution with shock waves

lsdyopts LS-DYNA analysis and database options

lsdyopts *options* ;
where *options* can be

nosu	CONTROL_ACCURACY
inn	
pidosu *set_id*	
factin *initial_relaxation_factor*	CONTROL_ADAPSTEP
dfactr *incremental_increase*	
adpfreq *time*	CONTROL_ADAPTIVE
adptol *tolerance*	
adpopt *type*	

where *type* can be

1 - angle change in degrees per adaptive refinement relative to surrounding elements for each element to be refined.
2 - total angle change in degrees relative to the surrounding element for each element to be refined.
7 - 3D r-adaptive remeshing for solid elements.
8 - 2D r-adaptive remeshing for axisymmetric and plane strain solid elements

maxlvl *max_number_of_levels*
tbirth *time*
tdeath *time*
lcadp *load_curve*
gsam
mnelsz *size*
npss *number_of_passes*
ireflg *level*
adpene *distance*
adpth *thickness*
imem *percentage*
orient
maxel *number_of_elements*
dct *type* CONTROL_ALE

where *type* can be

1 - lagrangian (default)
2 - eulerian
3 - arbitrary lagrangian eulerian
4 - eulerian ambient

nadv *cycles*
meth *method*

where *method* can be

1 - donor cell + half index shitft (first order accurate)
2 - van leer + half index shift (second order)
3 - van leer

afac *weight*
bfac *weight*
cfac *weight*
dfac *weight*
efac *weight*
tbeg *time*
tend *time*
aafac *factor*
vfact *factor*
vlimit *limit*

ebc *flag*
where *flag* can be
0 Off
1 On with stick condition
2 On with slip condition
q1 *quadratic_viscosity_coefficient* CONTROL_BULK_VISCOSITY
q2 *linear_viscosity_coefficient*
ibq *type*
where *type* can be
-1 standard (also additional shels)
1 standard (default)
itsflg *flag* CONTROL_CFD_AUTO
where *flag* can be
0 IAUTO=1 for fixed time step size
1 Fixed time step based on DTINIT
2 Time step based on CFL/stability for INSOL=3
3 Automatic time step selection
epsdt *tolerance*
dtsf *scale_factor*
adtmax *time_step_size*
insol *solver_type* CONTROL_CFD_GENERAL
dtinit *initial_time_step*
cfl *maximum_advective_grid-CFL*
ickdt *Reynolds_and_advective_CFL_check_interval*
iacurc *accuracy_flag*
mimass *mass_matrix_formula* CONTROL_CFD_MOMENTUM
where mass_matrix_formula can be
0 IMASS=1 (default)
1 Lumped mass matrix
2 Consistent mass matrix
3 Higher-order mass matrix
iadvec *balancing_tensor_diffusivity_flag*
where *balancing_tensor_diffusivity_flag* can be:
0 IADVEC=10 for forward-Euler with BTD (default)
-1 IADVEC=0 for forward-Euler without BTD
10 forward-Euler with BTD
40 fully-implicit with simplified trapezoid rule
ifct *advective_flux_limiting_advection_scheme_toggle*
where *advective_flux_limiting_advection_scheme_toggle* can be
0 IFCT=1 (default)
1 Advective flux limiting is on
-1 Advective flux limiting is off

divu *RMS_divergence_tolerance*
thetak *viscous_terms_time_weighting*
thetaa *advection_terms_time_weighting*
thetaf *body_forces_time_weighting*
msol *momentum_equations_solver_type*
where *momentum_equations_solver_type* can be:

0	MSOL=20 (default)
20	Jacobi preconditioned conjugate gradient method
30	Jacobi preconditioned conjugate gradient squared method (default when IADVEC=40)

maxit *maximum_number_of_iterations*
ichkit *convergence_check_interval*
idiag *diagnostic_information_output_toggle*
ihist *convergence_history_file_generation_toggle*
eps *convergence_criteria*
ihg *hourglass_stabilization_type*
where *hourglass_stabilization_type* can be:

0	IHG=1 (default)
1	LS-DYNA CFD viscous hourglass stabilization
2	γ-hourglass stabilization viscous form

ehg *hourglass_stabilization_multiplier*
ipsol *pressure_solver_type* CONTROL_CFD_PRESSURE
where the *pressure_solver_type* can be:

0	IPSOL=22 for serial, IPSOL=21 for MPP (default)
10	Sparse direct solver
11	PVS direct solver
20	Jacobi preconditioned conjugate gradient method
21	SSOR preconditioned conjugate gradient method
22	SSOR preconditioned conjugate gradient using the Eisenstat transformation

maxitr *maximum_number_of_pressure_solver_iterations*
ichcit *convergence_criteria_check_interval*
idiag
ihst
epsp *convergence_criteria*
nvec *number_of_A-conjugate_vectors*
istab *stabilization_type*
where *stabilization_type* can be:

0	ISTAB=1
1	Local jump stabilization
2	Global jump stabilization
-1	No stabilization is active

pbeta *stabilization_parameter*

sid *set_id*
plev *hydrostatic_pressure_level*
plcid *hydrostatic_pressure_load_curve*
itemp *energy_equation_solver_type* CONTROL_CFD_TRANSPORT
nspec *number_of_species_transport_equations_activated*
imss *mass_matrix_formulation*
ibaltd *balancing_tensor_diffusivity_flag*
iaflx *advective_flux_limiting_flag*
thetk *viscous/diffusion_weighting_term*
thtaa *advection_term_time_weighting*
thetf *body_forces_time_weighting*
itsol *equation_solver_type*
mxiter *maximum_number_of_iterations*
ickint *convergence_criteria_check_interval*
idiagn *diagnostic_information_output_flag*
ichist *convergence_history_file_generation_flag*
epst *convergence_criteria*
ihgt *hourglass_stabilization_type*
ehgt *stabilization_parameter*
itrb *turbulence_model_flag* CFD_TURBULENCE
smagc *Smagorinsky_constant*
sn1-sn8 *optional_seed_nodes*
icoarse *coarsening_toggle* (CONTROL_COARSEN)
fangl *allowable_flatness_angle*
sn1-sn8 *optional_seed_nodes*
slsfac *sliding_interface_penalties_scale_factor* CONTROL_CONTACT
rwpnal *rigid_wall_penalties_scale_factor*
islchk *initial_penetration_check*
shlthk *shell_thickness*
penopt *penalty_stiffness_value_option*
thkchg *shell_thickness_changes*
orien *contact_interface_segment_reorientation_flag*
dkeep *flag*
usrstr *storage_per_contact_interface*
usrfrc *storage_per_contact_interface*
nsbcs *number_of_cylces*
interm *intermittent_searching_flag*
xpene *multiplier*
tfst *actual_shell_thickness_flag*
itftss *time_step_size_flag*
itfpsn *bypass_projection_flag*
dsfric *default_static_coefficient_of_friction*

ddfric *default_dynamic_coefficient_of_friction*
dedc *default_exponential_decay_coefficient*
dvfc *default_viscous_friction_coefficient*
dth *default_contact_thickness*
dthsf *thickness_scale_factor*
dpensf *default_local_penalty_scale_factor*
ignore *initial_penetrations_flag*
frceng *frictional_energy_flag*
unleng *factor* CONTROL_COUPLING
untime *factor*
unforc *factor*
timidl *time*
flipx *flag* (**0**-off**, 1**-on)
flipy *flag* (**0**-off**, 1**-on)
flipz *flag* (**0**-off**, 1**-on)
sybcyl *interval*
cputim *seconds* CONTROL_CPU
nrcyck *iterations* CONTROL_DYNAMIC_RELAXATION
drtol *tolerance*
drfctr *factor*
drterm *time*
tssfdr *factor*
irelal *flag* (**0**-off, **1**-on)
edttl *tolerance*
idrflg *flag*

where *flag* can be

1 - activate dynamic relaxation
2 - initialize to a prescribed geometry

hgen *flag* (**1**-off, **2**-on) CONTROL_ENERGY
rwen *flag* (**1**-off, **2**-on)
slnten *flag* (**1**-off, **2**-on)
rylen *flag* (**1**-off, **2**-on)
expsh CONTROL_EXPLOSIVE_SHADOW
ihq *type* CONTROL_HOURGLASS

where *type* can be

1 - standard ls-dyna3d
2 - flanagan-belytschko integration
3 - flanagan-belytschko integration with exact volume
4 - stiffness form of type 2 (flanagan-belytschko)
5 - stiffness form of type 3 (flanagan-belytschko)
6 - Belytschko-Bindeman [1993]
8 - Applicable to the type 16 fully integrated shell element

qh *coefficient*
n36flg
iautf *time_step_control_flag* CONTROL_IMPLICIT_AUTO
where *time_step_control_flag* can be:
0 - constant time step size
1 - automatically adjusted time step size
iteropt *optimum_iteration_count*
iterwin *allowable_iteration_window*
dtmini *minimum_allowable_time_step*
dtmaxi *maximum_allowable_time_step*
inal *flag* CONTROL_IMPLICIT_DYNAMICS
where *flag* can be:
0 static,
1 dynamic, Newmark
2 dynamic, modal
newgam *constant*
newbet *constant*
neig *number_of_eigenvalues* CONTROL_IMPLICIT_EIGENVALUE
center *frequency*
lflag *flag* (**0**-left end point is -infinity, **1**-left end point is lftend)
lftend endpoint
rflag *flag* (**0**-right end point is infinity, **1**-right end point is rhtend)
rhtend endpoint
eigmth flag (**1**-subspace iteration, **2**-block shift and Lanczos)
shfscl *scale*
imflag *flag* CONTROL_IMPLICIT_GENERAL
where flag can be:
0 explicit
1 implicit
2 explicit & implicit
dt0 *initial_implicit_time_step*
imform *flag* (**1**-fully integrated formulation, **2**-original formulation)
nsbs *number_of_springback_steps*
istress *flag* (**1**-include initial stress, **2**-ignore initial stress)
cnstn *indicator_for_consistent_tangent_stiffness* (**0**-do not use, **1**-use)
form *element_formulation* (**0**-type 16, **1**-type 6)
nsolvr *flag* CONTROL_IMPLICIT_SOLUTION
where *flag* can be
1 - linear
2 - nonlinear with BFGS updates (default)
3 - nonlinear with Broyden updates
4 - nonlinear with DFP updates

5 - nonlinear with Davidon updates
6 - nonlinear with BFGS updates + arclength
7 - nonlinear with Broyden updates + arclength
8 - nonlinear with DFP updates + arclength
9 - nonlinear with Davidon updates + arclength

ilimit *iteration_limit*
maxref *stiffness_reformation_limit*
dctoln *displacement_convergence_tolerance*
ectoln *energy_convergence_tolerance*
lstoln *line_search_convergence_tolerance*
dnorm *flag* (**1**-vs. current time step, **2**-vs. total)
diverg *flag* (**1**-reform stiffness, **2**-ignore divergence)
istif *flag* (**1**-reform stiffness at start of each step, **n**-reform at each "n"th step)
nlprt *flag* (**1**-print to screen and files, **2**-print to files)
arcctl *arc_length_control_node*
arcdir *flag* (**1**-global X-translation, **2**-global Y-translation, **3**-global Z-translation)
arclen *arc_length_size*
arcmth *flag* (**1**-Crisfield, **2**-Ramm)
arcdmp *flag* (**2**-off, **1**-on)
lsolvr *flag* CONTROL_IMPLICIT_SOLVER

where *flag* can be

1 - direct, sparse, incore (default)
3 - direct, sparse, double precision
4 - SMP parallel multi-frontal sparse solver #2
5 - SMP parallel multi-frontal sparse solver #2, double precision
6 - BCSLIB-EXT, direct, sparse, double precision
10 - iterative, best of currently available
11 - iterative, Conjugate Gradient method
12 - iterative, CG, Jacobi preconditioner
13 - iterative, CG, Choleski preconditioner
14 - iterative, Lanczos method
15 - iterative, Lanczos, Jacobi preconditioner
16 - iterative, Lanczos, Choleski preconditioner

lprint *flag*

where flag can be:

0 - no printing
1 - summary statistics on memory, cpu time, and iteration count
2 - more statistics
3 - even more statistics

negev *flag* (**1**-stop on negative eigenvalue, **2**-print warning, continue)
sorder *option* (**0**-method set automatically, **1**-MMD, **2**-Metis)
drcm *method* (**1**-add stiffness, **2**-generate geometry based drilling constraint, **3**-neither)

drcprm *constraint_parameter*
autospc *switch* (**1**-automatically scan, **2**-do not scan)
autotol *tolerance*
ias *flag* (**1**-active, **2**-inactive) CONTROL_IMPLICIT_STABILIZATION
ascale *stabilization_scale_factor*
strttim *stabilization_start_time*
endtime *stabilization_end_time*
nmem *percentage* CONTROL_NONLOCAL
npopt *flag* CONTROL_OUTPUT
where *flag* can be
0 - no suppression
1 - nodal coordinates, element connectivities, rigid wall definitions and initial velocities are not printed
neecho *flag*
where *flag* can be
0 - all data printed
1 - nodal printing is suppressed
2 - element printing is suppressed
3 - both node and element printing is suppressed
nrefup *flag* (**0**-no update, **1**-update)
iaccop *flag* (**0**-no average, **1**-averaged between output intervals)
ofifs *interval*
ipnint *flag* (**0**-print 100 elements with smallest time steps, **1**-governing time steps printed)
ikedit *interval*
iflush *number_of_time_steps*
iprtf *print_flag*
where *print_flag* can be:
0 - write part data into both MATSUM and RBDOUT
1 - write data into RBDOUT only
2 - write data into MATSUM file only
3 - do not write data into RBDOUT and MATSUM
ncpu *number_of_cpus* CONTROL_PARALLEL
numrhs *number* (**0** - same as ncpu, **1** - write only one)
iconst *flag* (**1**-on, **2**-off, for a faster solution)
ipllacc *flag* (**0**-off, **1**-on)
remin *minimum_edge_length* CONTROL_REMESHING
remax *maximum_edge_length*
lmf *flag* (**0**-explicit penalty, **1**-implicit with Lagrange multipliers) CONTROL_RIGID
jntf *stiffness* (0-incremental update, 1-total formulation)
orthmd *flag* (**0**-true, 1-false)
partm *flag* (**0**-true, 1-false)

sparse *flag* (**0**-false, 1-true)
wrpang *degrees* CONTROL_SHELL
itrist *flag* (**0**-no sorting required, **1**-full sorting)
irnxx *option*
where *option* can be
-2 - unique nodal fibers
-1 - compute normals each cycle
0 - default set to -1
1 - compute on restarts
n - compute every n cycles
istupd *flag* (**0**-no change, **1**-membrane straining causes thickness change)
theory *theory*
where *theory* can be
1 - Hughes-Liu
2 - Belytschko-Tsay (default)
3 - bciz triangular shell
4 - c0 triangular shell
5 - Belytschko-Tsay membrane
6 - s/r Hughes Liu
7 - s/r co-rotational Hughes Liu
8 - Englemann-Whirley shell
9 - fully integrated Belytschko-Tsay membrane
10 - Belytschko-Wong-Chiang
11 - fast (co-rotational) Hughes-Liu
12 - plane stress (x-y plane)
13 - plane strain (x-y plane)
14 - axisymmetric solid (y-axis of symmetry) - area weighted
15 - axisymmetric solid (y-axis of symmetry) - volume weighted
16 - fully integrated shell element
17 - discrete Kirchhoff triangular shell (DKT)
18 - discrete Kirchhoff linear shell either quad or triangular
20 - C^0 linear shell element with drilling stiffness
bwc *option*
where *option* can be
1 - Belytschko-Wong-Chiang warping stiffness added
2 - Belytschko-Tsay (default)
miter *option*
where *option* can be
1 - iterative plasticity with 3 secant iterations (default)
2 - full iterative plasticity
3 - radial return noniterative plasticity
shproj *flag* (**0**-drill projection, **1**-full projection)

rotascl *scale_factor*
intgrd *rule* (**0**-Gauss integration, **1**-Lobatto integration)
lamsht *flag* (**0**-do not update shear corrections, **1**-activate laminated shell theory)
esort *flag* (**0**-no sorting required, **1**-full sorting) CONTROL_SOLID
ianprc *procedure* CONTROL_SOLUTION
where *procedure* can be:
0 - Structural analysis only
1 - thermal analysis only
2 - coupled structural thermal analysis
4 - incompressible/low-Mach CFD analysis only
5 - coupled incompressible fluid-structure interaction
ncbs *number_of_cycles* CONTROL_SPH
boxid *id*
sphdt *death_time*
idim *dimension* (**3** - 3D problems, **2** - 2D problems, **-2** - 2D axisymmetric)
struct CONTROL_STRUCTURED
iterm
subcyl CONTROL_SUBCYCLE
endtim *time* CONTROL_TERMINATION
endcyc *cycle*
dtmin *factor*
endeng *percent_change*
endmas *percent_change*
mxmrts *number_time_steps* CONTROL_THERMAL_NONLINEAR
ctolt *tolerance*
divcp *value*
atype *type* CONTROL_THERMAL_SOLVER
where *type* can be:
0 - steady state analysis
1 - transient analysis
ptype *type*
where *type* can be:
0 - linear problem
1 - nonlinear problem with material properties evaluated at gauss point temperature
2- nonlinear problem with material properties evaluated at element average temp.
thslvr *type*
where *type* can be:
1 - actol: symmetric direct solver
2 - dactol: nonsymmetric direct solver
3 - dscg: diagonal scaled conjugate gradient iterative (default)
4 - iccg: Incomplete Choleski conjugate gradient iterative

cgtol *tolerance*
gpt *number* (0-default is set to 8, 1-one point quadrature is used)
eqheat *value*
fwork *fraction*
sbc *constant*
ktst *time_step_control* CONTROL_THERMAL_TIMESTEP
where *time_step_control* can be:
0 - fixed time step
1 - variable time step
tipt *parameter* (0.0-set to 0.5 - Crank-Nicholson scheme, 1.0-fully implicit)
itst *timestep*
tmint *timestep*
tmaxt *timestep*
dtempt *temperature_change*
tscpt *parameter*
dtinit *time_step_size* CONTROL_TIMESTEP
scft *scale_factor*
isdo *flag*
where *flag* can be:
0 - characteristic length = area/(min. of longest side or longest diagonal)
1 - characteristic length = area/(longest diagonal)
2 - based on bar wave speed
3 - timestep size based on maximum eigenvalue
tslimt minimum_time_step
dt2ms *time_step_size*
lctm *load_curve_id*
erode *flag* (0-no, 1-yes)
ms1st (0-no, 1-yes)
dt2msf *scale_factor*
iddlc *load_curve_ID* (DAMPING_GLOBAL)
valdmp *system_damping_constant*
stx *scale_factor_on_global_x_translational_damping*
sty *scale_factor_on_global_y_translational_damping*
stz *scale_factor_on_global_z_translational_damping*
srx *scale_factor_on_global_x_rotational_damping*
sry *scale_factor_on_global_y_rotational_damping*
srz *scale_factor_on_global_z_rotational_damping*
dmpid *material_id* (DAMPING_PART_MASS)
lcid *load_curve_id*
sflc *scale_factor_for_load_curve*
lcflg *separate_global_scale_factors_flag*
idstf *part_ID* (DAMPING_PART_STIFFNESS)

drayl *Rayleigh_damping_coefficient*
cdamp *fraction_of_critical_damping* (DAMPING_RELATIVE)
frq *frequency*
pidrb *material_id_rigid_body*
psid *material_set_id*

Remarks

nosu objective stress update flag CONTROL_ACCURACY
inn invariant node numbering for shell elements
pidosu part set id for objective stress updates
factin initial relaxation factor for contact force CONTROL_ADAPSTEP
dfactr incremental increase of factin

CONTROL_ADAPTIVE

adpfreq time interval between automatic remeshing refinements
adptol adaptive error tolerance
adpopt to set adaptive refinement criterion
maxlvl maximum refinement levels
tbirth time at which the adaptive remeshing begins
tdeath time at which the adaptive remeshing ends
lcadp to specify a load curve for the adaptive interval
gsam flag to generate adaptive mesh at exit
mnelsz minimum element size
npss one or two pass adaptivity flag
ireflg uniform refinement level
adpene adapt the mesh when the contact surfaces approach or penetrate tooling surface
adpth absolute shell thickness level
imem memory limit (as a function of memory environmental variable)
orient choose orientation of contact surface
maxel adaptivity is stopped if this number of elements is exceeded

CONTROL_ALE

dct to set the default continuum treatment to lagrangian, eulerian, ale, or eulerian ambient
nadv number of cycles between advections
meth advection method (donor cell, van leer + half index shift, van leer)
afac smoothing weight
bfac smoothing weight factor - volume weighting
cfac smoothing weight factor - isoparametric
dfac smoothing weight factor - equipotential
efac smoothing weight factor - equilibrium
tbeg start time for smoothing
tend end time for smoothing
aafac ale advection factor

vfact volume fraction limit for stresses
vlimit velocity limit
ebc automatic Euler boundary condition
q2 default quadratic viscosity coefficient CONTROL_BULK_VISCOSITY
q1 default linear viscosity coefficient
ibq default bulk viscosity type
itsflg set time step control type CONTROL_CFD_AUTO
epsdt set tolerance for local truncation error in time
dtsf set maximum time step scale factor
adtmax set upper limit on time step size
insol set the solver type CONTROL_CFD_GENERAL
dtinit set the initial time step for the Navier-Stokes and all transport equations
cfl set the maximum advective grid-CFL number
ickdt set interval to check and report grid Reynolds and advective CFL numbers
iacurc activate use of full-quadrature for certain terms in the momentum/transport eqns.
mimass select the mass matrix formulation CONTROL_CFD_MOMENTUM
iadvec toggle treatment of advection
ifct toggle use of advective flux limiting advection scheme
divu set RMS divergence tolerance
thetak time weighting for viscous/diffusion terms
thetaa time weighting for advection terms
thetaf time weighting for body forces and boundary conditions
msol set equation solver type for momentum equations
maxit set maximum number of iterations for iterative equations solver
ichkit set interval to check convergence criteria for iterative equation solver
idiag activate output of diagnostic information
ihist activate generation of a convergence history file
eps set convergence criteria for iterative equation solver
ihg set type of hourglass stabilization to be used with momentum equations
ehg set hourglass stabilization multiplier
ipsol set pressure solver type CONTROL_CFD_PRESSURE
maxitr set maximum number of iterations for pressure solver
ichcit set interval to check convergence criteria for pressure solver
idiag activate output of diagnostic information from pressure solver
ihst activate generation of convergence history file for pressure solver
epsp set convergence criteria for pressure solver
nvec set number of A-conjugate vectors to use during iterative pressure solve
istab set stabilization type
pbeta stabilization parameter for certain type values (1 and 2)
sid solid or shell element set ID
plev set hydrostatic pressure level
plcid load curve to be used for setting hydrostatic pressure

itemp solve energy equation in terms of temperature CONTROL_CFD_TRANSPORT
nspec activate solution of NSPEC species transport equations
imss select mass matrix formulation to use
ibaltd toggle treatment of advection between explicit with balancing tensor diffusivity (BTD) or fully-implicit
iaflx toggle use of advective flux limiting advection scheme
thetk time weighting for viscous/diffusion terms
thtaa time weighting for advection terms
thetf time weighting for body forces and boundary conditions
itsol set equation solver type for transport equations
mxiter set maximum number of iterations for iterative equation solver
ickint set interval to check the convergence criteria for iterative equation solver
idiagn activate output of diagnostic information from equation solver
ichist activate generation of convergence history file from equation sover
epst set convergence criteria iterative equation solver
ihgt set type of hourglass stabilization to be used with momentum equations
ehgt set hourglass stabilization multiplier
itrb select the turbulence model CONTROL_CFD_TURBULENCE
smagc Smagorinsky constant
icoarse coarsening flag CONTROL_COARSEN
fangl allowable angle change between neighboring elements
sn1-**sn8** optional seen node numbers
slsfac scale factor for sliding interface penalties CONTROL_CONTACT
rwpnal scale factor for rigid wall penalties for treating rigid bodies interacting with fix rigid walls
islchk enable and disable initial penetration check in contact surfaces
shlthk set options for considering shell thickness in surface-to-surface and node-to-surface contact
penopt set penalty stiffness value options
thkchg set option to consider shell thickness in single-surface contact
orien enable (disable) automatic reorientation of contact interface segments during initialization
dkeep treatment of the mass of eroded nodes in contact
usrstr storage per contact interface for user supplied interface control subroutine
usrfrc storage per contact interface for user supplied interface friction subroutine
nsbcs number of time steps between contact searching using 3d bucket searches in single-surface and the new surface-to-surface contact
interm flag for intermittent searching in old surface-to-surface contact using the interval specified as **nsbcs** above
xpene contact surface maximum penetration check multiplier
tfst flag for using actual shell thickness in single surface contact logic-types 4, 13, 15 & 26
itftss time step size override for eroding contact

itfpsn bypass projection of slave nodes to master surfaces in certain contact types
dsfric default static coefficient of friction
ddfric default dynamic coefficient of friction
dedc default exponential decay coefficient
dvfc default viscous friction coefficient
dth default contact thickness
dthsf default thickness scale factor
dpensf default local penalty scale factor
ignore ignore initial penetrations in the *CONTACT_AUTOMATIC options
frceng flag to activate calculation of frictional sliding energy

CONTROL_COUPLING

unleng unit conversion factor for length. madymo3d/gm-cal3d lengths are multiplied by **unleng** to obtain LS-DYNA lengths
untime unit conversion factor for time. madymo3d/gm-cal3d time is multiplied by **untime** to obtain LS-DYNA time.
unforc unit conversion factor for force. madymo3d/gm-cal3d force is multiplied by unforc to obtain LS-DYNA force.
timidl idle time during which cal3d or madymo is computing and LS-DYNA remains inactive.
flipx flag for flipping x-coordinate of cal3d/madymo3d relative to LS-DYNA model
flipy flag for flipping y-coordinate of cal3d/madymo3d relative to LS-DYNA model
flipz flag for flipping z-coordinate of cal3d/madymo3d relative to LS-DYNA model
sybcyl cal3d/madymo3d subcycling interval (in cycles)
cputim limit (in seconds) of cpu time CONTROL_CPU

CONTROL_DYNAMIC_RELAXATION

nrcyck iterations between convergence checks for dynamic relaxation
drtol convergence tolerance for dynamic relaxation option
drfctr dynamic relaxation factor
drterm optional termination time for dynamic relaxation
tssfdr scale factor for computed time step during dynamic relaxation
irelal automatic control for dynamic relaxation option based on algorithm papadrakakis
edttl convergence tolerance on automatic control of dynamic relaxation
idrflg dynamic relaxation flag for stress initialization (inactive, dynamic relaxation is active, initialization to a prescribed geometry).
hgen set option for calculating hourglass energy CONTROL_ENERGY
rwen set option for calculating stonewall energy dissipation
slnten set option for calculating sliding interface energy dissipation
rylen set option for calculating Rayleigh energy dissipation

CONTROL_EXPLOSIVE_SHADOW

expsh flag to include the *CONTROL_EXPLOSIVE_SHADOW card
ihq specifies hourglass viscosity type CONTROL_HOURGLASS
qh default hourglass coefficient

iautf sets automatic time step control flag CONTROL_IMPLICIT_AUTO
iteropt set optimum equilibrium iteration count per time step
iterwin set the allowable iteration window
dtmini set the minimum allowable time step size
dtmaxi set the maximum allowable time step size
inal set the implicit analysis type CONTROL_IMPLICIT_DYNAMICS
newgam Newmark time integration constant, gamma
newbet Newmark time integration constant, beta
neig number of eigenvalues to extract CONTROL_IMPLICIT_EIGENVALUE
center center frequency
lflag left end point finite flag
lftend left end point of interval (when finite flag, **lflag**, is set)
rflag right end point finite flag
rhtend right end point of interval (when finite flag, **rflag**, is set)
eigmth eigenvalue extraction method
shfscl shift scale
imflag set implicit/explicit switching flag CONTROL_IMPLICIT_GENERAL
dt0 set the initial time step size for implicit analysis
imform set the element formulation switching flag
nsbs set the number of steps in nonlinear springback
istress set the geometric stiffness flag
cnstn indicator for consistent tangent stiffness
form element formulation when using **imform** flag
CONTROL_IMPLICIT_SOLUTION
nsolvr define the nonlinear solution method for implicit analysis
ilimit set the iteration limit between automatic stiffness reformations
maxref stiffness reformation limit per time step
dctoln displacement convergence tolerance
ectoln energy convergence tolerance
lstoln line search convergence tolerance
dnorm set the displacement norm for convergence test
diverg set the divergence flag (force imbalance increase during equilibrium iterations)
istif set the initial stiffness formation flag
nlprt set the nonlinear solver print flag
arcctl arc length controlling node ID
arcdir set the arc length controlling node direction
arclen arc length size
arcmth set the arc length method
arcdmp set the arc length damping option
lsolvr define the linear equation solver method CONTROL_IMPLICIT_SOLVER
lprint set the linear solver print flag
negev set the negative eigenvalue flag

sorder ordering option
drcm drilling rotation contraint method
drcprm drilling rotation constraint paramter
autospc AUTOSPC switch
autotol tolerance used with the previous switch
ias set the artificial stabilization flag CONTROL_IMPLICIT_STABILIZATION
ascale scale factor for artificial stabilization
strttim implicit stabilization start time
endtim implicit stabilization end time

CONTROL_NONLOCAL

nmem percentage increase of memory allocated for MAT_NONLOCAL option over that required initially
npopt set option for print suppression during input phase CONTROL_OUTPUT
neecho set option for print suppression during input phase for echo file
nrefup set option to update reference node coordinates for beam elements
iaccop set option to average accelerations from velocities in file "nodout" and time history file "d3thdt"
ofifs time interval for interface file
ipnint set option to print initial time step sizes for all elements on the first cycle
ikedit problem status report interval steps
iflush number of time steps interval for flushing I/O buffers. Default value is 5000.
iprtf default print flag for RBDOUT and MATSUM files.
ncpu number of cpu's used CONTROL_PARALLEL
numrhs number of right-hand-sides written
iconst consistency flag for parallel solution
ipllacc flag for parallel force assembly if **iconst**=1

CONTROL_REMESHING

remin minimum edge length for the surface mesh surrounding parts which should be remeshed
remax maximum edge length for the surface mesh surrounding parts which should be remeshed
lmf set the Lagrange multiplier flag CONTROL_RIGID
jntf set the generalized joint stiffness flag
orthmd orthogonalize modes with respect to each other
partm use global mass matrix to determine part mass distribution
sparse use sparse matrix multiply subroutines for the modal stiffness and damping matrices
wrpang limit (in degrees) for shell warpage CONTROL_SHELL
itrist option to set automatic sorting of triangular shell elements to treat degenerate quadrilateral shell elements at c0 triangular shells
irnxx sets the Hughes-Liu shell normal update option
istupd sets option to allow membrane straining to cause thickness change in shells
theory sets the shell theory option (Hughes-Liu, Belytschko-Tsay, etc.)

bwc sets warping stiffness option for Belytschko-Tsay shells
miter sets plane stress plasticity option (materials 3, 18, 19, 24 only)
shproj projection method for warping stiffness in the Belytschko-Tsay shell and Belytschko-Wong-Chiang elements
rotascl define a scale factor for the rotary shell mass
intgrd default shell through thickness numerical integration rule
lamsht laminated shell theory flag

CONTROL_SOLID

esort automatic sorting of tetrahedron and pentahedron elements to treat degenerate hexahedron elements as tetrahedron and pentahedron solids, respectively
ianprc analysis solution procedure CONTROL_SOLUTION
ncbs number of cycles between particle sorting CONTROL_SPH
boxid SPH approximations are computed inside a specified BOX
sphdt death time, the time that SPH calculations are stopped
idim space dimensions for SPH particles

CONTROL_STRUCTURED

struct flag to write out a structured input deck for Version 960
iterm flag to terminate after the structured input file is written
subcyl control time step subcycling CONTROL_SUBCYCLE
edntim termination time CONTROL_TERMINATION
endcyc termination cycle
dtmin reduction (scale) factor for initial time step size to determine minimum time step
endeng percent change in energy ration for termination of calculation
endmas percent change in the total mass for termination of calculation

CONTROL_THERMAL_NONLINEAR

mxmrts maximum number of matrix reformations per time step
ctolt convergence tolerance for temperature
divcp divergence control parameter
atype thermal analysis type CONTROL_THERMAL_SOLVER
ptype thermal problem type
thslvr thermal analysis solver type
cgtol convergence tolerance for solver types 3 and 4
gpt number of Gauss points to be used in the solid elements
eqheat mechanical equivalent of heat
fwork fraction of mechanical work converted into heat
sbc Stefan Boltzmann constant
ktst time step control CONTROL_THERMAL_TIMESTEP
tipt time integration parameter
itst initial thermal time step
tmint minimum thermal time step
tmaxt maximum thermal time step

dtempt maximum temperature change in each time step above which the thermal timestep will be decreased
tscpt time step control parameter
dtint initial time step size CONTROL_TIMESTEP
scft scale factor for computed time step
isdo basis of time size calculation for 4-node shell elements
tslimt shell element minimum time step assignment
dt2ms time step size for mass scaled solutions
lctm load curve number that limits maximum time step size
erode sets erosion option for solid shell elements
ms1st sets option to limit mass scaling to the first step and fix mass vector afterwards
dt2msf scale factor for initial time step size to determine the minimum time step size permitted

iddlc load curve ID which specifies node system damping DAMPING_GLOBAL
valdmp system damping constant, d
stx scale factor on global x translational damping forces
sty scale factor on global y translational damping forces
stz scale factor on global z translational damping forces
srx scale factor on global x rotational damping forces
sry scale factor on global x rotational damping forces
srz scale factor on global x rotational damping forces
dmpid damping part ID DAMPING_PART_MASS
lcid load curve ID which specifies system damping for parts
lcsf load curve scale factor
icflg set the flag to unity if the global components of damping require separate scaling
idstf part ID for stiffness DAMPING_PART_STIFFNESS
drayl Rayleigh damping coefficient

marcopts MARC analysis options

marcopts *options*

where any or all of the following options can be invoked:

maxall *number_of_words*

cpu *number_of_cpus*

vlength *vector_length*

bmpar (no arguments)

nolist (no arguments)

elastic (no arguments)

linear *flag*

where *flag* can be

wo to save the beta matrix (strain-displacement)

w to save the beta matrix and the stress-strain law

fourier (no arguments)

nfexp *number_expansions*

mxharm *max_harmonics*

nldff *number_degrees*

frsy *flag*

where the *flag* can be

-2 for only antisymmetric and skip increments 0, 1, and 2

-1 for only symmetric (cos) and skip increments 0 and 1

0 for full expansion

1 for only symmetric

2 for only antisymmetric

mxst *max_expansions*

imfss (no arguments)

dyop *option*

where the *option* can be

modal

newmark

houbolt

central

mxmo *max_superpositions*

lancz (no arguments)

msrec (no arguments)

harm (no arguments)

cxdp (no arguments)

xcit *number_excitations*

xdll *max_lists*

nelx *max_lists*

inrt (no arguments)

sdpf *number_points*
spra (no arguments)
rpfl (no arguments)
ldisp (no arguments)
uplgr (no arguments)
fspl (no arguments)
folf (no arguments)
buck *max_modes number_modes recovery_flag*
where the *flag* can be yes (**y**) or no (**n**)
creep *type kelvin_flag maxwell_flag*
where the *type* can be
0 for normal creep
1 for viscoplastic creep
2 for viscoplastic creep with non-associative flow rule
where the flag can be yes (**y**) or no (**n**)
visco (no arguments)
couple (no arguments)
pore *type analysis_flag*
where *type* can be
0 for pore pressure data to be entered
1 to perform steady state pore pressure calculation
2 to perform transient pore pressure calculation
where the *flag* can be yes (**y**) or no (**n**)

Remarks

maxall to specify the size of work space vector
cpu to specify the number of cpus
vlength for the optimal vector length
bmpar to select the beta matrices to be formed in parallel
nolist to suppress printout of the remainder of the input
elastic for elastic analysis with multi-loads
linear for matrices saved for linear analysis
fourier for arbitrary loading of axisymmetric structures
nfexp for the number of fourier expansions
mxharm for the maximum number of harmonics
nldff for the number of load degrees of freedom in fourier expansion
frsy for the expansion symmetry flag with the options for
only antisymmetric (sin) terms are present in the expansions and skip increments 0, 1, and 2
only symmetric (cos) terms are present in the expansion and skip increments 0 and 1

full expansion containing sine and cosine terms
only symmetric (cos) terms are present in the expansion
only antisymmetric (sin) terms are present in the expansions

mxst for the maximum number of fourier expansion stations around circumference
imfss for initial stress stiffness in modal fourier expansion
dyop for dynamic analysis with the options
modal for modal superposition dynamic response
newmark for newmark direct integration
houbolt for houbolt direct integration
central for explicit direct time integration using central difference
mxmo for maximum number of modes used in the modal superposition
lancz to select lanczos method
msrec for modal stress recovery or storing eigenvectors
harm for frequency response analysis
cxdp to use the complex damping matrix
xcit for the maximum number of excitation boundary conditions
xdll for the maximum number of excitation distribution load lists
nelx for the maximum number of elements in any excitation distribution list
inrt to include inertia effects in calculation of harmonic reaction force
sdpf for the number of data points used to specify spectral density function
spra for spectrum response analysis
rpfl to select rigid, perfectly-plastic flow flag
ldisp to select large displacement or buckling flag
uplgr for updated lagrange procedure
fspl to select finite strain plasticity
folf to select follower forces: distributed load application on current geometry
buck to select buckling load estimate via eigenvalue analysis
creep to select creep analysis
visco to select visco elastic analysis (kelvin model)
couple to select coupled thermal-stress analysis
pore to select soil analysis

nastopts NASTRAN v68 analysis options

Not supported yet.

nekopts 2d and 3d NEKTON 2.85 analysis options

k**tflow** to set the switch ifflow
ktheat to set the switch ifheat
kttran to set the switch iftran
ktnavn to set the switch ifnav & ifadvc where n is the sequence number of the flag from 1 to 11
kttmshn to set the switch iftmsh where n is the sequence number of the flag from 1 to 12
ktaxis to set the switch ifaxis
ktstrs to set the switch ifstrs
ktsplit to set the switch ifsplit
ktmgrid to set the switch ifmgrid
ktmodel to set the switch ifmodel
ktkeps to set the switch ifkeps
ktmvbd to set the switch ifmvbd
ktchar to set the switch ifchar
ktvsht to set the switch ifvsht
kttlog to set the switch iftlog
ktdirs to set the switch ifdirs
ktptbg to set the switch ifptbg
ktcomp to set the switch ifcomp
aneden to set the parameter density
anevis to set the parameter viscos
anebet to set the parameter betag
anegth to set the parameter gtheta
anerho to set the parameter rhocp
anecon to set the parameter conduct
aneqvo to set the parameter qvol
anefin to set the parameter fintime
anenst to set the parameter nsteps
anedt to set the parameter dt
aneiot to set the parameter iotime
aneios to set the parameter iostep
anenor to set the parameter norder
anetlr to set the parameter tolrel
anetla to set the parameter tolabs
anecou to set the parameter courant
anetor to set the parameter torder
anevhs to set the parameter vhscalefac
ktifxyo to set the switch to output coordinates
ktifvo to set the switch to output velocity
ktifpo to set the switch to output pressure
ktifto to set the switch to output temperature
ktiftgo to set the switch to output temperature gradient

nikeopts NIKE3D analysis options

nikeopts *options*

where any or all of the following options can be invoked:
accflg *option* where the *option* can be
0 for no acceleration in the plot file
1 to include relative acceleration data
2 to include absolute acceleration data
altol *tolerance*
anal *type*
where *type* can be **stat**, **dyn**, or **dyns**
arcl *arc_length*
arclcm *method*
where *method* can be **crisfield** or **ramm**
arcldamp (no arguments)
auto (no arguments)
bef *flag*
where *flag* can be **0** for neglect, **1** for include
begs *option*

where *option* can be
0 for hughes-lui out-of-core
1 for hughes-lui in-of-core
bfgscore (no arguments)
bfor *formulation*
where *formulation* can be **bbar** or **bbarwi**
brstif (no arguments)
bwmo *toggle*
where *toggle* can be **on** or **off**
dctol *tolerance*
delt *time_step*
dispnode *node_#*
dispdir *direction*
where *direction* can be **1**, **2**, or **3**
ectol *tolerance*
ngrav *x_acceleration load_curve*
y_acceleration load_curve
z_acceleration load_curve
ictol *tolerance*
ilsbuf *buffer_size*
iobuf *buffer_size*

iplt *dump_interval*
iprt *dump_interval*
lsolver *method*
where *method* can be **fissle**, **scaling**, **crout**, **gs**, or **cholesky**
lstol *tolerance*
mem *percent*
mnss *min_step*
msrf *max_#_step*
munload *method*
where *method* can be **bfgs**, **broy**, **dfp**, **dav**, or **mnewt**
mxitls *max_#_iterations*
mxnre *max_#_retries*
mxss *max_step_size*
nbei *#_steps*
nbsr *#_steps*
neig *#_eigenvectors*
nibsr *max_#_iterations*
nip1 *coefficient*
nip2 *coefficient*
noarclda (no arguments)
nrest *#_steps*
nsbrr *#_steps*
nsmd *method*
where *method* can be **bfgs**, **broy**, **dfp**, **mdav**, **mnewt**, **marc**, **mcls, mabfgs, mabroy, madfp, mambfgs**, **madav**, **newt**, **newtls**
nsteps *#_steps*
nunload *#_steps*
opnit *#_iterations*
prlis *flag*
where *flag* is an integer from **0** to **4**
retol *tolerance*
segs *flag*
where *flag* can be **0** for neglect, **1** for include
sfor *formulation*
where *formulation* can be
hl for hughes-liu
bt for belytschko-tsay
yase
shift *freqency*
ssdm (no arguments)
stifcore (no arguments)
sw3 (no arguments)

sw6 (no arguments)
sw7 (no arguments)
teo *value*
term *time*
xvel *load*
yvel *load*
zvel *load*

Remarks

accflg to select the acceleration data dump option
altol to set the convergence tolerance on augmented lagrangian
anal with the options
    stat for static
    dyn for dynamic analysis
    dyns for dynamic analysis with stresses initialized statically
arcl for desired arc length
arclcm for the arc length constraint method which can be
    crisfield
    ramm
arcldamp for arc length damping
auto for automatic time step control invoked
bef to select the beam element formulation
begs to set the beam element geometric stiffness flag
bfgscore for bfgs update vectors storage option
bfor for brick element formulation which can be
    bbar for b-bar
    bbarwi for b-bar with incompatible modes
brstif for brick element geometric stiffness included
bwmo for bandwidth minimization on/off switch
dctol for displacement convergence tolerance
delt for time step
dispnode for node number for displacement controlled arc length method
dispdir for direction of displacement at node for arc length control
    1 is for the global x-direction
    2 is for the global y-direction
    3 is for the global z-direction
ectol for energy convergence tolerance
ngrav for gravity
ictol for iteration convergence tolerance
ilsbuf for out of core linear solver buffer size
iobuf for buffer size (words) for element data i/o

iplt for tuarus dump interval
iprt for print dump interval
positive number causes printing each interval excluding step 0 0 causes printing each step including step 0 negative number causes printing each interval including step 0
itpro for the temperature profile load curve number
lsolver for linear equation solver which can be
fissle for direct solution with fissle
scaling for iterative solution with diagonal scaling
crout for iterative solution with crout element-by-element
gs for iterative solution with gauss-seidel ebe
cholesky for terative solution with cholesky ebe
lstol for line search convergence tolerance
mem for maximum memory
mnss for minimum allowable step size
msrf for maximum number of reform/time steps
munload for arc length unloading method which can be
bfgs for bfgs
broy for broyden
dfp for davidon-fletcher-powell
dav for davidon
mnewt for modified newton
mxitls for iteration limit for linear solver
mxnre for maximum number of retries allowable per step
mxss for maximum allowable step size
nbei for number of steps between equilibrium iterations
nbsr for number of steps between matrix reformations
nsbrr for the number of time steps between running restarts
neig for number of eigenvectors
nibsr for maximum number of equilibrium iter./matrix reform.
nip1 for first newmark integration parameter
nip2 for second newmark integration parameter
noarclda for no arc length damping
nrest for number of time steps between restart file generation
nsmd for nonlinear solution method which can be
bfgs for bfgs (default)
broy for broyden
dfp for davidon-fletcher-powell
mdav for modified davidon
mnewt for modified newton
MARC for modified constant arc length
mcls for modified constant arc length with line search

mabfgs for modified constant arc length with bfgs
mabroy for modified constant arc length with broyden
madfp for modified constant arc length with dfp
mambfgs for modified constant arc length with modified bfgs
madav for modified constant arc length with davidon
newt for full newton
newtls for full newton with line search

nsteps for number of time steps
nunload for number of unloading steps in modified arc length method
opnit for optimal number of iterations per step
prlis for linear iterative solver print-out flag with values
0 for no inner loop information
1 for time step convergence information
2 for iteration norm information
3 for residual input, solution output
4 for residual/solution each iteration

retol to set the convergence tolerance on residual norm
segs to set the shell element geometric stiffness flag
sfor for shell element formulation which can be
hl for hughes-liu
yase for yase
bt for belytschko-tsay (DYNA3D-like)

shift for freqency shift
ssdm for shell surface strain data dumps
stifcore for stiffness matrix storage option
sw3 for toggle the default sense switch number 3
sw6 for toggle the default sense switch number 6
sw7 for toggle the default sense switch number 7
teo for thermal effects option
xvel for load due to x-angular velocity
yvel for load due to y-angular velocity
zvel for load due to z-angular velocity

lsnkopts LS-NIKE3D analysis options

lsnkopts *options*

where any or all of the following options can be invoked:
accflg *option*
where the *option* can be

0 for no acceleration in the plot file
1 to include relative acceleration data
2 to include absolute acceleration data

altol *tolerance*

anal *type*

where *type* can be **stat**, **dyn**, or **dyns**

arcl *arc_length*

arclcm *method*

where *method* can be **crisfield** or **ramm**

arcldamp (no arguments)

auto (no arguments)

bef *flag*

where *flag* can be **0** for neglect, **1** for include

begs *option*

where *option* can be

0 for hughes-lui out-of-core
1 for hughes-lui in-of-core

bfgscore (no arguments)

bfor *formulation*

where *formulation* can be **bbar** or **bbarwi**

brstif (no arguments)

bwmo *toggle*

where *toggle* can be **on** or **off**

dctol *tolerance*

delt *time_step*

dispnode *node_#*

dispdir *direction*

where *direction* can be **1**, **2**, or **3**

ectol *tolerance*

ngrav *x_acceleration load_curve*
y_acceleration load_curve
z_acceleration load_curve

ictol *tolerance*

ilsbuf *buffer_size*

iobuf *buffer_size*

iplt *dump_interval*

iprt *dump_interval*

lsolver *method*

where *method* can be **fissle**, **scaling**, **crout**, **gs**, or **cholesky**

lstol *tolerance*

mem *percent*

mnss *min_step*

msrf *max_#_step*
munload *method*
where *method* can be **bfgs**, **broy**, **dfp**, **dav**, or **mnewt**
mxitls *max_#_iterations*
mxnre *max_#_retries*
mxss *max_step_size*
nbei *#_steps*
nbsr *#_steps*
neig *#_eigenvectors*
nibsr *max_#_iterations*
nip1 *coefficient*
nip2 *coefficient*
noarclda (no arguments)
nrest #_steps
nsbrr #_steps
nsmd *method*
where *method* can be **bfgs**, **broy**, **dfp**, **mdav**, **mnewt**, **marc**, **mcls, mabfgs, mabroy**, **madfp**, **mambfgs**, **madav**, **newt**, **newtls**
nsteps *#_steps*
nunload *#_steps*
opnit *#_iterations*
prlis *flag*
where *flag* is an integer from **0** to **4**
retol *tolerance*
segs *flag*
where *flag* can be **0** for neglect, **1** for include
sfor *formulation*
where *formulation* can be
hl for hughes-liu
bt for belytschko-tsay
yase
shift *freqency*
ssdm (no arguments)
stifcore (no arguments)
sw3 (no arguments)
sw6 (no arguments)
sw7 (no arguments)
teo *value*
term *time*
xvel *load*
yvel *load*
zvel *load*

Remarks

anal with the options
- **stat** for static
- **dyn** for dynamic analysis
- **dyns** for dynamic analysis with stresses initialized statically

arcl for desired arc length
arclcm for the arc length constraint method which can be
- **crisfield**
- **ramm**

arcldamp for arc length damping
auto for automatic time step control invoked
brstif for brick element geometric stiffness included
bwmo for bandwidth minimization on/off switch
dctol for displacement convergence tolerance
delt for time step
dispnode for node number for displacement controlled arc length method
dispdir for direction of displacement at node for arc length control
- **1** is for the global x-direction
- **2** is for the global y-direction
- **3** is for the global z-direction

dsfct for force vs. displacement table displacement scale factor
dtroc for type of force recovery resulting from enforced boundary conditions
ectol for energy convergence tolerance
frqsht for frequency shift, cycles per unit time
fsfct for force vs. displacement table force scale factor
ngrav for gravity
icnt1 for initial stiffness matrix formulation frequency
icnt2 for equilibrium iteration frequency
icnvrg for displacement criteria to check convergence
idynrst for number of materials to initialize stresses in LS-DYNA3D
ifb for output force balance table
ifdof for degree-of-freedom for node in force vs. displacement table
iffd for output force vs. displacement table
ifpri for force dump print interval
infnm1 for interface number to end first phase
infnm2 for interface number to end second phase
initstr for read initial element stresses
iobuf for buffer size (words) for element data i/o
ioro for quasi-newton update vectors storage
iplt for tuarus dump interval

iprt for print dump interval
positive number causes printing each interval excluding step 0 0 causes printing each step including step 0 negative number causes printing each interval including step 0
islvdt for linear equation solver option
lauto for parameter to control of increase in the time step
lclod1 for load curve for first phase
lclod2 for load curve for second phase
lstol for line search convergence tolerance
mnss for minimum allowable step size
msrf for maximum number of reform/time steps
munload for arc length unloading method which can be
bfgs for bfgs
broy for broyden
dfp for davidon-fletcher-powell
dav for davidon
mnewt for modified newton
memsol for percentage of memory for the solver
mxnre for maximum number of retries allowable per step
mxss for maximum allowable step size
nbei for number of steps between equilibrium iterations
nbsr for number of steps between matrix reformations
ncycsp for number of cycles for monotonic pressure after reversal
ndpt for node number for force vs. displacement table
necho for nodal input data echo flag
neig for number of eigenvectors
nibsr for maximum number of equilibrium iter./matrix reform.
nip1 for first newmark integration parameter
nip2 for second newmark integration parameter
npsd for discrete element input flag
nrest for number of time steps between restart file generation
nsmd for nonlinear solution method which can be
bfgs for bfgs (default)
broy for broyden
dfp for davidon-fletcher-powell
mdav for modified davidon
mnewt for modified newton
marc for modified constant arc length
mcls for modified constant arc length with line search
mabfgs for modified constant arc length with bfgs
mabroy for modified constant arc length with broyden
madfp for modified constant arc length with dfp

mambfgs for modified constant arc length with modified bfgs
madav for modified constant arc length with davidon
newt for full newton
newtls for full newton with line search
nsteps for number of time steps
nunload for number of unloading steps in modified arc length method
nusrldp for number of user loading subroutine parameters
n1decho for beam element input data echo flag
n2decho for shell element input data echo flag
n3decho for solid element input data echo flag
opnit for optimal number of iterations per step
penldc for load curve defining multiple vs. time
penstab for penalty stiffness multiplier for artificial stabilization
sclmin for minimum pressure load scale factor
sclmax for maximum pressure load scale factor
sprcnt1 for percentage of nodes in contact to end first phase
sprcnt2 for percentage of nodes in contact to end second phase
ssdm for shell surface strain data dumps
stifcore for stiffness matrix storage option
target for target strain rate
teo for thermal effects option
xvel for load due to x-angular velocity
yvel for load due to y-angular velocity
zvel for load due to z-angular velocity

tz3dopts TOPAZ3D analysis options

tz3dopts *options*

iunit *option*
where the *option* can be
dimensionless
centigrade
fahrenheit
kelvin
rankine
bwmo *option*
where the *option* can be
0 for no minimization (default)
1 for minimization

2 for minimization, nodal destination vector is read from input file

flux
phase
solver *flag* where the *flag* can be
0 for the fissle solver
1 for the actcol solver
radiation *type*
where *type* can be
view for view factors
exchange for exchange factors
steady
transien
fixed
variable
timin *time*
timend *time*
delt *step_size*
dtmin *time_step*
dtmax *time_step*
dcmx *temperature*
mfts *factor*
iprt *#_steps*
iplt *#_steps*
sbrf *#_steps*
tipar *alpha*
linear
nonlinear
nbsr *#_steps*
nbei *#_steps*
msrf *#_reformations*
nibsr *#_iterations*
dctol *tolerance*
tzrelax *parameter*
sbc *constant*
rctol *tolerance*
mrdi *iterations*
rband *list_breakpoints* ;
emissivity *curve_# list_emissivities* ;

Remarks

iunit for temperature units with the five exclusive options

dimensionless
centigrade
fahrenheit
kelvin
rankine

bwmo for bandwidth and profile minimization selection with the three exclusive options
no minimization (default)
minimization
minimization, nodal destination vector is read from input file

flux for node heat flux calculation (default is not to calculate)

phase to perform phase change calculations

solver to select the analysis solver

radiation to select the radiation calculation type with the two exclusive options
view for view factors
exchange for exchange factors

select **type** of analysis from the two exclusive options
steady for steady state or
transient with the options
fixed for fixed time step with the options
timin for the initial problem time
timend for the final problem time
delt for the time step size
variable for variable time step with the options
timin for the initial problem time
timend for the final problem time
delt for the initial time step size
dtmin for the minimum time step
dtmax for the maximum time step
dcmx for the desired maximum temperature change in each time **step** above which the time step will be decreased
mfts for the modification factor for increasing/decreasing time step
iprt for the number of time steps between printed data output
iplt for the number of steps between plotted data output
sbrf for the number of time steps between dump file generation
tipar for the time integration parameter

select **type** of problem with the two exclusive options
linear for a linear problem or
nonlinear for a nonlinear problem with the options
nbsr for the number of time steps between conductance matrix reformations
nbei for the number of time steps between equilibrium iterations
msrf for the maximum number of conductance matrix reformations per time step

nibsr for the maximum number of equilibrium iterations permitted per conductance matrix reformation
dctol for the convergence tolerance for equilibrium iterations
tzrelax for the relaxation parameter

sbc for the stefan-boltzman constant
rctol for the radiosity convergence tolerance
mrdi for the maximum number of radiosity iterations
rband to specify a set of wavelength breakpoints
emissivity to specify multiple numbered 2d emissivity curves with the same number of points as the number of wavelength breakpoints

III. Material Definitions

True*Grid*® generates input files for any of a large number of simulation codes. Each code has its own set of material models and its own way to specify them. You can specify your material models in **True*Grid*®** using a **True*Grid*®** dialogue box. For detailed information about material specifications, refer to the manuals of the appropriate simulation code. It is not practical to reproduce those manuals here.

For the sake of completeness, the currently available material definition commands are listed below. To use these commands, use **True*Grid*®**'s graphical user interface.

XYZ Scientific Applications is actively adding support of more simulation codes. By the time you get **True*Grid*®**, there may be more available than you will find in this section. We welcome your requests for even more simulation codes!

abaqmats ABAQUS materials

abaqmats *material_number options* ;

where an *option* can be any of the following:
aqdens *density* mass density
aqdepv *#_variables* number of dependent variables
aqtherm *opt1* thermal expansion coefficients
where *opt1* can be any of the following:
aqexze *temp_0* initial temperature
aqnpm *opt2* non-porous material
where *opt2* must be one of the following:
aqexis {*talpha tmpopt fldop*}; isotropic
aqexor (*alpha_11 ... alpha_33 tmpopt fldopt*); - orthotropic
aqexan (*alpha_11 ... alpha_23 tmpopt fldopt*); - anisotropic
aqmstr *opt2* - material structural
where *opt2* must be one of the following:
aqexis (*alpha tmpopt stropt*;); - isotropic
aqexor (*alpha_11 ... alpha_33 stropt tmpopt*); - orthotropic
aqexan (*alpha_11 ... alpha_23 stropt tmpopt*); - anisotropic
aqporf (*alpha tmpopt fldopt*); - pore fluid
aqcond *opt1* - conductivity
where *opt1* must be one of the following:
aqcdis (*conductivity tmpopt fldopt);* - isotropic

aqcdor (*k_11 k_22 k_33 tmpopt fldopt*); - orthotropic
aqcdan (*k_11 ... k_33 tmpopt fldopt);* - anisotropic
aqlath (*heat temp temp*); - latent heat
aqspec (*specific_heat tmpopt fldopt*); - specific heat
aqperm *opt1* - permeability
where *opt1* can be any of the following:
aqsww *weight* - specific weight of water
aqpris *k* - isotropic
aqpror *k11 k22 k33* - orthotropic
aqpran *k11 ... k33* - anisotropic
aqvoid - voids ratio
aqporo (*temp opt1);* - porous bulk moduli
where *opt1* can be any of the following:
aqslgr *modulus* - solid grain bulk modulus
aqprfl *modulus* - permeating fluid bulk modulus
aqacm *modulus* (drag frequency); - acoustic medium
aqdepl (*modulus ratio yield exp offset temp*); - deformation plasticity
aqusmt *opt1 list_params* ; - user material
where *opt1* must be one of the following:
aqumsy - symmetric
aqumusy - unsymmetric
aqelas *opt1* ;
where *opt1* must be one of the following:
aqelis (*modulus ratio tmpopt fldopt*); - isotropic
aqelec (*e1 e2 e3 v12 v13 v23 g12 g13 g23 tmpopt fldopt*); - engineering
aqella (*e1 e2 v12 g12 g13 g23 tmpopt fldopt*); - lamina
aqelor (*d1111 ... d2323 tmpopt fldopt*); - orthotropic
aqelan (*d1111 ... d2323 tmpopt fldopt*); - anisotropic
aqhyper *opt1* - hyperelastic
where *opt1* must be one of the following:
aqseps - strain energy by user subroutine
aqsepp 1 (*c10 c01 d1 r tmpopt fldopt*); - order 1
aqsepp 2 (*c10 ... c02 d1 d2 r tmpopt fldopt*); - order 2
aqsepp 3 (*c10 ... c03 d1 d2 d3 r tmpopt fldopt*); - order 3
aqsepp 4 (*c10 ... c04 d1 d2 d3 d4 r tmpopt fldopt*); - order 4
aqhypo (*modulus ratio i1 i2 i3*); - hypoelastic
aqpore *opt1* - porous elastic
where *opt1* must be one of the following:
aqctsm (*bulk shear limit tmpopt fldopt*); - constant shear
aqcpsm (*bulk ratio limit tmpopt fldopt*); - poisson's
aqheat - heat generation
aqmpmf *factor* - mass proportional damping factor

aqspdf *factor* - stiffness proportional damping factor
aqfcdf *factor* - fraction for composite damping factor
aqnocs - allow no compression stress
aqnots - allow no tension stress
aqconc (stress strain); - concrete
aqtens (stress strain); - tension stiffening
aqfara *opt1* ; - failure ratios
where *opt1* can be any of the following:
aqfr1 *ratio* - ultimate biaxial/uniaxial compression stress
aqfr2 *ratio* - uniaxial tension/compression stress at failure
aqfr3 *ratio* - plastic strain at ultimate stress biaxial/uniaxial
aqfr4 *ratio* - principle stress/uniaxial stress at cracking
aqsret *opt1* ; - shear retention
where *opt1* can be any of the following:
aqshrt1 *rho* - dry concrete
aqshrt2 *epsilon* - dry concrete
aqshrt3 *rho* - wet concrete
aqshrt4 *epsilon* - wet concrete
aqdppm (*list_opt1 tmpopt fldopt* ;); - drucker prager plasticity
where *opt1* can be any of the following:
aqdpm1 *angle* - material angle of friction
aqdpm2 *k* - ratio of flow stress in triaxial tension/compression
aqdpm3 *angle* - dilation angle
aqplas *opt1* (*stress strain tmpopt fldopt*); - plastic
where *opt1* can be one of the following:
aqishrd - isotropic hardening
aqkihard - kinematic hardening
aqayld *sigma11 sigma22 sigma33 tau12 tau13 tau23* - potential
aqrdvp (*d p tmpopt fldopt*); - rate dependent
aqcree *opt1* - creep
where *opt1* must be one of the following:
aquscre - user subroutine
aqthcre (*a n m temp*); - time hardening
aqshcre (*a n m temp*); - strain hardening
aqhscre (*a b n dh r*); - hyperbolic sine
aqornl *opt1* - ornl
where *opt1* can be any of the following:
aqaornl *rate* - saturation rate for kinematic shift
aqhornl *rate* - rate of kinematic shift w.r.t. creep strain
aqmornl - stainless steel hardening
aqrornl - invoke optional alpha reset procedure
aqswel *opt1* - swelling

where *opt1* must be one of the following:
aqusswe - user subroutine
aqdtswe (*strain_rate tmpopt fldopt*); - data specified
aqanswel *r11 r22 r33* - ratios
aqclay - clay plasticity
aqint *modulus stress_ratio beta k* - specify intercept
aqnoint *modulus stress_ratio surf_size beta k* - no intercept
aqwater - water penetration
aqyield (*stress strain tmpopt fldopt*); - yield
aqfoam (*stress strength stress modulus k tmpopt fldopt*); - foam
aqcycl *opt1* (*stress tmpopt fldopt);* - cycled plastic
where *opt1* can be one of the following:
aqcycl1 - every 10th cycle
aqcycl2 - every 100th cycle
aqinelst *heat_flux* - inelastic heat fraction
aqvisco *opt1* - viscoelastic
where *opt1* can be one of the following:
aqvisc1 *real_g1 imag_g1 a real_k1 imag_k1 b* - formula
aqvisc2 (*real_wg imag_wg real_wk imag_wk frequency*); - tabular
aqvisc3 (*opt2 time*); - prony
where *opt1* can be any of the following:
aqprny1 *ratio* - shear relaxation modulus ratio
aqprny2 *ratio* - bulk relaxation modulus ratio

Remarks

To understand the complex syntax for this command, keep the following rules in mind. use the dialogue box to make this easy. Upper case words are literals. Lower case words require a substitution. The ; are required to end lists of arbitrary length. Parenthesis denote argument sets that can be repeated.

tmpopt means optionally "aqotmp temperature".
fldopt means optionally "aqofv list_field_values ;".
stropt means optionally "aqeps effective_stress".
short comments are added after first dash in a line.

ansymats ANSYS materials

ansymats *mat_no stifn material_option list_of_values* ; *kopnm* ;

where

mat_no user-defined material number to be associated with the material and element (using **mate** or **ma**t commands to set a material type within **True*Grid***® will automatically invoke the element type of this command along with its material properties).

material_option can include some, all or none of the following (depending on element type - see the ansys user's manual or use a dialogue box).

- **ex** *elastic young's modulus* (x-direction)
- **ey** *elastic young's modulus* (y-direction)
- **ez** *elastic young's modulus* (z-direction)
- **alpx** *coefficient of thermal expansion* (x-direction)
- **alpy** *coefficient of thermal expansion* (y-direction)
- **alpz** *coefficient of thermal expansion* (z-direction)
- **nuxy** *poisson's ratio*
- **nuyz** *poisson's ratio*
- **nuxz** *poisson's ratio*
- **gxy** *shear modulus* (x-stress, y-strain)
- **gyz** *shear modulus* (y-stress, z-strain)
- **gxz** *shear modulus* (x-stress, z-strain)
- **dens** *density*
- **mu** *coefficient of friction*
- **damp** *damping coefficient*
- **kxx** *thermal conductivity* (x-direction)
- **kyy** *thermal conductivity* (y-direction)
- **kzz** *thermal conductivity* (z-direction)
- **ch** *specific heat*
- **hf** *convection film coefficient*
- **rsvx** *electrical resistivity* (x-direction)
- **rsvy** *electrical resistivity* (y-direction)
- **rsvz** *electrical resistivity* (z-direction)
- **visc** *viscosity*

list_of_values is a list of as many as 5 numbers associated with the material_option. Enter just one value for a non temperature-dependent property. The list must be terminated by a semi-colon. The *material_option* and *list_of_values* may be repeated as many times as desired.

kopnm is the same as the "keyopt(n)=m" ansys options for the element types (see the ansys user's manual); the permissible values of n and m depend on the element type. This option may be repeated as many times as desired. Remember to terminate the command by a semi-colon. The possible options are hard-wired into the dialogue box for ansymats so that you will be unable to choose invalid options for a given element type.

Remarks

Stifn is the element type (ANSYS type); only the following elements are supported.

stif5 3-d multi-field solid
stif7 joint element
stif8 3-d spar
stif10 tension-only or compression-only spar
stif14 spring-damper
stif16 elastic straight pipe
stif17 elastic pipe tee
stif18 elastic curved pipe (elbow)
stif24 3-d thin-walled plastic beam
stif30 3-d isoparametric acoustic fluid
stif33 3-d heat conducting bar
stif34 convection link
stif38 dynamic fluid coupling
stif39 nonlinear force-deflection element
stif40 combination element
stif41 3-d membrane shell
stif43 plastic quadrilateral shell
stif44 3-d tapered unsymmetrical beam
stif45 3-d isoparametric solid
stif46 8-node layered solid
stif52 3-d interface
stif57 isoparametric quadrilateral thermal shell
stif63 elastic quadrilateral shell
stif64 3-d anisotropic solid
stif65 reinforced concrete solid
stif66 transient thermal-flow pipe
stif69 3-d thermal-electrical solid
stif70 isoparametric thermal solid
stif80 3-d fluid element
stif85 crack-tip solid
stif86 3-d hyperelastic solid
stif91 8-node layered shell

ansynl ANSYS material non-linear properties

ansynl *property option_list* ;

where *property* can be

mp for magnetic and/or piezoelectric properties
material_# (references material set by ansymats)
any or all of the "nl" options can follow:

nl56
- **6** to specify 6 point b-h curve at 6 temperatures
  magnetic_field_intensity_list (5 values) *h2 .... h6*
  magnetic_flux_density_list (5 values) *b2 ... b6*
  magnetic_flux_density_list (5 values) *b2 ... b6*
  magnetic_flux_density_list (5 values) *b2 ... b6*
  magnetic_flux_density_list (5 values) *b2 ... b6*
  magnetic_flux_density_list (5 values) *b2 ... b6*
  magnetic_flux_density_list (5 values) *b2 ... b6*
- **21** to specify 21 point b-h curve at 1 temperature
  magnetic_field_intensity_list (20 values) *h2 ... h21*
  magnetic_flux_density_list (20 values) *b2* ... b21
- **14** to specify 14 point b-h curve at 2 temperatures
  magnetic_field_intensity_list (13 values) *h2 ... h14*
  magnetic_flux_density_list (13 values) *b2 ... b14*
  magnetic_flux_density_list (13 values) *b2 ... b14*

nl97 *kxx1 .... kxx6* values of thermal conductivity
nl151 *kyy1 .... kyy6* values of thermal conductivity
nl205 *kzz1 .... kzz6* values of thermal conductivity
nl103 *vxx1* ... vxx6 values of electrical conductivity
nl157 *vyy1* ... vyy6 values of electrical conductivity
nl211 *vzz1* ... vzz6 values of electrical conductivity
nl262 *e11 ... e33* imaginary dielectric matrix diagonal elements
nl109 *mgxx1 ... mgxx6* magnetic coercive forces
nl163 *mgyy1 ... mgyy6* magnetic coercive forces
nl217 *mgzz1 ... mgzz6* magnetic coercive forces
nl115 *tgxx1 ... tgxx6* thermal gradient
nl169 *tgyy1 ... tgyy6* thermal gradient
nl223 *tgzz1 ... tgzz6* thermal gradient
nl121 *vgxx1 ... vgxx6* electric gradient
nl175 *vgyy1 ... vgyy6* electric gradient
nl229 *vgzz1 ... vgzz6* electric gradient
nl133 piezoelectric matrix
e11 e12 e13
e21 e22 e23
e31 e32 e33
e41 e42 e43
e51 e52 e53

e61 e62 e63

nl259 ratios *mu* in x,y,z directions

nl271 enter upper half of symmetric stiffness matrix

c11 c12 c13 c14 c15 c16
c22 c23 c24 c25 c26
c33 c34 c35 c36
c44 c45 c46
c55 c56
c66

src or current sources

nl55 *value_of_permeability_of_free_space*

nl56 *number_of_times_to_repeat_element_sources*

nl57

mks for mks system of units

cgs for cgs system of units

nl58 *number_of_modified_newton_rhapson_iterations*

nl61

source_id (between 1 and 26, inclusive)

nl1

coil or coil source

bar for bar source

arc for arc source

nl2 *total_current*

nl3 *characteristic_length*

nl4 *cross_section_dimension_dx dz* (or *dy*)

nl6 coded_#_of_integration_points

nl7 *xc yc zc* origin of global cartesian source coordinates

thxy thyz thxz angles

nl131 *arc_angle* only for arc source

pla for plasticity followed by:

bikh *temperatures* ; *yield_stresses* ; *slopes* ;
for classical bilinear kinematic hardening

anis *tx ty tz tmx tmy tmz cx cy cz tmx tmy tmz sxy syz sxz tmx tmy tmz*
for anisotropic plasticity behaviour
where
tx ty tz tensile yield stresses
tmx tmy tmz tangent moduli
cx cy cz compressive yield stresses
tmx tmy tmz tangent moduli
sxy syz sxz shear yield stresses
tmx tmy tmz tangent moduli

drpr *cohesion angle dilatancy* for Drucker-Prager behaviour

where
cohesion is the cohesion value
angle is the angle of internal friction
dilatancy is the amount of dilatancy

mkhp for multilinear kinematic hardening
list_of_strains_for_stress_strain (up to 5) ;
nl19 (to set first stress-strain curve)
temperature
list_of_stresses (corresponding to strains) ;
nl25 nl31 nl37 nl43 (to set other stress-strain curves)
temperature
list_of_stresses

mihp for multilinear isotropic hardening
list_of_strains_for_stress_strain (up to 5) ;
nl19 (to set first stress-strain curve)
temperature
list_of_stresses (corresponding to strains) ;
nl25 nl31 nl37 nl43 (to set other stress-strain curves)
temperature
list_of_stresses

nlel for nonlinear eleastic behaviour
list_of_strains_for_stress_strain (up to 5) ;
nl19 (to set first stress-strain curve)
temperature
list_of_stresses (corresponding to strains) ;
nl25 nl31 nl37 nl43 (to set other stress-strain curves)
temperature
list_of_stresses

crp *creep*
option_list depends on the particular property, but generally follows the format nln *number_list* where n is the position in the ansys nonlinear table (see the **nl** command in the ansys user's manual) and number_list is a collection of numbers to be placed in the nonlinear table beginning at position n. Often the user must specify 6 values, corresponding to values at 6 different temperatures.

Remarks

Crp is followed by 0 1 2 9 10 11 13 14 15 depending on the chosen form of the creep equation values of constants c1 c2 c3 c4 c5 ...; Which constants must be specified depends on the form of the creep equation.

The exact form of the creep equations can be found in the dialogue box for ansynl.

Follow the last of the chosen nl61 options by a semi-colon then nl61 and its options can be repeated (end each section of such options with a semi-colon).

dynaeos DYNA3D equation of state

dynaeos *eos_type material_# parameter_list* ;

where the EOS type can be 1 through 9 or 11

1 for linear polynomial model type 1
with the following options
c0 *constant*
c1 *coefficient*
c2 *coefficien*t
c3 *coefficient*
c4 *coefficient*
c5 *coefficient*
c6 *coefficient*
e0 *energy*
v0 *volume*

2 for JWL model type 2
with the following options
a *constant*
b *constant*
r1 *constant*
r2 *constant*
omega *constant*
e0 *energy*
v0 *volume*

3 for the Sack model
with the following options
a1 constant
a2 constant
a3 constant
b1 constant
b2 constant
b0 constant
v0 constant

4 for the Gruneisen model

with the following options
vci intercept
s1 coefficient
s2 coefficient
s3 coefficient
gamma coefficient
sa coefficient
b0 energy
v0 volume

5 for the ratio of polynomials model
with the following options
a10 constant
a11 constant
a12 constant
a13 constant
a20 constant
a21 constant
a22 constant
a23 constant
a30 constant
a31 constant
a32 constant
a33 constant
a40 constant
a41 constant
a42 constant
a43 constant
a50 constant
a51 constant
a52 constant
a53 constant
a60 constant
a61 constant
a62 constant
a63 constant
a70 constant
a71 *constant*
a72 *constant*
a73 *constant*
alpha *constant*
beta *constant*

a14 *constant*
a24 *constant*
e0 *energy*
v0 *volume*

6 for the linear polynomial with energy deposition model with the following options
c0 *constant*
c1 *coefficient*
c2 *coefficient*
c3 *coefficient*
c4 *coefficient*
c5 *coefficient*
c6 *coefficient*
e0 *energy*
v0 *volume*
lc *load_curve*

7 for the ignition and growth of reaction in he model with the following options
ap *constant*
bp *constant*
r1p *constant*
r2p *constant*
g *coefficient*
wpcp *constant*
ae *constant*
be *constant*
wece *constant*
r1e *constant*
r2e *constant*
fcrit *fraction*
i *coefficient*
h *coefficient*
z *exponent*
x *exponent*
y *exponent*
cp *heat_capacity*
ce *heat_capacity*
m *exponent*
e0 *energy*
t0 *temperature*

8 for the tabulated model with compaction with the following options

eps *list_strains* ;
pc *list_constants* ;
t *list_temperatures* ;
ku *list_modulus* ;*compression*
gamma *gamma*
e0 *energy*
v0 *volume*

9 for the tabulated model
with the following options
eps *list_strains* ;
pc *list_constants* ;
t *list_temperatures* ;
gamma *gamma*
e0 *energy*
v0 *volume*

11 for the pore collapse model
with the following options
mu1 *compression*
mu2 *compresion*
e0 *energy*
mu0 *compression*
virgin *load_curve_pairs ;*
crushed *load_curve_pairs ;*

dynamats DYNA3D materials

dynamats *material_# material_type options_lis*t ;

where material type can be 0 through 35 except 29

The following options are available for all materials:

shell
with the following options
elfor *option*
where the option can be

hl	for Hughes-Lui shell
bt	for Belytschko-Tsay shell
bciz	for triangular shells
c0	for triangular shells

membrane
yase

shear *factor*

tsti #_points *propt option*

where option can be

1 for element center

2 for plan integration points

3 for through thickness and plan integration points

quad *integration_rule_#*

where the number can be

positive for the number of points using the trapezoidal rule

0 for gauss

-n negative of the user specified rule number (from sind command)

shth *thickness*

shth1 *thickness*

shth2 *thickness*

shth3 *thickness*

shth4 *thickness*

shloc *location*

where location can be

1 for top surface

0 for middle surface

-1 for bottom surface

beam

with the following options

elfom *option*

where the option can be

hl for Hughes-Lui beams

bt for Belytschko-Tsay beams

truss

shear *factor*

quad *option*

where the option can be

1 for a truss

2 for 2x2 Gauss quadrature

3 for 3x3 Gauss quadrature

4 for 3x3 Lobatto integration

5 for 4x4 Gauss quadrature

bmcross *shape*

where the shape can be

0 for rectangular

1 for tubular

sthi *thickness*
tthi *thickness*
sthi1 *thickness*
sthi2 *thickness*
tthi1 *thickness*
tthi2 *thickness*
sloc *location* where location can be
1 meaning the side where s is 1
0 meaning centered
-1 meaning the side where s is -1
tloc *location*
where location can be
1 meaning the side where t is 1
0 meaning centered
-1 meaning the side where t is -1

tshell
with the following options
shear *shear*
tsti *#_points*
quad *integration_rule_#*

rho *density*

The remainder is specific to the selected material type:

Material type 1
e *modulus*
pr *ratio*

Material type 2
ea *ea*
eb *eb*
ec *ec*
prba *vba*
prca *vca*
prcb *vcb*
gab *gab*
gbc *gbc*
gca *gca*
aopt *option parameters*
where the option can be one of
0 for by nodes
1 for by point and element center

2 for by normal vectors
3 for by cross product with shell normal (shell elements only)

xp *x-coordinate* for aopt 1
yp *y-coordinate* for aopt 1
zp *z-coordinate* for aopt 1
ax *x-component* for aopt 2
ay *y-component* for aopt 2
az *z-component* for aopt 2
dx *x-component* for aopt 2
dy *y-component* for aopt 2
dz *z-component* for aopt 2
vx *x-component* for aopt 3
vy *y-component* for aopt 3
vz *z-component* for aopt 3
beta *angle*

Material type 3

e *modulus*
pr *ratio*
sigy *stress*
etan *modulus*
beta *parameter*

Material type 4

temp *temperature_list* ;
e *modulus_list* ;
pr *ratio_list* ;
alpha *secant_list* ;
sigy *stress_list* ;
etan *modulus_list* ;

Material type 5

g *modulus*
ku *modulus*
a0 *yield*
a1 *yield*
a2 *yield*
pc *pressure*
vs *strain_list* ;
p *pressure_list* ;

Material type 6

k *modulus*
g0 *modulus*
gi *modulus*
beta *constant*
mflag *option*
where the option can be
0 which means that beta is the delay constant
1 which means that beta is the time relaxation constant

Material type 7
g *modulus*

Material type 8
d *velocity*
pcj *pressure*

Material type 9
pc *pressure*
mu *coefficient*

Material type 10
g *modulus*
sigy *stress*
etan *modulus*
pc *pressure*
a1 *coefficient*
a2 *coefficient*
ispall *model*
where the model can be
pl for pressure limit
max for maximum principal stress spall criterion
hydro for hydrostatic tension spall criterion
eps *strain_list* ; up to 16 values
es *stress_list* ; up to 16 values

Material type 11
g0 *modulus*
sig0 *stress*
beta *constant*
n *exponent*
gama *strain*
sigm *stress*

b *modulus*
bpm *stress*
h *coefficient*
f *exponent*
t0 *temperature*
gam0 *gamma*
sa *constant*
pc *pressure*
ispall *model*
where the model can be
pl for pressure limit
max for maximum principal stress spall criterion
hydro for hydrostatic tension spall criterion
a *atomic_weight*
r *r_prime*
spall
ivar *option*
where the option can be
0 for cold compression polynomial coefficient in eta
1 for cold compression polynomial coefficient in mu
ec0 *coefficient*
ec1 *coefficient*
ec2 *coefficient*
ec3 *coefficient*
ec4 *coefficient*
ec5 *coefficient*
ec6 *coefficient*
ec7 *coefficient*
ec8 *coefficient*
ec9 *coefficient*

Material type 12
g *modulus*
sigy *stress*
eh *modulus*
k *modulus*

Material type 13
g *modulus*
sigy *stress*
eh *modulus*
fs *strain*

fp *pressure*
k *modulus*

Material type 14
g *modulus*
ku *modulus*
a0 *constant*
a1 *constant*
a2 *constant*
pf *pressure*
iflag *flag*
where the flag can be
0 for hydrostatic tension
1 for maximum principal stress
sigmaf *stress*
vs *strain_list* ;
ps *pressure_lis* ;

Material type 15
g *modulus*
a *stress*
b *coefficient*
n *exponent*
sc *coefficient*
m *exponent*
tm *temperature*
tr *temperature*
x0 *rate*
sh *heat*
ispall *model*
where the model can be
pl for pressure limit
max for maximum principal stress spall criterion
hydro for hydrostatic tension spall criterion
iter *flag*
where the flag can be
0 for fast approximate solution
1 for accurate iterative solution
d1 *parameter*
d2 *parameter*
d3 *parameter*
d4 *parameter*

d5 *parameter*
e *modulus*
pr *ratio*
dtcrit *step_size*

Material type 16
pr *ratio*
g *modulus*
sigy *stress*
a0 *cohesion*
a1 *coefficient*
a2 *coefficient*
b1 *factor*
a0f *cohesion*
a1f *coefficient*
r *percent*
emr *modulus*
prr *ratio*
sigma0 *stress*
tm *modulus*
lc *load_curve*
lcr *load_curve*
eps *list_strain* ;
es *list_stress* ;
p *list_pressure* ;

Material type 17
e *modulus*
pr *ratio*
sigy *stress*
eh *modulus*
fs *strength*
pc *pressure*

Material type 18
e *modulus*
pr *ratio*
k *coefficient*
n *exponent*

Material type 19
e *modulus*

pr *ratio*
lcs0 *load_curve*
etan *modulus*
lce *load_curve*
lce *load_curve*
lcfs *load_curve*
tss *step_size*

Material type 20
e *modulus*
pr *ratio*
bpm *options* ;
where an *option* can be
dof flag
where *flag* can be
1 x-translational degree-of-freedom
2 y-translational degree-of-freedom
3 z-translational degree-of-freedom
4 translational motion in the given vector direction (use v below)
5 x-rotational degree-of-freedom
6 y-rotational degree-of-freedom
7 z-rotational degree-of-freedom
8 rotational motion about the given vector (use v below)
lcid *load_curve_#*
sf *scale_factor*
v x_0 y_0 z_0
rbv *load_curve amplitude fx fy fz* (obsolete)

Material type 21
ea *modulus*
eb *modulus*
ec *modulus*
prba *ratio*
prca *ratio*
prcb *ratio*
alpa *coefficient*
alpb *coefficient*
alpc *coefficient*
gab *modulus*
gbc *modulus*
gca *modulus*
aopt option

where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	for by cross product with shell normal (shell elements only)

xp *x-coordinate* for aopt 1
yp *y-coordinate* for aopt 1
zp *z-coordinate* for aopt 1
ax *x-component* for aopt 2
ay *y-component* for aopt 2
az *z-component* for aopt 2
dx *x-component* for aopt 2
dy *y-component* for aopt 2
dz *z-component* for aopt 2
vx *x-component* for aopt 3
vy *y-component* for aopt 3
vz *z-component* for aopt 3
beta *angle*

Material type 22 (Fiber composite with damage material type 22)

ro *density*
ea *modulus*
eb *modulus*
ec *modulus*
k *modulus*
sn *strength*
syz *strength*
szx *strength*
prba *ratio*
prcb *ratio*
prca *ratio*
gab *modulus*
gbc *modulus*
gca *modulus*
aopt *option*

where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	for by cross product with shell normal (shell elements only)

xp *x-coordinate* for aopt 1
yp *y-coordinate* for aopt 1

zp *z-coordinate* for aopt 1
ax *x-component* for aopt 2
ay *y-component* for aopt 2
az *z-component* for aopt 2
dx *x-component* for aopt 2
dy *y-component* for aopt 2
dz *z-component* for aopt 2
vx *x-component* for aopt 3
vy *y-component* for aopt 3
vz *z-component* for aopt 3
axes *flag*
where the flag can be
1 for the default
2 for switch material axes a and b
3 for switch material axes a and c
sc *strength*
xt *strength*
yt *strength*
yc *strength*
alpha *parameter*
beta *list_angles* ;

Material type 23
ea *ea_list* ;
eb *eb_list* ;
ec *ec_list* ;
vba *vba_list* ;
vca *vca_list* ;
vcb *vcb_list* ;
aa *aa_list* ;
ab *ab_list* ;
ac *ac_list* ;
gab *gab_list* ;
gbc *gbc_list* ;
gca *gca_list* ;
t *temperature_list* ;
angles *angle_list* ;
aopt *option parameters*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors

3 for by cross product with shell normal (shell elements only)
xp *x-coordinate* for aopt 1
yp *y-coordinate* for aopt 1
zp *z-coordinate* for aopt 1
ax *x-component* for aopt 2
ay *y-component* for aopt 2
az *z-component* for aopt 2
dx *x-component* for aopt 2
dy *y-component* for aopt 2
dz *z-component* for aopt 2
vx *x-component* for aopt 3
vy *y-component* for aopt 3
vz *z-component* for aopt 3

Material type 24
e *modulus*
pr *ratio*
sigy *stress*
et *modulus*
efp *strain*
dtcrit *time*
lc *load_curve*
eps *strain*
es *stress*

Material type 25
k *modulus*
g *modulus*
alpha *parameters*
theta *coefficient*
gamma *coefficient*
beta *exponent*
r *ratio*
d *exponent*
w *coefficient*
x0 *parameter*
cbar *coefficient*
n *parameter*
nplot *option*
where the options are
1 hardening variable, k
2 cap - j1 axis intercept, x(k)

3 volumetric plastic strain
4 first stress invariant, j1
5 second stress invariant, square root of j2
8 response mode number
9 number of iterations

ltype *option*

where the options are

1 soil or concrete (cap surface may contract)
2 rock (cap surface does not contract)

ivec *option*

where the options are

0 vectorization (fixed number of iterations)
1 fully iterative

t *cutoff*

Material type 26

e *modulus*
pr *ratio*
sigy *stress*
sigaa *load_curve*
sigbb *load_curve*
sigcc *load_curve*
ssrv *load_curve*
crv *volume*
ea *modulus*
eb *modulus*
ec *modulus*
gab *modulus*
gbc *modulus*
gca *modulus*

Material type 27

a *coefficient*
b *coefficient*
pr *ratio*

Material type 28

e *modulus*
pr *ratio*
sigy *stress*
etan *modulus*

Material type 30

e *modulus*
pr *ratio*
sigy *stress*
etan *modulus*

Material type 31

g001 *coefficient*
g010 *coefficient*
g020 *coefficient*
g100 *coefficient*
g101 *coefficient*
g110 *coefficient*
g200 *coefficient*
g210 *coefficient*
g300 *coefficient*
g400 *coefficient*
ilimit *option*

where the option can be

0	to stop if strain limits are exceeded
1	to continue if strain limits are exceeded

stmx *strain*
stmn *strain*

Material type 32

gammay *strain*
tauy *stress*
alpha *coefficient*
r *exponent*
k *modulus*

Material type 33

ea *modulus*
eb *modulus*
ec *modulus*
r *coefficient*
acp *coefficient*
qbc *coefficient*
qab *coefficient*
qac *coefficient*
prba *ratio*
prca *ratio*

prcb *ratio*
aopt *axes*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 for by cross product with shell normal (shell elements only)
xp *x-coordinate* for aopt 1
yp *y-coordinate* for aopt 1
zp *z-coordinate* for aopt 1
ax *x-component* for aopt 2
ay *y-component* for aopt 2
az *z-component* for aopt 2
dx *x-component* for aopt 2
dy *y-component* for aopt 2
dz *z-component* for aopt 2
vx *x-component* for aopt 3
vy *y-component* for aopt 3
vz *z-component* for aopt 3
sigya *stress*
beta *angle*
eap *modulus*
gbc *modulus*
gab *modulus*
gac *modulus*
npss *substeps*
epsap *list_strain* ;
sigmaya *list_stress* ;

Material type 34
e *modulus*
pr *ratio*
sigy *stress*
etan *modulus*
r *parameter*

Material type 35 (under development)
e *modulus*
ifld *option*
where the option can be
1 for total
2 for incremental

3 for damage

pr *ratio*
lclh *load_curve*
lcrh *load_curve*
lcpx *load_curve*
lcedf *load_curve*
lcedm *load_curve*
sigy *stress*
scldev *factor*
etan *modulus*
beta *parameter*
ep *list_strain* ;
sigmay *list_stress* ;

nastmats NASTRAN materials

nastmats *material_#* ***material_type*** *options_list* ;

where material type can be 1 through 5, and 8 through 10

1 for isotropic elastic material type 1
e *modulus*
g *modulus*
nu *ratio*
rh0 *density*
a *expansion*
tref *temperature*
ge *damping*
st *stress*
sc *stress*
ss *stress*

2 for anisotropic material type 2
g *g11 g12 g13 g21 g22 g23 g31 g32 g33*
rh0 *density*
a1 *expansion*
a2 *expansion*
a3 *expansion*
to *temperature*
ge *damping*
st *stress*
sc *stress*

ss *stress*

3 for orthotropic material type 3
ex *modulus*
etheta *modulus*
ez *modulus*
nuxtheta *ratio*
nuthetaz *ratio*
nuzx *ratio*
rh0 *density*
gzx *modulus*
ax *expansion*
atheta *expansion*
az *expansion*
tref *temperature*
ge *damping*

4 for isotropic material type 4
k *conductivity*
cp *capacity*

5 for anisotropic material type 5
kxx *conductivity*
kxy *conductivity*
kxz *conductivity*
kxy *conductivity*
kyy *conductivity*
kyz *conductivity*
kxz *conductivity*
kyz *conductivity*
kzz *conductivity*
cp *capacity*

8 for orthotropic material type 8
e1 *modulus*
e2 *modulus*
v12 *ratio*
g12 *modulus*
g1z *modulus*
g2z *modulus*
rh0 *density*
a1 *expansion*
a2 *expansion*
tref *temperature*
xt *stress*
xc *stress*

yt *stress*
yc *stress*
s *stress*
ge *damping*
f12 *interaction*

9 for anisotropic material type 9
matrix *g11 g12 g13 g14 g15 g16 g22 g23 g24 g25 g26 g33 g34 g35 g36 g44 g45 g46 g55 g56 g66*
rh0 *density*
a *expansion*
tref *temperature*
ge *damping*

10 for fluid material type 10
bulk *modulus*
rho *density*
ss *speed*

where each material has an element type and parameters
psolid *options*
where the options are
in *flag*
stress 1 for gaussian
isop 1 for full
fctn for fluid elements
pshell *options*
where the options are
t *thickness*
mid2 *material_id*
t3 *stiffness*
mid3 *material_id*
tst *ratio*
nsm *ratio*
z1 *distance*
z2 *distance*
mid4 *material_id*
pplate *options*
where the options are
t *thickness*
mid2 *material_id*
t3 *stiffness*
mid3 *material_id*
tst *ratio*

nsm *ratio*
z1 *distance*
z2 *distance*
mid4 *material_id*
pbeam

lsdymats LS-DYNA materials

lsdymats *material_# material_type parameter_list* ;

where *material_type* can be 1 through 75.

Check the LS-DYNA manual or use the dialogue boxes in **True*Grid*®** to avoid mismatches in options. Please report (xyz@truegrid.com) any irregularities.

The following commands are available for all materials:

rho *density*
hgqt *type* Hourglass Stabilization Method
where type can be
1 standard LS-DYNA
2 Flanagan-Belytschko integration
3 Flanagan-Belytschko integration with exact volume
4 stiffness form of type 2 (Flanagan-Belytschko)
5 stiffness form of type 3 (Flanagan-Belytschko)
hgq *coefficient* hourglass coefficient
bqt *type* bulk viscosity type
bqq *coefficient* quadratic viscosity coefficient
bql *coefficient* linear viscosity coefficient
migl *option*
where the gravity loading option can be
0 for all initialized
1 for only current material is initialized
head comment (1st 200 materials only)

aet *type* for ambient element type
where the ambient element type can be
1 for temperature
2 for pressure & temperature
3 for pressure outflow

4 for pressure inflow

The following command is available for materials that do not allow failure and erosion:

mat_add_erosion with the following options:
excl *exclusion_number*
pfail *pressure_at_failure*
sigp1 *principal_stress_at_failure*
sigvm *equivalent_stress_at_failure*
epsp1 *principal_strain_at_failure*
epssh *shear_strain_at_failure*
sigth *threshold_stress*
impulse *stress_impulse_for_failure*

(Element Class:)
brick with the following options
elfob *option*
where the option can be
csb for constant stress brick
i8b for 8 point integration brick
i14b for 14 point integration quadratic 8-node brick
aleb for 1 point ALE brick
e1b for 1 point Eulerian brick
ea1b for 1 point Eulerian ambient brick
apb for acoustic pressure brick
afac for simple average smoothing factor
bfac for volume weighting smoothing factor
cfac for isoparametric smoothing factor
dfac for equipotential smoothing factor
sts for start time for smoothing
ets for end time for smoothing
aaf for ALE advection factor
shell
with the following options
elfor *option*
where the option can be
hl for Hughes-Lui shell
bt for Belytschko-Tsay shell
bciz triangular shells
c0 triangular shells
membrane for Belytschko-Tsay membrane

srhl for s/r Hughes-Lui
srcr for s/r co-rotational Hughes-Lui
ew for Englemann-Whirley
fibt for fully integrated Belytschko-Tsay membrane
bwc for Belytschko-Wong-Chaing
chl for co-rotational Hughes-Lui

shear *factor*
tsti *#_points*
propt *option*
where option can be
1 for average resultants & fiber lengths
2 for resultants at plan points & fiber lengths
3 for resultants, stresses at all points & fiber lengths

userrl *integration_rule_#* (from sind command)
quad *integration_rule_#*
where the number can be positive for the number of points using the trapezoidal rule
0 for Gauss
-n negative of the user specified rule number (from sind command)

shth *thickness*
shth1 *thickness*
shth2 *thickness*
shth3 *thickness*
shth4 *thickness*
shloc *location*
where location can be
1 for top surface
0 for middle surface
-1 for bottom surface

beam
with the following options
elfom *option*
where the option can be
hl for Hughes-Lui beams
bs for Belytschko-Schwer beams
truss for truss
bsi for Belytschko-Schwer full integration beam
bst for Belytschko-Schwer tubular beam
dis1 Discrete 3D Beam
spw Spot Weld Beam

shear *factor*
userrl *integration_rule_#* (**bind** required)

stcs standart cross sctions (**bsd** is required)
quad *option*
where the option can be
1 for a truss
2 for 2x2 Gauss quadrature
3 for 3x3 Gauss quadrature
4 for 3x3 Lobatto integration
5 for 4x4 Gauss quadrature
bmcross *shape*
where the shape can be
0 for rectangular
1 for tubular
sthi *thickness*
tthi *thickness*
sthi1 *thickness*
sthi2 *thickness*
tthi1 *thickness*
tthi2 *thickness*
sloc *location*
where location can be
1 meaning the side where s is 1
0 meaning centered
-1 meaning the side where s is -1
tloc *location*
where location can be
1 meaning the side where t is 1
0 meaning centered
-1 meaning the side where t is -1

tshell
with the following options
elfot *option*
where the option can be
spp for single point in plane quadrature thick shell
r22 for reduced 2x2 thick shell
shear *factor*
tsti *#_points*
propt *option*
where option can be
1 for average resultants & fiber lengths
2 for resultants at plan points & fiber lengths
3 for resultants, stresses at all points & fiber lengths
quad *integration_rule_#*

where the number can be positive for the number of points using the trapezoidal rule

0 for gauss

-n negative of the user specified rule number (from sind command)

The remainder is specific to the selected material:

For material type **1 struct** (Elastic) (Available Elements:)

e *young's_modulus*
da *axial_damping_factor*
db *bending_damping_factor*
pr *poisson's_ratio*

For material type **1 fluid** (Elastic Fluid)

k *bulk_modulus*
vc *tensor_viscosity_coefficient*
cp *cavitation_pressure*
pr *poisson's_ratio*

For material type **2 ortho** (Orthotropic Elastic)

ea *coefficient_ea*
eb *coefficient_eb*
ec *coefficient_ec*
prba *coefficient_vba*
prca *coefficient_vca*
prcb *coefficient_vcb*
gab *coefficient_gab*
gbc *coefficient_gbc*
gca *coefficient_gca*
aopt *option*

where the option can be one of

0 for by nodes
1 for by point and element center
2 for by normal vectors
3 for by cross product with shell normal (shell elements only)
4 for by normal vectors in cylindrical coordinates

xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2

dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3 & 4
vy *y-component* aopt 3 & 4
vz *z-component* aopt 3 & 4
p1 *x-component* aopt 4
p2 *y-component* aopt 4
p3 *z-component* aopt 4
beta *angle*

For material type **2 aniso** (Anisotropic Elastic)

c11 *constitutive_matrix_coefficient*
c12 *constitutive_matrix_coefficient*
c22 *constitutive_matrix_coefficient*
c13 *constitutive_matrix_coefficient*
c23 *constitutive_matrix_coefficient*
c33 *constitutive_matrix_coefficient*
c14 *constitutive_matrix_coefficient*
c24 *constitutive_matrix_coefficient*
c34 *constitutive_matrix_coefficient*
c44 *constitutive_matrix_coefficient*
c15 *constitutive_matrix_coefficient*
c25 *constitutive_matrix_coefficient*
c35 *constitutive_matrix_coefficient*
c45 *constitutive_matrix_coefficient*
c55 *constitutive_matrix_coefficient*
c16 *constitutive_matrix_coefficient*
c26 *constitutive_matrix_coefficient*
c36 *constitutive_matrix_coefficient*
c46 *constitutive_matrix_coefficient*
c56 *constitutive_matrix_coefficient*
c66 *constitutive_matrix_coefficient*
aopt *option*

where the option can be one of

0 for by nodes
1 for by point and element center
2 for by normal vectors
3 for by cross product with shell normal (shell elements only)
4 for by normal vectors in cylindrical coordinates

xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1

zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
vx *x-component*	aopt 3 & 4
vy *y-component*	aopt 3 & 4
vz *z-component*	aopt 3 & 4
p1 *x-component*	aopt 4
p2 *y-component*	aopt 4
p3 *z-component*	aopt 4

For material type **3** (Kinematic/Isotropic Elastic-Plastic)

e *young's_modulus*
pr *poisson's_ratio*
src *strain_rate_parameter*
srp *strain_rate_parameter*
sigy *yield_stress*
etan *tangent_modulus*
beta *hardening_parameter*
fs ***fai****lure_strain*

For material type **4** (Thermo-Elastic-Plastic**)**

temp *temperature_list* **;**
e *young's_modulus_list* **;**
pr *poisson's_ratio_list* ;
alpha *secant_coefficient_list* ;
sigy *yield_stress_list* ;
etan *plastic_hardening_modulus_list* ;

For material type **5** (Soil and Crushable Foam)

g *shear_modulus*
ku *bulk_unloading_modulus*
a0 *yield_function_constant*
a1 *yield_function_constant*
a2 *yield_function_constant*
pc *pressure_cutoff*
vcr *volumetric_crushing_option*
where the volumetric crushing option can be
0 for on

1 for loading and unloading paths are the same

vs *volumetric_strain_list* ;

ps *pressure_list* ;

For material type **6** (Linear Viscoelastic)

k *bulk_modulus*

g0 *short_time_shear_modulus*

gi *long_time_shear_modulus*

beta decay_constant

For material type **7** (Blatz-Ko Rubber)

g *shear_modulus*

For material type **8** (High Explosive Burn)

d *detonation_velocity*

pcj *chapman-jouget_pressure*

beta *option*

where the option can be

0 for beta + programmed burn

1 for beta burn only

2 for programmed burn only

k *bulk_modulus*

g *shear_modulus*

sigy *yield_stress*

For material type **9** (Null (Hydrodynamic w/o Deviatoric Stress))

pc *pressure_cutoff*

mu *viscosity_coefficient*

rvt *relative_volume*

rvc *relative_volume*

e *young's_modulus*

pr *poisson's_ratio*

For material type **10** (Isotropic Elastoplastic Hydrodynamic)

g *shear_modulus*

sigy *yield_stress*

etan *plastic_hardening_modulus*

pc *pressure_cutoff*

eps *effective_plastic_strain_list* ;

es *yield_stress_list* ;

For material type **11** (Steinberg-Guinan Thermal Elastoplastic Hydrodynamic)

g0 *shear_modulus_constant*
sig0 *yield_stress_constant*
beta *strain_hardening_law_constant*
n *strain_hardening_exponent*
gama *initial_plastic_strain*
sigm *yield_stress_work_hardening_limit*
b *shear_modulus_pressure_constant*
bp *yield_stress_pressure_constant*
h *energy_coefficient*
f *energy_exponent_coefficient*
t0 *melting_temperature_constant*
gam0 *thermodynamic_gamma*
sa *thermodynamic_constant*
ispall *spall_model_option*

where the spall model option can be

pl	for p >= p_min
smax	for sigma_max >= sigma_p element spalls and tension
hydro	for p < p_min element spalls and tension

pcut *pressure_cutoff*
a *atomic_weight*
r *r_prime*
ivar *cold_compression_option*

where the cold compression option can be

0	for cold compression polynomial coefficient in eta
1	for cold compression polynomial coefficient in mu

ec0 *polynomial_coefficient_ec0*
ec1 *polynomial_coefficient_ec1*
ec2 *polynomial_coefficient_ec2*
ec3 *polynomial_coefficient_ec3*
ec4 *polynomial_coefficient_ec4*
ec5 *polynomial_coefficient_ec5*
ec6 *polynomial_coefficient_ec6*
ec7 *polynomial_coefficient_ec7*
ec8 *polynomial_coefficient_ec8*
ec9 *polynomial_coefficient_ec9*

For material type **12** (Isotropic Elastoplastic)

g *shear_modulus*
sigy *yield_stress*
eh *tangent_modulus*
k *bulk_modulus*

For material type **13** (Isotropic Elastoplastic with Failure)

g *shear_modulus*
sigy *yield_stress*
eh *tangent_modulus*
fs *effective_plastic_strain_at_failure*
fp *failure_pressure*
k *bulk_modulus*
rem *element_removal_option*

where the element removal option can be

0	for failed element eroded after failure
1	for element kept, no removal except by delta-t

dt *delta-t*

For material type **14** (Soil and Crushable Foam with Failure)

g *shear_modulus*
ku *bulk_unloading_modulus*
a0 *yield_function_constant*
a1 *yield_function_constant*
a2 *yield_function_constant*
pc *pressure_cutoff*
vcr *volumetric_crushing_option*

where the volumetric crushing option can be

0	for on
1	for loading and unloading paths are the same

vs *volumetric_strain_list* ;
ps *pressure_list* ;

For material type **15** (Johnson/Cook Plasticity)

g *shear_modulus*
a *yield_stress*
b *strain_hardening_coefficient*
n *strain_hardening_exponent*
sc *strain_rate_dependent_coefficient*
m *temperature_dependence_exponent*
tm *melt_temperature*
tr *room_temperature*
x0 *effective_plastic_strain_rate*
sh *specific_heat*
ispall *spall_model*

where the spall model option can be

pl	for p >= p_min
smax	for sigma_max >= sigma_p element spalls and tension

hydro for p < p_min element spalls and tension

pcut *pressure_cutoff*

iter *plastic_strain_iteration*

where the plastic strain iteration can be

fast for fast approximate solution for plastic strain

accurate for accurate iterative solution for plastic strain

d1 *failure_parameter*

d2 *failure_parameter*

d3 *failure_parameter*

d4 *failure_parameter*

d5 *failure_parameter*

e *young's_modulus*

pr *poisson's_ratio*

dtcrit *time_step_size*

For material type **16** (Pseudo Tensor Concrete/Geological model)

g *shear_modulus*

pr *poissons_ratio*

sigy *tensile_cutoff*

a0 *cohesion*

a1 *1st_pressure_hardening*

a2 *2nd_pressure_hardening*

a0f *failed_mat_cohesion*

a1f *failed_mat_pressure_hardening*

b1 *damage_scaling_factor*

r *percent_reinforcement*

emr *reinforcement_elastic_modulus*

prr *reinforcement_poissons_ratio*

sigma0 *initial_yield_stress*

tm *tangent_modulus*

lc *principle_material_load* curve_#

lcr *reinforcement_load_curve_#*

opt *option*

where the option can be

1 for effective plastic strain curve w/ tensile cutoff

2 for effective plastic strain curve w/ max. principal stress failure

3 for pressure curve w/ tensile cutoff

eps *effective_plastic_strain_list* ;

es *yield_stress_list* ;

For material type **17** (Elastoplastic with Oriented Crack)

e *young's_modulus*

p *poisson's_ratio*
sigy *yield_stress*
eh *tangent_modulus*
fs *fracture_strength*
pc *pressure_cutoff*

For material type **18** (Power Law Isotropic Plasticity)
e *young's_modulus*
pr *poisson's_ratio*
k *yield_stress_coefficient*
n *strain_hardening_exponent*
src *strain_rate_parameter*
srp *strain_rate_parameter*

For material type **19** (Strain Rate Dependent Plasticity)
e *young's_modulus*
pr *poisson's_ratio*
lcs0 *yield_stress_curve_#*
etan *tangent_modulus*
lce *young's_modulus_curve_#*
lcet *tangent_modulus_curve_#*
lcfs *effective_stress_failure_curve_#*
tss *auto_deletion_time_step*

For material type **20** (Rigid)
e *young's_modulus*
nmad *option*
where the madymo3d coupling option can be
upno for normal ls-dyna3d rigid body updates
upel for couple to madymo ellipsoid
uppl for couple to madymo plane
coup *option*
where the madymo3d/cal3d coupling option can be
vdacoup for attach vda surface and generate mesh for ls-taurus.
meshcoup for undeformed geometry corresponds to local system
undcoup for undeformed geometry corresponds to global system
plncoup for generate a mesh for the ellipsoids and planes
seatoup for generate madymo seatbelts
conoup for generate a mesh for a contact entity
mopt *madymo/cal3d_system_#*
pr *poisson's_ratio*
alias *vda_file_name*

cmo center of mass constraint option
con constraints in the global coordinate system
spc constraints in the local coordinate system

lco *option*

where the option can be
id *local_sys_#* (see lsys)
df *x y z vx vy vz* local x axis and in-plane vector

bpm *options* ;

where an *option* can be
dof *flag*

where *flag* can be

1	x-translational degree-of-freedom
2	y-translational degree-of-freedom
3	z-translational degree-of-freedom
4	translational motion in the given vector direction (use **v** below)
5	x-rotational degree-of-freedom
6	y-rotational degree-of-freedom
7	z-rotational degree-of-freedom
8	rotational motion about the given vector (use **v** below)
9	degree-of-freedom rotation about x-axis (use **offset** below)
-9	degree-of-freedom radial about x-axis (use **offset** below)
10	degree-of-freedom rotation about y-axis (use **offset** below)
-10	degree-of-freedom radial about y-axis (use **offset** below)
11	degree-of-freedom rotation about z-axis (use **offset** below)
-11	degree-of-freedom radial about z-axis (use **offset** below)

vad *flag*

where *flag* can be

0	velocity
2	displacement
3	velocity versus displacement
4	relative displacement

lcid *load_curve_#*
sf *scale_factor*
v x_0 y_0 z_0
birth *time*
death *time*
offset $offset_1$ $offset_2$
mrb *rigid_material_#*
nodes $node_1$ $node_2$

rbv *load_curve_# amplitude x y z* rigid body velocity (obsolete)
rbd *load_curve_# amplitude x y z* rigid body displacement (obsolete)

For material type **21** (Thermal Orthotropic w/ 12 Constants)

ea *a-elastic_modulus*
eb *b-elastic_modulus*
ec *c-elastic_modulus*
prba *ba-poisson's_ratio*
prca *ca-poisson's_ratio*
prcb *cb-poisson's_ratio*
alpa *a-thermal_expansion*
alpb *b-thermal_expansion*
alpc *c-thermal_expansion*
gab *ab-shear_modulus*
gbc *bc-shear_modulus*
gca *ca-shear_modulus*
aopt *option*

where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	for by cross product with shell normal (shell elements only)
4	for by normal vectors in cylindrical coordinates

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
vx *x-component*	aopt 3 & 4
vy *y-component*	aopt 3 & 4
vz *z-component*	aopt 3 & 4
p1 *x-component*	aopt 4
p2 *y-component*	aopt 4
p3 *z-component*	aopt 4

For material type **22** (Fiber Composite w/ Damage)

ea *ea-elastic_modulus*
eb *eb-elastic_modulus*
ec *ec-elastic_modulus*
prba *vba-poissons_ratio*
prca *vca-poissons_ratio*

prcb *vcb-poissons_ratio*
gab *gab-shear_modulus*
gbc *gbc-shear_modulus*
gca *gca-shear_modulus*
k *k-bulk_modulus*
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 for by cross product with shell normal (shell elements only)
axes *flag* material axes change flag (brick elements only)
where *flag* can be
1 default
2 switch material axes a and b
3 switch material axes a and c
xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
sc *strength* shear strength on a-b plane
xt *strength* longitudinal tensile strength along a-axis, xt
yt *strength* longitudinal tensile strength along b-axis, yt
yc *strength* transverse compressive strength, yc
alpha *stress* nonlinear shear stress parameter
sn *strength*
syz *strength*
szx *strength*
beta *list_of_material_angles* aopt 3

For material type **23** (Temperature Dependent Thermal Orthotropic w/ 12 Curves)
aopt *option*
where the option can be one of
0 for by nodes

1 for by point and element center
2 for by normal vectors
3 for by cross product with shell normal (shell elements only)

xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
beta *list_of_material_angles* aopt 3

ea *ea-orthotropic_list* ;
eb *eb-orthotropic_list* ;
ec *ec-orthotropic_list* ;
vba *vba-orthotropic_list* ;
vca *vca-orthotropic_list* ;
vcb *vcb-orthotropic_list* ;
aa *a**l**pha_a-orthotropic_list ;*
ab *alpha_b-orthotropic_list* **;**
ac *alpha_c-orthotropic_list* **;**
gab *gab-orthotropic_list* **;**
gbc *gbc-orthotropic_list* **;**
gca *gca-orthotropic_list* **;**
t *temperatures_list* **;**

For material type **24** (Rate-Dependent Tabular Isotropic Plasticity)

e *young's_modulus*
pr *poisson's_ratio*
sigy *yield_stress*
et *tangent_modulus*
efp *effective_strain_failure*
dtcrit *time_step_size*
srp *p-strain_rate*
src *c-strain_rate*
lcss *effective_stress_curve_#*
lc *yield_stress_curve_#*
vp *rate_effect_formulation*

eps *effective_strain_list* ;
es *yield_stress_list* ;

For material type **25** (Inviscid, Two Invariant Geologic Cap)

k *initial_bulk_modulus*
g *initial_shear_modulus*
alpha *failure_envelope*
theta *linear_failure_envelope*
gamma *exponential_failure_envelope*
beta *exponent_failure_envelope*
r *axis_ratio*
d *hardening_law_exponent*
w *hardening_law_coefficient*
x0 *hardening_law_parameter*
cbar *kinematic_hardening_coefficient*
n *kinematic_hardening_parameter*
nplot *option*

where the plot data base option can be

1	for hardening variable, k
2	for cap - j1 axis intercept, x(k)
3	for volumetric plastic strain
4	for first stress invariant, j1
5	for second stress invariant, square root of j2d
8	for response mode number
9	for number of iterations

ltype *option*

where the formulation option can be

1	for soil or concrete (cap surface may contract)
2	for rock (cap surface does not contract)

ivec *option*

where the vectorization option can be

0	for vectorized (fixed number of iterations)
1	for fully iterative

t *tension_cutoff*

For material type **26** (Orthotropic Crushable Honeycomb)

e *young's_modulus*
pr *poisson's_ratio*
sigy *yield_stress*
crv *relative_volume*
mu *material_viscosity*
bulk *option*

where the bulk viscosity option

0 for bulk viscosity is not used (recommended)

1 for bulk viscosity is active & mu=0, like previous versions of ls-dyna

sigaa *sigma_aa_curve_#*
sigbb *sigma_bb_curve_#*
sigcc *sigma_cc_curve_#*
ssrv *shear_stress_curve_#*
sigab *sig_ab_curve_#*
sigbc *sig_bc_curve_#*
sigca *sig_ca_curve_#*
sre *strain-rate effects_curve_#*
ea *ea-elastic_modulus*
eb *eb-elastic_modulus*
ec *ec-elastic_modulus*
gab *gab-shear_modulus*
gbc *gbc-shear_modulus*
gca *gca-shear_modulus*
aopt *option*

where the option can be one of

0 for by nodes

1 for by point and element center

2 for by normal vectors

3 by cross product with shell normal(Shell elements only)

xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
tsf *tensile_strain*
ssf *shear_strain*

For material type **27** (Compressible Mooney-Rivlin Hyperelastic Rubber)

pr *poisson's_ratio*
m1 *specify_constants*
a *first_invariant*
b *second_invariant*
m2 *least_square_fit*
sgl *specimen_gauge_length*

sw *specimen_width*
st *specimen_thickness*
lcid *ce_vs_change_curve_#*

For material type **28** (Resultant plasticity)
e *young's_modulus*
pr *poisson's_ratio*
sigy *yield_stress*
etan *hardening_modulus*

For material type **29** (Force Limited Resultant Formulation for Beams)
e *young's_modulus*
pr *poisson's_ratio*
df *damping_factor*
dept *for axial collapse force is dependent on the bending moment*
indept *for axial collapse force is not dependent on the bending moment*
noax *for no axial collapse*
bten *option*
where the beam tension option can be

0	for beam does not yield in tension
1	for beam can yield in tension

lpr *torsional_moment_curve_#*
sfr *lpr_scale_factor*
ymr *torsional_yield_moment*
asoft *softening_factor*
lps1 *1st_plastic_s-moment_curve_#*
sfs1 *lps1_scale_factor*
lps2 *2nd_plastic_s-moment_curve_#*
sfs2 *lps2_scale_factor*
yms1 *1st_s-axis_yield_moment*
yms2 *2nd_s-axis_yield_moment*
lpt1 *1st_plastic_t-moment_curve_#*
sft1 *lpt1_scale_factor*
lpt2 *2nd_plastic_t-moment_curve_#*
sft2 *lpt2_scale_factor*
ymt1 *1st_t-axis_moment*
ymt2 *2nd_t-axis_moment*

For material type **30** (Closed-Form Update Plasticity Shell)
e *young's_modulus*
pr *poisson's_ratio*
sigy *yield_stress*

etan *tangent_modulus*

For material type **31** (Slightly Compressible Rubber Model)

pr *poisson's_ratio*
c100 *constant*
c200 *constant*
c300 *constant*
c400 *constant*
c110 *constant*
c210 *constant*
c010 *constant*
c020 *constant*
exct *exit_or_continue*
mxst *maximum_strain_limit*
mnst *minimum_strain_limit*
spgg *specimen_guage_length*
spwd *specimen_width*
sptk *specimen_thickness*
ldgl *load_curve*

For material type **32** (Laminated Glass Model)

e *young's_modulus*	glass
pr *poisson's_ratio*	glass
sigy *yield_stress*	glass
etan *hardening_modulus*	glass
psf *strain*	glass plastic strain at failure
pe *young's_modlus*	polymer
ppr *poisson's_ratio*	polymer
psigy *yield_stress*	polymer
petan *hardening_modulus*	polymer
imt *mat_list* ;	list of 0s & 1s for integration points

For material type **33** (Barlat's Anisotropic Plasticity Model)

e *young's_modulus*
pr *poisson's_ratio*
ck *constant*
eps0 *constant*
cn *constant*

cm *potential*	flow potential in Barlat's model
ca *coef*	anistropy coefficient in Barlat's model
cb *coef*	anistropy coefficient in Barlat's model
cc *coef*	anistropy coefficient in Barlat's model

cf *coef* anistropy coefficient in Barlat's model
cg *coef* anistropy coefficient in Barlat's model
ch *coef* anistropy coefficient in Barlat's model
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal(Shell elements only)
xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3

For material type **34** (Fabric Model)
ea *direction* longitudinal direction
eb *direction* transverse direction
ec *direction* normal direction
vba *poisson's_ratio*
vca *poisson's_ratio*
vcb *poisson's_ratio*
csflg *flag* compressive stress flag
where the flag can be
0 for don't eliminate compressive stresses
1 for eliminate compressive stresses
gab *coefficient*
gbc *coefficient*
gca *coefficient*
csflg *flag*
e *young's_modulus* Young's modulus for elastic liner
pr *poisson's_ratio* Poisson's ratio for elastic liner
rlf *ratio* ratio of liner thickness to total fabric thickness
rla *coef* Rayleigh damping coefficient
aopt *option*

where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	by cross product with shell normal (Shell elements only)

flc *fabric_leakage_coefficient*
fac *fabric_area_coefficient*
ela *effective_leakage_area*
lnrc *liner_compression_flag*
form *membrane_formulation_flag*

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
vx *x-component*	aopt 3
vy *y-component*	aopt 3
vz *z-component*	aopt 3
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
beta *list_of_material_angles*	aopt 3

For material type **35** (Kinematic/Isotropic Elastic-Plastic Green-Naghdi Rate Model)

e *young's_modulus*

src *c*	strain rate parameter, c
srp *p*	strain rate parameter, p

pr *poisson's_ratio*
sigy *yield_stress*
etan *hardening_modulus*
eb *hardening_parameter*

For material type **36** (Barlat's 3-Parameter Plasticity Model)

e *young's_modulus*
pr *poisson's_ratio*
hr *rule* hardening rule

where the rule can be

1	for linear model
2	for exponential model

cm *exponent* exponent in Barlat's yield surface
r00 *constant*

r45 *constant*
r90 *constant*
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal(Shell elements only)

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
vx *x-component*	aopt 3
vy *y-component*	aopt 3
vz *z-component*	aopt 3

For material type **37** (Transversely Anisotropic Elastic-Plastic)
e *young's_modulus*
pr *poisson's_ratio*
sigy *yield_stress*
etan *hardening_modulus*
er *r* anisotropic hardening parameter
ldss *load_curve* effective stress vs. effective plastic strain

For material type **38** (Blatz-Ko Compressible Foam)
sm *shear_modulus*

For material type **39** (Transversely Anisotropic Elastic-Plastic with FLD)
e *young's_modulus*
ldfl *load_curve* defining the flow limit diagram
pr *poisson's_ratio*
sigy *yield_stress*
etan *hardening_modulus*
er *r* anisotropic hardening parameter
ldss *load_curve* effective stress vs. effective plastic strain

For material type **40** (Nonlinear Elastic Orthotropic Material)

ex0 *direction* modulus-longitudinal direction
ey0 *direction* modulus-transverse direction
ez0 *direction* modulus-normal direction
vba *poisson's_ratio*
vca *poisson's_ratio*
vcb *poisson's_ratio*
dtm *dt* temperature increment for stress initialization
trmp *t-ramp* time to ramp up to the final temperature
alpha *alpha* thermal expansion coefficient
gba *shear_modulus*
gca *shear_modulus*
gcb *shear_modulus*
aopt *option*

where the option can be one of

0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal(Shell elements only)

xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
beta *list_of_material_angles* aopt 3

ldnsa *load_curve* nominal stress vs. a-axis strain
ldnsb *load_curve* nominal stress vs. b-axis strain
efail *epsilon-fail* failure strain
dtfail *dt-fail* time step for automatic element erosion
cdamp *c-damp* damping coefficient

For material type **41** (User-Defined Material Model #41)

card1 *parameters* ;
card2 *parameters* ;
card3 *parameters* ;
card4 *parameters* ;

card5 *parameters* ;
card6 *parameters* ;
card7 *parameters* ;
card8 *parameters* ;
card9 *parameters* ;
card10 *parameters* ;
aopt *option*

where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	by cross product with shell normal(Shell elements only)

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
vx *x-component*	aopt 3
vy *y-component*	aopt 3
vz *z-component*	aopt 3
mopt *flag*	material axes change flag for bricks

where flag can be

1	default
2	switch material axes a and b
3	switch material axes a and c

For material type **42** (Planar Anisotropic Plasticity Model)

e *young's_modulus*
pr *poisson's_ratio*
k *bulk_modulus*
g *shear_modulus*
ca *yield_parameter*

cm *coef*	plastic hardening coefficient
cn *coef*	strain rate coefficient
lcss *load_curve*	stress vs. effective plastic strain

r00 *constant*
r45 *constant*
r90 *constant*

smin *epsilon-min* minimum strain rate
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal(Shell elements only)
xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
beta *list_of_material_angles* aopt 3
mopt *flag* material axes change flag for bricks
where flag can be
1 default
2 switch material axes a and b
3 switch material axes a and c

For material type **43** (User-Defined Material Model #43)
card1 *parameters* ;
card2 *parameters* ;
card3 *parameters* ;
card4 *parameters* ;
card5 *parameters* ;
card6 *parameters* ;
card7 *parameters* ;
card8 *parameters* ;
card9 *parameters* ;
card10 *parameters* ;
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors

3 by cross product with shell normal(Shell elements only)

xp *x-coordinate* aopt 1

yp *y-coordinate* aopt 1

zp *z-coordinate* aopt 1

ax *x-component* aopt 2

ay *y-component* aopt 2

az *z-component* aopt 2

dx *x-component* aopt 2

dy *y-component* aopt 2

dz *z-component* aopt 2

vx *x-component* aopt 3

vy *y-component* aopt 3

vz *z-component* aopt 3

mopt *flag* material axes change flag for bricks

where flag can be

1 default

2 switch material axes a and b

3 switch material axes a and c

For material type **44** (User-Defined Material Model #44)

card1 *parameters* ;

card2 *parameters* ;

card3 *parameters* ;

card4 *parameters* ;

card5 *parameters* ;

card6 *parameters* ;

card7 *parameters* ;

card8 *parameters* ;

card9 *parameters* ;

card10 *parameters* ;

aopt *option*

where the option can be one of

0 for by nodes

1 for by point and element center

2 for by normal vectors

3 by cross product with shell normal(Shell elements only)

xp *x-coordinate* aopt 1

yp *y-coordinate* aopt 1

zp *z-coordinate* aopt 1

ax *x-component* aopt 2

ay *y-component* aopt 2

az *z-component* aopt 2

dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
mopt *flag* material axes change flag for bricks
where flag can be
1 default
2 switch material axes a and b
3 switch material axes a and c

For material type **45** (User-Defined Material Model #45)
card1 *parameters* ;
card2 *parameters* ;
card3 *parameters* ;
card4 *parameters* ;
card5 *parameters* ;
card6 *parameters* ;
card7 *parameters* ;
card8 *parameters* ;
card9 *parameters* ;
card10 *parameters* ;
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal (Shell elements only)
xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
mopt *flag* material axes change flag for bricks

where flag can be

1	default
2	switch material axes a and b
3	switch material axes a and c

For material type **46** (User-Defined Material Model #46)

card1 *parameters* ;
card2 *parameters* ;
card3 *parameters* ;
card4 *parameters* ;
card5 *parameters* ;
card6 *parameters* ;
card7 *parameters* ;
card8 *parameters* ;
card9 *parameters* ;
card10 *parameters* ;
aopt *option*

where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	by cross product with shell normal(Shell elements only)

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
vx *x-component*	aopt 3
vy *y-component*	aopt 3
vz *z-component*	aopt 3
mopt *flag*	material axes change flag for bricks

where flag can be

1	default
2	switch material axes a and b
3	switch material axes a and c

For material type **47** (User-Defined Material Model #47)

card1 *parameters* ;

card2 *parameters* ;
card3 *parameters* ;
card4 *parameters* ;
card5 *parameters* ;
card6 *parameters* ;
card7 *parameters* ;
card8 *parameters* ;
card9 *parameters* ;
card10 *parameters* ;
aopt *option*

where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	by cross product with shell normal(Shell elements only)

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
vx *x-component*	aopt 3
vy *y-component*	aopt 3
vz *z-component*	aopt 3
mopt *flag*	material axes change flag for bricks

where flag can be

1	default
2	switch material axes a and b
3	switch material axes a and c

For material type **48** (Strain Rate Dependent Plasticity with Size Dependent Failure)

k *bulk_modulus*
c1 *strain_rate*
p1 *strain_rate*
c2 *strain_rate*
p2 *strain_rate*
g *shear_modulus*

ldse *load_curve*	failure strain vs. element size
meth *method*	determining element size

where the method can be

0 square root of area
1 minimum side length used for time step
2 maximum side length (area/minimum side)
3 based on direction of strains in element

iint *max_points* fully integrated element failure option
pef *option* including pressure effects in fracture
where the option can be

0 ignore pressure sensitivity (default)
1 include pressure sensitivity

ystr *yield_strength*
ldss *load_curve* effective stress vs. effective plastic stress
etan *tangent_modulus*
psf *strain* plastic strain at failure
dt *step* time step size for automatic element deletion
lc1 *load_curve* scale yield stress
lc2 *load_curve* scale yield stress
eps *strain_list* ; effective plastic strain values (up to 8)
sigy *stress_list* ; yield stress values (up to 8)

For material type **49** (User-Defined Material Model #49)

card1 *parameters* ;
card2 *parameters* ;
card3 *parameters* ;
card4 *parameters* ;
card5 *parameters* ;
card6 *parameters* ;
card7 *parameters* ;
card8 *parameters* ;
card9 *parameters* ;
card10 *parameters* ;
aopt *option*
where the option can be one of

0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal(Shell elements only)

xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2

az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
mopt *flag* material axes change flag for bricks
where flag can be
1 default
2 switch material axes a and b
3 switch material axes a and c

For material type **50** (User-Defined Material Model #50)
card1 *parameters* ;
card2 *parameters* ;
card3 *parameters* ;
card4 *parameters* ;
card5 *parameters* ;
card6 *parameters* ;
card7 *parameters* ;
card8 *parameters* ;
card9 *parameters* ;
card10 *parameters* ;
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal(Shell elements only)
xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3

mopt *flag* material axes change flag for bricks
where flag can be
1 default
2 switch material axes a and b
3 switch material axes a and c

For material type **51** (Temperature and Rate Dependent Plasticity)
e *young's_modulus*
pr *poisson's_ratio*
t *initial_temperature*
hc *coef* heat generation coefficient
c1 *constant*
c2 *constant*
c3 *constant*
c4 *constant*
c5 *constant*
c6 *constant*
c7 *constant*
c8 *constant*
c9 *constant*
c10 *constant*
c11 *constant*
c12 *constant*
c13 *constant*
c14 *constant*
c15 *constant*
c16 *constant*
c17 *constant*
c18 *constant*
al1 *alpha1* initial value of state variable 1
al2 *alpha2* initial value of state variable 2
al4 *alpha4* initial value of state variable 3
al5 *alpha5* initial value of state variable 4
al6 *alpha6* initial value of state variable 5
kap *kappa* initial value of state variable 6

For material type **52** (SANDIA's Damage Model)
e *young's_modulus*
pr *poisson's_ratio*
t *initial_temperature*
hc *coef* heat generation coefficient

c1 *constant*
c2 *constant*
c3 *constant*
c4 *constant*
c5 *constant*
c6 *constant*
c7 *constant*
c8 *constant*
c9 *constant*
c10 *constant*
c11 *constant*
c12 *constant*
c13 *constant*
c14 *constant*
c15 *constant*
c16 *constant*
c17 *constant*
c18 *constant*

al1 *alpha1*	initial value of state variable 1
al2 *alpha2*	initial value of state variable 2
al4 *alpha4*	initial value of state variable 3
al5 *alpha5*	initial value of state variable 4
al6 *alpha6*	initial value of state variable 5
kap *kappa*	initial value of state variable 6
exp *exponent*	damage evolution
por *porosity*	do initial damage

For material type **53** (Low Density Closed Cell Polyurethane Foam)

e *young's_modulus*
ca *constant*
cb *constant*
cc *constant*

p0 *pressure*	initial foam pressure
phi *ratio*	foam to polymer density
g0 *gamma0*	initial volumetric strain

For material type **54** (Composite Damage Model (Chang matrix failure))

ex *direction*	longitudinal direction
ey *dircetion*	transverse direction
ez *dircetion*	normal direction

vba *poisson's_ratio*
vca *poisson's_ratio*

vcb *poisson's_ratio*
gba *shear_modulus*
gca *shear_modulus*
gcb *shear_modulus*
aopt *option*
where the option can be one of

0	for by nodes
1	for by point and element center
2	for by normal vectors
3	by cross product with shell normal (Shell elements only)

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
vx *x-component*	aopt 3
vy *y-component*	aopt 3
vz *z-component*	aopt 3
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2

dfailm *maximum_strain_for_matrix*
dfails *maximum_shear_strain*

tfail *step*	time step for element deletion
ns *stress*	nonlinear shear stress parameter
soft *factor*	softening reduction factor for material strength in crashfront elements
frbt *softening*	fiber tensile strength
ycfac *compression_strength*	remainder longitudinal compression strength after compressive matrix
dfailt *strain*	failure strain for tensile fiber mode
dfailc *strain*	failure strain (negative) for compressive fiber mode

efs *effective_failure_strain*

xc *strength*	longitudinal compressive strength
xt *strength*	longitudinal tensile strength, a-axis
yc *strength*	transverse compressive strength
yt *strength*	transverse tensile strength, b-axis
sc *strength*	shear strength, ab plane
bt *factor*	weighting factor for shear term in tensile fiber mode
beta *list_of_material_angles*	aopt 3

For material type **55** (Composite Damage Model (tsay-wu matrix failure))

ex *direction* longitudinal direction
ey *dircetion* transverse direction
ez *dircetion* normal direction
vba *poisson's_ratio*
vca *poisson's_ratio*
vcb *poisson's_ratio*
gba *shear_modulus*
gca *shear_modulus*
gcb *shear_modulus*
aopt *option*
where the option can be one of
0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal (Shell elements only)
xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
dfailm *maximum_strain_for_matrix*
dfails *maximum_shear_strain*
tfail *step* time step for element deletion
ns *stress* nonlinear shear stress parameter
soft *factor* softening reduction factor for material strength in crashfront elements
frbt *softening* fiber tensile strength
ycfac *compression_strength* remainder longitudinal compression strength after compressive matrix
dfailt *strain* failure strain for tensile fiber mode
dfailc *strain* failure strain (negative) for compressive fiber mode
efs *effective_failure_strain*
xc *strength* longitudinal compressive strength
xt *strength* longitudinal tensile strength, a-axis
yc *strength* transverse compressive strength
yt *strength* transverse tensile strength, b-axis

sc *strength* shear strength, ab plane
bt *factor* weighting factor for shear term in tensile fiber mode
beta *list_of_material_angles* aopt 3

For material type **57** (Low Density Urethane Foam)
e *young's_modulus*
ldns *load_curve* nominal stress vs. strain
tcut *stress* tension cut-off stress
hunl *factor* hysteretic unloading factor
beta *decay*
vc *viscous_coefficient* stress oscillations and shock waves
sf *factor* shape factor for unloading
fopt *option* failure option after cutoff stress is reached
where the failure option can be
0 tensile stress remains at cut-off value
1 tensile stress is reset to zero
kflg *flag* bulk viscosity activation flag
where the flag can be
0 no bulk viscosity (recommended)
1 bulk viscosity active
oe *modulus* optional young's relaxation modulus for rate effects
obeta *decay* optional decay constant
sc *stiffness_coefficient* contact interface stiffness

For material type **59** (Composite Failure Model - Plasticity Based)
ea *direction* longitudinal direction
eb *direction* transverse direction
ec *direction* normal direction
kf *modulus* bulk modulus of failed material
sr *factor* reduction factor (default=0.447) (shells)
sf *factor* softening factor (default=0.0) (shells)
vba *poisson's_ratio*
vca *poisson's_ratio*
vcb *poisson's_ratio*
sba *strength* in plane shear strength (bricks)
sca *strength* transverse shear strength (bricks)
scb *strength* transverse shear strength (bricks)
gba *shear_modulus*
gca *shear_modulus*
gcb *shear_modulus*
xc *strength* long. compressive strength, a-axis
yc *strength* trans. compressive strength, b-axis

zc *strength* normal compressive strength, c-axis

aopt *option*

where the option can be one of

0 for by nodes

1 for by point and element center

2 for by normal vectors

3 by cross product with shell normal(Shell elements only)

xp *x-coordinate*	aopt 1
yp *y-coordinate*	aopt 1
zp *z-coordinate*	aopt 1
ax *x-component*	aopt 2
ay *y-component*	aopt 2
az *z-component*	aopt 2
dx *x-component*	aopt 2
dy *y-component*	aopt 2
dz *z-component*	aopt 2
vx *x-component*	aopt 3
vy *y-component*	aopt 3
vz *z-component*	aopt 3

mopt *flag* material axes change flag for bricks

where flag can be

1 default

2 switch material axes a and b

3 switch material axes a and c

tsize *step* time step for automatic element deletion

alp *stress* nonlinear shear stress parameter

soft *factor* softening reduction factor for strength in crash

fbrt *strength* softening of fiber tensile strength

xt *strength* long. tensile strength, a-axis (bricks)

yt *strength* trans. tensile strength, b-axis (bricks)

zt *strength* normal tensile strength, c-axis (bricks)

xcs *strength* long. compressive strength, a-axis (shells)

xts *strength* long. tensile strength, a-axis (shells)

ycs *strength* trans. compressive strength, b-axis (shells)

yts *strength* trans. tensile strength, b-axis (shells)

sc *strength* shear strength, ab-plane (shells)

For material type **60** (Elastic with Viscosity)

e *young's_modulus*

visc *viscosity*

ca *viscosity_coefficient*

cb *viscosity_coefficient*

cc *viscosity_coefficient*
ldft *load_curve* defining factor vs. time
prs *ratio_list* ; Poisson's ratio (up to 8 values)
ts *temp* ; temperature values (up to 8 values)
viscs *viscosity_list* ; viscosity values (up to 8 values)
es *modulus_list* ; Young's modulus values (up to 8 values)
ctes *coef_list* ; coefficient of thermal expansion (up to 8 values)

For material type **61** (Maxwell/Kelvin Viscoelastic with Maximum Strain)
k *bulk_modulus* elastic
g0 *modulus* short-time shear modulus
ginf *modulus* long-time shear modulus
form *option* formulation option
where the option can be
0 Maxwell
1 Kelvin
sopt *option* strain output option
where the option can be
0 maximum principal strain
1 maximum magnitude of principal strain
2 maximum effective strain

For material type **62** (Viscous Foam, Ove Arup & Partners Model)
e1 *modulud* initial Young's modulus
n1 *exponent* power law for Young's modulus
v2 *viscous_coefficient*
e2 *modulud* elastic modulus for viscosity
n2 *exponent* power law for viscosity
pr *poisson's_ratio*

For material type **63** (Crushable foam)
e *young's_modulus*
pr *poisson's_ratio*
ldyv *load_curve* yield stress vs. volumetric strain
tcut *stress* cutoff value for tensile stress
visc *viscous_coefficient*

For material type **64** (Strain Rate Sensitive Power-Law Plasticity)
me *modulus* modulus of elasticity
pr *poisson's_ratio*
mk *material_constant*
sm *coef* strain hardening coefficient

sn *coef* strain rate sensitivity coefficient
is *initial_strain*

For material type **65** (Modified Zerilli/Armstrong)

cg *constant*
ceps0 *constant*
cn *constant*
tr *room_temperature*
pc *pressure_cutoff*
sp *spall_type*
where the spall type can be
1 Minimum pressure limit
2 Maximum principal stress (default)
3 Minimum pressure cutoff
fs *strain* failure strain for erosion
c1 *constant*
c2 *constant*
c3 *constant*
c4 *constant*
c5 *constant*
c6 *constant*
b1 *constant*
b2 *constant*
b3 *constant*
g1 *constant*
g2 *constant*
g3 *constant*
g4 *constant*

For material type **66** (Linear Stiffness/Linear Viscous 3d Discrete Beam)

vold *volume* volume of Discrete Beam
lump *inertia* lumped inertia
cabrx rotational constraint r-direction
cabry rotational constraint s-direction
cabrz rotational constraint t-direction
tsr *stiffness* translational stiffness along r-axis
tss *stiffness* translational stiffness along s-axis
tst *stiffness* translational stiffness along t-axis
rsr *stiffness* rotational stiffness about r-axis
rss *stiffness* rotational stiffness about s-axis
rst *stiffness* rotational stiffness about t-axis
tvr *stiffness* translational viscous damper along r-axis
tvs *stiffness* translational viscous damper along s-axis
tvt *stiffness* translational viscous damper along t-axis
rvr *damper* rotational viscous damper about r-axis

rvs *damper* rotational viscous damper about s-axis
rvt *damper* rotational viscous damper about t-axis

For material type **67** (Nonlinear Stiffness/Viscous 3d Discrete Beam)

vold *volume* volume of Discrete Beam
lump *inertia* lumped inertia
cablcid *system_#* local coordinate system id number defined by the **lsys**
cabrx rotational constraint r-direction
cabry rotational constraint s-direction
cabrz rotational constraint t-direction
ldrr *load_curve* r-axis force vs. r-axis displacement
ldss *load_curve* s-axis force vs. s-axis displacement
ldtt *load_curve* t-axis force vs. t-axis displacement
ldmrr *load_curve* r-axis moment vs. r-axis rotation
ldmss *load_curve* s-axis moment vs. s-axis rotation
ldmtt *load_curve* t-axis moment vs. t-axis rotation
ldvrr *load_curve* r-axis damping force vs. r-axis rotational velocity
ldvss *load_curve* s-axis damping force vs. s-axis rotational velocity
ldvtt *load_curve* t-axis damping force vs. t-axis rotational velocity
ldvmrr *load_curve* r-axis damping moment vs. r-axis rotational velocity
ldvmss *load_curve* s-axis damping moment vs. s-axis rotational velocity
ldvmtt *load_curve* t-axis damping moment vs. t-axis rotational velocity

For material type **68** (Nonlinear Plastic/Linear Viscous 3d Discrete Beam)

vold *volume* volume of Discrete Beam
lump *inertia* lumped inertia
cablcid *system_#* local coordinate system id number defined by the **lsys**
cabrx rotational constraint r-direction
cabry rotational constraint s-direction
cabrz rotational constraint t-direction
tsr *stiffness* translational stiffness along r-axis
tss *stiffness* translational stiffness along s-axis
tst *stiffness* translational stiffness along t-axis
rsr *stiffness* rotational stiffness about r-axis
rss *stiffness* rotational stiffness about s-axis
rst *stiffness* rotational stiffness about t-axis
tvr *damper* translational viscous damper along r-axis
tvs *damper* translational viscous damper along s-axis
tvt *damper* translational viscous damper along t-axis
rvr *damper* rotational viscous damper about r-axis
rvs *damper* rotational viscous damper about s-axis
rvt *damper* rotational viscous damper about t-axis

ldyr *load_curve* yield force vs. plastic displacement, r-axis
ldys *load_curve* yield force vs. plastic displacement, s-axis
ldyt *load_curve* yield force vs. plastic displacement, t-axis
ldmr *load_curve* yield moment vs. plastic rotation, r-axis
ldms *load_curve* yield moment vs. plastic rotation, s-axis
ldmt *load_curve* yield moment vs. plastic rotation, t-axis
fr *para* fr-fail optional parameter
fs *para* fs-fail optional parameter
ft *para* ft-fail optional parameter
mr *para* mr-fail optional parameter
ms *para* ms-fail optional parameter
mt *para* mt-fail optional parameter
ur *para* ur-fail optional parameter
us *para* us-fail optional parameter
ut *para* ut-fail optional parameter
tr *para* thetar-fail optional parameter
ts *para* thetas-fail optional parameter
tt *para* thetaj-fail optional parameter

For material type **69** (Side Impact Dummy Damper, sid Damper)

vold *volume* volume of Discrete Beam
lump *inertia* lumped inertia
cablcid *system_#* local coordinate system id number defined by the **lsys**
cabrx rotational constraint r-direction
cabry rotational constraint s-direction
cabrz rotational constraint t-direction
st *st* piston stroke
cd *d* piston diameter
cr *r* default orifice radius
ch *h* orifice controller position
ck *k* damping constant
cc *c* discharge coefficient
ckk *k* stiffness coefficient if piston bottoms out
cds *list* ; orifice locations relative to fixed end (no more than 16)
crs *list* ; radii corresponding to orifice locations
rho *rho* fluid density
c1 *c1* coefficient for linear velocity term
c2 *c2* coefficient for quadratic velocity term
ldfd *load_curve* force vs. piston displacement
lddd *load_curve* damping coefficient vs. piston displacement
d0 *s0* initial displacement (typically 0)
c3 *c3* coefficient for fluid inertia term

For material type **70** (Hydraulic/Gas Damper Model)

vold *volume* volume of Discrete Beam
lump *inertia* lumped inertia
cablcid *system_#* local coordinate system id number defined by the **lsys**
cabrx rotational constraint r-direction
cabry rotational constraint s-direction
cabrz rotational constraint t-direction
c0 *c0* length of gas column
cn *n* adiabatic constant
p0 *p0* initial gas pressure
pa *pa* atmospheric pressure
ap *ap* piston cross sectional area
kh *kh* hydraulic constant
ldn *load_curve* orifice area vs. element deflection
fr *fr* return factor on orifice force
sclf *sclf* scale factor on force
clr *clearance*

For material type **71** (Cable Model)

vold *volume* volume of Discrete Beam
lump *inertia* lumped inertia
cabarea *area* cable area
caboff *offset* cable offset
cabrx rotational constraint r-direction
cabry rotational constraint s-direction
cabrz rotational constraint t-direction
e *young's_modulus*
ldn *load_curve* stress vs. strain

For material type **72** (Concrete Damage Model0

pr *pr* Constant Poission's Ratio Model
sigf *failure* Maximum Principal Stress Failure
a0 *cohesion* Cohesion
a1 *coefficient* Fiirst Pressure Hardening Coefficient
a2 *coefficient* Second Pressure Hardening Coefficient
b1 *factor* Damage Scaling Factor
a1f *coefficient* Pressure Hardening Coefficient for Failed Material
per *percent* Percent Reinforcement
er *elastic modulus* Elastic Modulus for Reinforcement
prr *poisson's ratio* Poisson's Ratio for Reinforcement
sigy *yield stress* Initial Yield Stress for Reinforcement
etan *tangent modulus* Tangent Modulus for Reinforcement

lcp *load curve* Load Curve Giving Rate Sensitivity for Principal Material
lcr *load curve* Load Curve giving Rate Sensitivity for Reinforcement
lambda *tabulated values (up to 13)* Damage Function
b3 *factor* Damage Scale Factor for Triaxial Tensile Path
a0y *cohesion* Cohesion for Yield Limit
a1y *hardening* Pressure Hardening for Yield Limit
eta *tabulated values (up to 13)* Scale Factor Function
b2 *factor* Damage Scaling Factor for Triaxial Tension Path
a2f *coefficient* Pressure Hardening Coefficient for Failed Material
a2y *coefficient* Pressure Hardening Coefficient for Yield Limit

For material type **73** (Low Density Viscoelastic Foam)

e *young's modulus* Young's Modulus
lcid *load_curve* Load Curve number of nominal stress versus strain
tc *cut-off_stress* Tension Cut-off Stress
hu *factor* Hysteretic Unloading Factor
beta *decay_constant* Decay Constant to Model Creep in Unloading
damp *coefficient* Viscous Coefficient for Stress Oscillations & Shock Waves
shape *factor* Shape Factor for Unloading
fail *flag* Failure Option After Cutoff Stres is Reached
where *flag* can be :
0 Tensile Stress remains at Cut-off Value
1 Tensile Stress is reset to zero
bvflag *flag* Bulk Viscosity Activation Flag
where *flag* can be :
0 No Bulk Viscosity (Recommended)
1 Bulk Viscosity Active
kcon *coefficient* Stiffness Coefficient for Contact Interface Stiffness

Viscoelastic Option

Calculate the Viscoelastic Constants

lcid2 *load_curve* Load Curve if Constants beta-t are determined via Least Squares Fit
bstart *initial_value* Starting Beta
nt *number_of_terms* Number of Terms in the Fit
tramp *time* Optional Ramp Time for Loading

List the Viscoelastic Constants

g1 *maxwell_constant* Maxwell Constant G1

beta1 *decay_constant* Decay Constant beta1
g2 *maxwell_constant* Maxwell Constant G2
beta2 *decay_constant* Decay Constant beta2
g3 *maxwell_constant* Maxwell Constant G3
beta3 *decay_constant* Decay Constant beta3
g4 *maxwell_constant* Maxwell Constant G4
beta4 *decay_constant* Decay Constant beta4
g5 *maxwell_constant* Maxwell Constant G5
beta5 *decay_constant* Decay Constant beta5
g6 *maxwell_constant* Maxwell Constant G6
beta6 *decay_constant* Decay Constant beta6

For material type **75** (Bikhu/Dubois Foam Model)

e *young's_modulus*
ld1 *load* curve pressure for plastic yielding vs. volumetric strain
ld2 *load* curve uniaxial yield stress vs. volumetric strain
visc *viscous_coefficient*
pcut *pressure_cutoff*

For material type **76** (General Viscoelastic)

k *bulk_modulus* Constant Elastic Bulk Modulus

Viscoelastic Option

Calculate the Viscoelastic Constants

lcid2 *load_curve* Load Curve if Constants beta-t are determined via Least Squares Fit
bstart *initial_value* Starting Beta
nt *number_of_terms* Number of Terms in the Fit
tramp *time* Optional Ramp Time for Loading

List the Viscoelastic Constants

g1 *maxwell_constant* Maxwell Constant G1
beta1 *decay_constant* Decay Constant beta1
g2 *maxwell_constant* Maxwell Constant G2
beta2 *decay_constant* Decay Constant beta2
g3 *maxwell_constant* Maxwell Constant G3
beta3 *decay_constant* Decay Constant beta3
g4 *maxwell_constant* Maxwell Constant G4
beta4 *decay_constant* Decay Constant beta4

g5 *maxwell_constant* Maxwell Constant G5
beta5 *decay_constant* Decay Constant beta5
g6 *maxwell_constant* Maxwell Constant G6
beta6 *decay_constant* Decay Constant beta6

Volumetric Relaxation Option

Calculate the Volumetric Relaxation

lcidk *load_curve* Load Curve if Constants beta-t are determined via Least Squares Fit
bstartk *initial_value* Volumetric Starting Beta
ntk *number_of_terms* Volumetric Number of Terms in the Fit
tramp *time* Volumetric Ramp Time for Loading

List the Viscoelastic Constants

k1 *maxwell_constant* Maxwell Constant K1
betak1 *decay_constant* Decay Constant beta-k1
k2 *maxwell_constant* Maxwell Constant K2
betak2 *decay_constant* Decay Constant beta-k2
k3 *maxwell_constant* Maxwell Constant K3
betak3 *decay_constant* Decay Constant beta-k3
k4 *maxwell_constant* Maxwell Constant K4
betak4 *decay_constant* Decay Constant beta-k4
k5 *maxwell_constant* Maxwell Constant K5
betak5 *decay_constant* Decay Constant beta-k5
k6 *maxwell_constant* Maxwell Constant K6
betak6 *decay_constant* Decay Constant beta-k6

For material type **77** (Hyperviscoelastic Rubber)
pr *poisson's_ratio* Poisson's Ratio
formf *flag* Formulation Flag
where *flag* can be :
0 Strain Energy Functional

Calculate the Viscoelastic Constants
n *order* Order of Fit To Experimental Data
nv *number_of_terms* Number of Terms in Fit
lcid2 *load_curve* Bt Least Square Fit Load Curve
sgl *specimen_gauge_length* Specimen Gauge Length
sw *specimen_width* Specimen Width

st *specimen_thickness* Specimen Thickness
lcid1 *load_curve* Force Versus Actual Change Load Curve
tramp *time* Ramp Time for Loading

n 0 List the Viscoelastic Constants

c01 *coefficient* General Hyperelastic Coefficient C01
c11 *coefficient* General Hyperelastic Coefficient C11
c20 *coefficient* General Hyperelastic Coefficient C20
c02 *coefficient* General Hyperelastic Coefficient C02
g1 *maxwell_constant* Maxwell Constant G1
beta1 *decay_constant* Decay Constant beta1
g2 *maxwell_constant* Maxwell Constant G2
beta2 *decay_constant* Decay Constant beta2
g3 *maxwell_constant* Maxwell Constant G3
beta3 *decay_constant* Decay Constant beta3
g4 *maxwell_constant* Maxwell Constant G4
beta4 *decay_constant* Decay Constant beta4
g5 *maxwell_constant* Maxwell Constant G5
beta5 *decay_constant* Decay Constant beta5
g6 *maxwell_constant* Maxwell Constant G6
beta6 *decay_constant* Decay Constant beta6

1 Ogden Model

Calculate the Viscoelastic Constants
n *order* Order of Fit To Experimental Data
nv *number_of_terms* Number of Terms in Fit
lcid2 *load_curve* Bt Least Square Fit Load Curve
sgl *specimen_gauge_length* Specimen Gauge Length
sw *specimen_width* Specimen Width
st *specimen_thickness* Specimen Thickness
data 1 Biaxial Data
lcid1 *load_curve* Force Versus Actual Change Load Curve
tramp *time* Ramp Time for Loading

n 0 List the Viscoelastic Constants

mu1 *coefficient* Ogden Coefficient Mu1
mu2 *coefficient* Ogden Coefficient Mu2

mu3 *coefficient* Ogden Coefficient Mu3
mu4 *coefficient* Ogden Coefficient Mu4
mu5 *coefficient* Ogden Coefficient Mu5
mu6 *coefficient* Ogden Coefficient Mu6
mu7 *coefficient* Ogden Coefficient Mu7
mu8 *coefficient* Ogden Coefficient Mu8
alpha1 *coefficient* Ogden Coefficient Alpha1
alpha2 *coefficient* Ogden Coefficient Alpha2
alpha3 *coefficient* Ogden Coefficient Alpha3
alpha4 *coefficient* Ogden Coefficient Alpha4
alpha5 *coefficient* Ogden Coefficient Alpha5
alpha6 *coefficient* Ogden Coefficient Alpha6
alpha7 *coefficient* Ogden Coefficient Alpha7
alpha8 *coefficient* Ogden Coefficient Alpha8
g1 *maxwell_constant* Maxwell Constant G1
beta1 *decay_constant* Decay Constant beta1
g2 *maxwell_constant* Maxwell Constant G2
beta2 *decay_constant* Decay Constant beta2
g3 *maxwell_constant* Maxwell Constant G3
beta3 *decay_constant* Decay Constant beta3
g4 *maxwell_constant* Maxwell Constant G4
beta4 *decay_constant* Decay Constant beta4
g5 *maxwell_constant* Maxwell Constant G5
beta5 *decay_constant* Decay Constant beta5
g6 *maxwell_constant* Maxwell Constant G6
beta6 *decay_constant* Decay Constant beta6

For material type **78** (Solid/Concrete)

g *shear_modulus* Shear Modulus
k *bulk_modulus* Bulk Modulus
lcpv *load_curve* Pressure/Volumetric Strain Load Curve
lcfp *load_curve* Plastic Strain/Pressure Load Curve
lcrp *load_curve* Plastic Strain/Residual Strength Load Curve
pc *pressure_cutoff* Pressure Cutoff for Plastic Strain
out *flag* Plastic Strain Output
where *flag* can be
0 for VolumetricPlastic Strain
1 for DeviatoricPlastic Strain
b *factor* Residual Strength Factor After Cracking
fail *flag* Failure of Element Flag
where *flag* can be
0 for No Failure

1 for After cut-off is reached, Element is Eroded
2 for After cut-off is reached, Tension is no longer carried

Choose only one:

lcvm *load_curve* Yield vs. Von Mises pressure Load Curve
lcyp *load_curve* Second Stress Invariant, J2, Yield/Pressure Load Curve

For material type **79** (Hysteretic Soil)

k0 *bulk_modulus* Bulk Modulus at the Reference Pressure
p0 *pressure_cutoff* Cutoff/Datum Pressure
b *exponent* Exponent to Pressure-Sensitive Moduli
a0 *constant* Yield Function Constant a0
a1 *constant* Yield Function Constant a1
a2 *constant* Yield Function Constant a2
df *flag* Damping Factor

where *flag* can be:

0 No Damping
1 Maximum Damping

rp *pressure* Reference Pressure
gam1 *strain* Shear Strain Gamma1
gam2 *strain* Shear Strain Gamma2
gam3 *strain* Shear Strain Gamma3
gam4 *strain* Shear Strain Gamma4
gam5 *strain* Shear Strain Gamma5
tau1 *stress* Shear Stress Tau1
tau2 *stress* Shear Stress Tau2
tau3 *stress* Shear Stress Tau3
tau4 *stress* Shear Stress Tau4
tau5 *stress* Shear Stress Tau5
lcid *load_curve* Load Curve Defining Shear Strain versus Shear Stress
sfld *factor* Scale Factor to Apply to Shear Stress in LCID

For material type **80** (Ramberg-Osgood Plasticity)

gy *strain* Reference Shear Strain
ty *stress* Reference Shear Stress
a *coefficient* Stress Coefficient
r *exponent* Stress Exponent
k *elastic_modulus* Elastic Bulk Modulus

For material type **81** (Plastic with Damage)

e *young's_modulus* Young's Modulus
cc *parameter* Strain Rate Parameter C

p *parameter* Strain Rate Parameter p
pr *poisson's_ratio* Poisson's Ratio
sigy *yield_stress* Yield Stress
eppf *strain* Plastic Strain at Failure
tdel *time_step* Automatic Element Deletion Time Step
lcsr *load_curve* Load Curve to Scale Yield Stress
eppfr *strain* Plastic Strain at Rupture
Effective Stress/Effective Plastic Strain
by Load Curve
lcss *load_curve* Effective Stress/Effective Plastic Strain Load Curve
etan *tangent_modulus* Tangent Modulus
by Strain & Stress Values (up to 8 values)
eps *list_of_effective_plastic_strains* Effective Plastic Strain Values
es *list_of_effective_plastic_stress* Yield Stress Values

For material type **83** (Fu-Chang"s Foam with Rate Effects)
e *young's_modulus* Young's Modulus for Tensile Strains
ed *coefficient* Stiffness Coefficient for Contact Interface Stiffness
tc *stress* Tension Cut-off Stress
fail *flag* Failure Option After Cutoff Stress is Reached
where *flag* can be
0 Tensile Stress remains in Cut-off Value
1 Bulk Viscosity Active

damp *coefficient* Viscous Coefficient
tbid *load_curve* Stress-Strain/Strain Rate Table
bvflag *flag* Bulk Viscosity Activation Flag
where *flag* can be :
0 No Bulk Viscosity (Recommended)
1 Bulk Viscosity Active
d0 *constant* Material Constant d0
n0 *constant* Material Constant n0
n1 *constant* Material Constant n1
n2 *constant* Material Constant n2
n3 *constant* Material Constant n3
c0 *constant* Material Constant c0
c1 *constant* Material Constant c1
c2 *constant* Material Constant c2
c3 *constant* Material Constant c3
c4 *constant* Material Constant c4
c5 *constant* Material Constant c5
aij *constant* Material Constant aij

sij *constant* Material Constant sij
ratemin *strain_rate* Minimum Strain Rate
ratemax *strain_rate* Maximum Strain Rate

For material type **86** (Orthotropic-Viscoelastic)

Ea *constant* Ea
Eb *constant* Eb
Ec *constant* Ec
vf *friction* Volume Friction of Viscoelastic Material
k *bulk_modulus* Elastic Bulk Modulus, K
g0 *shear_modulus* Short-Time Shear Modulus, G0
ginf *shear_modulus* Long-Time Shear Modulus, Ginf
b *decay_constant* Decay Constant, Beta
prba *poisson's_ratio* Vba, Poisson's ratio
prca *poisson's_ratio* Vca, Poisson's ratio
prcb *poisson's_ratio* Vcb, Poisson's ratio
gab *constant* Gab
gbc *constant* Gbc
gca *constant* Gca
aopt *option*

where the option can be one of

0 for by nodes
1 for by point and element center
2 for by normal vectors
3 by cross product with shell normal(Shell elements only)

xp *x-coordinate*	aopt 1	
yp *y-coordinate*	aopt 1	
zp *z-coordinate*	aopt 1	
ax *x-component*	aopt 2	
ay *y-component*	aopt 2	
az *z-component*	aopt 2	
dx *x-component*	aopt 2	
dy *y-component*	aopt 2	
dz *z-component*	aopt 2	
vx *x-component*	aopt 3	
vy *y-component*	aopt 3	
vz *z-component*	aopt 3	
beta *beta*	aopt 3	Material angle

For material type **87** (Cellular Rubber)

pr *poisson's_ratio* Poisson's Ratio

p0 *pressure* Initial Air Pressure
phi *ratio* Ratio of Cellur Rubber to Rubber Density
ivs *strain* Initial Volumetric Strain
g *modulus* Optional Shear Relaxation Modulus for Rate Effects
b *constant* Optional Decay Constant
Computed Viscoelastic Parameters
n *order* Order of Fit
sgl *specimen_gauge_length* Specimen Gauge Length
sw *specimen_width* Specimen Width
st *specimen_thickness* Specimen Thickness
lcid *load_curve* Force/Actual Change in Guage Length Load Curve
n 0 Viscoelastic Parameters
c01 *coefficient* General Hyperelastic Coefficient C01
c11 *coefficient* General Hyperelastic Coefficient C11
c20 *coefficient* General Hyperelastic Coefficient C20
c02 *coefficient* General Hyperelastic Coefficient C02

For material type **88** (Mechanical Threshold Stress)
siga *sigma* Dislocation Interactions with Long Range Barriers (force/area)
sigi *sigma* Dislocation Interactions with Interstitial Atoms (force/area)
sigs *sigma* Dislocation Interactions with Solute Atoms (force/area)
sig0 *sigma* Initial Value of Sigma at Zero Plastic Strain (force/area)
hf0 *constant* Dislocation Generation Material Constant (force/area) HF0
hf1 *constant* Dislocation Generation Material Constant (force/area) HF1
hf2 *constant* Dislocation Generation Material Constant (force/area) HF2
sigs0 *sigma* Saturation Threshold Stress at 0 K (force/area)
edots0 *strain_rate* Reference Strain Rate EDOTS0 (1/time)
burg *magnitude* Magnitude of Burger's Vector (interatomic slip distance), (distance)
capa *constant* Material Constant CAPA
boltz *constant* Boltzmann's Constant (energy/degree)
sm0 *shear_modulus* Shear Modulus at Zero Degrees Kelvin
sm1 *shear_modulus* Shear Modulus Constant (force/area) SM1
sm2 *shear_modulus* Shear Modulus Constant (force/area) SM2
edot0 *strain_rate* Reference Strain-Rate EDOT0 (1/time)
g0 *energy* Normalized Activation Energy for Dislocation/Dislocation interaction
pinv *constant* Material Constant PINV
qinv *constant* Material Constant QINV
edoti *strain_rate* Reference Strain Rate EDOTI (1/time)
g0i *energy* Normalized Activation Energy for Dislocation/Interstitial Interaction
pinvi *constant* Material Constant PINVI
qinvi *constant* Material Constant QINVI

edots *strain_rate* Reference Strain-Rate EDOTS (1/time)
g0s *energy* Normalized Activation Energy for Dislocation/Solute Interaction
pinvs *constant* Material Constant PINVS
qinvs *constant* Material Constant QINVS
rhocpr *product* Product of Density and Specific Heat
temprf *initial_temperature* Initial Element Temperature in Degrees K
bulk *bulk_modulus* Bulk Modulus for Shell Elements
alpha *constant* Material Constant ALPHA
eps0 *factor* Factor to Normalize Strain Rate

For material type **90** (Acoustic)
ss *speed* Sound Speed
b *factor* Damping Factor
cf *flag* Cavitation Flag
where *flag* can be
0 Off
1 On
atmos *pressure* Atmospheric Pressure
grav *constant* Gravitational Acceleration Constant
fsp *x_coord y_coord z_coord* Coordinates of Free Surface point
fsn *x_dir_cos y_dir_cos z_dir_cos* Direction Cosines of Free Surface Normal Vector

For material type **96** (Brittle Damage)
e *young's_modulus* Young's Modulus
frarf *friction* Friction of Reinforcement in Section
erf *young's_modulus* Young's Modulus of Reinforcement
ysrf *yield_stress* Yield Stress of Reinforcement
ehrf *hardening_modulus* Hardening Modulus of Reinforcement
fsrf *failure_strain* True Failure Strain of Reinforcement
pr *poisson's_ratio* Poisson's Ratio
tlimit *tensile_limit* Tensile Limit
slimit *shear limit* Shear Limit
ftough *fracture_toughness* Fracture Toughness
sreten *shear_retention* Shear Retention
visc *viscosity* Viscosity

For material type **100 beam elfom spw** (Spot Weld)
e *young's_modulus* Young's modulus
pr *poisson's_ratio* Poisson's ratio

es *yield_stress* Yield Stress
etan *hardening_modulus* Hardening modulus
tsms *time_step_size* Time step size for mass scaling
fs *failure_strain* Failure strain for eroding elements
nrrf *resultant_at_failure* Force resultant Nrrf at failure
nrsf *resultant_at_failure* Force resultant Nrsf at failure
nrtf *resultant_at_failure* Force resultant Nrtf at failure
mrrf *resultant_at_failure* Moment resultant Mrrf at failure
mssf *resultant_at_failure* Moment resultant Mssf at failure
trrf *resultant_at_failure* Moment resultant Trrf at failure

For material type **103** (Anisotropic Viscoplastic)

e Young's Modulus
pr Poisson's Ratio
sigy Initial Yeild Stress
Choose a Method
flag 0 List all Material Parameters
qr1 *parameter* Isotropic Hardening Paameter, Qr1
cr1 *parameter* Isotropic Hardening Paameter, Cr1
qr2 *parameter* Isotropic Hardening Paameter, Qr2
cr1 *parameter* Isotropic Hardening Paameter, Cr2
qx1 *parameter* Kinematic Hardening Paameter, Qx1
cx1 *parameter* Kinematic Hardening Paameter, Cx1
qx2 *parameter* Kinematic Hardening Paameter, Qx2
cx2 *parameter* Kinematic Hardening Paameter, Cx2
vk *parameter* Viscous Material Parameter Vk
vm *parameter* Viscous Material Parameter Vm
flag 1 Load curve and Optionally the Viscous Parameters
load_curve Load Curve Number
alpha *alpha* Distribution of Hardening Used in the Curve-Fitting
vk *parameter* Viscous Material Parameter Vk
vm *parameter* Viscous Material Parameter Vm
flag 1 Load Table (automatically calculates the Viscous Parameters)
load_table Load Table Number
alpha *alpha* Distribution of Hardening Used in the Curve-Fitting
Choose the Same Element Type
R00 for Shell
R45 for Shell
R90 for Shell
F for Brick
G for Brickh

H for Brick
L for Brick
M for Brick
N for Brick
aopt *option*

where the *option* can be one of

0	by nodes	
1	by point and element center	
2	by normal vectors	
3	by cross product with shell normal(Shell elements only)	
4	by normal vectors in cylindrical coordinates	
xp *x-coordinate*	aopt 1 & 4	
yp *y-coordinate*	aopt 1 & 4	
zp *z-coordinate*	aopt 1 & 4	
ax *x-component*	aopt 2	
ay *y-component*	aopt 2	
az *z-component*	aopt 2	
dx *x-component*	aopt 2	
dy *y-component*	aopt 2	
dz *z-component*	aopt 2	
vx *x-component*	aopt 3 & 4	
vy *y-component*	aopt 3 & 4	
vz *z-component*	aopt 3 & 4	
beta *beta*	aopt 3	Material angle

For material type **126** (Metallic Honeycomb)

e *young's_modulus*	Young's Modulus (required)
pr *poisson's_ratio*	Poisson's Ratio (required)
sigy *yield_stress*	Yield Stress for Fully Compacted Honeycomb (required)
lca *load_curve*	Sigma-aa versus Normal Strain-aa Load Curve (required)
lcb *load_curve*	Sigma-bb versus Normal Strain-bb Load Curve (default LCA)
lcc *load_curve*	Sigma-cc versus Normal Strain-cc Load Curve (default LCA)
lcs *load_curve*	Shear Stress versus Relative Volume/Volumetric Strain Load Curve (default LCA)
vf *relative_volume*	Relative Volume at which the Honeycomb is Fully Compacted (required)
eaau *elastic_modulus*	Elastic Modulus Eaau in Uncompressed Configuration (required)
ebbu *elastic_modulus*	Elastic Modulus Ebbu in Uncompressed Configuration (required)

eccu *elastic_modulus* Elastic Modulus Eccu in Uncompressed Configuration (required)
gabu *elastic_modulus* Elastic Shear Modulus Gabu in Uncompressed Configuration (required)
gbcu *elastic_modulus* Elastic Shear Modulus Gbcu in Uncompressed Configuration (required)
gcau *elastic_modulus* Elastic Shear Modulus Gcau in Uncompressed Configuration (required)
mu *coefficient* Material Viscosity Coefficient (default .05)
bulk *flag* Bulk Viscosity Flag (default 0)

where *flag* can be :

0 No Bulk Viscosity (Recommended)
1 Bulk Viscosity Active

lcab *load_curve* Sigma-ab versus Shear Strain ab Load Curve (default LCS)
lcbc *load_curve* Sigma-bc versus Shear Strain bc Load Curve (default LCS)
lcca *load_curve* Sigma-ca versus Shear Strain ca Load Curve (default LCS)
lcsr *load_curve* Strain Rate Effects Load Curve (optional)
tsef *strain* Tensile Strain at Element Failure (element will erode)
ssef *strain* Shear Strain at Element Failure (element will erode)
aopt *option*

where the *option* can be one of

0 by nodes
1 by point and element center
2 by normal vectors
3 by cross product with shell normal(Shell elements only)

xp *x-coordinate* aopt 1
yp *y-coordinate* aopt 1
zp *z-coordinate* aopt 1
ax *x-component* aopt 2
ay *y-component* aopt 2
az *z-component* aopt 2
dx *x-component* aopt 2
dy *y-component* aopt 2
dz *z-component* aopt 2
vx *x-component* aopt 3
vy *y-component* aopt 3
vz *z-component* aopt 3

For material type **134** (Viscoelastic Fabric)

k Constant Elastic Bulk Modulus
cs Compressive Stress

Viscoelastic Option
Calculate Viscoelastic Constants
lcid2 *load_curve* Bt Least Square Fit Load Curve
bstart *initial_value* Starting Beta
nt *number_of_terms* Number of Terms in Fit
tramp *time* Ramp Time for Loading
List the Viscoelastic Constants
g1 *maxwell_constant* Maxwell Constant G1
beta1 *decay_constant* Decay Constant beta1
g2 *maxwell_constant* Maxwell Constant G2
beta2 *decay_constant* Decay Constant beta2
g3 *maxwell_constant* Maxwell Constant G3
beta3 *decay_constant* Decay Constant beta3
g4 *maxwell_constant* Maxwell Constant G4
beta4 *decay_constant* Decay Constant beta4
g5 *maxwell_constant* Maxwell Constant G5
beta5 *decay_constant* Decay Constant beta5
g6 *maxwell_constant* Maxwell Constant G6
beta6 *decay_constant* Decay Constant beta6
Volumetric Option
Calculate the Volumetric Relaxation
lcidk *load_curve* Volumetric Load Curve for Least Square Fit
bstartk *initial_value* Volumetric Starting Beta
ntk *number_of_terms* Number of Terms in Fit
trampk *time* Volumetric Ramp Time for Loading
List the Volumetric Relaxation Constants
k1 *maxwell_constant* Maxwell Constant K1
betak1 *decay_constant* Decay Constant betak1
k2 *maxwell_constant* Maxwell Constant K2
betak2 *decay_constant* Decay Constant betak2
k3 *maxwell_constant* Maxwell Constant K3
betak3 *decay_constant* Decay Constant betak3
k4 *maxwell_constant* Maxwell Constant K4
betak4 *decay_constant* Decay Constant betak4
k5 *maxwell_constant* Maxwell Constant K5
betak5 *decay_constant* Decay Constant betak5
k6 *maxwell_constant* Maxwell Constant K6
betak6 *decay_constant* Decay Constant betak6

lsdyeos LS-DYNA3D equation of state

lsdyeos *eos_type material_# parameter_list* ;

where can be 1 through 11

For linear polynomial model type 1

c0 *constant*
c1 *coefficient*
c2 *coefficient*
c3 *coefficient*
c4 *coefficient*
c5 *coefficient*
c6 *coefficient*
e0 *energy*
v0 *volume*

For JWL model type 2

a *constant*
b *constant*
r1 *constant*
r2 *constant*
omega *constant*
e0 *energy*
v0 *volume*

For SACK model type 3

a1 *constant*
a2 *constant*
a3 *constant*
b1 *constant*
b2 *constant*
e0 *constant*
v0 *constant*

For GRUNEISEN model type 4

vci *intercept*
s1 *coefficient*
s2 *coefficient*
s3 *coefficient*
gamma *coefficient*
sa *coefficient*
e0 *energy*
v0 *volume*

For ratio of polynomials model type 5

a10 *constant*
a11 *constant*
a12 *constant*
a13 *constant*
a20 *constant*
a21 *constant*
a22 *constant*
a23 *constant*
a30 *constant*
a31 *constant*
a32 *constant*
a33 *constant*
a40 *constant*
a41 *constant*
a42 *constant*
a43 *constant*
a50 *constant*
a51 *constant*
a52 *constant*
a53 *constant*
a60 *constant*
a61 *constant*
a62 *constant*
a63 *constant*
a70 *constant*
a71 *constant*
a72 *constant*
a73 *constant*
alpha *constant*
beta *constant*
a14 *constant*
a24 *constant*
e0 *energy*
v0 *volume*

For linear polynomial with energy deposition model type 6

c0 *constant*
c1 *coefficient*
c2 *coefficient*
c3 *coefficient*
c4 *coefficient*

c5 *coefficient*
c6 *coefficient*
e0 *energy*
v0 *volume*
lc *load_curve*

For ignition and growth of reaction in he model type 7

ap *constant*
bp *constant*
r1p *constant*
r2p *constant*
g *coefficient*
wpcp *constant*
ae *constant*
be *constant*
wece *constant*
r1e *constant*
r2e *constant*
fcrit *fraction*
i *coefficient*
h *coefficient*
z *exponent*
x *exponent*
y *exponent*
cp *heat_capacity*
ce *heat_capacity*
m *exponent*
e0 *energy*
t0 *temperature*

For tabulated with compaction model type 8

eps *list_strains* ;
pc *list_constants* ;
t *list_temperatures* ;
ku *list_modulus* ;
gamma *gamma*
e0 *energy*
v0 *volume*

For tabulated model type 9

eps *list_strains* ;
pc *list_constants* ;

t *list_temperatures* ;
gamma *gamma*
e0 *energy*
v0 *volume*

For propellant model type 10

a *coef*	product JWL coefficient
b *coef*	product JWL coefficient
xp1 *coef*	product JWL coefficient
xp2 *coef*	product JWL coefficient
frer *volume*	unreacted c0-volume
g *product*	product wcv
r1 *coef*	unreacted JWL coefficient
r2 *coef*	unreacted JWL coefficient
r3 *coef*	unreacted wcv
r5 *coef*	unreacted JWL coefficient
r6 *coef*	unreacted JWL coefficient
fmxig *fraction*	initial fraction reacted f0
freq *pressure*	initial pressure p0
grow1 *coef*	first burn rate coefficient
em *exponent*	pressure exponent 1st term
ar1 *exponent*	exponent on F (1st term)
es1 *exponent*	exponent on 1-F (1st term)
cvp *capacity*	heat capacity products
cvr *capacity*	heat capacity unreacted
ccrit *volume*	product co-volume
enq *heat*	heat of reaction
tmp0 *temp*	initial temperature 298 k
grow2 *coef*	second burn rate coefficient
ar2 *exponent*	exponent on F (2nd term)
es2 *exponent*	exponent on 1-F (2nd term)
en *exponent*	pressure exponent 2nd term
fmxgr *max*	maximum F for 1st term
fmngr *min*	minimum F for 2nd term

For pore collapse model type 11

mu1 *compression*
mu2 *intersection*
e0 *energy*
mu0 *compression*
virgin *load_curve_pairs* ;
crushed *load_curve_pairs* ;

lcvir *load_curve_#*
lccru *load_curve_#*

nikemats NIKE3D materials

nikemats *material_# material_type parameter_list* ;

where material type can be 1 through 20, 23, 35 or 0 (experimental)

The following commands are available for all materials:

shell
with the following options
shear *factor*
tsti *#_points*
propt *option*
where option can be
1 for element center
2 for plan integration points
3 for through thickness and plan integration points
quad *integration_rule_#*
where the number can be
n positive for the number of points using the trapezoidal rule
0 for Gauss
-n negative of the user specified rule number (from sind command)
shth *thickness*
shth1 *thickness*
shth2 *thickness*
shth3 *thickness*
shth4 *thickness*
shloc *location*
where location can be
1 for top surface
0 for middle surface
-1 for bottom surface

beam
with the following options
shear *factor*
quad *option*
where the option can be

1 for a truss
2 for 2x2 Gauss quadrature
3 for 3x3 Gauss quadrature
4 for 3x3 Lobatto integration
5 for 4x4 Gauss quadrature

bmcross *shape*

where the shape can be

0 for rectangular
1 for tubular

sthi *thickness*
tthi *thickness*
sthi1 *thickness*
sthi2 *thickness*
tthi1 *thickness*
tthi2 *thickness*
sloc *location*

where location can be

1 meaning the side where s is 1
0 meaning centered
-1 meaning the side where s is -1

tloc *location*

where location can be

1 meaning the side where t is 1
0 meaning centered
-1 meaning the side where t is -1

rho *density*
rda *constant*
rdb *constant*

The remainder is specific to the selected material:

For material type 1

e *modulus*
pr *ratio*

For material type 2

ea *ea*
eb *eb*
ec *ec*
prba *vba*
prca *vca*
prcb *vcb*

gab *gab*
gbc *gbc*
gca *gca*
aopt *option parameters*
where option can be **0**, **1**, or **2** and
where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*

For material type 3
e *modulus*
pr *ratio*
sigy *stress*
etan *modulus*
beta *parameter*
es *strain_list* ;
eps *stress_list* ;

For material type 4
temp *temperature_list* ;
e *modulus_list* ;
pr *ratio_list* ;
alpha *secant_list* ;
sigy *stress_list* ;
etan *modulus_list* ;

For material type 5
g *modulus*
ku *modulus*
a0 *yield*
a1 *yield*
a2 *yield*
pc *pressure*
ul *option*
where option can be
0 volumetric crushing
1 no volumetric crushing
vs *strain_list* ;
p *pressure_list* ;

For material type 6

k *modulus*
g0 *modulus*
gi *modulus*
beta *decay*

For material type 7
ea *modulus*
eb *modulus*
ec *modulus*
prba *ratio*
prca *ratio*
prcb *ratio*
alpa *expansion*
alpb *expansion*
alpc *expansion*
gab *modulus*
gbc *modulus*
gca *modulus*
aopt *option parameters*
where *option* can be **0, 1**, or **2** and
where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*

For material type 8
temp *temperature_list* ;
g *modulus_list* ;
k *modulus_list* ;
alpha *secant_list* ;
a *creep_list* ;
b *creep_list* ;

For material type 9
e *modulus*
pr *ratio*
k *strength*
n *hardening*

For material type 10
temp *temperature*
e *modulus*

pr *ratio*
alpha *expansion*
k *strength*
n *exponent*

For material type 11 pr ratio
n *exponent*
temp *temperature*
e *modulus*
a *stress*
m *exponent*
alpha *coefficient*

For material type 12
gammay *strain*
tauy *stress*
alpha *coefficient*
r *exponent*
k *modulus*

For material type 13
matrix *c11 c12 c13 c14 c15 c16*
c22 c23 c24 c25 c26
c33 c34 c35 c36
c44 c45 c46
c55 c56
c66
alpha1 *expansion*
alpha2 *expansion*
alpha3 *expansion*
aopt *option parameters*
where option can **0**, **1**, or **2** and
where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*

For material type 14
e *young's_modulus*
pr *poisson's_ratio*
ft *tensile_strength*
fs *cracked_shear_strength*

sigy *compressive_yield_strength*
gc *fracture_toughness*
beta *shear_retention_factor*
eta *viscosity*

For material type 15
ai *first_invariant_term*
bi *second_invariant_term*
pr *poisson's_ratio*
aflg *option parameters*
where option can be **0** (off) or **1** (on)
where parameters that follow depend on the option
0
1 *altol* tolerance

For material type 16
lcyt *load_curve* Young's modulus load curve
lypt *load_curve* Poisson's ratio load curve
lyet *load_curve* thermal expansion load curve
fsm *option suboptions*
where suboptions that follow depend on the option
0
1 bulk *modulus* bulk modulus
lcyst *load_curve* yield stress load curve
lcptt *load_curve* plastic tangent load curve
2 bulk *modulus* bulk modulus
lcist *load_curve* initial strength load curve
lcfst *load_curve* flow strength load curve
lcsrt *load_curve* strain rate load curve
lcscbt *load_curve* strengthening coef load curve
lcsent *load_curve* strengthening exp load curve
3 bulk *modulus* bulk modulus
lcist *load_curve* initial strength load curve
lcfst *load_curve* flow strength load curve
lcsrt *load_curve* strain rate load curve
lcsck1t *load_curve* strengthening coef load curve
lcsen1t *load_curve* strengthening exp load curve
lcrck2t *load_curve* recovery coef load curve
lcfren2t *load_curve* first recovery coef load curve
lcsren3t *load_curve* second recovery coef load curve
ffm *option suboptions*
where suboptions that follow depend on the option

0

1 lcfvt *load_curve* fluid viscosity load curve

vgm *option suboptions*

where suboptions that follow depend on the option

0

1 ivs *strain* initial void strain

lctpt *load_curve* tensile pressure load curve

lccpt *load_curve* compressive pressure load curve

cemf *option*

where option can be

0 off

1 on

2 ivs *strain* initial void strain

g1 *parameter* first gurson parameter

g2 *parameter* second gurson parameter

cemf *option*

where option can be

0 off

1 on

lsm *option suboptions*

where suboptions that follow depend on the option

0

1 lcfsft *load_curve* fraction solid load curve

vardb *option*

where option can be

0 effectic plastic strain

1 void strain

2 flow strength

3 effective strain rate

For material type 17

matrix *k11 k12 k13 k14 k15 k16*
k22 k23 k24 k25 k26
k33 k34 k35 k36
k44 k45 k46
k55 k56
k66

For material type 18

c1 *constant*

c2 *constant*

c3 *stress_coef*

c4 *uncrimping_coef*
c5 *fiber_modulus*
k *bulk_modulus*
lambda *stretch*
isf *option*
where the option can be
0 off
1 on
lcis *load_curve* initial stretch load curve
aflg *option*
where the option can be
0 off
1 on
altol *tol* only for aflg=1
aopt *option suboptions*
where option can be **0**, **1**, or **2**
where suboptions that follow depend on the option
0 *node1 node2 node3* nodes of the element
1 *x y z* coordinates of point
2 *ax ay az* vector a
dx dy dz vector d

For material type 19
e *young's_modulus*
pr *poisson's_ratio*
sck *strength_coef*
hen *hardening_exponent*
srsem *strain_rate_exponent*
isr *initial_strain_rate*

For material type 20
e *young's_modulus*
pr *poisson's_ratio*
xtrans *x*
ytrans *y*
ztrans *z*
xrot *x*
yrot *y*
zrot *z*
comflg *option parameters*
where option can be **0** or **1**
where parameters that follow depend on the option

0
1 *xcom ycom zcom* x,y,z coordinates of center of mass

For material type 23

ea *ea_list* ;
eb *eb_list* ;
ec *ec_list* ;
vba *vba_list* ;
vca *vca_list* ;
vcb *vcb_list* ;
aa *aa_list* ;
ab *ab_list* ;
ac *ac_list* ;
gab *gab_list* ;
gbc *gbc_list* ;
gca *gca_list* ;
t *temperature_list* ;
angles *angle_list* ;
aopt *option parameters*
where option can **0**, **1**, **2**, or **3** and
where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*
3 *v1 v2 v3*

For material type 35

e *young's_modulus*
pr *poisson's_ratio*
sg0 *yield_stress*
lcxe *load_curve* tangent modulus load curve
lclh *load_curve* left side load curve
lcrh *load_curve* right side load curve
lcrx *load_curve* pressure load curve
lcedf *load_curve* fld rate load curve
lcedm *load_curve* yield stress load curve
eptr *strain* transient strain
epf *strain* effective strain
scldev *factor* failure scale factor

lsnkmats LS-NIKE3D materials

lsnkmats *material_# material_type parameter_list* ;

where material type can be 1 through 11, 13, or 23

The following commands are available for all materials:

shell

with the following options
shear *factor*
tsti *#_points*
propt *option*

where option can be
1 for element center
2 for plan integration points
3 for through thickness and plan integration points

userrl *integration_rule_#* (from sind command)
quad *integration_rule_#*

where the number can be
n positive for the number of points using the trapezoidal rule
0 for Gauss
-n negative of the user specified rule number (from sind command)

shth *thickness*
shth1 *thickness*
shth2 *thickness*
shth3 *thickness*
shth4 *thickness*
shloc *location*

where location can be
1 for top surface
0 for middle surface
-1 for bottom surface

beam

with the following options
shear *factor*
userrl *integration_rule_#* (from bind or bsd command)
quad *option*

where the option can be
1 for a truss
2 for 2x2 gauss quadrature
3 for 3x3 gauss quadrature

4 for 3x3 lobatto integration
5 for 4x4 gauss quadrature

bmcross *shape*
where the shape can be
0 for rectangular
1 for tubular

sthi *thickness*
tthi *thickness*
sthi1 *thickness*
sthi2 *thickness*
tthi1 *thickness*
tthi2 *thickness*
sloc *location*
where location can be
1 meaning the side where s is 1
0 meaning centered
-1 meaning the side where s is -1

tloc *location*
where location can be
1 meaning the side where t is 1
0 meaning centered
-1 meaning the side where t is -1

rho *density*
rda *constant*
rdb *constant*

The remainder is specific to the selected material:

For material type 1
e *modulus*
pr *ratio*

For material type 2
ea *ea*
eb *eb*
ec *ec*
prba *vba*
prca *vca*
prcb *vcb*
gab *gab*
gbc *gbc*
gca *gca*
aopt *option parameters*

where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*

For material type 3
e *modulus*
pr *ratio*
sigy *stress*
etan *modulus*
beta *parameter*
es *strain_list* ;
eps *stress_list* ;

For material type 4
temp *temperature_list* ;
e *modulus_list* ;
pr *ratio_list* ;
alpha *secant_list* ;
sigy *stress_list* ;
etan *modulus_list* ;

For material type 5
g *modulus*
ku *modulus*
a0 *yield*
a1 *yield*
a2 *yield*
pc *pressure*
ul *option*
where option can be
0 volumeteric crushing
1 no volumeteric crushing
vs *strain_list* ;
p *pressure_list* ;

For material type 6
k *modulus*
g0 *modulus*
gi *modulus*
beta *decay*

For material type 7

ea *modulus*
eb *modulus*
ec *modulus*
prba *ratio*
prca *ratio*
prcb *ratio*
alpa *expansion*
alpb *expansion*
alpc *expansion*
gab *modulus*
gbc *modulus*
gca *modulus*
aopt *option parameters*

where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*

For material type 8

temp *temperature_list* ;
g *modulus_list* ;
k *modulus_list* ;
alpha *secant_list* ;
a *creep_list* ;
b *creep_list* ;

For material type 9

e *modulus*
pr *ratio*
k *strength*
n *hardening*

For material type 10

temp *temperature*
e *modulus*
pr *ratio*
alpha *expansion*
k *strength*
n *exponent*

For material type 11

pr *ratio*

n *exponent*
temp *temperature*
e *modulus*
a *stress*
m *exponent*
alpha *coefficient*

For material type 12
gammay *strain*
tauy *stress*
alpha *coefficient*
r *exponent*
k *modulus*

For material type 13
matrix *c11 c12 c13 c14 c15 c16*
c22 c23 c24 c25 c26
c33 c34 c35 c36
c44 c45 c46
c55 c56
c66
alpha1 *expansion*
alpha2 *expansion*
alpha3 *expansion*
aopt *option parameters*
where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*

For material type 23
ea *ea_list* ;
eb *eb_list* ;
ec *ec_list* ;
vba *vba_list* ;
vca *vca_list* ;
vca *vca_list* ;
vcb *vcb_list* ;
aa *aa_list* ;
ab *ab_list* ;
ac *ac_list* ;
gab *gab_list* ;

gbc *gbc_list* ;
gca *gca_list* ;
t *temperature_list* ;
angles *angle_list* ;
aopt *option parameters*

where the parameters that follow depend on the option
0
1 *xp yp zp*
2 *a1 a2 a3 d1 d2 d3*
3 *v1 v2 v3*

marcmats MARC material models

marcmats *type model options ;*

where a type is followed by only certain models and options. Type can be **stm**, **tm**, **hbm**, **am**, **esm**, **msm**, or **elm**.

stm for stress or coupled thermal-stress analysis has the following material models:

prots *parameters* ; for the property material model. This model can be used with the **strainrate**, **workhard**, **temp**, **tempts**, **damage**, **faildata**, **damping**, and orient options.

a parameter can be any of:
e *young's_modulus*
pr *poisson's_ratio*
rho *density*
alpha *coefficient*
tinit *temperature*
ty *yield*
ab *stress*
rttc *temperature*
rtsh *temperature*
rrho *density*
emiss *emissivity*

isots *parameters* ; for the isotropic material model. This model can be used with the **strainrate**, **workhard**, **cc**, **viscelprop**, **temp**, **tempts**, **damage**, **faildata**, **damping**, and **orient** options.

a parameter can be any of:
e *young's_modulus*
pr *poisson's_ratio*
rho *density*
alpha *coefficient*
ty *yield*
ab *yield*
vm sets flag for von mises yield criteria
lmc sets flag for linear mohr coulomb yield criteria
pmc sets flag for parabolic mohr coulomb yield criteria
bmc sets flag for buyuzkoturk mohr coulomb concrete yield criteria
nornl sets flag for normal ornl yield criteria
cmnornl sets flag for crmo ornl yield criteria
rpornl sets flag for revp ornl yield criteria
aornl sets flag for full alpha reset ornl yield criteria
genplast sets flag for generalized plasticity model yield criteria
vspl sets flag for viscoplastic model thru sub uvscpl yield criteria
ih sets flag for isotropic hardening
kh sets flag for kinematic hardening
ch sets flag for combined hardening
co *conductivity*
sh *heat*
hrho *density*
emiss *emissivity*

isorpts *parameters* ; for the isotropic rigid plastic material model. This model can be used with the **strainrate**, **workhard**, **temp**, **tempts**, **damage**, **faildata**, **damping**, and **orient** options.

a parameter can be any of:
py *penalty*
rho *density*
alpha *coefficient*
ty *yield*
co *conductivity*
sh *heat*
hrho *density*
emiss *emissivity*

orthots *parameters* ; for the orthotropic material model. This model can be used with the **strainrate**, **workhard**, **viscelprop**, **orthotem**, **faildata**, **damping**, and **orient** options.

a parameter can be any of:
ianels sets flag for call user subroutines
e11 *e11-young's_modulus*
e22 *e22-young's_modulus*
e33 *e33-young's_modulus*
v12 *v12-poisson's_ratio*
v23 *v23-poisson's_ratio*
v31 *v31-poisson's_ratio*
rho *density*
g12 *g12-shear_modulus*
g23 *g23-shear_modulus*
g31 *g31-shear_modulus*
a11 *alpha11-coefficient*
a22 *alpha22-coefficient*
a33 *alpha33-coefficient*
ty *stress*
ab *yield*
vm set flag for von mises yield criteria
nornl set flag for normal ornl yield criteria
cmnornl set flag for crmo ornl yield criteria
rpornl set flag for revp ornl yield criteria
aornl set flag for full alpha reset ornl yield criteria
genplast set flag for generalized plasticity model yield criteria
vspl set flag for viscoplastic model thru sub uvscpl yield criteria
ih set flag for isotropic hardening
kh set flag for kinematic hardening
ch set flag for combined hardening
yrdir1 *ratio*
yrdir2 *ratio*
yrdir3 *ratio*
yrshr1 *ratio*
yrshr2 *ratio*
yrshr3 *ratio*
k11 *conductivity*
k22 *conductivity*
k33 *conductivity*
hrho *density*
sh *specific_heat*
r11 *resistivity*
r22 *resistivity*
r33 *resistivity*

vp *viscoplastic_parameters* ;

anisots *parameters* ; for the anisotropic material model. This model can be used with the **strainrate**, **workhard**, **viscelprop**, **orthotem**, **faildata**, **damping**, and **orient** options.

a parameter can be any of:
rho *density*
ty *yield*
ab *yield*
vm set flag for von mises yield criteria
nornl set flag for normal ornl yield criteria
cmnornl set flag for crmo ornl yield criteria
rpornl set flag for revp ornl yield criteria
aornl set flag for full alpha reset ornl yield criteria
ih set flag for isotropic hardening
kh set flag for kinematic hardening
ch set flag for combined hardening
hrho *density*
sh *specific_heat*
anels flags for user subroutine anisotropic data
anisodata flags for input of anisotropic data
c11 *coefficient*
c12 *coefficient*
c13 *coefficient*
c14 *coefficient*
c15 *coefficient*
c16 *coefficient*
c22 *coefficient*
c23 *coefficient*
c24 *coefficient*
c25 *coefficient*
c26 *coefficient*
c33 *coefficient*
c34 *coefficient*
c35 *coefficient*
c36 *coefficient*
c44 *coefficient*
c45 *coefficient*
c46 *coefficient*
c55 *coefficient*
c56 *coefficient*

c66 *coefficient*
a11 *coefficient*
a12 *coefficient*
a13 *coefficient*
a22 *coefficient*
a23 *coefficient*
a33 *coefficient*
yrdir1 *ratio*
yrdir2 *ratio*
yrdir3 *ratio*
yrshr1 *ratio*
yrshr2 *ratio*
yrshr3 *ratio*
k11 *thermal_conductivity*
k12 *thermal_conductivity*
k13 *thermal_conductivity*
k22 *thermal_conductivity*
k23 *thermal_conductivity*
k33 *thermal_conductivity*

hypoelas *parameters* ; for the hypoelastic material model. This model can be used with the **temp**, **tempts**, **faildata**, **damping**, and **orient** options.

a parameter can be any of:
anexp flag to call call user subroutines anexp and orient
rho *density*
alpha *coefficient*
co *conductivity*
sh *specific_heat*
rs *resistivity*
hrho *density*

mooney *parameters* ; for the mooney material model. This model can be used with the **temp**, **tempts**, **faildata**, and **damping** option.

a parameter can be any of:
c10 *constant*
c01 *constant*
rho *density*
alpha *coefficient*
c11 *constant*
c20 *constant*

c30 *constant*
co *conductivity*
sh *specific_heat*
hrho *density*
viscelmo *vmd multiplier.vs.time* ;

ogden *parameters* ; for the ogden material model. This model can be used with the **viscelogden**, **temp**, **tempts**, **faildata**, and damping options.

a parameter can be any of:
bm *modulus*
rho *density*
alpha *coefficient*
co *thermal_conductivity*
sh *specific_heat*
hrho *density*
mpse *modulus.vs.power.vs.power* ;

foam *parameters* ; for the foam material model. This model can be used with the **temp**, **tempts**, **faildata**, and **damping** options.

a parameter can be any of:
rho *density*
alpha *coefficient*
co *thermal_conductivity*
sh *specific_heat*
hrho *density*
mpse *mudulus.vs.power_de.vs.power_vo* ;

timetemp *parameters* ; for the time-temp material model. This model can be used with the **faildata** or **damping** options.

a parameter can be any of:
mintemp *temperature*
maxtemp *temperature*
e *property_curves* ;
where a property curve is:
A a b young's_modulus.vs.temperature ;
pr *property_curves* ;
where a property curve is:
A a b poisson's_ratio.vs.temperature ;
yld *property_curves* ;

where a property curve is:
A a b yield.vs.temperature ;
whr *property_curves* ;
where a property curve is:
A a b hardening.vs.temperature ;
alpha *property_curves* ;
where a property curve is:
A a b coefficient.vs.temperature ;
co *property_curves* ;
where a property curve is:
A a b conductivity.vs.temperature ;
sh *property_curves* ;
where a property curve is:
A a b specific_heat.vs.temperature ;

creep *parameters* ; for the creep material model

where a parameter can be any of:
slopes to flag for slopes curves
scvte *rate.vs.temperature* ;
scvts *rate.vs.stress* ;
scvcs *rate.vs.strain* ;
scvti *rate.vs.time* ;
values to flag for values curves
vcvte *rate.vs.temperature* ;
vcvts *rate.vs.stress* ;
vcvcs *rate.vs.strain* ;
vcvti *rate.vs.time* ;
ncsr *constant*
cstrat *increment*
cstret *increment*
avrf *testing_flag*
crplawf flag for user-supplied creep law crplaw

gap *parameters* ; for the gap data material model

where a parameter can be any of:
gd *gap*
mi *coefficient*
kgap *elastic_stiffness*
kfrict *elastic_stiffness*
mr1 *momentum_ratio*

mr4 *momentum_ratio*
fdg flag for fixed direction gap
tdg flag for true distance gap
oi1 flag for open for increment 1
ci1 flag for closed for increment 1

powder *parameters* ; for the powder material model. This model can be used with the **dens**, **temp**, **tempts**, **faildata**, and **damping** options.

a parameter can be any of:
e *young's_modulus*
pr *poisson's_ratio*
rho *density*
alpha *coefficient*
cys *stress*
gm *gamma*
bt *beta*
vy *viscosity*
co *conductivity*
sh *specific_heat*
hrho *density*
q1 *q1*
q2 *q2*
q3 *q3*
q4 *q4*
b1 *b1*
b2 *b2*
b3 *b3*
b4 *b4*

soil *parameters* ; for the soil material model. This model can be used with the **workhard**, **temp**, **tempts**, **faildata**, and **damping** options.

a parameter can be any of:
linear set flag for linear soil model
nonlinear set flag for nonlinear soil model
camclay set flag for camclay soil model
e *young's_modulus*
pr *poisson's_ratio*
rho *mass_density*
alpha *coefficient*
ys *yield*_strength

bmf *modulus*
dvf *viscosity*
ps *permeability*
vcr *compression_ratio*
rr *recompression_ratio*
scsl *slope*

composite *parameters* ; for the composite material model

where a parameter can be any of:
actual sets flag for actual thickness
percent sets flag for percent thickness
zpos *z-position*
ld *id.vs.thickness.vs.angle* ;

the following are the **stm** options referred to above:

strainrate *parameters* ; for the strain rate option

where a parameter can be any of:
slopes *ysl yield.vs.strain_rate_pairs* ;
values *yvl yield.vs.strain_rate_pairs* ;

workhard *parameters* ; for the work hard option

where a parameter can be any of:
slopes whsl *work_hard.vs.strain_rate* ;
slopes wh10sl *work_hard.vs.strain_rate* ;
values whvl *work_hard.vs.strain_rate* ;
values wh10vl *work_hard.vs.strain_rate* ;

cc *parameters* ; for the concrete crack data option

where a parameter can be any of:
ccs *stress*
mtsm *modulus*
csv *strain*
srf *factor*

viscelprop *parameters* ; for the viscoelastic isotropic option

where a parameter can be any of:

vpd *shear.vs.time*
vpv *bulk.vs.time*
shiftfn *paras* ; for the shift function option
where a para can be any of:
wlfe flags the williams-landel-ferry equation option
pse flags the power series expansion option
us flags the user subroutine option
c1 *c1*
c2 *c2*
rtt *temperature*
psd *coefficients* ;

temp *parameters* ; for the temperature effects (element prop's) option

where a parameter can be any of:
slopes flags for slopes
syldtemp *yield.vs.temperature* ;
smodtemp *young's_modulus.vs.temperature* ;
spoistemp *poisson's_ratio.vs.temperature* ;
stctemp *coeff.vs.temperature* ;
sy10temp *yield10.vs.temperature* ;
swhtemp *ratio.vs.temperature* ;
values flags for values
vyldtemp *yield.vs.temperature* ;
vmodtemp *young's_modulus.vs.temperature* ;
vpoistemp *poisson's_ratio.vs.temperature* ;
vtctemp *coeff.vs.temperature* ;
vy10temp *yield10.vs.temperature* ;
vwhtemp *ratio.vs.temperature* ;

tempts *parameters* ; for the temperature effects (thermal-stress) option

where a parameter can be any of:
slopes to flag for slopes
syldtemp *yield.vs.temperature* ;
smodtemp *young's_modulus.vs.temperature* ;
spoistemp *poisson's_ratio.vs.temperature* ;
stctemp *coeff.vs.temperature* ;
sy10temp *yield10.vs.temperature* ;
swhtemp *ratio.vs.temperature* ;
scotemp *conductivity.vs.temperature* ;
sshtemp *specific_heat.vs.temperature* ;

lhtemp *latent.vs.solidus.vs.liquidus* ;
setemp *emissivity.vs.temperature* ;
values to flag for values
vyldtemp *yield.vs.temperature* ;
vmodtemp *young's_modulus.vs.temperature* ;
vpoistemp *poisson's_ratio.vs.temperature* ;
vtctemp *coeff.vs.temperature* ;
vy10temp *yield10.vs.temperature* ;
vwhtemp *ratio.vs.temperature* ;
vcotemp *conductivity.vs.temperature* ;
vshtemp *specific_heat.vs.temperature* ;
vlhtemp *latent.vs.solidus.vs.liquidus* ;
vetemp *emissivity.vs.temperature* ;

damage *parameters* ; for the damage option

where a parameter can be any of:
nonucle to flag for the no nucleation option
pscn to flag for the plastic strain controlled nucleation option
scn to flag for the stress controlled nucleation option
uvoidn to flag for the user subroutine uvoidn option
edm to flag for the elastomeric damage model option
ueldam to flag for the user subroutine ueldam option
q1 *multiplier*
q2 *multiplier*
ivf *fraction*
fc *fraction*
ff *fraction*
msn *strain*
sdnr *standard_deviation*
vfvnp *fraction*
ddr *rate*
mddf *factor*
vdr *rate*
mvdf *factor*

faildata *parameters* ; for the fail data option

where a parameter can be any of:
pf flags for progressive failure
ffds flags for first fail data set
sfds flags for second fail data set

tfds flags for third fail data set
ufail selects type ufail
mxstress selects type max. stress
mxstrain selects type max. strain
hoffmn selects type hoffman
hill selects type hill
tsaiwu selects type tsai wu
x *stress*
xc *stress*
y *stress*
yc *stress*
z stress
zc *stress*
sxy *shear_stress*
syz *shear_stress*
szx *shear_stress*
ex *tensile_strain*
exc *strain*
ey *tensile_strain*
eyc *strain*
ez tensile_strain
ezc *strain*
gxy *shear_strain*
gyz *shear_strain*
gzx *shear_strain*
sxy *shear_strain*
syz *shear_strain*
szx *shear_strain*
fi *failure_index*
fxy *tensor_constant*
fyz *tensor_constant*
fzx *tensor_constant*

damping *parameters* ; for the damping option

where a parameter can be any of:
modal *fraction*
alpha *multiplier*
beta *multiplier*
gama *multiplier*

orient *parameters* ; for the orientation option

where a parameter can be any of:
edge12 flags edge 1-2 type of orientation angle
edge23 flags edge 2-3 type of orientation angle
edge34 flags edge 3-4 type of orientation angle
edge31 flags edge 3-1 type of orientation angle
edge41 flags edge 4-1 type of orientation angle
xyplane flags xy-plane type of orientation angle
yzplane flags yz-plane type of orientation angle
zxplane flags zx-plane type of orientation angle
xuplane flags xu-plane type of orientation angle
yuplane flags yu-plane type of orientation angle
zuplane flags zu-plane type of orientation angle
uuplane flags uu-plane type of orientation angle
uorient flags type of orientation angle
3daniso flags type of orientation angle
or *angle*
uv1w *w-coordinate_1*
uv1r *r-coordinate_1*
uv1t t-coordinate_1
uv2w *w-coordinate_2*
uv2r *r-coordinate_2*
uv2t *t-coordinate_2*

orthotem *parameters* ; for the temperature effects option

where a parameter can be any of:
slopes to flag for slopes
ve11 *e11.vs.temperature* ;
ve22 *e22.vs.temperature* ;
ve33 *e33.vs.temperature* ;
vv12 *v12.vs.temperature* ;
vv23 *v23.vs.temperature* ;
vv31 *v31.vs.temperature* ;
vg12 *g12.vs.temperature* ;
vg23 *g23.vs.temperature* ;
vg31 *g31.vs.temperature* ;
va11 *a11.vs.temperature* ;
va22 *a22.vs.temperature* ;
va33 *a33.vs.temperature* ;
vwhtemp *hardening.vs.temperature* ;
vk11 *k11.vs.temperature* ;
vk22 *k22.vs.temperature* ;

vk33 *k33.vs.temperature* ;
vshtemp *heat.vs.temperature* ;
lhtemp *latent.vs.solidus.vs.liquidus* ;
values to flag for values
ve11 *e11.vs.temperature* ;
ve22 *e22.vs.temperature* ;
ve33 *e33.vs.temperature* ;
vv12 *v12.vs.temperature* ;
vv23 *v23.vs.temperature* ;
vv31 *v31.vs.temperature* ;
vg12 *g12.vs.temperature* ;
vg23 *g23.vs.temperature* ;
vg31 *g31.vs.temperature* ;
va11 *a11.vs.temperature* ;
va22 *a22.vs.temperature* ;
va33 *a33.vs.temperature* ;
vwhtemp *hardening.vs.temperature* ;
vk11 *k11.vs.temperature* ;
vk22 *k22.vs.temperature* ;
vvk33 *k33.vs.temperature* ;
vshtemp *heat.vs.temperature* ;
vlhtemp *latent.vs.solidus.vs.liquidus* ;

viscelogden *parameters* ; for the viscoelasticity option

where a parameter can be any of:
vode *multiplier.vs.time* ;
vodi *constant.vs.time* ;

dens *parameters* ; for the density effects option

where a parameter can be any of:
slopes sets flag to slopes curves
symrd *young's_modulus.vs.density* ;
sprrd *poisson's_ratio.vs.density* ;
scrd *conductivity.vs.density* ;
sshrd *specific_heat.vs.density* ;
values set flag to values curves
vymrd *young's_modulus.vs.density* ;
vprrd *poisson's_ratio.vs.density* ;
vcrd *conductivity.vs.density* ;
vshrd *specific_heat.vs.density* ;

tm for thermal analysis has the following material models:

proth parameters ; for the property-thermal material model. This model can be used with the **tempth** option.

a parameter can be any of:
rttc *reference* temperature for thermal conductivity
rtsh *reference* temperature for specific heat
rrho *reference* mass density
rr *reference* resistivity
emiss *emissivity* for radiating cavities

isoth parameters ; for the isotropic-thermal material model. This model can be used with the **tempth** option.

a parameter can be any of:
ankorient flag for user subroutines ankond and orient
co *conductivity*
sh *specific_heat*
rho *density*
r *resistivity*
emiss *emissivity*

orthoth parameters ; for the orthotropic-thermal material model. This model can be used with the **orthotem** option.

a parameter can be any of:
ankorient flag for user subroutines ankond and orient
k11 *k11-conductivity*
k22 *k22-conductivity*
k33 *k33-conductivity*
rho *density*
sh *specific_heat*
r11 *r11-resistivity*
r22 *r22-resistivity*
r33 *r33-resistivity*
emiss *emissivity*

anisoth *parameters* ; for the anisotropic-thermal material model. This model can be used with the **orthotem** option.

a parameter can be any of:

ankond flag for user subroutine ankond
rho *density*
sh *specific_heat*
emiss *emissivity*
k11 *k11-conductivity*
k12 *k12-conductivity*
k13 *k13-conductivity*
k22 *k22-conductivity*
k23 *k23-conductivity*
k33 *k33-conductivity*

the following are the **tm** options referred to above:

tempth *parameters* ; for the temperature effects option

where a parameter can be any of:
slopes flags for slopes curves
scotemp *conductivity.vs.temperature* ;
sshtemp *specific_heat.vs.temperature* ;
lhtemp *latent.vs.solidus.vs.liquidus* ;
sretemp *resistivity.vs.temperature* ;
setemp *emissivity.vs.temperature* ;
values flags for values curves
vcotemp *conductivity.vs.temperature* ;
vshtemp *specific_heat.vs.temperature* ;
vlhtemp *latent.vs.solidus.vs.liquidus* ;
vretemp *resistivity.vs.temperature* ;
vetemp *emissivity.vs.temperature* ;

orthotem *parameters* ; for the orthotropic temperature effects option

where a parameter can be any of:
slopes flags for slopes curves
vk11 *k11.vs.temperature* ;
vk22 *k22.vs.temperature* ;
vk33 *k33.vs.temperature* ;
vshtemp *specific_heat.vs.temperature* ;
lhtemp *solidus.vs.liquidus* ;
vr11 *r11.vs.temperature* ;
vr22 *r22.vs.temperature* ;
vr33 *r33.vs.temperature* ;
values flags for values curves

vk11 *k11.vs.temperature* ;
vk22 *k22.vs.temperature* ;
vk33 *k33.vs.temperature* ;
vshtemp *specific_heat.vs.temperature* ;
vlhtemp *ilatent.vs.solidus.vs.liquidus* ;
vr11 *r11.vs.temperature* ;
vr22 *r22.vs.temperature* ;
vr33 *r33.vs.temperature* ;

hbm for hydrodynamic bearing analysis has the following material models:

isobear *parameters* ; for the isotropic-bearing material model. This model can be used with the **tempbear** option.

a parameter can be any of:
rtv *temperature*
cp *pressure*
rmdl *density*

the following are the **hbm** options referred to above:

tempbear *parameters* ; for temperature effects option

where a parameter can be any of:
slopes to flag for slopes curves
svitemp *viscosity.vs.temperature* ;
values to flag for values curves
svitemp *viscosity.vs.temperature* ;

am for acoustic analysis has the following material models:

isoacu *parameters* ; for the isotropic-acoustic material model

where a parameter can be any of:
bm *modulus*
rho *density*

esm for electrostatic analysis has the following material models:

isoelst *parameters* ; for the isotropic-electrostatic material model

where a parameter can be any of:

per *constant*

orthoelst *parameters* ; for the orthotropic-electrostatic material model

where a parameter can be any of:
uepsor flag for use of subroutines ueps and orient
e11 *e11-permitivity*
e22 *e22-permitivity*
e33 *e33-permitivity*

msm for magnetostatic analysis has the following material models:

isomgst *parameters* ; for the isotropic-magnetostatic material model. This model can be used with the bhrel option.

a parameter can be any of:
umorient flag for use of subroutines umu and orient
pab *permeability*
ipc *i-permeability*

orthomgst *parameters* ; for the orthotropic-magnetostatic material model. This model can be used with the **bhrel** option.

a parameter can be any of:
umorient flag for use of subroutines umu and orient
m11 *m11-permeability*
m22 *m22-permeability*
m33 *m33-permeability*
im11 *im11-i-permeability*
im22 *im22-i-permeability*
im33 *im33-i-permeability*

the following are the **msm** options referred to above:

bhrel *parameters* ; for the b-h relation option

where a parameter can be any of:
iso to flag for isotropic curve
hb *h.vs.b*
ortho to flag for orthotropic curves
h1b1 *h1.vs.b1*
h2b2 *h2.vs.b2*

h3b3 *h3.vs.b3*

elm for electromagnetic analysis has the following material models:

isoelmag *parameters* ; for the isotropic-electromagnetic material model. This model can be used with the **bhrel** option.

a parameter can be any of:
pab *permeability*
per *permitivity*
paa *permeability*
sigma *conductivity*

orthoelmag *parameters* ; for the orthotropic-electromagnetic material model. This model can be used with the **bhrel** option.

a parameter can be any of:
uuu flag for use of subroutines umu, ueps and usigma
m11 *m11-permeability*
m22 *m22-permeability*
m33 *m33-permeability*
e11 *e11-permitivity*
e22 *e22-permitivity*
e33 *e33-permitivity*
paa *paa-permeability*
sigma11 *sigma11-conductivity*
sigma22 *sigma22-conductivity*
sigma33 *sigma33-conductivity*

the following are the **elm** options referred to above:

bhrel *parameters* ; for the b-h relation option
where a parameter can be any of:
iso to flag for isotropic curve
hb *h.vs.b*
ortho to flag for orthotropic curves
h1b1 *h1.vs.b1*
h2b2 *h2.vs.b2*
h3b3 *h3.vs.b3*

IV. INDEX