*ELEMENT

The element cards in this section are defined in alphabetical order:

*ELEMENT_SEATBELT_SENSOR

*ELEMENT_SEATBELT_SLIPRING

*ELEMENT_SHELL

*ELEMENT_SHELL_NURBS_PATCH

*ELEMENT_SHELL_SOURCE_SINK

*ELEMENT_SOLID

*ELEMENT_SOLID_NURBS_PATCH

*ELEMENT_SOLID_PERI

*ELEMENT_SOLID_REFERENCE

*ELEMENT_SPH

*ELEMENT_TRIM

*ELEMENT_TSHELL

The ordering of the element cards in the input file is completely arbitrary. An arbitrary number of element blocks can be defined preceded by a keyword control card.

***ELEMENT_BEAM**_*{OPTION}_{OPTION}*

Available options include:

<BLANK>

THICKNESS, SCALAR, SCALR or SECTION

PID

OFFSET

ORIENTATION

WARPAGE

ELBOW (*beta*)

Purpose: Define two node elements including 3D beams, trusses, 2D axisymmetric shells, and 2D plane strain beam elements. The type of the element and its formulation is specified through the part ID (see *PART) and the section ID (see *SECTION_BEAM).

Two alternative methods are available for defining the cross sectional property data. The THICKNESS and SECTION options are provided for the user to override the *SECTION_BEAM data which is taken as the default if the THICKNESS or SECTION option is not used. The SECTION option applies only to resultant beams (ELFORM = 2 on *SECTION_BEAM). End release conditions are imposed using constraint equations, and caution must be used with this option as discussed in Remark 2.

The SCALAR and SCALR options apply only to material model type 146, *MAT_1DOF_GENERALIZED_SPRING.

The PID option is used by the type 9 spot weld element only and is ignored for all other beam types. When the PID option is active, you provide two part IDs that are tied by the spot weld element in an additional card. If the PID option is inactive for the type 9 element, the nodal points of the spot weld are located to the two nearest reference surface segments. In either case, *CONTACT_SPOTWELD must be defined with the spot weld beam part as the tracked surface and the shell parts (including parts PID1 and PID2) as the reference surface. The surface of each segment should project to the other and in the most typical case the node defining the weld, assuming only one node is used, should lie in the middle; however, this is not a requirement. Note that with the spot weld elements only one node is needed to define the weld, and two nodes are optional.

The OFFSET option is not available for discrete beam elements or spotweld beam elements. Neither the OFFSET option nor the ORIENTATION option is available for truss elements or 2D beam element forms 7 or 8.

The ELBOW option is a 3-node beam element with quadratic interpolation that is tailored for the piping industry. It includes 12 degrees of freedom, including 6 ovalization degrees of freedom for describing the ovalization, per node. That is a total of 36 DOFs for each element. An internal pressure can also be given that tries to stiffen the pipe. The pressure, if activated accordingly, can also contribute to the elongation of the pipe. The control node must be given, but it is only used for initially straight elbow elements. For curved elements the curvature center is used as the control node. See *SECTION_BEAM for more information about the physical properties such as pressure and output options.

Card Summary:

Card 1. This card is required,

EID	PID	N1	N2	N3	RT1	RR1	RT2	RR2	LOCAL

Card 2. This card is included if the THICKNESS keyword option is used.

PARM1	PARM2	PARM3	PARM4	PARM5

Card 3. This card is included if the SECTION keyword option is used.

STYPE	D1	D2	D3	D4	D5	D6	

Card 4. This card is included if the SCALAR keyword option is used.

VOL	INER	CID	DOFN1	DOFN2

Card 5. This card is included if the SCALR keyword option is used.

VOL	INER	CID1	CID2	DOFNS

Card 6. This card is included if the PID keyword option is used.

PID1	PID2								

Card 7. This card is included if the OFFSET keyword option is used.

WX1	WY1	WZ1	WX2	WY2	WZ2		

Card 8. This card is included if the ORIENTATION keyword option is used.

VX	VY	VZ					

Card 9. This card is included if the WARPAGE keyword option is used.

SN1	SN2						

Card 10. This card is included if the ELBOW keyword option is used.

MN							

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	RT1	RR1	RT2	RR2	LOCAL
Type	I	I	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	0	0	0	0	2
Remarks					1	2	2	2	2	

VARIABLE	DESCRIPTION
EID	Element ID. A unique ID is generally required, that is, EID must be different from the element IDs also defined under *ELEMENT_DISCRETE and *ELEMENT_SEATBELT. If the field BEAM is set to 1 on the keyword input for *DATABASE_BINARY_D3PLOT, the null beams used for visualization are not created for the latter two types, and the IDs used for the discrete elements and the seatbelt elements can be identical to those defined here.
PID	Part ID; see *PART.
N1	Nodal point (end) 1
N2	Nodal point (end) 2. This node is optional for the spot weld, beam type 9, since if it not defined it will be created automatically and given a non-conflicting nodal point ID. Nodes N1 and N2 are automatically positioned for the spot weld beam element. For the zero length discrete beam elements where one end is attached to ground, set N2 = -N1. For this case, a fully constrained nodal point will be created with a unique ID for node N2.
N3	Nodal point 3 for orientation. The third node, N3, is optional for beam types 3, 6, 7, and 8; if the cross-section is circular, it is optional for beam types 1 and 9. The third node is used for the discrete beam, type 6, if and only if SCOOR is set to 2.0 in the *SECTION_-

VARIABLE	DESCRIPTION
	BEAM input, but even in this case it is optional. An orientation vector can be defined directly by using the option, ORIENTATION. For this case N3 can be defined as zero.
RT1, RT2	Release conditions for translations at nodes N1 and N2, respectively: EQ.0: No translational degrees-of-freedom are released. EQ.1: x-translational degree-of-freedom EQ.2: y-translational degree-of-freedom EQ.3: z-translational degree-of-freedom EQ.4: x and y-translational degrees-of-freedom EQ.5: y and z-translational degrees-of-freedom EQ.6: z and x-translational degrees-of-freedom EQ.7: x, y, and z-translational degrees-of-freedom (3DOF) This option does not apply to the spot weld, beam type 9.
RR1, RR2	Release conditions for rotations at nodes N1 and N2, respectively: EQ.0: No rotational degrees-of-freedom are released. EQ.1: x-rotational degree-of-freedom EQ.2: y-rotational degree-of-freedom EQ.3: z-rotational degree-of-freedom EQ.4: x and y-rotational degrees-of-freedom EQ.5: y and z-rotational degrees-of-freedom EQ.6: z and x-rotational degrees-of-freedom EQ.7: x, y, and z-rotational degrees-of-freedom (3DOF) This option does not apply to the spot weld, beam type 9.
LOCAL	Coordinate system option for release conditions: EQ.1: Global coordinate system EQ.2: Local coordinate system (default)

Thickness Card. Additional card required for THICKNESS keyword option.

Card 2	1	2	3	4	5	6	7	8	9	10
Variable	PARM1		PARM2		PARM3		PARM4		PARM5	
Type	F		F		F		F		F	
Remarks	4		4		4		4		4, 5	

VARIABLE	DESCRIPTION
PARM1	Based on beam type: Type.EQ.1: Beam thickness, *s*-direction at node 1 Type.EQ.2: Area Type.EQ.3: Area Type.EQ.4: Beam thickness, *s*-direction at node 1 Type.EQ.5: Beam thickness, *s*-direction at node 1 Type.EQ.6: Volume, see description for VOL below. Type.EQ.7: Beam thickness, *s*-direction at node 1 Type.EQ.8: Beam thickness, *s*-direction at node 1 Type.EQ.9: Beam thickness, *s*-direction at node 1
PARM2	Based on beam type: Type.EQ.1: Beam thickness, *s*-direction at node 2 Type.EQ.2: I_{ss} Type.EQ.3: Ramp-up time for dynamic relaxation Type.EQ.4: Beam thickness, *s*-direction at node 2 Type.EQ.5: Beam thickness, *s*-direction at node 2 Type.EQ.6: Inertia, see description for INER below. Type.EQ.7: Beam thickness, *s*-direction at node 2 Type.EQ.8: Beam thickness, *s*-direction at node 2 Type.EQ.9: Beam thickness, *s*-direction at node 2
PARM3	Based on beam type:

VARIABLE	DESCRIPTION
	Type.EQ.1: Beam thickness, t-direction at node 1
	Type.EQ.2: I_{tt}
	Type.EQ.3: Initial stress for dynamic relaxation
	Type.EQ.4: Beam thickness, t-direction at node 1
	Type.EQ.5: Beam thickness, t-direction at node 1
	Type.EQ.6: Local coordinate ID
	Type.EQ.7: Not used
	Type.EQ.8: Not used
	Type.EQ.9: Beam thickness, t-direction at node 1
PARM4	Based on beam type:
	Type.EQ.1: Beam thickness, t-direction at node 2
	Type.EQ.2: I_{rr}
	Type.EQ.3: Not used
	Type.EQ.4: Beam thickness, t-direction at node 2
	Type.EQ.5: Beam thickness, t-direction at node 2
	Type.EQ.6: Area
	Type.EQ.7: Not used
	Type.EQ.8: Not used
	Type.EQ.9: Beam thickness, t-direction at node 2
PARM5	Based on beam type:
	Type.EQ.1: Not used
	Type.EQ.2: Shear area
	Type.EQ.3: Not used
	Type.EQ.4: Not used
	Type.EQ.5: Not used
	Type.EQ.6: Offset
	Type.EQ.7: Not used
	Type.EQ.8: Not used
	Type.EQ.9: Print flag to SWFORC file. The default is taken from the *SECTION_BEAM input. To override, set PAR-

VARIABLE	DESCRIPTION
	M5 to 1.0 to suppress printing and to 2.0 to print.

Section Card. Additional card required for SECTION keyword option.

Card 3	1	2	3	4	5	6	7	8
Variable	STYPE	D1	D2	D3	D4	D5	D6	
Type	A	F	F	F	F	F	F	

VARIABLE	DESCRIPTION
STYPE	Section type (A format) of resultant beam, see Figure 41-1:

EQ.SECTION_01: I-Shape	EQ.SECTION_12: Cross
EQ.SECTION_02: Channel	EQ.SECTION_13: H-Shape
EQ.SECTION_03: L-Shape	EQ.SECTION_14: T-Shape 2
EQ.SECTION_04: T-Shape	EQ.SECTION_15: I-Shape 3
EQ.SECTION_05: Tubular box	EQ.SECTION_16: Channel 2
EQ.SECTION_06: Z-Shape	EQ.SECTION_17: Channel 3
EQ.SECTION_07: Trapezoidal	EQ.SECTION_18: T-Shape 3
EQ.SECTION_08: Circular	EQ.SECTION_19: Box-Shape 2
EQ.SECTION_09: Tubular	EQ.SECTION_20: Hexagon
EQ.SECTION_10: I-Shape 2	EQ.SECTION_21: Hat-Shape
EQ.SECTION_11: Solid box	EQ.SECTION_22: Hat-Shape 2

VARIABLE	DESCRIPTION
D1-D6	Input parameters for section option using STYPE above.

Scalar card. Additional card for SCALAR keyword option.

Card 4	1	2	3	4	5	6	7	8	9	10
Variable	VOL		INER		CID		DOFN1		DOFN2	
Type	F		F		F		F		F	

VARIABLE	DESCRIPTION
VOL	Volume of discrete beam and scalar (MAT_146) beam. If the mass

VARIABLE	DESCRIPTION
	density of the material model for the discrete beam is set to unity, the magnitude of the lumped mass can be defined here instead. This lumped mass is partitioned to the two nodes of the beam element. The translational time step size for the type 6 beam is dependent on the volume, mass density, and translational stiffness values, so it is important to define this parameter. Defining the volume is also essential for mass scaling if the type 6 beam controls the time step size.
INER	Mass moment of inertia for the six degrees of freedom discrete beam and scalar (*MAT_146) beam. This lumped inertia is partitioned to the two nodes of the beam element. The rotational time step size for the type 6 beam is dependent on the lumped inertia and the rotational stiffness values, so it is important to define this parameter if the rotational springs are active. Defining the rotational inertia is also essential for mass scaling if the type 6 beam rotational stiffness controls the time step size.
CID	Coordinate system ID for orientation, material type 146; see *DEFINE_COORDINATE_SYSTEM. If CID = 0, a default coordinate system is defined in the global system.
DOFN1	Active degree-of-freedom at node 1, a number between 1 to 6 where 1, 2, and 3 are the x, y, and z-translations and 4, 5, and 6 are the x, y, and z-rotations. This degree of freedom acts in the local system given by CID above. This input applies to material model type 146.
DOFN2	Active degree-of-freedom at node 2, a number between 1 to 6 as described in DOF1 above. This degree-of-freedom acts in the local system given by CID above. This input applies to material model type 146.

Scalar Card (alternative). Additional card for SCALR keyword option.

Card 5	1	2	3	4	5	6	7	8	9	10
Variable	VOL		INER		CID1		CID2		DOFNS	
Type	F		F		F		F		F	

VARIABLE	DESCRIPTION
VOL	Volume of discrete beam and scalar (MAT_146) beam. If the mass density of the material model for the discrete beam is set to unity, the magnitude of the lumped mass can be defined here instead. This lumped mass is partitioned to the two nodes of the beam element. The translational time step size for the type 6 beam is dependent on the volume, mass density, and translational stiffness values, so it is important to define this parameter. Defining the volume is also essential for mass scaling if the type 6 beam controls the time step size.
INER	Mass moment of inertia for the six degrees of freedom discrete beam and scalar (MAT_146) beam. This lumped inertia is partitioned to the two nodes of the beam element. The rotational time step size for the type 6 beam is dependent on the lumped inertia and the rotational stiffness values, so it is important to define this parameter if the rotational springs are active. Defining the rotational inertia is also essential for mass scaling if the type 6 beam rotational stiffness controls the time step size.
CID1	Coordinate system ID at node 1 for orientation for material type 146; see *DEFINE_COORDINATE_SYSTEM. If CID1 = 0, a default coordinate system is defined in the global system.
CID2	Coordinate system ID at node 2 for orientation for material type 146; see *DEFINE_COORDINATE_SYSTEM. If CID2 = 0, a default coordinate system is defined in the global system.
DOFNS	Active degrees-of-freedom at node 1 and node 2. A two-digit number, the first for node 1 and the second for node 2, between 11 and 66 is expected where 1, 2, and 3 are the x, y, and z-translations and 4, 5, and 6 are the x, y, and z-rotations. These degrees-of-freedom acts in the local system given by CID1 and CID2 above. This input applies to material model type 146. For example, if DOFNS = 12, node 1 has an x-translation and node 2 has a y-translation.

Spot Weld Part Card. Additional card for PID keyword option.

Card 6	1	2	3	4	5	6	7	8	9	10
Variable	PID1	PID2								
Type	I	I								

VARIABLE	DESCRIPTION
PID1	Optional part ID for spot weld element type 9
PID2	Optional part ID for spot weld element type 9

Offset Card. Additional card for OFFSET keyword option.

Card 7	1	2	3	4	5	6	7	8
Variable	WX1	WY1	WZ1	WX2	WY2	WZ2		
Type	F	F	F	F	F	F		
Default	0.0	0.0	0.0	0.0	0.0	0.0		

VARIABLE	DESCRIPTION
WX1 - WZ1	Offset vector at nodal point N1. See Remark 7.
WX2 - WZ2	Offset vector at nodal point N2. See Remark 7.

Orientation Card. Additional card for ORIENTATION keyword option.

Card 8	1	2	3	4	5	6	7	8
Variable	VX	VY	VZ					
Type	F	F	F					
Default	0.0	0.0	0.0					

VARIABLE	DESCRIPTION
VX, VY, VZ	Components of an orientation vector relative to node N1. In this case, the orientation vector points to a virtual third node, so the field N3 should be left undefined.

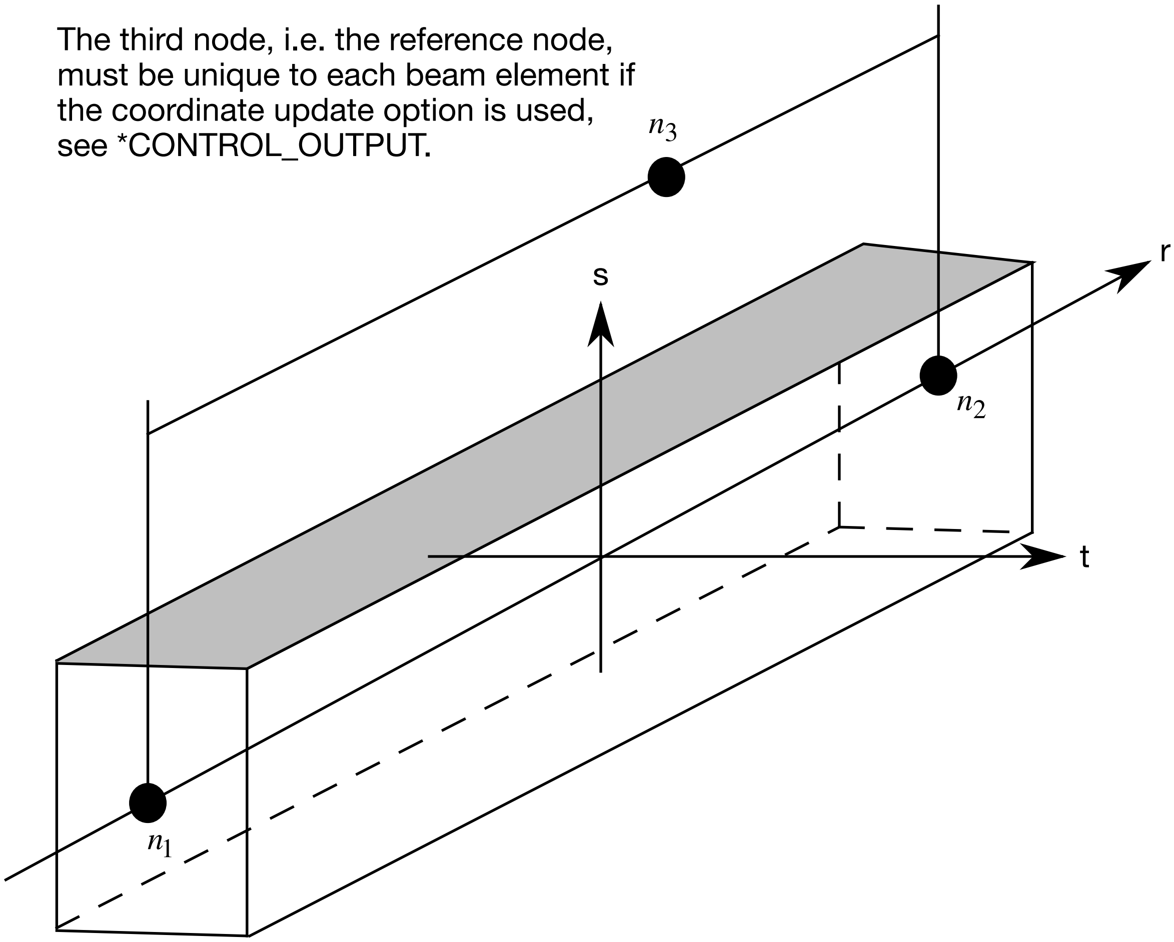


Figure 19-1. LS-DYNA beam elements. Node n_3 determines the initial orientation of the cross section.

Warpage Card. Additional card for WARPAGE keyword option.

Card 9	1	2	3	4	5	6	7	8
Variable	SN1	SN2						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
SN1	Scalar nodal point (end) 1. This node is required.
SN2	Scalar nodal point (end) 2. This node is required.

Elbow Card. Additional card for ELBOW keyword option.

Card 10	1	2	3	4	5	6	7	8
Variable	MN							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
MN	Middle node for the ELBOW element. See Remark 8.

Remarks:

1. **Beam orientation.** A plane through N_1, N_2, and N_3 defines the orientation of the principal *r*-*s* plane of the beam; see Figure 19-1.

2. **Release conditions.** The option to specify release conditions applies to all three-dimensional beam elements. The released degrees of freedom can be either global, or local relative to the local beam coordinate system as shown in Figure 19-1. A local coordinate system is stored for each node of the beam element, and the orientation of the local coordinate systems rotates with the node. To properly track the response, the nodal points with a released resultant are automatically replaced with new nodes to accommodate the added degrees of freedom. Then constraint equations are used to join the nodal points together with the proper release conditions imposed.

   > Nodal points belonging to beam elements that have release conditions applied cannot be subjected to other constraints, such as applied displacement/velocity/acceleration boundary conditions, nodal rigid bodies, nodal constraint sets, or any of the constraint type contact definitions.

   Force-type loading conditions and penalty-based contact algorithms may be used with this option.

   Please note that specification of translational release conditions may lead to nonphysical constraints, but this should not be a problem if the displacements are infinitesimal.

3. **Beam properties.** If the THICKNESS option is not used, or if THICKNESS is used but essential PARMx values are not provided, beam properties are taken from *SECTION_BEAM.

4. **Discrete beam elements.** In the case of the THICKNESS option for type 6, discrete beam elements, PARM1 through PARM5 replace the first five fields on Card 2 of *SECTION_BEAM. Cables are a subset of type 6 beams. PARM1 is for non-cable discrete beams and is optional for cables, PARM2 and PARM3 apply only to non-cable discrete beams, and PARM4 and PARM5 apply only to cables.

5. **PARM5.** In the THICKNESS option, PARM5 applies only to beam types 2, 6 (cables only), and 9.

6. **Stress resultants.** The stress resultants are output in the local coordinate system for the beam. Stress information is optional and is also output in the local system for the beam.

7. **Offsets.** Beam offsets are sometimes necessary for correctly modeling beams that act compositely with other elements such as shells or other beams. When the OFFSET option is specified, global X, Y, and Z components of two offset vectors are given, one vector for each of the two beam nodes. The offset vector extends from the beam node (N1 or N2) to the reference axis of the beam. The beam reference axis lies at the origin of the local s and t-axes. For beam formulations 1 and 11, this origin is halfway between the outermost surfaces of the beam cross-section. Note that for cross-sections that are not doubly symmetric, such as a T-section, the reference axis does not pass through the centroid of the cross-section. For beam formulation 2, the origin is at the centroid of the cross-section.

8. **Elbow.** The Elbow beam is defined with 4 nodes; see Figure 19-2. Nodes n_1 and n_2 are the end nodes, and node n_3 is the middle node. It is custom to set n_3 at the midpoint of the beam. Node n_4 is an orientation node that should be at the curvature center of the beam. If a straight beam is defined initially, the orientation node must be defined and should be on the convex side of the beam. If a curved beam is defined initially, the orientation node is automatically calculated as the center of the beam curvature. However, an orientation node is still required at the input.

   The extra nodes that include the ovalization degree of freedom are written to the messag file during initialization. These extra nodes have 3 DOFs each. That means that there are 2 extra nodes for each physical node. For example, it can look something like this:

```
ELBOW BEAM:           1
n1-n3-n2:             1     3     2
ovalization nodes:    5     7     6
                      8    10     9
```

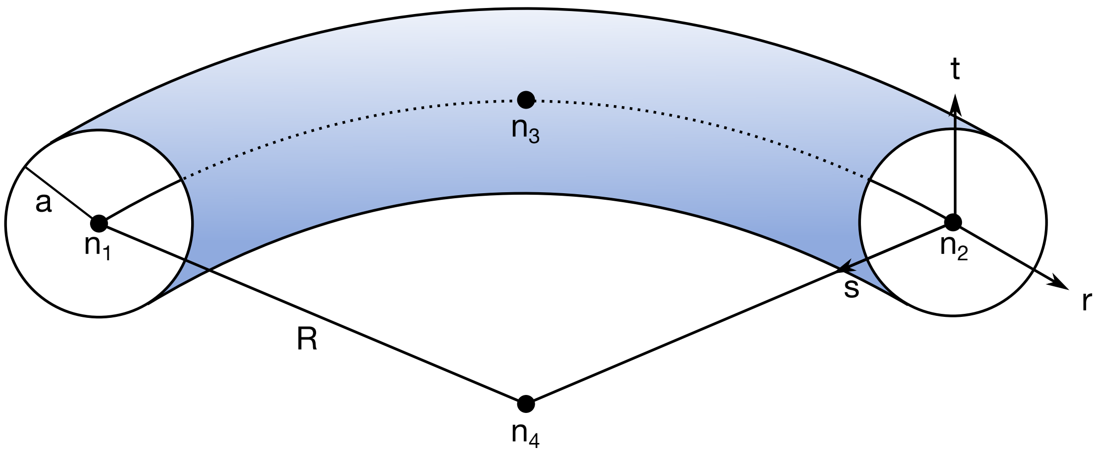


Figure 19-2. Elbow element. Node n_4 is the control node and is given as the beam center of curvature.

And it means that node 1 has the ovalization extra nodes 5 and 8. The first line of ovalization nodes includes the c_1, c_2 and c_3 parameters, and the second line includes the d_1, d_2 and d_3 parameters. That means that node 5 includes c_1, c_2 and c_3, and node 8 includes d_1, d_2 and d_3 for beam node 1. All ovalization dofs can be written to the ASCII file "elbwov" if the correct print flag IOVPR is set to 1 on *SECTION_BEAM. These extra nodes can be constrained as usual nodes. For example, for a cantilever beam that is mounted at node 1, the nodes 1, 5 and 8 should be constrained. The ovalization is approximated with the following trigonometric function:

$$w(r,\theta) = \sum_{k=1}^{3} \sum_{m=1}^{3} h_k(r)\left(c_m^k \cos 2m\theta + d_m^k \sin 2m\theta\right), \qquad -1 \le r \le 1, 0 \le \theta < 2\pi$$

where h_k is the interpolation function at the physical node k.

The Elbow beam only supports tubular cross sections and the pipe outer radius, a, should be smaller than the pipe bend radius, R. That is $a/R \ll 1$. Moreover, the ELBOW beams have 4 stresses: axial *rr*-, shear *rs*-, shear *rt*- and loop-stresses. The loop stress is written at each integration point and can be visualized in LS-PrePost with the user fringe plot file "elbwlp.k". NOTE that the loop-stress is not written to d3plot as default! The NEIPB flag on *DATABASE_EXTENT_BINARY must be set to enable d3plot support.

Right now there is only basic support from the material library. The following materials are currently supported for the ELBOW beam (if requested more materials might be added in the future):

*MAT_ELASTIC (MAT_001)

*MAT_PLASTIC_KINEMATIC (MAT_003)

*MAT_ELASTIC_PLASTIC_THERMAL (MAT_004)

*MAT_VISCOELASTIC (MAT_006)

*MAT_PIECEWISE_LINEAR_PLASTICITY (MAT_024)

*MAT_DAMAGE_3 (MAT_153, explicit only)

*MAT_CONCRETE_BEAM (MAT_195)

*ELEMENT_BEAM_PULLEY

Purpose: Define pulley for beam elements. This feature is implemented for truss beam elements (*SECTION_BEAM, ELFORM = 3) using materials *MAT_001 and *MAT_156 or discrete beam elements (ELFORM = 6) using *MAT_CABLE_DISCRETE_BEAM.

Card 1	1	2	3	4	5	6	7	8
Variable	PUID	BID1	BID2	PNID	FD	FS	LMIN	DC
Type	I	I	I	I	F	F	F	F
Default	none	0	0	0	0.0	0.0	0.0	0.0

VARIABLE	DESCRIPTION
PUID	Pulley ID. A unique number has to be used.
BID1	Truss beam element 1 ID.
BID2	Truss beam element 2 ID.
PNID	Pulley node, NID.
FD	Coulomb dynamic friction coefficient.
FS	Optional Coulomb static friction coefficient.
LMIN	Minimum length, see notes below.
DC	Optional decay constant to allow smooth transition between the static and dynamic friction coefficient, i.e., $$\mu_c = \mathrm{FD} + (\mathrm{FS} - \mathrm{FD})e^{-\mathrm{DC}\times\lvert v_{\mathrm{rel}}\rvert}$$

Remarks:

Elements 1 and 2 should share a node which is coincident with the pulley node. The pulley node should not be on any beam elements.

Pulleys allow continuous sliding of a truss beam element through a sharp change of angle. Two elements (1 & 2 in Figure 19-23 of *ELEMENT_SEATBELT_SLIPRING) meet at the pulley. Node *B* in the beam material remains attached to the pulley node, but beam material (in the form of unstretched length) is passed from element 1 to element 2 to achieve slip. The amount of slip at each time step is calculated from the ratio of forces in elements 1 and 2. The ratio of forces is determined by the relative angle between elements 1 and 2 and the coefficient of friction, FD. The tension in the beams are taken as T_1 and T_2, where T_2 is on the high tension side and T_1 is the force on the low tension side. Thus, if T_2 is sufficiently close to T_1, no slip occurs; otherwise, slip is just sufficient to reduce the ratio T_2/T_1 to $e^{\mathrm{FC}\times\theta}$, where θ is the wrap angle; see Figures 19-24 of *ELEMENT_SEATBELT_SLIPRING. The out-of-balance force at node B is reacted on the pulley node; the motion of node B follows that of pulley node.

If, due to slip through the pulley, the unstretched length of an element becomes less than the minimum length LMIN, the beam is remeshed locally: the short element passes through the pulley and reappears on the other side (see Figure 19-23). The new unstretched length of e_1 is 1.1 × minimum length. The force and strain in e_2 and e_3 are unchanged; while the force and strain in e_1 are now equal to those in e_2. Subsequent slip will pass material from e_3 to e_1. This process can continue with several elements passing in turn through the pulley.

To define a pulley, the user identifies the two beam elements which meet at the pulley, the friction coefficient, and the pulley node. If BID1 and BID2 are defined as 0 (zero), adjacent beam elements are automatically detected. The two elements must have a common node coincident with the pulley node. No attempt should be made to restrain or constrain the common node for its motion will automatically be constrained to follow the pulley node. Typically, the pulley node is part of a structure and, therefore, beam elements should not be connected to this node directly, but any other feature can be attached, including rigid bodies.

*DATABASE_PLLYOUT can be used to write a time history output database pllyout for the pulley which records beam IDs, slip, slip rate, resultant force, and wrap angle.

*ELEMENT_BEAM_SOURCE

Purpose: Define a nodal source for beam elements. This feature is implemented *only* for truss beam elements (*SECTION_BEAM, ELFORM = 3) with material *MAT_001, for discrete beam elements (ELFORM = 6) with material *MAT_071 or for Belytschko-Schwer resultant beam elements (ELFORM = 2) using material *MAT_001 or *MAT_028.

Card 1	1	2	3	4	5	6	7	8
Variable	BSID	BSNID	BSEID	NELE	LFED	FPULL	LMIN	NCUT
Type	I	I	I	I	F	F	F	I
Default	none	none	none	0	0.0	0.0	0.0	0

VARIABLE	DESCRIPTION
BSID	Beam Source ID. A unique number has to be used.
BSNID	Source node ID.
BSEID	Source element ID.
NELE	Number of elements to be pulled out.
LFED	Beam element fed length (typical element initial length).
FPULL	Pull-out force. GT.0: Constant value, LT.0: Load curve ID = |FPULL| which defines pull-out force as a function of time. Either *DEFINE_FUNCTION or *DEFINE_ CURVE_FUNCTION (with argument TIME) can be used.
LMIN	Minimum beam element length, see notes below. One to two tenth of the fed length LFED is usually a good choice.
NCUT	Option to cut the last element from the source node: EQ.0: Last element remains attached to the source node (default). EQ.1: Last element gets cut from the source node along with sudden release of the pull-out force, FPULL.

VARIABLE	DESCRIPTION
	GT.1: Similar to NCUT = 1 but the pull-out force, FPULL, decreases linearly to zero over the last NCUT elements to avoid the sudden release of the pull-out force.

Remarks:

The source node BSNID can be defined by itself or it can be part of another structure. It is free to move in space during the simulation process. Initially, the source node should have the same coordinates as one of the source element (BSEID) nodes, but not having the same ID. If the pre-defined pull-out force FPULL is exceeded in the element next to the source, beam material gets drawn out by increasing the length of the beam without increasing its axial force (equivalent to ideal plastic flow at a given yield force). If more than the pre-defined length LFED is drawn out, a new beam element is generated. A new beam element has an initial undeformed length of 1.1 × LMIN. The maximum number of elements, NELE, times the fed length, LFED, defines the maximum cable length that can be pulled out from the source node.

ELEMENT_BEARING_*OPTION

The available option is:

TITLE

Purpose: Define a bearing between two nodes. See Carney, Howard, Miller, and Benson [2014] for a description of this model.

Title Card. Additional card for title keyword option.

Title	1	2	3	4	5	6	7	8
Variable	TITLE							
Type	A70							
Default	none							
Remarks	1							

Card 1	1	2	3	4	5	6	7	8
Variable	ID	ITYPE	N1	CID1	N2	CID2	NB	
Type	I	I	I	I	I	I	I	
Default	0	0	0	0	0	0	0	

Material Properties Card.

Card 2	1	2	3	4	5	6	7	8
Variable	EBALL	PRBALL	ERACE	PRRACE	STRESL			
Type	F	F	F	F	F			
Default	0.0	0.0	0.0	0.0	0.0			

Geometry Card.

Card 3	1	2	3	4	5	6	7	8
Variable	D	DI	DO	DM				
Type	F	F	F	F				
Default	0.0	0.0	0.0	0.0				

Geometry Card 2.

Card 4	1	2	3	4	5	6	7	8
Variable	AO	BI	BO	PD				
Type	F	F	F	F				
Default	0.0	0.0	0.0	0.0				

Preloading Card.

Card 5	1	2	3	4	5	6	7	8
Variable	IPFLAG	XTRAN	YTRAN	ZTRAN	XROT	YROT		
Type	I	F	F	F	F	F		
Default	00	0.0	0.0	0.0	0.0	0.0		

VARIABLE	DESCRIPTION
ID	Bearing ID
ITYPE	Bearing type: EQ.1: Ball bearing EQ.2: Roller bearing
N1	Node on the centerline of the shaft (the shaft rotates)

VARIABLE	DESCRIPTION
CID1	Coordinate system ID orienting the shaft. The local z-axis defines the axis of rotation. See Remark 2.
N2	Node on the centerline of the bearing (the bearing does not rotate). It should initially coincide with N1.
CID2	Coordinate system ID orienting the bearing. The local z-axis defines the axis of rotation. See Remark 2.
NB	Number of balls or rollers
EBALL	Young's modulus for balls or rollers
PRBALL	Poisson's ratio for balls or rollers
ERACE	Young's modulus for races
PRRACE	Poisson's ratio for races
STRESL	Value of the bearing stress that causes LS-DYNA to print a warning message when it has been reached. If it is 0.0, then no message is printed. See Remark 1.
D	Diameter of balls or rollers
DI	Bore inner diameter
DO	Bore outer diameter
DM	Pitch diameter. If DM is not specified, it is calculated as the average of DI and DO.
A0	Initial contact angle in degrees
BI	Inner groove radius to ball diameter ratio for ball bearings and the roller length for roller bearings
BO	Outer race groove radius to ball diameter ratio. Unused for roller bearings.
PD	Total radial clearance between the ball bearings and races when no load is applied
IPFLAG	Preload flag (see Remark 3): EQ.0: No preload

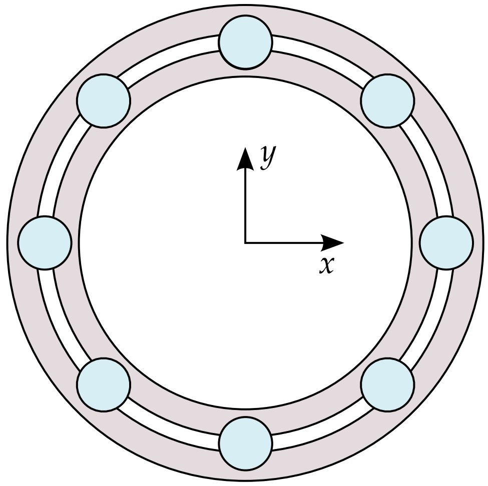


Figure 19-3. Schematic indicating the bearing's local coordinate system determined internally by LS-DYNA. LS-DYNA applies the preload in this coordinate system. The bearing rotates about the local z-axis determined with the right-hand rule.

VARIABLE	DESCRIPTION
	EQ.1: Displacement preload specified EQ.2: Force preload specified
XTRAN	Displacement or force preload in the local x-direction (see Remark 3)
YTRAN	Displacement or force preload in the local y-direction (see Remark 3)
ZTRAN	Displacement or force preload in the local z-direction (see Remark 3)
XROT	Angle (in radians) or moment preload in local x-direction (see Remark 3)
YROT	Angle (in radians) or moment preload in local y-direction (see Remark 3)

Remarks:

1. **Exceeding stress limit parameter.** If the bearing stress limit parameter, STRESL, is exceeded, LS-DYNA writes to the messag and d3hsp files. The element's behavior does not change when this value is surpassed.

2. **Local coordinate systems.** The coordinate systems, CID1 and CID2, must initially be aligned. Define these coordinate systems with *DEFINE_COORDINATE_NODES using FLAG = 1 because their orientations need to be updated.

3. **Preload coordinate system.** When LS-DYNA applies the preload, it applies values input on Card 5 exactly in the internally determined local, bearing coordinate system shown in Figure 19-3. In other words, LS-DYNA does not transform the values from the coordinate system into this system. LS-DYNA orients this local coordinate system such that the bearing rotates about the local z-direction and matches the system manufacturers use. Note that this bearing coordinate is not exactly the same as the coordinate system input with CID2. However, it will be very similar to CID2.

4. **Simulation precision.** We strongly suggest using double precision for solution stability.

5. **Bearing damping.** A realistic level of bearing damping (which can be included using *ELEMENT_DISCRETE and *MAT_DAMPER_VISCOUS) may be needed for solution stability.

6. **Bearing forces.** Bearing forces can be output in the brngout file, using *DATABASE_BEARING. This output includes the effects of the preload.

*ELEMENT_BLANKING

Purpose: Define a part set to be used by keyword *DEFINE_FORMING_BLANKMESH to generate a mesh on a sheet blank for metal forming simulations.

Card 1	1	2	3	4	5	6	7	8
Variable	PSID							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
PSID	Part set ID. See *SET_PART.

Revisions:

This feature is available in LS-DYNA R5 Revision 59165 or later releases.

ELEMENT_DIRECT_MATRIX_INPUT_*{OPTION}

Available options include:

<BLANK>

BINARY

Purpose: Define an element consisting of mass, damping, stiffness, and inertia matrices in a specified file which follows the format used in the direct matrix input, DMIG, of NASTRAN. The supported format is the type 6 symmetric matrix in real double precision. LS-DYNA supports both the standard and the extended precision formats. The binary format from *CONTROL_IMPLICIT_MODES or *CONTROL_IMPLICIT_STATIC_CONDENSATION is another input option. The mass and stiffness matrices are required. The inertia matrix is required when using *LOAD_BODY_OPTION to correctly compute the action of a prescribed base acceleration on the superelement, otherwise the inertia matrix is unused. The damping matrix is optional. The combination of these matrices is referred to as a superelement. Three input cards are required for each superelement.

The degrees-of-freedom for this superelement may consist of generalized coordinates as well as nodal point quantities. Degrees-of-freedom, defined using *NODE input, are called attachment nodes. Only attachment nodes are included in the output to the ASCII and binary databases.

The matrices for a given superelement can be of different order. However, the explicit integration scheme requires the inversion of the union of the element mass matrix and nodal masses associated with attachment nodes. Any degree of freedom included in the other (stiffness, damping, inertia) matrices but without nonzero columns in the combined mass matrix will be viewed as massless and constrained not to move. After deleting zero rows and columns, the combined mass matrix is required to be positive definite.

The inertia matrix is required to have 3 columns which corresponds to the 3 global coordinates. It is used to compute the forces acting on the superelement by multiplying the inertia matrix times the gravitational acceleration specified using *LOAD_BODY_OPTION.

There is no assumption made on the order of the matrices nor the sparse matrix structure of the element matrices, except that they are symmetric and the combined mass matrix is invertible as described above.

Multiple elements may be input using *ELEMENT_DIRECT_MATRIX_INPUT. They may share attachment nodes with other direct matrix input elements. Only *BOUNDARY_PRESCRIBED_MOTION and global constraints imposed using *NODE or *BOUNDARY_SPC on attachment nodes can be applied in explicit applications. Implicit applications can have additional constraints on attachment nodes.

Damping is included via the damping matrix; see the variable DAMP. That damping matrix may be computed by including *DAMPING_GLOBAL or *DAMPING_PART_-MASS when constructing the elemental matrices; see SE_DAMP in *CONTROL_IMPLICIT_MODES.

Card 1	1	2	3	4	5	6	7	8
Variable	EID	IFRMT						
Type	I	I						

Card 2	1	2	3	4	5	6	7	8
Variable	FILENAME							
Type	A80							

Card 3	1	2	3	4	5	6	7	8
Variable	MASS	DAMP	STIF	INERT				
Type	A10	A10	A10	A10				

VARIABLE	DESCRIPTION
EID	Super element ID.
IFRMT	Format: EQ.0: standard format NE.0: extended precision format
FILENAME	Name of file that has the element matrices.
MASS	Name of mass matrix in the file defined by FILENAME. This filename should be no more than eight characters to be compatible with NASTRAN.

VARIABLE	DESCRIPTION
DAMP	Name of damping matrix in the file defined by FILENAME. This filename should be no more than eight characters to be compatible with NASTRAN.
STIF	Name of stiffness matrix in the file defined by FILENAME. This filename should be no more than eight characters to be compatible with NASTRAN.
INERT	Name of inertia matrix in the file defined by FILENAME. This filename should be no more than eight characters to be compatible with NASTRAN. This file must be present when *LOAD_BODY is used to put gravitational forces on the model.

ELEMENT_DISCRETE_*{OPTION}

Available options include:

<BLANK>

LCO

Purpose: Define a discrete (spring or damper) element between two nodes or a node and ground.

An option, LCO, is available for using a load curve(s) to initialize the offset to avoid the excitation of numerical noise that can sometimes result with an instantaneous imposition of the offset. This can be done using a single curve at the start of the calculation or two curves where the second is used during dynamic relaxation prior to beginning the transient part. In the latter case, the first curve would simply specify the offset as constant during time. If the LCO option is active, a second card is read.

Beam type 6, see *ELEMENT_BEAM and SECTION_BEAM, may be used as an alternative to *ELEMENT_DISCRETE and *SECTION_DISCRETE, and is recommended if the discrete element's line of action is not node N1 to N2, i.e., if VID ≠ 0.

> **NOTE:** The discrete elements enter into the time step calculations. Care must be taken to ensure that the nodal masses connected by the springs and dampers are defined and unrealistically high stiffness and damping values must be avoided. **All rotations are in radians**.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	VID	S		PF	OFFSET	
Type	I	I	I	I	I	F		I	F	
Default	none	none	none	none	0	1.0		0	0.0	

Offset Load Curve Card. Additional card for LCO keyword option.

Card 2	1	2	3	4	5	6	7	8
Variable	LCID	LCIDDR						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
EID	Element ID. A unique number is required. Since null beams are created for visualization, this element ID should not be identical to element ID's defined for *ELEMENT_BEAM and *ELEMENT_-SEATBELT.
PID	Part ID, see *PART.
N1	Nodal point 1.
N2	Nodal point 2. If zero, the spring/damper connects node N1 to ground.
VID	Orientation option. The orientation option should be used cautiously since forces, which are generated as the nodal points displace, are not orthogonal to rigid body rotation unless the nodes are coincident. The type 6, 3D beam element, is recommended when orientation is required with the absolute value of the parameter SCOOR set to 2 or 3, since this option avoids rotational constraints. EQ.0: the spring/damper acts along the axis from node N1 to N2, NE.0: the spring/damper acts along the axis defined by the orientation vector, VID defined in the *DEFINE_SD_ORIENTATION section.
S	Scale factor on forces.
PF	Print flag: EQ.0: forces are printed in DEFORC file, EQ.1: forces are not printed DEFORC file.

VARIABLE	DESCRIPTION
OFFSET	Initial offset. The initial offset is a displacement or rotation at time zero. For example, a positive offset on a translational spring will lead to a tensile force being developed at time zero. Ignore this input if LCID is defined below.
LCID	Load curve ID defining the initial OFFSET as a function of time. Positive offsets correspond to tensile forces, and, likewise negative offsets result in compressive forces.
LCIDDR	Load curve ID defining OFFSET as a function of time during the dynamic relaxation phase.

ELEMENT_DISCRETE_SPHERE_*{OPTION}

Available options include:

<BLANK>

VOLUME

Purpose: Define a discrete spherical element for discrete element method (DEM) calculations. Currently, LS-DYNA's implementation of the DEM supports only spherical particles, as discrete element spheres (DES). Each DES consists of a single node with its mass, mass moment of inertia, and radius defined by the input below. Initial coordinates and velocities are specified using the nodal data. The element ID corresponds to the ID of the node. The discrete spherical elements are visualized in LS-PrePost using the same options as the SPH elements.

Please note that the DES part requires *PART, *SECTION, and *MAT keywords. The element type and formulation values in *SECTION are ignored. No *SECTION keyword specific to DES exists, so any type of *SECTION keyword can be used. *SECTION_SOLID is the most commonly used. DEM retrieves the bulk modulus from the *MAT input for coupling stiffness and time step size evaluation, and density from the *MAT input if VOLUME is used to calculate the proper mass. *MAT_ELASTIC and *MAT_RIGID are most commonly used, but other material models are also permissible.

Card Summary:

Card 1a. Include this card if not using the keyword option (<BLANK>).

NID	PID	MASS	INERTIA	RADIUS		IDIST	NID2

Card 1b. Include this card if using the VOLUME keyword option.

NID	PID	VOLUME	INERTIA	RADIUS		IDIST	NID2

Card 2a. Include this card for IDIST = -1 or 1.

		MSD	ISD	RSD			

Card 2b. Include this card for IDIST = -2 or 2.

		MM	MI	MR			

Data Card Definitions:

Card 1a	1	2	3	4	5	6	7	8
Variable	NID	PID	MASS	INERTIA	RADIUS		IDIST	NID2
Type	I	I	F	F	F		I	I
Default	none	none	none	none	none		0	0

VARIABLE	DESCRIPTION
NID	DES Node ID
PID	DES Part ID (see *PART)
MASS	Mass. For IDIST = -1 and 1, this value is the mean. For IDIST = -2 and 2 with MM ≠ 0, this value is the scale parameter.
INERTIA	Mass moment of inertia. For IDIST = -1 and 1, this value is the mean. For IDIST = -2 and 2 with MI ≠ 0, this value is the scale parameter.
RADIUS	Particle radius. Determining contact between particles requires the particle radius. For IDIST = -1 and 1, this value is the mean. For IDIST = -2 and 2 with MR ≠ 0, this value is the scale parameter.
IDIST	Distribution of DES properties (see Remarks 1 and 2): EQ.-2: Weibull distribution (non-deterministic) EQ.-1: Gaussian distribution (non-deterministic) EQ.0: Single property (default) EQ.1: Gaussian distribution (deterministic) EQ.2: Weibull distribution (deterministic)
NID2	Define more than one element with the same PID by setting this field without requiring additional cards. For IDIST = 0, these elements share equal MASS, INERTIA, and RADIUS. For IDIST ≠ 0, these elements each have their MASS, INERTIA, and/or RADIUS randomly distributed according to the distribution and Card 2 parameters assigned. If set, NID2 is a node ID that must have a value greater than NID. Then, LS-DYNA associates a DES to each node

VARIABLE	DESCRIPTION
	with an ID between NID and NID2 (including NID and NID2). If 0 or left blank, then only NID has a DES associated with it.

Card 1b	1	2	3	4	5	6	7	8
Variable	NID	PID	VOLUME	INERTIA	RADIUS		IDIST	NID2
Type	I	I	F	F	F		I	I
Default	none	none	none	none	none		0	0

VARIABLE	DESCRIPTION
NID	DES Node ID.
PID	DES Part ID, see *PART.
VOLUME	Volume. The mass is calculated from material density, $$M = \text{VOLUME} \times \rho_{\text{mat}} \ .$$ For IDIST = -1 and 1, this value is the mean. For IDIST = -2 and 2 with MM ≠ `0, this value is the scale parameter.
INERTIA	Inertia per unit density. The actual inertia is calculated from material density, $$I = \text{INERTIA} \times \rho_{\text{mat}} \ .$$ For IDIST = -1 and 1, this value is the mean. For IDIST = -2 and 2 with MI ≠ 0, this value is the scale parameter.
RADIUS	Particle radius. Determining contact between particles requires the particle radius. For IDIST = -1 and 1, this value is the mean. For IDIST = -2 and 2 with MR ≠ 0, this value is the scale parameter.
IDIST	Distribution of DES properties (see Remarks 1 and 2): EQ.-2: Weibull distribution (non-deterministic) EQ.-1: Gaussian distribution (non-deterministic) EQ.0: Single property (default) EQ.1: Gaussian distribution (deterministic) EQ.2: Weibull distribution (deterministic)

VARIABLE	DESCRIPTION
NID2	Define more than one element with the same PID by setting this field without requiring additional cards. For IDIST = 0, these elements share equal VOLUME, INERTIA, and RADIUS. For IDIST ≠ 0, these elements each have their VOLUME, INERTIA, and/or RADIUS randomly distributed according to the distribution and Card 2 parameters assigned. If set, NID2 is a node ID that must have a value greater than NID. Then, LS-DYNA associates a DES to each node with an ID between NID and NID2 (including NID and NID2). If 0 or left blank, then only NID has a DES associated with it.

Standard Deviation for Gaussian Distribution Card. Include this card if IDIST = -1 or 1.

Card 2a	1	2	3	4	5	6	7	8
Variable			MSD	ISD	RSD			
Type			F	F	F			
Default			0.0	0.0	0.0			

VARIABLE	DESCRIPTION
MSD	Standard deviation for mass or volume (if using the VOLUME keyword option). The mass or volume on Card 1 is the mean. EQ.0.0: Uniform distribution
ISD	Standard deviation for mass moment of inertia or inertia per unit density (if using the VOLUME keyword option). The inertia (or inertia per unit density) on Card 1 is the mean. EQ.0.0: Uniform distribution
RSD	Standard deviation for radius. The radius on Card 1 is the mean. EQ.0.0: Uniform distribution

Shape Parameters for Weibull Distribution Card. Include this card if IDIST = -2 or 2.

Card 2b	1	2	3	4	5	6	7	8
Variable			MM	MI	MR			
Type			F	F	F			
Default			0.0	0.0	0.0			

VARIABLE	DESCRIPTION
MM	Shape parameter (heterogeneity index) for mass or volume (if using the VOLUME keyword option). The mass (or volume) on Card 1 is the associated scale parameter. EQ.0.0: No distribution, MASS / VOLUME sets the value of mass / volume.
MI	Shape parameter (heterogeneity index) for mass moment of inertia or inertia per unit density (if using the VOLUME keyword option). The inertia (or inertia per unit density) on Card 1 is the associated scale parameter. EQ.0.0: No distribution. INERTIA sets the value of inertia or inertia per unit density.
MR	Shape parameter (heterogeneity index) for radius. The radius on Card 1 is the associated scale parameter. EQ.0.0: No distribution. RADIUS sets the value of the radius.

Remarks:

1. **Distributions of DES properties.** IDIST determines the distribution of DES properties for the DES. It specifically affects the mass (or volume), inertia, and radius. IDIST = 0 means all the DES have the same properties.

   The distribution of DES properties for IDIST = -1 and 1 follows a Gaussian distribution:

$$f(x|\text{prop\_mean},\text{prop\_sd}) = \frac{1}{\text{prop\_sd}\sqrt{2\pi}}e^{\frac{(x-\text{prop\_mean})^2}{2\times\text{prop\_sd}^2}} ,$$

where prop_mean is the mean value of the property, prop_sd is the standard deviation, and x is the statistical distribution of the property. For this distribution, you input the following pairs to specify the mean and standard deviation for the three DES properties: (MASS or VOLUME, MSD), (INERTIA, ISD), and (RADIUS, RSD).

The distribution of DES properties for IDIST = -2 and 2 follows a Weibull distribution:

$$f(n|\text{prop\_sf}, \text{mprop}) = \frac{\text{mprop}}{\text{prop\_sf}}\left(\frac{n}{\text{prop\_sf}}\right)^{\text{mprop}-1} e^{-\left(\frac{n}{\text{prop\_sf}}\right)^{\text{mprop}}} ,$$

where prop_sf is the scale factor for the property, mprop is the shape factor for the property, and n is the statistical distribution of the property. For this distribution, you input the following pairs to specify the scale factor and shape factor for the three DES properties: (MASS or VOLUME, MM), (INERTIA, MI), and (RADIUS, MR).

2. **Deterministic versus non-deterministic distributions.** We have implemented both deterministic and non-deterministic versions for the distributions of DES properties (IDIST ≠ 0). By deterministic, we mean that each run uses the same seed and has the same sequence of numbers that fit the distribution. Thus, multiple runs with the same input deck yield the same results. In contrast, the non-deterministic algorithms obtain a random seed each time. Therefore, each run with the same input deck may have a different solution as the sequence of numbers changes each time.

*ELEMENT_GENERALIZED_SHELL

Purpose: Define a general 3D shell element with an arbitrary number of nodes. The formulation of this element is specified in *DEFINE_ELEMENT_GENERALIZED_-SHELL, which is specified through the part ID (see *PART) and the section ID (see *SECTION_SHELL). For an illustration of this referencing, see Figure 19-4. This generalized shell implementation allows for rapid prototyping of new shell element formulations without requiring further coding.

The element formulation used in *SECTION_SHELL needs to be greater or equal than 1000.

Card 1	1	2	3	4	5	6	7	8
Variable	EID	PID	NMNP					
Type	I	I	I					
Default	none	none	none					

Connectivity Cards. Define the connectivity of the element by specifying NMNP-nodes (up to eight nodes per card). Include as many cards as needed. For example, for NMNP = 10, the deck should include two additional cards.

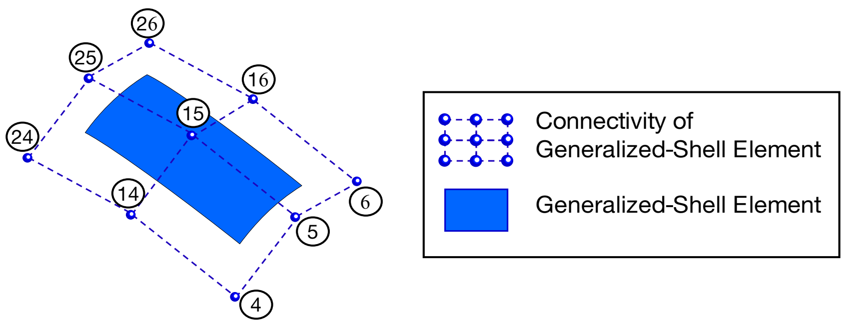
Card 2	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
EID	Element ID. Chose a unique number with respect to other elements.
PID	Part ID, see *PART.
NMNP	Number of nodes to define this element.
N_i	Nodal point *i* (defined via *NODE) to define connectivity of this element.

Remarks:

1. **Interpolation Shell Elements.** For post-processing and the treatment of contact boundary conditions, the use of interpolation shell elements (see *ELEMENT_-INTERPOLATION_SHELL and *CONSTRAINED_NODE_INTERPOLATION) is necessary.

2. **Connectivity.** The definition of the connectivity of the element is basically arbitrary, but it has to correlate with the definition of the element formulation in *DEFINE_ELEMENT_GENERALIZED_SHELL.




```
*ELEMENT_GENERALIZED_SHELL
$---+--EID----+--PID----+-NMNP----+----4----+----5----+----6----+----7----+----8
         1        11         9
$---+---N1----+---N2----+---N3----+---N4----+---N5----+---N6----+---N7----+---N8
        26        25        24        16        15        14         6         5
$---+---N9----+--Etc----+--Etc----+--Etc----+--Etc----+--Etc----+--Etc----+--Etc
         4

*PART
Part for generalized shell
$---+--PID----+SECID----+--MID----+----4----+----5----+----6----+----7----+----8
         11        15         3

*SECTION_SHELL
$---+SECID----ELFORM----+-SHRF----+--NIP----+----5----+----6----+----7----+----8
        15      1001                   2
$---+---T1----+---T2----+---T3----+---T4----+----5----+----6----+----7----+----8
       1.0

*DEFINE_ELEMENT_GENERALIZED_SHELL
$---ELFORM---+---NGP----+-NMNP----+IMASS----+-FORM----+----6----+----7----+----8
      1001         4         9         0         1
...
```

Figure 19-4. Example of the connection between *ELEMENT_GENERALIZED_-SHELL and *DEFINE_ELEMENT_GENERALIZED_SHELL.

*ELEMENT_GENERALIZED_SOLID

Purpose: Define a general 3D solid element with an arbitrary number of nodes. The formulation of this element is specified in *DEFINE_ELEMENT_GENERALIZED_SOLID, which is referenced through the part ID (see *PART) and the section ID (see *SECTION_SOLID). For an illustration of this referencing, see Figure 19-5. This generalized solid implementation allows for rapid prototyping of new solid element formulations without further coding.

The element formulation used in *SECTION_SOLID needs to be greater or equal than 1000.

Card 1	1	2	3	4	5	6	7	8
Variable	EID	PID	NMNP					
Type	I	I	I					
Default	none	none	none					

Connectivity Cards. Define the connectivity of the element by specifying NMNP nodes (up to eight nodes per card). Include as many cards as needed. For example, for NMNP = 10, the deck should include two additional cards.

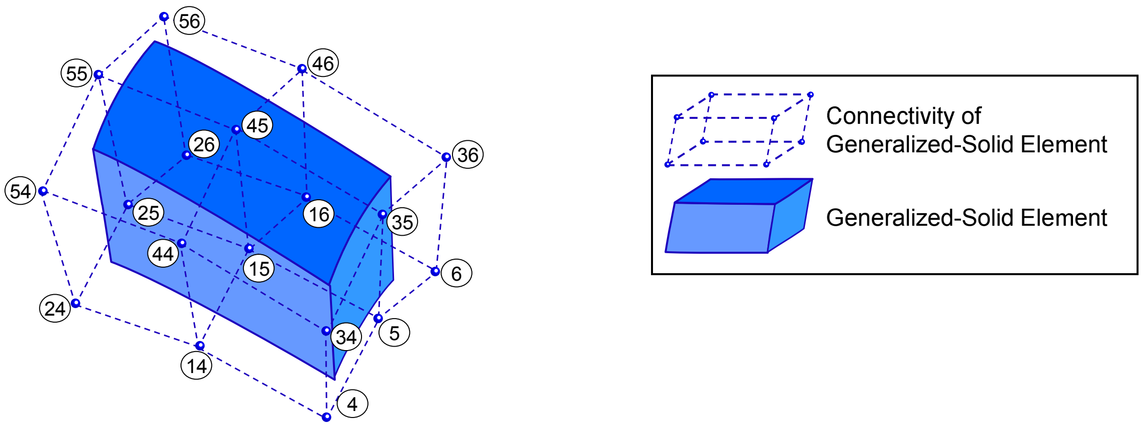
Card 2	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
EID	Element ID. Chose a unique number with respect to other elements.
PID	Part ID, see *PART.
NMNP	Number of nodes to define this element.
N_i	Nodal point i (defined via *NODE) to define connectivity of this element.

Remarks:

1. **Post-Processing.** For post-processing the use of interpolation solid elements (see *ELEMENT_INTERPOLATION_SOLID and *CONSTRAINED_NODE_INTERPOLATION) is necessary.

2. **Connectivity.** The definition of the connectivity of the element is basically arbitrary, but it must be in correlation with the definition of the element formulation in *DEFINE_ELEMENT_GENERALIZED_SOLID.




```
*ELEMENT_GENERALIZED_SOLID
$---+--EID----+--PID----+-NMNP----+----4----+----5----+----6----+----7----+----8
       1        11        18
$---+---N1----+---N2----+---N3----+---N4----+---N5----+---N6----+---N7----+---N8
      56        55        54        46        45        44        36        35
$---+---N9----+--N10----+--N11----+--N12----+--N13----+--N14----+--N15----+--N16
      34        26        25        24        16        15        14         6
$---+--N17----+--N18----+--Etc----+--Etc----+--Etc----+--Etc----+--Etc----+--Etc
       5         4

*PART
Part for generalized solid
$---+--PID----+SECID----+--MID----+----4----+----5----+----6----+----7----+----8
      11        15         3
*SECTION_SOLID
$---+SECID----ELFORM----+--AET----+----4----+----5----+----6----+----7----+----8
      15      1001                   2
*DEFINE_ELEMENT_GENERALIZED_SOLID
$---ELFORM---+---NGP----+-NMNP----+IMASS----+----5----+----6----+----7----+----8
     1001        8        18         0
```

Figure 19-5. Example of the connection between *ELEMENT_GENERALIZED_SOLID and *DEFINE_ELEMENT_GENERALIZED_SOLID.

*ELEMENT_INERTIA_{OPTION}

Available options include:

<BLANK>

OFFSET

Purpose: Define a lumped inertia and, optionally, a lumped mass. This inertia/mass may be located at a nodal point or may be offset from a nodal point. The nodal point can belong to either a deformable body or a rigid body.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	NID	CSID							
Type	I	I	I							
Default	none	none	global							

Card 2	1	2	3	4	5	6	7	8
Variable	IXX	IXY	IXZ	IYY	IYZ	IZZ	MASS	
Type	F	F	F	F	F	F	F	
Default	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

Offset Card. Additional card for offset keyword option.

Card 3	1	2	3	4	5	6	7	8
Variable	X-OFF	Y-OFF	Z-OFF					
Type	F	F	F					
Default	0.	0.	0.					

VARIABLE	DESCRIPTION
EID	Element ID. A unique number must be used.
NID	Node ID. Node to which the mass and inertia is assigned.
CSID	Coordinate system ID EQ.0: Components of the inertia tensor are input with orientation in the global coordinate system. GE.1: Principal moments of inertias are input with orientation of the principal axes defined by coordinate system CSID. See *DEFINE_COORDINATE_*OPTION*. See Remark 2.
IXX	*xx* component of inertia tensor
IXY	*xy* component of inertia tensor
IXZ	*xz* component of inertia tensor
IYY	*yy* component of inertia tensor
IYZ	*yz* component of inertia tensor
IZZ	*zz* component of inertia tensor
MASS	Lumped mass (optional)
X-OFF	*x*-offset from nodal point
Y-OFF	*y*-offset from nodal point
Z-OFF	*z*-offset from nodal point

Remarks:

1. **Inertia and Mass Contribution.** The inertia and mass specified by this keyword is added to the inertia and mass contributed by the elements.

2. **Principal Moments of Inertia.** If CSID is defined, then IXY, IXZ and IYZ are set to zero. Because its inverse is required, the nodal inertia tensor must be positive definite, meaning its determinant must be greater than zero. This check is done after the nodal inertia is added to the defined inertia tensor.

3. **Inertia Tensor for Deformable Bodies.** Nodes of deformable bodies do not support a full inertia tensor. For these bodies, only one value is needed. Specified off-diagonal terms will be ignored. If there is variation among terms on the

diagonal, the largest value will be used. Therefore, an easy way to input inertia for a deformable body node is to define only IXX.

*ELEMENT_INTERPOLATION_SHELL

Purpose: With the definition of interpolation shells, the stresses and other solution variables can be interpolated from the generalized shell elements (see *ELEMENT_GENERALIZED_SHELL and *DEFINE_ELEMENT_GENERALIZED_SHELL) permitting the solution to be visualized using standard 4-node shell elements with one integration point (one value of each solution variable per interpolation shell). The definition of the interpolation shells is based on interpolation nodes (see *CONSTRAINED_NODE_INTERPOLATION). The connections between these various keywords are illustrated in Figure 19-6.

Card 1	1	2	3	4	5	6	7	8
Variable	EIDS	EIDGS	NGP					
Type	I	I	I					
Default	none	none	none					

Weighting Factor Cards. These cards set the weighting factors used for interpolating the solution onto the center of *this* interpolation shell. Set one weight for each of the NGP integration points. Each card can accommodate 4 points; define as many cards as necessary. For example, if NGP = 10, three cards are required.

Card 2	1	2	3	4	5	6	7	8
Variable	IP1	W1	IP2	W2	IP3	W3	IP4	W4
Type	I	F	I	F	I	F	I	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
EIDS	Element ID of the interpolation shell. This needs to coincide with a proper definition of a 4-node shell element (*ELEMENT_SHELL) using interpolation nodes (*CONSTRAINED_NODE_INTERPOLATION).
EIDGS	Element ID of the master element defined in *ELEMENT_GENERALIZED_SHELL.

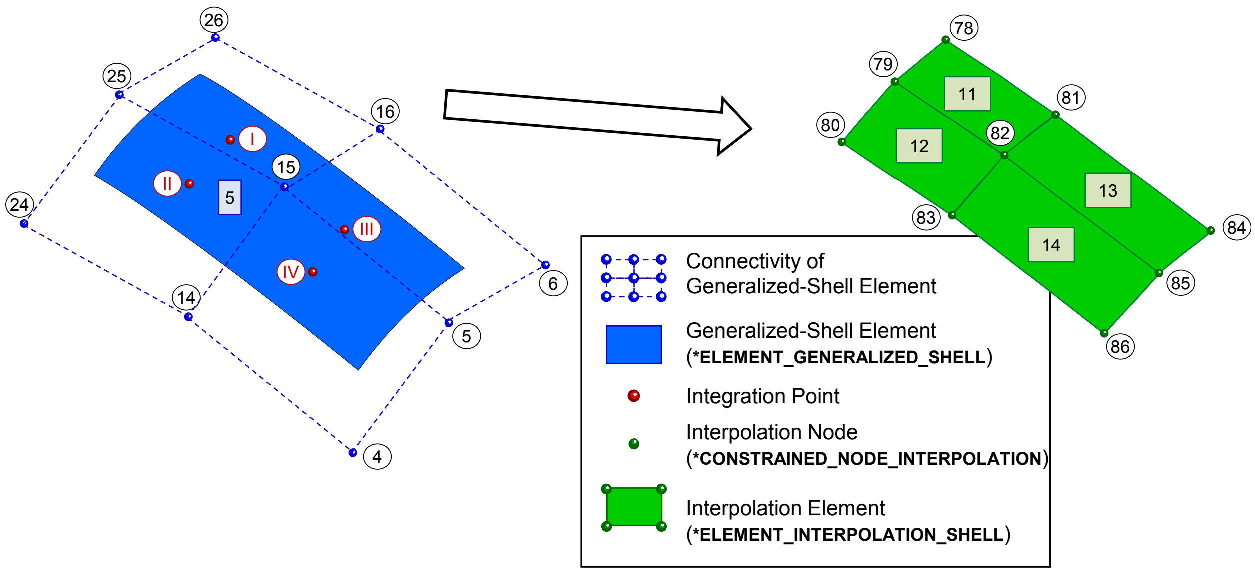
VARIABLE	DESCRIPTION
NGP	Number of in-plane integration points of the master element.
IP_i	Integration point number (1 to NGP) in the order they were defined in *DEFINE_ELEMENT_GENERALIZED_SHELL.
W_i	Interpolation weight of integration point i.

Remarks:

1. **Interpolation.** For each interpolation shell element, one single value (v_{IS}) of a solution variable is interpolated based on values at the integration points (v_i) of the master element (*ELEMENT_GENERALIZED_SHELL) and the appropriate weighting factors (w_i). The interpolation is computed as follows:

$$v_{IS} = \sum_{i=1}^{NGP} w_i v_i \ .$$

2. **Element Formulation.** To use *ELEMENT_INTERPOLATION_SHELL, ELFORM = 98 must be used in *SECTION_SHELL.




```
*CONSTRAINED_NODE_INTERPOLATION
$---+--NID----+NUMCN----+----3----+----4----+----5----+----6----+----7----+----8
        78        4
$---+--CN1----+---W1----+--CN2----+---W2----+--CN3----+---W3----+--CN4----+---W4
        26     0.35        25     0.32        15     0.18        16     0.15
*ELEMENT_SHELL
$--+-EID---+ PID---+- N1---+--N2---+--N3---+--N4---+--N5---+--N6---+--N7---+--N8
     11       33      78      79      82      81
*PART
Part for interpolation shell
$---+--PID----+SECID----+--MID----+----4----+----5----+----6----+----7----+----8
        33        45        3
*SECTION_SHELL
$---+SECID----ELFORM----+-SHRF----+--NIP----+----5----+----6----+----7----+----8
        45        98                    2
$---+---T1----+---T2----+---T3----+---T4----+----5----+----6----+----7----+----8
       1.0
*ELEMENT_INTERPOLATION_SHELL
$-----EIDS---+-EIDGS----+--NGP----+----4----+----5----+----6----+----7----+----8
       11         5         4
$---+--IP1---+----W1----+--IP2----+---W2----+--IP3----+---W3----+--IP4----+---W4
         1       0.5         2       0.2         3       0.2         4       0.1
```

Figure 19-6. Example for *ELEMENT_INTERPOLATION_SHELL.

*ELEMENT_INTERPOLATION_SOLID

Purpose: With the definition of interpolation solids, the stresses and other solution variables can be interpolated from the generalized solid elements (see *ELEMENT_GENERALIZED_SOLID and *DEFINE_ELEMENT_GENERALIZED_SOLID) permitting the solution to be visualized using standard 8-noded solid elements with one integration point (one value of each solution variable per interpolation solid). The definition of the interpolation solids is based on interpolation nodes (see *CONSTRAINED_NODE_INTERPOLATION). The connection between these various keywords are illustrated in Figure 19-7.

Card 1	1	2	3	4	5	6	7	8
Variable	EIDS	EIDGS	NGP					
Type	I	I	I					
Default	none	none	none					

Weighting Factor Cards. These cards set the weighting factors used for interpolating the solution onto the center of *this* interpolation solid. Set one weight for each of the element's NGP integration points. Each card can accommodate 4 points; define as many cards as necessary. As an example, for NGP = 10 three cards are required.

Card 2	1	2	3	4	5	6	7	8
Variable	IP1	W1	IP2	W2	IP3	W3	IP4	W4
Type	I	F	I	F	I	F	I	F
Default	none	none	none	none	none	none	none	none

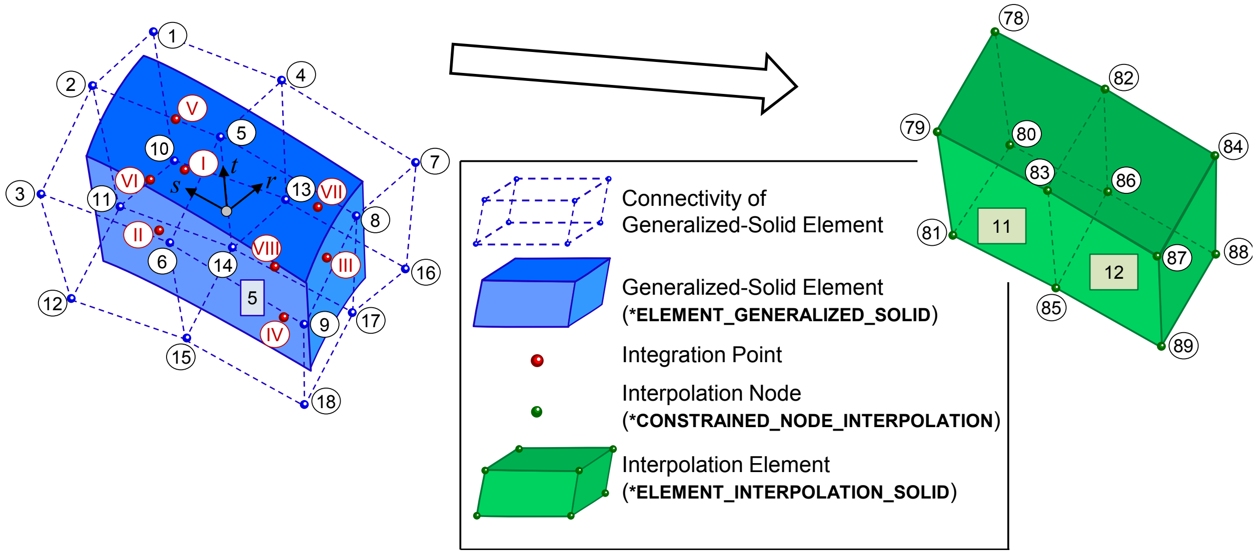
VARIABLE	DESCRIPTION
EIDS	Element ID of the interpolation solid. This needs to coincide with a proper definition of an 8-noded solid element (*ELEMENT_SOLID) using interpolation nodes (*CONSTRAINED_NODE_INTERPOLATION).
EIDGS	Element ID of the master element defined in *ELEMENT_GENERALIZED_SOLID.

VARIABLE	DESCRIPTION
NGP	Number of integration points of the master element.
IP_i	Integration point number (1 to NGP) in the order how they were defined in *DEFINE_ELEMENT_GENERALIZED_SOLID.
W_i	Interpolation weight of integration point *i*.

Remarks:

1. **Interpolation.** For each interpolation solid element, one single value (v_{IS}) of a solution variable is interpolated based on values at the integration points (v_i) of the master element (*ELEMENT_GENERALIZED_SOLID) and the appropriate weighting factors w_i. The interpolation is computed as follows: $v_{IS} = \sum_{i=1}^{NGP} w_i v_i$

2. **Element Formulation.** To use *ELEMENT_INTERPOLATION_SOLID, ELFORM = 98 must be used in *SECTION_SOLID.




```
*ELEMENT_SOLID
$---+-EID---+ PID---+- N1---+--N2---+--N3---+--N4---+--N5---+--N6---+--N7---+--N8
     11       33
$---+--N1---+--N2---+- N3---+--N4---+--N5---+--N6---+--N7---+--N8---+--N9---+-N10
      80       81       85       86       78       79       83       82
*PART
Part for interpolation solid
$---+--PID----+SECID----+--MID----+----4----+----5----+----6----+----7----+----8
        33         45          3
*SECTION_SOLID
$---+SECID----ELFORM----+--AET----+----4----+----5----+----6----+----7----+----8
        45         98
*ELEMENT_INTERPOLATION_SOLID
$-----EIDS---+-EIDGS----+--NGP----+----4----+----5----+----6----+----7----+----8
       11          5           8
$---+--IP1---+----W1----+--IP2----+---W2----+--IP3----+---W3----+--IP4----+---W4
         1      0.30           2      0.12           3      0.13           4      0.07
$---+--IP5---+----W5----+--IP6----+---W6----+--IP7----+---W7----+--IP8----+---W8
         5      0.20           6      0.08           7      0.07           8      0.03
```

Figure 19-7. Example for *ELEMENT INTERPOLATION SOLID.

*ELEMENT_LANCING

Purpose: This feature models a lancing process during a metal forming process by trimming along a curve. Two types of lancing, instant and progressive, are supported. This keyword is used together with *DEFINE_CURVE_TRIM_3D, and only applies to shell elements. The lanced scraps can be removed (trimming) during or after lancing when used in conjunction with *DEFINE_LANCE_SEED_POINT_COORDINATES; see Trimming remarks below. This feature is available for explicit dynamics for both SMP and MPP.

The element lancing feature is supported in LS-PrePost 4.3, under Application → Metal-Forming → Easy Setup.

For each trim include an additional card. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	IDPT	IDCV	IREFINE	SMIN	AT	ENDT	NTIMES	CIVD
Type	I	I	I	F	F	F	I	I
Default	none	none	1	none	none	↓	↓	↓

VARIABLE	DESCRIPTION
IDPT	A flag to indicate if a part to be lanced is a part or a part set. GT.0: IDPT is the PID of a part to be lanced; see *PART. LT.0: The absolute value is the part set ID as in *SET_PART_-LIST. This option allows the lancing to be performed across a tailor-welded line.
IDCV	A trim curve ID (the variable TCID in *DEFINE_CURVE_TRIM_-3D) defining a lancing route (see Remarks). XYZ format (TC-TYPE = 1) has always been supported; however, IGES format (TC-TYPE = 2) is not supported until Revision 110246.
IREFINE	Set IREFINE = 1 to refine elements along lancing route until no adapted nodes exist in the neighborhood. This feature results in a more robust lancing in the form of improved lancing boundary. Available starting in Revision 107708. Values greater than "1" are not allowed. See Figure 19-16 for an example of the mesh refinement.

VARIABLE	DESCRIPTION
SMIN	Minimum element characteristic length to be refined to (to be supported in the future). Currently, one level of refinement will be automatically done.
AT	Activation time for lancing operation. This variable needs to be defined for both instant and progressive lancing types (see Remarks). If CIVD is defined, AT becomes the distance from the punch home position.
ENDT	Lancing end time (for progressive lancing only). If CIVD is defined, ENDT becomes the distance from the punch home position. Do not define for instant lancing.
NTIMES	A progressive lancing operation is evenly divided into NTIMES segments between AT and ENDT; within each segment lancing is done instantly. Do not define for instant lancing.
CIVD	The load curve ID (LCID) to define the kinematics (time vs. velocity) of the tool, with VAD = 0; see *DEFINE_CURVE. When this variable is used, AT and ENDT will become the distances from the punch bottom position, unless the variables DT and BT are defined in *BOUNDARY_PRESCRIBED_MOTION_RIGID. See an example with CIVD.

Remarks:

Lancing the blank during forming at strategic locations under controlled conditions alleviates thinning and necking of sheet metal panels. Typically, the blank is lanced in the last few millimeters before the punch reaches its home position. Being an unstable process, lancing is not favored by all stampers; nevertheless, many users have devised processes which would be impossible without lancing.

The benefits of lancing are illustrated in Figure 19-8. In this figure two closed-loop holes are instantly lanced each along the C-pillar top (window opening area) and bottom (window regulator area) to improve the formability at those two corners. The right panel of Figure 19-8 is lanced and suffers less thinning compared to the no-lancing case, which is shown in the left-panel. This keyword offers two types of lancing operations:

1. **Instant lancing.** Instant lancing cuts the sheet metal once along the defined curve at a time specified in the AT field.

2. **Progressive lancing.** The cut is spatially divided into NTIMES sub-lances traveling along the curve in the direction of definition. See Figure 19-11. Progressive

lancing starts at AT and ends at ENDT thereby achieving a gradual and even release along the curve.

Modeling Guidelines:

Some modeling guidelines and limitations are listed below:

1. Both closed-loop (Figure 19-9) and open-loop (Figure 19-10) lancing curves are supported. A lancing curve may not cross another lancing curve or itself.

2. Since progressive lancing starts from the beginning of the curve and proceeds towards the end, the direction of the curve needs to be defined to match the direction of the physical cut (Figure 19-11). The direction can be set using LS-PrePost. The menu option GeoTol→ Measure with the Edge box checked can be used to show the direction of the curve. If the direction is not as desired, GeoTol → Rever can be used to reverse the direction.

3. The effect of NTIMES can be seen in Figure 19-12. Compared with NTIMES of 6, setting NTIMES to a value of 20 results in a smoother lancing boundary and less stress concentration along the separated route.

4. Although the IGES format curve is supported by the keyword *DEFINE_CURVE_TRIM_3D, curves defined with the keyword and used for lancing must be specified using only the XYZ format (TCTYPE = 1 or 0) for revisions prior to Revision 110246. The manual entry in the keyword manual for the *INTERFACE_BLANKSIZE_DEVELOPMENT keyword outlines a procedure for converting an IGES file into the required XYZ format. Note the IGES format support for lancing is enabled starting in Revision 110246.

5. The first two points as well as the last two points of any progressive lancing curve must be separated so that LS-DYNA can correctly determine the direction of the curve.

6. The lancing curve needs to be much longer than the element sizes in the lancing area.

7. To prevent mesh distortion at the end of the lancing route ENDT must be defined to be less than the simulation completion time (slightly less is sufficient).

8. Lancing can be in any direction. The direction is determined by *DEFINE_CURVE_TRIM_3D.

9. Tailor-welded blanks are supported; however, the lancing route should not cross the laser line, as currently only one part can be defined with one lancing curve.

10. Both *PARAMETER and *PARAMETER_EXPRESSION are supported for AT and ENDT as of Revision 92335. This makes it possible for users to input distance from punch home as onset of the lancing. Refer to Figure 19-14, where a punch's velocity profile is shown, the lancing activation time AT, is calculated based on the distance to home, "dhome" (the shaded area), punch travel velocity "vdraw", and total simulation time "ENDTIME". The variable CIVD implemented in Rev 110173 removes the need for the calculation of AT from punch distance to home.

11. Mesh adaptivity (*CONTROL_ADAPTIVE and the parameter "ADPOPT" under *PART) must be turned on during lancing.

12. All trim curves used to define lancing routes (*DEFINE_CURVE_TRIM_3D) must be placed after all other curves in the input deck. Furthermore, no curves defined by *DEFINE_CURVE_TRIM_3D that are not used for lancing should be present anywhere in the input deck. Note this restriction is removed starting in Revision 110316.

13. Only *DEFINE_CURVE_TRIM_3D is supported. If defined curve is far away from the blank, it will be projected onto the blank.

14. This keyword is input order sensitive: *DEFINE_CURVE_TRIM_3D must be placed after *ELEMENT_LANCING.

Application example:

A partial deck implementing instant lancing is listed below. A blank having a PID of 8 is being lanced along curves #119 and #202 instantly at 0.05 and 0.051 seconds, respectively.

```
*ELEMENT_LANCING
$      IDPT      IDCV   IREFINE      SMIN        AT      ENDT    NTIMES
          8       119                        0.0500
          8       202                        0.0510
*DEFINE_CURVE_TRIM_3D
$#     tcid    tctype      tflg      tdir     tctol      toln     nseed
        119         1         1               0.100         1
$#                 cx                  cy                  cz
            172.99310           42.632320           43.736160
            175.69769          -163.08299           46.547531
            177.46982          -278.03793           49.138161
            186.82404          -303.67191           51.217964
            205.16177          -315.33484           53.299248
                    ⋮                   ⋮                   ⋮
*DEFINE_CURVE_TRIM_3D
$#     tcid    tctype      tflg      tdir     tctol      toln     nseed
        202         1         1               0.100         1
$#                 cx                  cy                  cz
            187.46982          -578.73793           89.238161
            168.88404          -403.97191           61.417964
            215.18177          -215.03484           73.899248
                    ⋮                   ⋮                   ⋮
```

A partial keyword deck implementing two progressive lances is listed below. Both lances travel along paths starting at the same coordinate value. A sheet blank having a part ID of 9 is progressively lanced along both curves (IDCV = 1 and 2) as defined by *DEFINE_-CURVE_TRIM_3D. Both lancing operations commence at 0.05 seconds and finish at 0.053 seconds with 20 cuts along each curve in opposite directions. The lancing results are shown in Figure 19-13. Note that the termination time is 53.875 seconds, which is slightly larger than the ENDT.

```
*ELEMENT_LANCING
$     IDPT      IDCV   IREFINE      SMIN        AT      ENDT    NTIMES
         9         1                        0.0500   5.3E-02        20
         9         2                        0.0500   5.3E-02        20
*DEFINE_CURVE_TRIM_3D
$#    tcid    tctype      tflg      tdir     tctol      toln     nseed
         1         1         1               0.100         1
$#                cx                  cy                  cz
           172.99310           42.632320           43.736160
           175.69769          -163.08299           46.547531
           177.46982          -278.03793           49.138161
           186.82404          -303.67191           51.217964
           205.16177          -315.33484           53.299248
           223.13152          -308.03534           54.193089
           234.96263          -290.49695           54.885273
           222.03900          -270.08289           53.163551
           199.31226          -251.27985           50.401234
               ⋮                   ⋮                   ⋮
*DEFINE_CURVE_TRIM_3D
         2         1         1               0.100         1
$#                cx                  cy                  cz
           172.99310           42.632320           43.736160
           171.33121            47.22141           42.513367
           171.28690           128.84601           43.032799
           176.89932           149.39539           43.495331
           192.41418           159.53757           44.756699
           208.39861           158.93469           45.878036
           218.10101           149.34409           47.128345
           218.34503           135.23810           47.682144
           209.05414           122.82422           46.616959
           190.19659           117.66074           44.858204
               ⋮                   ⋮                   ⋮
```

Trimming after lancing:

As shown in Figure 19-15 and the following partial keyword file, the lanced scraps can be removed (trimming) after a lancing simulation. An extra keyword, *DEFINE_-LANCE_SEED_POINT_COORDINATES is needed to define the portion that would remain after the lancing and trimming. It should be obvious that the lancing curve defined by *DEFINE_CURVE_TRIM_3D must form a closed loop. The following example will trim a part ID 9 with a fully enclosed lancing curve #1, at time = 0.049 seconds. Since the termination time is 0.0525 seconds, the scrap will be deleted (trimmed off) before the simulation ends. The scrap, enclosed by the curve, is located outside of the seed node, defined by *DEFINE_LANCE_SEED_POINT_COORDINATES (starting in Revision 107262).

```
*CONTROL_TERMINATION
0.0525
*ELEMENT_LANCING
$      IDPT      IDCV   IREFINE      SMIN        AT      ENDT    NTIMES
          9         1                        0.0490
*DEFINE_CURVE_TRIM_3D
$#    tcid    tctype      tflg      tdir     tctol      toln     nseed
          1         1         1              0.100         1
$#                 cx                  cy                  cz
          172.99310           42.632320           43.736160
          175.69769          -163.08299           46.547531
          177.46982          -278.03793           49.138161
          186.82404          -303.67191           51.217964
          205.16177          -315.33484           53.299248
          223.13152          -308.03534           54.193089
              ⋮                   ⋮                   ⋮
          172.99310           42.632320           43.736160
          172.99310           42.632320           43.736160
*DEFINE_LANCE_SEED_POINT_COORDINATES
$    NSEED        X1        Y1        Z1        X2        Y2        Z2
          1    -289.4     98.13  2354.679
```

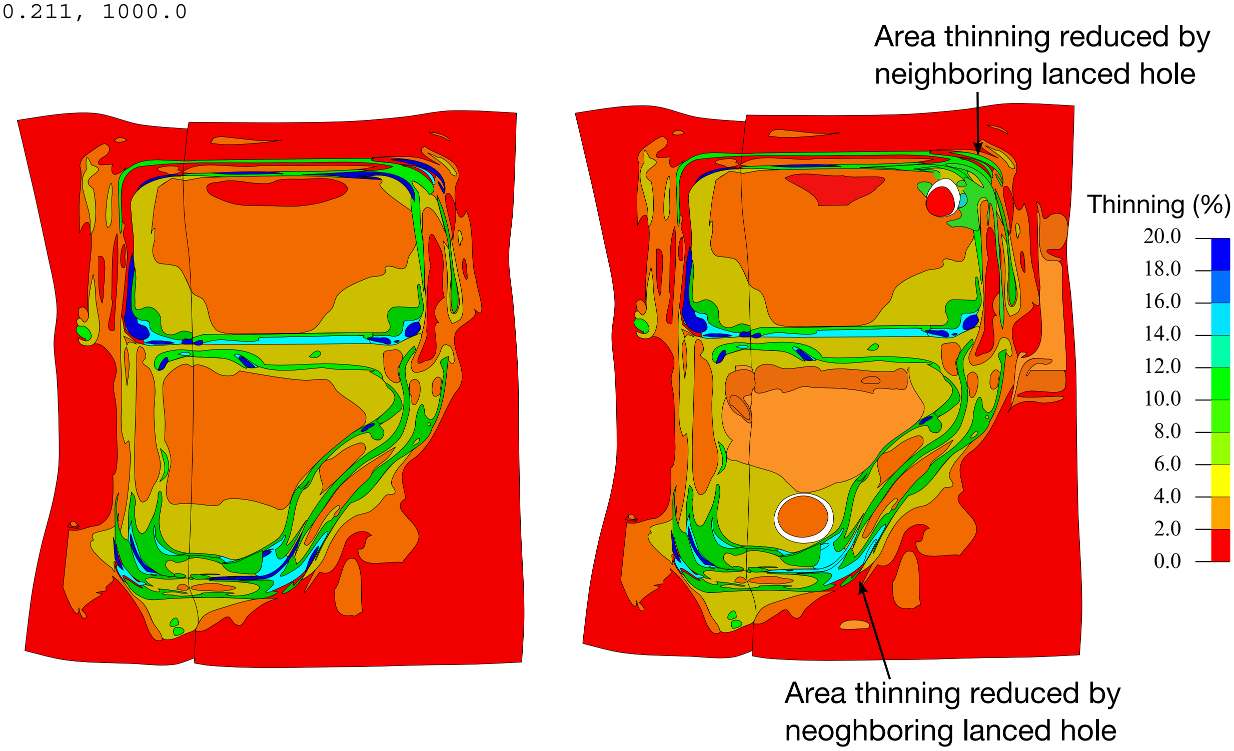
This feature also makes it possible to combine the trimming process with a forming simulation, saving a trimming step in a line-die process simulation by skipping writing out a formed dynain file and reading in the same file for the trimming simulation.

Lance-trimming with a part set, IGES curve, a seed node and CIVD:

The following partial keyword input shows instant lance-trimming across the weld line of a tailor-welded blank using the part set ID "blksid" 100, which consists of PIDs 1 and 9. The part set ID used for *ELEMENT_LANCING input is "idpt", which is set as the negative of blksid (-100). The lance-trimming curve ID 1117 is defined using the file lance4.iges in IGES format (TCTYPE = 2). The defined variable CIVD refers to the load curve ID 1115 under *DEFINE_CURVE, which is the kinematic curve for the upper die. The lancing starts at 15.5 mm (AT = 15.5) away from upper die bottom position. A lance seed coordinate (-382.0, -17.0, 76.0) is defined using the keyword *DEFINE_LANCE_-SEED_POINT_COORDINATES, resulting in the lanced scrap piece being removed after lancing.

```
*PARAMETER
I blk1pid         1
I blk2pid         9
I blksid        100
*SET_PART_LIST
&blksid
&blk1pid    &blk2pid
*PARAMETER_EXPRESSION
I idpt        -1*blksid
*ELEMENT_LANCING
$      IDPT      IDCV   IREFINE      SMIN        AT      ENDT    NTIMES      CIVD
      &idpt      1117         1                15.5                          1115
*DEFINE_CURVE_TRIM_3D
$#    tcid    tctype      tflg      tdir     tctol      toln    nseed1    nseed2
       1117         2         1         0    0.1000         0
lance4.iges
*DEFINE_LANCE_SEED_POINT_COORDINATES
$      NSUM        X1        Y1        Z1        X2        Y2        Z2
```

```
         1  -382.000    -17.000       76.0
*BOUNDARY_PRESCRIBED_MOTION_RIGID
$    TYPEID        DOF        VAD       LCID         SF        VID      DEATH      BIRTH
  &udiepid           3          0       1115
  &bindpid           3          0       1116
*DEFINE_CURVE
1115
0.0, 0.0
0.001, 1000.0
0.211, 1000.0
```



Areas of high thinning (in blue) - without lancing

Thinning contour with lancing in upper and lower C-pillar corners

Figure 19-8. Thinning improvement on a door as a result of lancing at the upper and lower corner of the C-pillar.

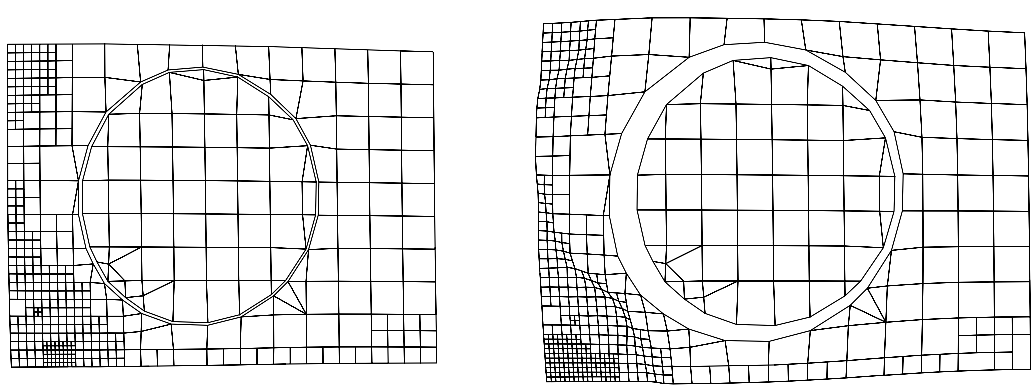


Figure 19-9. Instant lancing – closed-loop hole. The left mesh immediately after lancing at time AT while the right one when the punch is at punch home.

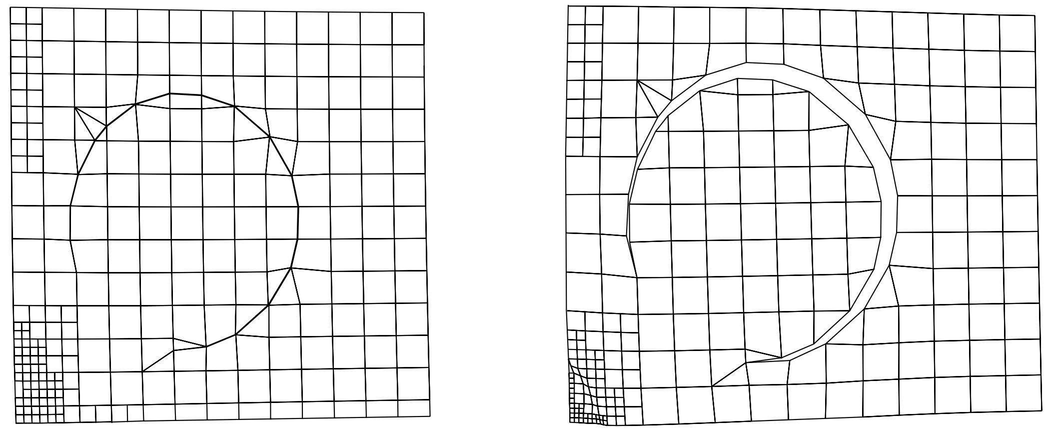


Figure 19-10. Instant lancing – open-loop hole. The left mesh immediately after lancing at time AT while the right one is when the punch is at punch home.

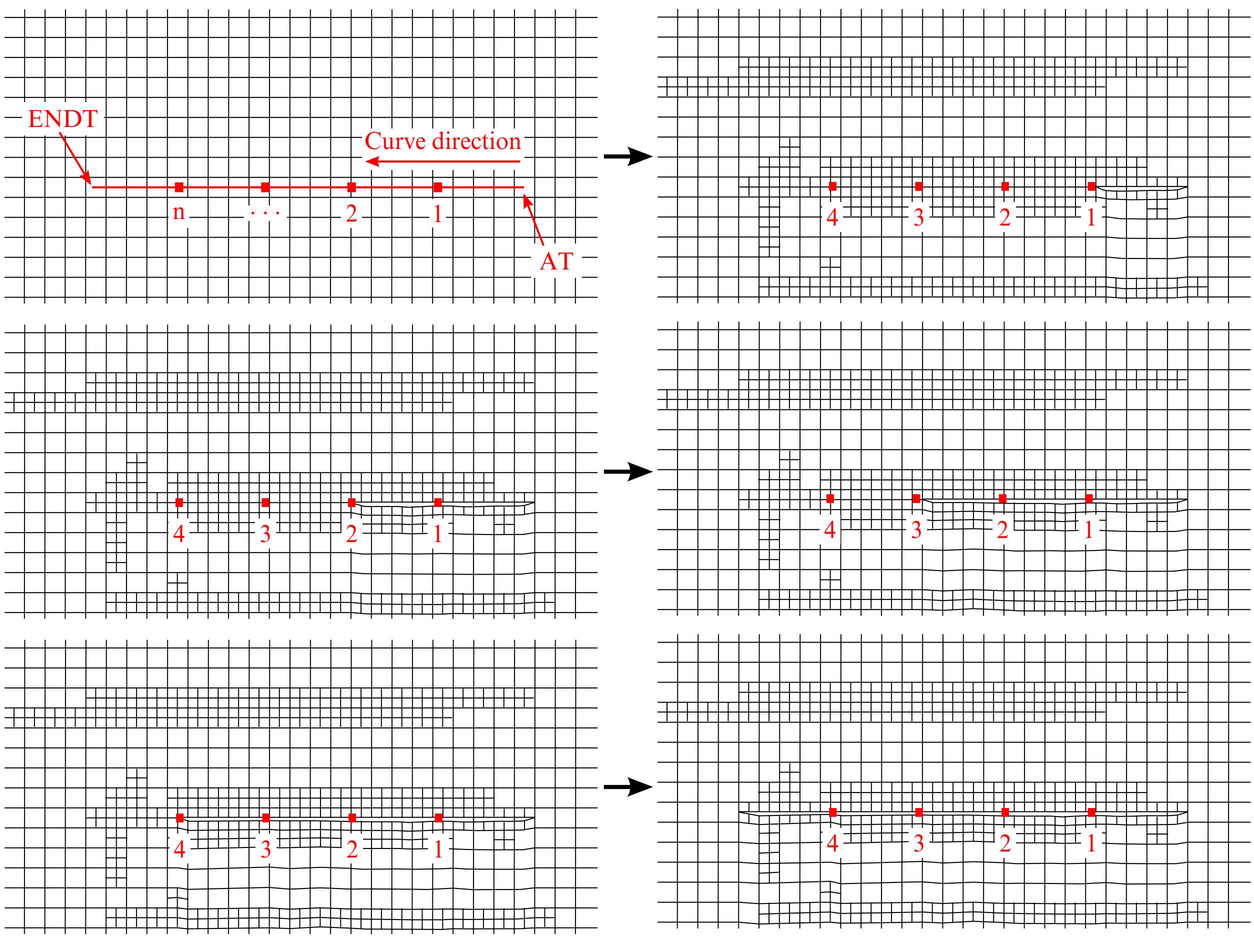


Figure 19-11. Progressive lancing - defining *AT*, *ENDT*, *NTIMES*, curve direction; mesh separation progression during progressive lancing.

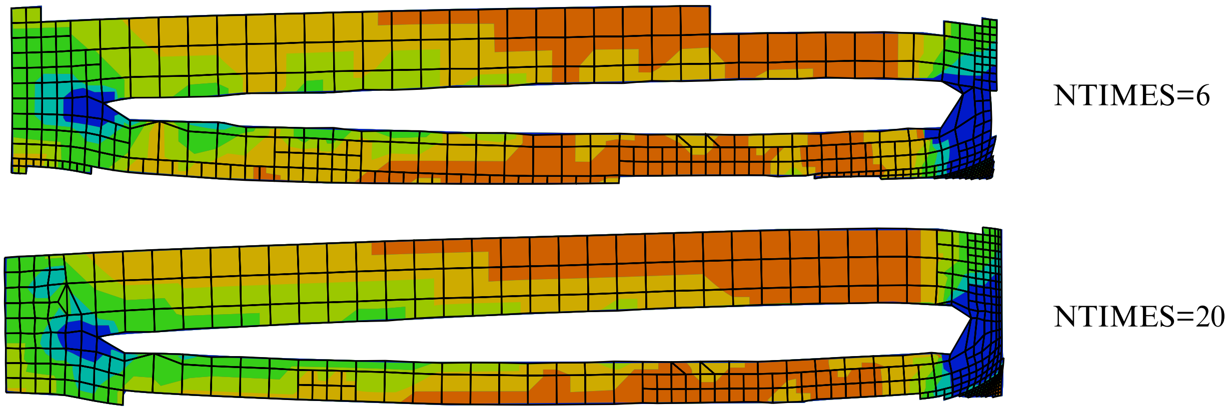


Figure 19-12. Larger NTIMES causes a smoother lancing boundary and less stress concentration.

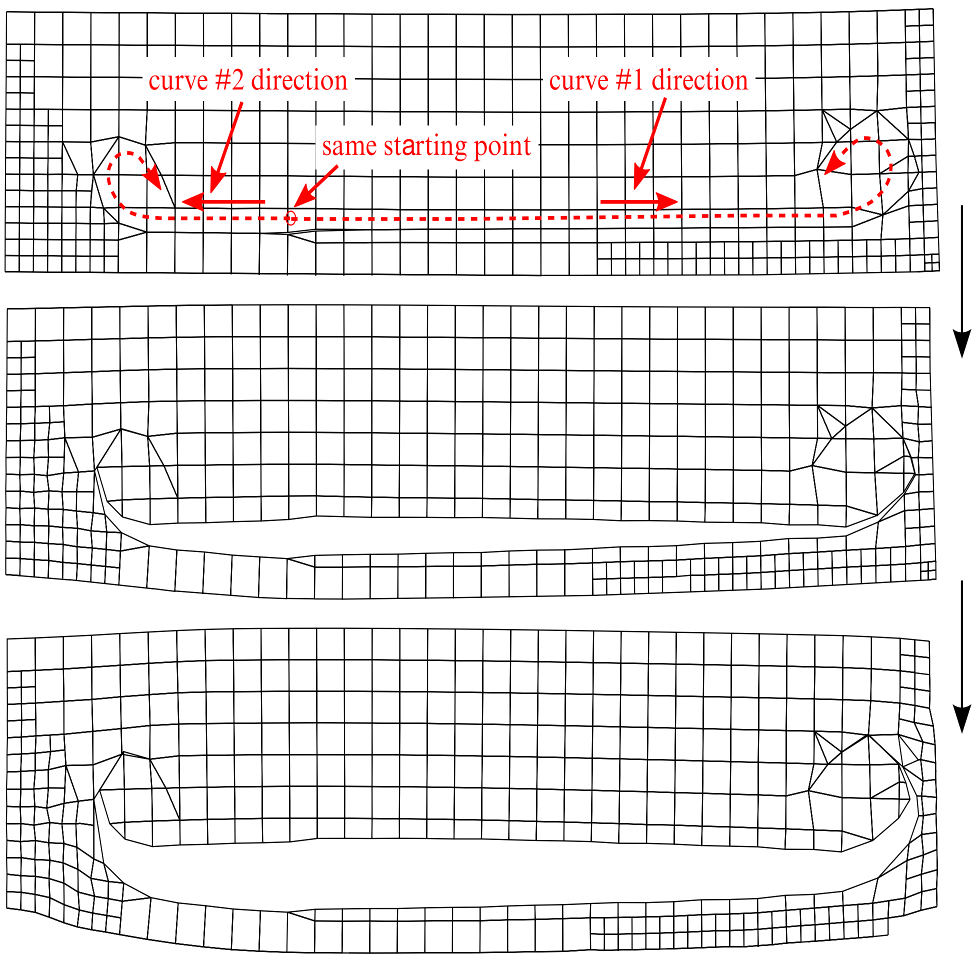
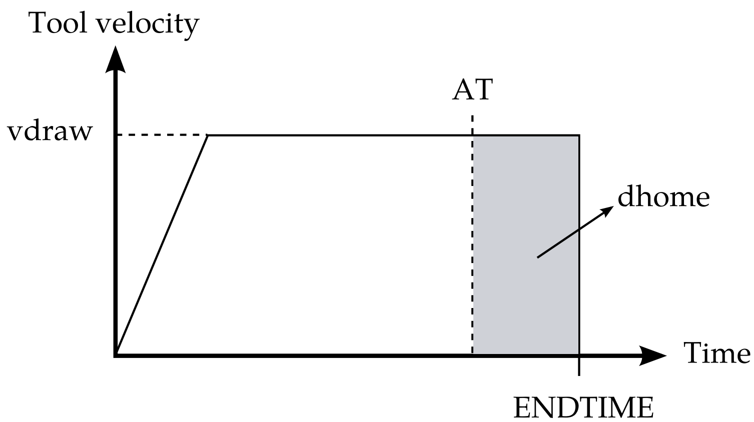


Figure 19-13. Progressive lancing – multiple lancing processes starting from the same coordinates.




```
*PARAMETER_EXPRESSION
R dhome      12.5
R at         ENDTIME-dhome/vdraw
*ELEMENT_LANCING
$ IDPT      IDCV      IREFINE      SMIN      AT
      9       112                            &at
```

Figure 19-14. An example of defining lancing activation time AT using tool's distance to home.

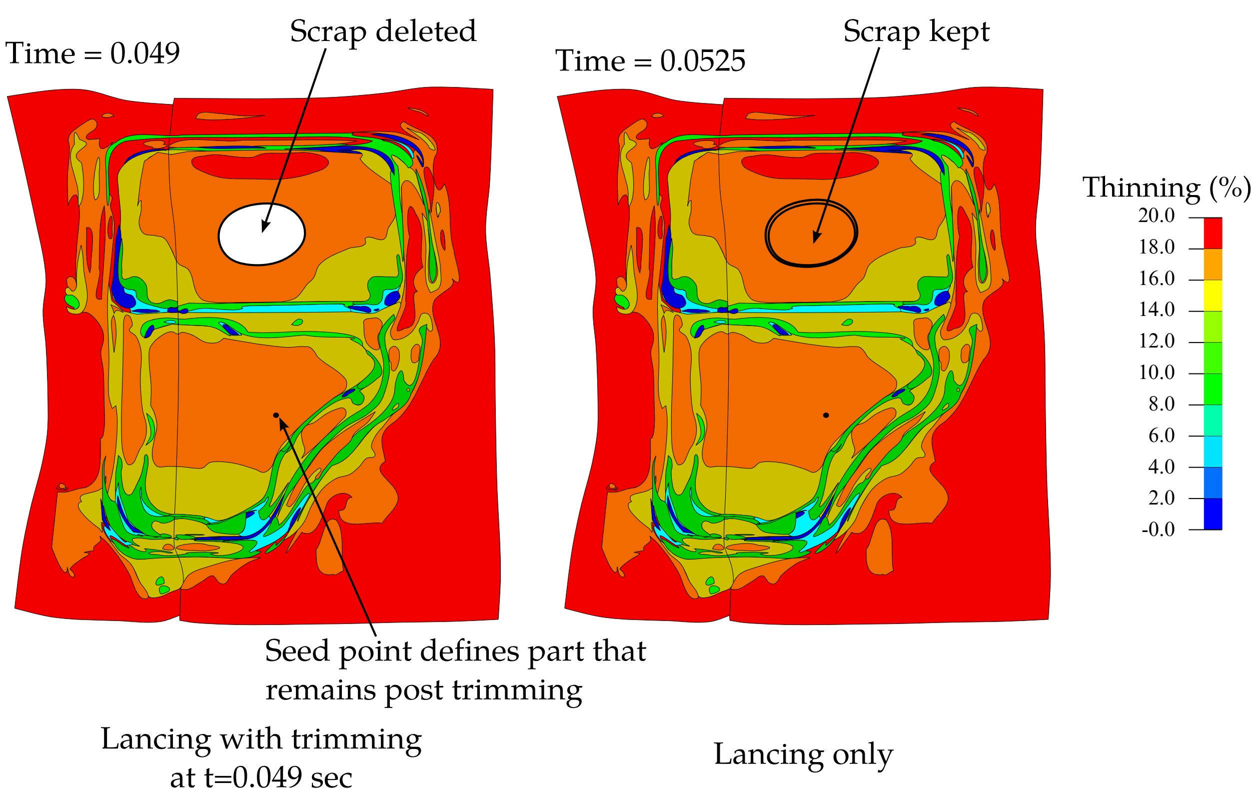
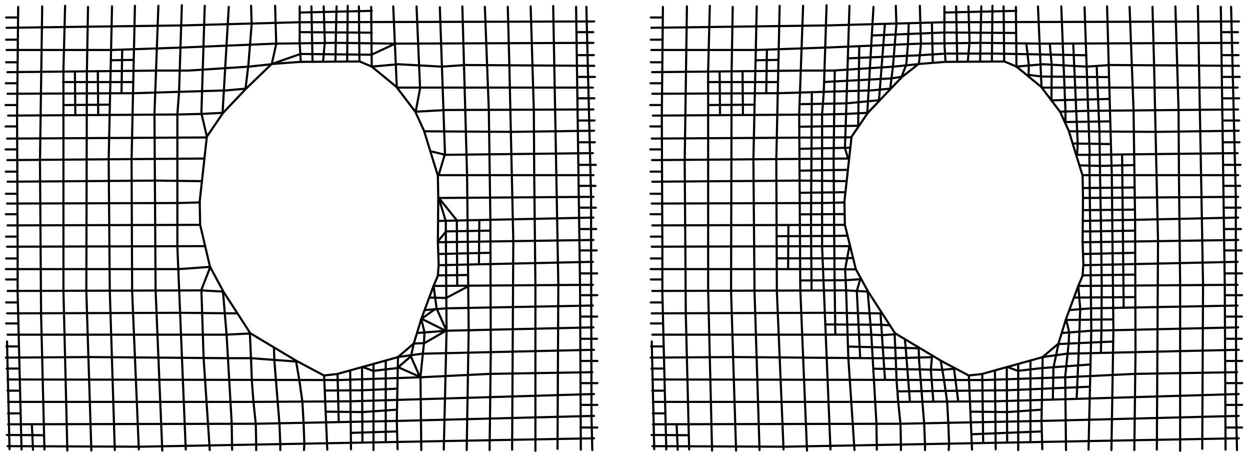


Figure 19-15. Lancing with trimming



Lanced mesh prior to Revision 107708

Improved lanced boundary mesh with IREFINE=1 after Revision 107708

Figure 19-16. Set IREFINE = 1 (Revision 107708 and later) for improved lanced boundary.

ELEMENT_MASS_*{OPTION}

Available options include:

<BLANK>

NODE_SET

Purpose: Define a lumped mass element assigned to a nodal point or equally distributed to the nodes of a node set. The nodal point can belong to either a deformable body or a rigid body.

Card	1	2	3	4	5	6	7	8	9	10
Variable	EID	ID	MASS		PID					
Type	I	I	F		I					
Default	none	none	0.		0					

VARIABLE	DESCRIPTION
EID	Element ID. A unique number is recommended. The nodes in a node set share the same element ID.
ID	Node ID or node set ID if the NODE_SET option is active. This is the node or node set to which the mass is assigned.
MASS	Mass value. When the NODE_SET option is active, the mass is equally distributed to all nodes in a node set.
PID	Part ID. This input is optional.

Remarks:

1. **Kinetic Energy Output.** Kinetic energy of lumped mass elements is output as kinetic energy of part 0 in matsum (*DATABASE_MATSUM) if IERODE is set to 1 on *CONTROL_OUTPUT.

*ELEMENT_MASS_MATRIX_{OPTION}

Available options include:

<BLANK>

NODE_SET

Purpose: Define a 6 × 6 symmetric nodal mass matrix assigned to a nodal point or each node within a node set. A node may not be included in more than one *ELEMENT_-MASS_MATRIX(_NODE_SET) command. The nodal point can only belong to a deformable body. This keyword cannot be applied to nodes that belong to rigid bodies.

Card 1	1	2	3	4	5	6	7	8
Variable	EID	ID	CID					
Type	I	I	I					
Default	none	none	0					

Card 2	1	2	3	4	5	6	7	8
Variable	M11	M21	M22	M31	M32	M33	M41	
Type	F	F	F	F	F	F	F	
Default	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

Card 3	1	2	3	4	5	6	7	8
Variable	M42	M43	M44	M51	M52	M53	M54	
Type	F	F	F	F	F	F	F	
Default	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

Card 4	1	2	3	4	5	6	7	8
Variable	M55	M61	M62	M63	M64	M65	M66	
Type	F	F	F	F	F	F	F	
Default	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

VARIABLE	DESCRIPTION
EID	Element ID. A unique number is recommended. The nodes in a node set share the same element ID.
ID	Node ID or node set ID if the NODE_SET option is active. This is the node or node set to which the mass is assigned.
CID	Local coordinate ID which defines the orientation of the mass matrix
M*ij*	The *ij*th term of the symmetric mass matrix. The lower triangular part of the matrix is defined.

ELEMENT_MASS_PART_*{OPTION}

Available options include:

<BLANK>

SET

Purpose: Define additional non-structural mass to be distributed by an area (shell) / volume (solid) or mass weighted distribution to all nodes of a given part, or part set, ID. As an option, the total mass can be defined and the additional non-structural mass is computed. This option applies to all part IDs defined by shell, tshell and solid elements.

Card	1	2	3	4	5	6	7	8	9	10
Variable	ID	ADDMASS		FINMASS		LCID		MWD		
Type	I	F		F		I		I		
Default	none	0.		0.		0		0		

VARIABLE	DESCRIPTION
ID	Part or part set ID if the SET option is active. A unique number must be used.
ADDMASS	Added translational mass to be distributed to the nodes of the part ID or part set ID. Set to zero if FINMASS is nonzero. Since the additional mass is not included in the time step calculation of the elements in the PID or SID, ADDMASS must be greater than zero if FINMASS is zero. Multiple applications of mass using the SET option are treated by the additional masses being summed. For multiple mass applications to the same PID (<BLANK> option), only the last definition is used.
FINMASS	Final translational mass of the part ID or part set ID. The total mass of the PID or SID is computed and subtracted from the final mass of the part or part set to obtain the added translational mass, which must exceed zero. Set FINMASS to zero if ADDMASS is nonzero. FINMASS is available in the R3 release of version 971.

VARIABLE	DESCRIPTION
LCID	Optional load curve ID to scale the added mass at time = 0. This curve defines the scale factor as a function versus time. The curve must start at unity at $t = 0$. This option applies to deformable bodies only.
MWD	Optional flag for mass-weighted distribution, valid for SET option only: EQ.0: non-structural mass is distributed by area (shell) / volume (solid) weighted distribution, EQ.1: non-structural mass is distributed by mass weighted, area × density × thickness (shell) / volume × density (solid), distribution. Mixed uses with MWD for the same part should be avoided.

*ELEMENT_PLOTEL

Purpose: Define a null beam element for visualization.

Card	1	2	3	4	5	6	7	8	9	10
Variable	EID	N1	N2							
Type	I	I	I							
Default	none	none	none							

VARIABLE	DESCRIPTION
EID	Element ID. A unique number must be used.
N1	Nodal point (end) 1.
N2	Nodal point (end) 2.

Remarks:

1. **Part ID.** Part ID, 10000000, is assigned to PLOTEL elements.

2. **Beam Element IDs.** PLOTEL element IDs must be unique with respect to other beam elements.

*ELEMENT_SEATBELT

Purpose: Define a seat belt element.

Card	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	SBRID	SLEN		N3	N4	
Type	I	I	I	I	I	F		I	I	
Default	none	none	none	none	Rem 2	0.0		0	0	

VARIABLE	DESCRIPTION
EID	Element ID. A unique number is required. Since null beams are created for visualization, this element ID should not be identical to element IDs defined for *ELEMENT_BEAM and *ELEMENT_DISCRETE.
PID	Part ID
N1	Node 1 ID
N2	Node 2 ID
SBRID	Retractor ID, see *ELEMENT_SEATBELT_RETRACTOR.
SLEN	Initial slack length. See Remarks 1 and 3.
N3	Optional node 3 ID. When N3 > 0 and N4 > 0, the element becomes a shell seat belt element. The thickness of the shell seatbelt is defined in *SECTION_SHELL, not in *SECTION_SEATBELT. The shell-type seatbelt must be of a rectangular shape as shown in Figure 19-17 and contained in a logically regular mesh. See Remark 4.
N4	Node 4 ID, which is required if and only if N3 is defined.

Remarks:

1. **Unstretched Length.** The unstretched length, l, of the belt is given as

$$l = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} + \mathrm{SLEN} \ ,$$

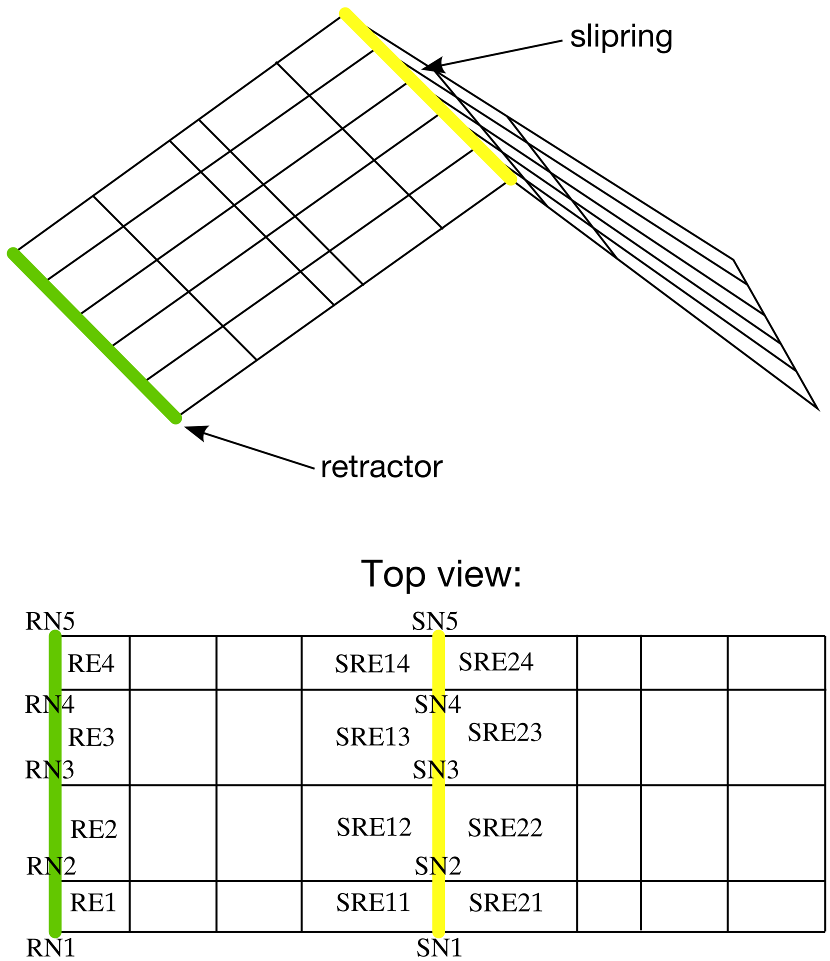


Figure 19-17. Definition of seatbelt shell elements. The ordering of the nodes and elements are important for seatbelt shells. See the input descriptions for SECTION_SHELL, ELEMENT_SEATBELT_RETRACTOR and ELEMENT_-SEATBELT_SLIPRING.

where (x_1, y_1, z_1) and (x_2, y_2, z_2) are the positions of the nodes defining the seatbelt element and SLEN is the slack length. All seatbelt elements *must* have an unstretched length *greater than* LMIN, where LMIN is a minimum length field defined on *MAT_SEATBELT. *However, when a seatbelt element is either initially connected to a slipring or is a mouth element for a retractor (see *ELEMENT_SEATBELT_SLIPRING and *ELEMENT_SEATBELT_RETRACTOR), l must be greater than 1.6 × LMIN. Also, if a seatbelt element has the potential of going through a slipring or retractor at some time during the simulation, it must have an unstretched length of at least 1.1 × LMIN.*

2. **Retractor.** The retractor ID should be defined only if the element is initially *inside* a retractor, see *ELEMENT_SEATBELT_RETRACTOR.

3. **Forces.** Belt elements are single degree of freedom elements connecting two nodes. When the strain in an element is positive (that is, the current length is

greater than the unstretched length), a tension force is calculated from the material characteristics and is applied along the current axis of the element to oppose further stretching.

4. **Shell Elements.** Seatbelt shell elements are included as a feature as of version 971 and must be used with caution. The seatbelt shells distribute the loading on the surface of the dummy more realistically than the two node belt elements. For the seatbelt shells to work with sliprings and retractors it is necessary to use a logically regular mesh of quadrilateral elements. *A seatbelt defined by a part ID must not be disjoint.*

5. **Material ID.** 1D and 2D seatbelt elements may not share the same material ID.

*ELEMENT_SEATBELT_ACCELEROMETER

Purpose: Define an accelerometer. Contrary to the keyword name, an accelerometer need not be associated with a seat belt. The accelerometer is fixed to a rigid body containing the three nodes defined below. An accelerometer will exhibit considerably less numerical noise than a deformable node, thereby reporting more meaningful data to the user. Whenever computed accelerations are compared to experimental data, or whenever computed accelerations are compared between different runs, this feature is essential.

Card	1	2	3	4	5	6	7	8
Variable	SBACID	NID1	NID2	NID3	IGRAV	INTOPT	MASS	
Type	I	I	I	I	I	I	F	
Default	none	none	none	none	0	0	0.	

VARIABLE	DESCRIPTION
SBACID	Accelerometer ID. A unique number must be used.
NID1	Node 1 ID
NID2	Node 2 ID
NID3	Node 3 ID
IGRAV	Gravitational accelerations due to body force loads: EQ.-6: z and x components removed from acceleration output EQ.-5: y and z components removed from acceleration output EQ.-4: x and y components removed from acceleration output EQ.-3: z component removed from acceleration output EQ.-2: y component removed from acceleration output EQ.-1: x component removed from acceleration output EQ.0: all components included in acceleration output EQ.1: all components removed from acceleration output GT.1: IGRAV is a curve ID defining the gravitation-flag versus time. The ordinate values, representing the gravitation-flag, can be -6, -5, -4, -3, -2, -1, 0 or 1, as described above.

VARIABLE	DESCRIPTION
	For example, a curve with 4 data points of (0. ,1), (10. ,1), (10.000001,0), (200. ,0) sets the gravitation flag to be 1 when time ≤ 10, and 0 when time > 10. In other words, all components of gravitational accelerations are removed when time ≤ 10., and then included when time > 10.0.
INTOPT	Integration option. If the accelerometer undergoes rigid body translation without rotation, this option has no effect; however, if rotation occurs, INTOPT affects how translational velocities (and displacements) are calculated. Note that the acceleration values written to the nodout file are unaffected by INTOPT. EQ.0: velocities are integrated from the global accelerations and transformed into the local system of the accelerometer. EQ.1: velocities are integrated directly from the local accelerations of the accelerometer.
MASS	Optional added mass for accelerometer. This mass is equally distributed to nodal points NID1, NID2, and NID3. This option avoids the need to use the *ELEMENT_MASS keyword input if additional mass is required.

Remarks:

The presence of the accelerometer means that the accelerations and velocities of node 1 will be output to **all** output files in local instead of global coordinates.

The local coordinate system is defined by the three nodes as follows:

1. local **x** from node 1 to node 2,

2. local **z** perpendicular to the plane containing nodes 1, 2, and 3 ($\mathbf{z} = \mathbf{x} \times \mathbf{a}$), where **a** is from node 1 to node 3,

3. local $\mathbf{y} = \mathbf{z} \times \mathbf{x}$.

The three nodes should all be part of the same rigid body. The local axis then rotates with the body.

*ELEMENT_SEATBELT_PRETENSIONER

Purpose: Define seat belt pretensioner. A combination with sensors and retractors is also possible.

Card 1	1	2	3	4	5	6	7	8
Variable	SBPRID	SBPRTY	SBSID1	SBSID2	SBSID3	SBSID4		
Type	I	I	I	I	I	I		
Default	none	none	0	0	0	0		

Card 2	1	2	3	4	5	6	7	8
Variable	SBRID	TIME	PTLCID	LMTFRC	LMTPIN			
Type	I	F	I	F	F			
Default	0	0.0	0	0	0			

VARIABLE	DESCRIPTION
SBPRID	Pretensioner ID. A unique number has to be used.
SBPRTY	Pretensioner type (see Activation): EQ.1: Pyrotechnic retractor with force limits (see Type 1), EQ.2: Pre-loaded spring becomes active (see Types 2 and 3), EQ.3: Lock spring removed (see Types 2 and 3), EQ.4: Force as a function of time retractor (see Type 4). EQ.5: Pyrotechnic retractor (older implementation of type 1) but with optional force limiter, LMTFRC (see Type 1 and Type 5). EQ.6: Combination of types 4 and 5 as described in Type 6 below. EQ.7: Independent pretensioner/retractor (see Type 7).

VARIABLE	DESCRIPTION
	EQ.8: Energy as a function of time retractor pretensioner with optional force limiter, LMTFRC (see Type 8).
	EQ.9: Energy as a function of time buckle or anchor pretensioner (see Type 9).
SBSID1	Sensor 1, see *ELEMENT_SEATBELT_SENSOR. See Activation.
SBSID2	Sensor 2, see *ELEMENT_SEATBELT_SENSOR.
SBSID3	Sensor 3, see *ELEMENT_SEATBELT_SENSOR.
SBSID4	Sensor 4, see *ELEMENT_SEATBELT_SENSOR.
SBRID	Retractor number (SBPRTY = 1, 4, 5, 6, 7 or 8) or spring element number (SBPRTY = 2, 3 or 9).
TIME	Time between sensor triggering and pretensioner acting.
PTLCID	Load curve for pretensioner (Time after activation, Pull-in) (SBPRTY = 1, 4, 5, 6, 7, 8 or 9).
LMTFRC	Optional limiting force for retractor types 5 and 8. If zero, this option is ignored. See Type 5 and Type 8.
LMTPIN	Optional limiting pull-in for retractor types 4, 6, and 7. LMTPIN must be ≥ 0.0. If zero, this option is ignored. See Type 4, Type 6, and Type 7.

Activation:

To activate the pretensioner, the following sequence of events must occur:

1. Any one of up to four sensors must be triggered.
2. Then a user-defined time delay occurs.
3. Then the pretensioner acts.

At least one sensor should be defined.

Pretensioners allow modeling of seven types of active devices which tighten the belt during the initial stages of a crash. Types 1 and 5 implement a pyrotechnic device which spins the spool of a retractor, causing the belt to be reeled in. The user defines a pull-in as a function of time curve which applies once the pretensioner activates. Types 2 and 3 implement preloaded springs or torsion bars which move the buckle when released.

Types 2 and 3:

The pretensioner is associated with any type of spring element including rotational. Note that the preloaded spring, locking spring, and any restraints on the motion of the associated nodes are defined in the normal way; the action of the pretensioner is merely to cancel the force in one spring until (or after) it fires. With the second type, the force in the spring element is canceled out until the pretensioner is activated. In this case the spring in question is normally a stiff, linear spring which acts as a locking mechanism, preventing motion of the seat belt buckle relative to the vehicle. A preloaded spring is defined in parallel with the locking spring. This type avoids the problem of the buckle being free to "drift" before the pretensioner is activated. Types 4, 6, and 7, force types, are described below.

Type 1:

The older implementation of the type 1 pretensioner (essentially available as type 5, but type 5 restricts the force value) requires a load curve that tabulates the pull-in of the pretensioner as a function of time. This pretensioner type interacts with the retractor, forcing it to pull in by the amount of belt indicated. It works well, and does exactly what it says it will do, but it can be difficult to use because it has no regard for the forces being exerted on the belt. If a pull-in of 20 mm is specified at a particular time, then 20 mm of belt will be pulled in, even if this results in unrealistic forces in the seatbelt. Furthermore, there was no explicit way to turn this pretensioner off. Once defined, it overrode the retractor completely, and the amount of belt passing into or out of the retractor depended solely on the load curve specified.

The behavior of the type 1 pretensioner was changed due to user feedback regarding these shortcomings. Each retractor has a loading (and optional unloading) curve that describes the force on the belt element as a function of the amount of belt that has been pulled out of the retractor since the retractor locked. The current type 1 pretensioner acts as a shift on this retractor load curve. An example will make this clear. Suppose at a particular time that 5 mm of belt material has left the retractor. The retractor will respond with a force corresponding to 5 mm pull-out on its loading curve. But suppose this retractor has a type 1 pretensioner defined, and at this instant of time the pretensioner specifies a pull-in of 20 mm. The retractor will then respond with a force that corresponds to (5mm + 20mm) on its loading curve. This results in a much larger force. The effect can be that belt material will be pulled in, but unlike with the older implementation, there is no guarantee. The benefit of this implementation is that the force as a function of pull-in load curve for the retractor is followed and no unrealistic forces are generated. Still, it may be difficult to produce realistic models using this option, so other pretensioners have been added.

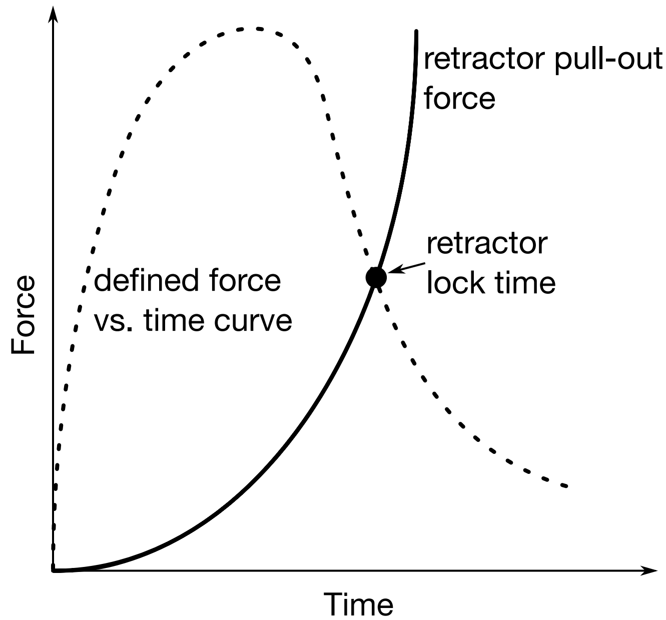


Figure 19-18. Force as a function of time for the pretensioner. At the intersection, the retractor locks.

Type 4:

The type 4 pretensioner requires a force as a function of time curve (see Figure 19-18). At each time step, the retractor computes the desired force without regard to the pretensioner. If the resulting force is less than that specified by the pretensioner load curve, then the pretensioner value is used instead. As time goes on, the pretensioner load curve should drop below the forces generated by the retractor, and the pretensioner is then essentially inactive. If LMTPIN > 0, the pretensioner will be disabled if the pull-in is > LMTPIN. This provides for good control of the actual forces, so no unrealistic values are generated. The actual direction and amount of belt movement is unspecified and will depend on the other forces being exerted on the belt. This is suitable when the force the pretensioner exerts over time is known.

Type 5:

The type 5 pretensioner is essentially the same as the old type 1 pretensioner, but with the addition of a force limiting value. The pull-in is given as a function of time, and the belt is drawn into the retractor exactly as desired. However, if at any point the forces generated in the belt exceed the pretensioner force limit, then the pretensioner is deactivated and the retractor takes over. In order to prevent a large discontinuity in the force at this point, the loading curve for the retractor is shifted (in the abscissa) by the amount required to put the current (pull-out,force) on the load curve. For example, suppose the current force is 1000, and the current pull-out is -10 (10 mm of belt has been *pulled in* by the pretensioner). If the retractor would normally generate a force of 1000 after 25 mm of belt had been *pulled out*, then the load curve is shifted to the left by 35 and remains that way for the duration of the calculation. So that at the current pull-in of 10, it will generate the force normally associated with a pull-out of 25. If the belt reaches a pull-out of 5, the force will be as if it were pulled out 40 (5 + the shift of 35), and so on. This option is included for those who liked the general behavior of the old type 1 pretensioner but has the added feature of the force limit to prevent unrealistic behavior.

Note that the retractor will be locked if either of the following two conditions are met:

- The current time exceeds the maximum abscissa of PTLCID, or
- LMTFRC is reached.

Type 6:

The type 6 pretensioner is a variation of the type 4 pretensioner, with features of the type 5 pretensioner. A force as a function of time curve is input and the pretensioner force is computed each cycle. The retractor linked to this pretensioner should specify a positive value for pull, which is the distance the belt pulls out before it locks. As the pretensioner pulls the belt into the retractor, the amount of pull-in is tracked. As the pretensioner force decreases and drops below the belt tension, belt will begin to move back out of the retractor. If LMTPIN > 0, the pretensioner will be disabled if the pull-in is > LMTPIN. Once pull amount of belt has moved out of the retractor (relative to the maximum pull in encountered), the retractor will lock. At this time, the pretensioner is disabled, and the retractor force curve is shifted to match the current belt tension. This shifting is done just like the type 5 pretensioner. It is important that a positive value of pull be specified to prevent premature retractor locking which could occur due to small outward belt movements generated by noise in the simulation.

Type 7:

The type 7 pretensioner is a simple combination of retractor and pretensioner. It is similar to the type 6 pretensioner except for the following changes:

1. when the retractor locks, the pretensioner is *not* disabled – it continues to exert force according to the force as a function of time curve until the end of the simulation. (The force as a function of time curve should probably drop to 0 at some time.)

2. The retractor load curve is not shifted – the retractor begins to exert force according to the force as a function of pull-out curve. These two forces are added together and applied to the belt. Thus, the pretensioner and retractor are essentially independent.

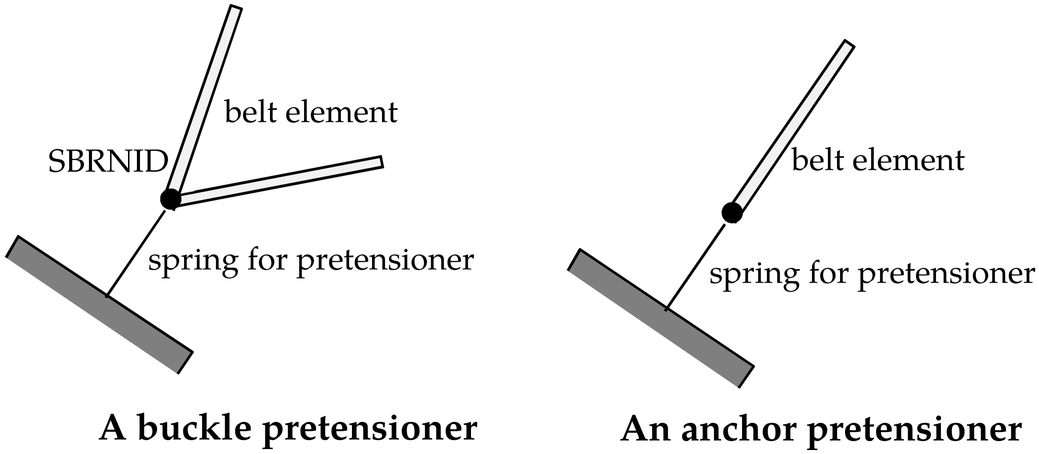


Figure 19-19. Schematics of different kinds of pretensioners.

Type 8:

The type 8 pretensioner is a variation of the type 5 pretensioner. The pretension energy, instead of pull-in for type 5, is given as a function of time. This enables users to use a single pretensioner curve, PTLCID, for various sizes of dummies. The energy could be yielded from the baseline test or simulation by

$$E(t) = \int_0^t f dp \ ,$$

where f is the force of the mouth element of the retractor and dp is the incremental pull-in.

Note that the retractor will be locked if either of the following two conditions are met:

- The current time exceeds the maximum abscissa of PTLCID, or
- LMTFRC is reached.

Type 9:

The type 9 pretensioner is designed for a pretension-energy based buckle or anchor pretensioner. The pretensioner is modeled as a spring element, SBRID. One end of the spring element is attached to the vehicle. For a buckle pretensioner, the other end of SBRID is the slip ring node, SBRNID, of a slip ring representing the buckle. For an anchor pretensioner, SBRID shares the other end with a belt element; see Figure 19-19.

*ELEMENT_SEATBELT_RETRACTOR

Purpose: Define seat belt retractor. See Remark 4 below for seat belt shell elements.

Card 1	1	2	3	4	5	6	7	8
Variable	SBRID	SBRNID	SBID	SID1	SID2	SID3	SID4	DSID
Type	I	I	I	I	I	I	I	I
Default	none	none	none	0	0	0	0	0
Remarks		3, 4	4	5				

Card 2	1	2	3	4	5	6	7	8
Variable	TDEL	PULL	LLCID	ULCID	LFED	LCFL	FLOPT	
Type	F	F	I	I	F	I	I	
Default	0.0	0.0	0	0	0.0	0	0	
Remarks			6, 8	7, 8	2, 1			

VARIABLE	DESCRIPTION
SBRID	Retractor ID. A unique number has to be used.
SBRNID	Retractor node ID
SBID	Seat belt element ID
SID1	Sensor ID 1
SID2	Sensor ID 2
SID3	Sensor ID 3
SID4	Sensor ID 4

VARIABLE	DESCRIPTION
DSID	Retractor deactivation sensor
TDEL	Time delay after sensor triggers
PULL	Amount of pull-out between the time delay ending and the retractor locking, a length value.
LLCID	Load curve for loading (Pull-out, Force); see Figure 19-20.
ULCID	Load curve for unloading (Pull-out, Force); see Figure 19-20.
LFED	Fed length
LCFL	Curve representing an adaptive multi-level load limiter (see Remark 10). The abscissa is the ID of a *SENSOR_SWITCH, and the ordinate is the corresponding force limit when the sensor switch meets the switch condition. For example, a curve of two data points (100, 4000.) and (200, 3000.) has a load limit of 4000 from switch 100 and a load limit of 3000 from switch 200. The setting of FLOPT determines how LS-DYNA uses these pairs to determine the load limit.
FLOPT	Flag giving the algorithm for determining the limiting force from the data points of curve LCFL. EQ.0: LS-DYNA checks the status of the sensor switches based on the sequence of data points. For example, LS-DYNA checks the switch of the 1st data point first. If its switch condition is met, the active force limit is the force of the 1st data point. That force level stays active until the 2nd switch meets the switch condition. EQ.1: LS-DYNA checks the status of all unfired switches in curve LCFL. The force of the latest fired switch gives the active force limit. LS-DYNA does not check that switch again in the future. That force level stays active until any other switch fires. The input order of the data points is irrelevant. EQ.2: LS-DYNA checks the status of all switches of all data points in curve LCFL. The input order of the data points is irrelevant. The active force limit is the minimum of the forces of all fired switches.

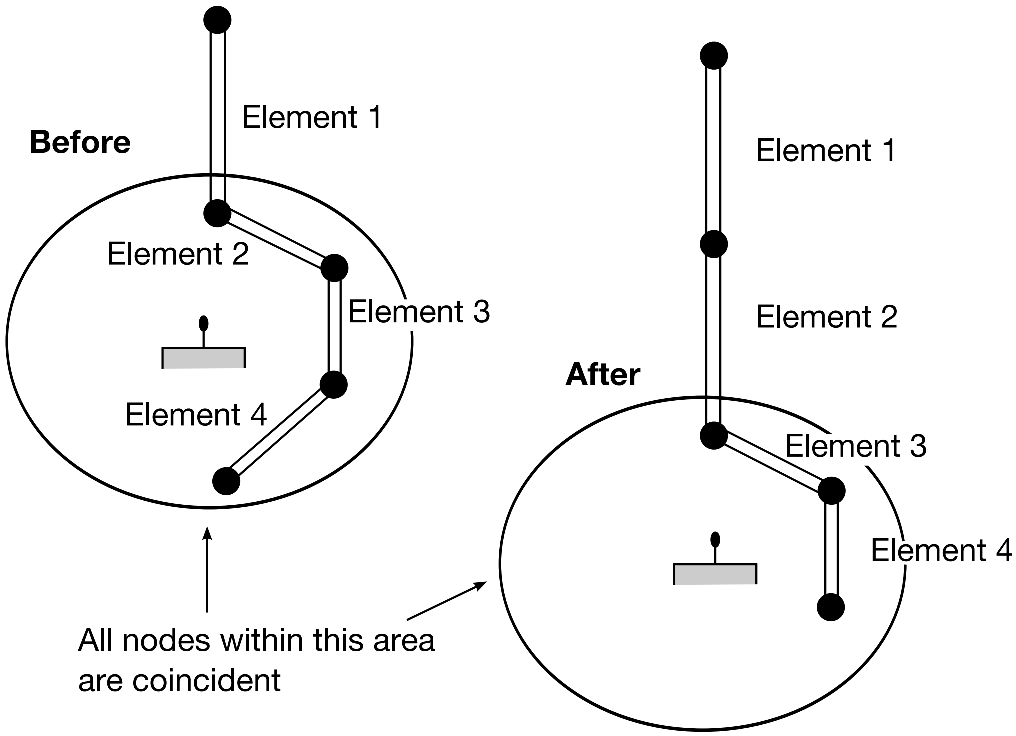


Figure 19-20. Elements in a retractor.

Remarks:

1. **Defining a Retractor.** To define a retractor, input the retractor node and the 'mouth' element (into which belt material will be fed), shown as Element 1 in Figure 19-20. Creating a retractor also requires specifying up to four sensors for triggering unlocking, a time delay, a pay-out delay (optional), load and unload curves, and the fed length. To optionally deactivated the retractor, provide a deactivation sensor, DSID. Deactivation stops any further pay-in or pay-out. Typically part of the vehicle contains the retractor node. Do not connect belt elements to this node directly. However, you can attach any other feature, including rigid bodies, to this node. The mouth element should have a node coincident with the retractor but should not be inside the retractor. The unstretched length of the mouth element during initialization *must be at least* 1.6 × LMIN, where LMIN is a field defined in *MAT_SEATBELT. Also, any other element that might become a mouth element *must have an unstretched length of at least* 1.1 × LMIN. The unstretched length, l, of a seat belt element is

$$l = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} + \text{SLEN} \ ,$$

where (x_1, y_1, z_1) and (x_2, y_2, z_2) are the locations of the two nodes defining the seat belt element, and SLEN is the slack length (see *ELEMENT_SEATBELT). The fed length, LFED, is usually set to either a typical element initial length or

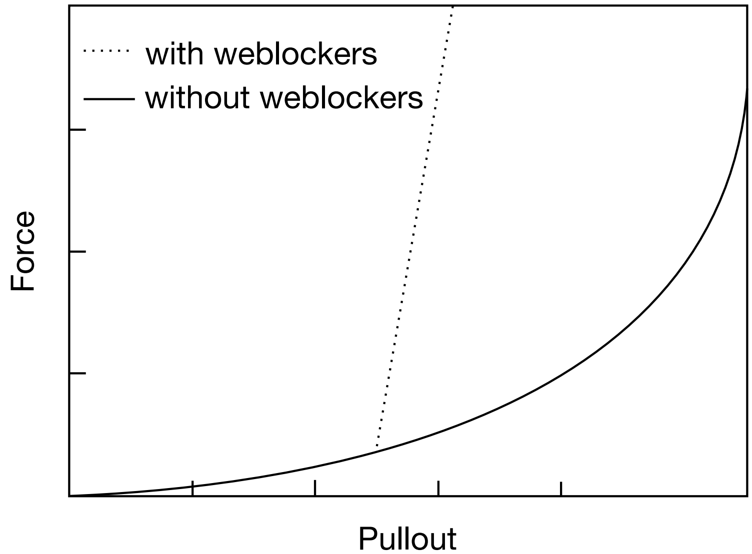


Figure 19-21. Retractor force pull characteristics.

the distance between painted marks on a real belt for comparisons with high-speed film. *The fed length should be at least three times the minimum length, LMIN.*

If the retractor initially has elements inside it (Elements 2, 3, and 4 in Figure 19-20), the retractor input should not refer to them, but the element input for these elements should identify the retractor. Their nodes should all coincide with the retractor node and should not be restrained or constrained. LS-DYNA automatically sets the initial slack to 1.1 × minimum length for these elements; this overrides any user-defined value.

To include weblockers in the retractor representation, enter a 'locking up' characteristic in the force pull-out curve; see Figure 19-21. The final section can be very steep (but must have a finite slope).

2. **Multiple Belt Elements.** If desired, you can define belt elements that are initially inside the retractor. These emerge as belt material is paid out and return into the retractor if sufficient material is reeled in during unloading. For example, the retractor in Figure 19-20 initially contains Elements 2, 3, and 4. It pays out material into Element 1. When the retractor has fed L_{crit} into Element 1, where

$$L_{\mathrm{crit}} = \mathrm{LFED} - 1.1 \times \mathrm{LMIN} \ ,$$

Element 2 emerges with an unstretched length of 1.1 × minimum length; the algorithm reduces the unstretched length of Element 1 by the same amount. Here, LMIN is the minimum length defined on the belt material input (see *MAT_-SEATBELT), and LFED is the fed length. The force and strain in Element 1 are unchanged; in Element 2, they are set equal to those in Element 1. The retractor now pays out material into Element 2.

If no elements are inside the retractor, Element 2 can continue to extend as more material is fed into it.

As the retractor pulls in the belt (for example, during initial tightening), if the unstretched length of the mouth element becomes less than the minimum length, the element goes into the retractor.

3. **Retractor node and belt elements.** No belt elements should include the retractor node. The element specified with SBID should have one node coincident with the retractor node but should not be inside the retractor.

4. **Shell-type seat belt.** When SBRNID < 0, this retractor is for shell-type seat belts. -SBRNID is the *SET_NODE containing RN1, RN2, …RN5. SBID is then a *SET_SHELL_LIST. Note that the numbering of –SBRNID and SBID must be consistent in the direction of numbering (see Figure 19-17). For example, if *SET_NODE for SBRNID has nodes of (RN1, RN2, RN3, RN4, RN5), then *SET_SHELL_LIST for SBID should have elements of (RE1, RE2, RE3, RE4). .

5. **Sensor Definitions.** Define at least one sensor.

6. **Loading Curve.** The first point of the load curve should be $(0, T_{\mathrm{min}})$. T_{min} is the minimum tension. All subsequent tension values must be greater than T_{min}.

7. **Unloading Curve.** The unloading curve should start at zero tension and increase monotonically (i.e., no segments of negative or zero slope).

8. **Retractor Characteristics and Loading.** Retractors allow belt material to be paid out into a belt element. Retractors operate in one of two regimes: unlocked when the belt material is paid out, or reeled in under constant tension and locked when a user-defined force-pull-out relationship applies.

   The retractor is initially unlocked. The following sequence of events must occur for it to become locked:

   a) Any one of up to four sensors must be triggered. (The sensors are described below.)

   b) Then, a user-defined time delay occurs.

   c) Next, a user-defined belt length must be paid out (optional).

   d) Finally, the retractor locks, and once locked, it remains locked.

   In the unlocked regime, the retractor attempts to apply constant tension to the belt. This feature allows an initial tightening of the belt and takes up any slack whenever it occurs. The tension value comes from the first point on the force-pull-out load curve. $0.01 \times$ fed length per time step gives the maximum rate of pull-out or pull-in. Because of this, the constant tension value is not always achieved.

In the locked regime, a user-defined curve describes the relationship between the force in the attached element and the amount of belt material paid out. If the tension in the belt subsequently relaxes, a different user-defined curve applies for unloading. The unloading curve is followed until the minimum tension is reached.

Define the curves in terms of the initial length of the belt. For example, if a belt is marked at 10 mm intervals and then wound onto a retractor, and the force required to make each mark emerge from the (locked) retractor is recorded, the curves used for input would be as follows:

0 mm	Minimum tension (should be > zero)
10 mm	Force to emergence of first mark
20 mm	Force to emergence of second mark
⋮	

You may define pyrotechnic pretensions, which cause the retractor to pull in the belt at a predetermined rate. This feature overrides the retractor force-pull-out relationship from the moment when the pretensioner activates.

9. **Locking and Pretensioners.** When the model only contains retractors, be aware that we measure the pull-out from the point when the retractor locks. If the belt has pulled in since the retractor locked, then the minimum force occurs in the retractor until the system pays out enough belt to return to its locking point.

   If the behavior described above is undesirable, we recommend the type 6 pretensioner model for the seat belt system. It defines a constant force as a function of time load curve with a force equal to minimum tension, with a small PULL value on the retractor. This setup has an active pretensioner until the belt pulls all the way in. As soon as the belt starts to move back out, the algorithm disables the pretensioner, and the retractor takes over.

10. **Load limiters of pretensioners and retractors.** The load limiter of a retractor differs from the load limiter of pretensioner types 5 and 8. The algorithm activates the pretensioner's load limiter when the pretensioner is on and the retractor has not locked yet. The algorithm applies the retractor's load limiter when the pretensioner is off.

*ELEMENT_SEATBELT_SENSOR

Purpose: Define seat belt sensor. Five types are possible; see Remark 3.

Card Summary:

Card 1. This card is required.

SBSID	SBSTYP	SBSFL					

Card 2a. This card is included if and only if SBSTYP = 1.

NID	DOF	ACC	ATIME				

Card 2b. This card is included if and only if SBSTYP = 2.

SBRID	PULRAT	PULTIM					

Card 2c. This card is included if and only if SBSTYP = 3.

TIME							

Card 2d. This card is included if and only if SBSTYP = 4.

NID1	NID2	DMX	DMN	NID1			

Card 2e. This card is included if and only if SBSTYP = 5.

SBRID	PAYMX	PAYMN					

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	SBSID	SBSTYP	SBSFL					
Type	I	I	I					
Default	none	none	0					

VARIABLE	DESCRIPTION
SBSID	Sensor ID. A unique number must be used.

VARIABLE	DESCRIPTION
SBSTYP	Sensor type: EQ.1: acceleration of node EQ.2: retractor pull-out rate EQ.3: time EQ.4: distance between nodes EQ.5: retractor pull-out
SBSFL	Sensor flag: EQ.0: sensor is not active during dynamic relaxation EQ.1: sensor can be triggered during dynamic relaxation

Additional card for SBSTYP = 1.

Card 2a	1	2	3	4	5	6	7	8
Variable	NID	DOF	ACC	ATIME				
Type	I	I	F	F				
Default	none	none	0.0	0.0				

VARIABLE	DESCRIPTION
NID	Node ID of sensor. See Remark 1.
DOF	Degree of freedom: EQ.1: x EQ.2: y EQ.3: z
ACC	Activating acceleration
ATIME	Time over which acceleration must be exceeded

Additional card for SBSTYP = 2.

Card 2b	1	2	3	4	5	6	7	8
Variable	SBRID	PULRAT	PULTIM					
Type	I	F	F					
Default	0	0.0	0.0					

VARIABLE	DESCRIPTION
SBRID	Retractor ID; see *ELEMENT_SEATBELT_RETRACTOR.
PULRAT	Rate of pull-out (length/time units)
PULTIM	Time over which rate of pull-out must be exceeded

Additional card for SBSTYP = 3.

Card 2c	1	2	3	4	5	6	7	8
Variable	TIME							
Type	F							
Default	0.0							

VARIABLE	DESCRIPTION
TIME	Time at which sensor triggers

Additional card for SBSTYP = 4.

Card 2d	1	2	3	4	5	6	7	8
Variable	NID1	NID2	DMX	DMN				
Type	I	I	F	F				
Default	0	0	0.0	0.0				
Remarks			2	2				

VARIABLE	DESCRIPTION
NID1	Node 1 ID
NID2	Node 2 ID
DMX	Maximum distance
DMN	Minimum distance

Additional card for SBSTYP = 5.

Card 2e	1	2	3	4	5	6	7	8
Variable	SBRID	PULMX	PULMN					
Type	I	F	F					
Default	0	10^{16}	-10^{16}					

VARIABLE	DESCRIPTION
SBRID	Retractor ID; see *ELEMENT_SEATBELT_RETRACTOR.
PULMX	Maximum pull-out
PULMN	Minimum pull-out

Remarks:

1. **Node ID for Acceleration of Node Sensor.** Node should not be on rigid body, velocity boundary condition, or other 'imposed motion' feature.

2. **Distance Sensor.** Sensor triggers when the distance between the two nodes is $d \geq d_{\mathrm{max}}$ or $d \leq d_{\mathrm{min}}$.

3. **Sensor Types.** Sensors are used to trigger locking of retractors and to activate pretensioners. Four types of sensors are available which trigger according to the following criteria:

   Type 1: When the magnitude of x-, y-, or z-acceleration of a given node has remained above a given level continuously for a given time, the sensor triggers. This does not work with nodes on rigid bodies.

   Type 2: When the rate of belt payout from a given retractor has remained above a given level continuously for a given time, the sensor triggers.

   Type 3: The sensor triggers at a given time.

   Type 4: The sensor triggers when the distance between two nodes exceeds a given maximum or becomes less than a given minimum. This type of sensor is intended for use with an explicit mass/spring representation of the sensor mechanism.

   Type 5: The sensor triggers when the retractor's payout is out of the range specified by PULMX and PULMN.

   By default, the sensors are inactive during dynamic relaxation. This allows initial tightening of the belt and positioning of the occupant on the seat without locking the retractor or firing any pretensioners. However, a flag can be set in the sensor input to make the sensors active during the dynamic relaxation phase.

*ELEMENT_SEATBELT_SLIPRING

Purpose: Define seat belt slip ring.

Card 1	1	2	3	4	5	6	7	8
Variable	SBSRID	SBID1	SBID2	FC	SBRNID	LTIME	FCS	ONID
Type	I	I	I	F	I	F	F	I
Default	0	0	0	0.0	0	10^{20}	0.0	0

Optional Card.

Card 2	1	2	3	4	5	6	7	8
Variable	K	FUNCID	DIRECT	DC		LCNFFD	LCNFFS	
Type	F	I	I	F		I	I	
Default	0.0	0	0	0.0		0	0	

VARIABLE	DESCRIPTION
SBSRID	Slip ring ID. A unique number has to be used. See Remark 4 below for the treatment of slip rings for shell belt elements.
SBID1	Seat belt element 1 ID
SBID2	Seat belt element 2 ID
FC	Coulomb dynamic friction coefficient. If less than zero, |FC| refers to a curve which defines the dynamic friction coefficient as a function of time.
SBRNID	Slip ring node, NID
LTIME	Slip ring lockup time. After this time no material is moved from one side of the slip ring to the other. This option is not active during dynamic relaxation.

VARIABLE	DESCRIPTION
FCS	Optional Coulomb static friction coefficient. If less than zero, |FCS| refers to a curve which defines the static friction coefficient as a function of time.
ONID	Optional orientation node ID used to define the skew angle, α (see Figures 19-22 and 19-25). If ONID is undefined, the skew angle is assumed to be 0.0.
K	Optional coefficient for determining the Coulomb friction coefficient related to the skew angle, α (see Figure 19-25). See Remark 5.
FUNCID	Function ID to determine friction coefficient.
DIRECT	Direction of belt movement. To disable the belt slip behavior imposed by DIRECT, see the Remark "Seatbelt Slip Ring" under *SENSOR_CONTROL. EQ.0: If the belt can move along both directions. EQ.12: If the belt is only allowed to slip along the direction from SBID1 to SBID2. EQ.21: If the belt is only allowed to slip along the direction from SBID2 to SBID1.
DC	Optional decay constant to allow a smooth transition between the static and dynamic friction coefficients, that is, $\mu_c = \mathrm{FC} + (\mathrm{FCS} - \mathrm{FC})e^{-\mathrm{DC}\times\lvert v_{\mathrm{rel}}\rvert}$
LCNFFD	Optional curve for normal-force-dependent Coulomb dynamic friction coefficient. When defined, the dynamic friction coefficient becomes $\mathrm{FC} + f_{\mathrm{LCNFFD}}(F_n)$, where $f_{\mathrm{LCNFFD}}(F_n)$ is the function value of LCNFFD at contact force, F_n. The normal direction is defined as the average of the directions along the length of the seatbelt elements SBID1 and SBID2. The normal (or contact) force, F_n, is the sum of the projection of the tension forces on these elements, T_1 and T_2, onto the normal direction.
LCNFFS	Optional curve for normal-force-dependent Coulomb static friction coefficient. When defined, the static friction coefficient becomes $\mathrm{FCS} + f_{\mathrm{LCNFFS}}(F_n)$, where $f_{\mathrm{LCNFFS}}(F_n)$ is the function value of LCNFFS at contact force, F_n.

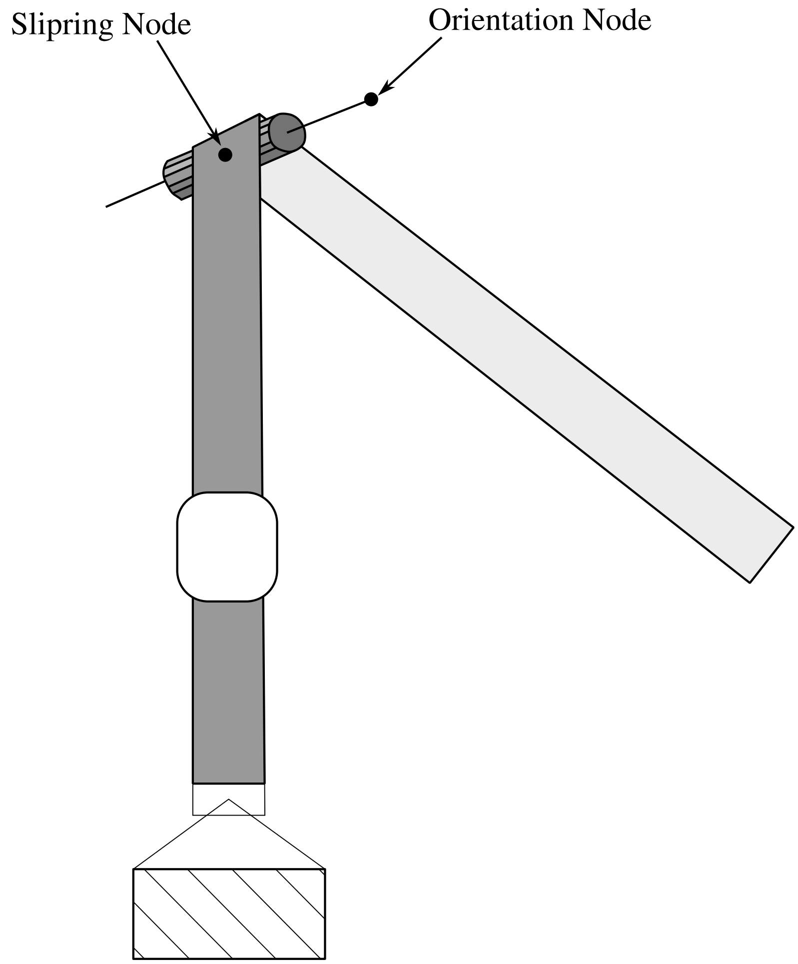


Figure 19-22. Orientation node.

Remarks:

1. **Slip Ring Model.** Slip rings allow continuous sliding of a belt through a sharp change of angle. To define a slip ring, the user identifies the two belt elements which meet at the slip ring (such as Elements 1 and 2 shown in Figure 19-23), the friction coefficient, and the slip ring node. The two elements must have a common node coincident with the slip ring node and should not be referenced by any other slip ring definition. No attempt should be made to restrain or constrain the common node because its motion will automatically be constrained to follow the slip ring node. Typically, the slip ring node is part of the vehicle body structure; therefore, belt elements should not be connected to this node directly, but any other feature can be attached, including rigid bodies.

   As shown in Figure 19-23, during slip Node B in the belt material remains attached to the slip ring node, but belt material (in the form of unstretched length) is passed from Element 1 to Element 2 to achieve slip. The amount of slip at each time step is calculated from the ratio of forces in Elements 1 and 2. The ratio of

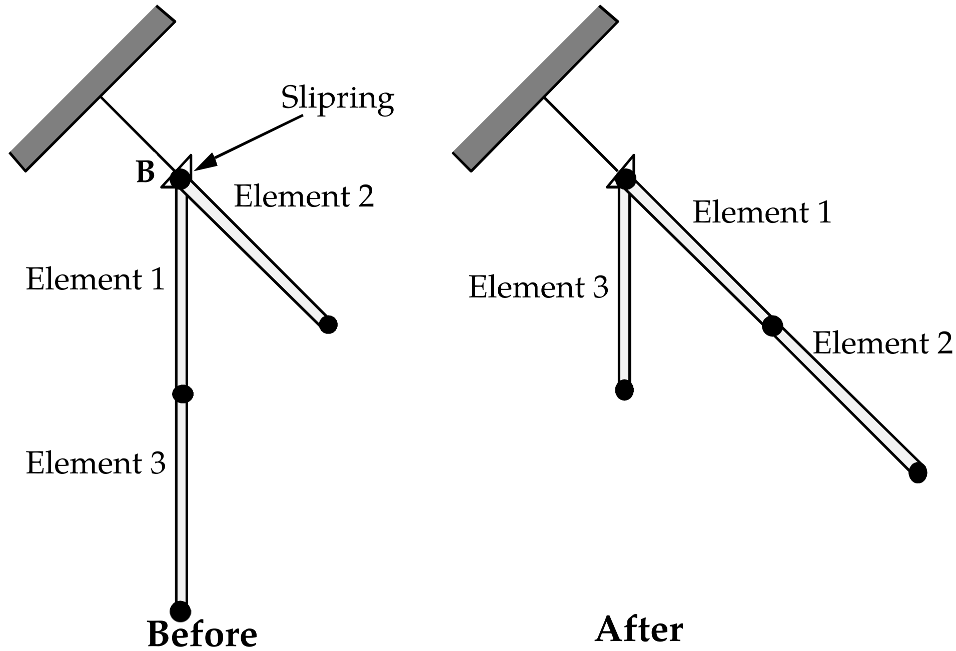


Figure 19-23. Elements passing through slipring.

forces is determined by the relative angle between the elements and the coefficient of friction, FC. The tension in the belts are T_1 and T_2, where T_2 is on the high tension side and T_1 is the force on the low tension side. Thus, if T_2 is sufficiently close to T_1, no slip occurs; otherwise, slip is just sufficient to reduce the ratio T_2/T_1 to $e^{\mathrm{FC}\times\theta}$, where θ is the wrap angle; see Figures 19-22 and 19-24. No slip occurs if both elements are slack. The out-of-balance force at Node B is reacted on the slip ring node; the motion of node B follows that of slip ring node.

2. **Elements Connected to Sliprings.** The unstretched length of a seatbelt element connected to a slipring during initialization *must be at least* 1.6 × LMIN where LMIN is a field defined in *MAT_SEATBELT. The unstretched length, l, of a seatbelt element is

$$l = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} + \mathrm{SLEN} \ ,$$

where (x_1, y_1, z_1) and (x_2, y_2, z_2) are the locations of the two nodes defining the seatbelt element and SLEN is the slack length (see *ELEMENT_SEATBELT). Any other element that may interact with the slipring *must have an unstretched length of at least* 1.1 × LMIN.

3. **Local Remeshing.** If, due to slip through the slip ring, the unstretched length of an element becomes less than the minimum length (as entered on the belt material card), the belt is remeshed locally: the short element passes through the slip ring and reappears on the other side (see Figure 19-23). The new unstretched length of element 1 is 1.1 × minimum length. Force and strain in elements 2 and 3 are unchanged; force and strain in element 1 are now equal to

those in element 2. Subsequent slip will pass material from element 3 to element 1. This process can continue with several elements passing in turn through the slip ring.

4. **Shell Elements.** When SBRNID < 0, this slip ring is for shell-type seatbelt; -SBRNID is the *SET_NODE containing SN1, SN2, …SN5. SBID1 and SBID2 are then *SET_SHELL_LIST. Note that the numbering of -SBRNID, SBID1 and SBID2 has to be consistent in the direction of numbering. For example, if *SET_-NODE for SBRNID has nodes (SN1, SN2, SN3, SN4, SN5) then *SET_SHELL_-LIST for SBID1 should have elements (SRE11, SRE12, SRE13, SRE14) and *SET_-SHELL_LIST for SBID2 should have elements (SRE21, SRE22, SRE23, SRE24). See Figure 19-22.

5. **Coulomb Friction.** If K is undefined, the limiting force ratio is taken as $e^{\mathrm{FC}\times\theta}$. If K is defined, the maximum force ratio is computed as

$$e^{\mathrm{FC}\times\theta(1+\mathrm{K}\times\alpha^2)}\ ,$$

where α is the angle shown in Figure 19-25. The function is defined using the *DEFINE_FUNCTION keyword input. This function is a function of three variables, and the ratio is given by evaluating

$$\frac{T_2}{T_1} = \mathrm{FUNC}(\mathrm{FCT}, \theta, \alpha)\ ,$$

where FCT is the instantaneous friction coefficient at time t, that is, it has the value of FC if the belt has moved in the last time-step and the value of FCS if the belt has been stationary. For example, the default behavior can be obtained using the function definition (assuming FCT has a value of 0.025 and the function ID is 1):

```
*DEFINE_FUNCTION
    1,
f(fct,theta,alpha) = exp(0.025*theta)
```

Behavior like the default option can be obtained with (K = 0.1):

```
*DEFINE_FUNCTION
    1,
f(fct,theta,alpha) = exp(0.025*theta*(1.+0.1*alpha*alpha))
```

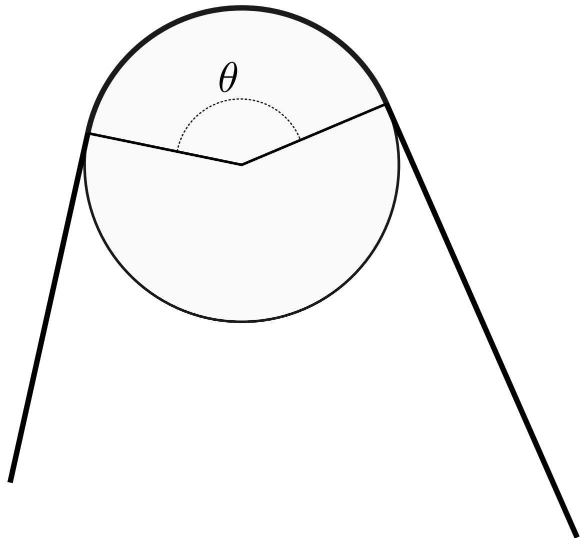


Figure 19-24. Front view showing wrap angle.

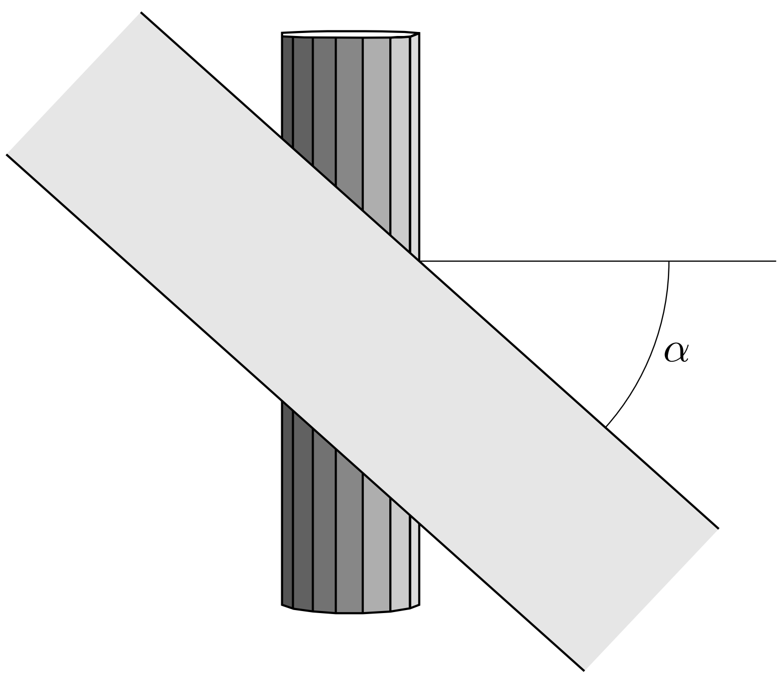


Figure 19-25. Top view shows orientation of belt relative to axis.

*ELEMENT_SHELL_{OPTION}

Available options include:

<BLANK>

THICKNESS

BETA or MCID

OFFSET

DOF

COMPOSITE

COMPOSITE_LONG

SHL4_TO_SHL8

Stacking of options, e.g., THICKNESS_OFFSET, is allowed in some cases. When combining options in this manner, check d3hsp to confirm that all the options are acknowledged.

Purpose: Define three, four, six, and eight node elements including 3D shells, membranes, 2D plane stress, plane strain, and axisymmetric solids. The type of the element and its formulation is specified through the part ID (see *PART) and the section ID (see *SECTION_SHELL). Also, the thickness of each element can be specified when applicable on the element cards or else a default thickness value is used from the section definition.

For orthotropic and anisotropic materials, a local material angle (variable BETA) can be defined which is cumulative with the integration point angles specified in *SECTION_SHELL, *PART_COMPOSITE, *ELEMENT_SHELL_COMPOSITE, or *ELEMENT_SHELL_COMPOSITE_LONG. Alternatively, the material coordinate system can be defined as the projection of a local coordinate system, MCID, onto the shell.

An offset option, OFFSET, is available for moving the shell reference surface from the nodal points that define the shell.

The COMPOSITE or COMPOSITE_LONG option allows an arbitrary number of integration points across the thickness of shells sharing the same part ID. This is independent of thickness defined in *SECTION_SHELL. To maintain a direct association of through-thickness integration point numbers with physical plies in the case where the number of plies varies from element to element, see Remark 12.

The option, SHL4_TO_SHL8, converts 3 node triangular and 4 node quadrilateral shell elements to 6 node triangular and 8 node quadrilateral quadratic shell elements, respectively, by the addition of mid-side nodal points. See Remark 9 below.

For the shell formulation that uses additional nodal degrees-of-freedom, the option DOF is available to connect the nodes of the shell to corresponding scalar nodes. Four scalar nodes are required for element type 25 to model the thickness changes that require 2 additional degrees-of-freedom per shell node. Defining these nodes is optional; if left undefined, they will be automatically created.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	0	0	0	0
Remarks			3	3	3	3				

Thickness Card. Additional card for THICKNESS, BETA, and MCID keyword options.

Card 2	1	2	3	4	5	6	7	8	9	10
Variable	THIC1		THIC2		THIC3		THIC4		BETA or MCID	
Type	F		F		F		F		F	
Default	0.		0.		0.		0.		0.	
Remarks	1								2	

Thickness Card. Additional card for THICKNESS, BETA, and MCID keyword options, is only required if mid-side nodes are defined (N5-N8).

Card 3	1	2	3	4	5	6	7	8	9	10
Variable	THIC5		THIC6		THIC7		THIC8			
Type	F		F		F		F			
Default	0.		0.		0.		0.		.	
Remarks	6									

Offset Card. Additional card for OFFSET keyword options.

Card 4	1	2	3	4	5	6	7	8	9	10
Variable	OFFSET									
Type	F									
Default	0.									
Remarks	7									

Scalar Node Card. Additional card for DOF keyword option.

Card 5	1	2	3	4	5	6	7	8	9	10
Variable			NS1	NS2	NS3	NS4				
Type			I	I	I	I				
Default										
Remarks			8	8	8	8				

COMPOSITE Cards. Additional set of cards for the COMPOSITE keyword option. Values of material ID, thickness, and material angle for each through-thickness integration point of a composite shell are defined using these cards (up to two integration points per card), and these values overrule values specified elsewhere. The integration point data should be given sequentially starting with the bottommost integration point. The total number of integration points is determined by the number of entries on these cards. The thickness of each shell is the summation of the integration point thicknesses. Define as many cards as needed to define all the through-thickness integration points. The fourth field must be zero or blank to be interpreted as a Card 6.

Card 6	1	2	3	4	5	6	7	8
Variable	MID1	THICK1	B1		MID2	THICK2	B2	
Type	I	F	F		I	F	F	

COMPOSITE_LONG Cards. Additional set of cards for the COMPOSITE_LONG keyword option. Values of material ID, thickness, and material angle for each through-thickness integration point of a composite shell are defined using these cards (one integration point per card), and these values overrule values specified elsewhere. The integration point data should be given sequentially starting with the bottommost integration point. The total number of integration points is determined by the number of entries on these cards. The thickness of each shell is the summation of the integration point thicknesses. Define as many cards as needed to define all the through-thickness integration points. The fourth field must be zero or blank to be interpreted as a Card 7.

Card 7	1	2	3	4	5	6	7	8
Variable	MID1	THICK1	B1		PLYID1			
Type	I	F	F		I			

VARIABLE	DESCRIPTION
EID	Element ID. Chose a unique number with respect to other elements.
PID	Part ID, see *PART.
N1	Nodal point 1
N2	Nodal point 2
N3	Nodal point 3

VARIABLE	DESCRIPTION
N4	Nodal point 4
N5 - N8	Mid-side nodes for eight node shell
THIC1	Shell thickness at node 1
THIC2	Shell thickness at node 2
THIC3	Shell thickness at node 3
THIC4	Shell thickness at node 4
BETA	Orthotropic material base offset angle (see Remarks 5 and 6 below). The angle is given in degrees. If blank, the default is set to zero.
MCID	Material coordinate system ID. The **a**-axis of the base (or element-level) material coordinate system is the projection of the x-axis of coordinate system MCID onto the surface of the shell element. The **c**-axis of the material coordinate system aligns with the shell normal. The **b**-axis is taken as $\mathbf{b} = \mathbf{c} \times \mathbf{a}$. Each layer in the element can have a unique material orientation by defining a rotation angle for each layer as described in Remark 5.
THIC5	Shell thickness at node 5
THIC6	Shell thickness at node 6
THIC7	Shell thickness at node 7
THIC8	Shell thickness at node 8
OFFSET	The offset distance from the plane of the nodal points to the reference surface of the shell in the direction of the normal vector to the shell.
NS1	Scalar node 1, parameter NDOF on the *NODE_SCALAR is normally set to 2. If the thickness is constrained, set NDOF = 0.
NS2	Scalar node 2
NS3	Scalar node 3
NS4	Scalar node 4
MID*i*	Material ID of integration point i, see *MAT_… Section.

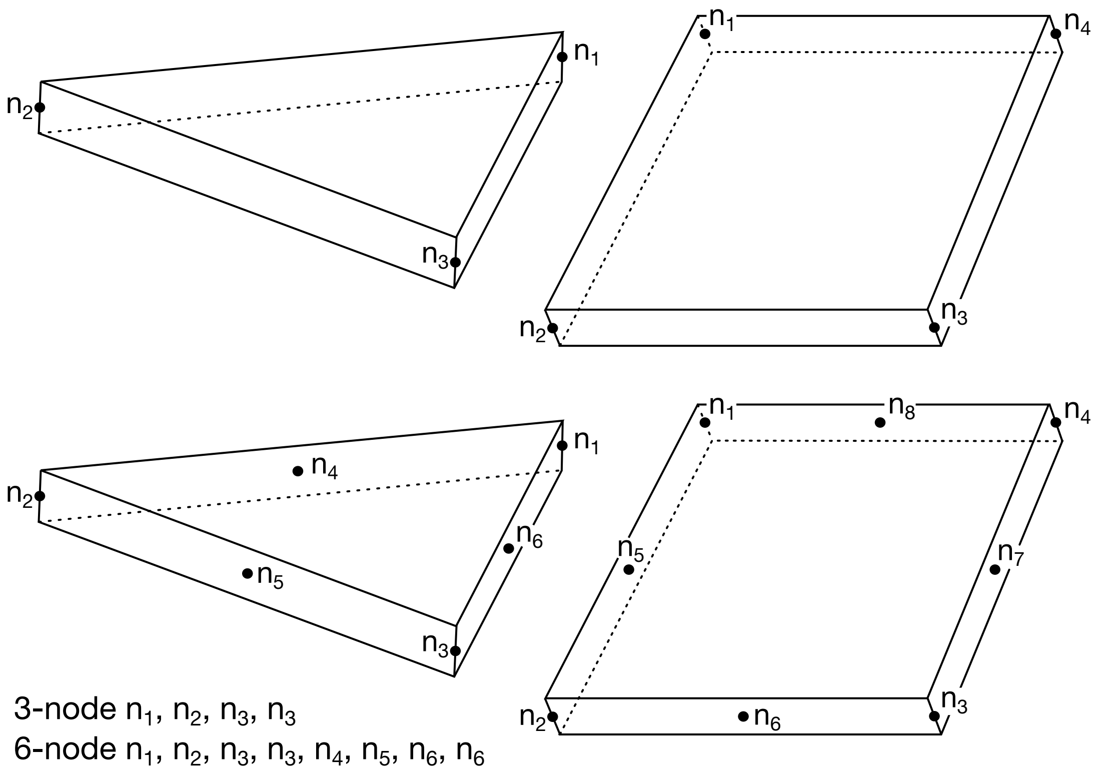


Figure 19-26. LS-DYNA shell elements. Counterclockwise node numbering determines the top surface.

VARIABLE	DESCRIPTION
THICK*i*	Thickness of integration point *i*.
B*i*	Material angle of integration point *i*.
PLYID*i*	Ply ID for integration point *i* (for post-processing purposes).

Remarks:

1. **Default thickness.** Default values in place of zero shell thicknesses are taken from the cross-section property definition of the PID; see *SECTION_SHELL.

2. **Ordering.** Counterclockwise node numbering determines the top surface, see Figure 19-26

3. **Coordinate systems.** Stresses and strain output in the binary databases are given in the global coordinate system, whereas stress resultants are output in the local coordinate system for the shell element.

4. **Convexity.** Interior angles must be less than 180 degrees.

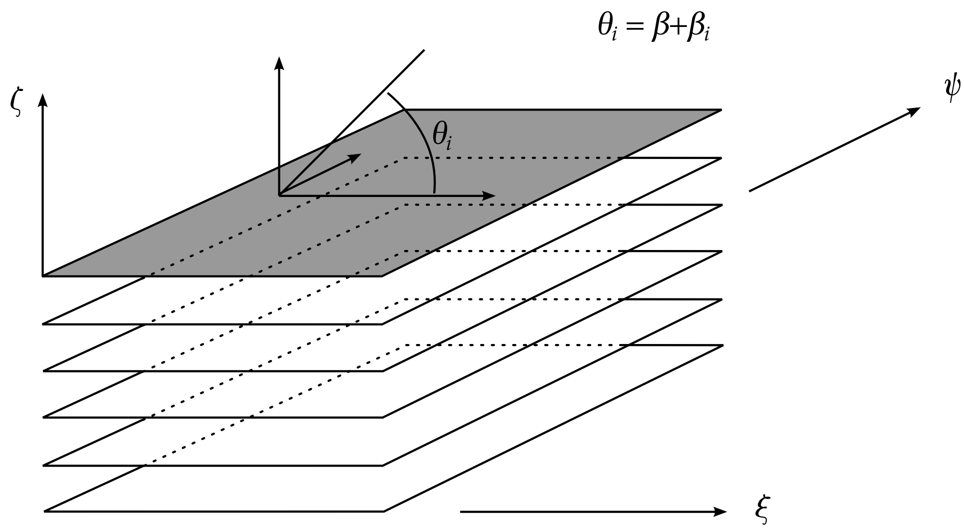


Figure 19-27. A multi-layer laminate can be defined. The angle β_i is defined for the i^{th} lamina (integration point), see *SECTION_SHELL.

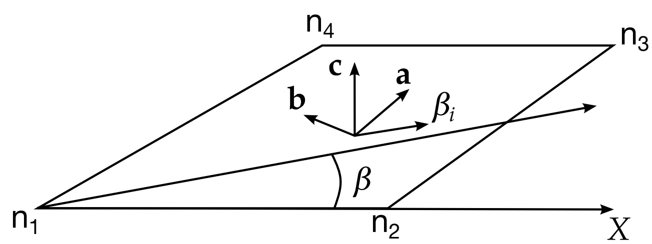


Figure 19-28. Orientation of material directions (shown relative to the 1-2 side as when AOPT = 0 in *MAT).

5. **Material orientation.** To allow the orientation of orthotropic and anisotropic materials to be defined for each shell element, a BETA angle can be defined. This BETA angle is used with the AOPT parameter and associated data on the *MAT card to determine an element reference direction for the element. The AOPT data defines a coordinate system and the BETA angle defines a subsequent rotation about the element normal to determine the element reference system. For composite modeling, each layer in the element can have a unique material direction by defining an additional rotation angle for the layer, using either the ICOMP and B*i* parameters on *SECTION_SHELL or the B*i* parameter on *PART_COMPOSITE. The material direction for layer *i* is then determined by a rotation angle, θ_i as shown in Figures 19-27 and 19-28.

6. **Activation of the BETA field and its interpretation.** To activate the BETA field, either the BETA or THICKNESS keyword option must be used. There is a difference in how a zero value or empty field is interpreted. When the BETA keyword option is used, a zero value or empty BETA field will override the BETA on *MAT. However, when the THICKNESS keyword option is used, a zero value or empty BETA field will not override the BETA value on *MAT.

Therefore, to input BETA = 0, the BETA keyword option is recommended. If a THIC value is omitted or input as zero, the thickness will default to the value on the *SECTION_SHELL card. If mid-side nodes are defined (N5 - N8), then a second line of thickness values will be read. This line may be left blank, but cannot be omitted.

7. **Offset for the reference surface.** The parameter OFFSET gives the offset from the nodal points of the shell to the reference surface. This option applies to most shell formulations excluding two-dimensional elements, membrane elements, and quadratic shell elements. Except for Mortar contacts, the reference surface offset given by OFFSET is not taken into account in the contact subroutines unless CNTCO is set to 1 in *CONTROL_SHELL. For Mortar contacts the OFFSET determines the location of the contact surface.

8. **Scalar nodes.** The scalar nodes specified on the optional card refer to the scalar nodes defined by the user to hold additional degrees of freedom for shells with this capability. Scalar nodes are used with shell element type 25 and 26.

9. **Automatic order increase.** The option, SHL4_TO_SHL8, converts 3 node triangular and 4 node quadrilateral shell elements to 6 node triangular and 8 node quadrilateral quadratic shell elements, respectively, by the addition of mid-side nodal points. The user node ID's for these generated nodes are offset after the largest user node ID defined in the input file. When defining the *SECTION_-SHELL keyword, the element type must be specified as either 23 or 24 corresponding to quadratic quadrilateral and triangular shells, respectively.

10. **Cohesive elements.** Cohesive elements (ELFORM = 29, 46 or 47 on *SECTION_SHELL) may be defined with zero depth to connect surfaces with no gap between them, but must have nodes 1 and 2 on one surface and nodes 3 and 4 on the other.

11. **Contact thickness when using *ELEMENT_SHELL_THICKNESS in MPP.** For MPP, THEORY = 1 in *CONTROL_SHELL has a special meaning when dealing with non-uniform-thickness shells: it serves to set the nodal contact thickness equal to the average of the nodal thicknesses from the shells sharing that node. Thus, when non-uniform-thickness shells comprise a contact surface, we recommend THEORY = 1 and setting the actual shell formulation with ELFORM in *SECTION_SHELL. (This remark does not apply to segment based (SOFT=2) contact wherein each contact segment is considered to be of uniform thickness.)

12. **Assignment of zero thickness to integration points.** The ability to assign zero- thickness integration points in the stacking sequence allows the number of integration points to remain constant even as the number of physical plies varies from element to element and eases post-processing since a particular integration point corresponds to a physical ply. Such a capability is important when one or more of the physical plies are not continuous across a part. To represent a

missing ply in *ELEMENT_SHELL_COMPOSITE, set THICK*i* to 0.0 for the corresponding integration point and additionally, either set MID = -1 or, if the LONG option is used, set PLYID to any nonzero value

When postprocessing the results using LS-PrePost version 4.5, read both the keyword deck and d3plot database into the code and then select Option → N/A gray fringe. Then, when viewing fringe plots for a particular integration point (FriComp → IPt → intpt#), the element will be grayed out if the selected integration point is missing (or has zero thickness) in that element.

13. **PID with COMPOSITE keyword option.** When using the COMPOSITE keyword option, the PID on Card 1 must reference a *PART card that does not also use the COMPOSITE keyword option.

ELEMENT_SHELL_NURBS_PATCH_*{OPTION1}_{OPTION2}

Available options for *OPTION1* include:

<BLANK>

TRIMMED

Available options for *OPTION2* include:

<BLANK>

TITLE

Purpose: Define a NURBS surface patch using two univariate knot vectors, a rectangular grid of control points, and optionally a set of control weights. The two knot vectors define the necessary shape functions and parameterize the surface by virtue of the tensor product scheme.

The *r*- and *s*-direction knot vectors have lengths of NPR+PR+1 and NPS+PS+1, respectively. NPR and NPS are the number of control points in each direction while PR and PS define the polynomial degree in each direction. The control grid consists of NPR × NPS control points. A control weight may be defined for each control point. There is no limit on the size of a NURBS surface patch.

The total number of necessary data cards depends on the parameters given in Cards 1 and 2, that is,

$$\#\text{ of cards } = 2 + \left\lceil \frac{\text{NPR} + \text{PR} + 1}{8} \right\rceil + \left\lceil \frac{\text{NPS} + \text{PS} + 1}{8} \right\rceil + (\text{WFL} + 1) \times \text{NPS} \times \left\lceil \frac{\text{NPR}}{8} \right\rceil ,$$

where $\lceil x \rceil = \text{ceil}(x)$. The flag WFL (see Card 2 below) indicates whether the control weights are user-defined. An example NURBS surface and the partial keyword deck using this keyword is given in Figures 19-29 and 19-30, respectively.

A NURBS surface patch may be trimmed using the TRIMMED option. This requires the definition of additional trimming loop(s) which can be defined using Cards E and F. Note that optional cards used to define trimming loops are not counted in the above expression. See Remark 6.

A description of the NURBS surface patch may be provided using the option TITLE which is also written to the d3hsp file.

Card Summary:

Card Title. This card is included if and only if the TITLE keyword option is used.

TITLE

Card 1. This card is required.

NPID	PID	NPR	PR	NPS	PS		

Card 2. This card is required.

WFL	ELFORM	INT	NISR	NISS	IMASS		IDFNE

Cards A. Include $\text{ceil}[(\text{NPR} + \text{PR} + 1)/8]$ of this card.

RK1	RK2	RK3	RK4	RK5	RK6	RK7	RK8

Cards B. Include $\text{ceil}[(\text{NPS} + \text{PS} + 1)/8]$ of this card.

SK1	SK2	SK3	SK4	SK5	SK6	SK7	SK8

Cards C. Include $\text{NPS} \times \text{ceil}(\text{NPR}/8)$ of this card.

N1	N2	N3	N4	N5	N6	N7	N8

Cards D. Include $\text{NPS} \times \text{ceil}(\text{NPR}/8)$ of this card if $\text{WFL} = 1$.

W1	W2	W3	W4	W5	W6	W7	W8

Cards E. This card is included if the keyword option TRIMMED is used.

TITLE

Cards F. This card is included if the keyword option TRIMMED is used.

C1	C2	C3	C4	C5	C6	C7	C8

Data Cards:

Title Card. Additional card for the TITLE keyword option.

Card Title	1	2	3	4	5	6	7	8
Variable	TITLE							
Type	A80							

VARIABLE	DESCRIPTION
TITLE	Description of the NURBS patch surface

Card 1	1	2	3	4	5	6	7	8
Variable	NPID	PID	NPR	PR	NPS	PS		
Type	I	I	I	I	I	I		
Default	none	none	None	none	none	none		

VARIABLE	DESCRIPTION
NPID	NURBS surface element / patch ID. A unique number must be chosen.
PID	PART ID. See *PART.
NPR	Number of control points in the local *r*-direction.
PR	Polynomial degree of the basis function in the local *r*-direction.
NPS	Number of control points in the local *s*-direction.
PS	Polynomial degree of the basis function in the local *s*-direction.

Card 2	1	2	3	4	5	6	7	8
Variable	WFL	ELFORM	INT	NISR	NISS	IMASS		IDFNE
Type	I	I	I	F	F	I		I
Default	0	0	0	PR	PS	0		0

VARIABLE	DESCRIPTION
WFL	Flag for user defined control weights: EQ.0: Control weights are assumed to be uniform and positive; that is, the surface is a B-spline surface. No optional Cards D is allowed. EQ.1: Control weights are defined using optional Cards D.
ELFORM	Shell formulation to be used (see Remark 3): EQ.0: Reissner-Mindlin with fibers at the control points EQ.1: Kirchhoff-Love with fibers at the control points (not recommended) EQ.2: Kirchhoff-Love with fibers at the integration points EQ.3: Reissner-Mindlin with fibers at the integration points EQ.-4/4: Combination of FORM = 0 and FORM = 1. See Remark 4.
INT	In-plane numerical integration rule: EQ.0: Uniformly reduced Gauss integration. NIP = PR × PS. EQ.1: Full Gauss integration. NIP = (PR + 1) × (PS + 1). EQ.2: Reduced, patch-wise integration rule for C^1-continuous quadratic NURBS surfaces.
NISR	Number or average edge length of automatically created interpolation shell elements per each knot span in the r-direction. See Remark 5 and Figure 19-30. GT.0: NINT(NISR) is the number of interpolation elements in the r-direction LT.0: |NISR| is the average edge length of the interpolation

VARIABLE	DESCRIPTION
	elements in the r-direction.
NISS	Number or average edge length of automatically created interpolation shell elements per each knot span in the s-direction. See Remark 5 and Figure 19-30. GT.0: NINT(NISS) is the number of interpolation elements in the s-direction LT.0: |NISS| is the average edge length of the interpolation elements in the s-direction
IMASS	Mass matrix lumping scheme: EQ.0: Row sum. EQ.1: Diagonal weighting.
IDFNE	Element ID of first NURBS-Element within this NURBS-Patch definition.

Knot Vector Cards (for r-direction). The knot vector in r-direction with length NPR + PR + 1 is given below requiring a total of ceil[(NPR + PR + 1)/8] cards.

Cards A	1	2	3	4	5	6	7	8
Variable	RK1	RK2	RK3	RK4	RK5	RK6	RK7	RK8
Type	F	F	F	F	F	F	F	F
Default	none	none	None	none	none	none	none	none

VARIABLE	DESCRIPTION
RKm	Values of the univariate knot vector in r-direction

Knot Vector Cards (for *s*-direction). The knot vector in *s*-direction with length NPS + PS + 1 is given below requiring a total of ceil[(NPS + PS + 1)/8] cards.

Cards B	1	2	3	4	5	6	7	8
Variable	SK1	SK2	SK3	SK4	SK5	SK6	SK7	SK8
Type	F	F	F	F	F	F	F	F
Default	none	none	None	none	none	none	none	none

VARIABLE	DESCRIPTION
SK*n*	Values of the univariate knot vector in *s*-direction

Connectivity Cards. The connectivity of the control grid is a two dimensional table of NPS rows and NPR columns. This data fills the NPS sets (one set for each row) of NPR points tightly packed into ceil(NPR/8) connectivity cards, for a total of NPS × ceil(NPR/8) cards.

Cards C	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
N*k*	Control point IDs, defined via *NODE, to define the control grid LT.0: Control point with rotational DOFs for FORM = 4 /-4; see Remark 4.

Control Weight Cards (Optional). Additional cards are used to set a weight for each control point if WFL = 1 on Card 2. These cards have an ordering identical to the connectivity cards (Cards C).

Cards D	1	2	3	4	5	6	7	8
Variable	W1	W2	W3	W4	W5	W6	W7	W8
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
Wk	Control weights of the surface patch

Trimming Loop Title Card.

Cards E	1	2	3	4	5	6	7	8
Variable	TITLE							
Type	A80							
Default	trimming loop							

Trimming Loop Connectivity Cards. Define a trimming loop. See Remark 6.

Cards F	1	2	3	4	5	6	7	8
Variable	C1	C2	C3	C4	C5	C6	C7	C8
Type	I	I	I	I	I	I	I	I

VARIABLE	DESCRIPTION
TITLE	Title of the trimming loop
Cl	Trimming curve ID pointing to a curve defined using *DEFINE_-NURBS_CURVE. A unique number has to be chosen.

Remarks:

1. **Shell Thickness.** The thickness of the shell is defined in *SECTION_SHELL and referenced via *PART.

2. **Element Formulation.** ELFORM = 201 must be used in *SECTION_SHELL.

3. **Thin Shell Formulation.** For thin IGA shells we do not recommend using FORM = 1. Instead we recommend using FORM = 2.

4. **Rotational Degrees of Freedom.** FORM = 4 allows the mixture of control points with and without rotational DOFs. This might be useful at the boundaries of NURBS patches where the continuity usually drops to C^0 and rotational DOFs are necessary. To indicate control points with rotational DOFs (6 DOFs/control point), the node number of the corresponding control point has to be set as the negative node ID in the connectivity Cards C. Positive node IDs indicate control points without rotational DOFs (3 DOFs/control point).

   If FORM = -4 is used, the control points at the patch boundary are automatically treated with rotational DOFs without the need to specify them explicitly in the connectivity cards (Cards C). This might be sufficient in many cases.

5. **Interpolation Elements.** The post-processing and the treatment of contact boundary conditions are presently dealt with interpolation elements, defined via interpolation nodes. These nodes and elements are automatically created, see Figure 19-30, where NISR and NISS indicate the number or average length of interpolation elements to be created per NURBS-element in the r- and s-directions, respectively.

6. **Trimming Loops.** Trimmed NURBS surface elements / patches can be analyzed by defining trimming loops. A trimming loop is formed by a set of NURBS curves defined in the surface parametric coordinate system. Each trimming curve is defined using the *DEFINE_NURBS_CURVE keyword.

   Trimming loops may be given a distinct title on Cards E and the connectivity of trimming curves defining a loop is stored on Cards F. The end and starting points of two consecutive curves *must* coincide. If the loop is defined by a single curve, the starting and end points of the curve *must* match. Furthermore, the orientation of the trimming loop is essential to define the trimmed surface. Travelling along the trimming loop, the surface on the right-hand side of the loop will be trimmed. There is no limitation on either the number of trimming curves forming a loop or the number of loops used to trim a NURBS surface element/patch.

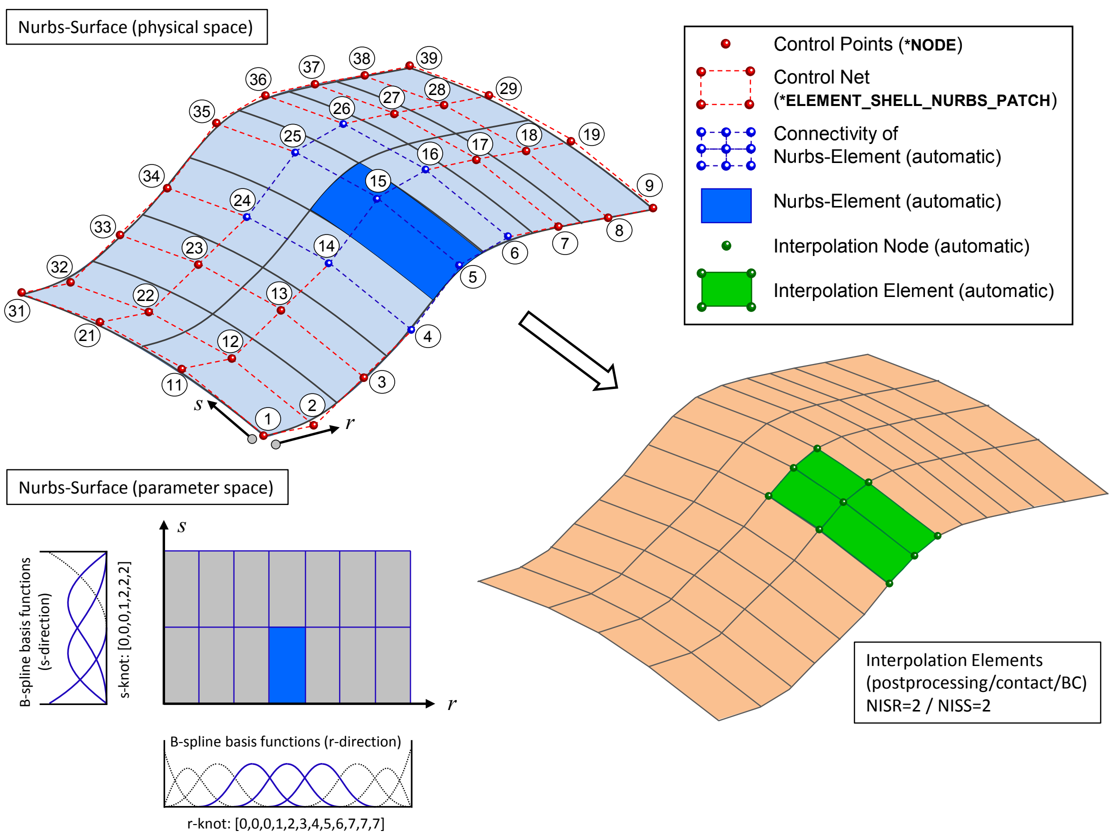


Figure 19-29. An example biquadratic NURBS surface patch and its interpolation mesh.

```
*ELEMENT_SHELL_NURBS_PATCH
$ Card 1
$---+-NPID----+--PID----+--NPR----+---PR----+--NPS----+---PS----+----7----+----8
        11        12         9         2         4         2
$ Card 2
$---+--WFL----+-FORM----+--INT----+-NISR----+-NISS----+IMASS----+----7----+----8
         1         0         1         2         2         0
$ Cards A
$rk-+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
       0.0       0.0       0.0       1.0       2.0       3.0       4.0       5.0
       6.0       7.0       7.0       7.0
$ Cards B
$sk-+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
       0.0       0.0       0.0       1.0       2.0       2.0       2.0
$ Cards C
$net+---N1----+---N2----+---N3----+---N4----+---N5----+---N6----+---N7----+---N8
         1         2         3         4         5         6         7         8
         9
        11        12        13        14        15        16        17        18
        19
        21        22        23        24        25        26        27        28
        29
        31        32        33        34        35        36        37        38
        39
$ Cards D (optional if WFL.ne.0)
$wgt+---W1----+---W2----+---W3----+---W4----+---W5----+---W6----+---W7----+---W8
       1.0       0.9       0.8       0.7       0.8       0.9       0.7       0.8
       1.0
       0.8       0.7       0.6       0.5       0.6       0.7       0.6       0.7
       0.8
       0.7       0.6       0.5       0.4       0.5       0.6       0.5       0.6
       0.7
       1.0       0.9       0.8       0.7       0.8       0.9       0.7       0.8
       1.0
```

Figure 19-30. Excerpt of the *ELEMENT_SHELL_NURBS_PATCH keyword defining the biquadratic NURBS surface patch in Figure 19-29.

*ELEMENT_SHELL_SOURCE_SINK

Purpose: Define a strip of shell elements of a single part ID to simulate a continuous forming operation. This option requires logical regular meshing of rectangular elements, which implies that the number of nodal points across the strip is constant along the length. Elements are created at the source and disappear at the sink. The advantage of this approach is that it is not necessary to define an enormous number of elements to simulate a continuous forming operation. Currently, only one source-sink definition is allowed. The boundary conditions at the source are discrete nodal point forces to keep the work piece in tension. At the sink, displacement boundary conditions are applied.

Card	1	2	3	4	5	6	7	8
Variable	NSSR	NSSK	PID					
Type	I	I	I					
Default	none	none	none					

VARIABLE	DESCRIPTION
NSSR	Node set at source. Provide an ordered set of nodes between corner nodes, which include the corner nodes.
NSSK	Node set at sink. Provide an ordered set of nodes between corner nodes, which include the corner nodes.
PID	Part ID of work piece.

ELEMENT_SOLID_*{OPTION}

Available options include:

<BLANK>

ORTHO

DOF

TET4TOTET10

H20

H8TOH20

H27

H64

H8TOH27

H8TOH64

P21

P40

T15

T20

Stacking of options, such as ORTHO_DOF, is allowed in some cases. When combining options in this manner, check d3hsp to confirm that all options are acknowledged.

Purpose: Define three-dimensional solid elements. The type of solid element and its formulation is specified through the part ID (see *PART) and the section ID (see *SECTION_SOLID_*OPTION*). Also, a local coordinate system for orthotropic and anisotropic materials can be defined by using the ORTHO option. If extra degrees of freedom are needed, the DOF option should be used. The option TET4TOTET10 converts 4 node tetrahedrons to 10 node tetrahedrons, and H8TOH20/H8TOH27 converts 8-node hexahedrons to 20-node/27-node hexahedrons. The option H8TOH64 converts elements with linear basis functions to elements with cubic basis functions. See Remark 1.

Card Summary:

Card Sets. Include as many of the following sets of cards matching the keyword options as desired. This input ends with the next keyword ("*") card.

Card 1. This card is required.

EID	PID								

Card 2. This card is required.

N1	N2	N3	N4	N5	N6	N7	N8	N9	N10

Card 3. Include as many of this card as needed to specify the nodes of the 15-node tetrahedron, 20-node tetrahedron, 20-node hexahedron, 21-node pentahedron, 40-node pentahedron, 27-node hexahedron, and 64-node hexahedron.

N11	N12	N13	N14	N15	N16	N17	N18	N19	N20

Card 4. Include this card for the ORTHO keyword option.

A1/BETA	A2	A3		

Card 5. Include this card for the ORTHO keyword option.

D1	D2	D3		

Card 6. Include this card for the DOF keyword option.

		NS1	NS2	NS3	NS4	NS5	NS6	NS7	NS8

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID								
Type	I	I								
Default	none	none								
Remarks	2									

VARIABLE	DESCRIPTION
EID	Element ID. A unique number must be chosen.
PID	Part ID, see *PART.

Card 2	1	2	3	4	5	6	7	8	9	10
Variable	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10
Type	I	I	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none	none	none

Additional Cards for Nodes. Additional cards as needed for the 15-node tetrahedron, 20-node tetrahedron and hexahedron, 21-node and 40 node pentahedron, and the 27-node and 64-node hexahedron.

Card 3	1	2	3	4	5	6	7	8	9	10
Variable	N11	N12	N13	N14	N15	N16	N17	N18	N19	N20
Type	I	I	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
N1	Nodal point 1
N2	Nodal point 2
N3	Nodal point 3
⋮	⋮
N64	Nodal point 64

Orthotropic Card 1. Additional card for ORTHO keyword option.

Card 4	1	2	3	4	5	6	7	8	9	10
Variable	A1 or BETA		A2		A3					
Type	F		F		F					
Default	0.		0.		0.					
Remarks	5									

Orthotropic Card 2. Second additional card for ORTHO keyword option.

Card 5	1	2	3	4	5	6	7	8	9	10
Variable	D1		D2		D3					
Type	F		F		F					
Default	0.		0.		0.					
Remarks	5									

VARIABLE	DESCRIPTION
A1 or BETA	x-component of local material direction **a**, or else rotation angle BETA in degrees (see Remark 5)
A2	y-component of local material direction **a**
A3	z-component of local material direction **a**
D1	x-component of vector in the plane of the material vectors **a** and **b**
D2	y-component of vector in the plane of the material vectors **a** and **b**
D3	z-component of vector in the plane of the material vectors **a** and **b**

Scalar Node Card. Additional card for DOF keyword option.

Card 6	1	2	3	4	5	6	7	8	9	10
Variable			NS1	NS2	NS3	NS4	NS5	NS6	NS7	NS8
Type			I	I	I	I	I	I	I	I
Default			none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
NS1	Scalar node 1. See Remark 7.
NS2	Scalar node 2
⋮	⋮
NS8	Scalar node 8

Remarks:

1. **Automatic Node Generation**. The option TET4TOTET10 automatically converts 4 node tetrahedral solids to 10 node quadratic tetrahedral solids. Additional mid-side nodes are created which are shared by all tetrahedral elements that contain the edge. The user node ID's for these generated nodes are offset after the largest user node ID defined in the input file. When defining the *SECTION_SOLID keyword, the element type must be specified as either 16 or 17 which are the 10-noded tetrahedrons in LS-DYNA. Mid-side nodes created as a result of TET4TOTET10 will not be automatically added to node sets that include the nodes of the original tetrahedron. So, for example, if the tetrahedrons are to have an initial velocity, velocity initialization by part ID or part set ID using *INITIAL_VELOCITY_GENERATION is necessary as opposed to velocity initialization by node set ID using *INITIAL_VELOCITY. The option H8TOH20/H8TOH27 provides the same functionality for converting 8-node to 20-node/27-node elements. The option H8TOH64 converts 8-node to 64-node hexahedra, 4-nodel to 20-node tetraheda, and 6-node to 40-node pentahedra. Mid-side nodes and face nodes created as a result of H8TOH20/H8TO27/H8TO64 will automatically be added to node sets that include the nodes of the original hexahedron.

2. **Node Numbering.** Four, six, and eight node elements are shown in Figure 19-31 where the ordering of the nodal points is shown. 27-node elements are shown

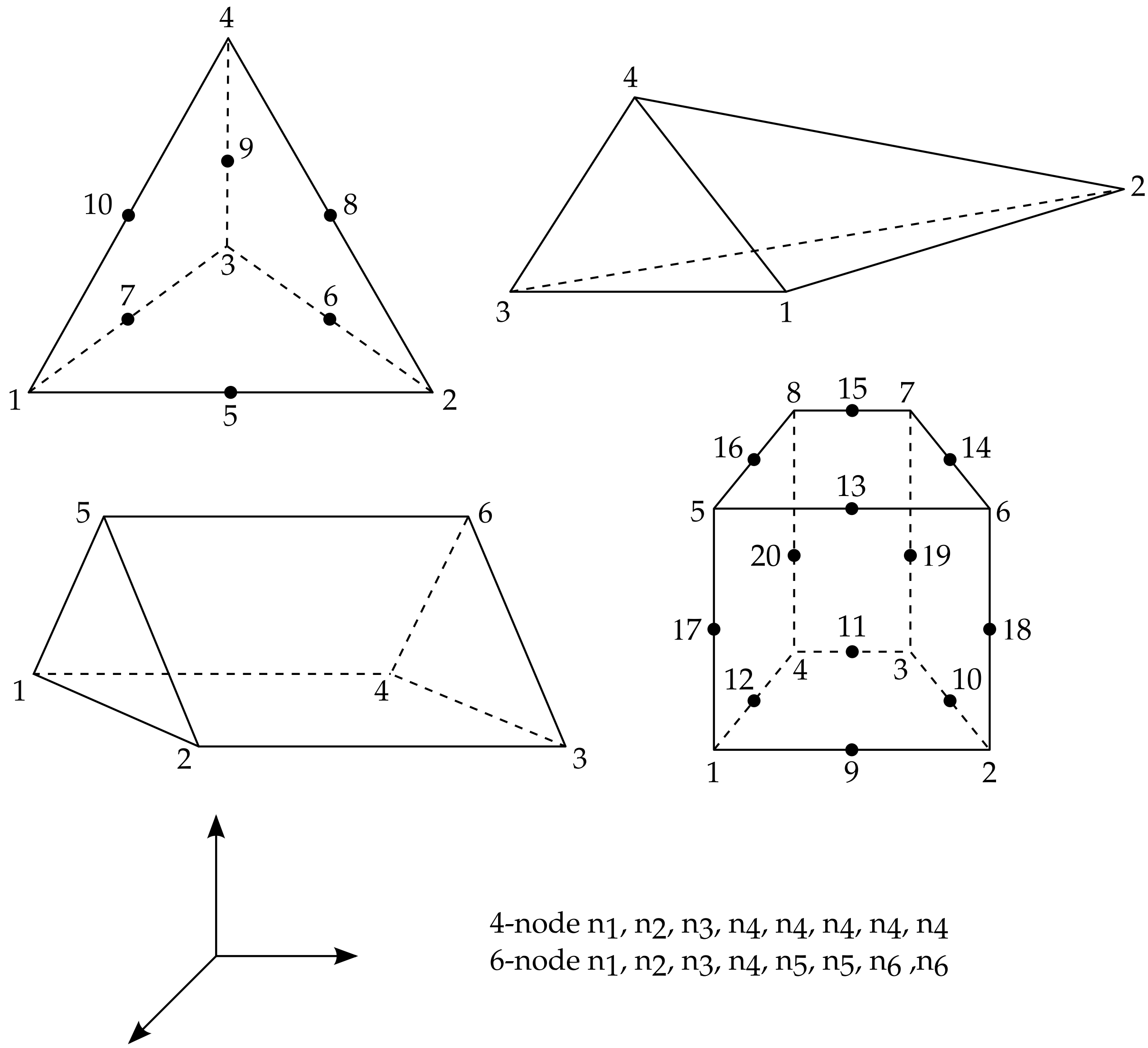


Figure 19-31. Four, six, eight, ten, and twenty node solid elements. For the hexahedral and pentahedral shapes, nodes 1-4 are on the bottom surface.

in Figure 19-32. This ordering must be followed or code termination with occur during the initialization phase with a negative volume message. The input of nodes on the element cards for the tetrahedral and pentahedral elements is given by:

<u>4-noded tetrahedron</u>	N1, N2, N3, N4, N4, N4, N4, N4, 0, 0
<u>6-noded pentahedron</u>	N1, N2, N3, N4, N5, N5, N6, N6, 0, 0

To visualize the node numbering for higher order elements, that is, options H27, H64, P21, P40, T15, and T20, see hexahedron 27, hexahedron 64, pentahedron 21, pentahedron 40, tetrahedron 15, and tetrahedron 20.

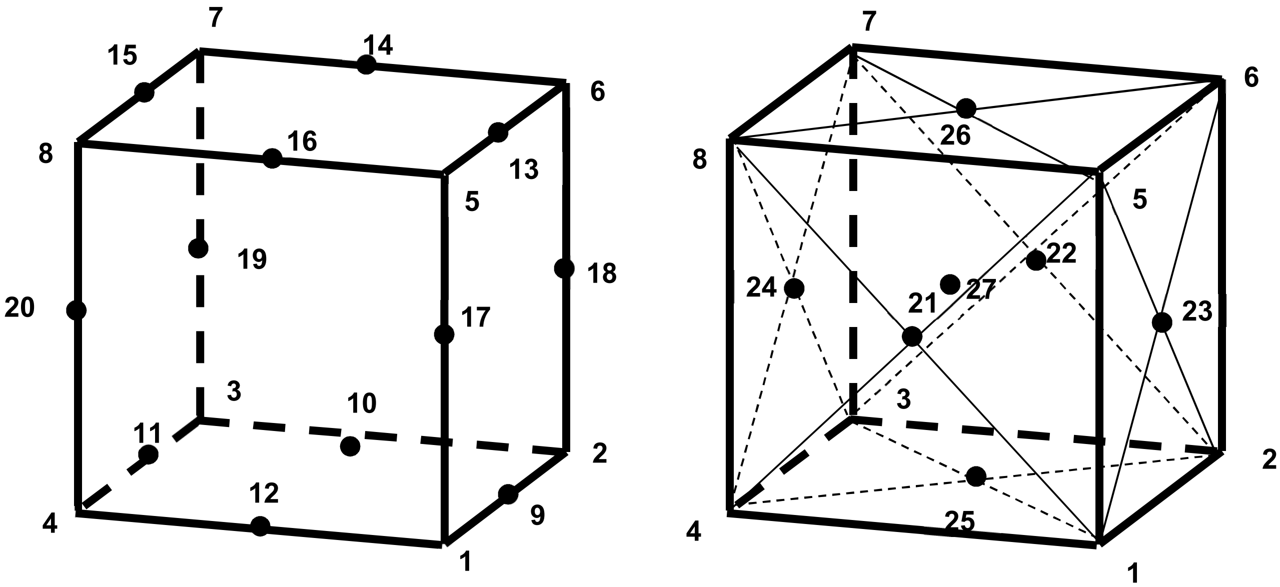


Figure 19-32. 27-node solid element.

3. **Degenerate Solids.** If hexahedrons are mixed with tetrahedrons and pentahedrons in the input under the same part ID, *degenerate* tetrahedrons and pentahedrons are used. One problem with degenerate elements is related to an uneven mass distribution (node 4 of the tetrahedron has five times the mass of nodes 1-3) which can make these elements somewhat unstable with the default time step size. By using the control flag under the keyword, *CONTROL_SOLID, automatic sorting can be invoked to treat the degenerate elements as type 10 and type 15 tetrahedral and pentahedral elements, respectively. The elements with cubic basis functions do not support degeneration.

4. **Obsolete Card Format.** For elements with 4-8 nodes the cards in the format of LS-DYNA versions 940-970 are still supported. The older format does not include Card 2.

Obsolete Element Solid Card.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I	I	I

5. **Local Directions.** For the orthotropic and anisotropic material models the local directions may be defined on the second card following the element connectivity definition. The local directions are then computed from the two vectors such that (see Figure 19-33):

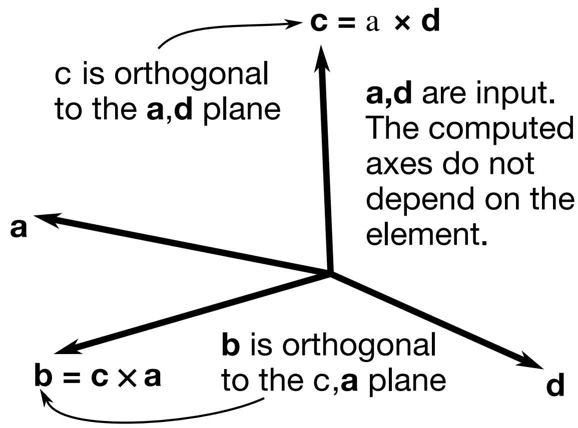


Figure 19-33. Two vectors **a** and **d** are defined and the triad is computed and stored.

$$\mathbf{c} = \mathbf{a} \times \mathbf{d} \text{ and } \mathbf{b} = \mathbf{c} \times \mathbf{a}.$$

These vectors are internally normalized within LS-DYNA. In this case, $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ are not calculated from AOPT and associated parameters for this element. If the material model uses AOPT = 3, the *a* and *b*-axes will be rotated about the *c*-axis by the angle BETA specified on the material card.

If vector **d** is input as a zero length vector, then A1 is interpreted as an offset rotation angle BETA in degrees. BETA describes a rotation about the **c**-axis of the **a**-**b**-**c** coordinate system that is found through AOPT and associated parameters on the *MAT input. Note that the axes may be switched before or after the application of BETA depending on the value of MACF in the material keyword input.. This BETA angle applies to all values of AOPT, and it overrides the BETA angle on the *MAT card in the case of AOPT = 3.

6. **Stress Output Coordinates.** Stress output for solid elements is in the global coordinate system by default.

7. **Optional "Scalar" Nodes.** The scalar nodes specified on Card 6 refer to extra nodes used by certain features (usually user defined) to store additional degrees of freedom.

*ELEMENT_SOLID_NURBS_PATCH

Purpose: Define a NURBS-block element (patch) based on a cuboid grid of control points. This grid consists of NPR*NPS*NPT control points, where NPR, NPS and NPT are the number of control points in local r-, s- and t-directions, respectively. The necessary shape functions are defined through three knot-vectors:

1. Knot-Vector in r-direction with length NPR + PR + 1

2. Knot-Vector in s-direction with length NPS + PS + 1

3. Knot-Vector in t-direction with length NPT + PT + 1

There is no limit on the size of the underlying grid to define a NURBS-block element, so the total number of necessary cards depends on the parameters given in the first two cards and is given by

$$\begin{aligned}\#\text{ of cards } &= \\ & 2 + \left\lceil \frac{\text{NPR} + \text{PR} + 1}{8} \right\rceil + \left\lceil \frac{\text{NPS} + \text{PS} + 1}{8} \right\rceil + \\ & \left\lceil \frac{\text{NPT} + \text{PT} + 1}{8} \right\rceil + \text{NPT} \times \text{NPS} \times \left\lceil \frac{\text{NPR}}{8} \right\rceil\end{aligned}$$

where $\lceil x \rceil = \text{ceil}(x)$. (NOTE: the last term in the sum is doubled if WFL = 1, indicating that the weights are user-specified).

Card Summary:

Card 1. This card is required.

NPID	PID	NPR	PR	NPS	PS	NPT	PT

Card 2. This card is required.

WFL	NISR	NISS	NIST	IMASS	IINT		IDFNE

Card 3. Include ceil[(NPR +PR + 1)/8] of this card to define the knot vector in the local r-direction.

RK1	RK2	RK3	RK4	RK5	RK6	RK7	RK8

Card 4. Include ceil[(NPS +PS + 1)/8] of this card to define the knot vector in the local s-direction.

SK1	SK2	SK3	SK4	SK5	SK6	SK7	SK8

Card 5. Include ceil[(NPT+PT + 1)/8] of this card to define the knot vector in the local *t*-direction.

TK1	TK2	TK3	TK4	TK5	TK6	TK7	TK8

Card 6. The connectivity of the control grid is a two-dimensional table of NPT × NPS rows and NPR columns. This data fills NPT × NPS sets (one set for each row) of NPR points tightly packed into ceil(NPR/8) Connectivity Cards, for a total of NPT × NPS × ceil(NPR/8) cards.

N1	N2	N3	N4	N5	N6	N7	N8

Card 7. Additional card for WFL ≠ 0. Set a weight for each control point. These cards have an ordering identical to the Connectivity Cards (Card 6).

W1	W2	W3	W4	W5	W6	W7	W8

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	NPID	PID	NPR	PR	NPS	PS	NPT	PT
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
NPID	NURBS-Patch Element ID. A unique number has to be chosen
PID	Part ID, see *PART.
NPR	Number of control points in local *r*-direction.
PR	Order of polynomial of univariate NURBS basis functions in local *r*-direction.
NPS	Number of control points in local *s*-direction.
PS	Order of polynomial of univariate NURBS basis functions in local *s*-direction.

VARIABLE	DESCRIPTION
NPT	Number of control points in local t-direction.
PT	Order of polynomial of univariate NURBS basis functions in local t-direction.

Card 2	1	2	3	4	5	6	7	8
Variable	WFL	NISR	NISS	NIST	IMASS	IINT		IDFNE
Type	I	I	I	I	I	I		I
Default	0	PR	PS	PT	0	0		0

VARIABLE	DESCRIPTION
WFL	Flag for weighting factors of the control points EQ.0: all weights at the control points are set to 1.0 (B-spline basis) and no optional Card 7 sets are allowed NE.0: the weights at the control points are defined with optional Card 7 which must be defined after sets of Card 6.
NISR	Number of (automatically created) Interpolation Solid elements in local r-direction per created NURBS-element for visualization (post-processing) and contact.
NISS	Number of (automatically created) Interpolation Solid elements in local s-direction per created NURBS-element for visualization (post-processing) and contact.
NIST	Number of (automatically created) Interpolation Solid elements in local t-direction per created NURBS-element for visualization (postprocessing) and contact.
IMASS	Option for lumping of mass matrix: EQ.0: row sum EQ.1: diagonal weighting.
IINT	Numerical integration rule:

VARIABLE	DESCRIPTION
	EQ.0: uniformly reduced Gauss integration. NIP = PR × PS × PT.
	EQ.1: full Gauss integration. NIP = (PR+1) × (PS+1) × (PT+1).
	EQ.2: reduced, patch-wise integration rule for C1-continuous quadratic NURBS surfaces.
	EQ.3: non-uniformly reduced Gauss integration. Full integration on the boundary elements and reduced integration on the interior elements.
IDFNE	Element ID of first NURBS-Element within this NURBS-Patch definition.

Knot Vector Cards (for *r*-direction). The knot-vector in local *r*-direction with length $\mathrm{NPR} + \mathrm{PR} + 1$ is given below (up to eight values per card) requiring a total of $\mathrm{ceil}[\,(\mathrm{NPR} + \mathrm{PR} + 1)/8\,]$ cards.

Card 3	1	2	3	4	5	6	7	8
Variable	RK1	RK2	RK3	RK4	RK5	RK6	RK7	RK8
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Knot Vector Cards (for *s*-direction). The knot-vector in local *s*-direction with length $\mathrm{NPS} + \mathrm{PS} + 1$ is given below (up to eight values per card) requiring a total of $\mathrm{ceil}[\,(\mathrm{NPS} + \mathrm{PS} + 1)/8\,]$ cards.

Card 4	1	2	3	4	5	6	7	8
Variable	SK1	SK2	SK3	SK4	SK5	SK6	SK7	SK8
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Knot Vector Cards (for t-direction). The knot-vector in local t-direction with length NPT + PT + 1 is given below (up to eight values per card) requiring a total of ceil[(NPT+PT + 1)/8] cards.

Card 5	1	2	3	4	5	6	7	8
Variable	TK1	TK2	TK3	TK4	TK5	TK6	TK7	TK8
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
RK*i*	Values of the univariate knot vector in local r-direction
SK*i*	Values of the univariate knot vector in local s-direction
TK*i*	Values of the univariate knot vector in local t-direction

Connectivity Cards. The connectivity of the control grid is a two-dimensional table of NPT × NPS rows and NPR columns. This data fills NPT × NPS sets (one *set* for each row) of NPR points tightly packed into ceil(NPR/8) Connectivity Cards, for a total of NPT × NPS × ceil(NPR/8) cards.

Card 6	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
N*i*	Control points *i* (defined via *NODE) to define the control grid

Weight Cards. Additional card for WFL ≠ 0. Set a weight for each control point. These cards have an ordering identical to the Connectivity Cards (Card 6).

Card 7	1	2	3	4	5	6	7	8
Variable	W1	W2	W3	W4	W5	W6	W7	W8
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
Wi	Weighting factors of control point i

Remarks:

1. **ELFORM.** ELFORM = 201 has to be used in *SECTION_SOLID.

2. **Interpolation Elements.** The post-processing and the treatment of contact boundary conditions are presently dealt with using interpolation elements, defined via interpolation nodes. These nodes and elements are automatically created, where NISR, NISS and NIST indicate the number of interpolation elements to be created per NURBS-element in the local r-, s- and t-direction, respectively.

Example:

```
$...|....1....|....2....|....3....|....4....|....5....|....6....|....7....|....8
$ An Isogeometric Solid NURBS Example :
$...|....1....|....2....|....3....|....4....|....5....|....6....|....7....|....8
*SECTION_SOLID
$#   secid    elform
         1       201
*ELEMENT_SOLID_NURBS_PATCH
$CARD 1
$#   npeid       pid       npr        pr       nps        ps       npt        pt
         1         1         3         2         6         2         3         2
$CARD 2
$#     wfl      nisr      niss      nist     imass
         0         2         2         2         0
$CARD A
$#     rk1       rk2       rk3       rk4       rk5       rk6       rk7       rk8
       0.0       0.0       0.0       1.0       1.0       1.0
$CARD B
       0.0       0.0       0.0      0.25       0.5      0.75       1.0       1.0
```

```
       1.0
$CARD C
       0.0       0.0       0.0       1.0       1.0       1.0
$CARD D
      1001      1002      1003
       …
      1052      1053      1054
$CARD E (Optional if wfl .eq.  0)
```

*ELEMENT_SOLID_PERI

Purpose: Define the connectivity of four node surface elements for Peridynamics laminate parts.

Include this card for each element. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	EID	PID	N1	N2	N3	N4		
Type	I	I	I	I	I	I		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
EID	Element ID. A unique number must be chosen.
PID	Part ID
N*i*	Nodal point *i*

Remarks:

1. **Discretizing the laminate.** To discretize a laminate, first, construct a layer of 3D surface mesh with 4 node elements, such as shell elements, according to the geometric shape of the laminate. Then, copy this layer of mesh and move it to the middle surface of a lamina. Continue copying and moving the layer of mesh until each lamina has one layer of mesh. See Figure 19-34.

2. **Coincident nodes.** To represent in-plane material split, Peridynamics laminate elements must not share nodes. As a result, shared corners will have coincident nodes, one for each element. In other words, the total number of nodes is 4 × the total number of Peridynamics laminate elements.

3. **MPP.** We recommend for computational efficiency decomposing the model for MPP such that corresponding elements in each layer are on the same processor. To ensure this decomposition, we suggest specifying box regions that have the lumped grouping in the pfile (see Appendix O) around the corresponding elements. Figure 19-34 illustrates this decomposition for four processors.

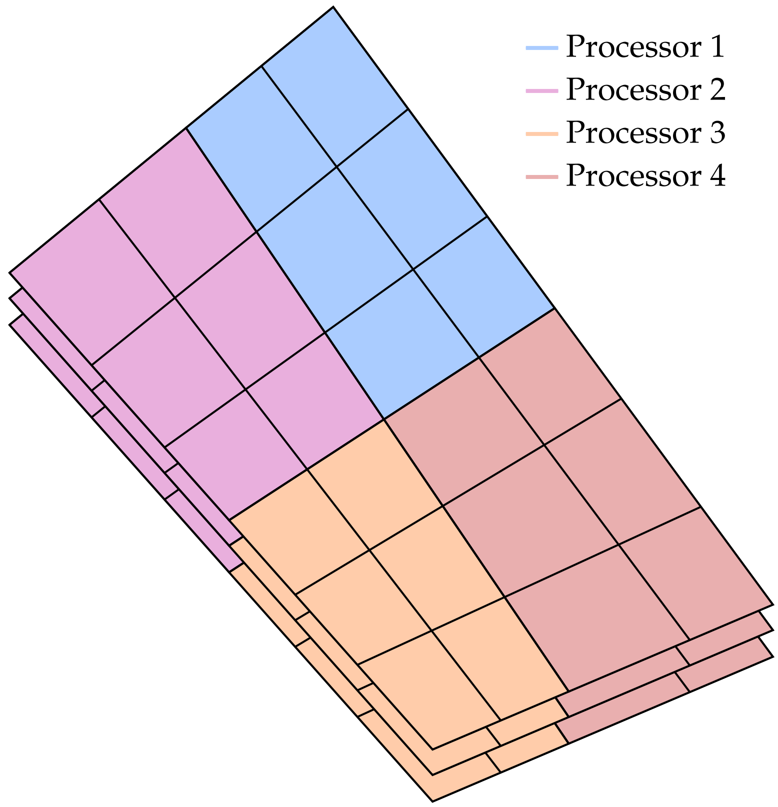


Figure 19-34. Simple example of the mesh for a three layer laminate. Each color represents an example decomposition of the mesh for MPP.

*ELEMENT_SOLID_REFERENCE

Purpose: Define a lower-order (hexahedral, tetrahedral, or pentahedral) or higher-order (10-noded tetrahedral) reference element. This keyword is a generalization of *INITIAL_FOAM_REFERENCE_GEOMETRY in the sense that the connectivity of the reference geometry is independent of the one for the current geometry. *NODE_REFERENCE specifies the corresponding reference coordinates for *ELEMENT_SOLID_REFERENCE. Furthermore, IREF on the *MAT cards does not need to be specified for this keyword. If a reference element is present, LS-DYNA automatically uses it for computing the required quantities, in particular deformation gradients. This keyword is primarily used in the processing of dynain and dynain.lsda files (see *INTERFACE_SPRINGBACK_LS-DYNA) and not explicitly introduced by the user. *INTERFACE_SPRINGBACK_LSDYNA causes LS-DYNA to automatically write it for the materials that require reference information.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID									
Type	I									
Default	none									

Card 2	1	2	3	4	5	6	7	8	9	10
Variable	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10
Type	I	I	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
EID	Element ID. This ID must correspond to the same ID that is used on the associated *ELEMENT_SOLID card.
N*i*	Node number in the connectivity, referring to the corresponding ID in *NODE_REFERENCE.

*ELEMENT_SPH_{OPTION}

Available options include:

<BLANK>

VOLUME

Purpose: Define a lumped mass element assigned to a nodal point.

If the VOLUME option is used, the field for MASS is treated as particle volume. It has the same effect as giving a negative number in the MASS field.

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	NID	PID	MASS		NEND					
Type	I	I	F		I					
Default	none	none	0.		0					

VARIABLE	DESCRIPTION
NID	Node ID and Element ID are the same for the SPH option.
PID	Part ID to which this node (element) belongs
MASS	Mass/volume (see Remark 1): GT.0.0: Mass value LT.0.0: Volume. The absolute value will be used as volume. The density, ρ, will be retrieved from the material card defined in PID. SPH element mass is calculated by $\lvert\mathrm{MASS}\rvert \times \rho$.
NEND	Optional input: GT.0: *ELEMENT_SPH cards are generated between NID to NEND using current PID and MASS data.

Remarks:

1. **Axisymmetric SPH.** Axisymmetric SPH (IDIM = -2 in *CONTROL_SPH) is defined on the global XY-plane with the Y-axis as the axis of rotation. An

axisymmetric SPH element has a mass of ρA, where ρ is its density and A is the area of the SPH element. A can be approximated by the area of its corresponding axisymmetric shell element (see Figure 19-35). The mass printout in the d3hsp file is the mass per radian, that is, $\rho A x_i$. See Figure 19-36.

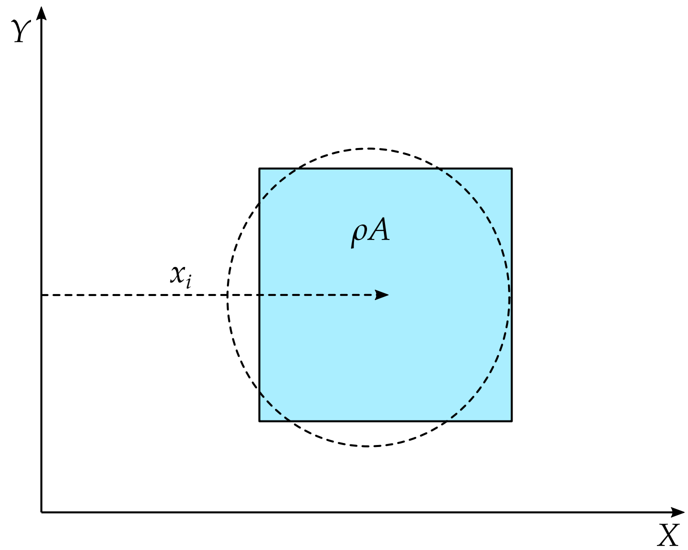


Figure 19-35. Schematic of axisymmetric SPH cross section

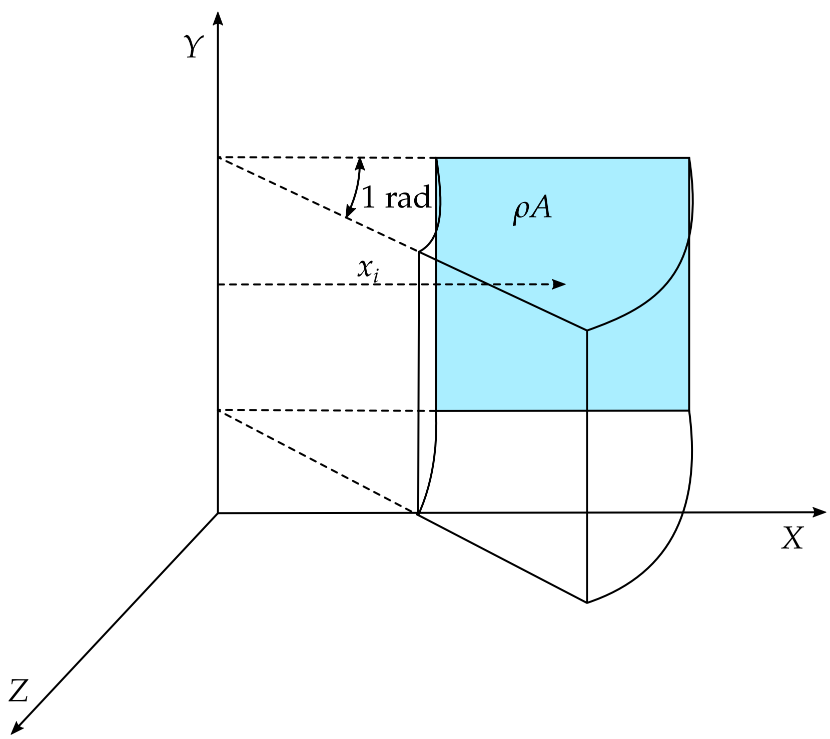


Figure 19-36. Mass printout in d3hsp

*ELEMENT_TRIM

> **NOTE:** This keyword was replaced by *CONTROL_FORMING_TRIMMING starting in Revision 87566.

*ELEMENT_TSHELL_{OPTION}

Available options include:

<BLANK>

BETA

COMPOSITE

Purpose: Define an eight node thick shell element which is available with either fully reduced or selectively reduced integration rules. Thick shell formulations 1, 2, and 6 are plane stress elements that can be used as an alternative to the 4 node shell elements in cases where an 8-node element is desired. Thick shell formulations 3, 5 and 7 are layered solids with 3D stress updates. Formulation 5, 6, and 7 are based on an enhanced strain. The number of through-thickness integration points is specified by the user.

For orthotropic and anisotropic materials, a local material angle (variable BETA) can be defined which is cumulative with the integration point angles specified in *SECTION_-TSHELL or *PART_COMPOSITE_TSHELL.

The COMPOSITE option for *ELEMENT_TSHELL allows a unique stack of integration points for each element sharing the same part ID, and is available only when combined with *PART. The COMPOSITE option is not available in combination with *PART_COMPOSITE_TSHELL. To maintain a direct association of through-thickness integration point numbers with physical plies in the case where the number of plies varies from element to element, see Remark 5.

Card Summary:

Card 1. This card is required.

EID	PID	N1	N2	N3	N4	N5	N6	N7	N8

Card 2a. This card is included if the BETA keyword option is used.

			BETA

Card 2b. This card is included if the COMPOSITE keyword option is used.

MID1	THICK1	B1		MID2	THICK2	B2	

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8	9	10
Variable	EID	PID	N1	N2	N3	N4	N5	N6	N7	N8
Type	I	I	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none	none	none
Remarks			1							

VARIABLE	DESCRIPTION
EID	Element ID. Unique numbers have to be used.
PID	Part ID, see *PART.
N*i*	Node *i*

Beta Card. Additional card for the BETA keyword option.

Card 2a	1	2	3	4	5	6	7	8	9	10
Variable									BETA	
Type									F	
Default									0.	
Remarks									4	

VARIABLE	DESCRIPTION
BETA	Orthotropic material base offset angle (see Remark 4). The angle is given in degrees. If blank, the default is set to zero.

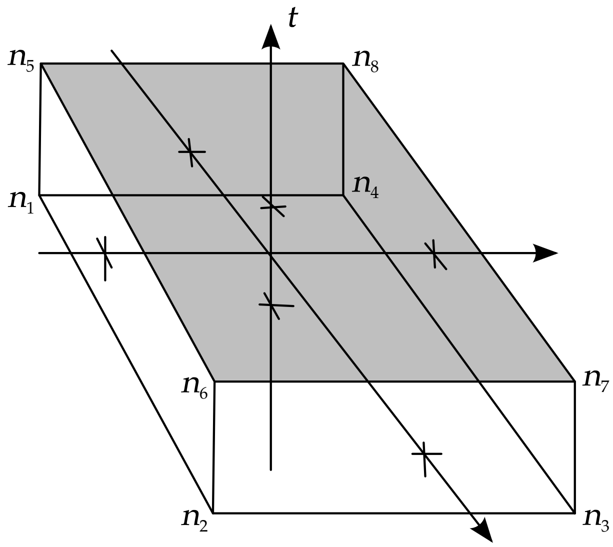


Figure 19-37. 8-node Thick Shell Element.

Composite Cards. Additional card for the COMPOSITE keyword option. Values of material ID, thickness, and material angle for each through-thickness integration point of a composite shell are defined using these cards, and these values override values specified elsewhere. The integration point data should be given sequentially starting with the bottommost integration point. The total number of integration points is determined by the number of entries on these cards. The total thickness is the distance between the top and bottom surface as determined by the element connectivity, so the THICK*i* values are scaled to fit the element. Define as many cards as needed to define all the through-thickness integration points. The fourth field must be zero or blank to be interpreted as a Card 2b.

Card 2b	1	2	3	4	5	6	7	8
Variable	MID1	THICK1	B1		MID2	THICK2	B2	
Type	I	F	F		I	F	F	

VARIABLE	DESCRIPTION
MID*i*	Material ID of integration point *i*, see the *MAT_… cards.
THICK*i*	Thickness of integration point *i*
B*i*	Material angle of integration point *i*

Remarks:

1. **Orientation.** Extreme care must be used in defining the connectivity to ensure proper orientation of the through-thickness direction. For a hexahedron, nodes n_1 to n_4 define the lower surface, and nodes n_5 to n_8 define the upper surface. For a pentahedron, nodes n_1, n_2, n_3 form the lower triangular surface and the eight variables N1 to N8 should be defined using nodes n_1, n_2, n_3, n_3, n_4, n_5, n_6, n_6, respectively. Note that node n_3 and node n_6 are each repeated.

2. **Integration.** Element formulations 1 and 5 (see *SECTION_TSHELL), use one point integration and the integration points then lie along the t-axis as shown in Figure 19-37. Element formulations 2 and 3 use two by two selective reduced integration in each layer.

3. **Stress Output.** The stresses for thick shell elements are output in the global coordinate system.

4. **Local Coordinate System.** To allow the orientation of orthotropic and anisotropic materials to be defined for each thick shell element, a beta angle can be defined. This beta angle is used with the AOPT parameter and associated data on the *MAT card to determine an element reference direction for the element.

   The AOPT data defines a coordinate system and the BETA angle defines a subsequent rotation about the element normal to determine the element reference system. For composite modeling, each layer in the element can have a unique material direction by defining an additional rotation angle for the layer, using either the ICOMP and B*i* fields on *SECTION_TSHELL or the B*i* parameter on *PART_COMPOSITE_TSHELL. The material direction for layer *i* is then determined by a rotation angle, θ_i.

5. **Assignment of Zero Thickness to Integration Points.** The ability to assign zero- thickness integration points in the stacking sequence allows the number of integration points to remain constant even as the number of physical plies varies from element to element and eases post-processing since a particular integration point corresponds to a physical ply. Such a capability is important when one or more of the physical plies are not continuous across a part. To represent a missing ply in *ELEMENT_TSHELL_COMPOSITE, set THICK*i* to 0.0 for the corresponding integration point and additionally, set MID to -1.

   When post-processing the results using LS-PrePost version 4.5, read both the keyword deck and d3plot database into the code and then select Option→N/A gray fringe. Then, when viewing fringe plots for a particular integration point (FriComp→IPt →intpt#), the element will be grayed out if the selected integration point is missing (or has zero thickness) in that element.

6. **PID with COMPOSITE keyword option.** When using the COMPOSITE keyword option, the PID on Card 1 must reference a *PART card that does not also use the COMPOSITE keyword option.