

***BOUNDARY**

The keyword ***BOUNDARY** provides a way of defining imposed motions on boundary nodes. The keyword control cards in this section are defined in alphabetical order:

- *BOUNDARY_ACOUSTIC_COUPLING_{OPTION}**
- *BOUNDARY_ACOUSTIC_COUPLING_SPECTRAL**
- *BOUNDARY_ACOUSTIC_FREE_SURFACE**
- *BOUNDARY_ACOUSTIC_IMPEDANCE**
- *BOUNDARY_ACOUSTIC_IMPEDANCE_COMPLEX**
- *BOUNDARY_ACOUSTIC_IMPEDANCE_MECHANICAL**
- *BOUNDARY_ACOUSTIC_INTERFACE**
- *BOUNDARY_ACOUSTIC_MAPPING**
- *BOUNDARY_ACOUSTIC_NON_REFLECTING**
- *BOUNDARY_ACOUSTIC_PRESCRIBED_MOTION**
- *BOUNDARY_ACOUSTIC_PRESSURE_SPECTRAL**
- *BOUNDARY_ALE_MAPPING**
- *BOUNDARY_AMBIENT**
- *BOUNDARY_AMBIENT_EOS**
- *BOUNDARY_CONVECTION_OPTION**
- *BOUNDARY_COUPLED**
- *BOUNDARY_CYCLIC**
- *BOUNDARY_DE_NON_REFLECTING**
- *BOUNDARY_FLUIDM**
- *BOUNDARY_FLUIDM_BOTTOM**
- *BOUNDARY_FLUIDM_FREE_SURFACE**
- *BOUNDARY_FLUIDM_INTERIOR**

***BOUNDARY**

*BOUNDARY_FLUX_OPTION
*BOUNDARY_FLUX_TRAJECTORY
*BOUNDARY_MCOL
*BOUNDARY_NON_REFLECTING
*BOUNDARY_NON_REFLECTING_2D
*BOUNDARY_PAP
*BOUNDARY_PORE_FLUID_OPTION
*BOUNDARY_PRECRACK
*BOUNDARY_PRESCRIBED_ACCELEROMETER_RIGID
*BOUNDARY_PRESCRIBED_FINAL_GEOMETRY
*BOUNDARY_PRESCRIBED_MOTION_OPTION
*BOUNDARY_PRESCRIBED_ORIENTATION_RIGID_OPTION
*BOUNDARY_PRESSURE_OUTFLOW_OPTION
*BOUNDARY_PWP_OPTION
*BOUNDARY_PZEPOT
*BOUNDARY_RADIATION_OPTION
*BOUNDARY_RADIATION_ENCLOSURE
*BOUNDARY_SALE_MESH_FACE
*BOUNDARY_SLIDING_PLANE
*BOUNDARY_SPC_OPTION
*BOUNDARY_SPC_SYMMETRY_PLANE
*BOUNDARY_SPH_FLOW
*BOUNDARY_SPH_NON_REFLECTING
*BOUNDARY_SPH_NOSLIP
*BOUNDARY_SPH_SYMMETRY_PLANE
*BOUNDARY_SYMMETRY_FAILURE

- *BOUNDARY_TEMPERATURE_OPTION
- *BOUNDARY_TEMPERATURE_PERIODIC_SET
- *BOUNDARY_TEMPERATURE_RSW
- *BOUNDARY_TEMPERATURE_TRAJECTORY
- *BOUNDARY_THERMAL_BULKFLOW
- *BOUNDARY_THERMAL_BULKNODE
- *BOUNDARY_THERMAL_WELD
- *BOUNDARY_THERMAL_WELD_TRAJECTORY
- *BOUNDARY_USA_SURFACE

***BOUNDARY_ACOUSTIC_COUPLING_{OPTION}**

Purpose: This keyword provides fluid-structure coupling when the fluid is modeled with *MAT_ACOUSTIC and solid formulation 8 or 14, *and* the nodes of the fluid at the fluid-structure interface are not merged/shared with nodes of the structure.

If acoustic fluid exists on one side of a structural surface *and* the nodes of the fluid and structure are merged/shared at the interface, this keyword is not required. Coupling between an acoustic fluid and a structural surface occurs automatically when nodes are shared at the interface.

Available options for the keyword are:

<BLANK>

MISMATCH

If the nodes of the fluid are not coincident with the nodes of the structure at the fluid-structure interface, the MISMATCH option must be used. If the nodes are coincident, the <BLANK> option is sufficient. The input for this keyword is a segment set that defines the structural surface(s) to be coupled to the acoustic fluid.

If acoustic fluid exists on both sides of a structural surface comprised of shells, that is, the shell surface constitutes a physical break between the two fluid masses (call these fluid A and fluid B), fluid A should not share nodes with fluid B. In this case, coupling may be defined (1) wholly through use of this keyword if the structural shells do not share nodes with either fluid, or (2) by fluid A sharing nodes with the structural shells and by using this keyword to couple fluid B to the structural shells. Note that this concept of coupling two fluid masses to a structural surface does not apply to structural solids because a solid face can have direct contact with only one fluid.

If no keyword option is used (<BLANK>), then the nodes of the structural element segments must be coincident with, but not merged to, the nodes of the acoustic element faces. Each structural element face to be coupled, regardless of whether there is fluid on one or both sides of that face, requires only one corresponding segment to be included in the segment set, and the orientation of the segment normal is unimportant.

The MISMATCH option permits the coupling of acoustic fluid elements to a structural surface when the meshes of the fluid and structural elements are moderately mismatched, meaning the nodes of the fluid and structure at the interface are not necessarily coincident. The segments in the segment set should define the structural surface and, following the right-hand rule, the normal vector for the segments should point toward the fluid volume elements with which coupling is intended. If fluid-structure coupling is required on both sides of structural shell elements, duplicate segments with opposite normal vectors should be defined. Every segment in the segment set must couple with the fluid volume at some integration point, but not all integration points on the segment need to

couple with the fluid. The meshes do not have to be mismatched to use the MISMATCH option, as long as the fluid and structural nodes are not merged.

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

VARIABLE**DESCRIPTION**

SSID

Segment set ID, see *SET_SEGMENT. This set defines the structural segments being coupled to the acoustic elements.

Remarks:

1. **Stability Condition.** For the stability of the acoustic-structure coupling, the following condition must be satisfied:

$$\frac{2\rho_a D}{\rho_s t_s} < 5 ,$$

where ρ_a is the density of the acoustic medium, D is the total thickness of the acoustic elements adjacent to the structural element, ρ_s is the density, and t_s is the thickness of the structural shell element. If the structural element is a solid or thick shell element, then t_s should be half the thickness of the element. If coupling is on both sides of the structural elements, then t_s should also be half the thickness of the structural element.

2. **Mismatched Coupling and Orientation.** In mismatched coupling, free fluid faces are considered for coupling with the structural segments if they are near one another and if they face each other. Faces and segments that differ in orientation by more than 45 degrees are excluded. In regions of high curvature, the surfaces, therefore, need to be more similar than when the surfaces are flat. If a fluid face couples with any structural segment, then all four integration points on the fluid face must couple with some structural segment. Fluid faces may not be partially coupled. Structural segments are allowed to be partially coupled.
3. **Mismatched Coupling Output Files.** The mismatched coupling process dumps two LS-DYNA files that can be imported into LS-PrePost for review of the results of the coupling process. The file bac_str_coupling.dyn contains shell

elements where structural segments have coupled with the fluid and mass elements at structural integration points with coupling. When the `messag` file indicates that some structural segments have partial coupling, this file can be used to check the unconnected segment integration points. The file `bac_flu_coupling.dyn` contains shell elements where free fluid faces have coupled with the structural segments and mass elements at free fluid face integration points with coupling. These files are only for visualization of the coupling and serve no other purpose.

4. **Mismatched Coupling Excludes *MAT_ACOUSTIC.** The free surface given on `*MAT_090` is excluded from the mismatched coupling logic. If there are multiple fluid regions with different free surfaces, then those regions should be given different material IDs.

***BOUNDARY_ACOUSTIC_COUPLING_SPECTRAL**

Purpose: Specify the acoustic and structural surfaces that are to participate in fluid-structure coupling. This keyword is only for coupling the structure with spectral acoustic elements. *CONTROL_ACOUSTIC_SPECTRAL must also be in the input deck.

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

VARIABLE**DESCRIPTION**

SSIDS	Segment set ID for the structural faces
SSIDF	Segment set ID for the fluid faces

Remarks:

1. **Segment Orientation.** The normal vectors of the segments in SSIDS and SSIDF should point toward each other.

*BOUNDARY

*BOUNDARY_ACOUSTIC_FREE_SURFACE

*BOUNDARY_ACOUSTIC_FREE_SURFACE

Purpose: Assign properties to an acoustic free surface in an SSD or spectral analysis (see *CONTROL_IMPLICIT_SSD_DIRECT and CONTROL_ACOUSTIC_SPECTRAL). The surface may exhibit a zero pressure condition, or for very low frequency response, a small amplitude (linear) wave surface.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	GRAV						
Type	I	F						
Default	none	0.0						

VARIABLE

DESCRIPTION

SSID Segment set ID of an acoustic surface

GRAV Acceleration of gravity

Remarks:

1. **Zero Pressure Surface.** If $\text{GRAV} = 0.0$, then the acoustic surface (SSID) exhibits a zero pressure boundary condition.
2. **Small Amplitude Waves.** If $\text{GRAV} \neq 0.0$, then the acoustic surface exhibits the small amplitude, linear wave condition of sloshing.

***BOUNDARY_ACOUSTIC_IMPEDANCE**

Purpose: Define a segment set to prescribe the acoustic impedance of acoustic volume element (type 8 and type 14 solid elements) faces.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	ZEE						
Type	I	F						
Default	none	none						

VARIABLE**DESCRIPTION**

SSID	Segment set ID, see *SET_SEGMENT
ZEE	Value of the acoustic impedance ρc

Remarks:

1. **Stability of Solutions.** The effect of the boundary impedance on the acoustic cavity response is incorporated in the forcing vector. Solutions are conditionally stable; low values of impedance relative to the impedance of the *MAT_ACOUSTIC elements cause instabilities. Reducing the factor of safety on the time step extends the range of applicability; however, it is recommended that pressure release conditions be handled by leaving the boundary free rather than by providing a relatively low boundary acoustic impedance value. If the boundary impedance value is less than 25 percent of the *MAT_ACOUSTIC impedance, LS-DYNA displays a warning. A value less than 1 percent of the *MAT_ACOUSTIC impedance is considered to be an error.
2. **Non-Reflecting Entrant Boundary Condition.** A non-reflecting entrant boundary condition is assumed when on the same segments both *LOAD_SEGMENT sets pressures and *BOUNDARY_ACOUSTIC_IMPEDANCE is defined. The pressures in the LOAD_SEGMENT_SET definition are treated as incoming incident pressure. Pressure waves within the *MAT_ACOUSTIC domain striking this boundary will exit the model. In contrast, a *LOAD_SEGMENT_SET on *MAT_ACOUSTIC volume faces in the absence of *BOUNDARY_ACOUSTIC_IMPEDANCE acts as a time-dependent, total pressure constraint. Pressure waves within the *MAT_ACOUSTIC domain striking this boundary will be reflected back into the model.

*BOUNDARY

*BOUNDARY_ACOUSTIC_IMPEDANCE_COMPLEX

*BOUNDARY_ACOUSTIC_IMPEDANCE_COMPLEX

Purpose: Assign complex impedance properties to an acoustic surface in an SSD analysis invoked with *CONTROL_IMPLICIT_SSD_DIRECT.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	ZR	ZI	LCIDR	LCIDI			
Type	I	F	F	I	I			
Default	none	0.0	0.0	0	0			

VARIABLE

DESCRIPTION

SSID	Segment set ID of an acoustic surface
ZR	Real part of the boundary impedance Z_R
ZI	Imaginary part of the boundary impedance Z_I
LCIDR	Frequency dependence of Z_R
LCIDI	Frequency dependence of Z_I

Remarks:

1. **Impedance.** On the boundary, the contribution to the damping matrix is:

$$[\bar{W}_f] = \frac{-Z_r}{(Z_r^2 + Z_i^2)} \int_S N_f^T N_f dS + \frac{iZ_i}{(Z_r^2 + Z_i^2)} \int_S N_f^T N_f dS .$$

2. **Frequency Dependence.** If the frequency dependence is not specified by a load curve, then the impedance is assumed to be constant.

***BOUNDARY_ACOUSTIC_IMPEDANCE_MECHANICAL**

Purpose: Assign mechanical impedance properties to an acoustic surface in an SSD analysis invoked with *CONTROL_IMPLICIT_SSD_DIRECT.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	MPAREA	CPAREA	KPAREA				
Type	I	F	F	F				
Default	none	0.0	0.0	0.0				

VARIABLE**DESCRIPTION**

SSID	Segment set ID of an acoustic surface
MPAREA	Mass per unit area m
CPAREA	Damping per unit area c
KPAREA	Stiffness per unit area k

Remarks:

1. **Impedance.** On the boundary, the impedance is given by

$$[\bar{W}_f] = \frac{-Z_r}{(Z_r^2 + Z_i^2)} \int_S N_f^T N_f dS + \frac{iZ_i}{(Z_r^2 + Z_i^2)} \int_S N_f^T N_f dS$$

where

$$Z_r = c, \quad Z_i = \frac{k}{\omega} - \omega m \quad .$$

***BOUNDARY_ACOUSTIC_INTERFACE**

Purpose: Read the interface file generated by *INTERFACE_ACOUSTIC from a prior analysis to load an acoustic fluid model in a second SSD analysis (see *CONTROL_IMPLICIT_SSD_DIRECT).

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

VARIABLE**DESCRIPTION**

SSID	Segment set ID, see *SET_SEGMENT. This set defines the structural segments being coupled to the acoustic elements.
IFID	Interface ID from the file baifac.db created by *INTERFACE_ACOUSTIC. It will be used to drive the acoustic fluid boundary.

Remarks:

1. **Coupling.** The faces of the acoustic boundary defined by SSID do not need to identically match up with the structural segments of IFID from the interface file. The fluid faces are projected on the structural faces, so the acoustic volume may be meshed with tetrahedra and the structure with hexahedra or shells.
2. **Interpolation.** The displacements that drive the acoustic fluid boundary are linearly interpolated from the bracketing displacement states in the interface file.

***BOUNDARY_ACOUSTIC_MAPPING**

Purpose: Define a set of elements or segments on a structure for mapping structural nodal velocity to acoustic volume boundary.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	STYP						
Type	I	I						
Default	none	0						

VARIABLE**DESCRIPTION**

SSID

Set or part ID

STYP

Set type:

EQ.0: part set ID, see *SET_PART,

EQ.1: part ID, see *PART,

EQ.2: segment set ID, see *SET_SEGMENT.

Remarks:

If acoustic elements are not overlapping the structural elements, this keyword passes structural velocity to acoustic volume boundary, for subsequent frequency domain acoustic computation.

*BOUNDARY

*BOUNDARY_ACOUSTIC_NON_REFLECTING

*BOUNDARY_ACOUSTIC_NON_REFLECTING

Purpose: Specify an absorbing acoustic boundary condition. This keyword differs from *BOUNDARY_NON_REFLECTING in the sense that higher order conditions are supported (see [Remark 2](#)).

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	NRBTYP	CRVOPT	DATA1	DATA2	DATA3		
Type	I	I	I	F	F	F		
Default	none	none	none	none	none	none		

VARIABLE

DESCRIPTION

SSID	Segment set ID of an acoustic surface
NRBTYP	Absorbing boundary type: EQ.1: Plane wave absorbing boundary EQ.2: Curved wave absorbing boundary (see CRVOPT)
CRVOPT	Curvature specification option for NRBTYP = 2: EQ.1: Provide average curvature, $1/R$ EQ.2: Provide coordinates of the center of curvature, (X_c, Y_c, Z_c)
DATA1	CRVOPT.EQ.1: Average curvature, $1/R$ CRVOPT.EQ.2: Coordinate X_c
DATA2	Coordinate Y_c for CRVOPT = 2
DATA3	Coordinate Z_c for CRVOPT = 2

Remarks:

1. **Orientation.** The boundaries are most effective if the incident wave is roughly normal to the boundary. Sharp corners with three faces coming together at a point should be avoided.

2. **Equivalence.** NRBTYP = 1 is the same as *BOUNDARY_NON_REFLECTING and the 0th order condition of Kallivokas, Bielak and McCamy. NRBTYP = 2 is equivalent to the 1st order condition of Kallivokas, Bielak and McCamy.

*BOUNDARY

*BOUNDARY_ACOUSTIC_PRESCRIBED_MOTION

*BOUNDARY_ACOUSTIC_PRESCRIBED_MOTION

Purpose: Impose a prescribed motion on the boundary of an acoustic volume.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	VAD	LCID	SF				
Type	I	I	I	F				
Default	none	none	none	1.0				

VARIABLE

DESCRIPTION

SSID	Segment set ID for the fluid boundary faces
VAD	Velocity/acceleration/displacement flag (see Remark 2): EQ.0: Velocity EQ.1: Acceleration EQ.2: Displacement
LCID	Load curve ID to describe motion as a function of time or frequency (see Remarks 1 and 2)
SF	Load curve scale factor

Remarks:

1. **SSD Analyses.** Fluid boundary velocities, accelerations, or displacements may be prescribed in SSD analyses (see *CONTROL_IMPLICIT_SSD_DIRECT). The abscissa of the load curve should then be in Hertz.
2. **Explicit Spectral Element Analyses.** Explicit simulations with *CONTROL_ACOUSTIC_SPECTRAL require the boundary motions be accelerations, VAD = 1. The abscissa of the load curve is in units of time.

***BOUNDARY_ACOUSTIC_PRESSURE_SPECTRAL**

Purpose: Impose a varying pressure condition on the boundary of an acoustic volume.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	LCID	SF	TDEATH				
Type	I	I	F	F				
Default	none	none	1.0	10 ¹⁰				

VARIABLE**DESCRIPTION**

SSID	Segment set ID for the fluid boundary faces
LCID	Load curve ID to specify pressure as a function of time
SF	Load curve scale factor
TDEATH	Time when the pressure boundary condition should no longer be applied

***BOUNDARY_ALE_MAPPING**

Purpose: This card maps ALE data histories from a previous run to a region of elements. Data are read or written in a mapping file called by the prompt “map=” on the command line (see [Remarks 4](#) and [5](#)). This keyword is supported for S-ALE and ALE. To map data at the initial time (not the histories) to the whole ALE domain (not just a region of elements), see *INITIAL_ALE_MAPPING

The following transitions are allowed:

1D → 2D

2D → 2D

3D → 3D

1D → 3D

2D → 3D

Card Summary:

Card 1. This card is required.

ID	TYP	AMMSID	IVOLTYP	BIRTH	DEATH	DTOUT	INI
----	-----	--------	---------	-------	-------	-------	-----

Card 2a. Include this card for |IVOLTYP| = 1.

THICK	RADIUS	X1	Y1	Z1			
-------	--------	----	----	----	--	--	--

Card 2b. Include this card for |IVOLTYP| = 2.

		X1	Y1	Z1	X2	Y2	Z2
--	--	----	----	----	----	----	----

Card 2c. Include this card for |IVOLTYP| = 3.

THICK	RADIUS	X1	Y1	Z1	X2	Y2	Z2
-------	--------	----	----	----	----	----	----

Card 2d. Include this blank card for |IVOLTYP| = 4.

|--|--|--|--|--|--|--|--|

Card 3. This card is required.

X0	Y0	Z0	VECID				
----	----	----	-------	--	--	--	--

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	ID	TYP	AMMSID	IVOLTYP	BIRTH	DEATH	DTOUT	INI
Type	I	I	I	I	F	F	F	I
Default	none	none	none	none	0.0	10 ²⁰	time step	0

VARIABLE**DESCRIPTION**

ID Part ID, part set ID, or element set ID

TYP Type of "ID" (see [Remark 1](#)):

EQ.0: Part set ID

EQ.1: Part ID

EQ.2: Shell set ID

EQ.3: Solid set ID

AMMSID Set ID of ALE multi-material groups defined in *SET_MULTI-MATERIAL_GROUP. See [Remark 1](#).

IVOLTYP Type of volume containing the selected elements for the mapping. The absolute value of IVOLTYPE indicates the type of volume and the sign indicates whether the data is being read or written.

Volume Type

|IVOLTYP|.EQ.1: Spherical surface with thickness (THICK)

|IVOLTYP|.EQ.2: Box

|IVOLTYP|.EQ.3: Cylindrical surface with thickness (THICK)

|IVOLTYP|.EQ.4: All the elements defined by ID

Read/Write

IVOLTYP.LT.0: Data from the mapping file are read for the elements of this volume.

IVOLTYP.GT.0: Data from the elements of this volume are written in the mapping file.

VARIABLE	DESCRIPTION
BIRTH	Birth time to write or read the mapping file. If a mapping file is written, the next run reading this file will begin at time BIRTH if this parameter for this next run is not larger.
DEATH	Death time to write or read the mapping file. If a mapping file is written, the next run will stop reading this file at time DEATH if this parameter for this next run is not smaller.
DTOUT	Time interval between outputs in the mapping file. This parameter is only used to write in the mapping file.
INI	Flag to initialize all the ALE domain of the next run: EQ.0: No initialization EQ.1: Initialization. *INITIAL_ALE_MAPPING will have to be in the input deck of the next run to read the data from the mapping file. The initial time of the next run will be BIRTH.

Spherical Surface with Thickness Card. This card is included for |IVOLTYP| = 1.

Card 2a	1	2	3	4	5	6	7	8
Variable	THICK	RADIUS	X1	Y1	Z1			
Type	F	F	F	F	F			
Default	0.0	0.0	0.0	0.0	0.0			

VARIABLE	DESCRIPTION
THICK	Thickness for the element selection using surfaces
RADIUS	Radius
X1	x-coordinate of the sphere center
Y1	y-coordinate of the sphere center
Z1	z-coordinate of the sphere center

Box Card. This card is included for $|IVOLTYP| = 2$.

Card 2b	1	2	3	4	5	6	7	8
Variable			X1	Y1	Z1	X2	Y2	Z2
Type			F	F	F	F	F	F
Default			0.0	0.0	0.0	0.0	0.0	0.0

VARIABLE**DESCRIPTION**

X1	x -coordinate of the box's minimum point
Y1	y -coordinate of the box's minimum point
Z1	z -coordinate of the box's minimum point
X2	x -coordinate of the box's maximum point
Y2	y -coordinate of the box's maximum point
Z2	z -coordinate of the box's maximum point

Cylindrical Surface with Thickness Card. This card is included for $|IVOLTYP| = 3$.

Card 2c	1	2	3	4	5	6	7	8
Variable	THICK	RADIUS	X1	Y1	Z1	X2	Y2	Z2
Type	F	F	F	F	F	F	F	F
Default	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

VARIABLE**DESCRIPTION**

THICK	Thickness for the element selection using surfaces
RADIUS	Radius
X1	x -coordinate of a point on the cylinder's axis
Y1	y -coordinate of a point on the cylinder's axis

VARIABLE	DESCRIPTION
Z1	z-coordinate of a point on the cylinder's axis
X2	x-coordinate of a vector parallel to the cylinder's axis
Y2	y-coordinate of a vector parallel to the cylinder's axis
Z2	z-coordinate of a vector parallel to the cylinder's axis

All Elements in Set Card. This blank card is included for $|IVOLTYP| = 4$.

Card 2d	1	2	3	4	5	6	7	8
Variable								
Type								

Card 3	1	2	3	4	5	6	7	8
Variable	X0	Y0	Z0	VECID				
Type	F	F	F	I				
Default	0.0	0.0	0.0	none				

VARIABLE	DESCRIPTION
X0	Origin position in global x -direction. See Remark 2 .
Y0	Origin position in global y -direction. See Remark 2 .
Z0	Origin position in global z -direction. See Remark 2 .
VECID	ID of the symmetric axis defined by *DEFINE_VECTOR. See Remark 3 .

Remarks:

- Mapping of multi-material groups.** The routines of this card need to know which mesh will be initialized with the mapping data and more specifically which multi-material groups. The first 2 parameters (ID and TYP) defines the

mesh and the third one (AMMSID) refer to the *SET_MULTI-MATERIAL_GROUP_LIST card. This card will define a list of material groups in the current run. The rank in this list should match the rank of the multi-material groups from the previous run (as a reminder the ranks of multi-material groups are defined by *ALE_MULTI-MATERIAL_GROUP). For instance, if the previous model has 3 groups, the current one has 5 groups, and the following mapping is wanted:

The 1st group (previous) \Rightarrow the 3rd group (current),
The 2nd group (previous) \Rightarrow the 5th group (current) and,
The 3rd group (previous) \Rightarrow the 4th group (current).

Then, the *SET_MULTI-MATERIAL_GROUP_LIST card should be as follows:

```
*SET_MULTI-MATERIAL_GROUP_LIST
3 0 0
3 , 5 , 4
```

2. **Origin.** The data can be mapped to different parts of the mesh by defining the origin of the coordinate system (X0, Y0, Z0).
3. **Orientation vector: VECID.** For a mapping file created by a previous axisymmetric model, the symmetric axis orientation in the current model is specified by VECID. For a mapping file created by a 3D or 1D spherical model, the vector VECID is read but ignored. The definitions of X0, Y0, Z0 and VECID change in the case of the following mappings:
 - a) plane strain 2D (ELFORM = 13 in *SECTION_ALE2D) to plane strain 2D
 - b) plane strain 2D to 3D

While VECID still defines the y -axis in the 2D domain, the 3 first parameters in *DEFINE_VECTOR, additionally, define the location of the origin. The 3 last parameters define a position along the y -axis. For this case when 2D data is used in a 3D calculation the point X0, Y0, Z0 together with the vector, VECID, define the plane.

4. **Mapping file.** To make one mapping: only the command-line argument "map=" is necessary. If IVOLTYP is positive, the mapping file will be created, and ALE data histories will be written in this file. If IVOLTYP is negative the mapping file will be read, and ALE data histories will be used to interpolate the ALE variables of the selected elements. This file contains the following nodal and element data:
 - nodal coordinates
 - nodal velocities
 - part IDs

- element connectivities
 - element centers
 - densities
 - volume fractions
 - stresses
 - plastic strains
 - internal energies
 - bulk viscosities
 - relative volumes
5. **Successive mappings.** To make several successive mappings: the prompt “map1=” is necessary. If IVOLTYP is positive and the prompt “map1=” is in the command line, the ALE data are written to the mapping file given by “map1=”. If IVOLTYP is negative and the prompt “map=” is in the command line, ALE data are read from the mapping file given by “map=”.
6. **Comparison to *ALE_MAPPING.** Both *ALE_MAPPING and *BOUNDARY_ALE_MAPPING, can write multiple sets of data to a file at a given frequency between certain times. The big difference between these two keywords is reading the data. With *ALE_MAPPING, you select only one set of data to map from the previous run to the current run which is mapped at a time you select. With *BOUNDARY_ALE_MAPPING, all data output between $t = \text{BIRTH}$ and $t = \text{DEATH}$ from the previous run is mapped to the elements in the current run. The discrete histories from the previous run are interpolated.

***BOUNDARY_AMBIENT**

Purpose: Define ALE “ambient” type element formulations (see [Remarks 1, 2, and 5](#)).

Card 1	1	2	3	4	5	6	7	8
Variable	SETID	MMG	AMBTYP	SIDR				
Type	I	I/A	I	I				
Default	none	none	none	0				

Optional Card. Additional optional card for AMBTYP = 4 with curves.

Card 1	1	2	3	4	5	6	7	8
Variable	LCID1	LCID2						
Type	I	I						
Default	Rem 1	Rem 1						

VARIABLE**DESCRIPTION**

SETID

The ambient element set ID for which the thermodynamic state is being defined. The element set can be *SET_SOLID for a 3D ALE model, *SET_SHELL for a 2D ALE model, or *SET_BEAM for a 1D ALE model.

MMG

ALE multi-material group ID. In the case of S-ALE, the AMMG name (AMMGNM) can be used in place of the AMMGID. See [Remark 6](#).

EQ.0: Keep the already existing filling. Note that this is only available if the ambient element is prefilled with ALE materials.

AMBTYP

Ambient element type:

EQ.4: Pressure inflow/outflow (see [Remarks 1 and 2](#))

EQ.5: Receptor for blast load (See *LOAD_BLAST_ENHANCED)

VARIABLE	DESCRIPTION
SIDR	<p>Flag controlling the use of this keyword during dynamic relaxation:</p> <p>EQ.0: LS-DYNA applies this keyword only during the normal analysis phase.</p> <p>EQ.1: LS-DYNA applies this keyword during the dynamic relaxation phase but not the normal analysis phase.</p> <p>EQ.2: LS-DYNA applies this keyword during the dynamic relaxation and the normal analysis phases.</p>
LCID1	A load curve ID for internal energy per unit reference volume (see Remark 4 and the *EOS section for details). If using *EOS_IDEAL_GAS, this ID refers to a temperature load curve ID.
LCID2	Load curve ID for relative volume, $v_r = (v/v_0 = \rho_0/\rho)$. See Remark 3 and the *EOS section for details.

Remarks:

1. **Ambient elements.** The term “ambient” refers to a medium with a predetermined thermodynamic state throughout the simulation. LS-DYNA resets the thermodynamic state of all “ambient” elements to this predetermined state every cycle. If using a *EOS card to determine this state, this predetermined thermodynamic state is constant throughout the simulation. For AMTYP = 4, if specifying the state with the load curves of Card 2, the thermodynamic state varies according to these defined load curves. The first and last abscissa values in these curves are this keyword’s birth and death times. The literature sometimes refers to “ambient” elements as “reservoir” elements as they may be used to simulate a semi-infinite region.
2. **Thermodynamic state.** In general, two thermodynamic variables define a thermodynamic state of a non-reacting and no-phase-change material. By defining (a) an internal energy per unit reference volume load curve (or a temperature load curve if using *EOS_IDEAL_GAS) and (b) a relative volume load curve, LS-DYNA can compute the pressure as a function of time for this ambient part ID using the equation of state (*EOS_...).
3. **Reference specific volume.** A reference specific volume, $v_0 = 1/\rho_0$, is the inverse of a reference density, ρ_0 . The reference density is the density at which the material is under a reference or nominal state. Refer to the *EOS section for a more complete explanation.

4. **Internal energy.** The internal energy per unit reference volume may be defined as

$$e_{\text{ipv0}} = \frac{C_v T}{v_0} .$$

The specific internal energy (or internal energy per unit mass) is defined as $C_v T$.

5. **Related cards.** This card does not require AET to be defined under *SECTION_SOLID or SECTION_ALE2D or SECTION_ALE1D card.
6. **AMMG name for S-ALE.** For the general ALE solver, you define each AMMG with *ALE_MULTI-MATERIAL_GROUP. In this case, you can only refer to each AMMG with their AMMGID. The order of appearance of the AMMGs in the input deck dictates the AMMGID for each AMMG. For the S-ALE solver, you can define the AMMG using *ALE_STRUCTURED_MULTI-MATERIAL_GROUP instead of *ALE_MULTI-MATERIAL_GROUP. With *ALE_STRUCTURED_MULTI-MATERIAL_GROUP, you give each AMMG a name with the field AMMGNM. You can then refer to each AMMG defined with that keyword with either its name or its AMMGID (which is again based on order of appearance). We recommend using the name as it leads to fewer errors. For instance, if you add or delete AMMGs, the AMMGIDs may change. Then, you must find all those references and alter them accordingly. With the name, you do not need to modify the input deck for unchanged AMMGs.

***BOUNDARY_AMBIENT_EOS**

Purpose: This command defines the IDs of 2 load curves: (1) internal energy per unit reference volume (or temperature if using *EOS_IDEAL_GAS) and (2) relative volume. These 2 curves completely prescribe the thermodynamic state as a function of time for any ALE or Eulerian part with an “ambient” type element formulation (please see [Remark 5](#)). For a more specific form of this keyword that allows specification of ambient boundary conditions by element set, see *BOUNDARY_AMBIENT.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	LCID1	LCID2					
Type	I	I	I					
Default	none	none	none					

VARIABLE**DESCRIPTION**

PID

The ambient part ID for which the thermodynamic state is being defined

LCID1

Load curve ID (see *DEFINE_CURVE or *DEFINE_CURVE_FUNCTION) for internal energy per unit reference volume (please read the beginning of the EOS section for details). If *EOS_IDEAL_GAS is being used, then this ID refers to a temperature load curve ID.

LCID2

Load curve ID (see *DEFINE_CURVE or *DEFINE_CURVE_FUNCTION) for relative volume,

$$v_r = \left(\frac{v}{v_0} = \frac{\rho_0}{\rho} \right) .$$

(Please read the beginning of the EOS section for details.)

Remarks:

1. **Ambient Parts.** The term “ambient” refers to a medium that has predetermined thermodynamic state throughout the simulation. All “ambient” parts/elements will have their thermodynamic state reset back to this predetermined state every cycle. If this state is defined via the *EOS card, then this predetermined thermodynamic state is constant throughout the simulation. If it is defined via this card, *BOUNDARY_AMBIENT_EOS, then its thermodynamic state will vary

according to these defined load curves. “Ambient” part is sometimes also referred to as “reservoir” part as it may be used to simulate semi-infinite region.

2. **Thermodynamic Variables.** In general, a thermodynamic state of a non-reacting and no-phase-change material may be defined by 2 thermodynamic variables. By defining (a) an internal energy per unit reference volume load curve (or a temperature load curve if using *EOS_IDEAL_GAS) and (b) a relative volume load curve, the pressure as a function of time for this ambient part ID can be computed directly via the equation of state (*EOS_...).
3. **Reference Specific Volume.** A reference specific volume, $v_0 = 1/\rho_0$, is the inverse of a reference density, ρ_0 . The reference density is defined as the density at which the material is under a reference or nominal state. Please refer to the *EOS section for additional explanation on this.
4. **Internal Energy.** The internal energy per unit reference volume may be defined as

$$e_{ipv0} = \frac{C_v T}{v_0}.$$

The specific internal energy (or internal energy per unit mass) is defined as $C_v T$.

5. **“Ambient” Elements.** This card is only to be used with an “ambient” element type as defined by the parameters under the *SECTION_SOLID card:
 - a) ELFORM = 7,
 - b) ELFORM = 11 and AET = 4, or
 - c) ELFORM = 12 and AET = 4.

Example:

Consider an ambient ALE part ID 1 which has its internal energy per unit reference volume in a load curve ID 2 and relative volume load curve ID 3:

```
$...|...1...|...2...|...3...|...4...|...5...|...6...|...7...|...8
*BOUNDARY_AMBIENT_EOS
$      PID  e/T_LCID rvol_LCID
      1      2      3
$...|...1...|...2...|...3...|...4...|...5...|...6...|...7...|...8
```

*BOUNDARY

*BOUNDARY_CONVECTION

*BOUNDARY_CONVECTION_OPTION

Available options include:

SEGMENT

SET

Purpose: Apply a convection boundary condition on a SEGMENT or SEGMENT_SET for a thermal analysis. Two cards are defined for each option.

Card 1 for SET keyword option.

Card 1a	1	2	3	4	5	6	7	8
Variable	SSID	PSEMOD						
Type	I	I						
Default	none	none						

Card 1 for SEGMENT keyword option.

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

Card 2	1	2	3	4	5	6	7	8
Variable	HLCID	HMULT	TLCID	TMULT	LOC			
Type	I	F	I	F	I			
Default	none	0.	none	0.	0			

VARIABLE	DESCRIPTION
SSID	Segment set ID, see *SET_SEGMENT.
PSEROD	Part set ID for updating boundary segments exposed to the environment as solid elements erode; see Remark 4 .
N1, N2,	Node ID's defining segment.
HLCID	<p>Convection heat transfer coefficient, h. This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DEFINE_FUNCTION and Remark 2). When the reference is to a curve, HLCID has the following interpretation:</p> <p>GT.0: h is given as a function of time, t. The curve consists of $(t, h(t))$ data pairs.</p> <p>EQ.0: h is a constant defined by the value HMULT.</p> <p>LT.0: h is given as a function of temperature, T_{film}. The curve consists of (T_{film}, h) data pairs. HLCID references the *DEFINE_CURVE load curve ID. See Remark 1.</p>
HMULT	Convection heat transfer coefficient, h , curve multiplier.
TLCID	<p>Environment temperature, T_{∞}. This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DEFINE_FUNCTION and Remark 3). When the reference is to a curve, TLCID has the following interpretation:</p> <p>GT.0: T_{∞} is defined by a curve indexed by time consisting of $(t, T_{\infty}(t))$ data pairs.</p> <p>EQ.0: T_{∞} is a constant defined by the value TMULT.</p>
TMULT	Environment temperature, T_{∞} , curve multiplier.
LOC	<p>For a thick thermal shell, the convection will be applied to the surface identified by LOC. See parameter, THSHEL, on the *CONTROL_SHELL keyword.</p> <p>EQ.-1: lower surface of thermal shell element</p> <p>EQ.0: middle surface of thermal shell element</p> <p>EQ.1: upper surface of thermal shell element</p>

Remarks:

1. **Film Temperature.** A convection boundary condition is calculated using $\dot{q}'' = h(T_{\text{surface}} - T_{\infty})$ where h is the heat transfer coefficient, and $T_{\text{surface}} - T_{\infty}$ is a temperature potential. If h is a function of temperature, h is evaluated at the average or “film” temperature defined by

$$T_{\text{film}} = \frac{(T_{\text{surface}} + T_{\infty})}{2}.$$

2. **Function Arguments for HLCID.** If HLCID references a *DEFINE_FUNCTION, the convective heat transfer coefficient can be a function of the segment centroid coordinates, the segment centroid velocity components, the segment centroid temperature, the environment temperature (T_{∞}), and the solution time, that is, “f(x, y, z, vx, vy, vz, temp, tinf, time).”
3. **Function Arguments for Environment Temperature.** If TLCID references a DEFINE_FUNCTION, the environment temperature can be a function of the segment centroid coordinates, the segment centroid velocity components, and the solution time, that is, “f(x, y, z, vx, vy, vz, time).”
4. **Erosion.** When convection is applied to a segment set associated with solid elements that erode, the convection condition on a deleted segment will not by default be transferred to newly exposed segments. It will instead cease to exist. By specifying a part set through the parameter PSEROD, any such new segment, *that is attached to an element in this part set*, will inherit this boundary condition, using the same data as prescribed for all original segments.

***BOUNDARY_COUPLED**

Purpose: Define a boundary that is coupled with an external program. Two cards are required for each coupled boundary.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	TITLE						
Type	I	A70						

Card 2	1	2	3	4	5	6	7	8
Variable	SET	TYPE	PROG					
Type	I	I	I					
Default	none	none	none					

VARIABLE**DESCRIPTION**

ID	ID for this coupled boundary
TITLE	Descriptive name for this boundary
SET	Node set ID
TYPE	Coupling type: EQ.1: node set with force feedback EQ.2: node set for multiscale spotwelds
PROG	Program to couple to EQ.1: MPP-DYNA

Remarks:

This option is only available in the MPP version and allows for loose coupling with other MPI programs using a “multiple program” execution method. Currently it is only useful when linking with MPP-DYNA for the modeling of multiscale spotwelds (TYPE = 2,

PROG = 1). See *INCLUDE_MULTISCALE_SPOTWELD for information about using this capability.

***BOUNDARY_CYCLIC_{OPTION}**

The *OPTION* allows an optional ID to be given that applies to each cyclic definition

ID

Purpose: Define nodes in boundary planes for cyclic symmetry.

These boundary conditions can be used to model a segment of an object that has rotational symmetry such as an impeller (see [Figure 5-1](#)). The segment boundary, denoted as a side 1 and side 2, may be curved or planar. In this section, a paired list of points is defined on the sides that are to be joined.

ID Card. Additional card for ID keyword option.

Card ID	1	2	3	4	5	6	7	8
Variable	ID	HEADING						
Type	I	A70						

Card 1	1	2	3	4	5	6	7	8
Variable	XC	YC	ZC	NSID1	NSID2	IGLOBAL	ISORT	
Type	F	F	F	I	I	I	I	
Default	none	none	none	none	none	0	0	

VARIABLE**DESCRIPTION**

ID	ID for boundary definition
HEADING	Description of cyclic boundary
XC	<i>x</i> -component axis vector of axis of rotation
YC	<i>y</i> -component axis vector of axis of rotation
ZC	<i>z</i> -component axis vector of axis of rotation
NSID1	Node set ID for first boundary (side 1, see Figure 5-1).

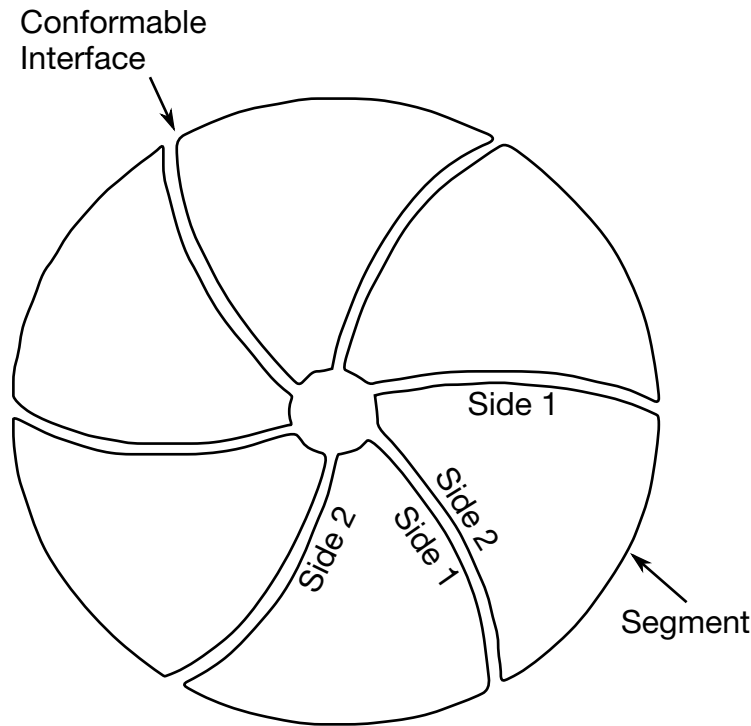


Figure 5-1. With axisymmetric cyclic symmetry, only one segment is modeled.

VARIABLE	DESCRIPTION
NSID2	Node set ID for second boundary (side 2, see Figure 5-1). Each node in this set is constrained to its corresponding node in the first node set. Node sets NSID1 and NSID2 must contain the same number of nodal points. The shape of the two surfaces formed by the two node sets need not be planar, but the shapes should match.
IGLOBAL	Flag for repeating symmetry: EQ.0: Axisymmetric cyclic symmetry (default) EQ.1: Repeating symmetry in planes normal to global X EQ.2: Repeating symmetry in planes normal to global Y EQ.3: Repeating symmetry in planes normal to global Z
ISORT	Set to 1 for automatic sorting of nodes in node sets. See Remark 2 .

Remarks:

1. **Node Sets.** Each node set should generally be boundaries of the model.
2. **Node Sorting.** Prior to version 970, it was assumed that the nodes are correctly ordered within each set, that is, the n^{th} node in NSID1 is equivalent to the n^{th}

node in NSID2. In version 970 and later versions, if the ISORT flag is active, the nodes in NSID2 are automatically sorted to achieve equivalence, so the nodes can be picked by the quickest available method. However, for axisymmetric cyclic symmetry (IGLOBAL = 0), it is assumed that the axis passes through the origin, i.e., only globally defined axes of rotation are possible.

*BOUNDARY

*BOUNDARY_DE_NON_REFLECTING

*BOUNDARY_DE_NON_REFLECTING

Purpose: Define a non-reflecting boundary for discrete element.

Card 1	1	2	3	4	5	6	7	8
Variable	NSID	CX	CY	CZ				
Type	I	F	F	F				
Default	none	1.0	1.0	1.0				
Remarks	1							

VARIABLE

DESCRIPTION

NSID	Node set ID.
CX	Decimal fraction of critical damping for translations in the x direction
CY	Decimal fraction of critical damping for translations in the y direction
CZ	Decimal fraction of critical damping for translations in the z direction

Remarks:

- Boundary Model.** Non-reflecting boundaries are used on the exterior boundaries of an analysis model of an infinite domain, such as a half-space to prevent artificial stress wave reflections generated at the model boundaries from re-entering the model and contaminating the results.

***BOUNDARY_FLUIDM**

Purpose: Request the calculation of the external, fluid boundary mass on a structural surface in an inviscid, incompressible fluid. Unless accompanied by *BOUNDARY_FLUIDM_FREE_SURFACE or *BOUNDARY_FLUIDM_BOTTOM, the fluid is assumed to be infinite.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	RHOF	IEIGEN	ICURV	BESIMF			
Type	I	F	I	I	I			
Default	none	none	0	0	0			
Remarks	1, 2, 3		4	2	5			

VARIABLE**DESCRIPTION**

SSID

Segment set ID (see *SET_SEGMENT) for the structural faces on which the fluid boundary mass is to be calculated

RHOF

Fluid mass density in the same units as the structural model

IEIGEN

Eigenvalue solution flag:

EQ.0: Use the fluid mass directly in the calculation of the in-fluid structural modes of vibration.

EQ.1: Write a superelement for the fluid-added mass matrix to file FLUIDM.DMIG, and then stop the solution. This is only advisable for very small matrices.

ICURV

Curvature usage flag:

EQ.0: If no segment radii of curvature are included in the segment set, they are estimated from the angle between the neighboring segments.

EQ.1: Disable estimating the radii of curvature.

BESIMF

Boundary element solution method for external fluid mass:

EQ.0: LAPACK direct matrix solution

EQ.1: Block low rank (BLR) direct solution without pivoting

VARIABLE	DESCRIPTION
	EQ.2: Pseudo-block GMRES iterative solution

Remarks:

1. **Normal vectors.** The outward normal vectors for the segments in the set should be directed into the external fluid.
2. **Segment curvatures.** Knowledge of local surface radii of curvature enhances the accuracy of the fluid boundary element formulation. These curvatures may be entered for individual segments in the fields DA1 and DA2 of the *SET_SEGMENT data cards. DA1 corresponds to the radius of curvature for the curve between nodes 1 and 2 while DA2 corresponds to the radius of curvature for the curve between nodes 1 and 4. If this information is not known, then it will be estimated from the change in the outward normal vectors for neighboring elements. You may disable this estimate with ICURV = 1.
3. **Eccentricity and back-to-back segments.** If the external fluid appears on both sides of a plate element, the fluid mass segments must be offset from the plane of the element. To do this, you specify the eccentricity of the segment in the DA3 field in *SET_SEGMENT. The eccentricity is half the plate thickness. If back-to-back segments are not offset, then a fatal, singularity error will occur.
4. **Eigenvalue solutions.** Currently, *BOUNDARY_FLUIDM may only be used in eigenvalue solutions and those which use eigenvectors as the basis of the solution, such as *FREQUENCY_DOMAIN solutions. Furthermore, the eigenvalue solution must use *CONTROL_IMPLICIT_EIGENVALUE method EIGMTH = 102 (LOBPCG).
5. **Boundary mass solution method.** The LAPACK solution method requires at least $NSEGS^2$ words of memory and is generally the most robust of the solution strategies. The BLR solution method is faster and requires less memory, typically $1/4^{th}$ of the memory for LAPACK. GMRES requires the least amount of memory but is the slowest method.

***BOUNDARY_FLUIDM_BOTTOM**

Purpose: Includes the effects of a flat, arbitrarily oriented bottom in the calculation of the *BOUNDARY_FLUIDM mass matrix. A finite bottom impedance may be included with a normalized, reflection coefficient. *BOUNDARY_FLUIDM_BOTTOM can be combined with *BOUNDARY_FLUIDM_FREE_SURFACE in very shallow conditions.

Card 1	1	2	3	4	5	6	7	8
Variable	DISTBT	CXBT	CYBT	CZBT	BNORM			
Type	F	F	F	F	F			
Default	none	none	none	none	1.0			
Remarks	1							

VARIABLE**DESCRIPTION**

DISTBT	Perpendicular distance from the plane of the bottom to the origin of the global coordinate system
CXBT	X-direction cosine for the normal vector of the bottom pointing into the bottom
CYBT	Y-direction cosine for the normal vector of the bottom pointing into the bottom
CZBT	Z-direction cosine for the normal vector of the bottom pointing into the bottom
BNORM	Low frequency bottom reflection coefficient ($0 < \text{BNORM} \leq 1$)

Remarks:

1. **Proximity Effect.** When nearby, a bottom will increase the added mass experienced by a submerged structure.

*BOUNDARY

*BOUNDARY_FLUIDM_FREE_SURFACE

*BOUNDARY_FLUIDM_FREE_SURFACE

Purpose: Includes the effects of a flat, arbitrarily oriented free surface in the calculation of the *BOUNDARY_FLUIDM mass matrix. Keyword *BOUNDARY_FLUIDM_BOTTOM can be combined with *BOUNDARY_FLUIDM_FREE_SURFACE in very shallow conditions.

Card 1	1	2	3	4	5	6	7	8
Variable	DISTFS	CXFS	CYFS	CZFS				
Type	F	F	F	F				
Default	none	none	none	none				
Remarks	1, 2							

VARIABLE

DESCRIPTION

DISTFS	Perpendicular distance from the plane of the free surface to the origin of the global coordinate system
CXFS	X-direction cosine for the normal vector of the free surface pointing into the air
CYFS	Y-direction cosine for the normal vector of the free surface pointing into the air
CZFS	Z-direction cosine for the normal vector of the free surface pointing into the air

Remarks:

1. **Proximity Effect.** When nearby, a free surface will reduce the added mass experienced by a submerged structure.
2. **Elements Penetrating Free Surface.** Any FLUIDM segment which penetrates the free surface will be removed from the segment set.

***BOUNDARY_FLUIDM_INTERIOR**

Purpose: Request calculating the added mass for an inviscid, incompressible fluid in a closed, internal volume/tank with a free surface. An input deck may contain multiple instantiations of this keyword when there are multiple separate internal volumes. It is intended for structural eigenvalue solutions and linear dynamic solutions based on modal superposition.

Card 1	1	2	3	4	5	6	7	8
Variable	TNAME							
Type	A6							
Default	none							

Card 2	1	2	3	4	5	6	7	8
Variable	FSTYPE	IEIGEN	SSID		SSIDBF	BESIMF		
Type	I	I	I		I	I		
Default	none	0	none		0	0		
Remarks		1, 5	2		2, 3	4		

Card 3	1	2	3	4	5	6	7	8
Variable	RHOF		DISTFS	CXFS	CYFS	CZFS		
Type	F		F	F	F	F		
Default	none		0.0	0.0	0.0	0.0		

VARIABLE	DESCRIPTION
TNAME	Name of the tank or volume. If forming a fluid mass superelement, this name is the file name.
FSTYPE	Free surface type: EQ.2: A quiescent zero pressure surface located in space by DISTFS, CXFS, CYFS, and CZFS.
IEIGEN	Eigenvalue solution flag: EQ.0: Use the fluid mass directly to calculate the in-fluid structural modes of vibration. EQ.1: Write a superelement for the fluid-added mass matrix to file [TNAME].DMIG, and then stop the solution. This is only advisable for very small matrices.
SSID	Segment set ID for the structural faces for which the fluid boundary mass is calculated, excluding the free surface.
SSIDBF	Segment set ID for internal baffle faces having fluid on both sides of plate elements.
BESIMF	Boundary element solution method for internal fluid mass: EQ.0: LAPACK direct matrix solution (default) EQ.1: Block low rank (BLR) direct solution without pivoting EQ.2: Pseudo-block GMRES iterative solution
RHOF	Fluid mass density in the same units as the structural model
DISTFS	Perpendicular distance from the plane of the free surface to the origin of the global coordinate system
CXFS	x -direction cosine for the normal vector of the free surface pointing into the air
CYFS	y -direction cosine for the normal vector of the free surface pointing into the air
CZFS	z -direction cosine for the normal vector of the free surface pointing into the air

Remarks:

1. **Consistency with *BOUNDARY_FLUIDM.** The same choice for IEIGEN should be made for all internal volumes and be consistent with the choice made for external fluid if it also exists.
2. **Normal vectors.** The outward normal vectors for the segments in the set should be directed out of the internal fluid volume.
3. **Baffle segments.** Only one side of the baffle should be specified with the segment set SSIDBF. Duplicate faces with opposite normal vectors are automatically generated. The thickness of the baffle must be specified with DA3 on *SET_SEGMENT.
4. **Boundary mass solution method.** The LAPACK solution method requires at least $NSEGS^2$ words of memory and is generally the most robust of the solution strategies. The BLR solution method is faster and requires less memory, typically $1/4^{th}$ of the memory for LAPACK. GMRESM requires the least amount of memory but is the slowest method.
5. **Eigenvalue solutions.** Currently, *BOUNDARY_FLUIDM_INTERIOR may only be used in eigenvalue solutions and solutions that use eigenvectors as the basis of the solution, such as *FREQUENCY_DOMAIN solutions. Furthermore, the eigenvalue solution must use *CONTROL_IMPLICIT_EIGENVALUE method EIGMTH = 102 (LOBPCG).

***BOUNDARY_FLUX_OPTION**

Available options include:

SEGMENT

SET

Purpose: Apply a flux boundary condition (power / area) on a SEGMENT or SEGMENT_-SET for a thermal analysis. History variables can be associated with the boundary condition which will invoke a call to a user defined boundary flux subroutine for computing the flux.

Card Summary:

Card 1a. This card is included if and only if the SET option is used.

SSID	PSEROD						
------	--------	--	--	--	--	--	--

Card 1b. This card is included if and only if the SEGMENT option is used.

N1	N2	N3	N4				
----	----	----	----	--	--	--	--

Card 2. This card is required.

LCID	MLC1	MLC2	MLC3	MLC4	LOC	NHISV	
------	------	------	------	------	-----	-------	--

Card 3. Include as many of this card as needed to define NHISV history variables.

HISV1	HISV2	HISV3	HISV4	HISV5	HISV6	HISV7	HISV8
-------	-------	-------	-------	-------	-------	-------	-------

Data Cards:**Card 1 for SET option.**

Card 1a	1	2	3	4	5	6	7	8
Variable	SSID	PSEROD						
Type	I	I						
Default	none	none						

VARIABLE	DESCRIPTION
SSID	Segment set ID, see *SET_SEGMENT
PSEMOD	Part set ID for updating boundary segments exposed to the environment as solid elements erode; see Remark 5 .

Card 1 for SEGMENT option.

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

VARIABLE	DESCRIPTION
N1, N2, ...	Node IDs that define the segment

Card 2	1	2	3	4	5	6	7	8
Variable	LCID	MLC1	MLC2	MLC3	MLC4	LOC	NHISV	
Type	I	F	F	F	F	I	I	
Default	none	0.	0.	0.	0.	0	0	

VARIABLE	DESCRIPTION
LCID	<p>This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DEFINE_FUNCTION and Remark 2) for heat flux. When the reference is to a curve, LCID has the following interpretation:</p> <p>GT.0: the flux is defined by a curve consisting of (time, flux) data pairs using the *DEFINE_CURVE keyword. The flux value applied to the nodal points is the curve value multiplied by the values MLC1, MLC2, MLC3, and MLC4, respectively.</p>

VARIABLE	DESCRIPTION
	EQ.0: a constant flux is applied to each node defined by the values MLC1, MLC2, MLC3, and MLC4, respectively.
	LT.0: the flux is defined by a curve consisting of (temperature, flux) data pairs using the *DEFINE_CURVE keyword. The flux value applied to the nodal points is the curve value multiplied by the values MLC1, MLC2, MLC3, and MLC4. Enter –LCID on the *DEFINE_CURVE keyword.
MLC1	Curve multiplier at node N1.
MLC2	Curve multiplier at node N2.
MLC3	Curve multiplier at node N3.
MLC4	Curve multiplier at node N4.
LOC	For a thick thermal shell, the flux will be applied to the surface identified by LOC. See parameter, THSHEL, on the *CONTROL_SHELL keyword. EQ.-1: lower surface of thermal shell element EQ.0: middle surface of thermal shell element EQ.1: upper surface of thermal shell element
NHISV	Number of history variables associated with the flux definition: GT.0: A user defined subroutine will be called to compute the flux. See Remark 3 .

Define as many cards as necessary to initialize NHISV history variables.

Card 3	1	2	3	4	5	6	7	8
Variable	HISV1	HISV2	HISV3	HISV4	HISV5	HISV6	HISV7	HISV8
Type	F	F	F	F	F	F	F	F
Default	0.	0.	0.	0.	0.	0.	0.	0.

VARIABLE	DESCRIPTION
HISV1	Initial value of history variable 1
HISV2	Initial value of history variable 2
⋮	⋮
HISV n	Initial value of history variable n , where $n = \text{NHISV}$

Remarks:

1. **Flux Direction.** The segment normal has no bearing on the flux. A negative flux transfers energy into the volume; a positive flux transfers energy out of the volume.
2. **Flux Definition with *DEFINE_FUNCTION.** If LCID references a *DEFINE_FUNCTION, the heat flux can be a function of the segment centroid coordinates, the segment velocity components, the segment centroid temperature, the environment temperature, and the solution time, that is “f(x, y, z, vx, vy, vz, temp, tinf, time).”
3. **Flux Definition with User Subroutines.** When NHISV>0, the user subroutine

```
subroutine usrflux(fl, flp, ...)
```

will be called to compute the heat flux (fl). For more details see Appendix S.
4. **SPH Elements.** This keyword is supported for SPH elements to define the flux boundary conditions for a thermal or coupled thermal/structural analysis. The values n_1, n_2, n_3, n_4 from the SPH particles or segments are used to define the flux segments.
5. **Erosion.** When flux is applied to a segment set associated with solid elements that erode, the flux condition on a deleted segment will not by default be transferred to newly exposed segments. It will instead cease to exist. By specifying a part set through the parameter PSEMOD, any such new segment, *that is attached to an element in this part set*, will inherit this boundary condition, using the same data as prescribed for all original segments.

*BOUNDARY

*BOUNDARY_FLUX_TRAJECTORY

*BOUNDARY_FLUX_TRAJECTORY

Purpose: Define a moving surface heat source to model for example laser heated surfaces of solid or shell structures. Motion of the source is described by a nodal path and a prescribed velocity on this path. This keyword is applicable in coupled thermal-structural and thermal-only simulations.

Card Summary:

Card 1. This card is required.

SSID	PSEROD	NSID1	SPD1	NSID2	SPD2	RELVEL	
------	--------	-------	------	-------	------	--------	--

Card 2. This card is required.

EROD	LOC	LCROT	LCLAT				
------	-----	-------	-------	--	--	--	--

Card 3. This card is required.

IFORM	LCTIM	Q	LCINC	ENFOR			
-------	-------	---	-------	-------	--	--	--

Card 4. This card is required.

P1	P2	P3	P4	P5	P6	P7	P8
----	----	----	----	----	----	----	----

Card 5. This card is included if and only if NSID2 = 0.

TX	TY	TZ					
----	----	----	--	--	--	--	--

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	PSEROD	NSID1	SPD1	NSID2	SPD2	RELVEL	
Type	I	I	I	F	I	F	I	
Default	none	none	none	0.	none	0.	0	

VARIABLE

DESCRIPTION

SSID

Segment set ID containing segments that are potentially heated by surface heat source (flux).

VARIABLE	DESCRIPTION
PSEROD	Part set ID for updating boundary segments exposed to the environment as solid elements erode; see Remark 5 .
NSID1	Node set defining the path of the weld source. The source travels along the path at speed SPD1. The nodes are traversed according to their ordering in the node set. See Remark 1 .
SPD1	Speed of the heat source on the weld trajectory GT.0.0: constant speed LT.0.0: SPD1 is a load curve ID defining weld speed as a function of time.
NSID2	Node or segment set containing information for the weld source aiming direction: GT.0: NSID2 together with SPD2 define a curve in the same way that NSID1 and SPD1 define a curve. Aiming direction is taken to be the vector pointing from the current position along NSID2 (for example your hand holding the torch) to the current position on NSID1 (the weld source). EQ.0: beam aiming direction is (TX, TY, TZ) input on Card 5.
SPD2	Speed of reference point in NSID2 (ignored unless NSID2 > 0) GT.0: constant speed LT.0: SPD2 is a load curve ID defining weld speed as a function of time.
RELVEL	Defines if SPD1 and SPD2 are relative or absolute speeds in coupled simulations EQ.0: absolute speeds EQ.1: relative speeds with respect to underlying structure

BOUNDARY**BOUNDARY_FLUX_TRAJECTORY**

Card 2	1	2	3	4	5	6	7	8
Variable	EROD	LOC	LCROT	LCLAT				
Type	I	I	I	I				
Default	0	none	none	none				

VARIABLE**DESCRIPTION**

EROD

Flag for updating boundary segments exposed to the environment as solid elements (defined in PSEROD) erode; see [Remark 5](#).

EQ.0: no propagation onto new segments

EQ.1: propagation onto new segments

LOC

For a thick thermal shell, the flux will be applied to the surface identified by LOC. See field THSHEL on the *CONTROL_SHELL keyword.

EQ.-1: lower surface of thermal shell element

EQ.0: middle surface of thermal shell element

EQ.1: upper surface of thermal shell element

LCROT

Load curve defining the rotation (angle in degrees) of weld source around the trajectory as function of time. See [Remark 2](#).

LCLAT

Load curve for lateral offset of weld source as function of time. See [Remark 2](#).

Card 3	1	2	3	4	5	6	7	8
Variable	IFORM	LCTIM	Q	LCINC	ENFOR			
Type	I	I	F	I	I			
Default	none	none	0.	none	0			

VARIABLE	DESCRIPTION
IFORM	<p>Geometry description for energy rate density distribution (see Remark 3):</p> <p>EQ.1: double elliptic with constant density</p> <p>EQ.2: double elliptic with Gaussian distribution</p> <p>EQ.3: user defined distribution</p>
LCTIM	<p>Load curve ID for flux energy input rate multiplier $q_1(t)$ as a function of time, see Remark 4.</p> <p>EQ.0: use constant multiplier value $q_1(t) = 1.0$.</p>
Q	<p>Base energy input rate Q_b [energy/time], serves as base multiplier for the energy rate distributions defined by IFORM. See Remark 4 for details.</p>
LCINC	<p>Load curve ID for flux energy input rate multiplier $q_2(\alpha)$ as a function of inclination angle α (in degree), see Remark 4.</p> <p>EQ.0: use constant multiplier value $q_2(\alpha) = 1.0$.</p>
ENFOR	<p>Flag for heat input enforcement option (see Remark 4)</p> <p>EQ.0: no additional scaling of heat source</p> <p>EQ.1: account for inclination angle, α, of heat source by energy input rate multiplier $q_3(\alpha) = \cos(\alpha)$</p> <p>EQ.2: scale the nodal fluxes by a multiplier $q_4(t)$ such that the resulting heat input equals the user input $Q(t) = Q_b \times q_1(t)$</p> <p>EQ.3: apply option 1 and 2</p>

Card 4	1	2	3	4	5	6	7	8
Variable	P1	P2	P3	P4	P5	P6	P7	P8
Type	F	F	F	F	F	F	F	F
Default	0.	0.	0.	0.	0.	0.	0.	0.

VARIABLE	DESCRIPTION
P_i	Parameters defining flux geometry, depending on field IFORM. See Remark 3 for details.

Optional Weld Source Aiming Direction Card. Additional card for NSID2 = 0.

Card 5	1	2	3	4	5	6	7	8
Variable	TX	TY	TZ					
Type	F	F	F					
Default	0.	0.	0.					

VARIABLE	DESCRIPTION
TX, TY, TZ	Weld beam direction vector in global coordinates (NSID2 = 0 only)

Remarks:

1. **Heat Source Motion.** This card can be applied to coupled thermal-structure and thermal-only simulations. The source motion can be defined independently from the motion of nodes. This keyword applies to both solid and thermal thick shells.

If NSID1 consists of work piece nodes, then the heat source will follow the motion of the work piece. By setting field RELVEL to 1 the velocity of the heat source can be defined relative to the motion of the nodes in NSID1 (the work piece).

2. **Heat Source Direction (Tilting the Torch).** There is a natural local coordinate system $(\mathbf{r}, \mathbf{s}, \mathbf{t})$ associated with the motion of the heat source. The relative velocity vector on the trajectory of the heat source defines the "forward" direction \mathbf{r} , so that material points that are approaching the heat source are in "front" of the beam. The weld source aiming direction, denoted by \mathbf{t} , determines the direction of the weld pool depth. The coordinate direction \mathbf{s} is normal to the plane containing both the relative velocity and the aiming direction; note that $\mathbf{s} = \mathbf{t} \times \mathbf{r}$ to have a right handed coordinate system. It determines the direction of the weld pool width.

The position and aiming direction of the heat source (for example tilting the torch) can be adjusted by rotating and translating the local coordinate system.

To do this, the system is first rotated around the vector \mathbf{r} by a value given by the load curve LCROT, resulting in a new local coordinate system $(\mathbf{r}, \mathbf{s}', \mathbf{t}')$. Then, the system is translated in direction \mathbf{s}' using LCLAT, respectively.

3. **Energy Rate Surface Distributions.** Several different heat source geometries, described below, can be defined with this keyword. The local coordinate system needed for the description of the geometry is discussed in [Remark 2](#).

For IFORM = 1 heat is generated in an elliptical region centered at the heat source with a constant energy rate surface density. The half-width a of the ellipse in the \mathbf{s}' -direction is given by input parameter P1. The half-widths of the ellipse in \mathbf{r} -direction are different in front and behind the beam. The half-width in front of the beam is denoted by b_f (input parameter P2) and behind the beam is referred to as b_r (input parameter P3). Consequently, the flux value at a point p in the ellipse is given as

$$q = \frac{F}{\pi ab},$$

where:

$$F = \begin{cases} F_f & \text{if point } p \text{ is in front of beam} \\ F_r & \text{if point } p \text{ is behind beam} \end{cases}$$

$$b = \begin{cases} b_f & \text{if point } p \text{ is in front of beam} \\ b_r & \text{if point } p \text{ is behind beam} \end{cases}$$

It is expected that the sum of the weighting factors, F_f and F_r (defined as input parameters P4 and P5), equals 2.

The normalized surface density q for IFORM = 2 is assumed to be controlled by an exponential decay from the center of the ellipse to the boundary. The geometry of the source and its input correspond to IFORM = 1, but the energy rate surface density is here given as

$$q = \frac{nF}{\pi ab} \exp\left(\frac{-nx^2}{a^2}\right) \exp\left(\frac{-ny^2}{b^2}\right)$$

with the same assumptions for F and b as above and with (x, y) being the first two coordinates of point p with respect to the new local coordinate system $(\mathbf{r}, \mathbf{s}', \mathbf{t}')$. The additional parameter n is defined by input parameter P6.

In contrast to the above, IFORM = 3 refers to a user defined function that determines the surface density q as function of the local coordinates (x, y) , the current time t , and the temperature T . Consequently, the definition of an arbitrarily shape equivalent heat source may read as follows:

```
*DEFINE_FUNCTION
1, surface density distribution
q(r,s,time,temp) = ...
```

To speed-up the computation, the function is evaluated for nodes within a distance less than a from the current center position. Consequently, the function ID and this distance are the only parameters needed in the input for Card 4.

Note that in all of the above cases, the normalized surface density q is multiplied by the energy rate $Q(t, \alpha)$, which is discussed in [Remark 4](#).

IFORM	1	2	3
P1	a	a	FID
P2	b_f	b_f	a
P3	b_r	b_r	
P4	F_f	F_f	
P5	F_r	F_r	
P6		n	
P7			
P8			

4. **Energy Rate Input and Enforcement.** Naturally the calculation of the surface fluxes as discussed in [Remark 3](#) requires the definition of a base energy input rate Q . It can be defined as a function of time, t , and inclination angle, α :

$$Q(t, \alpha) = Q_b \times q_1(t) \times q_2(\alpha) \times q_3(\alpha) \times q_4(t)$$

The multipliers $q_1(t)$ and $q_2(\alpha)$ are given as user-defined load curves. The multiplier $q_3(\alpha)$, if activated by a corresponding choice of ENFOR, account for the fact that a tilted heat source affects a larger area of the surface and is set to $q_3(\alpha) = \cos(\alpha)$. The value of α ranges from 0° (the segment normal and heat source point in the same direction) to 180° (the segment normal and heat source point in opposite directions). Naturally, the values of $q_2(\alpha)$ and $q_3(\alpha)$ vary among the individual segments.

For some applications it is required that the total heat flux from the boundary condition always agrees with the user input, meaning $Q_b \times q_1(t)$. The integration of the surface fluxes does not always meet this requirement, particularly for coarse meshes and highly curved structures. For that purpose, option ENFOR = 2 or 3 are implemented, which modify the nodal fluxes by a scaling factor $q_4(t)$. Note that this factor is not a user input but internally calculated and the same for all nodes with the particular flux boundary condition.

5. **Erosion.** When flux is applied to a segment set associated with solid elements that erode, the flux condition on a deleted segment will not by default be

transferred to newly exposed segments. It will instead cease to exist. By specifying a part set through the parameter PSEROD and activating the feature with EROD, any such new segment, *which is attached to an element in this part set*, will inherit this boundary condition, using the same data as prescribed for all original segments.

***BOUNDARY_MCOL**

Purpose: Define parameters for MCOL coupling. The MCOL Program is a rigid body mechanics program for modeling the dynamics of ships. See [Remark 1](#) for more information.

Card 1	1	2	3	4	5	6	7	8
Variable	NMCOL	MXSTEP	ETMCOL	TSUBC	PRTMCOL			
Type	I	I	F	F	F			
Default	2	none	0.0	0.0	none			
Remarks			2					

Ship Card. Include NMCOL cards, one for each ship.

Card 2	1	2	3	4	5	6	7	8
Variable	RBMCOL	MCOLFILE						
Type	I	A60						
Default		none						

VARIABLE**DESCRIPTION**

NMCOL	Number of ships in MCOL coupling.
MXSTEP	Maximum number of time steps allowed in MCOL calculation. If the number of MCOL time steps exceeds MXSTEP, then LS-DYNA will terminate.
ETMCOL	Uncoupling termination time; see Remark 2 below. EQ.0.0: set to LS-DYNA termination time
TSUBC	Time interval for MCOL subcycling. EQ.0.0: no subcycling

VARIABLE	DESCRIPTION
PRTMCOL	Time interval for output of MCOL rigid body data.
RBMCOL	LS-DYNA rigid body material assignment for the ship.
MCOLFILE	Filename containing MCOL input parameters for the ship.

Remarks:

1. **MCOL Coupling.** The basis for MCOL is a convolution integral approach for simulating the equations of motion. A mass and inertia tensor are required as input for each ship. The masses are then augmented to include the effects of the mass of the surrounding water. A separate program determines the various terms of the damping/buoyancy force formulas which are also input to MCOL. The coupling is accomplished in a simple manner: at each time step LS-DYNA computes the resultant forces and moments on the MCOL rigid bodies and passes them to MCOL. MCOL then updates the positions of the ships and returns the new rigid body locations to LS-DYNA. A more detailed theoretical and practical description of MCOL can be found in a separate report (to appear).
2. **ETMCOL.** After the end of the LS-DYNA/MCOL calculation, the analysis can be continued using MCOL alone. ETMCOL is the termination time for this analysis. If ETMCOL is smaller than the LS-DYNA termination time, the uncoupled analysis will not be activated.
3. **Output Files.** The MCOL output is written to the files mcolout (ship position) and mcolenergy (energy breakdown). In LS-PrePost, mcolout can be plotted through the rigid body time history option and MCOLENERGY.

***BOUNDARY_NON_REFLECTING**

Purpose: Define a non-reflecting boundary. This option applies to continuum domains modeled with solid or thick shell elements. For geomechanical problems this option is important for limiting the spatial extent of the finite element mesh and thus the number of solid elements.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	AD	AS					
Type	I	F	F					
Default	none	0.0	0.0					
Remarks	1, 2	3	3					

VARIABLE**DESCRIPTION**

SSID	Segment set ID, see *SET_SEGMENT.
AD	Default activation flag for dilatational waves. EQ.0.0: on NE.0.0: off
AS	Default activation flag for shear waves. EQ.0.0: on NE.0.0: off

Remarks:

1. **Restrictions.** Non-reflecting boundaries defined with this keyword are only used with three-dimensional solid elements. Boundaries are defined as a collection of segments, and segments are equivalent to element faces on the boundary. Segments are defined by listing the corner nodes in either a clockwise or counterclockwise order.
2. **Model Description.** Non-reflecting boundaries are used on the exterior boundaries of an analysis model of an infinite domain, such as a half-space to prevent artificial stress wave reflections generated at the model boundaries from

reentering the model and contaminating the results. Internally, LS-DYNA computes an impedance matching function for all non-reflecting boundary segments based on an assumption of linear material behavior. Thus, the finite element mesh should be constructed so that all significant nonlinear behavior is contained within the discrete analysis model.

3. **Reflecting Waves.** With the two optional flags, the influence of reflecting waves can be studied.
4. **Dynamic Relaxation.** During the dynamic relaxation phase (optional), nodes on non-reflecting segments are constrained in the normal direction. Nodal forces associated with these constraints are then applied as external loads and held constant in the transient phase while the constraints are replaced with the impedance matching functions. In this manner, soil can be quasi-statically prestressed during the dynamic relaxation phase and dynamic loads (with non-reflecting boundaries) subsequently applied in the transient phase.
5. **Time Step Size.** In explicit analyses, TSSFAC in *CONTROL_TIMESTEP is set to 0.667 by default if not specified, in order to obtain a stable solution. This is typically necessary if three non-reflecting boundary planes meet at a corner. A larger value may be specified to obtain a shorter simulation time if it does not affect stability.

***BOUNDARY_NON_REFLECTING_2D**

Purpose: Define a non-reflecting boundary. This option applies to continuum domains modeled with two-dimensional solid elements in the xy plane. For geomechanical problems, this option is important for limiting the size of the models. For ALE 2D models, including this keyword without data cards (Card 1) causes the non-reflecting boundary condition to be applied to all the boundary edges of the 2D ALE mesh.

Card 1	1	2	3	4	5	6	7	8
Variable	NSID	AD	AS					
Type	I	I	I					
Default	none	0	0					
Remarks	1, 2							

VARIABLE**DESCRIPTION**

NSID

Node set ID; see *SET_NODE. See [Figure 5-2](#).

LT.0: |NSID| is the ID of *SET_SEGMENT.

AD

Default activation flag for dilatational waves:

EQ.0.0: On

NE.0.0: Off

AS

Default activation flag for shear waves:

EQ.0.0: On

NE.0.0: Off

Remarks:

1. **Boundary Definition.** Non-reflecting boundaries defined with this keyword are only used with two-dimensional solid elements in either plane strain or axisymmetric geometries. Boundaries are defined as a sequential string of nodes moving *counterclockwise* around the boundary.

2. **Model Assumptions.** Non-reflecting boundaries are used on the exterior boundaries of an analysis model of an infinite domain, such as a half-space to prevent artificial stress wave reflections generated at the model boundaries from re-entering the model and contaminating the results. Internally, LS-DYNA computes an impedance matching function for all non-reflecting boundary segments based on an assumption of linear material behavior. Thus, the finite element mesh should be constructed so that all significant nonlinear behavior is contained within the discrete analysis model.

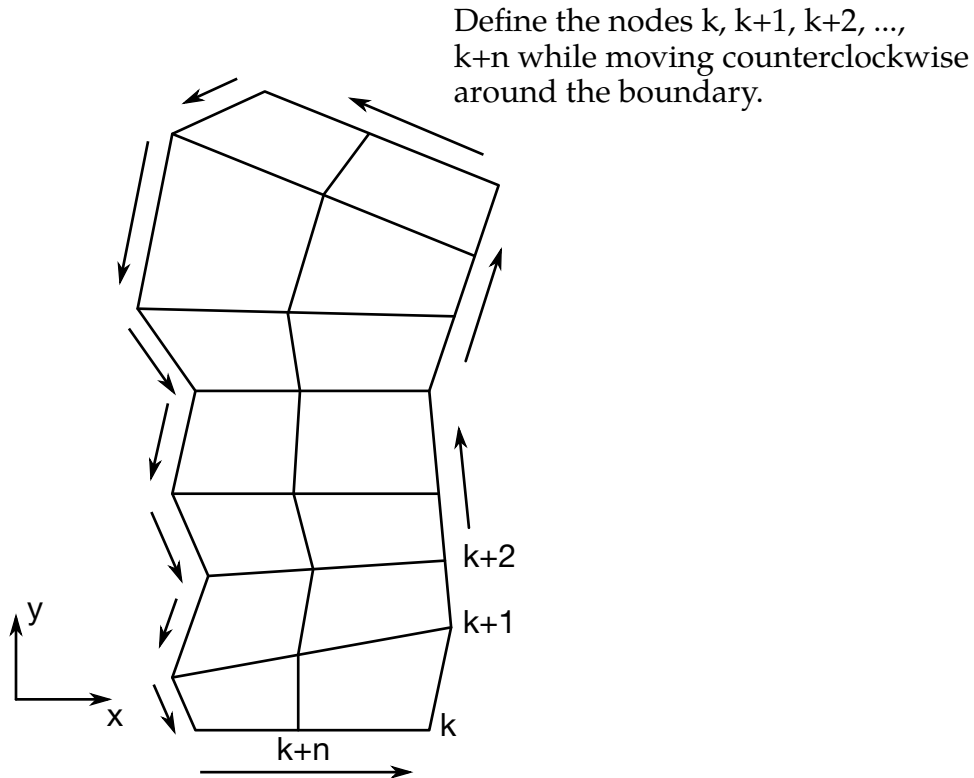


Figure 5-2. When defining a transmitting boundary in 2D define the node numbers in the node set in consecutive order while moving counterclockwise around the boundary.

***BOUNDARY_PAP**

Purpose: Define pressure boundary conditions for pore air flow calculation, such as at the structure surface exposed to atmospheric pressure.

Card 1	1	2	3	4	5	6	7	8
Variable	SEGID	LCID	CMULT	CVMASS	BLOCK	TBIRTH	TDEATH	CVRPER
Type	I				F	F	F	F
Default	none	CMULT	none	none	0.0	0.0	10 ²⁰	1.0
Remarks				1, 2				3

VARIABLE**DESCRIPTION**

SEGID	Segment set ID
LCID	Load curve giving pore air pressure as a function of time. EQ.0: constant pressure assumed equal to CMULT
CMULT	Factor on curve or constant pressure head if LCID = 0.
CVMASS	Initial mass of a control volume next to the segment set SEGID
BLOCK	Contact blockage effect, EQ.0: When all segments in SEGID are subject to the pressure defined by LCID and CMULT; EQ.1: When only elements in SEGID not involved in contact are subject to the pressure defined by LCID and CMULT.
TBIRTH	Time at which boundary condition becomes active
TDEATH	Time at which boundary condition becomes inactive
CVRPER	Permeability factor of cover material, where cover refers to a shell layer coating the surface of the solid. Default value is 1.0 when it is not defined.

$$0.0 \leq \text{CVRPER} \leq 1.0$$

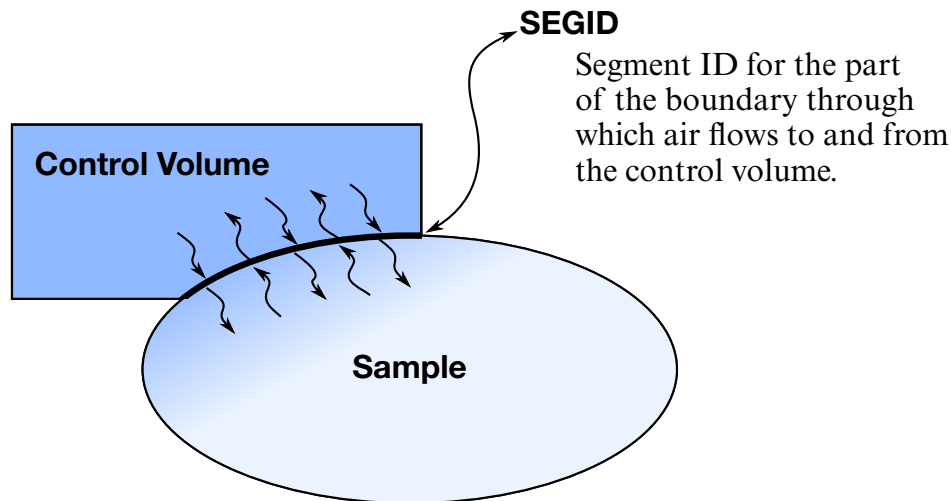


Figure 5-3. Air flows between the control volume and the sample. CVMASS specifies the control volume's initial mass, and CVMULT sets the initial pressure.

Remarks:

1. **Structure Surfaces.** All structure surfaces subject to specified pressure must be defined.
2. **Air Mass Transfer.** A non-zero CVMASS, together with a non-zero CMULT and an un-defined LCID, can be used to simulate air mass transfer between a control volume and a test specimen containing pore air. The control volume is assumed to have a fixed volume, an initial pressure of CMULT, and an initial mass of CVMASS. Air mass transfer happens between the control volume and its neighboring specimen. Such mass transfer results in pressure change in control volume and test specimen.
3. **Porosity Properties.** The porosity properties of the cover material can be modeled using CVRPER. If SEGID is covered by a material of very low permeability (e.g., coated fabric), CVRPER should be set to 0.0. In this case, P_c , the pressure calculated assuming no boundary condition, is applied to SEGID. If SEGID is not covered by any material, CVRPER should be set to 1.0, the default value. In this case, the applied pressure becomes P_b , the boundary pressure determined by CMULT and LCID.

***BOUNDARY_PORE_FLUID_OPTION**

Available options include:

PART

SET

Purpose: Define parts that contain pore fluid. Defaults are given on *CONTROL_PORE_FLUID.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	WTABLE	PF_RHO	ATYPE	PF_BULK	ACURVE	WTCUR	SUCLIM
Type	I	F	F	I	F	I	I	F
Default	none	*	*	*	*	0	0	0.

*Defaults are taken from *CONTROL_PORE_FLUID.

VARIABLE	DESCRIPTION
PID	Part ID or part set ID. All elements within the part must lie below the water table.
WTABLE	Z-coordinate at which pore pressure = 0 (water table)
PF_RHO	Density of pore water in soil skeleton (see Remark 2): EQ.0: Default density specified on *CONTROL_PORE_FLUID card is used.
ATYPE	Analysis type for parts: EQ.0: Default to value specified on *CONTROL_PORE_FLUID EQ.1: Undrained analysis EQ.2: Drained analysis EQ.3: Time dependent consolidation (coupled) EQ.4: Consolidate to steady state (uncoupled) EQ.5: Drained in dynamic relaxation, undrained in transient

VARIABLE	DESCRIPTION
PF_BULK	Bulk modulus of pore fluid: EQ.0: Default to value specified on *CONTROL_PORE_FLUID
ACURVE	Curve of analysis type as a function of time (see Remark 3)
WTCUR	Curve of water table (z-coordinate) as a function of time
SUCLIM	Suction limit (defined in head, that is, length units). It must not be negative. See Remark 4 .

Remarks:

1. **Pore Water Parts.** This card must be present for all parts having pore water.
2. **Density.** The density on this card is used only to calculate pressure head. To ensure the correct gravity loading, the density of the soil material should be increased to include the mass associated with the pore water.
3. **Analysis Type Curve.** The y -axis values of the curve of analysis type as a function of time can only be 1, 2, 3, or 4, where these values have the same meanings as for ATYPE. During dynamic relaxation, the analysis type will be taken from the first value on the curve
4. **Suction.** The default for SUCLIM is zero, meaning that the pore fluid cannot generate suction. To allow unlimited suction, set this parameter to a large positive number.

*BOUNDARY

*BOUNDARY_PRECRACK

*BOUNDARY_PRECRACK

Purpose: Define pre-cracks in XFEM shell formulations 52 or 54 for purposes of fracture analysis.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	CTYPE	NP					
Type	I	I	I					
Default	none	1	none					

Pre-Crack Point Cards. Include NP cards, one for each point in the pre-crack.

Card 2	1	2	3	4	5	6	7	8
Variable	X	Y	Z					
Type	F	F	F					
Default	none	none	none					

VARIABLE

DESCRIPTION

PID Part ID where the pre-crack is located.

CTYPE Type of pre-crack:
 EQ.1: straight line

NP Number of points defining the pre-crack.

X, Y, Z Coordinates of the points defining the pre-crack. The points should be on or close to the shell surface and the line segments should not pass any nodal points in the part.

***BOUNDARY_PRESCRIBED_ACCELEROMETER_RIGID**

Purpose: Prescribe the motion of a rigid body based on experimental data obtained from accelerometers affixed to the rigid body.

Note: This feature is available starting with LS-DYNA 971R3.

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

Accelerometer Cards. Define one card for each accelerometer affixed to the rigid body. Input is terminated at the next keyword ("*") card. A minimum of three accelerometers are required (see Remarks below).

Card 2	1	2	3	4	5	6	7	8
Variable	NID	CID	LCIDX	LCIDY	LCIDZ			
Type	I	I	I	I	I			
Default	none	none	none	none	none			

VARIABLE**DESCRIPTION**

PID	Part ID for rigid body whose motion is prescribed.
SOLV	Solver type: EQ.1: Gaussian elimination (default) EQ.2: linear regression EQ.3: Reciprocal, inverse explicit time integration
NID	Node ID corresponding to the location of the accelerometer.

VARIABLE	DESCRIPTION
CID	Coordinate system ID describing the orientation of the accelerometer's local axes (see *DEFINE_COORDINATE_NODES). All nodes must reside on the same part. Set FLAG = 1.
LCIDX	Load curve ID containing the local <i>x</i> -acceleration time history from the accelerometer.
LCIDY	Load curve ID containing the local <i>y</i> -acceleration time history from the accelerometer.
LCIDZ	Load curve ID containing the local <i>z</i> -acceleration time history from the accelerometer.

Remarks:

1. **Time Histories.** Acceleration time histories from a minimum of three accelerometers each providing output from three channels are required. Load curves must have the same number of points and data must be uniformly spaced.
2. **Local Axes.** Local axes of the accelerometers must be orthogonal.
3. **SOLV=3.** The following is a theoretical description of SOLV=3.

As a prerequisite we emphasize that the accelerations given by load curves LCIDX, LCIDY and LCIDZ in this keyword is in the *local* system CID which is affixed to the rigid body and therefore rotates with it accordingly. This is *exactly* the data that is output for the *first* node in *ELEMENT_SEATBELT_ACCELEROMETER and therefore this can be used as input here to reproduce a desired motion. It may be worth mentioning that the local system in that other keyword is the local system determined by the nodes involved in the accelerometer and should therefore match the initial configuration of CID in this keyword. In general, sample data from a physical test, or a simulation with a deformable body for that matter, will not provide data that corresponds to an exact rigid body motion. Therefore, the motion cannot be determined uniquely from the data, but SOLV=3 is an option that ensures that *if* the input is exactly a rigid body motion from a previous simulation with full range of sample data (essentially accelerations from every time step), it will reproduce the motion within round-off errors assuming the *same* time step is used in the latter simulation. Since accelerations are used for input, the sensitivity is high and small errors in the data may result in large errors in the resulting motion. It is therefore recommended to use several sample locations in space that span the rigid body, as this will presumably increase the likelihood of getting better results. With this in mind, the theory amounts to establish a relationship between a rigid body node

acceleration and the translational and rotational increments of the rigid body and solve for the unknowns.

An arbitrary point on the rigid body in question has the coordinate x , and can be expressed as

$$x = X + Q\zeta$$

where X is the coordinate of the rigid body center of mass, Q is the orientation of the rigid body, and ζ is a constant vector associated with the relative position of x . Using the explicit time integration scheme for that node, we have

$$\begin{aligned} x_{n+1} &= x_n + \Delta t_{n+\frac{1}{2}} v_{n+\frac{1}{2}} \\ v_{n+\frac{1}{2}} &= v_{n-\frac{1}{2}} + \Delta t_n a_n \end{aligned}$$

where the right-hand side quantities are in general known and $\Delta t_n = \frac{\Delta t_{n+\frac{1}{2}} + \Delta t_{n-\frac{1}{2}}}{2}$. We also have for the rigid body translations

$$X_{n+1} = X_n + \Delta t_{n+\frac{1}{2}} V_{n+\frac{1}{2}}$$

and for the rigid body rotations

$$Q_{n+1} = \Omega(\theta) Q_n$$

where $\Omega(\theta)$ is the orthogonal Hughes-Winget matrix obtained from the rotational increment $\theta = \omega_{n+\frac{1}{2}} \Delta t_{n+\frac{1}{2}}$ of the rigid body. Using these expressions, the updated velocity of a rigid body node is given as

$$\begin{aligned} v_{n+\frac{1}{2}} &= \frac{x_{n+1} - x_n}{\Delta t_{n+\frac{1}{2}}} = \frac{X_{n+1} + Q_{n+1}\zeta - X_n - Q_n\zeta}{\Delta t_{n+\frac{1}{2}}} = V_{n+\frac{1}{2}} + \frac{Q_{n+1} - Q_n}{\Delta t_{n+\frac{1}{2}}} \zeta \\ &= V_{n+\frac{1}{2}} + \frac{\Omega(\theta) - I}{\Delta t_{n+\frac{1}{2}}} Q_n \zeta = V_{n+\frac{1}{2}} + \frac{\Omega(\theta) - I}{\Delta t_{n+\frac{1}{2}}} (x_n - X_n) \end{aligned}$$

and therefore a_n is given by

$$a_n = \frac{v_{n+\frac{1}{2}} - v_{n-\frac{1}{2}}}{\Delta t_n} = \frac{V_{n+\frac{1}{2}} + \frac{\Omega(\theta) - I}{\Delta t_{n+\frac{1}{2}}} (x_n - X_n) - v_{n-\frac{1}{2}}}{\Delta t_n}$$

The intention here is to establish $U = V_{n+\frac{1}{2}} \Delta t_{n+\frac{1}{2}}$ and θ from at least three acceleration points on the rigid body. We first establish U as a function of θ by rewriting the expression for a_n as

$$U = a_n \Delta t_n \Delta t_{n+\frac{1}{2}} + v_{n-\frac{1}{2}} \Delta t_{n+\frac{1}{2}} - (\Omega(\theta) - I)(x_n - X_n).$$

Then assume we have sample points $i = 1, 2, \dots, k$, where k is at least 3, then we can use the mean of these equations to solve for U

$$U = \bar{a}_n \Delta t_n \Delta t_{n+\frac{1}{2}} + \bar{v}_{n-\frac{1}{2}} \Delta t_{n+\frac{1}{2}} - (\Omega(\theta) - I)(\bar{x}_n - X_n)$$

where the bar indicates the mean value of the quantity. Then we insert this expression into the other equations, $i = 1, 2, \dots, k$

$$(\Omega(\theta) - I)(x_n^i - \bar{x}_n) - (a_n^i - \bar{a}_n) \Delta t_n \Delta t_{n+\frac{1}{2}} - \left(v_{n-\frac{1}{2}}^i - \bar{v}_{n-\frac{1}{2}} \right) \Delta t_{n+\frac{1}{2}} = 0$$

which is an overdetermined set of nonlinear equations for θ . So, form

$$f^i(\theta) = (\Omega(\theta) - I)(x_n^i - \bar{x}_n) - (a_n^i - \bar{a}_n)\Delta t_n \Delta t_{n+\frac{1}{2}} - \left(v_{n-\frac{1}{2}}^i - \bar{v}_{n-\frac{1}{2}}\right)\Delta t_{n+\frac{1}{2}}$$

and then minimize

$$f(\theta) = \frac{1}{2} \sum_{i=1}^k f^i(\theta)^T f^i(\theta).$$

The solution to this minimization problem is the extreme point given by

$$\mathbf{0} = \frac{\partial f}{\partial \theta} = \sum_{i=1}^k f^i(\theta)^T \frac{\partial f^i}{\partial \theta}(\theta)$$

which is a nonlinear set of equations. This is solved by a Newton method, and the resulting θ is then used to establish \mathbf{U} . These two increments are then used for determining the motion of the rigid body.

***BOUNDARY_PRESCRIBED_FINAL_GEOMETRY**

Purpose: The final displaced geometry for a subset of nodal points is defined. The nodes of this subset are displaced from their initial positions specified in the *NODE input to the final geometry along a straight line trajectory. A load curve defines a scale factor as a function of time that is bounded between zero and unity corresponding to the initial and final geometry, respectively. A unique load curve can be specified for each node, or a default load curve can apply to all nodes. The external work generated by the displacement field is included in the energy ratio calculation for the glstat file.

Card 1	1	2	3	4	5	6	7	8
Variable	BPFGID	LCIDF	DEATHD					
Type	I	I	F					
Default	0	0	infinity					

Node Cards. The next “*” keyword card terminates this input.

Card 2	1	2	3	4	5	6	7	8	9	10
Variable	NID	X		Y		Z		LCID	DEATH	
Type	I	F		F		F		I	F	
Default	none	0.		0.		0.		LCIDF	infinity	

VARIABLE**DESCRIPTION**

BPFGID

ID for this set of imposed boundary conditions

LCIDF

Default load curve ID. This curve varies between zero and unity.

DEATHD

Default death time. At this time the prescribed motion is inactive and the nodal point is allowed to move freely.

NID

GT.0: Node ID for which the final position is defined. Nodes defined in this section must also appear under the *NODE input.

LT.0: |NID| is a node set ID for which the final projection plane

VARIABLE	DESCRIPTION
	normal to the global z -axis is defined. In this case, only the offset Z value is needed, and all the nodes in this node set are displaced from their initial positions to the projected points on the x - y plane with Z offset. They share the same LCID and DEATH.
X	x -coordinate of final geometry
Y	y -coordinate of final geometry
Z	z -coordinate of final geometry
LCID	Load curve ID for NID. If zero, the default curve ID, LCIDF, is used.
DEATH	Death time for NID. If zero, the default value, DEATHD, is used.

***BOUNDARY_PRESCRIBED_MOTION_OPTION1_{OPTION2}_{OPTION3}**

Available options for *OPTION1* include:

NODE

SET

SET_BOX

SET_SEGMENT

RIGID

RIGID_LOCAL

SET_LINE

POINT_UVW

EDGE_UVW

FACE_XYZ

SET_POINT_UVW

SET_EDGE_UVW

SET_FACE_XYZ

OPTION2 allows an optional ID to be given for the single node, node set, segment set, and rigid body options. If you define a heading, LS-DYNA writes the ID and the heading to the beginning of the ASCII file, bndout.

ID

LS-DYNA only supports *OPTION3* for motion imposed on a rigid body. Thus, *OPTION1* must include RIGID in the name.

BNDOUT2DYNAIN

With this feature, LS-DYNA includes the reaction force as a parameter in the output to the dynain file for use in a subsequent simulation. With this output, you can replace the motion with an equivalent force between simulations without manual intervention. We only support this feature for implicit analysis (IMFLAG = 1 on *CONTROL_IMPLICIT_GENERAL) and only for the lsda format of the dynain file (FTYPE = 3 on *INTERFACE_SPRINGBACK_LSDYNA).

Purpose: Define an imposed nodal motion (velocity, acceleration, or displacement) on a node or a set of nodes. You can also impose velocities and displacements on rigid bodies. If you activate the RIGID_LOCAL option, LS-DYNA prescribes the motion with respect to the local coordinate system for the rigid body; see fields LCO and CID of keywords *MAT_RIGID and *CONSTRAINED_NODAL_RIGID_BODY, respectively. This keyword allows translational nodal velocity and acceleration specifications for rigid body nodes. We describe the application of these motions in [Remark 6](#). For nodes on rigid bodies use the NODE option. Do not use the NODE option in *r*-adaptive problems since the node IDs may change during the adaptive step.

The SET_LINE option causes LS-DYNA to include a generated node set comprising existing nodes and new nodes created from *h*-adaptive mesh refinement along the straight line connecting two specified nodes in prescribed boundary conditions.

The POINT_UVW, EDGE_UVW, FACE_XYZ, SET_POINT_UVW, SET_EDGE_UVW, and SET_FACE_XYZ options are used when prescribed boundary conditions are applied to IGA shells.

Card Summary:

Card ID. Include this card if using the ID keyword option.

ID	HEADING
----	---------

Card 1. This card is required.

TYPEID	DOF	VAD	LCID	SF	VID	DEATH	BIRTH
--------	-----	-----	------	----	-----	-------	-------

Card 2. Include this card for the SET_BOX keyword option.

BOXID	TOFFSET	LCBCHK					
-------	---------	--------	--	--	--	--	--

Card 3. Additional card that is expected if |DOF| = 9, 10, or 11 or VAD = 4 on Card 1; otherwise skip this card.

OFFSET1	OFFSET2	LRB	NODE1	NODE2			
---------	---------	-----	-------	-------	--	--	--

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

NBEG	NEND						
------	------	--	--	--	--	--	--

Card 5. Include this card for the BNDOUT2DYNAIN keyword option.

PRMR							
------	--	--	--	--	--	--	--

Card 6. Include this card for the POINT_UVW, EDGE_UVW, FACE_XYZ, SET_POINT_UVW, SET_EDGE_UVW, and SET_FACE_XYZ keyword options.

FORM	SFD						
------	-----	--	--	--	--	--	--

Data Card Definitions:

ID Card. Additional card for ID keyword option.

Card ID	1	2	3	4	5	6	7	8
Variable	ID	HEADING						
Type	I	A70						

VARIABLE

DESCRIPTION

ID

PRESCRIBED MOTION set ID to which this node, node set, segment set, or rigid body belongs. This ID does not need to be unique.

HEADING

An optional descriptor for the given ID that will be written into the d3hsp file and the bndout file.

Card 1	1	2	3	4	5	6	7	8
Variable	TYPEID	DOF	VAD	LCID	SF	VID	DEATH	BIRTH
Type	I	I	I	I	F	I	F	F
Default	none	none	0	none	1.0	0	10 ²⁸	0.0

VARIABLE

DESCRIPTION

TYPEID

Node ID (NID in *NODE), nodal set ID (SID in *SET_NODE), segment set ID (SID in *SET_SEGMENT, see DOF = 12), part ID (PID in *PART) for a rigid body, parametric point ID (PID in *IGA_POINT_UVW), parametric edge ID (EID in *IGA_EDGE_UVW), physical face ID (FID in *IGA_FACE_XYZ), parametric point set ID (SID in *SET_IGA_POINT_UVW), parametric edge set ID (SID in

VARIABLE	DESCRIPTION
	*SET_IGA_EDGE_UVW) or physical face set ID (SID in *SET_IGA_FACE_XYZ).
DOF	<p>Applicable degrees of freedom:</p> <p>EQ.1: x-translational degree of freedom,</p> <p>EQ.2: y-translational degree of freedom,</p> <p>EQ.3: z-translational degree of freedom,</p> <p>EQ.4: Translational motion in direction given by the VID. Movement on plane normal to the vector is permitted.</p> <p>EQ.-4: Translational motion in direction given by the VID. Movement on plane normal to the vector is not permitted. This option only applies to rigid bodies if $CMO = 2$ on *MAT_RIGID or *CONSTRAINED_NODAL_RIGID_BODY.</p> <p>EQ.5: x-rotational degree of freedom,</p> <p>EQ.6: y-rotational degree of freedom,</p> <p>EQ.7: z-rotational degree of freedom,</p> <p>EQ.8: Rotational motion about an axis, parallel to vector VID, passing through the center-of-gravity of the node, node set, or rigid body (or passing through a specified point when $CMO = 2$ on *MAT_RIGID or *CONSTRAINED_NODAL_RIGID_BODY). Rotation about the normal axes, namely, axes in a plane normal to VID, is permitted.</p> <p>EQ.-8: Rotational motion about an axis, parallel to vector VID, passing through the center-of-gravity of the node, node set, or rigid body (or passing through a specified point when $CMO = 2$ on *MAT_RIGID or *CONSTRAINED_NODAL_RIGID_BODY). Rotation about the normal axes is not permitted. This option only applies to rigid bodies if $CMO = 2$.</p> <p>EQ.9: Rotation with axis parallel to the x-axis and passing through the yz-plane at $y = \text{OFFSET1}$ and $z = \text{OFFSET2}$. Radial motion is NOT permitted. Not applicable to rigid bodies.</p> <p>EQ.-9: Rotation with axis parallel to the x-axis and passing through the yz-plane at $y = \text{OFFSET1}$ and $z = \text{OFFSET2}$. Radial motion is permitted. Not applicable</p>

VARIABLE	DESCRIPTION
	to rigid bodies.
	EQ.10: Rotation with axis parallel to the y -axis and passing through the zx -plane at $z = \text{OFFSET1}$ and $x = \text{OFFSET2}$. Radial motion is NOT permitted. Not applicable to rigid bodies.
	EQ.-10: Rotation with axis parallel to the y -axis and passing through the zx -plane at $z = \text{OFFSET1}$ and $x = \text{OFFSET2}$. Radial motion is permitted. Not applicable to rigid bodies.
	EQ.11: Rotation with axis parallel to the z -axis and passing through the xy -plane at $x = \text{OFFSET1}$ and $y = \text{OFFSET2}$. Radial motion is NOT permitted. Not applicable to rigid bodies.
	EQ.-11: Rotation with axis parallel to the z -axis and passing through the xy -plane at $x = \text{OFFSET1}$ and $y = \text{OFFSET2}$. Radial motion is permitted. Not applicable to rigid bodies.
	EQ.12: Translational motion in direction given by the normals to the segments. Applicable to SET_SEGMENT option only.
VAD	Velocity/Acceleration/Displacement flag: EQ.0: Velocity (rigid bodies and nodes) EQ.1: Acceleration (rigid bodies and nodes) EQ.2: Displacement (rigid bodies and nodes; see Remark 2) EQ.3: Velocity as a function of displacement. This option only applies to rigid bodies with $ \text{CMO} \neq 2$ on *MAT_RIGID or *CONSTRAINED_NODAL_RIGID_BODY. See Remark 3 . EQ.4: Relative displacement. This option only applies to rigid bodies with $ \text{CMO} \neq 2$ on *MAT_RIGID or *CONSTRAINED_NODAL_RIGID_BODY. See Remark 4 .
LCID	Curve ID or function ID to describe motion value as a function of time; see *DEFINE_CURVE, *DEFINE_CURVE_FUNCTION, or *DEFINE_FUNCTION. If LCID refers to *DEFINE_FUNCTION, the function has four arguments: time and x , y and z coordinates of the node or rigid body, such as $f(t, x, y, z) = 10.0 \times t + \max(x - 100, 0)$. If VAD = 2, the function has one argument which is time,

VARIABLE	DESCRIPTION
	such as $f(t) = 10.0 \times t$ (see Remark 2). See BIRTH below.
SF	Load curve scale factor. Default set to 1.0.
VID	Vector ID for DOF values of 4 or 8; see *DEFINE_VECTOR. The direction of this vector is not updated with time.
DEATH	Removal time for the imposed motion/constraint. For alternatives to DEATH and BIRTH for specifying the transitory application of prescribed motion, see the SET_BOX option, or *DEFINE_DEATH_TIMES, or *SENSOR_CONTROL. EQ.0.0: default set to 10^{28}
BIRTH	Activation time for the imposed motion/constraint. The prescribed motion begins acting at time = BIRTH but with the motion from the zero abscissa value of the curve or function (*DEFINE_FUNCTION). In other words, the abscissae are shifted by an amount BIRTH, such that it has the same effect as setting OFFA = BIRTH in *DEFINE_CURVE. Warning: BIRTH is ignored if the LCID is defined as a function, as in *DEFINE_CURVE_FUNCTION.

For the SET_BOX keyword option, define the following additional card.

Card 2	1	2	3	4	5	6	7	8
Variable	BOXID	TOFFSET	LCBCHK					
Type	I	I	I					
Default	none	0	0					

VARIABLE	DESCRIPTION
BOXID	A box ID defining a box region in space in which the constraint is activated. The prescribed motion will <i>only</i> be applied to the nodes falling inside the box. If the LCBCHK field is not defined, the box volume is reevaluated every time step to determine the nodes for which the prescribed motion is active. This reevaluation of the volume is referred to as a "box-check."

VARIABLE	DESCRIPTION
TOFFSET	Time offset flag for the SET_BOX option: EQ.0: No time offset is applied to LCID. EQ.1: The time value of the load curve, LCID, will be offset by the time when the node enters the box.
LCBCHK	Optional load curve allowing more flexible and efficient use of SET_BOX option. Instead of performing box-check at every time step, discrete box-check times could be given as x -values of LCBCHK. LCBCHK's y -values specify corresponding death times. For example, a curve with points (20,30) and (50,70) will result in two box checks. The first will occur at 20, and the prescribed motion will be active from 20 to 30. The second will occur at 50, and the prescribed motion will be active from 50 to 70. A y -value of "0" means the prescribed motion will stay active until next box-check. For example, an additional 3 rd point of (90,0) will lead to another box-check at 90, and the prescribed motion will be active from 90 until the end of the simulation.

Additional card that is expected if $|DOF| = 9, 10, \text{ or } 11$ or $VAD = 4$ on Card 1; otherwise skip this card.

Card 3	1	2	3	4	5	6	7	8
Variable	OFFSET1	OFFSET2	LRB	NODE1	NODE2			
Type	F	F	I	I	I			
Default	0.	0.	0	0	0			

VARIABLE	DESCRIPTION
OFFSET1	Offset for DOF types 9 - 11 (y, z, x directions)
OFFSET2	Offset for DOF types 9 - 11 (z, x, y directions)
LRB	Lead rigid body for measuring the relative displacement (for $VAD = 4$).
NODE1	Optional orientation node, n_1 , for relative displacement (for $VAD = 4$).

*BOUNDARY

*BOUNDARY_PRESCRIBED_MOTION

VARIABLE	DESCRIPTION
NODE2	Optional orientation node, n_2 , for relative displacement (for VAD = 4).

For the SET_LINE keyword option, include the following card.

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

VARIABLE	DESCRIPTION
NBEG	Node ID of a starting node
NEND	Node ID of an ending node. All existing nodes and new nodes generated by h -adaptive mesh refinement along the straight line connecting NBEG and NEND will be included in the prescribed boundary motions.

For the BNDOUT2DYNAIN keyword option, include the following card.

Card 5	1	2	3	4	5	6	7	8
Variable	PRMR							
Type	A							
Default	none							

VARIABLE	DESCRIPTION
PRMR	String representing the name of the parameter to be output to the dynain file. Its value will be the reaction force from the constraint at the end of the simulation. PRMR is only an A9 string because when the dynain file is read the rules for parameters apply (see *PARAMETER for details). Thus, make sure to not use the first character position.

For the POINT_UVW, EDGE_UVW, FACE_XYZ, SET_POINT_UVW, SET_EDGE_UVW and SET_FACE_XYZ keyword options, include the following card (see [Remark 8](#)).

Card 6	1	2	3	4	5	6	7	8
Variable	FORM	SFD						
Type	I	F						
Default	0	1.0						

VARIABLE**DESCRIPTION**

FORM

Formulation type:
EQ.0: Penalty method

SFD

Scale factor for penalty stiffness

Remarks:

1. **Rotational degrees of freedom.** When $DOF = 5, 6, 7$, or 8 , nodal rotational degrees of freedom are prescribed in the case of deformable nodes ($OPTION1 = NODE$ or SET), whereas body rotations are prescribed in the case of a rigid body ($OPTION1 = RIGID$). In the case of a rigid body, the axis of prescribed rotation passes through the body's center of mass, unless $|CMO| = 2$ on $*MAT_RIGID$ or $*CONSTRAINED_NODAL_RIGID_BODY$. For $|DOF| = 8$, the axis of the prescribed rotation is parallel to vector VID. To prescribe a body rotation of a set of deformable nodes, with the axis of rotation parallel to global axes x , y , or z , use $OPTION1 = SET$ with $|DOF| = 9, 10$, or 11 , respectively. The load curve scale factor can be used for simple modifications or unit adjustments.
2. **Limitation for prescribed displacement.** $VAD = 2$ cannot be used if LCID references a $*DEFINE_CURVE_FUNCTION$ which involves anything other than arithmetic functions or the variable "time". This limitation is because velocities must be computed from the displacement curve using an interpolation scheme which requires data for a future time step; those future data are unknown if the curve function involves simulation response.
3. **Velocity as a function of displacement.** For $VAD = 3$, the LCID gives velocity as a function of displacement for the rigid body in the direction specified by DOF.

4. **Relative displacement.** The relative displacement prescribed motion (VAD = 4) can be measured in either of two ways:
 - a) Along a straight line between the mass centers of the rigid bodies.
 - b) Along a vector beginning at node n_1 and terminating at node n_2 .

With the first option, a positive displacement will move the rigid bodies further apart, and, likewise, a negative motion will move the rigid bodies closer together. The mass centers of the rigid bodies must not be coincident when this option is used. With the second option, the relative displacement is measured along the vector, and the rigid bodies may be coincident. Note that the motion of the lead rigid body is not directly affected by VAD = 4, that is, no forces are generated on the lead rigid body.

5. **Activation time.** The activation time, BIRTH, is the time during the solution that the constraint begins to act. Until this time, the prescribed motion card is ignored. The function value of the load curves will be evaluated at the offset time given by the difference of the solution time and BIRTH (solution time - BIRTH). Relative displacements that occur prior to reaching BIRTH are ignored. Only relative displacements that occur after BIRTH are prescribed.
6. **Rigid body nodes.** When the node is on a rigid body, LS-DYNA imposes the translational motion without altering the angular velocity of the rigid body by calculating the appropriate translational velocity for the center of mass of the rigid body using the equation:

$$\mathbf{v}_{\text{cm}} = \mathbf{v}_{\text{node}} - \boldsymbol{\omega} \times (\mathbf{x}_{\text{cm}} - \mathbf{x}_{\text{node}}) .$$

Here \mathbf{v}_{cm} is the velocity of the center of mass, \mathbf{v}_{node} is the specified nodal velocity, $\boldsymbol{\omega}$ is the angular velocity of the rigid body, \mathbf{x}_{cm} is the current coordinate of the mass center, and \mathbf{x}_{node} is the current coordinate of the nodal point. Extreme care must be used when prescribing motion of a rigid body node. Typically, for nodes on a given rigid body, the motion of no more than one node should be prescribed or unexpected results may be obtained.

7. **Rigid body rotations.** When the RIGID option is used to prescribe rotation of a rigid body, the axis of rotation will, unless |CMO| = 2, be shifted such that it passes through the center of mass of the rigid body. To override the motion around the internally-calculated center-of-mass location, set |CMO| = 2 (see *MAT_RIGID and *CONSTRAINED_NODAL_RIGID_BODY) or use *PART-INERTIA or *CONSTRAINED_NODAL_RIGID_BODY_INERTIA.

When the RIGID_LOCAL option is invoked, the orientation of the local coordinate system rotates with time in accordance with rotation of the rigid body. The local coordinate system is specified with LCO and CID in *MAT_RIGID and *CONSTRAINED_NODAL_RIGID_BODY, respectively. If LCO/CID is 0, the

Angular displacements are applied in an incremental fashion; hence, a successive set of rotations about multiple axes cannot be correctly prescribed. In light of this, the command `*BOUNDARY_PRESCRIBED_ORIENTATION_RIGID` should be used for the purpose of prescribing the general orientation of a rigid body.

- For boundary conditions to be applied to IGA shells, the point and edge IDs specified in TYPEID have to be included in the parametric point set (PSID) and parametric edge set (ESID) in *IGA_FACE_XYZ of the corresponding *IGA_SHELL. This does not apply to the edges already in *IGA_1D_BREP.

When boundary conditions are prescribed for physical faces and parametric edges, the boundary conditions will be directly imposed on the control points if these faces are untrimmed and these edges are the actual edges of the patch. In this case, FORM and SFD are ignored.

[illegible]

```

$
$$$$ *BOUNDARY_PRESCRIBED_MOTION_RIGID
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$ A rigid body is given a prescribed rotational displacement about the
$ z-axis according to a specified displacement-time curve.
$
*BOUNDARY_PRESCRIBED_MOTION_RIGID
$
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$      pid      dof      vad      lcid      sf      vid      death
$      84        7        2        9        14.0
$
$      pid = 84    apply motion to part number 84
$      dof = 7     rotation is prescribed about the z-axis
$      vad = 2     the prescribed motion is displacement (angular)
$      lcid = 9    rotation follows load curve 9, requires a *DEFINE_CURVE
$                  (rotation should be in radians)
$      sf         use default (sf = 1.0)
$      vid        not used in this example
$      death = 14  prescribed motion is removed at 14 ms (assuming time is in ms)
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$

```

An example using the SET_LINE option is shown below. The example is of a flat plate which is fixed on one edge and pulled using a velocity boundary condition on the opposite edge as indicated in [Figure 5-4](#). The velocity boundary condition is applied to the line connecting nodes 98 and 105 which includes the nodes created from *h*-adaptive mesh refinement on that line.

```

$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
*BOUNDARY_PRESCRIBED_MOTION_SET_LINE
$#      nsid      dof      vad      lcid      sf      vid      death      birth
$      122        3        0        2
$      NBEG      NEND
$      98        105
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
*DEFINE_CURVE
2
          0.0          &velo
&endtime          &velo
          1000.0       &velo

```

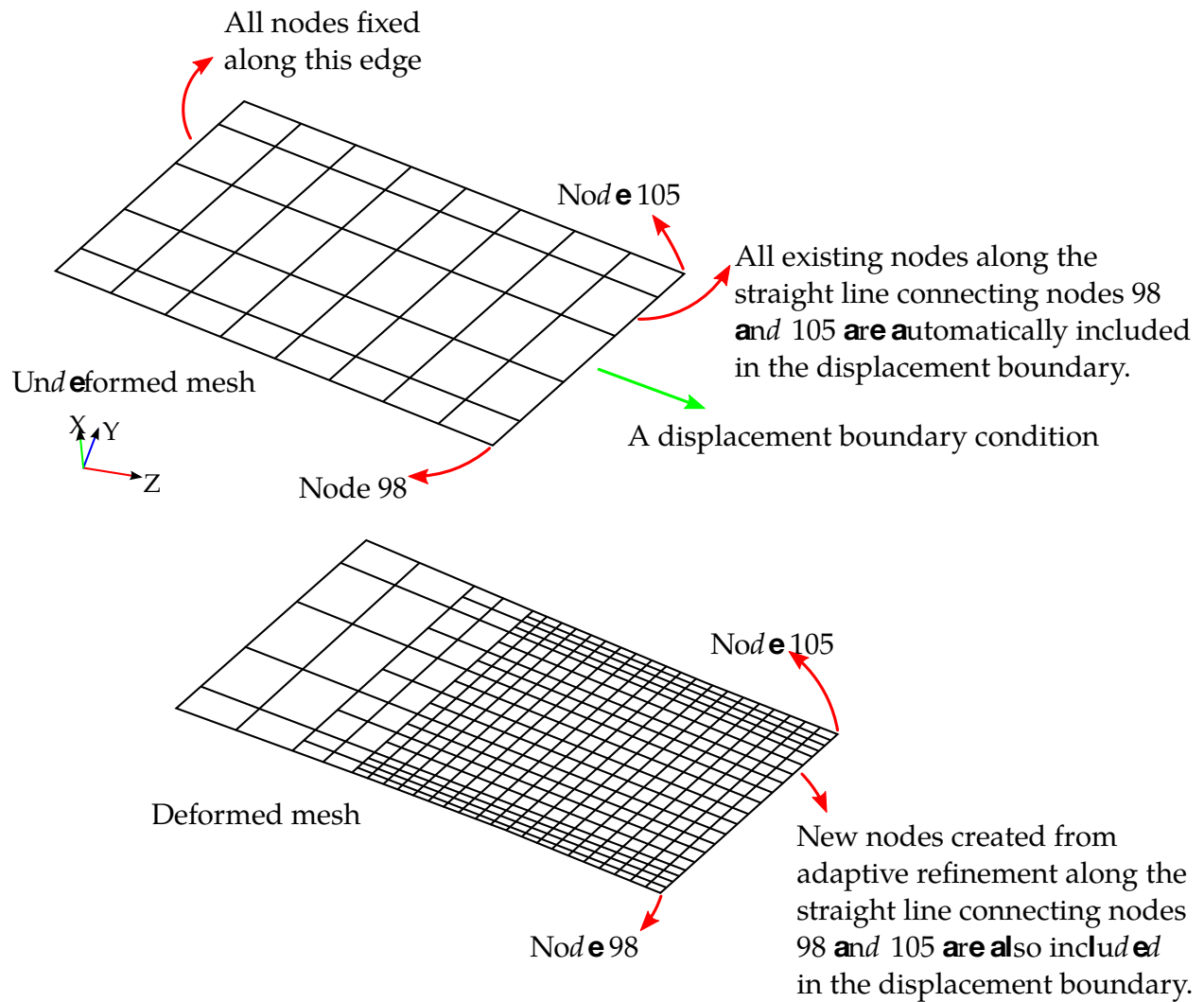


Figure 5-4. Example using the SET_LINE option.

*BOUNDARY

*BOUNDARY_PRESCRIBED_ORIENTATION_RIGID

*BOUNDARY_PRESCRIBED_ORIENTATION_RIGID_OPTION1_{OPTION2}

Available options for *OPTION1* include:

DIRCOS

ANGLES

EULERP

VECTOR

OPTION2 allows an optional ID:

ID

The defined ID can be referred to by *SENSOR_CONTROL.

Purpose: Prescribe the orientation of rigid body as a function of time.

Card Summary:

Card ID. Include this card if using the ID option.

ID	HEADING
----	---------

Card 1. This card is required.

PIDB	PIDA	INTRP	BIRTH	DEATH	TOFFSET		
------	------	-------	-------	-------	---------	--	--

Card 2a.1. Include this card if using the DIRCOS option.

LCIDC11	LCIDC12	LCIDC13	LCIDC21	LCIDC22	LCIDC23	LCIDC31	LCIDC32
---------	---------	---------	---------	---------	---------	---------	---------

Card 2a.2. Include this card if using the DIRCOS option.

	LCIDC33						
--	---------	--	--	--	--	--	--

Card 2b. Include this card if using the ANGLES option.

LCIDQ1	LCIDQ2	LCIDQ3	ISEQ	ISHFT	BODY		
--------	--------	--------	------	-------	------	--	--

Card 2c. Include this card if using the EULERP option.

LCIDE1	LCIDE2	LCIDE3	LCIDE4				
--------	--------	--------	--------	--	--	--	--

Card 2d. Include this card if using the VECTOR option.

LCIDV1	LCIDV2	LCIDV3	LCIDS	VALSPIN			
--------	--------	--------	-------	---------	--	--	--

Data Cards:

ID Card. Optional card for ID keyword option.

Card ID	1	2	3	4	5	6	7	8
Variable	ID	HEADING						
Type	I	A70						

VARIABLE

DESCRIPTION

ID

Optional ID for PRESCRIBED ORIENTATION that can be referred to by *SENSOR_CONTROL. When not defined, the sequential definition order will be used as ID when referred to by *SENSOR_CONTROL.

HEADING

An optional descriptor for the given ID

Card 1	1	2	3	4	5	6	7	8
Variable	PIDB	PIDA	INTRP	BIRTH	DEATH	TOFFSET		
Type	I	I	I	F	F	I		
Default	none	↓	1	0.	10 ²⁰	0		

VARIABLE

DESCRIPTION

PIDB

Part ID for rigid body B whose orientation is prescribed

PIDA

Part ID for rigid body A. The orientation of PIDB is measured with respect to the coordinate system of PIDA, as defined by LCO on *MAT_RIGID. If zero, then the orientation of PIDB is measured with respect to the global reference frame except for when BODY = 1 for the ANGLES option; see [Remark 5](#).

VARIABLE	DESCRIPTION
INTRP	Interpolation method used on time history curves: EQ.1: Linear interpolation (default)
BIRTH	Prior to this time the body moves freely under the action of other agents.
DEATH	The body is freed at this time and subsequently allowed to move under the action of other agents.
TOFFSET	Time offset flag: EQ.0: No time offset is applied. EQ.1: The time value of all load curves will be offset by the birth time.

Cosine Card 1. Additional card for DIRCOS option.

Card 2a.1	1	2	3	4	5	6	7	8
Variable	LCIDC11	LCIDC12	LCIDC13	LCIDC21	LCIDC22	LCIDC23	LCIDC31	LCIDC32
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

Cosine Card 2. Additional card for DIRCOS option.

Card 2a.2	1	2	3	4	5	6	7	8
Variable	LCIDC33							
Type	I							
Default	none							

VARIABLE	DESCRIPTION
LCIDC $_{ij}$	Load curve ID specifying direction cosine C_{ij} as a function of time. C_{ij} is defined as:

VARIABLE

DESCRIPTION

$$C_{ij} = \mathbf{a}_i \cdot \mathbf{b}_j$$

where the $\{\mathbf{a}_i\}$ are mutually perpendicular unit vectors fixed in PIDA and the $\{\mathbf{b}_j\}$ are mutually perpendicular unit vectors fixed in PIDB. If PIDA = 0 then the $\{\mathbf{a}_i\}$ are unit vectors aligned with the global x , y , and z . See [Remark 1](#).

Angles Card. Additional card for ANGLES option, see [Remark 5](#).

Card 2b	1	2	3	4	5	6	7	8
Variable	LCIDQ1	LCIDQ2	LCIDQ3	ISEQ	ISHFT	BODY		
Type	I	I	I	I	I	I		
Default	none	none	none	none	1	0		

VARIABLE

DESCRIPTION

LCIDQ i

Load curve ID specifying the orientation angle q_i in radians as a function of time. See [Remark 1](#).

ISEQ

Specifies the sequence in which the rotations are performed. In this first set of sequences, three unique axes are involved. This sequence is associated with what are commonly called Cardan or Tait-Bryan angles. *All angles must be in units of radians.* Whether these rotations are intrinsic or extrinsic is determined by the BODY field.

EQ.123: The first rotation is performed about the x -axis through an angle of q_1 , the second about the y -axis through an angle of q_2 , and the third about the z -axis through an angle of q_3 .

EQ.231: The first rotation is performed about the y -axis through an angle of q_1 , the second about the z -axis through an angle of q_2 , and the third about the x -axis through an angle of q_3 .

EQ.312: The first rotation is performed about the z -axis through an angle of q_1 , the second about the x -axis through an angle of q_2 , and the third about the y -axis through an angle of q_3 .

VARIABLE	DESCRIPTION
	EQ.132: The first rotation is performed about the x -axis through an angle of q_1 , the second about the z -axis through an angle of q_2 , and the third about the y -axis through an angle of q_3 .
	EQ.213: The first rotation is performed about the y -axis through an angle of q_1 , the second about the x -axis through an angle of q_2 , and the third about the z -axis through an angle of q_3 .
	EQ.321: The first rotation is performed about the z -axis through an angle of q_1 , the second about the y -axis through an angle of q_2 , and the third about the z -axis through an angle of q_3 .
	The second set of sequences involve only two unique axes where the first and third are repeated. This sequence is associated with what are commonly called Euler angles.
	EQ.121: The first rotation is performed about the x -axis through an angle of q_1 , the second about the y -axis through an angle of q_2 , and the third about the x -axis through an angle of q_3 .
	EQ.131: The first rotation is performed about the x -axis through an angle of q_1 , the second about the z -axis through an angle of q_2 , and the third about the x -axis through an angle of q_3 .
	EQ.212: The first rotation is performed about the y -axis through an angle of q_1 , the second about the x -axis through an angle of q_2 , and the third about the y -axis through an angle of q_3 .
	EQ.232: The first rotation is performed about the y -axis through an angle of q_1 , the second about the z -axis through an angle of q_2 , and the third about the y -axis through an angle of q_3 .
	EQ.313: The first rotation is performed about the z -axis through an angle of q_1 , the second about the x -axis through an angle of q_2 , and the third about the z -axis through an angle of q_3 .
	EQ.323: The first rotation is performed about the z -axis through an angle of q_1 , the second about the y -axis through an angle of q_2 , and the third about the z -axis through an angle of q_3 .

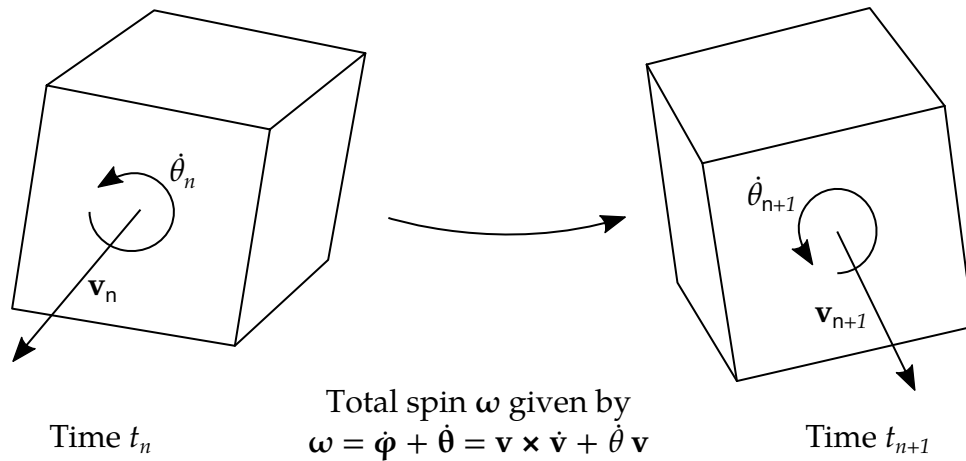


Figure 5-5. Orientation of rigid body computed using the VECTOR option.

VARIABLE	DESCRIPTION
ISHFT	<p>Angle shift.</p> <p>EQ.1: Angle curves are unaltered.</p> <p>EQ.2: Shifts angle data in the LCIDQ<i>i</i> curves as necessary to eliminate discontinuities. If angles are confined to the range $[-\pi, \pi]$ and the data contains excursions exceeding π, then set ISHFT = 2.</p>
BODY	<p>Reference axes:</p> <p>EQ.0: Rotations are performed about axes fixed to PIDA (extrinsic rotation, default). The coordinate system in question is the one defined by LCO on *MAT_RIGID for body A.</p> <p>EQ.1: Rotations are performed about axes fixed to PIDB (intrinsic rotation). The coordinate system in question is the one defined by LCO on *MAT_RIGID for body B.</p>

Euler Parameter Card. Additional card for EULERP option.

Card 2c	1	2	3	4	5	6	7	8
Variable	LCIDE1	LCIDE2	LCIDE3	LCIDE4				
Type	I	I	I	I				
Default	none	none	none	none				

VARIABLE	DESCRIPTION
LCIDE i	<p>Load curve ID specifying Euler parameter e_i as a function of time. See Remark 1. The Euler parameters are defined as follows:</p> $\varepsilon_i = \varepsilon \cdot \mathbf{a}_i = \varepsilon \cdot \mathbf{b}_i, \quad (i = 1, 2, 3)$ $\varepsilon_4 = \cos\left(\frac{\theta}{2}\right)$ <p>where ε is the Euler vector, $\{\mathbf{a}_i\}$ and $\{\mathbf{b}_i\}$ are dextral sets of unit vectors fixed in PIDA and PIDB, respectively, and θ (in radians) is associated with the rotation of PIDB in PIDA about Euler vector. If PIDA = 0, then the $\{\mathbf{a}_i\}$ are unit vectors aligned, respectively, with the global x, y, and z-axes. See Remark 2.</p>

Vector Card. Additional card for VECTOR option.

Card 2d	1	2	3	4	5	6	7	8
Variable	LCIDV1	LCIDV2	LCIDV3	LCIDS	VALSPIN			
Type	I	I	I	I	F			
Default	none	none	none	0	0.			

VARIABLE	DESCRIPTION
LCIDV i	<p>Load curve ID specifying the vector measure number v_i as a function of time. See Remark 1. The vector measure numbers are defined as follows:</p> $v_i = \mathbf{v} \cdot \mathbf{n}_i, \quad i = 1, 2, 3.$ <p>where \mathbf{v} is a vector and $\{\mathbf{n}_i\}$ are unit vectors aligned, respectively, with the global axes x, y, and z-axes. Note that the vector \mathbf{v} is attached to the body in question, so changing the direction of this vector will induce a rotation of the body defined by $\dot{\boldsymbol{\phi}} = \mathbf{v} \times \dot{\mathbf{v}}$. See Figure 5-5.</p>
LCIDS	<p>Load curve ID which specifies the overlaid spin speed $\dot{\theta}$ of PIDB about the axis parallel to the vector \mathbf{v}.</p> <p>EQ.0: A constant spin speed as defined by VALSPIN is used.</p> <p>GT.0: Load curve for spin speed (radians per unit time)</p>

VARIABLE	DESCRIPTION
VALSPIN	Value for the constant spin speed of PIDB (radians per unit time $\dot{\theta}$). This field is ignored if the load curve number defined above is non-zero.

Remarks:

1. **Load Curves.** All load curves must contain the same number of points and the data must be uniformly spaced.
2. **Euler Parameters.** The Euler parameters are defined as

$$\boldsymbol{\varepsilon} = \sin\left(\frac{\theta}{2}\right) \mathbf{n}$$

$$\varepsilon_4 = \cos\left(\frac{\theta}{2}\right)$$

where \mathbf{n} is a unit vector defining the axis of rotation and θ is the angle with which the rotation occurs. Consequently, the four parameters are subjected to the condition

$$\boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} + \varepsilon_4^2 = 1.$$

It is therefore recommended that the control points of the curves already fulfil this or else LS-DYNA will internally normalize these values. From the Euler parameters at time t , a unique rotation matrix \mathbf{Q}_t is computed that is used to determine the total orientation \mathbf{Q} . The rotation is performed with respect to the *reference state* \mathbf{Q}_0 given by the Euler parameters at time 0. In general, $\mathbf{Q}_0 \neq \mathbf{I}$ and the rotation of the rigid body is given by $\mathbf{Q} = \mathbf{Q}_t \mathbf{Q}_0^T$. If the parameters are initially $\boldsymbol{\varepsilon} = \mathbf{0}$ and $\varepsilon_4 = 1$, then the reference state is $\mathbf{Q}_0 = \mathbf{I}$ and $\mathbf{Q} = \mathbf{Q}_t$ defines the orientation of the rigid body.

For a nonzero PIDA, the rotation matrix \mathbf{Q} as defined above is expressed in a system that is fixed in rigid body A. If this system is denoted \mathbf{R}_t at time t , and assuming $\mathbf{R}_0 = \mathbf{I}$, the orientation with respect to a global system is \mathbf{RQ} .

3. **Coordinate System.** LC0 in *MAT_RIGID must be used to identify a coordinate system for each rigid body. The coordinate system must be defined with *DEFINE_COORDINATE_NODES and FLAG = 1. Nodes used in defining the coordinate system must reside on the same body.
4. **Incompatibilities.** This feature is incompatible with *DEFINE_CURVE_FUNCTION.
5. **ANGLES Option.** For the ANGLES option we assume that bodies A and B are defined by local systems attached to the bodies, \mathbf{R}_a for body A and consequently

\mathbf{R}_b for body B , and we intend to determine the latter. The initial configurations are determined by the parameter LCO on each respective *MAT_RIGID card. If this parameter is absent, the initial system is taken as the configuration of principal inertia directions, so we recommend defining it whenever PIDA or BODY are nonzero. We denote the initial ($t = 0$) configuration matrices for each body \mathbf{R}_A and \mathbf{R}_B , respectively. We, also, define rotation matrices \mathbf{Q}_i , $i = 1, 2, 3$ as functions of time according to the formula

$$\mathbf{Q}_i = \cos q_i (\mathbf{u}_i \mathbf{u}_i^T + \mathbf{v}_i \mathbf{v}_i^T) + \sin q_i (\mathbf{v}_i \mathbf{u}_i^T - \mathbf{u}_i \mathbf{v}_i^T) + \mathbf{w}_i \mathbf{w}_i^T.$$

Here $\{\mathbf{u}_i, \mathbf{v}_i, \mathbf{w}_i\}$ is an arbitrary orthonormal system with \mathbf{w}_i being the rotation direction associated with the parameter ISEQ. *This system is fixed in time.* Furthermore, q_i is the rotation angle associated with LCIDQ i and is thus time dependent. More specifically, for a “1” in ISEQ we have $\mathbf{w}_i^T = (1 \ 0 \ 0)$, for a “2” we have $\mathbf{w}_i^T = (0 \ 1 \ 0)$, and for a “3” we have $\mathbf{w}_i^T = (0 \ 0 \ 1)$. Depending on the values of BODY and PIDA, we define the configuration of body B at time t as

$$\mathbf{R}_b = \begin{cases} \mathbf{Q}_3 \mathbf{Q}_2 \mathbf{Q}_1 \mathbf{R}_B & \text{if BODY} = 0 \text{ and PIDA} = 0 \\ \mathbf{R}_A \mathbf{Q}_3 \mathbf{Q}_2 \mathbf{Q}_1 \mathbf{R}_A^T \mathbf{R}_B & \text{if PIDA} \neq 0 \\ \mathbf{R}_B \mathbf{Q}_1 \mathbf{Q}_2 \mathbf{Q}_3 & \text{if BODY} = 1 \text{ and PIDA} = 0 \end{cases}$$

***BOUNDARY_PRESSURE_OUTFLOW_OPTION**

Available options include:

SEGMENT

SET

Purpose: Define pressure outflow boundary conditions. These boundary conditions are attached to solid elements using the Eulerian ambient formulation (refer to ELFORM in *SECTION_SOLID_ALE) and defined to be pressure outflow ambient elements (refer to AET in *SECTION_SOLID_ALE).

Card 1 for SET option.

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

Card 1 for SEGMENT option.

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

VARIABLE**DESCRIPTION**

SSID

Segment set ID

N1, N2, ...

Node ID's defining segment

***BOUNDARY_PWP_OPTION1_{OPTION2}**

Purpose: Define pressure boundary conditions for pore water, such as at soil surface. The TABLE/TABLE_SET option applies to a whole part/part set, while the other options apply to specified nodes.

OPTION1 is required. The available choices for *OPTION1* are:

NODE

SET

TABLE

TABLE_SET

OPTION2 is optional. It allows an optional ID to be given that applies to this boundary condition:

ID

If defined, the ID may be referenced by *SENSOR_CONTROL to activate and/or deactivate the boundary condition. The ID has no other function in LS-DYNA.

Card Summary:

Card ID. This card is included if *OPTION2* is ID

ID							
----	--	--	--	--	--	--	--

Card 1a. This card is included if *OPTION1* is NODE or SET.

NID	LC	CMULT	LCDR	TBIRTH	TDEATH		
-----	----	-------	------	--------	--------	--	--

Card 1b. This card is included if *OPTION1* is TABLE or TABLE_SET.

PID				TBIRTH	TDEATH		
-----	--	--	--	--------	--------	--	--

Card 2a. This card is included if *OPTION1* is NODE or SET

IPHRE	ITOTEX	IDRFLAG		LCLEAK	CLEAK	LCPUMP	TYPPMP
-------	--------	---------	--	--------	-------	--------	--------

Card 2b. This card is included if *OPTION1* is TABLE or TABLE_SET

	ITOTEX		TABLE				
--	--------	--	-------	--	--	--	--

Data Card Definitions:**ID Card.** Additional card for the ID keyword option.

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

VARIABLE**DESCRIPTION**

ID

ID for this boundary condition

Node Card. This card is included if the keyword *OPTION1* is NODE or SET.

Card 1a	1	2	3	4	5	6	7	8
Variable	NID	LC	CMULT	LCDR	TBIRTH	TDEATH		
Type	I	F	F	I	F	F		
Default	none	none	0.0	none	0.0	10 ²⁰		

VARIABLE**DESCRIPTION**

NID

Node ID (option = NODE) or node set ID (option = SET)

LC

Load curve or function giving pore water pressure head (length units) as a function of time (see [Remark 5](#)).

EQ.0: Constant pressure head assumed equal to CMULT.

CMULT

Factor on curve or constant pressure head if LC = 0

LCDR

Load curve or function giving pore water pressure head during dynamic relaxation.

EQ.0: During dynamic relaxation, use first pressure head value on LC.

VARIABLE	DESCRIPTION
TBIRTH	Time at which boundary condition becomes active
TDEATH	Time at which boundary condition becomes inactive

Table Card. This card is included if the keyword *OPTION1* is TABLE or TABLE_SET.

Card 1b	1	2	3	4	5	6	7	8
Variable	PID				TBIRTH	TDEATH		
Type	I				F	F		
Default	none				0.0	10 ²⁰		

VARIABLE	DESCRIPTION
PID	Part ID (option = TABLE) or part set ID (option = TABLE_SET)
TBIRTH	Time at which boundary condition becomes active
TDEATH	Time at which boundary condition becomes inactive

This card is included if the keyword *OPTION1* is NODE or SET.

Card 2a	1	2	3	4	5	6	7	8
Variable	IPHRE	ITOTEX	IDRFLAG		LCLEAK	CLEAK	LCPUMP	TYPMP
Type	I	I	I		I	F	I	I
Default	0	0	0		0	0.0	optional	0

VARIABLE	DESCRIPTION
IPHRE	Phreatic behavior option: EQ.0: Default behavior EQ.1: For phreatic behavior (water can be removed by the boundary condition but not added, such as at a sloping free surface). See Remark 7 .

VARIABLE	DESCRIPTION
ITOTEX	Flag for the type of pressure boundary condition (see Remark 8): EQ.0: Total head EQ.1: Excess head EQ.2: Hydraulic head EQ.4: z-coord where head = 0 (piezometric level)
IDRFLAG	Active flag: EQ.0: Active only in transient analysis EQ.1: Active only in dynamic relaxation EQ.2: Active in all analysis phases
LCLEAK	Optional load curve ID (see *DEFINE_CURVE) giving the area of the hole through which pore fluid leaks to the boundary condition. See Remark 9 .
CLEAK	Discharge coefficient, applicable when LCLEAK is nonzero
LCPUMP	Optional load curve ID (see *DEFINE_CURVE) giving the volumetric outflow rate per node or the upper limit on the volume flow rate, depending on the choice of TYPPMP. The curve x -axis is time while the y -axis is in units of volume per unit time. See Remarks 11 and 12 .
TYPPMP	Determines the action of LCPUMP (see Remarks 11 and 12): EQ.0: LCPUMP defines the volume flow rate to the boundary condition. LC, CVAL, and ITOTEX are ignored. EQ.1: LCPUMP defines an upper limit on volume flow rate to the boundary condition; only outflow is permitted. EQ.2: Same as 1, except that inflow is permitted as well as outflow.

This card is included if the keyword *OPTION1* is TABLE or TABLE_SET.

Card 2b	1	2	3	4	5	6	7	8
Variable		ITOTEX		TABLE				
Type		I		I				
Default		0		0				

VARIABLE**DESCRIPTION**

ITOTEX	Flag for type of pressure boundary condition (see Remark 8): EQ.0: Total head EQ.1: Excess head EQ.2: Hydraulic head
TABLE	Table ID for TABLE/TABLE_SET option only. See Remark 6 .

Remarks:

1. **Pressure Dimensions.** Pressure should be given as pressure head in length units. Pressure head is defined as $\text{pressure}/\rho g$ where ρ is the fluid density given on *BOUNDARY_PORE_FLUID and g is the acceleration due to gravity given on *CONTROL_PORE_FLUID.
2. **Applicability of NODE and SET options.** The NODE and SET options should be applied only to nodes belonging to parts whose analysis type (see ATYPE and ACURVE on *BOUNDARY_PORE_FLUID) is Undrained, Time-dependent consolidation or Steady-state consolidation. The NODE and SET options have no effect on the pore pressure in Drained parts.
3. **Applicability of TABLE and TABLE_SET options.** The TABLE and TABLE_SET options should be used *only* with Drained parts. They have no effect on the pore pressure in parts with other pore fluid analysis types.
4. **Nodal Boundary Conditions.** *BOUNDARY_PWP_NODE or SET overrides pressure head from *BOUNDARY_PWP_TABLE or TABLE_SET at nodes where both are present. This situation can arise at the boundary between Drained and Undrained parts.

5. **Input Arguments for LC Function.** If LC is a *DEFINE_FUNCTION, the input arguments are (time, x , y , z , $x0$, $y0$, $z0$), where x , y , and z are the current coordinates and $x0$, $y0$, and $z0$ are the initial coordinates of the node.
6. **TABLE and TABLE_SET Options.** The table consists of a list of times in ascending order, followed immediately by curves of z -coordinate versus pore pressure head. Each curve represents the pore water pressure head distribution with z -coordinate at the corresponding time. There must be the same number of curves as time values, arranged immediately after the *DEFINE_TABLE and in the correct order to correspond to the time values. Each curve should be arranged in ascending order of z -coordinate – they look upside-down on the page. The z -coordinate is the x -axis of the curve, the pore water pressure head (in length units) is the y -axis. Each curve should have the same z -coordinates (x -values) and use the same value of LCINT. Ensure that the range of z -coordinates in the curve exceeds by at least 5% the range of z -coordinates of the nodes belonging to the parts to which the boundary condition is applied.
7. **IPHRE.** “Phreatic” means that water can be removed by the boundary condition but not added. The boundary condition enforces the pressure head being less than or equal to zero. This condition occurs when the free surface of the soil is sloping so that any water emerging from the soil runs away down the slope. When IPHRE is nonzero, the input fields LC, CMULT, LCDR and ITOTEX are ignored. See also [Remark 9](#) below.
8. **ITOTEX.** Descriptions for the different types of pressure conditions are as follows:
 - a) For ITOTEX = 0, the value from the curve or table is total head. This may be used with any pore pressure analysis type.
 - b) For ITOTEX = 1, the value from the curve or table is excess head. Total head will be determined by adding the hydrostatic head. This option cannot be used with drained analysis, which sets excess head to zero.
 - c) For ITOTEX = 2, the value from the curve or table is hydraulic head, to which excess head may be added due to volume change in the soil if the analysis type is not drained.
 - d) For ITOTEX = 4, the curve value is the z -coordinate of the water surface; pore pressure head at any node in this boundary condition, $h_{BC,node}$, is given by,

$$h_{BC,node} = z_{\text{surface}} - z_{\text{node}}$$

This option allows a single boundary condition to be used for nodes at any depth, provided that the pressure distribution is hydrostatic below the

given surface. This option is not available for the TABLE and TABLE_SET options.

9. **Leakage through a hole.** If LCLEAK is nonzero, the boundary condition models flow through a hole with the pressure head outside the hole, h_{ext} , determined by LC, CVAL, and ITOTEX. The size of the hole and the pressure head difference across the hole determine the flow rate. When h_{ext} is zero or negative (e.g., if LC, CVAL, and ITOTEX are left blank), the space outside the hole is assumed to be dry. In this case, if the pressure head at the node is positive, fluid flows out to the boundary condition. However, if the pressure head at the node is negative, no inflow can occur, and the boundary condition has no effect. In contrast, when h_{ext} is greater than zero, the space outside the hole is assumed to be a fluid-filled reservoir. Thus, either outflow or inflow occurs, depending on whether the pressure head at the node, h , is greater than or less than h_{ext} .

At time t , the y -axis value of LCLEAK, $A(t)$, is the area of the hole, while CLEAK is a non-dimensional discharge coefficient, C . LCLEAK should be defined with time on the x -axis and hole area on the y -axis. For the SET option, the same hole area, $A(t)$, and discharge coefficient, C , are applied to every node in the set. Therefore, the total leakage area is $nA(t)$ where n is the number of nodes in the set.

When $h > h_{\text{ext}}$, the volume outflow rate from the node, \dot{Q} , is given by:

$$\dot{Q} = CA(t)\sqrt{2(h - h_{\text{ext}})g}$$

This rate is subject to h being above the suction limit (see SUCLIM on [*BOUNDARY_PORE_FLUID](#)). If h is at or below the suction limit, then $\dot{Q} = 0$.

When $h < h_{\text{ext}}$, the volume flow rate (with inflow being treated as negative) is:

$$\dot{Q} = \begin{cases} -CA(t)\sqrt{2(h_{\text{ext}} - h)g} & \text{if } h_{\text{ext}} > 0 \\ 0 & \text{if } h_{\text{ext}} \leq 0 \end{cases}$$

10. **Activation by Sensor.** Optionally, *BOUNDARY_PWP_NODE_ID or *BOUNDARY_PWP_SET_ID can be activated by *SENSOR_CONTROL with TYPE set to BPWPN. In this case the sensor overrides TBIRTH and TDEATH.
11. **Using LCPUMP to model pumping water out.** When used with TYPPMP set to zero, LCPUMP represents situations where water is pumped out of the soil at a known rate. Instead of constraining the pore pressure, the boundary condition removes water from the nodes to which it applies at a prescribed rate. The y -axis of the curve gives the outflow rate (volume per unit time) at each node to which this boundary condition applies. Water is removed by the boundary condition only while the pressure head at the affected node is above the suction limit (see SUCLIM on [*BOUNDARY_PORE_FLUID](#)). For nodes at the suction limit,

the boundary condition has no effect. When TYPPMP is zero and LCPUMP is nonzero, LC, CVAL, and ITOTEX are ignored.

12. **Using LCPUMP to limit flow rate to the boundary condition.** When used with TYPPMP set to 1 or 2, LCPUMP defines an upper limit on the volume flow rate between each node and the boundary condition. The boundary condition pressure head, $h_{BC, node}$, is defined by LC, CVAL, and ITOTEX. The pressure head at the node attempts to follow $h_{BC, node}$ subject to (a) the maximum volume flow rate to or from the boundary condition not exceeding the current value of LCPUMP, and (b) the pressure head at the node not falling below the suction limit (see SUCLIM on [*BOUNDARY_PORE_FLUID](#)). For TYPPMP set to 1, only outflow is permitted. For TYPPMP set to 2, both outflow and inflow are permitted. The values on the y-axis of LCPUMP should be input as positive, and are treated as limiting the absolute value of the volume flow rate per node to or from the boundary condition.

***BOUNDARY_PZEPOT**

Purpose: Define prescribed electric potential on a node set for a piezoelectric material; see *MAT_ADD_PZELECTRIC.

Include as many cards as necessary. This input ends at the next keyword (**) card.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	NSID	LCID	SF				
Type	I	I	I	I				
Default	none	none	0	none				

VARIABLE**DESCRIPTION**

ID	ID of this boundary condition, which can be referred to by *SENSOR_CONTROL with TYPE='PZBC' or *DEFINE_CURVE_FUNCTION with FUNCTION='ECHGBC'
NSID	Node set ID, see *SET_NODE
LCID	Load curve giving prescribed electric potential as a function of time
SF	Scale factor on curve or constant electric potential if LCID = 0.

***BOUNDARY_RADIATION_OPTION1_{OPTION2}_{OPTION3}**

Available values for *OPTION1* include:

SET

SEGMENT

ENCLOSURE

Available values for *OPTION2* include (available if *OPTION1* is **not** ENCLOSURE):

VF_READ

VF_CALCULATE

<BLANK>

Available values for *OPTION3* include (available if *OPTION1* is **not** ENCLOSURE):

RESTART

<BLANK>

OPTION1 specifies radiation boundary surface definition by a surface set (SET) or by a segment list (SEGMENT). The option ENCLOSURE can be used to transfer heat between segments in an enclosure. The ENCLOSURE option is available in LS-DYNA MPP only.

OPTION2 indicates the radiation boundary surface is part of an enclosure. When set to VF *OPTION2* specifies the use of view factors. The suffix, READ, indicates that the view factors should be read from the file "viewfl". The suffix, CALCULATE, indicates that the view factors should be calculated. The Stefan Boltzmann constant must be defined for radiation in an enclosure on the *CONTROL_THERMAL_SOLVER keyword. The parameter DTVF entered on the CONTROL_THERMAL_SOLVER keyword defines the time interval between VF updates for moving geometries.

OPTION3 is the keyword suffix RESTART. This is only applicable in combination with the keyword VF_CALCULATE. In very long runs, it may be necessary to halt execution. This is accomplished by entering Ctrl-C followed by sw1. To restart the view factor calculation, add the suffix RESTART to all VF_CALCULATE keywords in the input file.

The status of an in-progress view factor calculation can be determined by using the sense switch. This is accomplished by first typing Control-C followed by:

sw1.	Stop run and save viewfl file for restart
sw2.	Viewfactor run statistics

A list of acceptable keywords are:

- *BOUNDARY_RADIATION_ENCLOSURE
- *BOUNDARY_RADIATION_SEGMENT
- *BOUNDARY_RADIATION_SEGMENT_VF_READ
- *BOUNDARY_RADIATION_SEGMENT_VF_CALCULATE
- *BOUNDARY_RADIATION_SET
- *BOUNDARY_RADIATION_SET_VF_READ
- *BOUNDARY_RADIATION_SET_VF_CALCULATE

Remarks:

In models that include radiation boundary conditions, a thermodynamic temperature scale is required, i.e., zero degrees must correspond to absolute zero. The Kelvin and Rankine temperature scales meet this requirement whereas Celsius and Fahrenheit temperature scales do not.

***BOUNDARY_RADIATION_ENCLOSURE**

Purpose: Define an enclosure by a list of segment sets to be used for a thermal radiation analysis. This keyword is only available in LS-DYNA MPP.

Card Summary:

Card 1. This card is required.

BRENCID	ENCNAME
---------	---------

Card 2. This card is required.

CALOPT	OUTOPT	CONOPT					
--------	--------	--------	--	--	--	--	--

Card 3. This card is required.

FILENAME

Card 4. This card is required.

SMFLAG	SMMAXI	SMABST	SMRELT				
--------	--------	--------	--------	--	--	--	--

Card 5. This card is required.

STYPE	SLMAXI	SLABST	SLRELT	SLMLEV			
-------	--------	--------	--------	--------	--	--	--

Card 6.1. Repeat pairs of Cards 6.1 and 6.2 until all the segment sets that form the enclosure are defined. This input ends at the next keyword ("*") card.

SSID							
------	--	--	--	--	--	--	--

Card 6.2. Repeat pairs of Cards 6.1 and 6.2 until all the segment sets that form the enclosure are defined. This input ends at the next keyword ("*") card.

NINT	BLOCK	SELCID	SEMULT	LOC			
------	-------	--------	--------	-----	--	--	--

*BOUNDARY

*BOUNDARY_RADIATION_ENCLOSURE

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	BRENCID	ENCNAME						
Type	I	A70						
Default	none	none						

VARIABLE

DESCRIPTION

BRENCID Boundary radiation ID for this enclosure

ENCNAME Name of enclosure, used for output purposes

Card 2	1	2	3	4	5	6	7	8
Variable	CALOPT	OUTOPT	CONOPT					
Type	I	I	I					
Default	0	0	0					

VARIABLE

DESCRIPTION

CALOPT Calculation option:
EQ.0: view factors

OUTOPT Output option:
EQ.0: no output
EQ.1: output in LSDA format

CONOPT Control option:
EQ.0: calculate view factors matrix and preform thermal analysis

Card 3	1	2	3	4	5	6	7	8
Variable	FILENAME							
Type	A80							
Default	<jobid>.viewfactor.<suffix>							

VARIABLE**DESCRIPTION**

FILENAME

Name of view factor output file

Card 4	1	2	3	4	5	6	7	8
Variable	SMFLAG	SMMAXI	SMABST	SMRELT				
Type	I	I	F	F				
Default	0	500	10^{-10}	10^{-6}				

VARIABLE**DESCRIPTION**

SMFLAG

View factor matrix smoothing flag:

EQ.0: no smoothing

EQ.1: smoothing

SMMAXI

Maximum number of iterations for view factor matrix smoothing
(default = 500)

SMABST

Absolute convergence tolerance for view factor matrix smoothing
(default = 10^{-10})

SMRELT

Relative convergence tolerance for view factor matrix smoothing
(default = 10^{-6})

BOUNDARY**BOUNDARY_RADIATION_ENCLOSURE**

Card 5	1	2	3	4	5	6	7	8
Variable	STYPE	SLMAXI	SLABST	SLRELT	SLMLEV			
Type	I	I	F	F	I			
Default	0	500	10^{-10}	10^{-6}	0			

VARIABLE**DESCRIPTION**

STYPE

Solver type:

EQ.0: reverse conjugated gradient

SLMAXI

Maximum number of iterations for radiosity solver (default = 500)

SLABST

Absolute convergence tolerance for radiosity solver (default is 10^{-10})

SLRELT

Relative convergence tolerance for radiosity solver (default = 10^{-6})

SLMLEV

Radiosity solver message level:

EQ.0: no output

EQ.1: debug output level I

EQ.2: debug output level II

EQ.3: debug output level III

Segment Set Card. Repeat pairs of Cards 6.1 and 6.2 until all segment sets which form the enclosure are listed. This input ends at the next keyword ("*") card.

Card 6.1	1	2	3	4	5	6	7	8
Variable	SSID							
Type	I							
Default	none							

Segment Set Characteristics Card. Repeat pairs of Cards 6.1 and 6.2 until all segment sets which form the enclosure are listed. This input ends at the next keyword ("**") card.

Card 6.2	1	2	3	4	5	6	7	8
Variable	NINT	BLOCK	SELCID	SEMULT	LOC			
Type	I	I	I	F	I			
Default	none	0	0	0.	0			

VARIABLE**DESCRIPTION**

SSID SSID specifies the ID for a set of segments that comprise a portion of, or possibly, the entire enclosure. See *SET_SEGMENT.

NINT Number of integration points for view factor calculation, $1 \leq \text{NINT} \leq 10$

EQ.0: LS-DYNA determines the number of integration points based on the segment size and separation distance.

BLOCK Flag indicating if this surface blocks the view between any other 2 surfaces:

EQ.0: no blocking (default)

EQ.1: blocking

SELCID Load curve ID for surface emissivity (see *DEFINE_CURVE):

GT.0: surface emissivity as a function of time

EQ.0: use constant multiplier value, SEMULT

LT.0: surface emissivity as a function of temperature. The value of -SELCID must be an integer, and it is interpreted as a load curve ID.

SEMULT Curve multiplier for surface emissivity; see *DEFINE_CURVE

LOC Application of surface for thermal shell elements (see THSHEL in the *CONTROL_SHELL input):

EQ.-1: lower surface of thermal shell element

EQ.0: middle surface of thermal shell element

VARIABLE	DESCRIPTION
	EQ.1: upper surface of thermal shell element

Remarks:

1. **Temperature Scale.** In models that include radiation boundary conditions, a thermodynamic temperature scale is required, meaning zero degrees must correspond to absolute zero. The Kelvin and Rankine temperature scales meet this requirement whereas Celsius and Fahrenheit temperature scales do not.

***BOUNDARY_RADIATION_SEGMENT**

Purpose: Apply a radiation boundary condition on a segment to transfer heat between the segment and the environment.

Include the following 2 cards for each segment. Setting TYPE = 1 on Card 1 below indicates that the segment transfers heat to the environment.

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

Card 2	1	2	3	4	5	6	7	8
Variable	FLCID	FMULT	TLCID	TMULT	LOC			
Type	I	F	I	F	I			
Default	none	0.	none	0.	0			

VARIABLE**DESCRIPTION**

N1, N2,
N3, N4

Node ID's defining segment

TYPE

Radiation type:

EQ.1: Radiation to environment

FLCID

Radiation heat transfer coefficient, $f = \sigma \varepsilon F$, specification where σ is the Stefan Boltzmann constant, ε is the surface emissivity, F is the surface view factor. See [Remark 1](#). This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DEFINE_FUNCTION and [Remark 2](#)). When the reference is to a curve, FLCID has the following interpretation:

GT.0: f is defined as a function of time, t , having points consisting of $(t, f(t))$ data pairs.

VARIABLE	DESCRIPTION
	EQ.0: f is a constant defined by the value FMULT. LT.0: f is defined as a function of temperature, T , by a curve consisting of $(T, f(T))$ data pairs. Enter FLCID on the *DEFINE_CURVE keyword.
FMULT	Radiation heat transfer coefficient, f , curve multiplier
TLCID	Environment temperature, T_∞ , specification. This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DEFINE_FUNCTION and Remark 3). When the reference is to a curve, TLCID has the following interpretation: GT.0: T_∞ is defined as a function of time, t , by a curve consisting of $(t, T_\infty(t))$ data pairs. EQ.0: T_∞ a constant defined by the value TMULT.
TMULT	Environment temperature, T_∞ , curve multiplier
LOC	For a thick thermal shell, the radiation will be applied to the surface identified by LOC. See the parameter THSHEL on the *CONTROL_SHELL keyword. EQ.-1: lower surface of thermal shell element EQ.0: middle surface of thermal shell element EQ.1: upper surface of thermal shell element

Remarks:

1. **Radiation Boundary Condition.** A radiation boundary condition is calculated using

$$\dot{q}'' = \sigma \epsilon F (T_{\text{surface}}^4 - T_\infty^4) = f (T_{\text{surface}}^4 - T_\infty^4) ,$$

where f is the radiation heat transfer coefficient. If f is a function of temperature, f is evaluated at the surface temperature, T_{surface} .

2. **FLCID Function Arguments.** If FLCID references a *DEFINE_FUNCTION, the radiation heat transfer coefficient can be a function of the segment centroid coordinates, the segment centroid velocity components, the segment centroid temperature, the environment temperature (T_∞), and the solution time, that is, “ $f(x, y, z, vx, vy, vz, temp, tinf, time)$.”

3. **TLCID Function Arguments.** If TLCID references a *DEFINE_FUNCTION, the environment temperature, T_{∞} , can be a function of the segment centroid coordinates, the segment centroid velocity components, and the solution time, that is “f(x, y, z, vx, vy, vz, time).”

*BOUNDARY

*BOUNDARY_RADIATION_SEGMENT_VF

*BOUNDARY_RADIATION_SEGMENT_VF_OPTION

Available options include:

READ

CALCULATE

Purpose: Apply a radiation boundary condition on a SEGMENT to transfer heat between the segment and an enclosure surrounding the segment using view factors.

The file viewfl must be present for the READ option, whereas it will be created with the CALCULATE option. If the file viewfl exists when using the CALCULATE option, LS-DYNA will terminate with an error message to prevent overwriting the file. The file viewfl contains the surface-to-surface area \times view factor products (that is, $A_i F_{ij}$). These products are stored by row and formatted as 5E16.0.

Include the following two cards for each segment. The enclosure is defined by additional segments using this keyword. Setting TYPE = 2 on Card 1 below specifies that the segment belongs to an enclosure.

Card 1	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4	TYPE	BLOCK	NINT	
Type	I	I	I	I	I	I	I	
Default	none	none	none	none	2	0	0	

Card 2	1	2	3	4	5	6	7	8
Variable	SELCID	SEMULT						
Type	I	F						
Default	SEMULT	0.						

VARIABLE

DESCRIPTION

N1, N2,
N3, N4

Node ID's defining segment

VARIABLE	DESCRIPTION
TYPE	Radiation type: EQ.2: Radiation within an enclosure
BLOCK	Flag indicating if this surface blocks the view between any other 2 surfaces. EQ.0: no blocking (default) EQ.1: blocking
NINT	Number of integration points for viewfactor calculation: EQ.0: LS-DYNA determines the number of integration points based on the segment size and separation distance GT.0: User specified number between (and including) 1 and 10.
SELCID	Load curve ID for surface emissivity (see *DEFINE_CURVE) GT.0: surface emissivity as a function of time EQ.0: use constant multiplier value, SEMULT LT.0: surface emissivity as a function of temperature. The value of -SELCID must be an integer, and it is interpreted as a load curve ID.
SEMULT	Curve multiplier for surface emissivity; see *DEFINE_CURVE.

*BOUNDARY

*BOUNDARY_RADIATION_SET

*BOUNDARY_RADIATION_SET

Purpose: Apply a radiation boundary condition on a SEGMENT_SET to transfer heat between the segment set and the environment.

Include the following 2 cards for each set. Setting TYPE = 1 on Card 1 below indicates that the segment transfers energy to the environment.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	TYPE					PSEROD	
Type	I	I					I	
Default	none	1					none	

Card 2	1	2	3	4	5	6	7	8
Variable	FLCID	FMULT	TLCID	TMULT	LOC			
Type	I	F	I	F	I			
Default	none	0.	none	0.	0			

VARIABLE

DESCRIPTION

SSID

SSID specifies the ID for a set of segments that comprise a portion of, or possibly, the entire enclosure. See *SET_SEGMENT.

TYPE

Radiation type:

EQ.1: radiation to environment

PSEROD

Part set ID for updating boundary segments exposed to the environment as solid elements erode; see [Remark 4](#).

FLCID

Radiation heat transfer coefficient, $f = \sigma \varepsilon F$, specification, where σ is the Stefan Boltzmann constant, ε is the surface emissivity, F is the surface view factor. See [Remark 1](#). This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DE-

VARIABLE	DESCRIPTION
	<p>FINE_FUNCTION and Remark 2). When the reference is to a curve, FLCID has the following interpretation:</p> <p>GT.0: f is defined as a function of time, t, by a curve consisting of $(t, f(t))$ data pairs.</p> <p>EQ.0: f is a constant defined by the value FMULT.</p> <p>LT.0: f is defined as a function of temperature, T, by a curve consisting of $(T, f(T))$ data pairs. Enter FLCID on the *DEFINE_CURVE keyword.</p>
FMULT	Curve multiplier of f (see FLCID)
TLCID	<p>Environment temperature, T_∞, specification. This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DEFINE_FUNCTION and Remark 3). When the reference is to a curve, TLCID has the following interpretation:</p> <p>GT.0: T_∞ is defined as a function of time, t, by a curve consisting of $(t, T_\infty(t))$ data pairs.</p> <p>EQ.0: T_∞ a constant defined by the value TMULT</p>
TMULT	Curve multiplier for T_∞
LOC	<p>For a thick thermal shell, the radiation will be applied to the surface identified by LOC. See the parameter THSHEL on the *CONTROL_SHELL keyword.</p> <p>EQ.-1: lower surface of thermal shell element</p> <p>EQ.0.: middle surface of thermal shell element</p> <p>EQ.1: upper surface of thermal shell element</p>

Remarks:

1. **Radiation Boundary Condition.** A radiation boundary condition is calculated using

$$\dot{q}'' = \sigma \epsilon F (T_{\text{surface}}^4 - T_\infty^4) = f (T_{\text{surface}}^4 - T_\infty^4) ,$$

where f is the radiation heat transfer coefficient. If f is a function of temperature, f is evaluated at the surface temperature, T_{surface} .

2. **FLCID Function Arguments.** If FLCID references a *DEFINE_FUNCTION, the radiation heat transfer coefficient can be a function of the segment centroid

coordinates, the segment centroid velocity components, the segment centroid temperature, the environment temperature (T_{∞}), and the solution time, that is, "f(x, y, z, vx, vy, vz, temp, tinf, time)."

3. **TLCID Function Arguments.** If TLCID references a *DEFINE_FUNCTION, the environment temperature can be a function of the segment centroid coordinates, the segment centroid velocity components, and the solution time, that is, "f(x, y, z, vx, vy, vz, time)."
4. **Erosion.** When radiation is applied to a segment set associated with solid elements that erode, the radiation condition on a deleted segment will not by default be transferred to newly exposed segments but ceases to exist. By specifying a part set through the parameter PSEROD, any such new segment, *that is attached to an element in this part set*, will inherit this boundary condition, using the same data as prescribed for all original segments.

***BOUNDARY_RADIATION_SET_VF_OPTION**

Available options include:

READ

CALCULATE

Include the following 2 cards for each set. This keyword applies a radiation boundary condition on a segment set to transfer heat using view factors between the segment set and an enclosure surrounding the segments. Segments contained in the segment set may form the enclosure. Setting TYPE = 2 on Card 1 below specifies that the segment set belongs to an enclosure.

The file “viewfl” must be present for the READ option. The file “viewfl” will be created for the CALCULATE option. If the file “viewfl” exists when using the CALCULATE option, LS-DYNA will terminate with an error message to prevent overwriting the file. The file “viewfl” contains the surface-to-surface area × view factor products (i.e. $A_i F_{ij}$). These products are stored by row and formatted as 5E16.0.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	TYPE	RAD_GRP	FILE_NO	BLOCK	NINT		
Type	I	I	I	I	I	I		
Default	none	2	0	0	0	0		

Card 2	1	2	3	4	5	6	7	8
Variable	SELCID	SEMULT						
Type	I	F						
Default	none	0.						

VARIABLE**DESCRIPTION**

SSID

SSID specifies the ID for a set of segments that comprise a portion of, or possibly, the entire enclosure. See *SET_SEGMENT.

VARIABLE	DESCRIPTION
TYPE	Radiation type: EQ.2: Radiation within an enclosure
RAD_GRP	Radiation enclosure group ID. The segment sets from all radiation enclosure definitions with the same group ID are augmented to form a single enclosure definition. If RAD_GRP is not specified or set to zero, then the segments are placed in group zero. All segments defined by the SEGMENT option are placed in set zero.
FILE_NO	File number for view factor file. FILE_NO is added to "viewfl_" to form the name of the file containing the view factors. For example, if FILE_NO is specified as 22, then the view factors are read from viewfl_22. For radiation enclosure group zero FILE_NO is ignored, and view factors are read from viewfl. The same file may be used for different radiation enclosure group definitions.
BLOCK	Flag indicating if this surface blocks the view between any other 2 surfaces. EQ.0: No blocking (default) EQ.1: Blocking
NINT	Number of integration points for view factor calculation EQ.0: LS-DYNA determines the number of integration points based on the segment size and separation distance GE.11: Not allowed
SELCID	Load curve ID for surface emissivity, see *DEFINE_CURVE GT.0: Function of time EQ.0: Use constant multiplier value, SEMULT LT.0: Function of temperature. SELCID is a load curve ID.
SEMULT	Curve multiplier for surface emissivity

Remarks:

Multiple enclosures can be modeled when using view factors. Consider the following example input. The order of segments in the view factor file follows the order the sets are assigned to the boundary radiation definition.

Enclosure radiation group 9 is composed of all the segments in segment set 15 followed by those in segment set 12. The view factors are stored in the file `viewfl_10`. Enclosure radiation group 8 is composed of the segments in segment set 13. The view factors are stored in the file `viewfl_21`.

***BOUNDARY_SALE_MESH_FACE**

Purpose: Define boundary conditions at S-ALE mesh faces. This keyword acts as a macro for other keywords to simplify specifying the boundary conditions. As a result, you can use a single keyword to apply boundary conditions.

During the keyword reader phase, this keyword translates to a combination of several different keywords. Those keywords include [*SET_NODE_GENERAL](#) (with SALEFAC option), [*SET_SEGMENT_GENERAL](#) (with SALEFAC option), [*BOUNDARY_SPC_SET](#), [*BOUNDARY_NON_REFLECTING](#), [*BOUNDARY_PRESCRIBED_MOTION](#), [*LOAD_SEGMENT_SET](#), and [*BOUNDARY_AMBIENT](#).

Include as many cards as needed with each card representing one boundary condition. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	BCTYPE	MSHID	NEGX	POSX	NEGY	POSY	NEGZ	POSZ
Type	A	I	I	I	I	I	I	I
Default	none	none	0	0	0	0	0	0

VARIABLE**DESCRIPTION**

BCTYPE

Available boundary conditions:

EQ.FIXED: All nodes at the face are fixed in all directions. (Applied with [*BOUNDARY_SPC_SET](#).)

EQ.NOFLOW: No flow allowed through the face. (Applied with [*BOUNDARY_SPC_SET](#).)

EQ.SYM: The face is a symmetric plane (same as NOFLOW). (Applied with [*BOUNDARY_SPC_SET](#).)

EQ.NONREFL: Non-reflective boundary condition. (Applied with [*BOUNDARY_NON_REFLECTING](#).)

EQ.FLOWVEL: Nodes constrained by time-dependent velocities. (Applied with [*BOUNDARY_PRESCRIBED_MOTION](#).)

EQ.PRES: Faces loaded by time-dependent pressures. (Applied with [*LOAD_SEGMENT_SET](#).)

EQ.AMBIENT: Ambient elements attached to the faces. See

VARIABLE	DESCRIPTION
	Remarks 1 and 2. (Applied with *BOUNDARY_AMBIENT.)
MSHID	S-ALE Mesh ID
NEG[X,Y,Z], POS[X,Y,Z]	Determine where the boundary condition is applied to the mesh. NEGX, POSX, NEGY, POSY, NEGZ, or POSZ means the mesh faces with an outward normal vector in the local $-x$, $+x$, $-y$, $+y$, $-z$, or $+z$ -directions, respectively. EQ.0: The boundary condition is <i>not</i> applied to faces with this outward normal. EQ.1: The boundary condition is applied to faces with this outward normal.

Remarks:

1. **AMBIENT.** If AMBIENT is applied to mesh faces, the elements attached to these faces are turned into “ambient” elements controlled by *BOUNDARY_AMBIENT with AMBTYP = 4. If these elements have their pressures initialized by *INITIAL_ALE_HYDROSTATIC, *ALE_AMBIENT_HYDROSTATIC controls the ambient elements.
2. **Boundary conditions combined with AMBIENT.** If AMBIENT and NONREFL or AMBIENT and FLOWVEL are applied to the same mesh faces, the elements attached to these faces are turned into ambient elements. The faces connecting the ambient elements to the mesh are selected to apply NONREFL. The nodes connecting the ambient elements to the mesh are selected to apply FLOWVEL.

Example:

The following input specifies no-flow boundary conditions on faces with normal vectors in the $-y$ and $-z$ -directions and non-reflecting boundary conditions on the faces with normal vectors in the $-x$, $+x$, $+y$, and $+z$ -directions.

```
*BOUNDARY_SALE_MESH_FACE
$  option      mshid      NEGX      POSX      NEGY      POSY      NEGZ      POSZ
   NOFLOW       1              1              1              1
   NONREFL      1          1          1              1              1
```

Internally, the above input is translated to the following keywords:

```
*BOUNDARY_SPC_SET
```

*BOUNDARY

*BOUNDARY_SALE_MESH_FACE

```

      1
*BOUNDARY_SPC_SET
      2
*SET_NODE_GENERAL
$      SID
      1
$  OPTION      MSHID      XMN      XMX      YMN      YMX      ZMN      ZMX
  SALEFAC      1
*SET_NODE_GENERAL
$      SID
      2
$  OPTION      MSHID      XMN      XMX      YMN      YMX      ZMN      ZMX
  SALEFAC      1
*BOUNDARY_NON_REFLECTING
$      SID
      11
*SET_SEGMENT_GENERAL
$      SID
      11
$  OPTION      MSHID      XMN      XMX      YMN      YMX      ZMN      ZMX
  SALEFAC      1      1      1      1      1      1      1
```

In comparison to the translated version, *BOUNDARY_SALE_MESH_FACE provides a more streamlined, intuitive approach to facilitate the boundary condition setup for S-ALE models. This method is particularly useful when the mesh is tilted since the local coordinate system is already assumed.

***BOUNDARY_SLIDING_PLANE**

Purpose: Define a sliding symmetry plane. This option applies to continuum domains modeled with solid elements.

Card 1	1	2	3	4	5	6	7	8
Variable	NSID	VX	VY	VZ	COPT			
Type	I	F	F	F	I			
Default	none	0	0	0	0			

VARIABLE**DESCRIPTION**

NSID	Nodal set ID, see *SET_NODE
VX	<i>x</i> -component of vector defining normal or vector
VY	<i>y</i> -component of vector defining normal or vector
VZ	<i>z</i> -component of vector defining normal or vector
COPT	Option: EQ.0: node moves on normal plane. EQ.1: node moves only in vector direction.

Remarks:

Any node may be constrained to move on an arbitrarily oriented plane or line depending on the choice of COPT. Each boundary condition card defines a vector originating at (0,0,0) and terminating at the coordinates defined above. Since an arbitrary magnitude is assumed for this vector, the specified coordinates are non-unique and define only a direction. Use of *BOUNDARY_SPC is preferred over *BOUNDARY_SLIDING_PLANE as the boundary conditions imposed using the latter have been seen to break down somewhat in lengthy simulations owing to numerical roundoff.

***BOUNDARY_SPC_OPTION1_{OPTION2}_{OPTION3}**

OPTION1 is required since it specifies whether the SPC applies to a single node or to a set. The two choices are:

NODE

SET

OPTION2 allows optional birth and death times to be assigned the single node or node set:

BIRTH_DEATH

This option requires one additional line of input. The BIRTH_DEATH option is inactive during the dynamic relaxation phase, which allows the SPC to be removed during the subsequent normal analysis phase. The BIRTH_DEATH option can be used only once for any given node and if used, no other *BOUNDARY_SPC commands can be used for that node.

OPTION3 allows an optional ID to be given that applies either to the single node or to the entire set:

ID

If a heading is defined with the ID, then the ID with the heading will be written at the beginning of the ASCII file, *spcforc*. A node which appears in more than one SPC set containing an ID will only be associated with one of those IDs in *spcforc*.

Purpose: Define nodal single point constraints. Do not use this option in r-adaptive problems since the nodal point ID's change during the adaptive step. If possible use CONSTRAINED_GLOBAL instead.

ID Card. Additional card for the ID keyword option.

Optional	1	2	3	4	5	6	7	8
Variable	ID	HEADING						
Type	I	A70						

Card 1	1	2	3	4	5	6	7	8
Variable	NID/NSID	CID	DOFX	DOFY	DOFZ	DOFRX	DOFRY	DOFRZ
Type	I	I	I	I	I	I	I	I
Default	none	0	0	0	0	0	0	0

Birth/Death Card. Additional card for the BIRTH_DEATH keyword option.

Card 2	1	2	3	4	5	6	7	8
Variable	BIRTH	DEATH						
Type	F	F						
Default	0.0	10 ²⁰						

VARIABLE**DESCRIPTION**

ID	Optional SPC set ID to which this node or node set belongs. This ID does not need to be unique
HEADING	An optional SPC descriptor that will be written into the d3hsp file and the spcforc file.
NID/NSID	Node ID or nodal set ID, see *SET_NODE.
CID	Coordinate system ID, see *DEFINE_COORDINATE_SYSTEM.
DOFX	Insert 1 for translational constraint in local x -direction.
DOFY	Insert 1 for translational constraint in local y -direction.
DOFZ	Insert 1 for translational constraint in local z -direction.
DOFRX	Insert 1 for rotational constraint about local x -axis.
DOFRY	Insert 1 for rotational constraint about local y -axis.
DOFRZ	Insert 1 for rotational constraint about local z -axis.

VARIABLE	DESCRIPTION
BIRTH	Activation time for SPC constraint. The birth time is ignored during dynamic relaxation.
DEATH	Deactivation time for the SPC constraint. The death time is ignored during dynamic relaxation.

Remarks:

Constraints are applied if for each DOF_{*ij*} field set to 1. A value of zero means no constraint. No attempt should be made to apply SPCs to nodes belonging to rigid bodies (see *MAT_RIGID for application of rigid body constraints).

```

$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$  *BOUNDARY_SPC_NODE
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$  Make boundary constraints for nodes 6 and 542.
$
*BOUNDARY_SPC_NODE
$
$.>...1...>...2...>...3...>...4...>...5...>...6...>...7...>...8
$      nid      cid      dofx      dofy      dofz      dofrx      dofry      dofrz
$      6         0         1         1         1         1         1         1
$      542        0         0         1         0         1         0         1
$
$      Node 6 is fixed in all six degrees of freedom (no motion allowed).
$
$      Node 542 has a symmetry condition constraint in the x-z plane,
$      no motion allowed for y translation, and x & z rotation.
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

```

***BOUNDARY_SPC_SYMMETRY_PLANE_{OPTION}**

Purpose: Constrain nodes that are within some distance (a tolerance) of a plane to have no motion orthogonal to that plane. This keyword is usually used for a geometric symmetry plane, so that the full geometry does not have to be modeled. This keyword supports both *h*- and *r*-adaptivity. Related keywords are: *BOUNDARY_SPC_SET, *CONSTRAINED_GLOBAL, and *CONSTRAINED_LOCAL. Note this keyword should not be used on any node that also has *BOUNDARY_SPC applied. This keyword is supported for shells, thick shells, solids, and beams.

Available options include:

<BLANK>

SET

Card Sets. For each symmetry plane input one pair of Cards 1 and 2. This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	IDSP	PID/PSID	X	Y	Z	VX	VY	VZ
Type	I	I	F	F	F	F	F	F
Default	none	none	0.0	0.0	0.0	0.0	0.0	0.0

Card 2	1	2	3	4	5	6	7	8
Variable	TOL							
Type	F							
Default	0.0/0.2							

VARIABLE**DESCRIPTION**

IDSP

Identification number of the constraint. Must be unique.

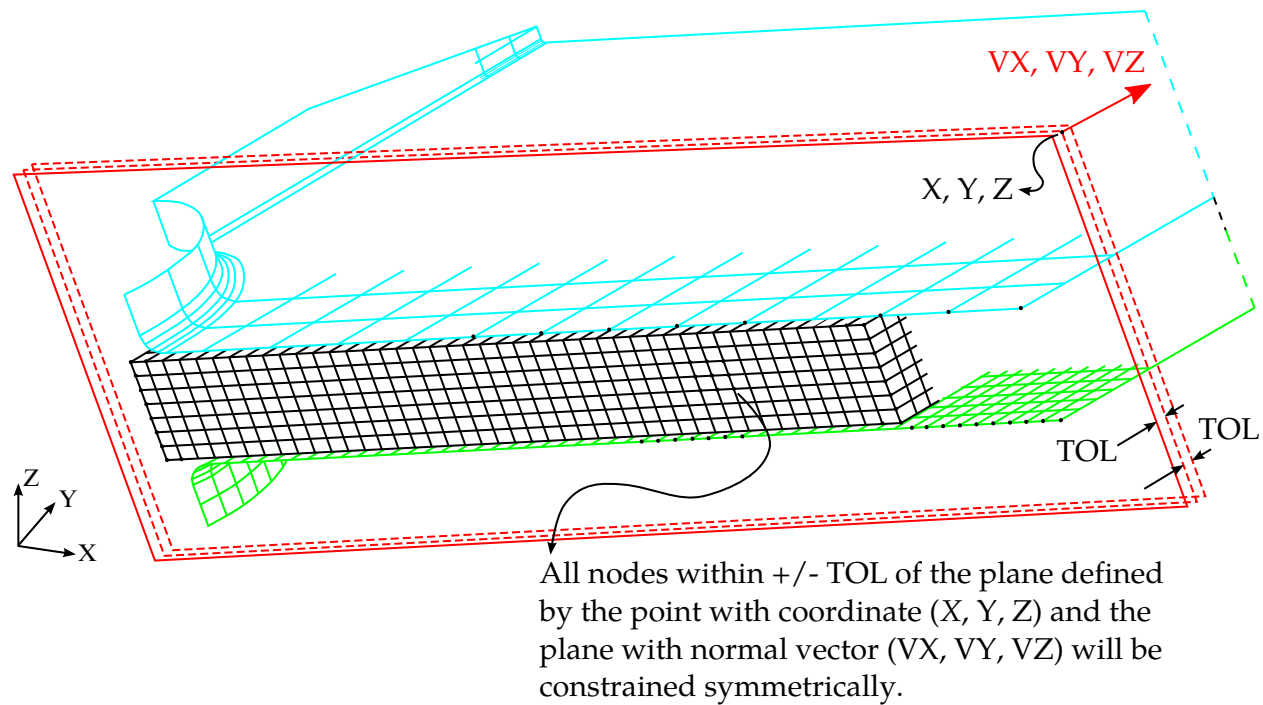


Figure 5-6. The nodes within a tolerance of the plane shown above are constrained by this keyword.

VARIABLE	DESCRIPTION
PID/PSID	When the option is blank, PID is a part ID specifying the part on which the constraint will be imposed. When the SET option is active, this field must contain a part set ID specifying the parts on which the constraint will be imposed.
X, Y, Z	Position coordinates on the symmetry plane.
VX, VY, VZ	Vector components of the symmetry plane's normal.
TOL	A distance tolerance value within which the nodes on the deformable part will be constrained. For shell elements, the default tolerance is 0.2.

Remarks:

- Keyword restrictions.** An input deck can contain multiple symmetry planes. However, multiple instantiations of this keyword cannot appear in the input deck with identical symmetry planes, meaning planes with the same position coordinates and normal vectors. Thus, to have two parts with the same symmetry plane, use the SET keyword option and include them in the same part set specified with PSID.

2. **Plane definition.** As shown in [Figure 5-6](#), a symmetry plane's position is defined using a point's coordinates X, Y, and Z, and the plane's orientation is defined by the vector component VX, VY, and VZ. All nodes within a specified tolerance TOL have constrained motion in the direction orthogonal to the plane.
3. **Tailor-welded blanks.** The option SET allows for symmetry boundary conditions to be applied on tailor-welded blanks (TWB), where two pieces of sheet metal with different thickness or properties are welded in the butt joint configuration.

Example:

The following input creates symmetric constraints on nodes (from PID 11) within distance of 0.1 mm from the defined symmetry plane having a normal vector (1.0,1.0,1.0) passing through point (10.5,40.0,20.0).

```
*BOUNDARY_SPC_SYMMETRY_PLANE
$.>...1...>...2...>...3...>...4...>...5...>...6...>...7...>...8
$  IDSP      PID      X      Y      Z      VX      VY      VZ
    1        11      10.5    40.0    20.0     1.0     1.0     1.0
$.>...1...>...2...>...3...>...4...>...5...>...6...>...7...>...8
$   TOL
    0.10

.
```

***BOUNDARY_SPH_FLOW**

Purpose: Define a flow of particles. This option applies to continuum domains modeled with SPH elements.

Card 1	1	2	3	4	5	6	7	8
Variable	ID	STYP	DOF	VAD	LCID	SF	DEATH	BIRTH
Type	I	I	I	I	I	F	F	F
Default	none	none	none	0	none	1.0	10 ²⁰	0.0

Card 2	1	2	3	4	5	6	7	8
Variable	NODE	VID						
Type	I	I						
Default	none	0						

VARIABLE**DESCRIPTION**

NSID, PID

Nodal set ID (NSID) or part ID (PID).

STYP

Set type:

EQ.1: part set ID, see *SET_PART,

EQ.2: part ID, see *PART,

EQ.3: node set ID, see *SET_NODE.

DOF

Applicable degrees-of-freedom:

EQ.1: x -translational degree-of-freedom,EQ.2: y -translational degree-of-freedom,EQ.3: z -translational degree-of-freedom,

EQ.4: translational motion in direction given by VID. Movement on plane normal to the vector is permitted.

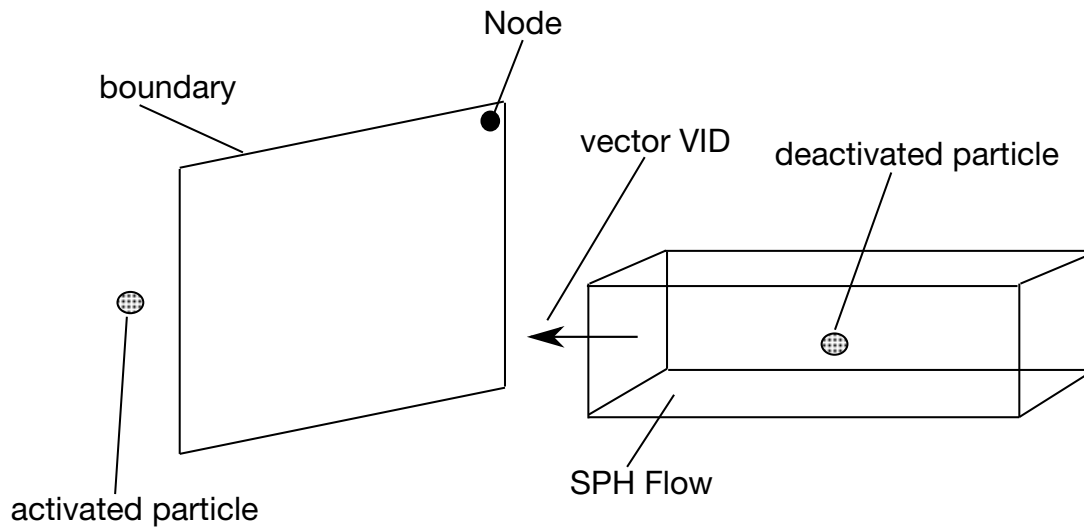


Figure 5-7. Vector VID determines the orientation of the SPH flow

VARIABLE	DESCRIPTION
VAD	Velocity / Acceleration / Displacement flag applied to SPH elements before activation: EQ.0: velocity, EQ.1: acceleration, EQ.2: displacement.
LCID	Load curve ID to describe motion value as a function of time; see *DEFINE_CURVE.
SF	Load curve scale factor.
DEATH	Time imposed motion / constraint is removed: EQ.0.0: default set to 10^{20} .
BIRTH	Time imposed motion / constraint is activated.
NODE	Node fixed in space which determines the boundary between activated particles and deactivated particles.
VID	Vector ID for defining the orientation of the SPH flow; see *DEFINE_VECTOR.

Remarks:

Initially, the user defines the set of particles that are representing the flow of particles during the simulation. At time $t = 0$, all the particles are deactivated which means that

no particle approximation is calculated. The boundary of activation is a plane determined by the NODE and normal to the vector VID. The particles are activated when they reached the boundary. When they are activated, particle approximation begins.

***BOUNDARY_SPH_NON_REFLECTING**

Purpose: Define a non-reflecting boundary plane for SPH. This option applies to continuum domains modeled with SPH elements.

Card 1	1	2	3	4	5	6	7	8
Variable	VTX	VTY	VTZ	VHX	VHY	VHZ		
Type	F	F	F	F	F	F		
Default	0.	0.	0.	0.	0.	0.		

VARIABLE**DESCRIPTION**

VTX	x -coordinate of tail of a normal vector originating on the wall (tail) and terminating in the body (head); that is, the vector points from the non-reflecting boundary plane to the body.
VTY	y -coordinate of tail
VTZ	z -coordinate of tail
VHX	x -coordinate of head
VHY	y -coordinate of head
VHZ	z -coordinate of head

Remarks:

- 1. Non-reflecting Boundary Plane Orientation.** The non-reflecting boundary plane has to be normal to either the x , y or z direction.
- 2. Non-reflecting Boundary Plane Location.** The non-reflecting boundary plane has to be placed $h/2$ away from the SPH part, where h is the interparticle distance.

***BOUNDARY_SPH_NOSLIP**

Purpose: Impose a no-slip boundary condition for continuum domains modeled with SPH elements. This is accomplished by doing the following to dummy particles: (1) adding a user-specified fictitious velocity to the (possibly non-zero) existing velocity, (2) ensuring the density stays constant throughout the simulation, (3) interpolating pressure using only the neighboring particles that are not part of the no-slip boundary condition, (4) zeroing out forces due to other SPH phenomena, and (5) subtracting the fictitious velocity at the end of the time step.

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

VARIABLE**DESCRIPTION**

ID	Node set, node, part set, or part ID used to identify dummy particles.
IDTYP	ID type: EQ.0: Node set ID (see *SET_NODE) EQ.1: Node ID (see *NODE) EQ.2: Part set ID (see *SET_PART) EQ.3: Part ID (see *PART)
CID	Local coordinate system ID used to define the components of the velocity vector; see *DEFINE_COORDINATE_SYSTEM for example.
VID	Vector ID for defining the fictitious velocity applied to no-slip particles; see *DEFINE_VECTOR .
LCID	Load curve ID to describe the scaling factor for the velocity vector as a function of time; see *DEFINE_CURVE .

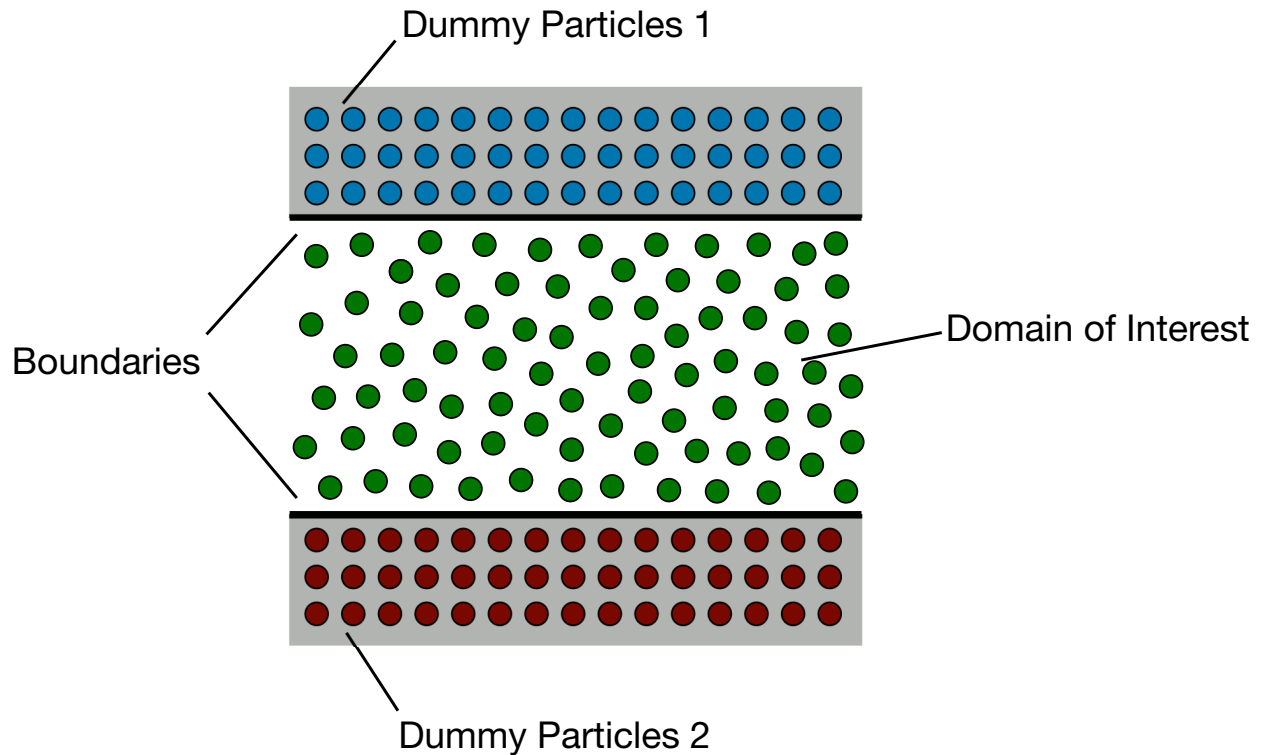


Figure 5-8. Neighboring particles (dummy particles 1 and 2) beyond the boundary must be defined in the input file

Remarks:

1. **SPH elements.** The SPH elements used to impose the no-slip boundary conditions are dummy particles that extend beyond the domain of interest and must be defined in the input file. We recommend using three to four layers of dummy particles as shown in [Figure 5-8](#).
2. **Velocity.** The velocity specified here is added to any other imposed velocity when calculating internal forces that involve the dummy particles but omitted when computing displacement of said dummy particles. Hence, this keyword can be used in combination with others that assign a velocity to SPH particles.

***BOUNDARY_SPH_PERIODIC**

Purpose: Impose a periodic boundary condition for domains modeled with SPH elements. To apply this condition, two inward-pointing vectors must be provided with ***DEFINE_VECTOR** to identify the periodic boundary planes. The vector tails are treated as local origins on their corresponding planes. Particles that cross one of the two planes are shifted to the equivalent position relative to the other plane (and within the SPH domain). Additionally, ghost particles are created for each interior particle that is near a periodic plane. The ghost particles are placed outside of the domain and at a position equivalent to the parent particle, but relative to the other plane. If an interior particle next to the edge moves sufficiently into the domain, its equivalent ghost particle is removed. Symmetric boundary planes do *not* need to be parallel but note that the relative rotation is calculated with the Rodrigues rotation formula. Thus, no in-plane rotations are accounted for.

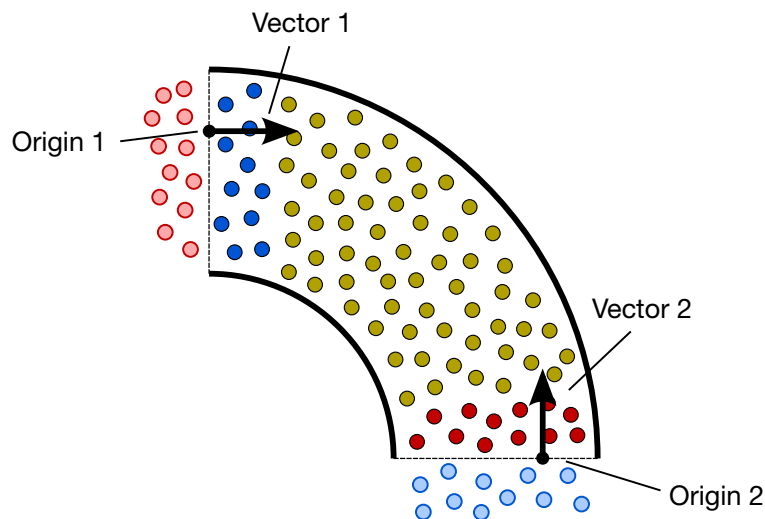


Figure 5-9. SPH particles inside of an annular domain with periodic boundaries identified by their corresponding vectors and origins. A ghost particle is generated for each particle near one of the boundary planes at the equivalent location relative to the opposite plane. In this figure, the exterior, light blue particles next to the second surface are the ghost particles for the interior, dark blue particles next to the first surface. Similarly, the light red particles are the ghost particle equivalents of the dark red particles.

Card 1	1	2	3	4	5	6	7	8
Variable	VID1	VID2						
Type	I	I						
Default	none	none						

VARIABLE**DESCRIPTION**

VID1, VID2

Vector IDs defining the inward-pointing normal vectors of the periodic boundary planes. See [*DEFINE_VECTOR](#). The vector's tail is taken as a reference point for the given plane.

Remarks:

1. **Incompressible SPH.** Periodic boundary conditions are not supported for ISPH (FORM = 13 in [*CONTROL_SPH](#)).

***BOUNDARY_SPH_SYMMETRY_PLANE**

Purpose: Define a symmetry plane for SPH. This option applies to continuum domains modeled with SPH elements.

Card 1	1	2	3	4	5	6	7	8
Variable	VTX	VTY	VTZ	VHX	VHY	VHZ		
Type	F	F	F	F	F	F		
Default	0.	0.	0.	0.	0.	0.		

VARIABLE**DESCRIPTION**

VTX	x -coordinate of tail of a normal vector originating on the wall (tail) and terminating in the body (head) (that is, vector points from the symmetry plane into the body).
VTY	y -coordinate of tail
VTZ	z -coordinate of tail
VHX	x -coordinate of head
VHY	y -coordinate of head
VHZ	z -coordinate of head

Remarks:

1. **SPH Elements.** A plane of symmetry is assumed for all SPH elements defined in the model.
2. **Plane Orientation.** The plane of symmetry has to be normal to either the x , y or z -direction.
3. **Plane Location.** The plane of symmetry is typically placed $h/2$ away from the SPH part, where h is the interparticle distance.
4. **Axisymmetric SPH.** For axisymmetric SPH analysis, IDIM = -2 (see *CONTROL_SPH), a plane of symmetry centered at the global origin and normal to x -direction is automatically created by LS-DYNA.

***BOUNDARY_SYMMETRY_FAILURE**

Purpose: Define a symmetry plane with a failure criterion. This option applies to continuum domains modeled with solid elements.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	FS	VTX	VTY	VTZ	VHX	VHY	VHZ
Type	I	F	F	F	F	F	F	F
Default	none	0.	0.	0.	0.	0.	0.	0.

VARIABLE**DESCRIPTION**

SSID	Segment set ID, see *SET_SEGMENT
FS	Tensile failure stress > 0.0. The average stress in the elements surrounding the boundary nodes in a direction perpendicular to the boundary is used.
VTX	x -coordinate of tail of a normal vector originating on the wall (tail) and terminating in the body (head), that is, vector points from the symmetry plane into the body.
VTY	y -coordinate of tail
VTZ	z -coordinate of tail
VHX	x -coordinate of head
VHY	y -coordinate of head
VHZ	z -coordinate of head

Remarks:

A plane of symmetry is assumed for the nodes on the boundary at the tail of the vector given above. Only the motion perpendicular to the symmetry plane is constrained. After failure the nodes are set free.

***BOUNDARY_TEMPERATURE_OPTION**

Available options include:

NODE

SET

Purpose: Define temperature boundary conditions for a thermal or coupled thermal/structural analysis. In a structural-only analysis, use *LOAD_THERMAL_OPTION to define temperatures (see also SOLN in *CONTROL_SOLUTION).

Card 1	1	2	3	4	5	6	7	8
Variable	NID	TLCID	TMULT	LOC	TDEATH	TBIRTH		
Type	I	I	F	I	F	F		
Default	none	0	0.	0	10 ²⁰	0.		

VARIABLE**DESCRIPTION**

NID

Node or node set ID

TLCID

Temperature, T , specification. This parameter can reference a load curve ID (see *DEFINE_CURVE) or a function ID (see *DEFINE_FUNCTION and [Remark 2](#)). When the reference is to a curve, TLCID has the following interpretation:

GT.0: T is defined by a curve consisting of (t, T) data pairs.

EQ.0: T is a constant defined by the value TMULT.

TMULT

Temperature, T , curve multiplier.

LOC

Application of surface for thermal shell elements, see parameter, THSHEL, in the *CONTROL_SHELL input:

EQ.-1: Lower surface of thermal shell element

EQ.0: Middle surface of thermal shell element

EQ.1: Upper surface of thermal shell element

TDEATH

Deactivation time for temperature boundary condition. At this point in time the temperature constraint is removed.

VARIABLE	DESCRIPTION
TBIRTH	Activation time for temperature boundary condition. Before this point in time the temperature constraint is ignored.

Remarks:

1. **SPH and ISPH particles.** This keyword can be used to apply temperature boundary conditions to SPH and ISPH particles.
2. **Temperature function.** If TLCID references a *DEFINE_FUNCTION, temperature may be a function of the nodal point coordinates, the nodal point velocity components, and the solution time, that is, "f(x,y,z,vx,vy,vz,time)."

*BOUNDARY

*BOUNDARY_TEMPERATURE_PERIODIC_SET

*BOUNDARY_TEMPERATURE_PERIODIC_SET

Purpose: Specify different kinds of periodic boundary conditions for temperature.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID1	PTYPE	SSID2	TDLCID	AXE	NID	ANGLE	
Type	I	I	I	I	I	I	F	
Default	none	none	none	0	none	none	none	

VARIABLE

DESCRIPTION

SSID1 First segment set on which the periodic temperature boundary condition will be applied. See [Remark 1](#).

PTYPE Type of periodic boundary condition:
EQ.1: Rotation boundary condition defined by an axis, an origin point and a rotation angle
EQ.2: Reflective boundary condition defined by an axis and origin point
EQ.3: Sliding boundary condition

SSID2 Second segment set on which the periodic temperature boundary condition will be applied.

TDCLID Optional load curve specifying the temperature drop, T_{drop} , between the two surfaces in the periodic boundary condition as a function of time. Note that $T_{\text{drop}} = T_1 - T_2$ where T_1 is the temperature of the surface specified with SSID1 and T_2 is the temperature of the surface specified with SSID2.

EQ.0: No temperature drop between that surfaces, that is, $T_{\text{drop}} = 0.0$

AXE Axis for PTYPE = 1 or 2:
EQ.1: X-axis
EQ.2: Y-axis
EQ.3: Z-axis

VARIABLE	DESCRIPTION
	Flag for meaning of ANGLE for PTYPE = 3. Setting AXE = 1 means that ANGLE is the contact distance. Otherwise, it is a scale factor on the contact distance search.
NID	Node ID giving the origin point coordinates
ANGLE	Rotation angle if PTYPE = 1. Scaling factor on contact distance search if PTYPE = 3 (default applies a scale factor of 0.3 on local element size). If AXE = 1 and PTYPE = 3, then ANGLE becomes the contact distance.

Remarks:

1. **Constraint-based method.** LS-DYNA applies these boundary conditions with a constraint-based method. For each node in SSID1 LS-DYNA finds a host face in SSID2 and applies correct weights to define a new constraint to add to the system. For PTYPE = 1 and 2, the meshes on both sides do not have to be identical, but we recommend matching meshes. PTYPE = 3 can be used to define a “perfect” thermal contact between two solids.

*BOUNDARY

*BOUNDARY_TEMPERATURE_RSW

*BOUNDARY_TEMPERATURE_RSW

Purpose: Define temperature boundary conditions within an ellipsoidal region of the solid or shell structure. Temperatures are prescribed to nodes found in a region defined by this keyword. The boundary condition is tailored to represent the so-called weld nuggets evolving during resistive spot welding (RSW) processes. It is applicable to a thermal or coupled thermal/structural analysis.

Card 1	1	2	3	4	5	6	7	8
Variable	SID	OPTION	NID1	NID2	TDEATH	TBIRTH	LOC	
Type	I	I	I	I	F	F	I	
Default	none	0	none	none	10 ²⁰	0.	0	

Geometry and Temperature Card.

Card 2	1	2	3	4	5	6	7	8
Variable	DIST	H1	H2	R	TEMPC	TEMPB	LCIDT	
Type	F	F	F	F	F	F	I	
Default	0.	0.	0.	0.	0.	0.	none	

Heat Affected Zone Card. Additional card for OPTION = 1.

Card 3	1	2	3	4	5	6	7	8
Variable	HZ1	HZ2	RZ	TEMPZB				
Type	R	F	F	R				
Default	0.	0.	0.	0.				

VARIABLE

DESCRIPTION

SID

Node Set ID; see *SET_NODE_OPTION. Nodes in the set will be checked to see if they are in the nugget or heat affected zone. If

VARIABLE	DESCRIPTION
	they are, the boundary condition will be applied. The boundary condition will not be applied to nodes in these regions if they are not included in the set.
NID1	Node defining the tail of the orientation vector (axis of rotation of the ellipsoidal region) and the base for positioning of the nugget. See Remarks 1 and 2 .
NID2	Node defining the head of the orientation vector (axis of rotation of the ellipsoidal region). See Remarks 1 and 2 .
OPTION	Option for heat affected zone around the weld nugget: EQ.0: No heat affected zone EQ.1: Ellipsoidal region considered
TDEATH	Deactivation time for temperature boundary condition. At this point in time the temperature constraint is removed.
TBIRTH	Activation time for temperature boundary condition. Before this point in time the temperature constraint is ignored.
LOC	Application of surface for thermal shell elements; see parameter, THSHEL, in the *CONTROL_SHELL input: EQ.-1: Lower surface of thermal shell element EQ.0: Middle surface of thermal shell element EQ.1: Upper surface of thermal shell element
DIST	Position of center of nugget on the axis of rotation. Parameter defines the distance to NID1 along the orientation vector. See Remark 1 .
H1	Half width h_1 of nugget in the lower half, meaning in direction to NID1. See Remark 2 .
H2	Half width h_2 of nugget in the upper half, meaning in direction to NID2. See Remark 2 .
R	Radius r_{weld} of the nugget in surface normal to orientation vector. See Remark 2 .
TEMPC	Base temperature at the center of the nugget. See Remark 3 .
TEMPB	Base temperature at the boundary of the nugget. See Remark 3 .

VARIABLE	DESCRIPTION
LCIDT	<p> LCIDT refers to the load curve ID prescribing the temperature evolution in the nugget as a function of time. The abscissa of the load curve will be normalized between the birth and death times of the boundary condition.</p> <p>GT.0: The ordinate values of the load curve scale the respective base temperature of a particular point.</p> <p>EQ.0: No temperature evolution. Base temperatures are used.</p> <p>LT.0: The ordinate values of the load curve are used to define a linear combination between the temperature at the birth time and the base temperature of a particular point. Load curve ordinate values should range between 0.0 and 1.0. We recommend $LCIDT < 0$ to ensure a smooth temperature evolution.</p> <p>See Remark 3.</p>
HZ1	Half width h_{z1} of heat affected zone in the lower half, meaning in direction to NID1. Only active for OPTION = 1. See Remark 4 .
HZ2	Half width h_{z2} of heat affected zone in the upper half, meaning in direction to NID1. Only active for OPTION = 1. See Remark 4 .
RZ	Radius r_{haz} of the heat affected zone in surface normal to orientation vector. See Remark 4 .
TEMPBZ	Base temperature at the boundary of the heat affected zone for OPTION = 1. See Remark 4 .

Remarks:

1. **Positioning.** The position of the center of the nugget is defined starting from a base node (NID1). It is then translated by a distance DIST along the orientation vector \mathbf{v} of the nugget. Vector \mathbf{v} points from NID1 to NID2. See [Figure 5-10](#).
2. **Geometry of the nugget.** Orientation vector \mathbf{v} defines the axis of rotational symmetry for the ellipsoidal region, which reflects the weld nugget. The radius around this axis is defined by r_{weld} . The nugget consists of two half ellipsoids of possibly different height. The lower half, i.e. the half between NID1 and the center, has a height of h_1 , whereas the upper half has a height of h_2 . See [Figure 5-10](#).
3. **Prescribing temperature values.** You can prescribe the base temperatures at the boundary and the center of the nugget with TEMPC and TEMPB,

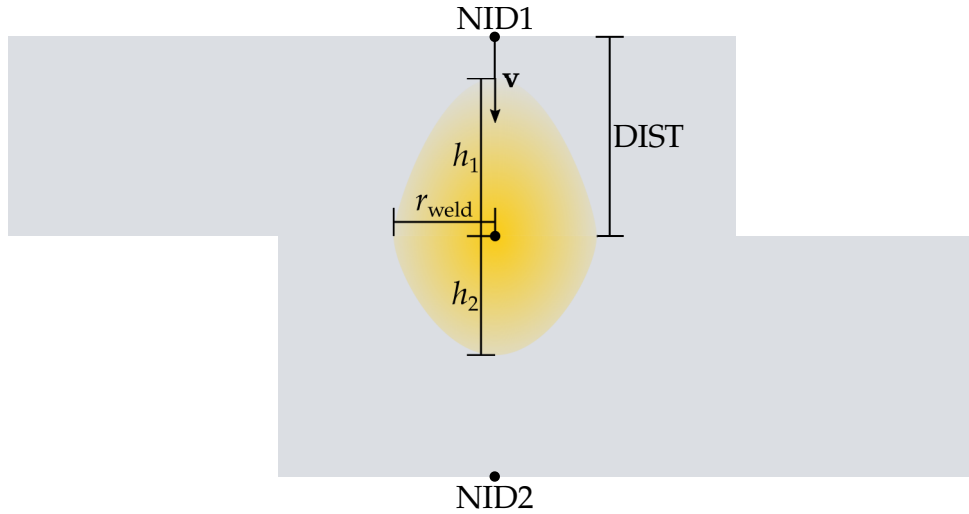


Figure 5-10. Example of geometry for OPTION = 0

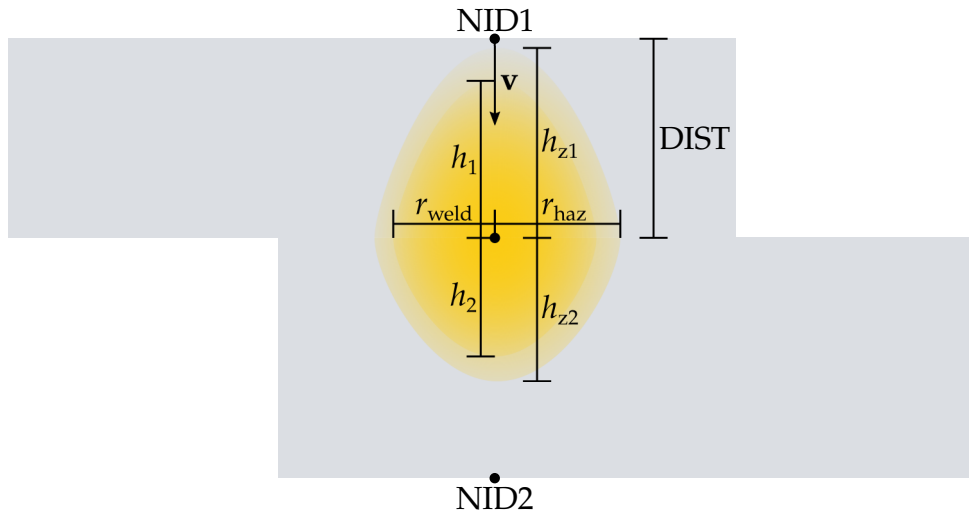


Figure 5-11. Example of geometry with OPTION = 1 (heat affected zone)

respectively. In between LS-DYNA calculates the base temperature in the nugget, $T_{\text{nug},b}(x, y, z)$, with a quadratic interpolation.

For this boundary condition you must provide activation and deactivation times, t_{ac} and t_{deac} , respectively. With LCIDT, you can specify the temperatures as a function, $\theta(\tilde{t})$, of the normalized time, $\tilde{t} = (t - t_{\text{ac}}) / (t_{\text{deac}} - t_{\text{ac}})$, within this time span. For positive values of field LCIDT, a simple scaling is applied:

$$T_{\text{nug}}(x, y, z, t) = T_{\text{nug},b}(x, y, z) \times \theta(\tilde{t}(t)) .$$

For negative values of LCIDT, a smooth evolution of the temperature from the birth time is ensured, so we recommend this type of load curve. The temperature is given by:

$$T_{\text{nug}}(x, y, z, t) = T_{\text{ac}}(x, y, z) + \left(T_{\text{nug},b}(x, y, z) - T_{\text{ac}}(x, y, z) \right) \times \theta(\tilde{t}) .$$

Here T_{ac} corresponds to the temperature of a particular point right before the birth time.

4. **Heat affected zone.** With input field OPTION set to 1, the boundary condition expands to a heat affected zone that has the same general shape definition as the weld nugget described in [Remark 2](#). The dimensions are here given by r_{haz} , h_{z1} , and h_{z2} . See [Figure 5-11](#).

You can prescribe the base temperature value at the boundary of the heat affected zone. LS-DYNA finds the base temperature in the heat affected zone, $T_{HAZ,b}(x, y, z)$ with a linear interpolation between the base boundary temperature of the nugget and the base boundary temperature of the heat affected zone. The load curve LCIDT governs the temperature evolution of the heat affected zone as described in [Remark 3](#).

***BOUNDARY_TEMPERATURE_TRAJECTORY**

Purpose: Apply a temperature boundary condition on nodes enclosed in either a cylindrical or rectangular prism volume moving along a trajectory. The center of the volume moves along the trajectory defined by a nodal path at a prescribed velocity. This keyword applies only to solid elements.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	PTYP	NSID1	SPD1	NSID2	SPD2		RELVEL
Type	I	I	I	F	I	F		I
Default	none	1	none	none	none	none		0

Card 2	1	2	3	4	5	6	7	8
Variable	IFORM	LCID	TMULT	LCROT	LCMOV	LCLAT		
Type	I	I	F	I	I	I		
Default	none	none	none	none	none	none		

Card 3	1	2	3	4	5	6	7	8
Variable	P1	P2	P3	P4	P5	P6	P7	P8
Type	F	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

Optional Volume Orientation Card. Additional card for NSID2 = 0.

Card 4	1	2	3	4	5	6	7	8
Variable	TX	TY	TZ					
Type	F	F	F					
Default	none	none	none					

VARIABLE**DESCRIPTION**

PID	Part ID or part set ID to which the temperature boundary condition will be applied
PTYP	PID type: EQ.1: Part ID EQ.2: Part set ID
NSID1	Node set defining the path of the moving volume. The moving volume travels along the path at speed SPD1. The nodes are traversed according to their order in the node set. See Remark 1 .
SPD1	Speed of the moving volume on the trajectory: GT.0.0: Constant speed LT.0.0: SPD1 is a load curve ID defining the speed as a function of time.
NSID2	Node or segment set that specifies the orientation of the moving volume's center axis. GT.0: NSID2 together with SPD2 define a curve in the same way that NSID1 and SPD1 define a curve. Orientation of the moving volume's center axis is defined as a vector pointing from the current position on NSID2 to the current position on NSID1. EQ.0: The moving volume's center axis is oriented as (TX, TY, TZ) input on Card 4. LT.0: NSID2 specifies a segment set. The moving volume's center axis is aligned with normals to segments in this set. To ensure that the axis orientation can be unambiguously determined at each point of the nodal path, LS-DYNA

VARIABLE	DESCRIPTION
	requires that each pair of consecutive nodes in NSID1 must both be in at least one segment of NSID2 . When the center of the moving volume is on a node that is part of more than one segment in NSID2 , the direction is determined by averaging the adjacent segment normals.
SPD2	Speed of reference point in NSID2 (ignored unless NSID2 > 0): GT.0: Constant speed LT.0: SPD2 is a load curve ID defining the speed as a function of time.
RELVEL	Flag for whether SPD1 and SPD2 are relative or absolute speeds in a thermo-mechanical coupled analysis: EQ.0: Absolute speeds EQ.1: Relative speeds with respect to underlying structures
IFORM	Geometric description of the moving volume. See Remark 3 for details.
LCID	Load curve ID for temperature as a function of time. EQ.0: Temperature is a constant defined by the value TMULT.
TMULT	Curve multiplier for temperature
LCROT	Load curve defining the rotation angle (in degrees) of the moving volume around the trajectory as a function of time. See Remark 2 .
LCMOV	Load curve defining the offset of the moving volume along its center axis as a function of time. See Remark 2 .
LCLAT	Load curve defining the lateral offset of the moving volume as a function of time. See Remark 2 .
Pi	Parameters defining the moving volume's geometry. The meaning of each parameter depends on field IFORM. See Remark 3 and Figure 5-12 for details.
TX, TY, TZ	Orientation vector of the moving volume's center axis in global coordinates (NSID2 = 0 only)

Remarks:

1. **Volume Motion.** If NSID1 consists of nodes belonging to a certain part, the moving volume will follow the motion of the part. By setting field RELVEL to 1 the velocity of the moving volume can be defined relative to the motion of the nodes in NSID1.
2. **Volume Orientation.** There is a natural local coordinate system $(\mathbf{r}, \mathbf{s}, \mathbf{t})$ associated with the motion of the moving volume. The relative velocity vector on the trajectory of the moving volume defines the "forward" direction \mathbf{r} . The orientation vector aligned with the moving volume's center axis is denoted by \mathbf{t} . \mathbf{s} is defined as $\mathbf{s} = \mathbf{t} \times \mathbf{r}$ by the right-handed rule and denotes the lateral direction.

The position and orientation of the moving volume can be adjusted by rotating and translating the local coordinate system. Note that the system is always first rotated around the vector \mathbf{r} by a value given by the load curve LCROT, resulting in a new local coordinate system $(\mathbf{r}, \mathbf{s}', \mathbf{t}')$. Then, the system is translated in directions \mathbf{t}' and \mathbf{s}' according to values specified by LCMOV and LCLAT, respectively.

3. **Volume Geometry.** The moving volume can be either a cylinder or rectangular prism, depending on the values defined in field IFORM (see Figure 5-12).

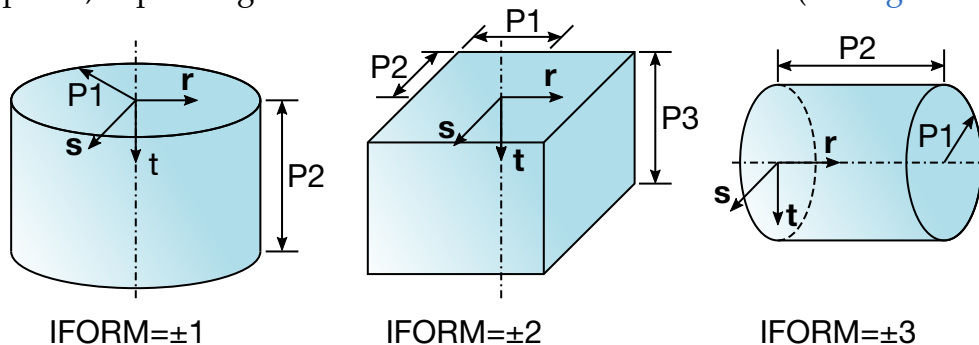


Figure 5-12. Definition of the geometry parameters

- a) *Cylinder (IFORM = 1 or IFORM = -1).* P1 denotes the radius of the cylinder and P2 denotes the height of the cylinder. In the case that IFORM = -1, P1 and P2 are load curves that define the cylinder radius and height as functions of time.
- b) *Rectangular Prism (IFORM = 2 or IFORM = -2).* P1 denotes half the length of the rectangular prism, P2 denotes half width of the rectangular prism, and P3 denotes the height of the rectangular prism. In the case that IFORM = -2, P1, P2 and P3 are load curves that define the prism's half length, half width and height as functions of time.

- c) *Radially Oriented Cylinder* ($IFORM = 3$ or $IFORM = -3$). P1 denotes the radius of the cylinder and P2 denotes the height of the cylinder. In the case that $IFORM = -3$, P1 and P2 are load curves that define the cylinder radius and height as functions of time. This cylinder differs from that defined by $IFORM = \pm 1$ in that the orientation vector, \mathbf{t} , is along a radial direction instead of the length of the cylinder.

*BOUNDARY

*BOUNDARY_THERMAL_BULKFLOW

*BOUNDARY_THERMAL_BULKFLOW_OPTION1_OPTION2

Purpose: Used to define bulk fluid flow elements.

OPTION1 is required since it specifies whether the BULKFLOW applies to an element or set.

ELEMENT

SET

OPTION2 if used turns on the fluid upwind algorithm

UPWIND

Card 1	1	2	3	4	5	6	7	8
Variable	EID/SID	LCID	MDOT					
Type	I	I	F					
Default	none	none	none					

VARIABLE

DESCRIPTION

EID / SID

Beam element ID (EID) for ELEMENT option
Beam set ID (SID) for SET option

LCID

Load Curve ID for mass flow rate versus time.

MDOT

Mass flow rate (e.g. kg/sec).

***BOUNDARY_THERMAL_BULKNODE**

Purpose: Used to define thermal bulk nodes.

Card 1	1	2	3	4	5	6	7	8
Variable	NID	PID	NBNSEG	VOL	LCID	H	AEXP	BEXP
Type	I	I	I	F	I	F	F	F
Default	none	none	0	none	0	0.	0.	0.

Bulk Node Cards. Include NBNSEG cards, one for each bulk node segment.

Card 2	1	2	3	4	5	6	7	8
Variable	N1	N2	N3	N4				
Type	I	I	I	I				

VARIABLE	DESCRIPTION
NID	Bulk node number.
PID	Bulk node part ID.
VOL	Bulk node volume.
NBNSEG	Number of element surface segments that transfer heat with this bulk node.
N1, N2, N3, N4	Nodal point numbers
LCID	Load curve ID for H
H	Heat transfer coefficient
AEXP	<i>a</i> exponent
BEXP	<i>b</i> exponent

Remarks:

The heat flow between a bulk node (T_B) and a bulk node segment (T_S) is given by

$$q = h(T_B^a - T_S^a)^b$$

1. For convection, set $a = b = 1$.
2. For radiation, set $a = 4$, $b = 1$.
3. For flux, set $a = b = 0$. Mathematically, anything to the 0 power is 1. This produces the expression, $(T_B^0 - T_S^0)^0 = (1 - 1)^0 = 0^0 = 1$. However, some computer operating systems don't recognize 0^0 . It is safer to set $a = b =$ very small number.

***BOUNDARY_THERMAL_WELD**

Purpose: Define a moving heat source to model welding. Only applicable for coupled thermal-structural simulations in which the weld source or work piece is moving.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	PTYP	NID	NFLAG	X0	Y0	Z0	N2ID
Type	I	I	I	I	F	F	F	I
Default	none	1	↓	1	↓	↓	↓	none

Card 2	1	2	3	4	5	6	7	8
Variable	a	b	cf	cr	LCID	Q	Ff	Fr
Type	F	F	F	F	I	F	F	F
Default	none	none	none	none	Q	none	none	none

Beam Aiming Direction Card. Additional card for N2ID = -1.

Card 3	1	2	3	4	5	6	7	8
Variable	TX	TY	TZ					
Type	F	F	F					
Default	none	none	none					

VARIABLE**DESCRIPTION**

PID

Part ID or Part Set ID to which weld source is applied

PTYP

PID type:

EQ.1: PID defines a single part ID

EQ.2: PID defines a part set ID

VARIABLE	DESCRIPTION
NID	Node ID giving location of weld source EQ.0: location defined by (X0, Y0, Z0) below
NFLAG	Flag controlling motion of weld source EQ.1: source moves with node NID EQ.2: source is fixed in space at original position of node NID
X0, Y0, Z0	Coordinates of weld source, which remains fixed in space (ignored if NID is nonzero)
N2ID	Second node ID for weld beam aiming direction GT.0: beam is aimed from N2ID to NID, moves with these nodes EQ.-1: beam aiming direction is (TX, TY, TZ) input on Card 3
a	Weld pool radius (i.e., half width)
b	Weld pool depth (in beam aiming direction)
cf	Weld pool forward direction
cr	Weld pool rearward direction
LCID	Load curve ID for weld energy input rate as a function of time EQ.0: use constant multiplier value Q.
Q	Curve multiplier for weld energy input rate [energy/time, e.g., Watt] LT.0.0: use absolute value and accurate integration of heat
Ff	Forward distribution function
Fr	Rear distribution function (Note: $F_f + F_r = 2.0$)
TX, TY, TZ	Weld beam direction vector in global coordinates (N2ID = -1 only)

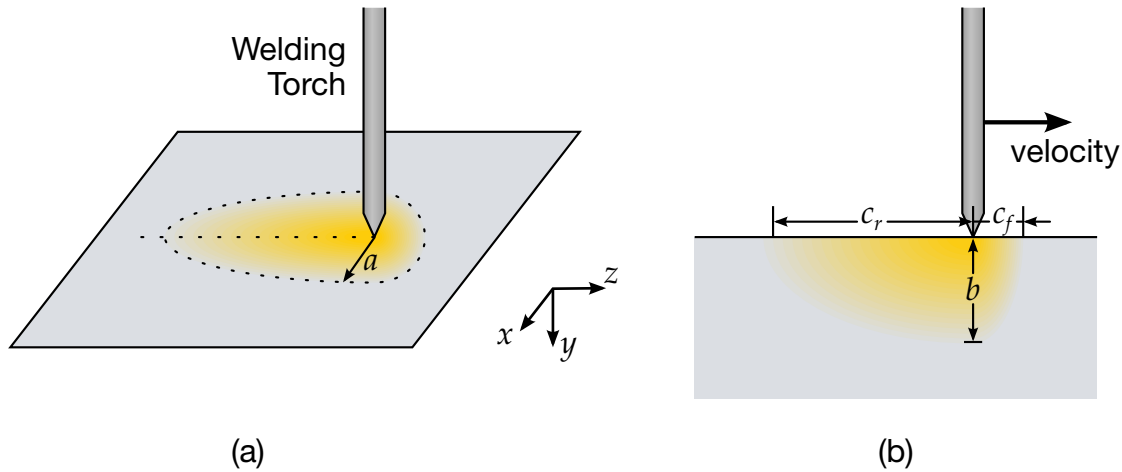


Figure 5-13. Schematic illustration of welding with moving torch. The left figure (a) shows the surface of the material from above, while the right figure (b) shows a slice along the dotted line in the y - z plane.

Remarks:

1. **Boundary Condition Model.** This boundary condition allows simulation of a moving weld heat source, following the work of Goldak, Chakravarti, and Bibby [1984]. Heat is generated in an ellipsoidal region centered at the weld source, and decaying exponentially with distance according to:

$$q = \frac{6\sqrt{3}FQ}{\pi\sqrt{\pi abc}} \exp\left(\frac{-3x^2}{a^2}\right) \exp\left(\frac{-3y^2}{b^2}\right) \exp\left(\frac{-3z^2}{c^2}\right),$$

where

$$\begin{aligned} q &= \text{weld source power density} \\ (x, y, z) &= \text{coordinates of point } p \text{ in weld material} \\ F &= \begin{cases} F_f & \text{if point } p \text{ is in front of beam} \\ F_r & \text{if point } p \text{ is behind beam} \end{cases} \\ c &= \begin{cases} c_f & \text{if point } p \text{ is in front of beam} \\ c_r & \text{if point } p \text{ is behind beam} \end{cases} \end{aligned}$$

A local coordinate system center at the heat source is constructed as shown in [Figure 5-13](#). The relative velocity vector of the heat source defines the "forward" direction, so material points that are approaching the heat source are in "front" of the beam. The beam aiming direction is used to compute the weld pool depth. The weld pool width is measured normal to the relative velocity - aiming direction plane.

If Q is defined as a negative quantity in the input, then the formula above uses the absolute value of Q , and a more accurate integration of the heat source is performed with some additional cost in CPU time.

2. **Fixed Welding Torch Simulation.** To simulate a welding process during which the welding torch is fixed in space, NID and N2ID must be set to 0 and -1 respectively. The X0, Y0, and Z0 fields specify the global coordinates of the welding torch, and the TX, TY, and TZ fields specify the direction of the welding beam. The motion of the work piece is prescribed using the *BOUNDARY_PRESCRIBED_MOTION keyword.
3. **Fixed Work Piece Simulation.** To simulate a welding process for which the work piece fixed in space, NID and N2ID specify both the beam source location and direction. The X0, Y0, Z0, TX, TY, and TZ fields *are ignored*. The motion of welding source is prescribed with using the *BOUNDARY_PRESCRIBED_MOTION keyword applied to the two nodal points specified in the NID and N2ID fields.

***BOUNDARY_THERMAL_WELD_TRAJECTORY**

Purpose: Define a moving heat source to model welding of solid or shell structures. Motion of the source is described by a nodal path and a prescribed velocity on this path. This keyword is applicable in coupled thermal-structural and thermal-only simulations (see [Remark 1](#)) and also supports thermal dumping. Different equivalent heat source descriptions are implemented.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	PTYP	NSID1	SPD1	NSID2	SPD2	NCYC	RELVEL
Type	I	I	I	F	I	F	I	I
Default	none	1	none	none	none	none	1	0

Card 2	1	2	3	4	5	6	7	8
Variable	IFORM	LCID	Q	LCROT	LCMOV	LCLAT	DISC	ENFOR
Type	I	I	F	I	I	I	F	I
Default	none	none	none	none	none	none	none	0

Card 3	1	2	3	4	5	6	7	8
Variable	P1	P2	P3	P4	P5	P6	P7	P8
Type	F	F	F	F	F	F	F	F

Optional Weld Source Aiming Direction Card. Additional card for NSID2 = 0.

Card 4	1	2	3	4	5	6	7	8
Variable	TX	TY	TZ					
Type	F	F	F					
Default	none	none	none					

VARIABLE**DESCRIPTION**

PID

Part ID or part set ID of solid and/or shell elements that are heated by the weld source. In the case of shell elements present in the heated structure, the thermal thick element formulation must be activated. See field THSHEL on *CONTROL_SHELL or field ITHELFM on *SECTION_SHELL_THERMAL.

PTYP

PID type:

EQ.1: Part ID

EQ.2: Part set ID

NSID1

Node set defining the path of the weld source. The source travels along the path at speed SPD1. The nodes are traversed according to their ordering in the node set. See [Remark 1](#).

SPD1

Speed of the heat source on the weld trajectory:

GT.0.0: Constant speed

LT.0.0: |SPD1| is a load curve ID defining weld speed as a function of time.

NSID2

Node or segment set containing information for the weld source aiming direction:

GT.0: NSID2 together with SPD2 define a curve in the same way that NSID1 and SPD1 define a curve. Aiming direction is taken to be the vector pointing from the current position along NSID2 (for example your hand holding the torch) to the current position on NSID1 (the weld source).

EQ.0: Beam aiming direction is (TX, TY, TZ), input on Card 4.

LT.0: |NSID2| specifies a segment set. The heat source is aimed perpendicularly to segments in this set. To ensure that a

VARIABLE	DESCRIPTION
	direction can be unambiguously associated with each point of the weld path, LS-DYNA requires that each pair of consecutive nodes in NSID1 must <i>both</i> be in at least one segment of NSID2 . When the source is on a node that part of more than one segment in NSID2 , then the direction is determined by averaging the adjacent segment normal vectors.
SPD2	Speed of reference point in NSID2 (ignored unless NSID2 > 0) GT.0: Constant speed LT.0: SPD2 is a load curve ID defining weld speed as a function of time.
NCYC	Number of sub-steps for sub-cycling in evaluation of boundary condition. Allows thermal dumping. See Remark 3 .
RELVEL	Defines if SPD1 and SPD2 are relative or absolute speeds in coupled simulations EQ.0: Absolute speeds EQ.1: Relative speeds with respect to underlying structure
IFORM	Geometry description for energy rate density distribution (see Remark 4): EQ.1: Goldak-type heat source EQ.2: Double ellipsoidal heat source with constant density EQ.3: Double conical heat source with constant density EQ.4: Frustum-shaped heat source with constant density EQ.5: User-defined function
LCID	Load curve ID for weld energy input rate as a function of time EQ.0: Use constant multiplier value Q .
Q	Curve multiplier for weld energy input rate [energy/time] LT.0: Take absolute value and accurate integration of heat using integration cells with edge length DISC.
LCROT	Load curve defining the rotation (angle in degrees) of weld source around the trajectory as function of time. See Remark 2 .

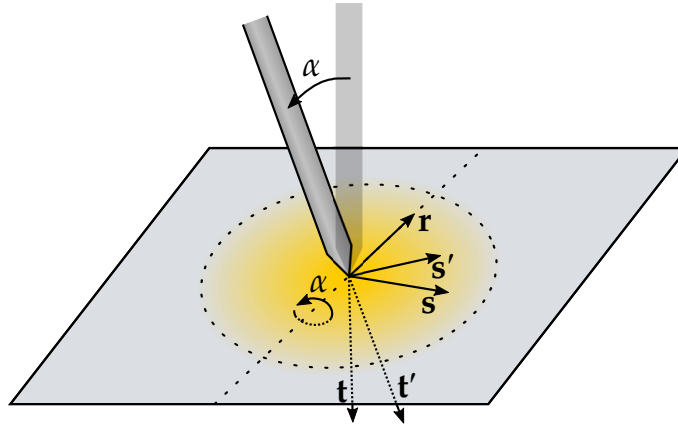


Figure 5-14. Local coordinates of the welding torch. These coordinates can be rotated about the r -axis to tilt the welding torch, forming a new set of local coordinates.

VARIABLE	DESCRIPTION
LCMOV	Load curve for offset of weld source in direction of the weld beam as function of time. See Remark 2 .
LCLAT	Load curve for lateral offset of weld source as function of time. See Remark 2 .
DISC	Resolution for accurate integration. Parameter defines edge length for integration cubes. Default is 5% of weld pool depth.
ENFOR	Flag for heat input enforcement option. If set, the nodal heat input is scaled such that the resulting heat inputs equals the user input as given by Q and $LCID$.
P_i	Parameters defining for weld pool geometry, depending on field IFORM. See Remark 4 and Table 5-1 for details.
TX, TY, TZ	Weld beam direction vector in global coordinates (NSID2 = 0 only)

Remarks:

- Heat source motion.** As with the *BOUNDARY_THERMAL_WELD card this card can be applied to coupled thermal-structure simulations. Additionally, since the source motion can be defined independently from the motion of nodes, this card can be applied to thermal-only simulations. This keyword applies to both solid and thermal thick shells.

If NSID1 consists of work piece nodes, then the heat source will follow the motion of the work piece. By setting field RELVEL to 1 the velocity of the heat

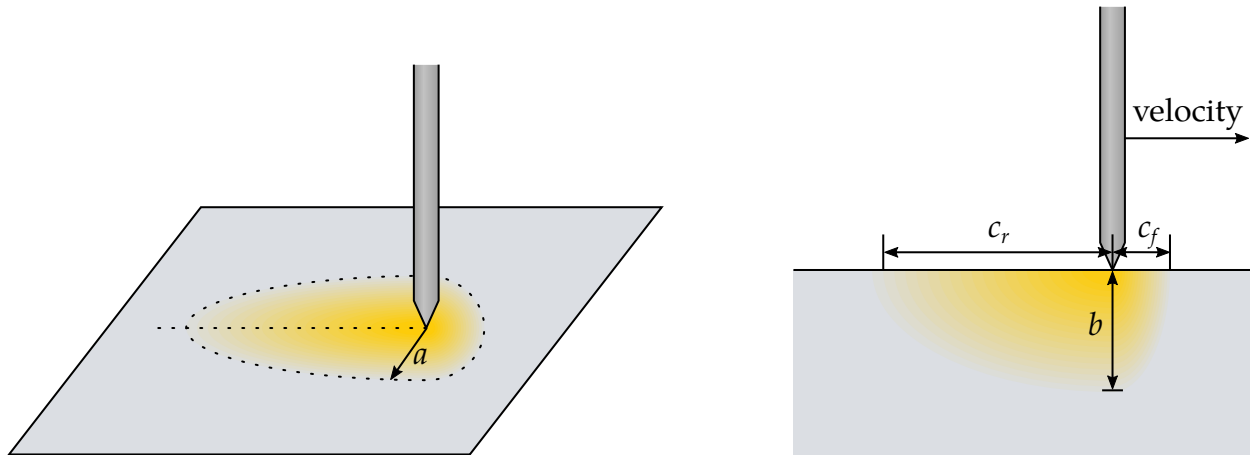


Figure 5-15. Heat source distribution geometry for IFORM = 1 and IFORM = 2

source can be defined relative to the motion of the nodes in NSID1 (the work piece).

2. **Heat source direction (tilting the torch).** There is a natural local coordinate system ($\mathbf{r}, \mathbf{s}, \mathbf{t}$) associated with the motion of the heat source. The relative velocity vector on the trajectory of the heat source defines the "forward" direction \mathbf{r} , so that material points that are approaching the heat source are in "front" of the beam. The weld source aiming direction, denoted by \mathbf{t} , determines the direction of the weld pool depth. The coordinate direction \mathbf{s} is normal to the plane containing both the relative velocity and the aiming direction; note that $\mathbf{s} = \mathbf{t} \times \mathbf{r}$ to have a right handed coordinate system. It determines the direction of the weld pool width.

The position and aiming direction of the heat source (for example tilting the torch) can be adjusted by rotating and translating the local coordinate system. To do this, the system is first rotated around the vector \mathbf{r} by a value given by the load curve LCROT, resulting in a new local coordinate system ($\mathbf{r}, \mathbf{s}', \mathbf{t}'$). Then, the system is translated in directions \mathbf{t}' and \mathbf{s}' using LCMOV and LCLAT, respectively. See [Figure 5-14](#).

3. **Weld source sub-cycling.** The sub-cycling method introduces an individual time step size for the weld source evaluation. During each step of the heat transfer solver, NCYC steps are used to determine the energy rate distribution of the weld source. In each sub-step, the geometry of the weld pool is updated. Therefore, even with larger thermal time steps a relatively smooth temperature field around the weld source can be obtained and a jumping heat source across elements can be suppressed.
4. **Energy rate distribution geometries.** Several different heat source geometries that can be used with this keyword are described below. The local coordinate system needed for the description of the geometry is discussed in [Remark 2](#).

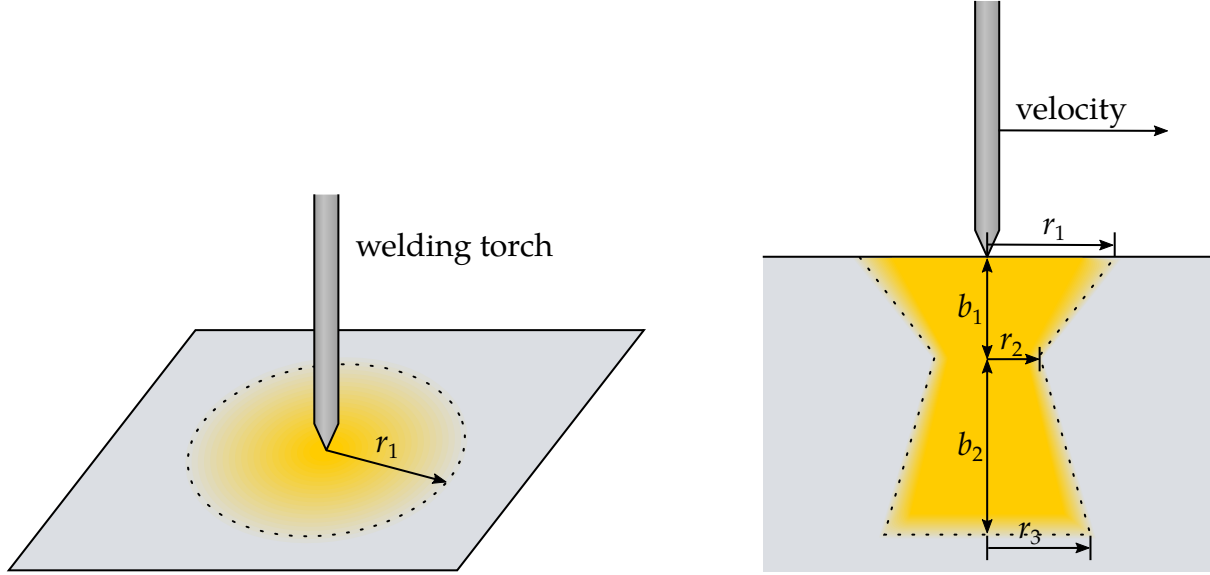


Figure 5-16. Heat source distribution geometry for IFORM = 3

For IFORM = 1 heat is generated in an ellipsoidal region centered at the weld source and decaying exponentially with distance according to the work of Goldak, Chakravarti, and Bibby [1984]. Energy rate distribution is governed by

$$q = \frac{2n\sqrt{nFQ}}{\pi\sqrt{\pi abc}} \exp\left(\frac{-nx^2}{a^2}\right) \exp\left(\frac{-ny^2}{b^2}\right) \exp\left(\frac{-nz^2}{c^2}\right),$$

where:

$$F = \begin{cases} F_f & \text{if point } p \text{ is in front of beam} \\ F_r & \text{if point } p \text{ is behind beam} \end{cases}$$

$$c = \begin{cases} c_f & \text{if point } p \text{ is in front of beam} \\ c_r & \text{if point } p \text{ is behind beam} \end{cases}$$

The local coordinates of point p are denoted by (x, y, z) and it is expected that the sum of the weighting factors F_f, F_r equals 2. The half-width of the ellipsoid is given by a , the welding depth by b . See [Figure 5-15](#). The complete set of parameters $(a, b, c_f, c_r, F_f, F_r, n)$ is input in the fields P1 to P7; see [Table 5-1](#).

The energy rate density q for IFORM = 2 is assumed to be constant in the double ellipsoidal region as defined for Goldak-type heat sources. Its value is given by

$$q = \frac{3FQ}{2\pi abc}$$

with the same assumptions for F and c as above. The set of parameters consequently reduces to $(a, b, c_f, c_r, F_f, F_r)$ which are input in fields P1 to P6.

In contrast to the above, IFORM = 3 defines an equivalent heat source with a constant energy rate density on a double conical region. The shape is defined

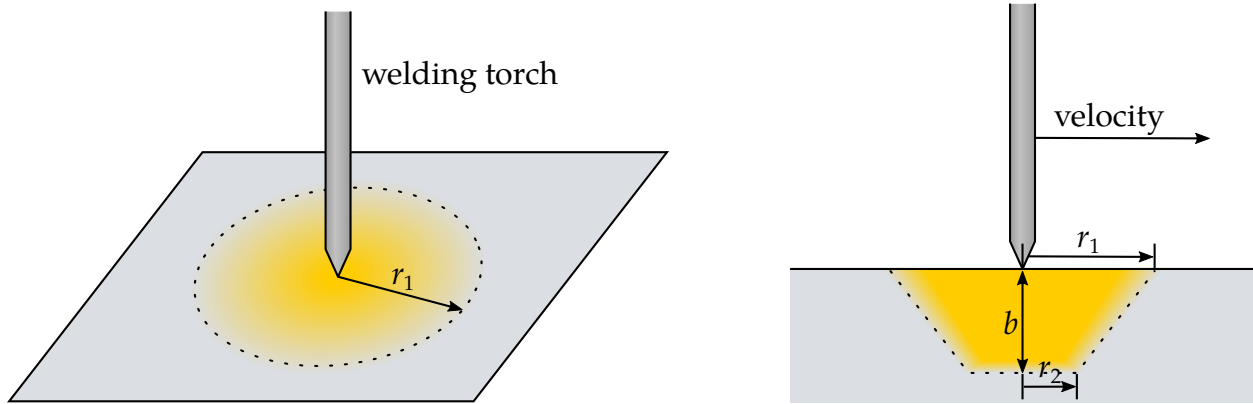


Figure 5-17. Heat source distribution geometry for IFORM = 4

by three radii r_1, r_2, r_3 and two values b_1, b_2 defining the heights of the two parts of the shape. See [Figure 5-16](#). The respective power densities of the parts are given by

$$q_i = \frac{3F_i Q}{2\pi b_i (r_i^2 + r_{i+1}^2 + r_i r_{i+1})}, \quad i = 1, 2.$$

The parameter set $(r_1, r_2, r_3, b_1, b_2, F_1, F_2)$ is input in fields P1 to P7.

The last pre-defined form is IFORM = 4, which defines a constant power density over a frustum shaped region. See [Figure 5-17](#). The density and the shape can easily be described using three geometrical parameters P1 to P3 corresponding to the radii r_1 (at the heat source origin) and r_2 and the height b :

$$q = \frac{3Q}{\pi b (r_1^2 + r_2^2 + r_1 r_2)}$$

Finally, with IFORM = 5 the user can define an arbitrarily shaped equivalent heat source with the *DEFINE_FUNCTION keyword. The input of the function comprises the coordinates in the local coordinate system $(\mathbf{r}, \mathbf{s}', \mathbf{t}')$, as discussed in [Remark 2](#), the time and the weld speed. It may thus read as follows:

```
*DEFINE_FUNCTION
1,user heat source
heat(r,s,t,time,weldspd) = ...
```

To speed-up the computation, the function is evaluated for nodes within a distance less than a from the current center position. Consequently, the function ID and this distance are the only parameters needed in Card 3 of the input.

IFORM	1	2	3	4	5
P1	a	a	r_1	r_1	FID
P2	b	b	r_2	r_2	a
P3	c_f	c_f	r_3	b_1	
P4	c_r	c_r	b_1		
P5	F_f	F_f	b_2		
P6	F_r	F_r	F_1		
P7	n		F_2		
P8					

Table 5-1. Parameters defining each weld pool geometry

***BOUNDARY_USA_SURFACE**

Purpose: Define a surface for coupling with the USA code [DeRuntz 1993]. The outward normal vectors should point into the fluid media. The coupling with USA is operational in explicit transient and in implicit natural frequency analyses.

Card	1	2	3	4	5	6	7	8
Variable	SSID	WETDRY	NBEAM					
Type	I	I	I					
Default	none	0	0					

VARIABLE**DESCRIPTION**

SSID

Segment set ID, see *SET_SEGMENT

WETDRY

Wet surface flag:

EQ.0: Dry, no coupling for USA DAA analysis, or Internal fluid coupling for USA CASE analysis

EQ.1: Wet, coupled with USA for DAA analysis, or External fluid coupling for USA CASE analysis

NBEAM

The number of nodes touched by USA Surface-of-Revolution (SOR) elements. It is not necessary that the LS-DYNA model has beams where USA has beams (i.e., SOR elements), merely that the LS-DYNA model has nodes to receive the forces that USA will return.

Remarks:

The underwater shock analysis code is an optional module. To determine availability, contact Ansys sales (<https://www.ansys.com/contact-us>).

The wet surface of 3 and 4-noded USA general boundary elements is defined in LS-DYNA with a segment set of 4-noded surface segments, where the fourth node can duplicate the third node to form a triangle. The segment normal vectors should be directed into the USA fluid. If USA overlays are going to be used to reduce the size of the DAA matrices, the user should nonetheless define the wet surface here as if no overlay were being used.

If Surface-of -Revolution elements (SORs) are being used in USA, then NBEAM should be non-zero on one and only one card in this section.

The wet surface defined here can cover structural elements or acoustic fluid volume elements, but it cannot touch both types in one model.

When running a coupled problem with USA, the procedure requires an additional input file of USA keyword instructions. These are described in a separate USA manual. The name of this input file is identified on the command line with the **usa = flag**:

LSDYNA.USA **i=inf** **usa=uin**

where **uin** is the USA keyword instruction file.