LS-DYNA®
KEYWORD USER'S MANUAL

VOLUME IV

Multiscale Solvers

09/09/21 (r:14083)
LS-DYNA Dev

LIVERMORE SOFTWARE TECHNOLOGY (LST), AN ANSYS COMPANY

Support Addresses

Livermore Software Technology
7374 Las Positas Road
Livermore, California 94551
Tel: 925-449-2500 ♦ Fax: 925-449-2507
Website: **www.lstc.com**

Livermore Software Technology
1740 West Big Beaver Road
Suite 100
Troy, Michigan 48084
Tel: 248-649-4728 ♦ Fax: 248-649-6328

Disclaimer

TABLE OF CONTENTS

TABLE OF CONTENTS

Introduction

Modern computer aided engineering (CAE) simulation often deals with phenomena of intricate multi-physics that are governed by various mechanisms integrated across several levels of detail. Multiscale modeling/analysis, in essence the consideration of problems across different spatial scales, is one of the major recent developments in CAE software. It has developed into a large spectrum of techniques in computational mechanics with different meanings and flavors depending on the applications.

In LS-DYNA, the development of multiscale solvers aims to bridge manufacturing results with structure analysis for the next-generation virtual product design and engineering. Therefore, there are three main paradigms, which are the unifying threads of this manual – Material Processing and Manufacturing (MPM) simulation, Advanced Dynamic Failure Analysis (ADFA), and Multiscale Analysis (MSA). These paradigms are not completely independent of each other. In many cases, the pursuit of the comprehensive model may require their simultaneous combined efforts.

In the first Chapter, several LS-DYNA numerical methods are introduced to offer accurate solutions gaining insight into the interactions between material flow fields and underlying material properties in a wide range of MPM simulations. Those numerical methods help engineers to analyze and improve the manufacturing process in obtaining the optimal quality of material strength for the structure analysis.

In the second Chapter, LS-DYNA ADFA assists engineers to understand how a product structure in the component level or assembly reacts under stress or critical loading conditions. In particular, we emphasize on the dynamic failure analysis. Several LS-DYNA numerical methods and their keywords are introduced in this chapter to provide the essential information using those advanced numerical methods in dynamics material and structural analyses.

In the physical world, product structure is a hierarchical decomposition of a product. LS-DYNA MSA in the next two chapters helps analyze how the hierarchical product structure controls the linkages between MPM and ADFA. LS-DYNA MSA is the major integrating theme of this manual, as indicated by its manual title. The current MSA development focuses on two multiscale analyses. While Multiscale Material Analysis seeks to exchange the nonlinear material characteristics between micro and macro scales (e.g., microstructure in a solder), the Multiscale Geometrical Analysis focuses on the linkage of geometrical information between meso/macro and larger structure scales (e.g., solders in a circuit board). They are deep finite element analyses for profound physics, and will be our major on-going development for the years to come.

Besides the continuation of furnishing the keywords and their application perspective in this manual, theory behind those numerical methods are also important for the advanced

simulation. A theory manual will be prepared in addition to this Users' manual in the future.

In what follows, we describe all numerical methods and their main keywords throughout four chapters. Since some numerical methods are interconnected, they inevitably go from-side-to-side depending on the topic of applications.

Material Processing and Manufacturing

LS-DYNA Material Processing and Manufacturing (MPM) is an integration of advanced numerical solvers for MPM simulations from basic metal forging process to complex reflow solding.

The MPM package utilizes FEM, Element-free Galerkin (EFG) method, Smoothed Particle Galerkin (SPG) method, Incompressible SPG (ISPG) and Smoothed Particle Hydrodynamics (SPH).

The MPM applications include but not limited to: Forging, Extrusion, Screwing, Machining, Joining, Stir Friction Welding, Compression Molding, Reflow Soldering, and many other material processing and manufacturing simulations involving material removal, thermal-mechanical effect, and severe deformation. A wide range of materials from metal, concrete, wood, foam to rubber and composite can be covered in MPM simulations.

In this chapter, we present the adaptive FEM/EFG, ISPG, and SPH methods. The SPG method will be introduced in the next chapter.

Adaptive FEM/EFG

Manufacturing process usually involves large deformation of the bulk materials, which requires special numerical method to conduct the simulation. Adaptivity is one of the methods suitable for the manufacturing process analysis.

LS-DYNA adaptivity includes the adaptive FEM and adaptive EFG methods. Both methods share many keywords while the adaptive EFG has more features.

Adaptive FEM/EFG brings an accurate solution with enriched explicit/implicit analysis features for a wide variety of manufacturing simulations in metal forging, punching, drawing and powder sintering processes. It is a unique solver to model compression molding process of long fiber reinforced composite material. It can be applied to many material processing simulations involving large material deformation such as needle insertion simulation in bio-medical application.

The adaptive FEM/EFG solver includes the following major keywords:

*CONSTRAINED_BEAM_IN_SOLID

*CONTROL_ADAPTIVE

*CONTROL_REMESHING_{EFG}

*DEFINE_BOX_ADAPTIVE

*SECTION_SOLID_{EFG}

It may also use other keywords, such as *PART (to set adaptive part, ADPOPT = 2 or 3), *BOUNDARY_SPC_SYMMETRY_PLANE, *CONSTRAINED_GLOBAL/LOCAL, *CONTACT, etc. Please refer to the Keyword Manual Volume I for detailed description of those keywords.

*CONSTRAINED_BEAM_IN_SOLID_{OPTION1}_{OPTION2}

This keyword could take the following two forms:

*CONSTRAINED_BEAM_IN_SOLID

*CONSTRAINED_BEAM_IN_SOLID_PENALTY

To define a CONSTRAINED_BEAM_IN_SOLID ID and heading the following options are available:

ID

TITLE

Purpose: This keyword provides either constraint-based or penalty-based coupling between beams embedded in solid or thick shell elements. For shells embedded in solid or thick shell elements; see *CONSTRAINED_SHELL_IN_SOLID.

*CONSTRAINED_BEAM_IN_SOLID invokes constraint-based coupling. It constrains beam structures to move with Lagrangian solids/tshells, which serve as the master component. Both acceleration and velocity are constrained. Notable features of this keyword include:

1. **CDIR = 1 Feature.** With the CDIR = 1 option, coupling occurs only in the normal directions. This coupling allows for releasing the constraints along the beam axial direction.

2. **Axial Coupling Force.** Debonding processes can be modeled with a user defined function or user provided subroutine giving the axial shear force based on the slip between rebar nodes and concrete solid elements. This feature is invoked by setting the AXFOR flag to a negative integer which refers to the *DEFINE_FUNCTION ID or a number greater than 1000. In the latter case, the user must modify the subroutine rebar_bondslip_get_force() in dyn21.F to add in one or more debonding laws, each tagged with a "lawid." An AXFOR value greater than 1000 will call the user subroutine and pass AXFOR in as "lawid." CDIR must be set to 1 in this case to release the axial constraints.

3. **NCOUP Feature.** NCOUP is used to invoke coupling not only at beam nodes but also at coupling points between the two beam nodes of each beam element. This feature improves the energy balance when the ratio of the length of beam to solid element size is large.

4. **Tetrahedral and Pentahedral Solid Elements.** These element shapes are supported in this keyword.

5. **Constraints on Nodes.** Beam nodes can be embedded in elements whose nodes have prescribed motion or other constraints.

6. **R-Adaptivity Support.** Severely deformed solid elements could terminate the simulation prematurely. One way to address this problem is to use LS-DYNA's 3D tetrahedral *r*-adaptivity method to remesh and restart. This keyword supports *r*-adaptivity and can be used to model manufacturing processes with severe deformations, such as compression molding with embedded fibers.

7. **Implicit Support.** Both SMP and MPP are supported.

8. **Optimized Sorting.** The sorting subroutine is optimized for larger problems to achieve better performance with less memory usage.

9. **Thermal Support.** In case the implicit thermal solver is invoked to perform either a thermal analysis (SOLN = 1 in *CONTROL_SOLUTION) or a coupled structural thermal analysis (SOLN = 2 in *CONTROL_SOLUTION), the temperature field is also constrained between beam and solid nodes.

*CONSTRAINED_BEAM_IN_SOLID_PENALTY invokes penalty-based coupling. A penalty spring is attached between coupling points on the beam and in the solid/tshell element. Penalty spring stiffness is calculated based on the geometric mean of beam and solid's bulk modulus. The magnitude of this coupling force can be controlled through field PSSF (penalty spring stiffness scale factor). This penalty coupling conserves kinetic energy much better in transient problems, such as blast loading.

If COUPID is not defined on the Title Card, LS-DYNA will automatically create an ID value for the coupling.

Title Card. Additional card for TITLE and ID keyword options.

Title	1	2	3	4	5	6	7	8
Variable	COUPID	TITLE						
Type	I	A70						

Card 1	1	2	3	4	5	6	7	8
Variable	SLAVE	MASTER	SSTYP	MSTYP			NCOUP	CDIR
Type	I	I	I	I			I	I
Default	none	none	0	0			0	0

Card 2	1	2	3	4	5	6	7	8
Variable	START	END		AXFOR		PSSF		XINT
Type	F	F		I		F		F
Default	0	10^{10}		0		0.1		10^{16}

VARIABLE	DESCRIPTION
COUPID	Coupling (card) ID number (I10). If not defined, LS-DYNA will assign an internal coupling ID based on the order of appearance in the input deck.
TITLE	A description of this coupling definition.
SLAVE	Slave set ID defining a part or part set ID of the Lagrangian beam structure (see *PART, *SET_PART).
MASTER	Master set ID defining a part or part set ID of the Lagrangian solid elements or thick shell elements (see *PART or *SET_PART). See Remark 1.
SSTYP	Slave set type of SLAVE: EQ.0: Part set ID (PSID) EQ.1: Part ID (PID)
MSTYP	Master set type of MASTER: EQ.0: Part set ID (PSID) EQ.1: Part ID (PID)

VARIABLE	DESCRIPTION
NCOUP	Number of coupling points generated in one beam element. If set to 0, coupling only happens at beam nodes. Otherwise, coupling is done at both the beam nodes and those automatically generated coupling points.
CDIR	Coupling direction. Only available in constraint form. EQ.0: Constraint applied along all directions. EQ.1: Constraint only applied along normal directions; along the beam axial direction there is no constraint.
START	Start time to activate the coupling: LT.0: Start time is set to |START|. When negative, start time is followed during the dynamic relaxation phase of the calculation. After dynamic relaxation has completed, coupling is activated regardless of the value of END. EQ.0: Start time is inactive, meaning coupling is always active GT.0: If END = -9999, START is interpreted as the curve or table ID defining multiple pairs of start-time and end-time. Otherwise, if END > 0, start time applies both during and after dynamic relaxation.
END	End time to deactivate the coupling: LT.0: If END = -9999, START is interpreted as the curve or table ID defining multiple pairs of start-time and end-time. Otherwise, negative END indicates that coupling is inactive during dynamic relaxation. After dynamic relaxation the start and end times are followed and set to |START| and |END|, respectively. EQ.0: END defaults to 10^{20}. GT.0: END sets the time at which the coupling is deactivated.

VARIABLE	DESCRIPTION
AXFOR	ID of a user defined function that calculates the coupling force in the beam axial direction. Only available in constraint form. GE.0: Off LT.0: |AXFOR| is a function ID; see *DEFINE_FUNCTION. GT.1000: Debonding law ID, "lawid," in the user defined subroutine rebar_bondslip_get_force().
PSSF	Penalty spring stiffness scale factor. Only available in penalty form.
XINT	Interval distance. This field is designed to deal with beam elements having a wide variation in lengths. Coupling points are generated at an interval of length equal to XINT. Hence the number of coupling points in a beam element is no longer a fixed number (NCOUP), but rather variable, depending on the length of the beam element. This field can be used together with NCOUP. In that case, in each element, we will take the larger number of coupling points from these two options.

Remarks:

1. **Rigid Body.** The solid part cannot be defined as rigid body. If the solid part shares nodes with rigid bodies and constraint-based coupling is used, make sure rebars are not buried in elements which contain rigid body nodes. This is because constraint-based coupling distribute/collect momentum/force between rebar and solid nodes.

Examples:

1. The example below shows how to define a function and use it to prescribe the debonding process. User defined functions are supported. The function computes the debonding force and has two internally calculated arguments: slip and leng. Slip is the relative axial displacement between the beam node (or coupling point) and the material in which the beam is embedded. Leng is the tributary length of the beam node or coupling point. Implicit calculations require a third argument which is output by the function: stiff. Stiff is the debonding spring stiffness. The asterisk in front of stiff (*stiff) is required to indicate that it is called-by-reference, meaning that its value is returned after the function is evaluated. Please note that in the function body this asterisk cannot be placed in the

first column because the LS-DYNA keyword reader would assume that it is the start of a new keyword.

```
$...|....1....|....2....|....3....|....4....|....5....|....6....|....7....|....8
*CONSTRAINED_BEAM_IN_SOLID
$#   slave    master     sstyp     mstyp                          ncoup      cdir
         2         1         1         1                              2         1
$#   start       end               axfor
     0.000     0.000                 -10
*DEFINE_FUNCTION
        10
float force(float slip,float leng, float *stiff)
{
  float force,pi,d,area,shear,pf;
  pi=3.1415926;
  d=0.175;
  area=pi*d*leng;
  pf=1.0;
  if (slip<0.25) {
    shear=slip*pf;
  } else {
    shear=0.25*pf;
  }
  force=shear*area;
  *stiff=pf*area;
  return force;
}
```

2. The example below shows how to define a user subroutine and use it to prescribe the debonding process.

```
$...|....1....|....2....|....3....|....4....|....5....|....6....|....7....|....8
*CONSTRAINED_BEAM_IN_SOLID
$#   slave    master     sstyp     mstyp                          ncoup      cdir
         2         1         1         1                              2         1
$#   start       end               axfor
     0.000     0.000                1001
*CONSTRAINED_BEAM_IN_SOLID
$#   slave    master     sstyp     mstyp                          ncoup      cdir
         3         1         1         1                              2         1
$#   start       end               axfor
     0.000     0.000                1002
*USER_LOADING
$    parm1     parm2     parm3     parm4     parm5     parm6     parm7     parm8
       1.0       6.0
$...|....1....|....2....|....3....|....4....|....5....|....6....|....7....|....8
```

The user debonding law subroutine:

```
      subroutine rebar_bondslip_get_force(slip,dl,force,hsv,
     .  userparm,lawid)
      real hsv
      dimension hsv(12),cm(8),userparm(*)
c
c in this subroutine user defines debonding properties and
c call his own debonding subroutine to get force
      cm(1)=userparm(1)
      cm(2)=userparm(2)
      cm(3)=2.4*(cm(2)/5.0)**0.75
      cm(8)=0.
c
      pi=3.1415926
      d=0.175
      area=pi*0.25*d*d*dl
      pf=1.0
```

```
c
      if (lawid.eq.1001) then
        if (slip.lt.0.25) then
          shear=slip*pf
        else
          shear=0.25*pf
        endif
        force=sign(1.0,slip)*shear*area
      elseif (lawid.eq.1002) then
        if (slip.lt.0.125) then
          shear=slip*pf
        else
          shear=0.125*pf
        endif
      endif
      return
      end
```

*CONTROL_ADAPTIVE

Purpose: Activate 3D adaptive remeshing for applications, such as bulk metal forming. The field ADPOPT in *PART identifies the part(s) whose mesh to be adapted.

Available adaptivity types:

- Solid *r*-adaptivity in which a new mesh of tetrahedrons is created (ADPOPT = 2).
- 3D axisymmetric (or orbital) adaptivity of axisymmetric geometries comprised of hexahedral and/or pentahedral solids (ADPOPT = 3).

Remeshing hyperelastic materials or material models based on a total Lagrangian formulation may lead to numerical instabilities or inaccurate results.

For complete description of this keyword, please refer to the Keyword Manual Volume I.

Card 1	1	2	3	4	5	6	7	8
Variable	ADPFREQ		ADPTYP		TBIRTH	TDEATH	LCADP	
Type	F		I		F	F	I	
Default	none		1		0.0	10^{20}	0	

VARIABLE	DESCRIPTION
ADPFREQ	Time interval between adaptive remeshings.
ADPTYP	Adaptive options: EQ.7: 3D *r*-adaptive remeshing for solid elements. Tetrahedrons are used in the adaptive remeshing process (solid formulation 10 or 13, or if EFG, formulation 42), or in the case of 3D axisymmetry (orbital) adaptivity, hexahedral and pentahedral elements are used in the adaptive remeshing. A completely new mesh is generated which is initialized from the old mesh using a least squares approximation. The mesh size is currently based on the minimum and maximum edge lengths defined on the *CONTROL_REMESHING keyword input.

VARIABLE	DESCRIPTION
TBIRTH	Birth time at which the adaptive remeshing begins.
TDEATH	Death time at which the adaptive remeshing ends.
LCADP	Load curve ID, defining how the adaptive interval is changed as a function of time. If this option is nonzero, the ADPFREQ will be replaced by LCADP. The *x*-axis is time, and the *y*-axis is the varied adaptive time interval.

Card 2	1	2	3	4	5	6	7	8
Variable				ADPENE		MEMORY		
Type				F		I		
Default				0.0		inactive		

VARIABLE	DESCRIPTION
ADPENE	For three dimensional *r*-adaptive solid remeshing (ADPOPT = 2 in *PART), the mesh refinement is based on the curvature of the tooling when ADPENE is positive. See Remark 1.
MEMORY	This flag can have two meanings depending on whether the memory environmental variable is or is not set. The command "setenv LSTC_MEMORY auto" (or for bourne shell "export LSTC_MEMORY=auto") sets the memory environmental variable which causes LS-DYNA to expand memory automatically. Note that automatic memory expansion is not always 100% reliable depending on the machine and operating system level; consequently, it is not yet the default. To see if this is set on a particular machine type the command "env". If the environmental variable *is not set* then when memory usage reaches this percentage, MEMORY, further adaptivity is prevented to avoid exceeding the memory specified at execution time. Caution is necessary since memory usage is checked after each adaptive step, and, if the memory usage increases by more than the residual percentage, 100-PERCENT, the calculation will terminate.

VARIABLE	DESCRIPTION
	If the memory environmental variable *is set* then when the number of words of memory allocated reaches or exceeds this value, MEMORY, further adaptivity is stopped.

Card 4	1	2	3	4	5	6	7	8
Variable						D3TRACE	IADPCF	
Type						I	I	
Default						0	0	

VARIABLE	DESCRIPTION
D3TRACE	Output flag: EQ.0: No additional output states EQ.1: A d3plot state will be output just before and after an adaptive step even though it may not be requested. You may want this output so that the LS-PrePost particle trace algorithm will work in the case of adaptivity.
IADPCF	Flag to enable adaptive user control files: EQ.0: No user control files EQ.1: Perform run-time control on 3D adaptivity through control files. See Remark 2.

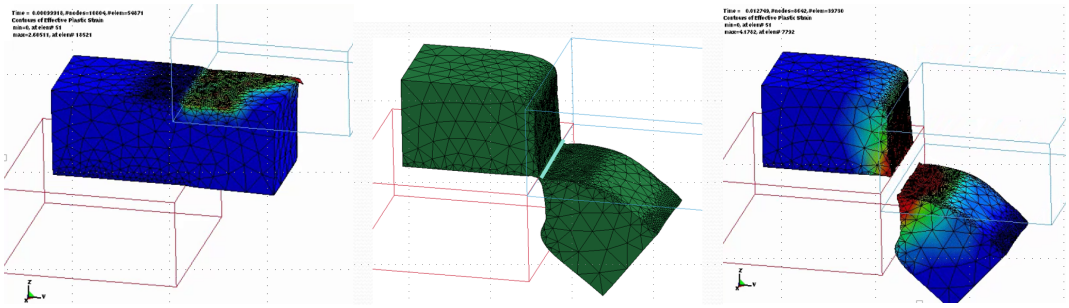
Remarks:

1. **Contact and ADPENE.** In three dimensions when ADPENE > 0, the solid part to be adapted is assumed to be on the slave side of a contact while the "tooling", consisting of a shell surface, is assumed to be on the master side of that same contact. ADPENE > 0 represents a distance from the tooling surface within which the adapted mesh refinement of the slave part is influenced by the radius of curvature of the tooling surface. This feature is currently *unavailable* for SOFT = 2 in *CONTACT.

2. Users perform run-time control on 3D adaptivity through control files in the following manners:
   1) Trigger additional adaptive step in run time
   2) Setup a special adaptive step with the option of manual remeshing by users
   3) For multiple adaptive parts, define remeshing parameters individually

   There are two control files: "adapt.fc1" and "adapt.fc2", which are automatically generated by running LS-DYNA if they are not pre-defined. The file "adapt.fc1" has three control parameters:
   a) C1 is the overall switch for this module
   b) C2 defines the time for the next additional adaptive step
   c) C3 is the switch for manual remeshing

   The detail can be found in the following three examples:
   a) 1, 0.0, 0: C1 = 1 to turn on run-time control; C2 = 0.0 to start additional adaptivity immediately; C3 = 0 to turn off manual remeshing
   b) 1, 0.01, 0: C2 = 0.01 to start additional adaptivity at the time 0.01
   c) 1, 0.01, 3: C3 = 3 to turn on manual remeshing of the adaptive part 3 in the additional adaptive step at time 0.01. In this case, LS-DYNA is paused and the mesh information of part 3 is output into a keyword file, named "user.mesh". Users can either manually edit "user.mesh" file or use third-party software to generate a new mesh and then save into the same file. LS-DYNA continues the analysis by users setting C1 = -1 in "adapt.fc1".

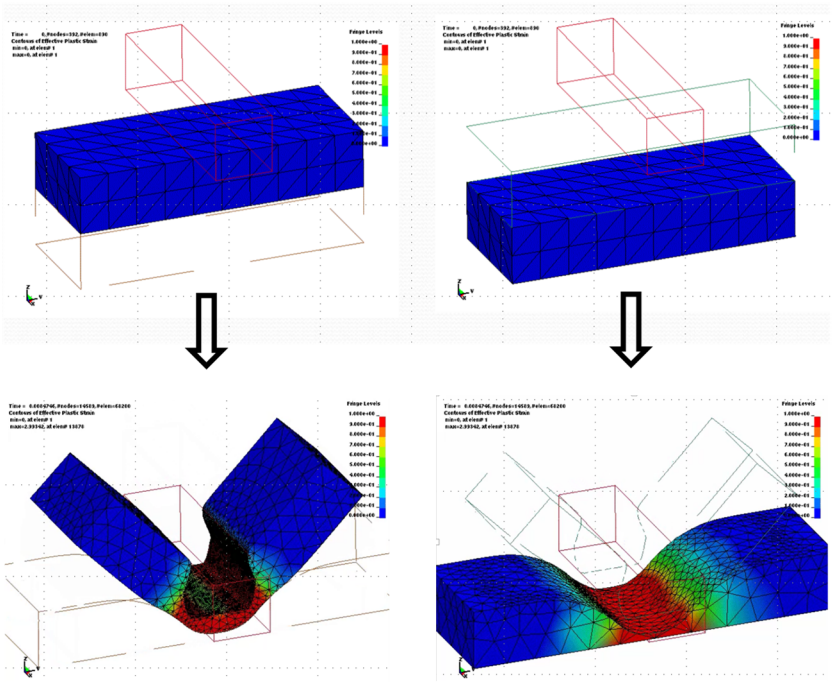


In the metal cutting simulation shown above, it can be estimated when the tool front meets the bottom die so that the additional adaptive step can be triggered for users to trim the mesh using third-party software and then continue the analysis in LS-DYNA.

In the control file "adapt.fc2", users define the birth and death time of adaptivity (same as TBIRTH and TDEATH in *CONTROL_ADAPTIVE) as well as the mesh size (same as RMIN and RMAX in *CONTROL_REMESHING) for each adaptive part. Here is an example:

2	Set the parameters for two adaptive parts
2, 0.0, 0.1, 1.0, 4.0	Part 2: TBIRTH = 0.0, TDEATH = 0.1, RMIN = 1.0, RMAX = 4.0

3, 0.01, 0.2, 2.0, 4.0 Part 3: TBIRTH = 0.01, TDEATH = 0.2, RMIN = 2.0, RMAX = 4.0



In the example shown above, there are adaptive parts: the top one contacts with a very sharp tool so that a smaller mesh size is needed to well represent the contact surface and better simulate the local large deformation; the bottom part has much smoother profile of material deformation so that the less frequent adaptivity and larger mesh size are used to reduce the overall computational cost. All of these differences in defining adaptivity for different parts can be made in adapt.fc2.

*CONTROL_REMESHING_{EFG}

Purpose: Provide control over the adaptive remeshing of solids. See also *CONTROL_-ADAPTIVE and the variable ADPOPT in *PART.

There are two different types of 3-D solid adaptivity affected by *CONTROL_REMESH-ING.

1. Tetrahedral adaptivity is invoked by ADPOPT = 2 in *PART. The initial mesh must be comprised of only tetrahedrons. FEM solid formulation 10, 13 and EFG solid formulation 42 (see details in *SECTION_SOLID) are recommended. When adaptivity is triggered by any of several available criteria (volume loss, minimum time step, etc.), the surface mesh consisting of triangles is first improved, subject to the element size criteria set forth in *CONTROL_REMESH-ING, and then the automatic remesher creates the tetrahedral elements of the new mesh starting from the improved surface mesh. When the EFG option is included, additional adaptivity criteria may be invoked using Card 2 and, optionally, Card 3.

2. Axisymmetric adaptivity, sometimes called orbital adaptivity, in which remeshing is done with hexahedral and pentahedral elements, is invoked by AD-POPT = 3 in *PART. Only the variables RMIN, RMAX, CID, and SEGANG are used in this type of adaptivitiy. Though EFG solid formulation 41 is permissible, the EFG option of *CONTROL_REMESHING does not apply for this type of adaptivity.

Card 1	1	2	3	4	5	6	7	8
Variable	RMIN	RMAX	VF_LOSS	MFRAC	DT_MIN	ICURV	CID	SEGANG
Type	F	F	F	F	F	I	I	F
Default	none	none	1.0	0.0	0.	4	0	0.0

Additional card for EFG option.

Card 2	1	2	3	4	5	6	7	8
Variable	IVT	IAT	IAAT		MM			
Type	I	I	I		I			
Default	1	0	0		0			

Second additional card for EFG option. This card is optional.

Card 3	1	2	3	4	5	6	7	8
Variable	IAT1	IAT2	IAT3					
Type	F	F	F					
Default	10^{20}	10^{20}	10^{20}					

VARIABLE	DESCRIPTION
RMIN	Minimum edge length for the surface mesh surrounding the parts which should be remeshed. See Remark 1.
RMAX	Maximum edge length for the surface mesh surrounding the parts which should be remeshed. See Remark 1.
VF_LOSS	Volume fraction loss required in a type 10/13 tetrahedral elements to trigger a remesh. For the type 13 solid elements, the pressures are computed at the nodal points; therefore, it is possible for overall volume to be conserved but for individual tetrahedrons to experience a significant volume loss or gain. The volume loss can lead to numerical problems. Recommended values for VF_LOSS in the range of 0.10 to 0.30 may be reasonable.
MFRAC	Mass ratio gain during mass scaling required for triggering a remesh. For a one percent increase in mass, set MFAC = 0.010. This variable applies to both to general three-dimensional tetrahedral remeshing and to three dimensional axisymmetric remeshing.

VARIABLE	DESCRIPTION
DT_MIN	Time step size required for triggering a remesh. This option applies only to general three-dimensional tetrahedral remeshing and is checked before mass scaling is applied and the time step size reset.
ICURV	Define number of elements along the radius in the adaptivity. See Remark 3.
CID	Coordinate system ID for three dimensional axisymmetric remeshing. The z-axis in the defined coordinate system is the orbital axis and must be parallel to the global z-axis in the current axisymmetric remesher. EQ.0: use global coordinate, and the global z-axis is the orbital axis (default)
SEGANG	For Axisymmetric 3-D remeshing: Angular element size (degrees) For General (TET) 3-D remeshing: Critical angle specified in radians to preserve feature lines
IVT	Internal variable transfer in adaptive EFG (see Remark 4): EQ.1: moving least square approximation with Kronecker-delta property (recommended in general case) EQ.-1: moving least square approximation without Kronecker-delta property EQ.2: partition of unity approximation with Kronecker-delta property EQ.-2: partition of unity approximation without Kronecker-delta property EQ.-3: finite element approximation
IAT	Flag for interactive adaptivity (see Remark 2): EQ.0: no interactive adaptivity EQ.1: interactive adaptivity combined with predefined adaptivity. EQ.2: purely interactive adaptivity. The time interval between two successive adaptive steps is bounded by ADPFREQ. EQ.3: purely interactive adaptivity

VARIABLE	DESCRIPTION
IAAT	Interactive adaptivity adjustable tolerance: EQ.0: the tolerance to trigger interactive adaptivity is not adjusted. EQ.1: the tolerance is adjusted in run-time to avoid over-activation.
MM	Interactive adaptive remeshing with monotonic resizing: EQ.1: the adaptive remeshing cannot coarsen a mesh. The current implementation only supports IAT = 1, 2, 3 and IER = 0.
IAT1	Shear strain tolerance for interactive adaptivity. If the shear strain in any formulation 42 EFG tetrahedral element exceeds IAT1, remeshing is triggered. (0.1 ~ 0.5 is recommended).
IAT2	L_{max}/L_{min} tolerance where L_{max} and L_{min} are the maximum and minimum edge lengths of any given formulation 42 EFG tetrahedral element. If this ratio in any element exceeds IAT2, remeshing is triggered. (RMAX/RMIN is recommended.)
IAT3	Volume change tolerance. If the normalized change in volume of any formulation 42 tetrahedral element, defined as $\|v_1 - v_0\|/\|v_0\|$ where v_1 is the current element volume and v_0 is the element volume immediately after the most recent remeshing, exceeds IAT3, remeshing is triggered. (0.5 is recommended.)

Remarks:

1. **Edge Length.** The value of RMIN and RMAX should be of the same order. The value of RMAX can be set to 2-5 times greater than RMIN. RMIN and RMAX may be superceded in specific regions of the part being adapted; see the variables PID, BRMIN, and BRMAX in *DEFINE_BOX_ADAPTIVE.

2. **Interactive Adaptivity.** Interactive adaptivity is only supported for the adaptive 4-noded mesh-free (EFG) solid formulation (ELFORM = 42 in *SECTION_SOLID_EFG). When interactive adaptivity is invoked (IAT > 0), even if none of the tolerances IAT1, IAT2, and IAT3 for the three respective indicators (shear strain, edge length ratio, normalized volume change) are exceeded, remeshing will still be triggered if any of the three indicators over a single explicit time step changes by more than 50%, that is, if

$$\frac{|[\text{value}]_n - [\text{value}]_{n-1}|}{|[\text{value}]_{n-1}|} > 0.5$$

where $[\text{value}]_n$ denotes value of indicator in n^{th} (current) time step and $[\text{value}]_{n-1}$ denotes value of indicator in previous time step. This condition is checked only if $[\text{value}]_{n-1}$ is nonzero.

3. **ICURV.** ICURV represents a number of elements and applies only when ADPENE > 0 in *CONTROL_ADAPTIVE. The "desired element size" at each point on slave contact surface is computed based on the tooling radius of curvature (see the description of ADPENE in *CONTROL_ADAPTIVE), so that ICURV elements would be used to resolve a hypothetical 90 degree arc at the tooling radius of curvature. The value of ICURV is (internally) limited to be >=2 and <=12. The final adapted element size is adjusted as necessary to fall within the size range set forth by RMIN and RMAX.

4. **IVT.** IVT defines different remapping schemes for the adaptive 4-noded meshfree (EFG) solid formulation (ELFORM = 42 in *SECTION_SOLID_EFG). The finite element approximation (IVT = -3) is the most efficient one in terms of computational cost. The moving least square approximation (IVT = 1 or -1) can significantly reduce the remapping error when there are many adaptive steps and the solution fields have local features, such as a high gradient. But this option has a high CPU and memory cost. The partition of unity approximation (IVT = 2 or -2) is slightly more accurate than the finite element approximation (IVT = -3).

*DEFINE_BOX_ADAPTIVE

Purpose: Define a box-shaped volume enclosing the solids where the tetrahedral *r*-adaptive mesh size is to be specified.

Card 1	1	2	3	4	5	6	7	8
Variable	BOXID	XMN	XMX	YMN	YMX	ZMN	ZMX	
Type	I	F	F	F	F	F	F	
Default	None	0.0	0.0	0.0	0.0	0.0	0.0	

Card 2	1	2	3	4	5	6	7	8
Variable	PID	LEVEL	LIDX/NDID	LIDY	LIDZ	BRMIN	BRMAX	
Type	I	I	I	I	I	F	F	
Default	0	1	0	0	0	0.0	0.0	

VARIABLE	DESCRIPTION
BOXID	Box ID. Define unique numbers.
XMN	Minimum *x*-coordinate. Define in the local coordinate system if the option LOCAL is active.
XMX	Maximum *x*-coordinate. Define in the local coordinate system if the option LOCAL is active.
YMN	Minimum *y*-coordinate. Define in the local coordinate system if the option LOCAL is active.
YMX	Maximum *y*-coordinate. Define in the local coordinate system if the option LOCAL is active.
ZMN	Minimum *z*-coordinate. Define in the local coordinate system if the option LOCAL is active.

VARIABLE	DESCRIPTION
ZMX	Maximum z-coordinate. Define in the local coordinate system if the option LOCAL is active.
PID	Deformable part ID. EQ.0: all active elements within box are considered.
LIDX/NDID	Load curve ID / Node ID (see Remark 1). GT.0: load curve ID. Define adaptive box movement (displacement as a function of time) in global X-axis. LT.0: absolute value is a node ID, whose translation will be followed by the moving adaptive box. The node ID can be on a moving rigid body. EQ.0: no movement
LIDY	Load curve ID (see Remark 1). GT.0: load curve ID. Define adaptive box movement (displacement as a function of time) in global Y-axis. EQ.0: no movement
LIDZ	Load curve ID (see Remark 1). GT.0: load curve ID. Define adaptive box movement (displacement as a function of time) in global Z-axis. EQ.0: no movement
BRMIN	Minimum mesh size in 3D tetrahedron adaptivity
BRMAX	Maximum mesh size in 3D tetrahedron adaptivity

Remarks:

1. **Moving Box.** Only when one of the LCIDX/NDID, LICDY, or LCIDZ is defined, the adaptive box will be moving; otherwise it will be stationary.

*SECTION_SOLID_{EFG}

Purpose: Define section properties for FEM/EFG solid elements.

For complete description of this keyword, please refer to the Keyword Manual Volume I.

The first card is required.

Card 1	1	2	3	4	5	6	7	8
Variable	SECID	ELFORM						
Type	I/A	I						

VARIABLE	DESCRIPTION
SECID	Section ID. SECID is referenced on the *PART card. A unique number or label must be specified.
ELFORM	Element formulation options. EQ.1: Constant stress solid element EQ.10: 1 point tetrahedron element EQ.13: 1 point nodal pressure tetrahedron element EQ.41: Mesh-free Galerkin (EFG) solid formulation (see Remark 5) EQ.42: Adaptive 4-noded mesh-free (EFG) solid formulation (see Remark 5)

Additional card for the EFG option

Card 2	1	2	3	4	5	6	7	8
Variable	DX	DY	DZ	ISPLINE	IDILA	IEBT	IDIM	TOLDEF
Type	F	F	F	I	I	I	I	F
Default	1.01	1.01	1.01	0	0	1	2	0.01

VARIABLE	DESCRIPTION
DX, DY, DZ	Normalized dilation parameters of the kernel function in x, y and z directions. The normalized dilation parameters of the kernel function are introduced to provide the smoothness and compact support properties on the construction of the mesh-free shape functions. Values between 1.0 and 1.5 are recommended. Values smaller than 1.0 are not allowed. Larger values will increase the computation time and will sometimes result in a divergence problem.
ISPLINE	Replace the choice for the EFG kernel functions definition in *CONTROL_EFG which allows for different spline functions in different sections. EQ.0: Cubic spline function (default) EQ.1: Quadratic spline function EQ.2: Cubic spline function with circular shape
IDILA	Replace the choice for the normalized dilation parameter definition in *CONTROL_EFG. This allows users to define different IDILA in different sections. EQ.0: Maximum distance based on the background elements. EQ.1: Maximum distance based on surrounding nodes.

VARIABLE	DESCRIPTION
IEBT	Essential boundary condition treatment (see Remarks 1 and 2): EQ.1: Full transformation method (default) EQ.-1: (W/o transformation) EQ.2: Mixed transformation method EQ.3: Coupled FEM/EFG method EQ.4: Fast transformation method EQ.-4: (W/o transformation) EQ.5: Fluid particle method for E.O.S and *MAT_ELASTIC_FLUID materials, currently supports only 4-noded background elements. EQ.7: Maximum entropy approximation
IDIM	Domain integration method (see Remark 3): EQ.1: Local boundary integration EQ.2: Two-point Gauss integration (default) EQ.3: Improved Gauss integration for IEBT = 4 or -4 EQ.-1: Stabilized EFG integration method (apply to 6-noded cell, 8-noded cell or combination of these two) EQ.-2: EFG fracture method (apply to 4-noded cell and SMP only). The EFG fracture method is obsolete – see Remark 6.
TOLDEF	Deformation tolerance for the activation of adaptive EFG Semi-Lagrangian and Eulerian kernel. See Remark 4. EQ.0.0: Lagrangian kernel GT.0.0: Semi-Lagrangian kernel LT.0.0: Eulerian kernel

Additional card for the EFG option. This card is optional.

Card 3	1	2	3	4	5	6	7	8
Variable	IPS	STIME	IKEN	SF	CMID	IBR	DS	ECUT
Type	I	F	I	I	I	I	F	F
Default	0	10^{20}	0	0.0	0	1	1.01	0.1

VARIABLE	DESCRIPTION
IPS	Pressure smoothing flag: EQ.0: No pressure smoothing (default) EQ.1: Moving-least squared pressure recovery (material response is slightly more compressible when turning on this function). Only the adaptive 4-noded mesh-free (EFG) solid formulation (ELFORM = 42) is supported.
STIME	Time to switch from stabilized EFG to standard EFG formulation. Obsolete - see Remark 6.
IKEN	EQ.0: Moving-least-square approximation (default, recommended) EQ.1: Maximum entropy approximation Obsolete - see Remark 6.
SF	Failure strain, recommended as an extra condition for the crack initiation under slow loading besides the stress-based cohesive law. Obsolete - see Remark 6.
CMID	Cohesive material ID for EFG fracture analysis (only Mode I crack is considered and only *MAT_COHESIVE_TH is available). Obsolete - see Remark 6.
IBR	EQ.1: No branching allowed. EQ.2: Branching is allowed. Obsolete - see Remark 6.
DS	Normalized support defined for computing the displacement jump in fracture analysis. Obsolete - see Remark 6.

VARIABLE	DESCRIPTION
ECUT	Define the minimum distance to the node that a crack surface can cut to the edge. Obsolete - see Remark 6.

Remarks:

1. **Automatic Sorting for EFG Background Mesh.** The current EFG formulation performs automatic sorting for finite element tetrahedral, pentahedron, and hexahedral elements as the background mesh to identify the mesh-free geometry and provide the contact surface definition in the computation.

2. **Essential Boundary Conditions.** The mixed transformation method, the coupled FEM/EFG method, and the fast transformation method were implemented in EFG 3D solid formulation. These three features were added to improve the efficiency on the imposition of essential boundary conditions and the transfer of real nodal values and generalized nodal values. The mixed transformation method is equivalent to the full transformation method with improved efficiency. The behavior of the coupled FEM/EFG method is between FEM and EFG. The fast transformation method provides the most efficient and robust results.

3. **IDIM.** For compressible material like foam and soil, IDIM = 1 is recommended. For nearly incompressible material like metal and rubber, IDIM = 2 (default) is recommended.

4. **TOLDEF.** This field is introduced to improve the negative volume problem usually seen during large deformation analysis. For the same analysis, the larger the magnitude of TOLDEF, the earlier the Semi-Lagrangian or Eulerian kernel is introduced into the EFG computation, so more CPU time is expected. A value between 0.0 and 0.1 is suggested for crashworthiness analysis. Semi-Lagrangian kernel is suggested for the solid materials while Eulerian kernel is suggested for the fluid and EOS materials.

5. **EFG Solid Elements: Types 41 and 42.** EFG element type 41 supports 4-node, 6-node and 8-node solid elements. For three-dimensional tetrahedron *r*-adaptive analysis (ADPTYP = 7 in *CONTROL_ADAPTIVE and ADPOPT = 2 in *PART), the initial mesh should be comprised of tetrahedrons and element type 42 should be used. Element type 42 only supports 4-node tetrahedron mesh and is optimized to achieve better computational efficiency compared to 41.

6. **EFG Fracture.** The EFG fracture method (IDIM = -2) is obsolete. For ductile failure, consider using *SECTION_SOLID_SPG.

ISPG

Solder joints are used to provide both mechanical and electrical connection in electronic packaging for IC industry. The solder geometry such as standoff height, contact angle, and so on, has the significant influence on the quality of solder joints such as their fatigue life. Thus, the prediction of shape evolution of solder joints in the reflow process is very important in electronic applications. In the past, analytical approaches, such as the truncated sphere method, the force-balanced solution, and the energy-based method, were usually used to predict the final shapes of the solder joints. However, the applications of these approaches have following major limitations: 1) their simulations of the shape evolution of the solder joints have no physical meanings; 2) they cannot simulate the reflowing with complex boundaries. Under this situation, it will be too difficult for those analytical approaches to simulate such reflow processes. Thus, there is a great demand from IC industry to develop a physics-based numerical method to predict the shape evolution of solder joints during the electronic reflow process. The Incompressible Smoothed Particle Galerkin (ISPG) method can meet the need.

LS-DYNA ISPG solver is a new generation CFD solver. Since ISPG is Lagrangian-based, the positions of the fluid material are automatically tracked and there is no need to use VOF or Level-set methods like in traditional Eulerian-based CFD solvers. The ISPG solver can have more accurate and efficient capturing of the free surfaces of fluids and interfaces with solids than traditional CFD software. It also guarantees very accurate conservation of fluid volume, even with very large deformation of fluids.

The ISPG solver includes following keywords:

*DEFINE_FP_TO_SURFACE_COUPLING

*MAT_IFPD

*SECTION_FPD

*DEFINE_FP_TO_SURFACE_COUPLING

Purpose: To define tied coupling interface between the fluid particles and a surface defined by the segments set. This coupling is used for ISPG simulation.

Card 1	1	2	3	4	5	6	7	8
Variable	SLAVE	MASTER	STYPE	MTYPE				
Type	I	I	I	I				
Default	None	None	1	0				

VARIABLE	DESCRIPTION
SLAVE	Part set ID defined in the coupling on the slave side.
MASTER	Segments set ID defined in the coupling on the master side. Currently the segments set should be generated from the 8-noded hexahedron elements.
STYPE	Currently only STYPE = 1 is used, which means that the part ID should be defined via SLAVE.
MTYPE	Currently only MTYPE = 0 is used, which means that the segments set ID should be define via MASTER

Card 2	1	2	3	4	5	6	7	8
Variable	SBC	SCA				SFPN		
Type	I	I				F		
Default	None	None				0.2		

VARIABLE	DESCRIPTION
SBC	Type of boundary condition. EQ.0: free-slip boundary EQ.1: non-slip boundary
SCA	Static contact angle in radian.
SFPN	stiffness coefficient along the normal direction of the contact interface. SFP < 1.0. Too small SFPN can cause large penetration.

*MAT_IFPD

This material is used to define the fluid particles' property in the incompressible free surface flow simulation with ISPG.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	RO	DYNVIS	SURF				
Type	I/A	F	F	F				

VARIABLE	DESCRIPTION
MID	Material identification. MID is referenced on the *PART card. A unique number or label must be specified.
RO	Fluid density.
DYNVIS	Dynamic viscosity of the fluid.
SURF	Surface tension coefficient of the fluid.

*SECTION_FPD

Purpose: Define section properties for incompressible smoothed particle Galerkin (ISPG) element.

Card 1	1	2	3	4	5	6	7	8
Variable	SECID	ELFORM						
Type	I/A	I						

VARIABLE	DESCRIPTION
SECID	Section ID. SECID is referenced on the *PART card. A unique number or label must be specified.
ELFORM	Element formulation options. EQ.49: Incompressible smoothed particle Galerkin formulation

Card 2	1	2	3	4	5	6	7	8
Variable	DX	DY	DZ					
Type	F	F	F					
Default	1.5	1.5	1.5					

VARIABLE	DESCRIPTION
DX, DY, DZ	Normalized dilation parameters of the kernel function in x, y and z directions. The normalized dilation parameters of the kernel function are introduced to provide the smoothness and compact support properties on the construction of the mesh-free shape functions. Values between 1.5 and 1.8 are recommended. The nodal support size of particles will be automatically adjusted with the material's deformation, but it is not allowed to be decreased.

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

VARIABLE	DESCRIPTION
MCVISC	Relaxation parameter to control the numerical dissipation in the momentum-consistently smoothing algorithm. Range is between 0.0 and 1.0. Lager value of MCVISC will give more numerical dissipation. According to the momentum consistently smoothing algorithm (see https://doi.org/10.1002/nme.6396), the unsmoothed value of particle velocities is updated for the computation in the next time step. In present fluid formulation, the unsmoothed particle velocity is updated using the following equation

$$\hat{v}_J^{n+1} = (1-\alpha)\left(\hat{v}_J^n + \sum_{I\in Z_I} \Psi_I^a(\boldsymbol{x}_J)(v_I^{n+1} - v_I^n)\right) + \alpha \sum_{I\in Z_I} \Psi_I^a(\boldsymbol{x}_J) v_I^{n+1}$$

Here α is the relaxation factor to control the numerical dissipation given by

$$\Delta E(\alpha) = -\alpha(2-\alpha)\frac{1}{2}\sum_{I\in Z_I} \hat{m}_I \left(\hat{v}_I - \sum_{I\in Z_I} \Psi_I^a(\boldsymbol{x}_J) v_J\right) \tag{42}$$

By varying α from 0 to 1, one can change the amount of dissipation introduced in the system.

VARIABLE	DESCRIPTION

SPH

Smooth particles hydrodynamics is a meshfree, Lagrangian particle method for modeling fluid flows and solid bodies. The method was developed to avoid the limitations of mesh tangling encountered in extreme deformation problems with the finite element method and to model the complex free surface and material interface behavior, including break-up into fragments naturally. The main difference between classical methods and SPH is the absence of a grid. Therefore, the particles are the computational framework on which the governing equations are resolved. It has been applied extensively to the incompressible flows, heat conduction, high explosive problems, bird strike, high velocity impact, forging and extrusion, fluid-structure interaction and so on.

The SPH solver in LS-DYNA is coupled with Lagrangian Solid elements and with thermal solver through different methods for solving all sorts of engineering problems. Multiple coupling options (interaction methods) between different SPH parts, and between SPH particles and other particle methods are developed to adapt to the different engineering application situations.

The SPH particles include the following major input keywords:

*CONTROL_SPH

*ELEMENT_SPH

*SECTION_SPH

The following keywords are used to apply the boundary conditions, initial conditions and loading conditions for the SPH particles:

*BOUNDARY_SPH_FLOW

*BOUNDARY_SPH_SYMMETRY_PLANE

*DEFINE_BOX_SPH

*DEFINE_SPH_INJECTION

*INITIAL_STRESS_SPH

For the coupling between SPH particles and solid elements, two keywords can be used:

*DEFINE_ADAPTIVE_SOLID_TO_SPH

*DEFINE_SPH_TO_SPH_COUPLING

*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.

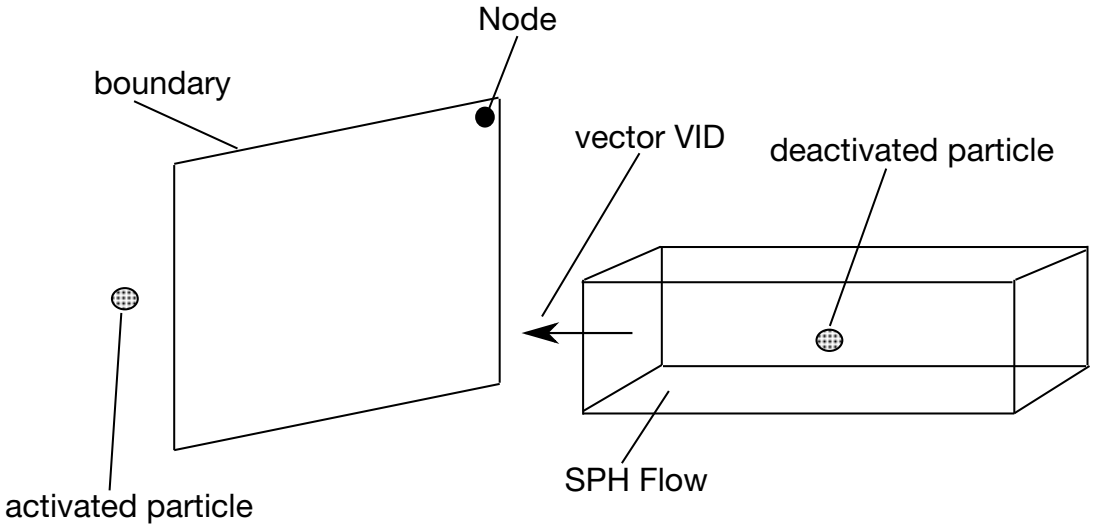


Figure 2-1. Vector VID determines the orientation of the SPH flow

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 DOF value of 4; 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_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.

*CONTROL_SPH

Purpose: Provide controls related to SPH (Smooth Particle Hydrodynamics).

Card 1	1	2	3	4	5	6	7	8
Variable	NCBS	BOXID	DT	IDIM	NMNEIGH	FORM	START	MAXV
Type	I	I	F	I	I	I	F	F
Default	1	0	10^{20}	none	150	0	0.0	10^{15}

Remaining cards are optional.†

Card 2	1	2	3	4	5	6	7	8
Variable	CONT	DERIV	INI	ISHOW	IEROD	ICONT	IAVIS	ISYMP
Type	I	I	I	I	I	I	I	I
Default	0	0	0	0	0	0	0	100

Card 3	1	2	3	4	5	6	7	8
Variable	ITHK	ISTAB	QL		SPHSORT	ISHIFT		
Type	I	I	F		I	I		
Default	0	0	0.01		0	0		

VARIABLE	DESCRIPTION
NCBS	Number of time steps between particle sorting.
BOXID	SPH approximations are computed inside a specified BOX. When a particle has gone outside the BOX, it is deactivated. This will save computational time by eliminating particles that no longer interact with the structure.

VARIABLE	DESCRIPTION
DT	Death time. Determines when the SPH calculations are stopped.
IDIM	Space dimension for SPH particles: EQ.3: for 3D problems EQ.2: for 2D plane strain problems EQ.-2: for 2D axisymmetric problems (see Remark 2)
NMNEIGH	Defines the initial number of neighbors per particle (see Remark 1 below).
FORM	Particle approximation theory (Remark 2): EQ.0: default formulation EQ.1: renormalization approximation EQ.2: symmetric formulation EQ.3: symmetric renormalized approximation EQ.4: tensor formulation EQ.5: fluid particle approximation EQ.6: fluid particle with renormalization approximation EQ.7: total Lagrangian formulation EQ.8: total Lagrangian formulation with renormalization EQ.9: adaptive SPH formulation (ASPH) with anisotropic smoothing tensor (Remark 2g) EQ.10: renormalization approximation for adaptive SPH formulation (ASPH) with anisotropic smoothing tensor EQ.12: moving least-squares based formulation (MPP only, see Remark 2e) EQ.15: enhanced fluid formulation EQ.16: enhanced fluid formulation with renormalization
START	Start time for particle approximation. Particle approximations will be computed when time of the analysis has reached the value defined in START.
MAXV	Maximum value for velocity for the SPH particles. Particles with a velocity greater than MAXV are deactivated. A negative MAXV will turn off the velocity checking.

VARIABLE	DESCRIPTION
CONT	Flag for inter-part particle interaction by "particle approximation": EQ.0: particle approximation is used for computing inter-part particle interaction for all SPH parts (default). EQ.1: particle approximation is not used for inter-part particle interaction, except as specified by the INTERACTION option of *SECTION_SPH. As an alternative to particle approximation, inter-part particle interaction can be accomplished using *DEFINE_SPH_TO_SPH_COUPLING.
DERIV	Time integration type for the smoothing length: EQ.0: $\frac{d}{dt}[h(t)] = \frac{1}{d}h(t)\nabla \cdot \mathbf{v}$, (default) EQ.1: $\frac{d}{dt}[h(t)] = \frac{1}{d}h(t)(\nabla \cdot \mathbf{v})^{1/3}$
INI	Computation of the smoothing length during the initialization: EQ.0: bucket sort based algorithm (default, very fast) EQ.1: global computation on all the particles of the model EQ.2: based on the mass of the SPH particle
ISHOW	Display option for deactivated SPH particles: EQ.0: no distinction in active SPH particles and deactivated SPH particles when viewing in LS-PrePost EQ.1: Deactivated SPH particles are displayed only as points and active SPH particles are displayed as spheres when Setting → SPH → Style is set to "smooth" in LS-PrePost.
IEROD	Deactivation control for SPH particles: EQ.0: particles remain active. See Remark 3. EQ.1: SPH particles are partially deactivated and stress states are set to 0 when erosion criteria are satisfied. See Remark 3. EQ.2: SPH particles are totally deactivated and stress states are set to 0 when erosion criteria are satisfied. See Remark 3.

VARIABLE	DESCRIPTION
	EQ.3: SPH particles are partially deactivated and stress states are set to 0 when erosion criteria are satisfied. If an EOS is defined, the volumetric response is unaffected. See Remark 3.
ICONT	Controls contact behavior for deactivated SPH particles: EQ.0: any contact defined for SPH remains active for deactivated particles. EQ.1: contact is inactive for deactivated particles.
IAVIS	Defines artificial viscosity formulation for SPH elements (Remark 4): EQ.0: Monaghan type artificial viscosity formulation is used. EQ.1: standard type artificial viscosity formulation from solid element is used (this option is not supported in SPH 2D and 2D axisymmetric elements).
ISYMP	Defines the percentage of original SPH particles used for memory allocation of SPH symmetric planes ghost nodes generation process (default is 100%). Recommended for large SPH particles models (value range 10~20) to control the memory allocation for SPH ghost particles with *BOUNDARY_SPH_SYMMETRY_PLANE keyword.
ITHK	Contact thickness option: EQ.0: the contact thickness is set to zero (default). EQ.1: the contact thickness is automatically calculated based on the volume of each SPH particle. This contact thickness calculation is ignored if a non-zero contact thickness for slave surface (SST) is provided by the contact card.
ISTAB	Stabilization type, only used when IFORM = 12: EQ.0: incremental stabilization (default). Adequate for most materials. EQ.1: total stabilization. Only recommended for hyperelastic materials.
QL	Quasi-Linear coefficient, only used when IFORM = 12. See Remark 5.

VARIABLE	DESCRIPTION
SPHSORT	For the implicit solver, sort and move SPH nodes from *NODE list to the end of the list. EQ.0: no sorting (default) EQ.1: perform sorting.
ISHIFT	Shifting algorithm option (available from R13.0): EQ.0: no shifting algorithm applied to SPH particles (default) EQ.1: shifting algorithm applied to SPH particles: particles are advanced, shifted slightly across streamlines in a more isotropic manner, reducing particle clustering in the maximum compression and stretching directions

Remark:

1. **NMNEIGH.** NMNEIGH is used to determine the initial memory allocation for the SPH arrays. Its value can be positive or negative. If NMNEIGH is positive, memory allocation is dynamic such that the number of neighboring particles is initially equal to NMNEIGH but that number is subsequently allowed to exceed NMNEIGH as the solution progresses. If NMNEIGH is negative, memory allocation is static and |NMNEIGH| is the maximum allowed number of neighboring particles for each particle throughout the entire solution. Using this static memory option can avoid memory allocation problems.

2. **FORM.** The user must be careful to pick the right FORM which depends upon the application. Below are some guidelines for selecting the FORM value:

   a) For most solid structure applications, FORM = 1 is recommended for more accurate results around the boundary area.

   b) For fluid or fluid-like material applications, FORM = 15 or 16 is recommended. FORM = 16 usually has better accuracy but requires more CPU time. Also note that formulations 15 and 16 include a smoothing of the pressure field and are, therefore, not recommended for materials with failure or problems with important strain localization.

   c) All SPH formulations with Eulerian kernel, that is, FORM with values 0 to 6, 15 and 16, can be used for large or extremely large deformation applications but will have tensile instability issues. FORM = 2 or 3 is not recommended for any case.

d) All SPH formulations with Lagrangian kernel (FORM = 7 or 8) can be used to avoid tensile instability issue but they cannot endure very large deformations.

e) For improved accuracy and tensile stability, a formulation based on moving least-squares (FORM = 12) is available. This formulation can be used for extremely large deformation applications but entails a significant computational cost. FORM = 12 is available for MPP simulations only. It is strongly recommended to keep a constant smoothing length for this formulation by setting HMIN = 1.0 and HMAX = 1.0 in *SECTION_SPH.

f) Only formulations 0, 1, 15 and 16 are implemented for 2D axisymmetric problems (IDIM = -2).

g) Formulations 9 and 10 are adaptive smoothed particle hydrodynamics formulations with an anisotropic kernel (Eulerian kernel) whose axes evolve automatically to follow the mean particles spacing as it varies in time, space and direction based on the strain rate tensors. These forms must be used with the *SECTION_SPH_ELLIPSE keyword. They have better accuracy and stability than the standard SPH. These forms can be used for extremely large deformation problems and are available for 3D case only.

3. **Erosion.** The erosion criteria, which triggers particle deactivation when IEROD = 1, 2 or 3, may come from the material model, from *MAT_ADD_EROSION, or from the ERODE field in *CONTROL_TIMESTEP.

a) For IEROD = 0, SPH particles remain active. This option is generally not recommended when materials with erosion are used, as many material models will still reset the stress field to zero periodically. If an unaltered stress field is desired, simply remove the erosion criteria in the material model parameters.

b) For IEROD = 1, SPH particles are partially deactivated; that is, the stress states of the deactivated SPH particles will be set to zero, but these particles still remain in the domain integration for more stable results.

c) For IEROD = 2, SPH particles are totally deactivated, so the stress states will be set to 0 and the deactivated particles do not remain in the domain integration.

d) For IEROD = 3, SPH particles remain active. The deviatoric stress is set to zero. If an equation of state is used, the volumetric response remains unaltered; otherwise the volumetric stress is set to zero as well.

e) Deactivated particles can be distinguished from active particles by setting ISHOW = 1.

f) To disable contact for deactivated particles, set ICONT = 1.

4. **Artificial Viscosity.** The artificial viscosity for standard solid elements, which is active when AVIS = 1, is given by:

$$q = \rho l\left(Q_1 l\dot{\varepsilon}_{kk}^2 - Q_2 a\dot{\varepsilon}_{kk}\right) \quad \dot{\varepsilon}_{kk} < 0$$
$$q = 0 \quad \dot{\varepsilon}_{kk} \geq 0$$

where Q_1 and Q_2 are dimensionless input constants, which default to 1.5 and .06, respectively (see *CONTROL_BULK_VISCOSITY); l is a characteristic length given as the square root of the area in two dimensions and as the cube root of the volume in three; and a is the local sound speed. This formulation, which is consistent with solid artificial viscosity, has better energy balance for SPH elements.

For general applications, Monaghan type artificial viscosity is recommended since this type of artificial viscosity is specifically designed for SPH particles. The Monaghan type artificial viscosity, which is active when AVIS = 0, is defined as follows:

$$q = \begin{cases} \dfrac{-Q_2\bar{c}_{ij}\phi_{ij} + Q_1\phi_{ij}^2}{\bar{\rho}_{ij}} & v_{ij}x_{ij} < 0 \\ 0 & v_{ij}x_{ij} \geq 0 \end{cases}$$

where

$$\phi_{ij} = \frac{h_{ij}v_{ij}x_{ij}}{|x_{ij}|^2 + \varphi^2}$$
$$\bar{c}_{ij} = 0.5(c_i + c_j)$$
$$\bar{\rho}_{ij} = 0.5(\rho_i + \rho_j)$$
$$h_{ij} = 0.5(h_i + h_j)$$
$$\varphi = 0.1h_{ij}$$

and Q_1/Q_2 are input constants. When using Monaghan type artificial viscosity,

it is recommended that the user set both Q_1 and Q_2 to 1.0 on either the *CONTROL_BULK_VISCOSITY or *HOURGLASS keywords; see for example G. R. Liu.

5. **Quasi-Linear Coefficient.** The moving least-squares based formulation contains a quasi-linear approximation term to combine accuracy with stability in extremely large deformations simulations. The default value QL = 0.01 gives a good compromise between accuracy and stability in most cases. For greater accuracy, its value can be reduced to QL = 0.001 in simulations with small

deformations, or increased to QL = 0.1 for extreme deformations, if instabilities are present.

*DEFINE_ADAPTIVE_SOLID_TO_SPH_{OPTION}

Purpose: Create SPH particles to either replace or supplement solid Lagrangian elements.

Applications of this feature include adaptively transforming a Lagrangian solid Part or Part Set to SPH particles, when the Lagrangian solid elements comprising those parts fail. One or more SPH particles (elements) will be generated for each failed element. The SPH particles replacing the failed solid Lagrangian elements inherit all the Lagrange nodal quantities and all the Lagrange integration point quantities of these failed solid elements. Those properties are assigned to the newly activated SPH particles. The constitutive properties assigned to the new SPH part will correspond to the MID and EOSID referenced by the SPH *PART definition. This keyword with options of ICPL = 0, 1 has been extended to 2D cases too.

The available options include:

<BLANK>

ID

ID Card. Additional card for the ID keyword option.

ID	1	2	3	4	5	6	7	8
Variable	DID	HEADING						
Type	I	A70						
Default	none	none						

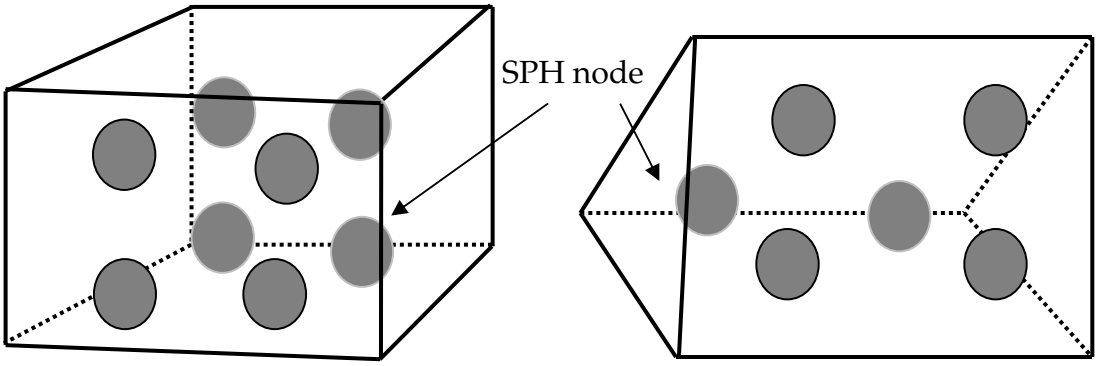
Card 1	1	2	3	4	5	6	7	8
Variable	IPID	ITYPE	NQ	IPSPH	ISSPH	ICPL	IOPT	CPCD
Type	I	I	I	I	I	I	I	F
Default	none	none	none	none	none	none	none	average

VARIABLE	DESCRIPTION
DID	Definition ID. This must be a unique number.

VARIABLE	DESCRIPTION
HEADING	Definition descriptor. It is suggested that unique descriptions be used.
IPID	ID of the solid part or part set to transform.
ITYPE	IPID type: EQ.0: Part ID, NE.0: Part set ID.
NQ	Adaptive option for hexahedral elements. For tetrahedral and pentahedral elements, see Remark 1: EQ.n: Adapt one 8-node solid element to $(n \times n \times n)$ SPH elements. The range of n is from 1 to 6.
IPSPH	Part ID for newly generated SPH elements. See Remark 2.
ISSPH	Section ID for SPH elements. See Remark 2.
ICPL	Coupling of newly generated SPH elements to the adjacent solid elements: EQ.0: Failure without coupling (debris simulation), EQ.1: Coupled to solid element, EQ.3: Provide only thermal coupling between SPH part and solid part (must be combined with IOPT = 0 option; see Remark 4).
IOPT	Coupling method: EQ.0: Coupling from beginning (used as constraint between SPH elements and solid elements), EQ.1: Coupling begins when Lagrangian solid element fails. See Remark 3.
CPCD	Thermal coupling conductivity between SPH part and solid part for ICPL = 3 option. The default value is set as the average value of the conductivity from SPH part and the conductivity from solid part.

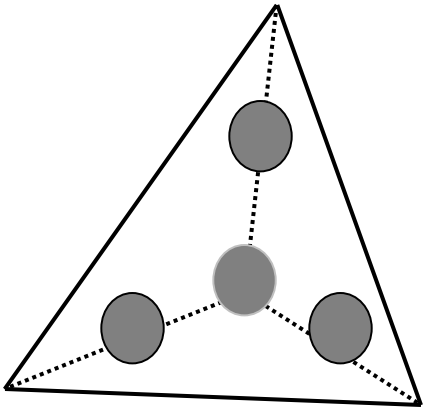
Remarks:

1. **Number of SPH Particles per Element.** The SPH particles are evenly distributed within the solid element. For hexahedral elements the number of the



Example of SPH nodes for hexahedral element with NQ = 2

Example of SPH nodes for pentahedral element with NQ = 2



Example of SPH nodes for tetrahedral element with NQ = 2

Figure 2-2. Examples of distributions of SPH particles converted from solid Lagrangian elements.

generated SPH particles is NQ × NQ × NQ. For pentahedral elements, the number of generated SPH particles is 1, 6, and 18 for NQ equal to 1, 2, and 3, respectively. For tetrahedral elements, the number of generated SPH particles is 1, 4, and 10 for NQ equal to 1, 2, and 3, respectively. See Figure 2-2.

2. **SPH Part ID**. The Part ID for newly generated SPH particles can be either a new Part ID or the ID of an existing SPH Part. For constraint coupling, such as when ICPL = 1 and IOPT = 0, the newly generated SPH part ID should be different from the existing one.

3. **Mechanical Failure Coupling.** ICPL = 0 is used for debris simulation, so no coupling happens between newly generated SPH particles and solid elements;

the user, therefore, needs to define node to surface contact for the interaction between those two parts. When ICPL = 1 and IOPT = 1, the newly generated SPH particles are bonded with solid elements as one part through the coupling, and the new material ID with different failure criteria can be applied to the newly generated SPH particles.

4. **Thermal Coupling.** ICPL = 3, which must be combined with IOPT = 0, is used to thermally couple an SPH part and solid part(s). There is no structural coupling provided. A thermal conductivity value may be defined using the variable CPCD.

*DEFINE_BOX_SPH_{OPTION}

Available options include:

<BLANK>

LOCAL

Purpose: Define a box-shaped volume. Two diagonally opposite corner points of a box are specified in global coordinates. Particle approximations of SPH elements are computed when particles are located inside the box. SPH particles are deactivated as they leave the box; the deactivated SPH particles cannot be reactivated even if they reenter the box. The motion of the maximum and minimum coordinates of the box can be defined with a load curve.

Card 1	1	2	3	4	5	6	7	8
Variable	BOXID	XMN	XMX	YMN	YMX	ZMN	ZMX	VID
Type	I	F	F	F	F	F	F	I
Default	none	0.0	0.0	0.0	0.0	0.0	0.0	0

Card 2	1	2	3	4	5	6	7	8
Variable	LCID	VD	NID					
Type	I	I	I					
Default	0	0	0					

Local Card 1. First additional card for the LOCAL keyword option. See *DEFINE_BOX for a description of the LOCAL option

Card 3	1	2	3	4	5	6	7	8
Variable	XX	YX	ZX	XV	YV	ZV		
Type	F	F	F	F	F	F		
Default	0.0	0.0	0.0	0.0	0.0	0.0		

Local Card 2. Second additional card for the LOCAL keyword option.

Card 4	1	2	3	4	5	6	7	8
Variable	CX	CY	CZ					
Type	F	F	F					
Default	0.0	0.0	0.0					

VARIABLE	DESCRIPTION
BOXID	Box ID. Define unique numbers.
XMN	Minimum x-coordinate. Define in the local coordinate system if the option LOCAL is active.
XMX	Maximum x-coordinate. Define in the local coordinate system if the option LOCAL is active.
YMN	Minimum y-coordinate. Define in the local coordinate system if the option LOCAL is active.
YMX	Maximum y-coordinate. Define in the local coordinate system if the option LOCAL is active.
ZMN	Minimum z-coordinate. Define in the local coordinate system if the option LOCAL is active.
ZMX	Maximum z-coordinate. Define in the local coordinate system if the option LOCAL is active.

VARIABLE	DESCRIPTION
VID	Vector ID to describe direction of box motion when VD = 0 or 1; see *DEFINE_VECTOR.
LCID	Load curve ID to describe value of box motion as a function of time; see *DEFINE_CURVE
VD	Velocity/Displacement flag: EQ.0: velocity, EQ.1: displacement, EQ.2: referential node
NID	Referential nodal ID for VD = 2 (SPH box will move with this node).
XX	*X*-coordinate on local *x*-axis. Origin lies at (0,0,0). Define if the LOCAL option is active.
YX	*Y*-coordinate on local *x*-axis. Define if the LOCAL option is active.
ZX	*Z*-coordinate on local *x*-axis. Define if the LOCAL option is active.
XV	*X*-coordinate of local *xy*-vector. Define if the LOCAL option is active.
YV	*Y*-coordinate of local *xy*-vector. Define if the LOCAL option is active.
ZV	*Z*-coordinate of local *xy*-vector. Define if the LOCAL option is active.
CX	*X*-global coordinate of offset vector to origin of local system. Define if the LOCAL option is active.
CY	*Y*-global coordinate of offset vector to origin of local system. Define if the LOCAL option is active.
CZ	*Z*-global coordinate of offset vector to origin of local system. Define if the LOCAL option is active.

*DEFINE_SPH_INJECTION

Purpose: Inject SPH elements from user defined grid points.

Card 1	1	2	3	4	5	6	7	8
Variable	PID	NSID	CID	VX	VY	VZ	AREA	VMAG
Type	I	I	I	F	F	F	F	I
Default	none	none	none	0.0	0.0	0.0	0.0	0

Card 2	1	2	3	4	5	6	7	8
Variable	TBEG	TEND	NID					
Type	F	F	I					
Default	0.0	10^{20}	0					

VARIABLE	DESCRIPTION
PID	Part ID of newly generated SPH elements.
NSID	Node set ID. Nodes are used for initial injection position for the SPH elements.
CID	Local coordinate system ID; see *DEFINE_COORDINATE_SYSTEM for example. The local x and y-directions define the injection plane, while the local z-direction defines the normal to the injection plane. A CID is always required. This coordinate system is used for the definition of injection velocity (VX,VY,VZ) as well.
VX, VY, VZ	Velocity of the injected particles: $\mathbf{v}$ = (VX, VY, VZ), defined in the local coordinate system from CID.
AREA	The area of initial injection surface. The density of injection flow comes from the material models; see *MAT definition.
VMAG	Injected particle velocity multiplier:

VARIABLE	DESCRIPTION
	GT.0: The velocity of the injected particles is multiplied by VMAG. LT.0: |VMAG| is a curve ID defining the magnitude of the velocity vector with respect to time, for variable injection speed.
TBEG	Birth time
TEND	Death time
NID	An optional node ID. If defined, the center of the injection plane follows the motion of this node.

*DEFINE_SPH_TO_SPH_COUPLING

Purpose: Define a penalty-based, node-to-node contact for particles of SPH parts. Note that this contact type is an alternative to inter-part particle interaction by "particle approximation;" see the field CONT in *CONTROL_SPH and the INTERACTION option of *SECTION_SPH.

Card Sets: Each set consists of a Card 1 and may include an additional Card 2. Unless the card following Card 1 contains an "&" in its first column, the optional card is not read. Provide as many sets as necessary. This input terminates at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	SSID	MSID	SSTYP	MSTYP	IBOX1	IBOX2	PFACT	SRAD
Type	I	I	I	I	I	I	F	F
Default	none	none	none	none	none	none	1.0	1.0

Optional. The keyword reader identifies this card by an "&" in the first column.

Card 2	1	2	3	4	5	6	7	8
Variable	DFACT	ISOFT						
Type	F	I						
Default	0.0	0						

VARIABLE	DESCRIPTION
SSID	Slave part or part set ID.
MSID	Master part or part set ID.
SSTYP	Slave part type: EQ.0: Part set ID, EQ.1: Part ID.

MSTYP	Master part type: EQ.0: Part set ID, EQ.1: Part ID.
IBOX1	Box ID for slave parts; see Remark 1.
IBOX2	Box ID for master parts; see Remark 1.
PFACT	Penalty scale factor; see Remark 2.
SRAD	Scale factor for nodes to nodes contact criteria; see Remark 3.
DFACT	Penalty scale factor for contact damping coefficient; see Remark 4.
ISOFT	Soft constraint option: EQ.0: penalty formulation (default) EQ.1: soft constraint formulation The soft constraint may be necessary if the material constants of the parts in contact have a wide variation in the elastic bulk moduli. In the soft constraint option, the interface stiffness is based on the nodal mass and the global time step size.

Remarks:

1. **Contact Particles.** IBOX1 and IBOX2 are used to define the box IDs for the slave parts and the master parts respectively. Only the particles inside the boxes are defined for the node to node contacts.

2. **Impact Velocity.** For High Velocity Impact problems, a smaller value ranging from 0.01 to 10^{-4} for the PFACT field is recommended. A value ranging from 0.1 to 1 is recommended for low velocity contact between two SPH parts.

3. **Contact Detection.** Contact between two SPH particles from different parts is detected when the distance of two SPH particles I and J is less than SRAD $\times \left(h_I + h_J\right)/2.0$. If SRAD < 0, the contact distance is based on the nodal volumes instead of the smoothing lengths: SRAD $\times \left(\sqrt[3]{V_I} + \sqrt[3]{V_J}\right)/2.0$.

4. **DFACT.** The default value, DFACT = 0.0, is recommended. For DFACT > 0.0, interaction between SPH parts includes a viscous effect, providing some stickiness similar to the particle approximation invoked when CONT = 0 in *CONTROL_SPH. At present, no recommendation can be given for a value of DFACT other than the value should be less than 1.0.

.

*ELEMENT_SPH_{OPTION}

Available options include:

<BLANK>

VOLUME

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

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

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

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

Remarks:

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

axisymmetric SPH element has a mass of ρA, where ρ is its density and A is the area of the SPH element. A can be approximated by the area of its corresponding axisymmetric shell element (see Figure 2-3). The mass printout in the d3hsp file is the mass per radian, that is, $\rho A x_i$. See Figure 2-4.

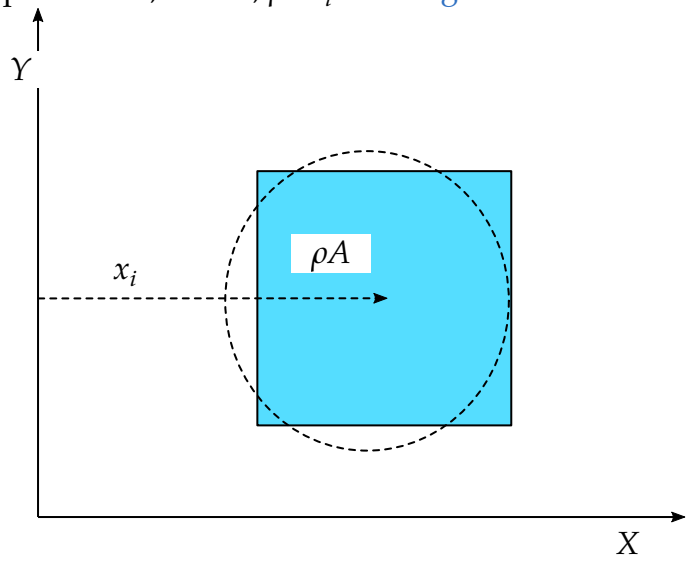


Figure 2-3. Schematic of axisymmetric SPH cross section

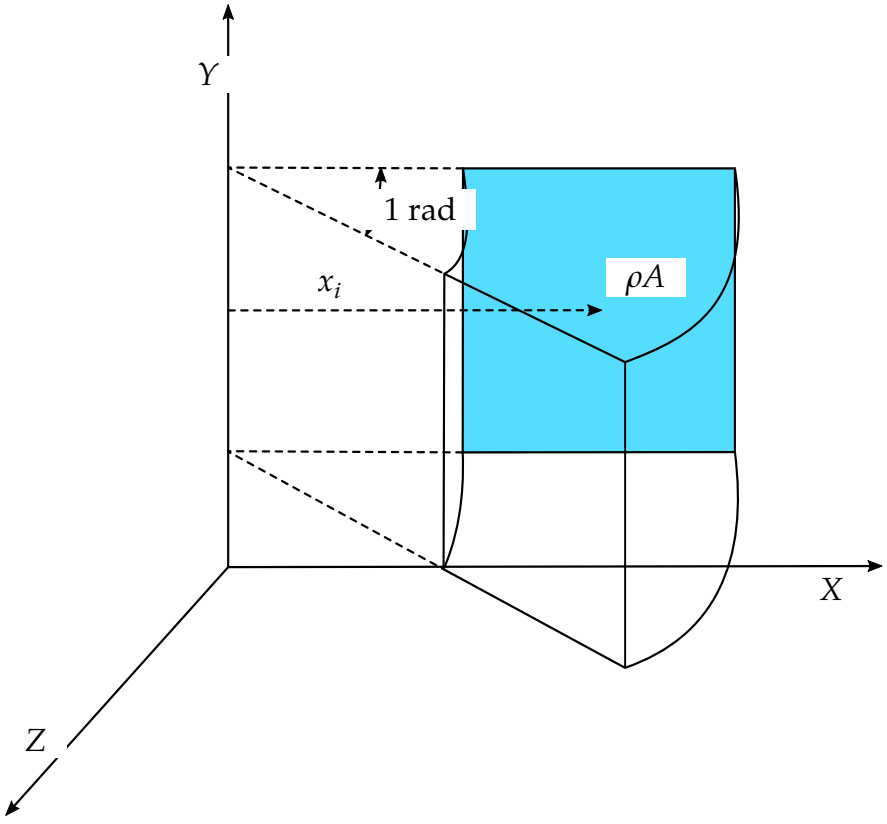


Figure 2-4. Mass printout in d3hsp

*INITIAL_STRESS_SPH

Purpose: Initialize stresses and plastic strains for SPH elements. This command is not applicable to hyperelastic materials or any material model based on a Total Lagrangian formulation. For *MAT_005, *MAT_014, and any material that requires an equation-of-state (*EOS), the initialized stresses are deviatoric stresses, not total stresses.

Element Cards. Define as many SPH elements in this section as desired. The input is assumed to terminate when a new keyword is detected.

Card 1	1	2	3	4	5	6	7	8
Variable	EID	SIGXX	SIGYY	SIGZZ	SIGXY	SIGYZ	SIGZX	EPS
Type	I	F	F	F	F	F	F	F
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
EID	SPH particle ID
SIG*ij*	Define the ij^{th} stress component. Stresses are defined in the GLOBAL Cartesian system.
EPS	Effective plastic strain.

***SECTION_SPH**_*{OPTION}*

Available options include:

<BLANK>

ELLIPSE

INTERACTION (See Remark 3)

USER (See Remark 2)

Purpose: Define section properties for SPH particles.

> **NOTE:** This feature is not supported for use in implicit calculations.

Card Sets. For each SPH section add one set of the following cards (depending on the keyword option). This input ends at the next keyword ("*") card.

Card 1	1	2	3	4	5	6	7	8
Variable	SECID	CSLH	HMIN	HMAX	SPHINI	DEATH	START	SPHKERN
Type	I/A	F	F	F	F	F	F	I
Default	none	1.2	0.2	2.0	0.0	10^{20}	0.0	0

Ellipse Card. Additional card for ELLIPSE keyword option.

Card 2	1	2	3	4	5	6	7	8
Variable	HXCSLH	HYCSLH	HZCSLH	HXINI	HYINI	HZINI		
Type	F	F	F	F	F	F		

VARIABLE	DESCRIPTION
SECID	Section ID. SECID is referenced on the *PART card. A unique number or label must be specified.

VARIABLE	DESCRIPTION
CSLH	Constant used to calculate the initial smoothing length of the particles. The default value works for most problems. Values between 1.05 and 1.3 are acceptable. Taking a value less than 1 is inadmissible. Values larger than 1.3 will increase the computational time. The default value is recommended. See Remark 1.
HMIN	Scale factor for the minimum smoothing length. Ignored if FORM = 12 in *CONTROL_SPH. See Remark 1.
HMAX	Scale factor for the maximum smoothing length. Ignored if FORM = 12 in *CONTROL_SPH. See Remark 1.
SPHINI	Optional initial smoothing length (overrides true smoothing length). With this option LS-DYNA will not calculate the smoothing length during initialization, and the field CSLH is ignored.
DEATH	Time imposed SPH approximation is stopped.
START	Time imposed SPH approximation is activated.
SPHKERN	Option for SPH kernel functions (smoothing functions): EQ.0: Cubic spline kernel function (default). EQ.1: Quintic spline kernel function: a higher order smoothing function with a larger support size. It is only available for the 3D case with FORM = 0, 1, 5, 6, 9 and 10 (see *CONTROL_SPH). EQ.2: Quadratic spline kernel function: it helps to relieve the problem of compressive instability and aims for HVI problems. It is only available for the 3D case with FORM = 0, 1, 5, and 6 (see *CONTROL_SPH). EQ.3: Quartic kernel function: this kernel function is very close to cubic spline kernel function but is more stable. It is only available for the 3D case with FORM = 0, 1, 5, and 6 (see *CONTROL_SPH).
HXCSLH	Constant applied for the smoothing length in the x-direction for the ellipse case.
HYCSLH	Constant applied for the smoothing length in the y-direction for the ellipse case.
HZCSLH	Constant applied for the smoothing length in the z-direction for the ellipse case.

VARIABLE	DESCRIPTION
HXINI	Optional initial smoothing length in the x-direction for the ellipse case (overrides true smoothing length)
HYINI	Optional initial smoothing length in the y-direction for the ellipse case (overrides true smoothing length)
HZINI	Optional initial smoothing length in the z-direction for the ellipse case (overrides true smoothing length)

Remarks:

1. **Smoothing Length.** The SPH processor in LS-DYNA employs a variable smoothing length. LS-DYNA computes the initial smoothing length, h_0, for each SPH part by taking the maximum of the minimum distance between every particle and then scaling this value by CSLH. The recommended values of CSLH should be used so that the radius of the support domain covers more than two layers of SPH particles along each direction. Every particle has its own smoothing length which varies in time according to the following equation:

$$\frac{d}{dt}h(t) = h(t)\nabla \cdot \mathbf{v} \ ,$$

where $h(t)$ is the smoothing length, and $\nabla \cdot \mathbf{v}$ is the divergence of the flow. The smoothing length increases as particles separate and reduces as the concentration increases. This scheme is designed to hold constant the number of particles in each neighborhood. In addition to being governed by the above evolution equation, the smoothing length is constrained to be between a user-defined upper and lower value,

$$\text{HMIN} \times h_0 < h(t) < \text{HMAX} \times h_0.$$

Defining a value of 1 for HMIN and 1 for HMAX will result in a constant smoothing length in time and space.

When formulation 12 is employed (FORM = 12 in *CONTROL_SPH), the smoothing length remains constant with HMIN and HMAX set to 1.0 internally.

2. **USER Option.** The USER option allows the definition of customized subroutine for the variation of the smoothing length. A subroutine called *hdot* is defined in the file dyn21.F (Unix/linux) or lsdyna.f (Windows).

3. **Inter-Part Particle Interaction.** There are two fundamental ways that particles from different SPH parts can interact with each other. One way is through "particle approximation" and the other way is through *DEFINE_SPH_TO_SPH_-

COUPLING. When CONT = 0 in *CONTROL_SPH, "particle approximation" is used for all SPH parts in treating inter-part particle interaction. When CONT = 1 in *CONTROL_SPH, inter-part particle interaction by "particle approximation" occurs only for those SPH parts that use *SECTION_SPH_INTERACTION, while any SPH part that does not make use of *SECTION_SPH_INTERACTION will not participate in inter-part particle interaction, *except* as defined using *DEFINE_SPH_TO_SPH_COUPLING.

Advanced Dynamic Failure Analysis

In LS-DYNA Advanced Dynamic Failure Analysis (ADFA) package, a set of advanced numerical methods adopt the strong discontinuity (element deletion-free) approaches for modern dynamics failure analyses from ductile rupture to brittle fracture and crack propagation across a wide range of materials and speeds.

The LS-DYNA ADFA solvers include Peridynamics, Smoothed Particle Galerkin (SPG) method, Smoothed Particle Hydrodynamics (SPH), and Extended Finite Element Method (XFEM).

The ADFA applications include but not limited to: simulate material removal processes in manufacturing, model damage and fracture in dynamic structure analysis, and predict crack propagation in low and high-speed events. A wide range of materials from metal, concrete, wood, foam to rubber and composite can be used in the ADFA simulations.

In this chapter, we present Peridynamics, SPG and XFEM methods. The SPH method is described in the previous chapter.

Peridynamics

The Peridynamics 3D solid solver is designed for the dynamic process of multiple cracks and fragmentation in brittle materials. It includes following keywords:

*SECTION_SOLID_PERI

*MAT_ELASTIC_PERI

The Peridynamics laminate solver is designed for capture in-plane and delamination failure modes of Unidirectional Fiber Reinforced Polymer (UD-FRP) laminate. It includes following keywords:

*SECTION_SOLID_PERI

*MAT_ELASTIC_PERI_LAMINATE

*SET_PERI_LAMINATE

*ELEMENT_SOLID_PERI

*ELEMENT_SOLID_PERI

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

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

VARIABLE	DESCRIPTION
EID	The element ID.
N1,N2,N3,N4	The nodal point 1 to nodal point 4.

Repeat Card 1 to list all elements.

Remarks:

1. To discretize a laminate, first, construct a layer of 3D surface mesh by 4-noded elements, such as shell elements according to the geometric shape of laminate. Then, copy this layer of mesh and move it to the middle surface of a lamina. Repeat this process until each lamina has one layer of mesh.
2. To represent in-plane material split, all the nodes of Peridynamics laminate elements must be detached, i.e., the total number of nodes = total number of Peridynamics laminate elements*4.
3. To improve efficiency, the Peridynamic laminate model mpp version requires that all laminae have the same decomposition. Using the boxes, defined as decomposition {region{box…}…} in pfile (Appendix O, Vol 2, Keyword user manual), can ensure the consistent decomposition. Here a box contains the same elements from each lamina and lump them into one CPU.

*MAT_ELASTIC_PERI

Purpose: Define material properties of elastic materials used in Peridynamics brittle solid parts.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	RO	E	GT	GS			
Type	I/A	I	F	F	F			
Default				1E20	1E20			

VARIABLE	DESCRIPTION
MID	Material identification. MID is referenced on the *PART card. A unique number or label must be specified.
RO	Material density.
E	Young's modulus.
GT	Fracture energy release rate.
GS	Fracture energy release rate for compression.

*MAT_ELASTIC_PERI_LAMINATE

Purpose: Define material properties of elastic materials used in Peridynamics laminate parts.

Card 1	1	2	3	4	5	6	7	8
Variable	MID	RO	E1	E2	v12	G12		
Type	I/A	F	F	F	F	F		
Default								

VARIABLE	DESCRIPTION
MID	Material identification. MID is referenced on the *PART card. A unique number or label must be specified.
RO	Material density.
E1	Young's modulus-longitudinal direction for one lamina. 1-direction.
E2	Young's modulus-transverse direction for one lamina. 2-direction.
v12	Poisson's ration in the lamina plane.
G12	Shear modulus on 1-2 directions.

Card 2	1	2	3	4	5	6	7	8
Variable	FOPT	FC1	FC2	FCC1	FCC2	FCD	FCDC	
Type	I	F	F	F	F	F	F	
Default								

VARIABLE	DESCRIPTION
FOPT	Failure criterion type for FC1, FC2, FCC1, FCC2, FCD, and FCDC: EQ.1: Energy release rate EQ.2: Failure stretch ratio for tension (recommended).
FC1	Tension failure criterion for longitudinal direction, 1-direction.
FC2	Tension failure criterion for transverse direction, 2-direction.
FCC1	Compression failure criterion for longitudinal direction, 1-direction.
FCC2	Compression failure criterion for transverse direction, 2-direction.
FCD	Tension delamination failure criterion.
FCDC	Compression delamination failure criterion.

Card 3	1	2	3	4	5	6	7	8
Variable	V1	V2	V3					
Type	F	F	F					
Default								

VARIABLE	DESCRIPTION
V1	Fiber direction, 1-direction, in global coordinate.
V2	Fiber direction, 2-direction, in global coordinate.
V3	Fiber direction, 3-direction, in global coordinate.

*SECTION_SOLID_PERI

Purpose: Define section properties for Peridynamics solid and laminate elements.

Card 1	1	2	3	4	5	6	7	8
Variable	SECID	ELFORM						
Type	I/A	I						

VARIABLE	DESCRIPTION
SECID	Section ID. SECID is referenced on the *PART card. A unique number or label must be specified.
ELFORM	Element formulation options. EQ.48: For peridynamics solids, accepting 4, 6, and 8-noded elements For peridynamics laminates, accepting 4-noded surface (shell) elements

Card 2	1	2	3	4	5	6	7	8
Variable	DR	PTYPE						
Type	F	I						
Default	1.01	1						

VARIABLE	DESCRIPTION
DR	Normalized horizon size, 0.6 – 1.2 is recommended
PTYPE	Peridynamics formulation: EQ.1: Bond based formulation (default)

Remarks:

4.The Peridynamics solid parts use the elements listed as *ELEMENT_SOLID. These elements can be built by 3D solid mesher. To represent material split, the nodes of these elements must be detached, i.e., the total number of nodes equals the summation of nodal numbers of Peridynamics elements. It also implies that every nodal point of each element has its own nodal ID. Adjacent elements have nodal points at the same position initially.
5.The Peridynamics laminate parts use the elements listed as *ELEMENT_SOLID_-PERI. See Keyword *ELEMENT_SOLID_PERI for the details.

*SET_PERI_LAMINATE

Purpose: Assemble laminae (each lamina must be defined as one individual part) to a laminate and define the fiber angles for laminae.

Card 1	1	2	3	4	5	6	7	8
Variable	SID							
Type	I							
Default								

VARIABLE	DESCRIPTION
SID	The set ID.

Card 2	1	2	3	4	5	6	7	8
Variable	PID1	A1	T1	PID2	A2	T2		
Type	I	F	F	I	F	F		
Default								

VARIABLE	DESCRIPTION
PID1	The Part ID of first lamina.
A1	The fiber angle of first lamina. The zero degree is defined in *MAT_ELASTIC_PERI_LAMINATE as (V1,V2,V3)
T1	The thickness of first lamina.
PID2	The Part ID of second lamina.

VARIABLE	DESCRIPTION
A2	The fiber angle of second lamina.
T2	The thickness of second lamina.

Repeat Card 2 until all laminae are assembled.

SPG

The Smoothed Particle Galerkin (SPG) method is a new generation numerical method for dynamic material failure simulation. SPG theory was proposed by Livermore Software Technology (LST) R&D team in 2014. The first LS-DYNA version was released in 2017. A momentum consistent SPG (MCSPG) formulation was developed in 2019 which improves the original version in both accuracy and efficiency and enables coupled thermal mechanical analysis as well. The 2021 version supports more features with more efficient computation.

The SPG weak form is integrated with the Direct Nodal Integration (DNI) technique whereas the low energy modes accompanying with DNI are alleviated with a stabilization enhancement term derived from displacement smoothing theory. As opposed to the residual type stabilization, the SPG stabilization formulation is a non-residual type in which the penalized stabilization functional is parameterized by a measure of the local length scale without a need of stabilization control parameters. DNI (without background cells) facilitates weak form integration upon material separation, otherwise it is very difficult, if not impossible, to build conforming integration cells in grid-based integration schemes (e.g. Gauss integration) when material failure occurs. Furthermore, the SPG framework is equipped with a bond-based failure mechanism such that material separation can be captured without material deletion (typically in mesh or grid – based methods) and thus not violating conservation laws.

As SPG preserves the desired conservation properties in mass and momentum, it can predict physical responses for a wide range of material failure applications in various industries, such as automobile, electronics, civil, defense, wood, and biomedical. Typical SPG applications include but not limited to screwing, machining, joining, impact penetration, and many other manufacturing process simulations involving material failure and removal, thermal mechanical effect, and severe deformation. SPG can perform various material failure analyses for isotropic and anisotropic (orthotropic) materials such as metal, concrete, wood, foam, rubber, composites, etc.

The SPG solver includes the following keywords:

*SECTION_SOLID_SPG

*CONSTRAINED_IMMERSED_IN_SPG

*CONTACT_SPG

*INTERFACE_SPG_1

*INTERFACE_SPG_2

***CONTACT_SPG**

Purpose: This keyword was developed to deal with contact between SPG particles from different SPG parts or self-contact of SPG particles from the same SPG part. It was developed for high speed deformations such as projectile impact penetration problems.

Card 1	1	2	3	4	5	6	7	8
Variable	IPART1	IPART2	IPART3	IPART4	IPART5	IPART6	IPART7	IPART8
Type	I	I	I	I	I	I	I	I
Default	0	0	0	0	0	0	0	0

Card 2	1	2	3	4	5	6	7	8
Variable	ISELF1	ISELF2	ISELF3	ISELF4	ISELF5	ISELF6	ISELF7	ISELF8
Type	I	I	I	I	I	I	I	I
Default	0	0	0	0	0	0	0	0

Card 3	1	2	3	4	5	6	7	8
Variable	PFAC1	PFAC2	PFAC3	PFAC4	PFAC5	PFAC6	PFAC7	PFAC8
Type	F	F	F	F	F	F	F	F
Default	0	0	0	0	0	0	0	0

Card 4	1	2	3	4	5	6	7	8
Variable	COFS	COFD	DC	NFEQ				
Type	F	F	F	F				
Default	0	0	0	0				

Variable	Description
IPARTs	Part IDs of SPG parts involved in particle contact
ISELFs	Self-contact indicators = 0: no self-contact = 1: self-contact is defined for the corresponding part

PFACs	Penalty factors
COFS	Static coefficient of friction
COFD	Dynamic coefficient of friction
DC	Decay factor of coefficient of friction (refer to regular contact)
NFEQ	Contact searching frequency

*INTERFACE_SPG_1

Purpose: This keyword is used when a two-stage SPG analysis is desired, and it indicates the current one is the first stage. Whether or not this keyword is included in an SPG calculation, the result is not changed.

This keyword does not require any parameter.

To use the two-stage SPG analysis feature, **at least** LS-DYNA **R13** is needed.

When this keyword is included, an ASCII file named "1234spg" will be generated at normal termination. This file contains the information (stress, strain, total displacement, etc.) of SPG nodes at the end of the calculation.

To prepare for the second stage analysis, *INTERFACE _SPRINGBACK_LSDYNA is also needed in the first stage analysis (besides *INTERFACE_SPG_1) to output "dynain" at normal termination. The part set for *INTERFACE _SPRINGBACK_LSDYNA must include SPG parts since latest SPG coordinates are needed.

*INTERFACE_SPG_2

Purpose: This keyword is used when a second stage SPG analysis is performed in the two-stage SPG analysis.

This keyword does not require any parameter.

To use the two-stage SPG analysis feature, **at least** LS-DYNA **R13** is needed.

To perform a second stage analysis, the ASCII file "1234spg" generated in the first stage calculation must be copied to the current folder and renamed as "1234spg0". The "dynain" file generated in the first stage calculation must also be included in current input. Other setups (material laws, boundary conditions, etc.) are the same as a regular analysis, except the element connectivity and nodal coordinates since the ones in "dynain" will be used.

*SECTION_SOLID_SPG

Purpose: Define section properties for SPG solid.

To use the default parameters, **at least** LS-DYNA **R12** is needed.

Card Summary:

Card Sets. For each unique solid section, include one set of data cards. This input ends at the next keyword ("*") card. For detailed description of keyword *SECTION_SOLID, please refer to the Keyword Manual Volume I.

Card 1. This card is required.

SECID	ELFORM	AET					

Card 2b.1. This card is required for the SPG keyword option.

DX	DY	DZ	ISPLINE	KERNEL		SMSTEP	MSC

Card 2b.2. This card is read if the SPG keyword option is used. It is optional.

IDAM	FS	STRETCH	ITB	MSFAC	ISC	IDBOX	PDAMP

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	SECID	ELFORM	AET					
Type	I/A	I	I					
Default		47						

VARIABLE	DESCRIPTION
SECID	Section ID. SECID is referenced on the *PART card. A unique number or label must be specified.
ELFORM	Element formulation options: EQ.47: Smoothed Particle Galerkin (SPG) method (see Remark 1)

SPG Card. Additional card for the SPG keyword option.

Card 2b.1	1	2	3	4	5	6	7	8
Variable	DX	DY	DZ	ISPLINE	KERNEL		SMSTEP	MSC
Type	F	F	F	I	I		I	F
Default				0	0		0	0

VARIABLE	DESCRIPTION
DX, DY, DZ	Normalized dilation parameters of the kernel function in *x*-, *y*-, and *z*- directions, respectively. The normalized dilation parameters of the kernel function are introduced to provide the smoothness and locality on the construction of the mesh-free shape functions. Values between 1.4 and 1.8 are recommended. Values smaller than 1.0 are not allowed. Larger values will increase the computational time and will sometimes result in divergence of the solution. Default: 1.6 if KERNEL = 0 1.8 if KERNEL = 1 1.5 if KERNEL = 2 (see following for KERNEL)
ISPLINE	Type of kernel function: EQ.0: Cubic spline function with cubical support (default) EQ.1: Quadratic spline function with cubical support EQ.2: Cubic spline function with spherical support
KERNEL	Type of kernel support update scheme: EQ.0: Updated Lagrangian, failure or no failure analysis, tension dominant problem EQ.1: Eulerian, failure analysis, global extreme deformation EQ.2: Pseudo Lagrangian, failure analysis, local extreme deformation
SMSTEP	Interval of time steps to conduct displacement smoothing Default: 15, if KERNEL = 0 5, if KERNEL = 1 30, if KERNEL = 2 Code will determine according to KERNEL if SMSTEP = 0 from input

VARIABLE	DESCRIPTION
MSC	For **ITB = 3** (see Card 2b.2 below) only EQ.0: Regular smoothing scheme EQ.1: New smoothing scheme for very low speed deformation, with better controls of the low energy modes than the regular smoothing scheme

Optional SPG Card. Additional optional card for the SPG keyword option.

Card 2b.2	1	2	3	4	5	6	7	8
Variable	IDAM	FS	STRETCH	ITB	MSFAC	ISC	IDBOX	PDAMP
Type	I	F	F	I	F	F	I	F
Default	0	↓	↓	0	1	0	0	-0.001

VARIABLE	DESCRIPTION
IDAM	Option of bond failure mechanism EQ.1: Effective plastic strain (phenomenological strain damage, default) EQ.2: Maximum principal stress EQ.3: Maximum shear strain EQ.4: Minimum principal strain (input must be positive) EQ.5: Effective plastic strain and maximum shear strain EQ.7: Anisotropic damage for honeycomb only Only used with *MAT_126 (**At least R13**) **Recommended** to be used with **ITB = 3** (see below) EQ.11: Pre-damage model for brittle material failure (with crack propagation) **At least R13**. Bond failure with stress degradation **Recommended** to be used with **ITB = 3** (see below) EQ.13: Pre-damage model for ductile material failure

VARIABLE	DESCRIPTION
	At least R13. No bond failure, stress degradation **Recommended** to be used with **ITB = 3** (see below)
FS	Critical value of the quantity indicated by IDAM for bond failure triggering. Default: 1.0E+10, i.e., no failure analysis For *MAT_3 and *MAT_24, "FS" on material cards overwrites this value. *When FS is defined on material cards, not only bond failure will occur when it is reached, but also the stress will be set to zero according to material law.* *If FS is defined on SPG card, only bond failure will occur without setting stress to zero.*
STRETCH	Critical relative deformation (stretching or compression ratio) between the two nodes forming the bond for bond failure
ITB	Option of stabilization: EQ.1: Fluid particle approximation (accurate but slow), used with KERNEL = 0 or 1 EQ.2: Simplified fluid particle approximation (efficient and robust), used with KERNEL = 2 EQ.3: Momentum consistent SPG (MCSPG) formulation (latest beta or R14 is recommended). MCSPG can be applied for large deformation, tension dominant problems. For coupled thermal mechanical problems, MCSPG is the only option. KERNEL = 1 is recommended for MCSPG. Default: 1, if KERNEL = 0 or 1 2, if KERNEL = 2
MSFAC	For **ITB = 3** only Quadrature factor for surface nodes (latest beta or R14): To suppress shear locking in thin structure Default: 1.00: for regular solid structure 0.75: for thin structure
ISC	Self-contact indicator EQ.0: No self-contact between the bond-failed particles in the same part

VARIABLE	DESCRIPTION
	GT.0: Self-contact is defined between the bond-failed particles in the same part. Usually 0.01~0.1 Young's modulus. This option is available with **SMP only!**
IDBOX	ID of a box defining the active SPG region. Outside this region, the particles are eliminated from SPG calculation. See *DEFINE_-BOX.
PDAMP	For **ITB = 3** only Particle-to-particle damping coefficient Default: -0.001 Range: -0.01~-0.001, positive value is not recommended

Remarks:

1. **Smoothed Particle Galerkin (SPG) method: Type 47.** In SPG method, finite element nodes are converted into particles, and 4-node, 6-node and 8-node *solid* elements are supported. The method is suitable for severe deformation problem and failure analysis.

2. When SPG particles are used as the passive part in a contact, a node set must be defined to include the SPG particles, not part or part set ID. In other words, SSTYP must be 4, not 2 or 3.

Examples:

1. Default setup (updated Lagrangian)

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$$   *SECTION_SOLID_SPG
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*SECTION_SOLID_SPG
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$    secid    elform
         1
$       dx        dy        dz   ispline    kernel              smstep       msc

$     idam        fs   stretch       itb     msfac       isc     idbox     pdamp
                0.45
$  The discretization for SPG part is exactly the same as that for FEM
$  Only one parameter which is FS is needed for this case
$  At least R12 is needed for this card where the majority takes the default value
```

```
$  The above card means:
$    DX=DY=DZ=1.6
$    ISPLINE=0(Cubic B-Spline)
$    KERNEL=0 (updated Lagrangian)
$    SMSTEP=15 (support updating is performed every 15 time steps)
$    IDAM=1 (effective plastic strain as bond failure criterion)
$    FS=0.45 (critical effective plastic strain)
$      If *MAT_3 or *MAT_24 is used while FS is input in material card, this can be
$      blank or 0 as well
$    STRETCH=1.2
$      If *MAT_3 or *MAT_24 is used, this can be blank or 0 and will be set to
$      1.0+0.4*FS
$    ITB=1 (regular fluid particle algorithm for stabilization)
$
$$$$$$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

2. Eulerian support update

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$$   *SECTION_SOLID_SPG
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*SECTION_SOLID_SPG
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$    secid    elform
         1
$       dx        dy        dz   ispline    kernel              smstep       msc
                                                1
$     idam        fs   stretch       itb     msfac       isc     idbox     pdamp
                0.45                                                1
$  The discretization for SPG part is exactly the same as that for FEM
$  At least R12 is needed for this card where the majority takes the default value
$  The above card means:
$    DX=DY=DZ=1.8
$    ISPLINE=0(Cubic B-Spline)
$    KERNEL=1 (Eulerian)
$    SMSTEP=5 (support updating is performed every 5 time steps)
$    IDAM=1 (effective plastic strain as bond failure criterion)
$    FS=0.45 (critical effective plastic strain)
$      If *MAT_3 or *MAT_24 is used while FS is input in material card, this can be
$      blank or 0 as well
$    STRETCH=1.2
$      If *MAT_3 or *MAT_24 is used, this can be blank or 0 and will be set to
$      1.0+0.4*FS
$    ITB=1 (regular fluid particle algorithm for stabilization)
$    IDBOX=1 (a box with ID=1 must be defined using *DEFINE BOX)
$
$$$$$$$$$$$$$$$$$$
```

3. Pseudo Lagrangian support update

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$$   *SECTION_SOLID_SPG
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*SECTION_SOLID_SPG
```

```
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$    secid    elform
         1
$       dx        dy        dz   ispline    kernel              smstep       msc
                                                2
$     idam        fs   stretch       itb     msfac       isc     idbox     pdamp
                0.45                                    1.0
$  The discretization for SPG part is exactly the same as that for FEM
$  At least R12 is needed for this card where the majority takes the default value
$  The above card means:
$    DX=DY=DZ=1.5
$    ISPLINE=0(Cubic B-Spline)
$    KERNEL=2 (Pseudo Lagrangian)
$    SMSTEP=30 (support updating is performed every 30 time steps)
$    IDAM=1 (effective plastic strain as bond failure criterion)
$    FS=0.45 (critical effective plastic strain)
$      If *MAT_3 or *MAT_24 is used while FS is input in material card, this can be
$      blank or 0 as well
$    STRETCH=1.2
$      If *MAT_3 or *MAT_24 is used, this can be blank or 0 and will be set to
$      1.0+0.4*FS
$    ITB=1 (regular fluid particle algorithm for stabilization)
$    ISC=1.0GPa (self-contact is activated, pay attention to the unit, SMP only)
$
$$$$$$$$$$$$$$$$$$$
```

4. MCSPG

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$$   *SECTION_SOLID_SPG
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*SECTION_SOLID_SPG
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$    secid    elform
         1
$       dx        dy        dz   ispline    kernel              smstep       msc
                                                1
$     idam        fs   stretch       itb     msfac       isc     idbox     pdamp
                0.45                    3
$  The discretization for SPG part is exactly the same as that for FEM
$  Latest beta version is recommended for this application
$  The above card means:
$    DX=DY=DZ=1.8
$    ISPLINE=0(Cubic B-Spline)
$    KERNEL=1 (Eulerian)
$    SMSTEP=15 (support updating is performed every 15 time steps)
$    IDAM=1 (effective plastic strain as bond failure criterion)
$    FS=0.45 (critical effective plastic strain)
$      If *MAT_3 or *MAT_24 is used while FS is input in material card, this can be
$      blank or 0 as well
$    STRETCH=1.2
$      If *MAT_3 or *MAT_24 is used, this can be blank or 0 and will be set to
$      1.0+0.4*FS
$    ITB=3 (Momentum consistent SPG formulation)
$    MSC=1 (Default for MCSPG)
$    MSFAC=1 (Default for MCSPG)
$    PDAMP=-0.001 (Default for MCSPG)
$
$$$$$$$$$$$$$$$$$$$
```

$$$

XFEM

The Extended Finite Element Method (XFEM) is developed to simulate dynamic material failure and crack propagation in 2D plane strain plates and 3D shell structures. It can conduct both brittle and ductile fracture analyses.

In the XFEM theory, the fracture processing zone around a crack tip is simplified with the cohesive zone model. A certain stress-based failure criterion (for brittle/semi-brittle material) or strain-based failure criterion (for ductile material) is used to predict crack initiation and propagation. When an element reaches the failure criterion, a cohesive crack is inserted in the element and a strong discontinuity is enhanced in the finite element solution. A cohesive material law governs the kinetics in the cohesive crack surface and the work done by the cohesive traction force represents either the energy release rate (in brittle fracture) or plastic work (in ductile fracture) needed to create or propagate the physical crack when the traction in the cohesive surface becomes zero. In ductile fracture analysis, the non-physical strain localization associated with local constitutive material law can be suppressed with non-local material model. We provide with two approaches for ductile fracture: standard constitutive material model with strain regularization and non-local continuum damage material model. The later can be easily achieved by adding non-local GISSMO model to any material law for ductile materials, and in this case, the cohesive material law is no longer needed since the stresses in the failed element are zero due to the damage induced stress softening.

The XFEM solver includes the following keywords specific to this method:

*BOUNDARY_PRECRACK

*SECTION_SHELL_XFEM

Some other keywords used by the XFEM solver include:

*DATABASE_EXTENT_BINARY

*MAT_ADD_EROSION/*MAT_ADD_DAMAGE_GISSMO

*MAT_COHESIVE_TH

*MAT_COHESIVE_MIXED_MODE_ELASTOPLASTIC_RATE_3MODES

*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.

DATABASE_EXTENT_BINARY_*{OPTION}

Purpose: Control to some extent the content of binary output databases d3plot, d3thdt, and d3part. See also *DATABASE_BINARY_*OPTION* and *DATBASE_EXTENT_D3-PART. The content of the binary output database intfor may be modified using *DATA-BASE_EXTENT_INTFOR.

This keyword is used in XFEM to add a history variable to write out Failure Indicator in d3plot.

For the complete description of this keyword, please refer to the Keyword Manual Volume I.

Card Summary:

Card 1b. This card is included if no keyword option is used.

NEIPH	NEIPS	MAXINT	STRFLG	SIGFLG	EPSFLG	RLTFLG	ENGFLG

Data Card Definitions:

Card 1b	1	2	3	4	5	6	7	8
Variable	NEIPH	**NEIPS**	MAXINT	STRFLG	SIGFLG	EPSFLG	RLTFLG	ENGFLG
Type	I	I	I	I	I	I	I	I
Default	0	0	3	0	1	1	1	1

VARIABLE	DESCRIPTION
NEIPS	Number of additional integration point history variables written to the binary databases (d3plot, d3part, d3drlf) for both shell and thick shell elements for each integration point. To output Failure Indicator for XFEM elements, increase the value of NEIPS by 1 (or set it to be 1 if it is zero). The Failure Indicator will be "historyvar #1" in d3plot.

*MAT_COHESIVE_TH

This is Material Type 185. It is a cohesive model by Tvergaard and Hutchinson [1992] for use with cohesive element fomulations; see the variable ELFORM in *SECTION_SOLID and *SECTION_SHELL. The implementation is based on the description of the implementation in the Sandia National Laboratory code, Tahoe [2003].

Card 1	1	2	3	4	5	6	7	8
Variable	MID	RO	ROFLG	INTFAIL	SIGMAX	NLS	TLS	TLS2
Type	A8	F	F	F	F	F	F	F

Card 2	1	2	3	4	5	6	7	8
Variable	LAMDA1	LAMDA2	LAMDAF	STFSF	ISW	ALPHA1	ALPHA2	
Type	F	F	F	F	I	F	F	

Additional card 3 may be used for XFEM shell, see *SECTION_SHELL_XFEM.

Card 3	1	2	3	4	5	6	7	8
Variable	DR	ALPHA3						
Type	F	F						

VARIABLE	DESCRIPTION
MID	Material identification. A unique number or label not exceeding 8 characters must be specified.
RO	Mass density
ROFLG	Flag for whether density is specified per unit area or volume. ROFLG = 0 specified density per unit volume (default), and ROFLG = 1 specifies the density is per unit area for controlling the mass of cohesive elements with an initial volume of zero.

VARIABLE	DESCRIPTION
INTFAIL	The number of integration points required for the cohesive element to be deleted. If it is zero, the element won't be deleted even if it satisfies the failure criterion. The value of INTFAIL may range from 1 to 4, with 1 the recommended value.
SIGMAX	Peak traction.
NLS	Length scale (maximum separation) in the normal direction.
TLS	Length scale (maximum separation) in the tangential direction.
LAMDA1	Scaled distance to peak traction (Λ_1).
LAMDA2	Scaled distance to beginning of softening (Λ_2).
LAMDAF	Scaled distance for failure (Λ_{fail}).
STFSF	Penetration stiffness multiplier. The penetration stiffness, *PS*, in terms of input parameters becomes: $$\mathrm{PS} = \frac{\mathrm{STFSF} \times \mathrm{SIGMAX}}{\mathrm{NLS} \times \left(\frac{\mathrm{LAMDA1}}{\mathrm{LAMDAF}}\right)}$$

Following parameters are used only in XFEM shell:

TLS2	Length scale (maximum separation) in the tear direction.
ISW	EQ. -1: initially rigid cohesive law (type I) EQ. -2: initially rigid cohesive law (type II)
ALPHA1	Ratio of maximum mode II shear traction to normal traction
ALPHA2	Ratio of maximum mode III shear traction to normal traction (available in shell)
DR	Critical rotation scale (available in shell)
ALPHA3	Ratio of maximum bending moment to normal traction (available in shell)

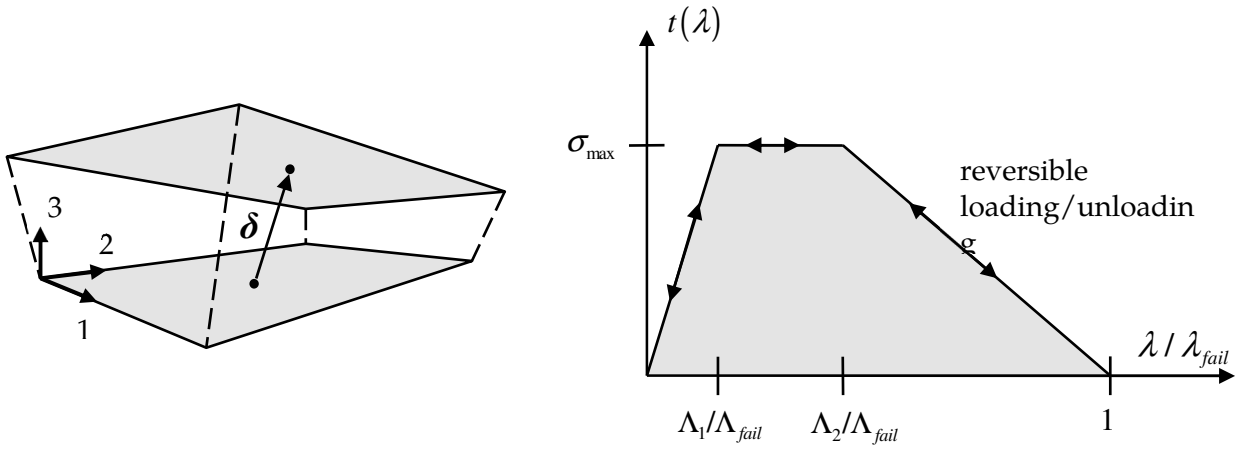


Figure 3-1. Relative displacement and trilinear traction-separation law

Remarks:

In this cohesive material model, a dimensionless separation measure λ is used, which grasps for the interaction between relative displacements in normal (δ_3 - mode I) and tangential (δ_1, δ_2 - mode II) directions (see Figure 3-1 left):

$$\lambda = \sqrt{\left(\frac{\delta_1}{\mathrm{TLS}}\right)^2 + \left(\frac{\delta_2}{\mathrm{TLS}}\right)^2 + \left(\frac{\langle \delta_3 \rangle}{\mathrm{NLS}}\right)^2}$$

where the Mc-Cauley bracket is used to distinguish between tension ($\delta_3 \geq 0$) and compression ($\delta_3 < 0$). NLS and TLS are critical values, representing the maximum separations in the interface in normal and tangential direction. For stress calculation, a trilinear traction-separation law is used, which is given by (see Figure 3-1 right):

$$t(\lambda) = \begin{cases} \sigma_{\max} \dfrac{\lambda}{\Lambda_1/\Lambda_{\mathrm{fail}}} & \lambda < \Lambda_1/\Lambda_{\mathrm{fail}} \\ \sigma_{\max} & \Lambda_1/\Lambda_{\mathrm{fail}} < \lambda < \Lambda_2/\Lambda_{\mathrm{fail}} \\ \sigma_{\max} \dfrac{1-\lambda}{1-\Lambda_2/\Lambda_{\mathrm{fail}}} & \Lambda_2/\Lambda_{\mathrm{fail}} < \lambda < 1 \end{cases}$$

With these definitions, the traction drops to zero when $\lambda = 1$. Then, a potential ϕ is defined as:

$$\phi(\delta_1, \delta_2, \delta_3) = \mathrm{NLS} \times \int_0^{\lambda} t(\bar{\lambda})\, d\bar{\lambda}$$

Finally, tangential components (t_1, t_2) and normal component (t_3) of the traction acting on the interface in the fracture process zone are given by:

$$t_{1,2} = \frac{\partial \phi}{\partial \delta_{1,2}} = \frac{t(\lambda)}{\lambda} \frac{\delta_{1,2}}{\mathrm{TLS}} \frac{\mathrm{NLS}}{\mathrm{TLS}}, \qquad t_3 = \frac{\partial \phi}{\partial \delta_3} = \frac{t(\lambda)}{\lambda} \frac{\delta_3}{\mathrm{NLS}}$$

which in matrix notation is

$$\begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \frac{t(\lambda)}{\lambda} \begin{bmatrix} \frac{\text{NLS}}{\text{TLS}^2} & 0 & 0 \\ 0 & \frac{\text{NLS}}{\text{TLS}^2} & 0 \\ 0 & 0 & \frac{1}{\text{NLS}} \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix}$$

In case of compression ($\delta_3 < 0$), penetration is avoided by:

$$t_3 = \frac{\text{STFSF} \times \sigma_{\max}}{\text{NLS} \times \Lambda_1 / \Lambda_{\text{fail}}} \delta_3$$

Loading and unloading follows the same path, i.e. this model is completely reversible.

This cohesive material model outputs three tractions having units of force per unit area into the D3PLOT database rather than the usual six stress components. The in plane shear traction t_1 along the 1-2 edge replaces the x-stress, the orthogonal in plane shear traction t_2 replaces the y-stress, and the traction in the normal direction t_3 replaces the z-stress.

Used in XFEM, TLS corresponding to δ_2 *in* above equations is replaced with TLS2. Since the initially rigid cohesive law is used, element fails only when the stress level reaches the SIGMAX and Λ_1 is only used to define the penetration stiffness in case of crack closing (compression).

*SECTION_SHELL_XFEM

Purpose: Define section properties for XFEM plate or shell elements.

Card Summary:

Card Sets. For each shell section, of a type matching the keyword's options, include one set of data cards. This input ends at the next keyword ("*") card. For detailed description of keyword *SECTION_SHELL, please refer to the Keyword Manual Volume I.

Card 1. This card is required.

SECID	ELFORM	SHRF	NIP	PROPT	QR / IRID	ICOMP	SETYP

Card 2. This card is required.

T1	T2	T3	T4	NLOC	MAREA	IDOF	EDGSET

Card 4c. This card is required for XFEM.

CMID	BASELM	DOMINT	FAILCR	PROPCR	FS	LS/FS1	NC/CL

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	SECID	ELFORM	SHRF	NIP	PROPT	QR / IRID	ICOMP	SETYP
Type	I/A	I	F	F	F	F	I	I
Default	none	none	1.0	2	0.0	0.0	0	1

VARIABLE	DESCRIPTION
SECID	Section ID. SECID is referenced on the *PART card. A unique number or label must be specified.
ELFORM	Element formulation option: EQ.52: Plane strain (*xy*-plane) XFEM, base element type 13 with full integration. See Remarks 1 and 2. EQ.54: Shell XFEM, base element type defined by BASELM (default 2). See Remark 2.

XFEM Card. Input card for XFEM keyword option. See Remark 2.

Card 4c	1	2	3	4	5	6	7	8
Variable	CMID	BASELM	DOMINT	FAILCR	PROPCR	FS	LS/FS1	NC/CL
Type	I	I	I	I	I	F	F	I/F
Default	none	none	0	1	0	0.0	0.0	none

VARIABLE	DESCRIPTION
CMID	Cohesive material ID. *MAT_185/*MAT_COHESIVE_TH is available for both brittle and ductile fracture and *MAT_240/ *MAT_COHESIVE_MIXED_MODE_ELASTOPLASTIC_RATE_3MODES is available for ductile material. See Remark 4.
BASELM	Base element type for XFEM (type 13 for 2D, types 2, 16 or 17 for shell)
DOMINT	Option for domain integration in XFEM: EQ.0: Phantom element integration (default) EQ.1: Subdomain integration with triangular local boundary integration (available in 2D only)
FAILCR	Option for different failure criteria: EQ.0: Failure criterion from *MAT_ADD_EROSION/GISSMO model EQ.1: Maximum tensile stress (failure value given in cohesive law) EQ.2: Maximum shear stress (failure value given in cohesive law) EQ.-1: Effective plastic strain (EPS) EQ.-2: Crack length dependent EPS. See Remark 3. EQ.-4: Stress triaxiality based failure plastic strain
PROPCR	PROPCR is interpreted digit-wise: $\text{PROPCR} = [IP] = P + 10 \times I$ P determines the crack propagation direction:

VARIABLE	DESCRIPTION
	P.EQ.0: first principal strain direction if FAILCR < 0 (default for ductile fracture), first principal stress if FAILCR = 1, or maximum shear stress if FAILCR = 2. P = 0 is the only option for brittle fracture (FAILCR > 0). P.EQ.2: center of effective plastic strain P.EQ.3: directional center of effective plastic strain *I* determines crack initiation: I.EQ.0: crack initiates at boundary (default) I.EQ.3: crack initiates anywhere
FS	Failure value for FAILCR = -1 or -2: FAILCR.EQ.-1: Failure strain/failure critical value FAILCR.EQ.-2: Initial failure plastic strain Curve ID or Table ID for stress triaxiality based failure plastic strain for FAILCR = -4.
LS	Length scale for strain regularization (FAILCR = -1 only)
FS1	Final failure plastic strain (FAILCR = -2 only)
NC	Number of cracks allowed in the part (FAILCR .NE. -2)
CL	Crack length at which the failure strain is FS1 (FAILCR = -2 only)

Remarks:

1. **2D Formulations.** For 2D formulations (ELFORM=52), nodes must lie in the global *xy*-plane, so the *z*-coordinate must be zero. Furthermore, the element normal should be in positive *z* direction.

   Shell thickness values on Card 2 are ignored. For defining contact in 2D simulations, see the entry for the *CONTACT_2D keyword.

2. **Fracture.** XFEM 2D and shell formulations can be used for brittle or semi-brittle fracture with pre-crack (see *BOUNDARY_PRECRACK) or with geometry imperfection such as a notch or a hole, and for ductile fracture analysis with regularized effective plastic strain criterion or nonlocal continuum damage material models (provided with GISSMO model).

3. **XFEM ductile fracture.** This feature is supported by a joint research among Honda, JSOL and LSTC. For FAILCR = -2, the failure strain is given by:

$$\mathrm{EPS} = \mathrm{FS} + (\mathrm{FS1} - \mathrm{FS}) \times \min(L/\mathrm{CL}, 1.0)$$

4. **Cohesive Material Law for XFEM.** In XFEM formulation, the fracture process zone around the crack tip is simplified with cohesive zone model. The initially-rigid cohesive material law provides the kinetic relationship (traction force as a function of crack opening displacement) in the cohesive crack surface inserted in the element reaching the failure criterion and the work done by the traction represents the energy release rate in brittle fracture or plastic work in ductile fracture to create a real crack surface where the traction force becomes zero. When nonlocal continuum damage constitutive model is used in ductile fracture, the cohesive law is not needed since the stresses in the failed element become zero due to the damage induced stress softening.

Example:

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$$  *SECTION_SHELL_XFEM
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$  Define an XFEM shell section that specifies the following:
$     elform = 54  XFEM shell formulation.
$        nip = 3   Three through the shell thickness integration points.
$    t1 - t4 = 2.0 A shell thickness of 2 mm at all nodes.
$       cmid = 100 Cohesive material ID 100
$     baselm = 16  Base FEM type 16
$     failcr = -1  Critical effective plastic strain criterion
$     propcr = 0   First principal strain direction as normal to crack direction
$         fs = 0.1 Failure strain
$         ls = 1.0 Length scale for strain regularization
$
*SECTION_SHELL_XFEM
$
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
$      sid    elform      shrf       nip     propt   qr/irid     icomp
         1        54              3.0000
$
$       t1        t2        t3        t4      nloc
       2.0       2.0       2.0       2.0
$
$     cmid    baselm    domint    failcr    propcr        fs        ls
       100        16         0        -1         0       0.1       1.0
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

Multiscale Geometric Analysis

Multiscale Geometric Analysis is an advanced numerical tool to help users resolve geometric details in large scale structural analysis, where these details often have relatively smaller dimension in meso scale compared to the overall computational domain but play a very important role in structural response. The typical application includes but not limited to connectors or assembly components in products, e.g. solder joints on printed circuit board (PCB) and chips for reliability analysis of consumer electronics products, spotwelds and rivets in the performance analysis of car components and the crashworthiness simulation of full cars.

In order to capture the geometric details in meso scale, small mesh size is needed compared to the large scale structure model, which makes the computation infeasible due to dramatically increased model size, much smaller time step size and load balance issue in parallelization. Further, solid FEM and advanced element formulations, e.g. meshfree and particle methods, are often used to perform material failure analysis in meso scale, which brings challenge on creating conforming mesh and coupling with large-scale solid and shell structures.

In this chapter, the keywords of two-scale co-simulation method are presented, which enable the capability to run two separated LS-DYNA MPP jobs simultaneously: one for large-scale and the other for meso-scale. Two jobs run at different time step sizes and synchronize automatically at every large scale time step. To minimize the work on modeling, non-conforming mesh at the coupling interface can be used so that there is no need to modify the large scale mesh when coupling with the meso scale model.

In the current implementation, there are two different types of two-scale co-simulation:

1. The weak coupling

   *BOUNDARY_COUPLED

   *DEFINE_MULTISCALE

   *INCLUDE_MULTISCALE

2. The strong coupling

   *INCLUDE_COSIM

In the weak coupling, the large scale model imposes kinematic constraints on the meso scale model at the coupling interface and drives its deformation. The failure of large scale representative beams is determined by the meso scale analysis. The strong coupling, on the other hand, is fully concurrent simulation across two scales, where the large scale

model imposes the kinematic constrains on meso scale and obtains the kinetic response as return. Therefore, there is no simplified representation of meso scale model in the large scale computation in the strong coupling.

*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/general local models
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 or general

multiscale (TYPE = 2, PROG = 1). See *INCLUDE_MULTISCALE_SPOTWELD or *INCLUDE_MULTISCALE for information about using this capability.

*DEFINE_MULTISCALE

Purpose: Associate beam sets with multi-scale local model IDs for modeling detailed local model failure via the multi-scale method.

Local model/Beam Set Association Cards. Provide 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	BSET	ID	BSET	ID	BSET	ID	BSET
Type	I	I	I	I	I	I	I	I
Default	none	none	none	none	none	none	none	none

VARIABLE	DESCRIPTION
ID	MULTISCALE local model ID to use. See *INCLUDE_MULTISCALE.
BSET	Beam set which uses this multi-scale local model ID for failure modeling.

Remarks:

See *INCLUDE_MULTISCALE for a detailed explanation of this capability.

*INCLUDE_MULTISCALE

Purpose: To define a type of MULTISCALE platform to be used for coupling and for modeling of local model failure. This keyword is an extension of **INCLUDE_MULTISCALE_SPOTWELD for more general local model usage.**

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

Card 2	1	2	3	4	5	6	7	8
Variable	FILENAME							
Type	C							
Default	none							

VARIABLE	DESCRIPTION
ID	ID for this multiscale local model. This ID is used in the keyword *DEFINE_MULTISCALE. Any unique integer will do.
FILENAME	Name of file from which to read the local model definition.

Remarks:

This capability is available *only* in the MPP version of LS-DYNA.

With the multiscale feature, heuristic beams models are replaced with zoomed-in geometrically and constitutively correct continuum models, which, in turn are coupled to the large-scale calculation without reducing the time step. Because the detailed local models are run in a separate process, they can run at a much smaller time step without slowing down the rest of the simulation. A brief outline of their use looks like this:

- The user creates one (or more) detailed local model of their beams, and includes these definitions into their model using the keyword *INCLUDE_MULTISCALE.

In the large-scale model, the beams must be on the microscale model side of a tied interface.

- The user indicates which beam should be coupled to these models via the keyword *DEFINE_MULTISCALE.
- When MPP-DYNA is started, a special (MPI dependent) invocation is required in order to run in a "multiple program" mode. Effectively, two separate instances of MPP-DYNA are started together, one to run the full model and a separate instance to run the local models.
- As the master process runs, each cycle it communicates to the microscale process deformation information for the area surrounding each coupled beam. The microscale process imposes this deformation on the detailed solid local models, computes a failure flag for each, and communicates this back to the master process.
- The coupled beams in the master process have their failure determined solely by these failure flags.

The file referred to on the *INCLUDE_MULTISCALE card should contain one generic instance of a detailed local model. For each coupled beam in the main model, a specific instance of this beam will be generated which is translated, rotated, and scaled to match the solid local model to which it is coupled. In this way, many beams can be coupled with only a single *INCLUDE_MULTISCALE. The included file should contain everything required to define a detailed local model, such as *MAT and *PART definitions, any required *DEFINE_CURVEs, etc., as well as *NODE and *ELEMENT definitions. In order for the translation and scaling to work properly there are some assumptions made about the detailed local model:

- It consists entirely of solid elements.
- The z-axis is aligned with the coupled local model in the main model, with $z = 0$ and $z = 1$ at the two ends of the local model.
- The cross sectional area of the local model in the xy plane is equal to 1.
- That portion of the "top" and "bottom" of the local model that are coupled are identified using a single *SET_NODE_LIST card.
- One *BOUNDARY_COUPLED card referencing the *SET_NODE_LIST of the boundary nodes is required. It must specify a coupling type of 2 and a coupling program of 1.
- The detailed local model can include multiple parts.
- The detailed local model does not support *INCLUDE cards.

Failure of the local model is determined topologically. Any element of the detailed local model having all four nodes of one of its faces belonging to the *SET_NODE_LIST of tied nodes is classified as a "tied" element. The "tied" elements are partitioned into two disjoint sets: the "top," and "bottom". When there is no longer a complete path from any

"top" to any "bottom" element (where a "path" passes through non-failed elements that share a common face), then this detailed model has failed. Note that this places some restrictions on the *SET_NODE_LIST and element geometry, namely that some "tied" elements exist, and the set of "tied" elements consists of exactly two disjoint subsets.

The specifics of launching a multi-program MPI program are installation dependent. But the idea behind running a coupled model is that you want to run one set of MPI ranks as if you were running a normal MPP-DYNA job, such as:

mpirun –np 4 mppdyna i=**input.k** memory=**200m** p=**pfile**

and a second set with just the command line argument "microscale" (no input file):

mpirun –np 4 mppdyna microscale memory=**100m** p=**pfile**

In practice, an "appfile" file can be created with the following:

–np 4 mppdyna i= input.k

–np 4 mppdyna microscale

Then run the problem via "*mpirun -f appfile*" in Linux operating system. And in windows operating system, the running command is "*mpiexec -configfile appfile*"

The main instance knows to look for the microscale model (because of the presence of the *INCLUDE_MULTISCALE card) and will run the main model. The "microscale" instance will run all the detailed local models. Due to the nature of the coupling, the main model cannot progress when the detailed local models are being processed, nor can the detailed local models run while the main model is being computed. From a processor efficiency standpoint, it therefore makes sense to run as many microscale processes as master processes, and run them on the same CPUs, so that each processing core has one microscale and one master process running on it. But you don't have to – the processes are independent and you can have any number of either.

Example of Co-simulation: Weak Coupling

This example conducts the drop test analysis of a printed circuit board (PCB) and the solder ball failure analysis. In the global model with PCB (*minput.k*), the solder balls are represented by beam elements. In the local model (*solder.k*), detailed solid solder ball are described. The mesoscale solder ball model is coupled concurrently with the macroscale PCB model using the two-scale co-simulation platform.

In the input of global model *(minput.k)*, the coupled beams are defined through beam set 1. The coupled detailed local model ID is 2 and is defined in *solder.k*.

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
*INCLUDE_MULTISCALE
         2      solder ball
solder.k
*DEFINE_MULTISCALE
$       lm ID       beam set
         2              1
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

In the input of local model(*solder.k*), the coupling interface of the global and the local model is defined by a node set 1.

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
*BOUNDARY_COUPLED
         1
$       set id      cpl type        prog
         1              2              1
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

In terms of running this MPI program, an "appfile" file can be created:

```
–np 4 mppdyna i = minput.k

–np 4 mppdyna microscale
```

In Linux operating system, the problem runs via "*mpirun -f appfile*"; In windows operating system,

the running command is "*mpiexec -configfile appfile*".

*INCLUDE_COSIM

Purpose: To define coupling interface across two models in different scales running on MPI based co-simulation. This feature needs the MPP LS-DYNA version R13 and newer versions.

Card Summary:

Card Sets.

Card 1. This card is required.

FILENAME

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	FILENAME							
Type	C							
Default	none							

VARIABLE	DESCRIPTION
FILENAME	Name of an input file that contains the coupling information, which includes sets of segments in global scale model and sets of nodes in local scale one. See the remarks below.

Remarks:

This keyword defines the coupling interface to exchange kinematic and kinetic information between global (large-scale) and local (small-scale) models running on two LS-DYNA MPP jobs. In real application, the global model is usually large-scale structure, e.g. full car model or printed circuit board (PCB), while the local model, e.g. spotwelds, rivets and solders, is relatively much smaller in dimension that leads to smaller mesh and time step size in explicit analysis. To accelerate the computation across global and local scales, users are able to use two input files and run two LS-DYNA MPP jobs simultaneously with different time step sizes. The synchronization of time stepping information across coupling interface is performed automatically at

every time step of global model, while the time step size of local model is adjusted accordingly to guarantee the numerical consistency and accuracy of state variables at the coupling interface in spatial and temporal domain.

1. The coupling interface consists of two sets of entities:

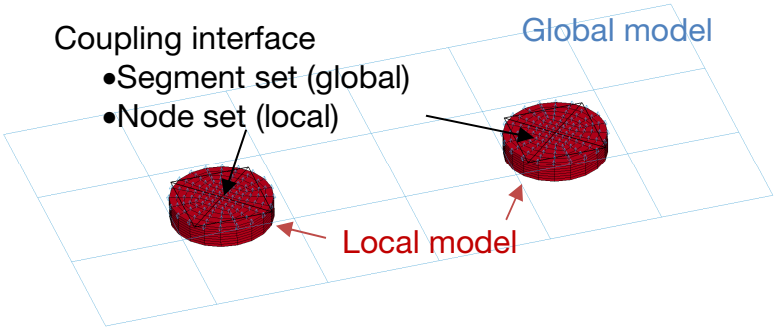
   - One set of segments using *SET_SEGMENT is defined for the global model at the coupling interface, and included in the global input deck using *INCLUDE_COSIM;

   - The other set of nodes using *SET_NODE is defined for the local model at the coupling interface, and included in the local input deck using *INCLUDE_COSIM.

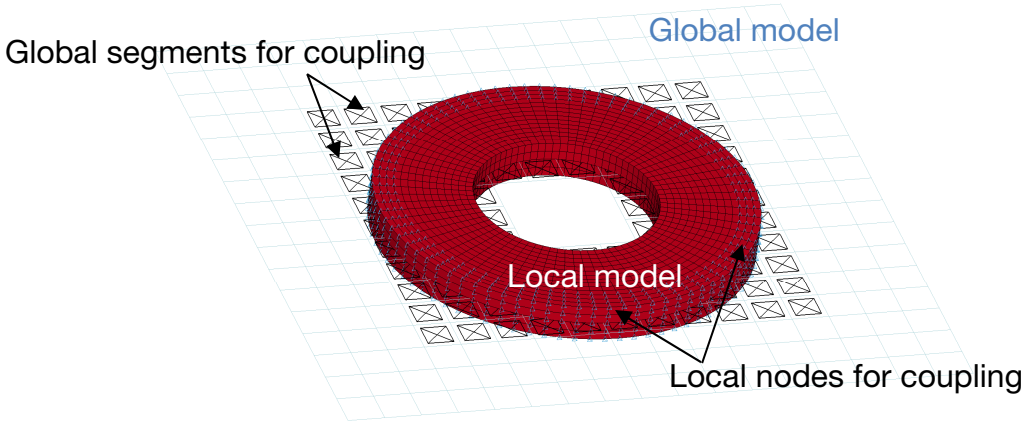
   These two sets defined in global and local included files must have the same set ID (SID) to enable the coupling between global and local jobs. Multiple sets of segments and nodes can be defined in the included files using *INCLUDE_COSIM for different coupling interfaces. Two types of coupling algorithms are currently implemented:

   - Tied contact, where the set of segments in global model drives the motion of the set of nodes in local model by imposing the kinematic constraints on nodal translational DOFs, and the local model returns the constraint forces interpolated onto the nodes of the set of global segments. In this case, it must specify the flag 'ITS' as 1 in both *SET_SEGMENT and *SET_NODE.

   - Solid-in-shell immerse, where the local solid model occupies the same space as global shell model at the coupling interface. The set of nodes in local model follows both the translational and rotational motion of the set of shell segments in global model, while the nodes of the global segments obtain the constraint forces and moments as return. In this case, it must specify the flag 'ITS' as 2 in both *SET_SEGMENT and *SET_NODE.



Tied contact coupling



Solid-in-shell immersed coupling

Figure 1. Two types of coupling supported by the two-scale co-simulation

2. The specifics of running two LS-DYNA MPP jobs are installation dependent. The following is an example of two sets of MPI running command lines:

   - *mpirun -np 12 mppdyna i=global_input.k p=pfile1 jobid=jid1*
   - *mpirun -np 24 mppdyna slave=7 i=local_input.k p=pfile2 jobid=jid2*

   The "slave" flag informs LS-DYNA which job runs the local model. There are quite a few different co-simulation modules based on MPI communication framework currently implemented in LS-DYNA. It must specify slave=7 to enable the two-scale co-simulation. In practice, an "appfile" file can be created with the following lines:

   -np 12 mppdyna i=global.k p=pfile1 jobid=jid1

   -np 24 mppdyna i=local.k slave=7 p=pfile2 jobid=jid2

   Then run the problem via "*mpirun -f appfile*" in Linux operating system. In Windows operating system, the running command is "*mpiexec -configfile appfile*". There is a category named 'Two-scale Cosim' in timing information report of LS-DYNA message file at normal termination, which shows the CPU time spent on the data exchange between two jobs through MPI communication. To achieve the best performance with good load balance between two jobs, the number of MPI processes for two jobs needs to be adjusted according to the time step size difference and other factors, e.g. number of elements and contacts.

3. All the LS-DYNA output and temporary files are separated for two jobs. It is recommended to specify different JOBIDs for global and local job runs. In case that the JOBIDs are the same or there is no JOBID specified, 'cs_' is prefixed to all the files of local job run in order to avoid the conflict with those of global one.

4. It is required to define the same termination time using *CONTROL_TERMINATION in global and local input files to achieve normal termination for both job runs. The time interval of d3plot output defined in both inputs must be the same to have time-consistent postprocessing database.

Example:

global_input.k:

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$  * Main input file for global model
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*KEYWORD
$
*CONTROL_TERMINATION
$   ENDTIM    ENDCYC     DTMIN    ENDENG    ENDMAS     NOSOL
      1e-4
$
*DATABASE_D3PLOT
      1e-6
$
*INCLUDE_COSIM
global_cosim.k
$
$$ Other keyword cards including control, mesh, contact, …
…
$
*END
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

global_cosim.k:

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$  * Define coupling interface in global model
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*KEYWORD
$
*SET_SEGMENT
$      SID       DA1       DA2       DA3       DA4    SOLVER       ITS
        22       0.0       0.0       0.0       0.0MECH               1
$       N1        N2        N3        N4        A1        A2        A3        A4
         8         9        12        11       0.0       0.0       0.0       0.0
        36        37        40        39       0.0       0.0       0.0       0.0
…
$
*SET_SEGMENT
$      SID       DA1       DA2       DA3       DA4    SOLVER       ITS
        11       0.0       0.0       0.0       0.0MECH               1
$       N1        N2        N3        N4        A1        A2        A3        A4
        23        21        25        27       0.0       0.0       0.0       0.0
        47        45        49        51       0.0       0.0       0.0       0.0
…
$
*END
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

local_input.k:

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$  * Main input file for local model
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*KEYWORD
$
*CONTROL_TERMINATION
$   ENDTIM    ENDCYC     DTMIN    ENDENG    ENDMAS     NOSOL
      1e-4
$
*DATABASE_D3PLOT
      1e-6
$
*INCLUDE_COSIM
local_cosim.k
$
$$ Other keyword cards including control, mesh, contact, …
…
$
*END
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

local_cosim.k:

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$  * Define coupling interface in local model
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
*KEYWORD
$
*SET_NODE_LIST
$      SID       DA1       DA2       DA3       DA4    SOLVER       ITS
        11       0.0       0.0       0.0       0.0MECH               1
$       N1        N2        N3        N4        N5        N6        N7        N8
         1         2         3         4         5         6         7         8
         9        10        11        12        13        14        15        16
…
$
*SET_NODE_LIST
$      SID       DA1       DA2       DA3       DA4    SOLVER       ITS
        22       0.0       0.0       0.0       0.0MECH               1
$       N1        N2        N3        N4        N5        N6        N7        N8
       929       930       931       932       933       934       935       936
       937       938       939       940       941       942       943       944
…
$
*END
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

Multiscale Material Analysis

A key to the success of the simulation-based material design and structural analysis is the reliable prediction of material behaviors. In modern engineering designs, materials with complex microstructures (e.g., fiber/particle-reinforced polymers, metallic alloys, laminar composites, etc.) are widely used. Obviously, capturing the multiscale material behaviors is of great importance as material microstructures and microconstituents play a significant role in macroscopic constitutive behaviors.

For microstructure-sensitive materials, phenomenological constitutive models usually involve many parameters to calibrate yet may still suffer from the loss of physics due to model simplifications, especially for nonlinear material behaviors. Multiscale physics-informed material models, on the other hand, are difficult to build using the conventional micromechanical approach. In this chapter, we present the LS-DYNA keywords for two advanced multiscale material modeling methods, which enable high-fidelity micromechanical finite element analysis, mechanistic machine learning-based reduced-order modeling, and accelerated concurrent multiscale simulation of composite structures.

Firstly, we will introduce the high-fidelity Representative Volume Element (RVE) analysis method, which can predict macroscopic material properties for various types of composites from their microstructural information. Using this method, users can conduct virtual testing of numerically re-constructed material samples at their characteristic length scales. The RVE module automatically creates periodic (or linear) displacement boundary conditions, performs FEM-based computational homogenization, and predicts the homogenized macroscopic constitutive responses as well as the detailed microscopic stress/strain fields.

Next, we will present a concurrent multiscale simulation approach for composite structures. In this concurrent multiscale analysis, there is no need to specify the homogenized constitutive laws at the macroscopic material points (e.g., Gauss integration points of a finite element model). Instead, the macroscopic constitutive response is determined through numerical modeling of the local microstructure. If every macroscopic material point is linked to a FEM-based high-fidelity RVE model, however, the resulting computational cost is not affordable for large-scale simulations. To this end, efficient reduced-order models of RVEs are created in LS-DYNA by employing a mechanistic machine learning technique named Deep Material Network (DMN). DMN is trained to learn the microscale material morphologies hidden in composites, and it is capable of predicting macroscopic constitutive behaviors at a computational speed orders-of-magnitude faster than direct high-fidelity FEM simulations. As a multiscale constitutive model, the trained DMN network is coupled to macroscale finite element models in LS-DYNA, enabling highly efficient concurrent multiscale simulations. While DMN can be applied to various types of composite materials, currently this multiscale simulation module has been developed for the injection-molded short-fiber-reinforced composites.

*DATABASE_RVE

Purpose: Output of the RVE homogenization results to the *rveout* file. Double precision SMP/MPP LS-DYNA version R13 and newer versions support this RVE analysis function.

Card Summary:

Card Sets.

Card 1. This card is required.

DT	BINA						

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	DT	BINA						
Type	F	I						
Default	0	0						

VARIABLE	DESCRIPTION
DT	Time interval for the output of RVE homogenization results to the *rveout* file.
BINA	Type of the output file: EQ. 0: ASCII database file named "rveout". EQ. 1: LS-DYNA binary database.

Remarks:

At each output time t, the *rveout* file records the macroscopic material responses, including the macroscopic deformation gradient $\tilde{\boldsymbol{F}}^t$, Green strain $\tilde{\boldsymbol{E}}^t$, Cauchy stress $\tilde{\boldsymbol{\tau}}^t$, and the 1st-Piola-Kirchhoff (PK1) stress

*RVE_ANALYSIS_FEM

Purpose: Predict the macroscopic effective constitutive behaviors of composite materials under the micromechanics-based computational homogenization framework. Composite materials that can be analyzed by this function include but are not limited to fiber-reinforced composites, particulate composites, laminar composites, polycrystalline aggregates, single-phase or multiphase porous media, etc.

The nonlinear computational homogenization theory is adopted to account for both geometrical and material nonlinearities of the RVE (*Representative Volume Element*, also referred to as *Representative Elementary Volume* or *Unit Cell*). Given the material microstructural information (geometry and constitutive properties for base materials), this *RVE_ANALYSIS_FEM keyword enables LS-DYNA to (1) automatically create periodic displacement boundary conditions (or linear displacement boundary conditions) on the RVE finite element mesh; (2) perform nonlinear quasi-static implicit finite element analysis of RVE under user-defined loading conditions; and (3) calculate the macroscopic effective material responses through homogenization of the RVE's microscopic material responses. Meanwhile, detailed distribution and evolution of microscopic stress/strain fields within the RVE are obtained.

Double precision SMP/MPP LS-DYNA version R13 and newer versions support this RVE analysis function.

Card Summary:

Card Sets.

Card 1. This card is required.

MESHFILE

Card 2. This card is required.

INPT	OUPT	LCID	IDOF	BC	IMATCH		

Card 3. This card is required.

H11	H22	H33	H12	H23	H13		

Data Card Definitions:

Card 1	1	2	3	4	5	6	7	8
Variable	MESHFILE							
Type	C							
Default	none							

VARIABLE	DESCRIPTION
MESHFILE	Name of an input file that contains the mesh information (nodal coordinates, element connectivity) of the RVE model. Note that this keyword variable should be of the format "XXX.k", where file name extension ".k" is included. The finite element mesh given in this file is used for the spatial discretization of the material micro-structures, and it does not involve any special 'control nodes' or 'control elements'.

Card 2	1	2	3	4	5	6	7	8
Variable	INPT	OUPT	LCID	IDOF	BC	IMATCH		
Type	I	I	I	I	I	I		
Default	0	1	none	none	0	1		

VARIABLE	DESCRIPTION
INPT	Type of input:

VARIABLE	DESCRIPTION
	EQ.0: RVE boundary conditions are fully defined by two factors: (1) the parameter "BC" of this input card, and (2) the mesh information in the file [MESHFILE]. When running an RVE simulation, LS-DYNA automatically creates a file named "rve_[MESHFILE].k", which contains all the necessary information (e.g., control nodes, displacement constraints, etc.) for boundary condition enforcement.
	EQ.1: Users provide a file named "rve_[MESHFILE].k" to define the boundary condition keywords (e.g. *CONSTRAINED_MULTIPLE_GLOBAL, *BOUNDARY_SPC_NODE, *BOUNDARY_MOTION_NODE, etc.) and control nodes for enforcing RVE boundary conditions. Note that, it is usually non-trivial to manually define all the keywords for RVE boundary conditions. If the file "rve_[MESHFILE].k" is not given when running RVE simulations, then the option *INPT* = 1 will be ignored, and LS-DYNA will create "rve_[MESHFILE].k" based on the parameter "BC" of this input card and the mesh information in the file [MESHFILE].
OUPT	Type of output:
	EQ.1: RVE homogenization results will be written out to a file named "*rveout*". Please refer to the keyword *DATABASE_RVE.
LCID	ID of the loading curve. To perform RVE analysis, a loading curve defined by the keyword *DEFINE_CURVE is required to specify the loading history. There are two columns in the loading curve, where the first column is adopted as a scaling factor for the user-defined macroscopic deformation measure (H11, H22, …, H13, which are defined in CARD3 of this *RVE_ANALYSIS_FEM keyword), and the second column provides the corresponding scaling factor for the loading time (1.0 in the second column denotes the end of the loading).
IDOF	Dimension of the RVE.
	EQ.2: 2D geometry.
	EQ.3: 3D geometry.
BC	Type of the RVE boundary condition:

VARIABLE	DESCRIPTION
	EQ. 0: Periodic Displacement Boundary Condition (PDBC). EQ. 1: Linear Displacement Boundary Condition (LDBC).
IMATCH	Type of the given RVE mesh: EQ. 0: The mesh is non-matching for PDBC. EQ. 1: The mesh is PDBC-matching. This variable is effective only when the user chooses to impose PDBC by setting BC = 0. When the mesh is PDBC-matching, the nodal distributions on the RVE's opposite sides match well with each other. For instance, let us consider two opposite surfaces (surface A, and surface B) that are both perpendicular to the X-axis, for any FEM node on surface A, if we draw a straight line that is parallel to the X-axis, then the intersection point of this line with surface B must also be an FEM node. For such PDBC-matching meshes, an efficient direct nearest neighbor search algorithm can be used for the PDBC imposition, so a PDBC-matching mesh is preferred if users would like to impose the periodic displacement boundary condition for RVE analysis. However, it is not always straightforward to create PDBC-matching meshes for RVE models if very complex material micro-structures exist. In this case, IMATCH= 0 can be chosen to impose PDBC on a non-matching mesh by employing a projection-based constraint imposition method.

Card 3	1	2	3	4	5	6	7	8
Variable	H11	H22	H33	H12	H23	H13		
Type	F	F	F	F	F	F		
Default	none	none	none	none	none	none		

VARIABLE	DESCRIPTION
H*ij*	Component *ij* of the prescribed macroscopic displacement gradient $\widetilde{\boldsymbol{H}} \equiv \nabla_{\tilde{\boldsymbol{X}}}\tilde{\boldsymbol{u}} = \tilde{\boldsymbol{F}} - \boldsymbol{I}$, where $\nabla_{\tilde{\boldsymbol{X}}}$ denotes the gradient operator with respect to the macro-scale, $\tilde{\boldsymbol{u}}$ denotes the macroscopic displacement field, $\tilde{\boldsymbol{F}}$ denotes the macroscopic deformation gradient, and $\boldsymbol{I}$ denotes an identity tensor. By polar decomposition of the deformation gradient, we have $\tilde{\boldsymbol{F}} = \widetilde{\boldsymbol{R}}\widetilde{\boldsymbol{U}}$, where $\widetilde{\boldsymbol{R}}$ represents the macroscopic rigid body rotation, and $\widetilde{\boldsymbol{U}}$ is a symmetric stretch tensor describing the pure material deformation. Since material constitutive behaviors are not affected by macroscopic rigid body rotations, we will consider $\widetilde{\boldsymbol{R}} = \boldsymbol{I}$ in the RVE analysis, and under this condition, both the macroscopic deformation gradient $\tilde{\boldsymbol{F}}$ and macroscopic displacement gradient $\widetilde{\boldsymbol{H}}$ become symmetric. As a result, only six components of $\widetilde{\boldsymbol{H}}$ are needed to prescribe boundary conditions for 3D RVE analysis. As for 2D RVE problems (*IDOF* = 2), the inputs for H33, H23, and H13 are ignored, and only the inputs for H11, H22, and H12 are adopted for enforcing the boundary condition. Note that we should leave the component HIJ (I, J = 1, 2, 3) empty (instead of setting it to be zero) in the input card, if we do not want to impose the corresponding constraints on the RVE. Please refer to Remark 2 and Remark 6.

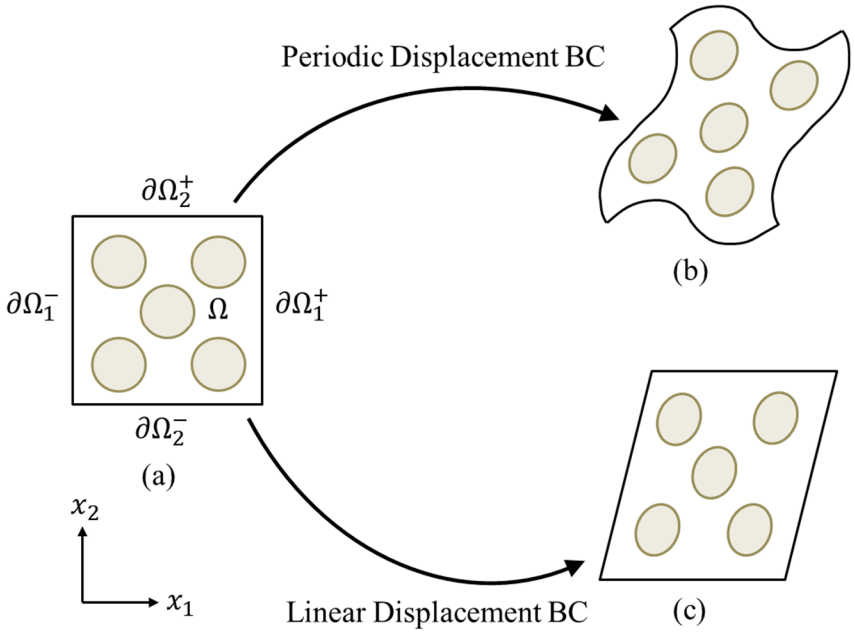


Figure RVE-1. Schematic of the two types of displacement boundary conditions for RVE analysis. (a) The initial configuration of RVE occupying the domain Ω, where $\partial\Omega_\alpha^+$ and $\partial\Omega_\alpha^-$ denote a pair

VARIABLE	DESCRIPTION
	of opposite external boundaries, and $\alpha = 1,2$ for this 2D model. (b) The current configuration of RVE subjected to periodic displacement boundary conditions. (c) The current configuration of RVE subjected to linear displacement boundary conditions.

Remarks:

1. **RVE mesh.** The RVE's mesh information (nodal coordinates, element connectivity) should be provided in a separate input file, of which the file name should be given in Card 1 of the keyword *RVE_ANALYSIS_FEM. Usually, a 3D RVE model has a cuboid shape with its edges parallel to the X-, Y-, and Z- axes of the global coordinate system, respectively, whereas a 2D RVE model is parallelogram-shaped with its edges parallel to the X- and Y- axes of the global coordinate system, respectively. The size of RVE should be large enough so that sufficient statistical microstructural information (fibers, particles, voids, grains, etc.) of composite material is included. In the meanwhile, the RVE should remain small enough to be considered as a volume element of continuum mechanics. Users can apply standard mesh generation software (e.g., *LS-PrePost*, etc.) or specialized RVE reconstruction tools to prepare the finite element mesh for the composite material of interest, and store the nodal coordinates and element connectivity data in the input mesh file. For a 3D RVE model, this mesh file should contain the keywords *ELEMENT_SOLID and *NODE, whereas for a 2D RVE model, the keywords *ELEMENT_SHELL and *NODE should be included.

2. **Boundary conditions.** Given the macroscopic displacement gradient $\widetilde{\boldsymbol{H}}$ and the input parameter $INPT = 0$, LS-DYNA can automatically create boundary conditions on the external boundary $\partial\Omega$ of the RVE. Two types of RVE boundary conditions can be created, depending on the user's input for the parameter BC:

   If $BC = 0$, the following PDBC (*periodic displacement boundary condition*, or also referred to as *periodic boundary condition*) is imposed:

$$\boldsymbol{w}_\alpha^+ - \boldsymbol{w}_\alpha^- = \widetilde{\boldsymbol{H}}(\boldsymbol{X}_\alpha^+ - \boldsymbol{X}_\alpha^-),$$

   where $\boldsymbol{X}_\alpha^+ \in \partial\Omega_\alpha^+$ and $\boldsymbol{X}_\alpha^- \in \partial\Omega_\alpha^-$ denote the micro-scale material points located on a pair of opposite external boundaries $\partial\Omega_\alpha^+$ and $\partial\Omega_\alpha^-$, respectively, in the RVE's initial configuration (refer to **Figure RVE-1** for a 2D illustration), $\boldsymbol{w}_\alpha^+$ and $\boldsymbol{w}_\alpha^-$ denote the micro-scale displacements of the material points $\boldsymbol{X}_\alpha^+$ and $\boldsymbol{X}_\alpha^-$, respectively, and the subscript $\alpha = 1,2\ldots,d$, in which $d = 2$ for 2D RVE models, and $d = 3$ for 3D RVE models. To impose the periodicity constraint, a control point-based method is implemented, which is similar to the keywords *INCLUDE_UNITCELL and *CONSTRAINED_MULTIPLE_GLOBAL, but *RVE_ANALYSIS_FEM saves the users' efforts in manually defining the control

points/elements or constrained degrees of freedom. By simply setting the parameter $BC = 0$in the keyword *RVE_ANALYSIS_FEM, LS-DYNA will automatically prepare control points for the enforcement of PDBC.

If $BC = 1$, the following LDBC (*linear displacement boundary condition*) will be imposed:

$$\boldsymbol{w}_\alpha = \widetilde{\boldsymbol{H}}\boldsymbol{X}_\alpha,$$

where $\boldsymbol{X}_\alpha \in \partial\Omega_\alpha$ denotes any micro-scale material point located on the external boundaries $\partial\Omega_\alpha = \partial\Omega_\alpha^+ \cup \partial\Omega_\alpha^-$ of the RVE. Note that the RVE's complete external boundary can be expressed as $\partial\Omega = \bigcup_{\alpha=1}^{d} \partial\Omega_\alpha$, in which $d = 2$ for 2D RVE models, and $d = 3$ for 3D RVE models. It is noteworthy to mention that RVEs with LDBC usually appear to be stiffer than RVEs with PDBC. When the size of RVE is large enough, however, the influence of different types of boundary conditions on the homogenized material properties becomes negligibly small.

The assignment of zero value to any component of the macroscopic displacement gradient $\widetilde{\boldsymbol{H}}$ will indicate the imposition of the boundary constraint based on the above constraint equations. If we do not want to impose constraints in certain directions (i.e., allow the associated RVE boundaries deform freely), then we should leave the corresponding component of $\widetilde{\boldsymbol{H}}$ empty in the input card. Please refer to Remark 6 for an application scenario where most components of $\widetilde{\boldsymbol{H}}$ should be set empty.

3. **Constitutive models for base materials.** Depending on the actual morphology of material microstructures, an RVE finite element model can consist of many parts, and each part can be assigned a unique constitutive law to describe the behaviors of the corresponding base material (e.g., fiber, particle, matrix, grain, etc.). Any constitutive model (linear/nonlinear, isotropic/anisotropic, etc.) can be selected for the base materials, as long as the model is supported by LS-DYNA implicit finite element simulation.

4. **Accuracy control for the implicit calculation.** To ensure high accuracy of the nonlinear implicit finite element simulation, the 2nd-order objective stress update scheme is recommended, which is specified as $OSU = 1$ in the keyword *CONTROL_ACCURACY. Users are also suggested to specify other control parameters for the implicit solver in relevant keywords, such as *CONTROL_IMPLICIT_GENERAL, *CONTROL_IMPLICIT_SOLUTION, and *CONTROL_IMPLICIT_SOLVER.

5. **Homogenization results.** LS-DYNA writes out the homogenized material responses to a file named "*rveout*", in which the macroscopic deformation gradient $\tilde{\boldsymbol{F}}^t$, Green strain $\tilde{\boldsymbol{E}}^t$, Cauchy stress $\tilde{\boldsymbol{\tau}}^t$, and the 1st-Piola-Kirchhoff (PK1) stress $\widetilde{\boldsymbol{P}}^t$, are recorded at each output time t. Using these results, different measures for

the macro-scale stress and deformation can be obtained based on their relationships with the variables given in the *rveout* file. For instance, the 2nd-Piola-Kirchhoff (PK2) stress $\tilde{\boldsymbol{S}}$ is defined as $\tilde{\boldsymbol{S}} = \tilde{\boldsymbol{P}}\tilde{\boldsymbol{F}}^{-T}$, which can be easily calculated since $\tilde{\boldsymbol{P}}$ and $\tilde{\boldsymbol{F}}$ are available in the file *rveout*, which can be read using any text editor or Python/MATLAB script, etc.

Note that, as we discussed in the definition of Card 3, the macro-scale rigid body rotations are excluded from the macroscopic displacement gradient. Accordingly, the macroscopic deformation gradient $\tilde{\boldsymbol{F}}^t$and PK1 stress $\tilde{\boldsymbol{P}}^t$ obtained from RVE analysis are both symmetric, so only six components of each symmetric tensors are written out to the *rveout* file.

6. **Degeneration to small strain analysis**. Although a nonlinear computational homogenization formulation is implemented in LS-DYNA to capture the RVE's finite deformation effects, the homogenized quantities will be close to those calculated by small strain homogenization theories when the actual macroscopic deformation specified by users is small. In this situation, the macroscopic Green strain $\tilde{\boldsymbol{E}}$ given in the *rveout* file will be approximately equal to the infinitesimal strain, and different stress measures (PK1 stress $\tilde{\boldsymbol{P}}$, PK2 stress $\tilde{\boldsymbol{S}}$, Cauchy stress $\tilde{\boldsymbol{\tau}}$) will have an identical magnitude.

For small strain linear analysis, if we conduct 3D RVE simulations with six orthogonal loading conditions (e.g., three uniaxial tensile loadings and three pure shear loadings), respectively, then we will obtain the full macroscopic elasticity tensor for the composite material. Recall that the macro-scale linear elastic constitutive relationship can be expressed as follows:

$$\tilde{\boldsymbol{\varepsilon}} = \tilde{\boldsymbol{C}}\tilde{\boldsymbol{\sigma}},$$

where $\tilde{\boldsymbol{C}}$ is a 6×6 macroscopic material compliance matrix:

$$\tilde{\boldsymbol{C}} = \begin{bmatrix} \tilde{C}_{11} & \tilde{C}_{12} & \tilde{C}_{13} & \tilde{C}_{14} & \tilde{C}_{15} & \tilde{C}_{16} \\ \tilde{C}_{21} & \tilde{C}_{22} & \tilde{C}_{23} & \tilde{C}_{24} & \tilde{C}_{25} & \tilde{C}_{26} \\ \tilde{C}_{31} & \tilde{C}_{32} & \tilde{C}_{33} & \tilde{C}_{34} & \tilde{C}_{35} & \tilde{C}_{36} \\ \tilde{C}_{41} & \tilde{C}_{42} & \tilde{C}_{43} & \tilde{C}_{44} & \tilde{C}_{45} & \tilde{C}_{46} \\ \tilde{C}_{51} & \tilde{C}_{52} & \tilde{C}_{53} & \tilde{C}_{54} & \tilde{C}_{55} & \tilde{C}_{56} \\ \tilde{C}_{61} & \tilde{C}_{62} & \tilde{C}_{63} & \tilde{C}_{64} & \tilde{C}_{65} & \tilde{C}_{66} \end{bmatrix},$$

and the vectors $\tilde{\boldsymbol{\varepsilon}} = [\tilde{\varepsilon}_{11} \;\; \tilde{\varepsilon}_{22} \;\; \tilde{\varepsilon}_{33} \;\; \tilde{\varepsilon}_{12} \;\; \tilde{\varepsilon}_{23} \;\; \tilde{\varepsilon}_{31}]^T$ and $\tilde{\boldsymbol{\sigma}} = [\tilde{\sigma}_{11} \;\; \tilde{\sigma}_{22} \;\; \tilde{\sigma}_{33} \;\; \tilde{\sigma}_{12} \;\; \tilde{\sigma}_{23} \;\; \tilde{\sigma}_{31}]^T$ contain six macroscopic strain and stress components, respectively.

If we specify a uniaxial tensile loading condition ($H11 = \epsilon$, where ϵ is a non-zero small number, and all other components of the macroscopic displacement gradient are left empty) in Card 3 of the keyword *RVE_ANALYSIS_FEM, then the

finite element simulation of RVE will yield a homogenized stress vector $\tilde{\sigma} = [\tilde{\sigma}_1 \quad 0.0 \quad 0.0 \quad 0.0 \quad 0.0 \quad 0.0]^T$ and a homogenized strain vector $\tilde{\varepsilon} = [\tilde{\varepsilon}_1 \quad \tilde{\varepsilon}_2 \quad \tilde{\varepsilon}_3 \quad \tilde{\varepsilon}_4 \quad \tilde{\varepsilon}_5 \quad \tilde{\varepsilon}_6]^T$, which are available in the *rveout* file. We can then calculate the first column of the macroscopic material compliance matrix as follows:

$$\tilde{C}_{11} = \tilde{\varepsilon}_1/\tilde{\sigma}_1, \tilde{C}_{21} = \tilde{\varepsilon}_2/\tilde{\sigma}_1, \tilde{C}_{31} = \tilde{\varepsilon}_3/\tilde{\sigma}_1,$$

$$\tilde{C}_{41} = \tilde{\varepsilon}_4/\tilde{\sigma}_1, \tilde{C}_{51} = \tilde{\varepsilon}_5/\tilde{\sigma}_1, \tilde{C}_{61} = \tilde{\varepsilon}_6/\tilde{\sigma}_1.$$

Similarly, all the other macroscopic compliance coefficients can be computed by applying different uniaxial tensile and pure shear loading conditions. By inverting the compliance matrix $\tilde{C}$, we can obtain the macroscopic material stiffness matrix.

7. **Calibration of macroscale constitutive laws.** If a functional form of the macroscopic constitutive equation is available, then a series of numerical material tests can be properly designed and conducted on the RVE model to identify the material parameters for the assumed constitutive model. In other words, by treating RVEs as virtual material samples, we can fit macro-scale material model parameters while reducing the amount of expensive (or unfeasible) physical experiments for composite materials.

8. **De-homogenization/Localization analysis.** After performing a standard macro-scale structural finite element analysis in LS-DYNA, we can obtain the macro-scale deformation history at any element or integration point of the macro-structure. If we convert such information to the macroscopic displacement gradient $\tilde{\boldsymbol{H}} \equiv \nabla_X \tilde{\boldsymbol{u}}$, we can then apply this *RVE_ANALYSIS_FEM keyword to perform the RVE localization analysis (also called de-homogenization analysis) in order to evaluate the actual micro-scale material responses, including the evolution and distribution of microscopic stress/strain fields within the heterogeneous RVE. These detailed microscale material responses are available in the standard LS-DYNA output files (e.g., d3plot). The multiscale simulation can provide useful guidance to both microscopic material design and macroscopic structural analysis.

Example:

Main_RVE_input.k:

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$$  *RVE_ANALYSIS_FEM
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$  * This example presents a 3D RVE model of a composite material
$  * The composite material consists of two base materials
```

```
$  * Finite element mesh is provided in a separate file named "Mesh_RVE.k"
$  * PDBC with a macroscopic displacement gradient H12=0.001 is specified
$  * Macroscopic results are written out to the "rveout" file
$  * Microscopic results are written out to the "d3plot" file
$
*KEYWORD
$
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
*CONTROL_TERMINATION
$   ENDTIM    ENDCYC     DTMIN    ENDENG    ENDMAS     NOSOL
      1e-4
*CONTROL_ACCURACY
$      OSU       INN
         1         1
*CONTROL_IMPLICIT_GENERAL
$   IMFLAG       DT0
         1      1e-4
*CONTROL_IMPLICIT_SOLUTION
$   NSOLVR    ILIMIT    MAXREF     DCTOL     ECTOL     RCTOL     LSTOL    ABSTOL
                                     1e-6     1e-10                         1e-30
*CONTROL_IMPLICIT_SOLVER
$   LSOLVR    LPRINT     NEGEV     ORDER     DRCM     DRCPRM   AUTOSPC   AUTOTOL
         6         0
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
*RVE_ANALYSIS_FEM
$ MESHFILE
Mesh_RVE.k
$     INPT      OUPT      LCID      IDOF        BC    IMATCH
         0         1         1         3         0         0
$      H11       H22       H33       H12       H23       H13
                                     1e-3
$...>....1....>....2....>....3....>....4....>....5....>....6....>....7....>....8
*DEFINE_CURVE
         1
       0.0       0.0
      1e-4       1.0
*DATABASE_RVE
      1e-4
$---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
*part
         1         6        21
*part
         2         6        22
$---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
*MAT_ELASTIC
        21      1e-3      10.2       0.3
*MAT_ELASTIC
        22      1e-3       5.8       0.4
$---+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
*section_solid
$#   secid    elform
         6        10
$
*END
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

Mesh_RVE.k:

```
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$$$$  This is the mesh file accompanying *RVE_ANALYSIS_FEM
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$
$  * The finite element mesh represents the composite material micro-structures.
```

```
$  * For 3D models, this file should contain keywords *ELEMENT_SOLID & *NODE
$  * For 2D models, this file should contain keywords *ELEMENT_SHELL & *NODE
$  * Only a reduced input is given below
$
*KEYWORD
$
*NODE
…

*ELEMENT_SOLID
…

$
*END
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

References

1Wu CT, Koishi M, Hu W. A displacement smoothing induced strain gradient stabilization for the meshfree Galerkin nodal integration method. *Comput. Mech.* **56**, 19-37, 2015.

2Wu CT, Chi SW, Koishi M, Wu Y. Strain gradient stabilization with dual stress points for the meshfree nodal integration method in inelastic analysis. *Int. J. Numer. Methods Engrg*. **107**, 3-30, 2016.

3Wu CT, Bui TQ, Wu Y, Luo TL, Wang M, Liao CC, Chen PY, Lai YS. Numerical and experimental validation of a particle Galerkin method for metal grinding simulation. *Comput. Mech.* **61**, 365-383, 2018.

4Wu CT, Wu Y, Crawford JE, Magallanes JM. Three-dimensional concrete impact and penetration simulations using the smoothed particle Galerkin method. *Int. J. Impact Engrg.* **106**, 1-17, 2017.

5Wu Y, Wu CT. Simulation of Impact Penetration and Perforation of Metal Targets Using the Smoothed Particle Galerkin Method. *Journal of Engineering Mechanics*, 144(8), 04018057, 2018.

6Wu CT, Wu Y, Liu Z, Wang D. A stable and convergent Lagrangian particle method with multiple nodal stress points for large strain and material failure analyses in manufacturing processes. *Finite Elements in Analysis and Design*, 146, 96-106, 2018.

7Pan X, Wu CT, Hu W, Wu Y. A momentum-consistent stabilization algorithm for Lagrangian particle methods in the thermo-mechanical friction drilling analysis. *Comput. Mech.* 64, 625–644, 2019.

8Hayashi S, Wu CT, Hu W, Wu Y, Pan X, Chen H. New Methods for Compression Molding Simulation and Component Strength Validation for Long Carbon Fiber Reinforced Thermoplastics. *12th European LS-DYNA Conference*, Koblenz, Germany, May 2019.

9Huang L, Wu Y, Huff G, Huang S, Ilinich A, Freis A, Luchey G. Simulation of Self-Piercing Rivet Insertion Using Smoothed Particle Galerkin Method. *15th International LS-DYNA Users Conference*, Detroit, MI, June, 2018

10Wu CT, Wu Y, Lyu D, Pan X, Hu W. The momentum-consistent smoothed particle Galerkin (MC-SPG) method for simulating the extreme thread forming in the flow drill screw-driving process. *Comp. Part. Mech.*, 7, 177-191, 2020.

11Wu CT, Ma N, Takada K, Okada H. A meshfree continuous–discontinuous approach for the ductile fracture modeling in explicit dynamics analysis, Computational Mechanics, 58 (2016) 391-409

12Wu CT, Wu Y, Koishi M. A strain-morphed nonlocal meshfree method for the regularized particle simulation of elastic-damage induced strain localization problems, Computational Mechanics, 56 (2015) 1039-1054

REFERENCES