*ALE_STRUCTURED_MESH_VOLUME_FILLING

Purpose: Perform volume filling operations on a structured ALE mesh generated by the *ALE_STRUCTURED_MESH keyword card.

In a typical Structured ALE (S-ALE) simulation, *ALE_STRUCTURED_MESH is used first to construct the S-ALE mesh, which includes a set of hex solid elements and their nodes. Secondly, “material parts” which contains all ALE material properties need to be defined and listed under *ALE_MULTI-MATERIAL_GROUP card. Those “material part” definitions are used as wrappers to contain material properties (*MAT_+*EOS_+*HOURGLASS_). As the name suggests, “material parts” have no mesh.

Now we have the S-ALE mesh which defines our domain of ALE simulation, and the materials flowing inside. But we have no idea of how those materials occupy the domain. There are two ways to do it. The first is to use *INITIAL_VOLUME_FRACTION and list element one by one and then for each AMMG in each element, its volume fraction. This approach is tedious and impractical, therefore rarely used. The second way is to do volume filling automatically, based on user-provided instructions. *ALE_STRUCTURED_MESH_VOLUME_FILLING card is used to set up these instructions.

Each card represents an instruction; and those instructions are executed consecutively, based on the order of appearance in the input file. Each instruction is to fill the volume either inside or outside of a geometry type, with certain AMMG. Those geometry types are: ALL, PARTSET, PART, SEGSET, PLANE, CYLINDER, BOX defined by coordinates (BOXCOR), BOX defined by S-ALE mesh indices (BOXCPT) and ELLIPSOID.

Each *ALE_STRUCTURED_MESH_VOLUME_FILLING keyword contains two cards. Multiple *ALE_STRUCTURED_MESH_VOLUME_FILLING keywords can be defined and executed in the order of appearance. Please see the examples below.

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

Card 2	1	2	3	4	5	6	7	8
Variable	GEOM	IN/OUT	E1	E2	E3	E4	E5	
Type	A	I	I or F	I or F	I or F	I or F	I or F	
Default	none	0	none	none	None	none	none	

VARIABLE	DESCRIPTION
MSHID	S-ALE Mesh ID. A unique number must be specified.
AMMGTO	The ID of AMMG filling the geometry. Please see *ALE_MULTI-MATERIAL_GROUP for reference.
NSAMPLE	Number of sampling points. In case of an element is partially filled, in each direction, 2*NSAMPLE+1 points are generated. These (2*NSAMPLE+1)^3 points, each representing a volume, are used to determine if its volume is in or out.
VID	ID of *DEFINE_VECTOR card. This flag is used to assign initial velocity to material filling the domain. Field 2 to 5 (XT,YT,ZT) of the *DEFINE_VECTOR card are used to define the initial translational velocities. Please refer to Example 1 below for usage.
GEOM	Geometry types. They are: PARTSET, PART, SEGSET, PLANE, CYLINDER, BOXCOR, BOXCPT and SPHERE. See the table below for more details.
IN/OUT	To fill inside or outside. For PARTSET/PART/SEGSET options, inside are taken as the normal direction of their containing segments. EQ.0: inside (default) EQ.1: outside
E1, E2, E3, E4, E5	These values have different definitions for different options. See the table below for details.

The “GEOM” column in the table below enumerates the allowed values for the geometry variable as well as describing E1, …, E5 for each geometry type. Each of the following operations accepts up to 5 arguments but may take fewer. Values of E*n* left unspecified are ignored.

OPTION	DESCRIPTION
ALL	Fill all volume inside the mesh. No additional variables needed.
PARTSET	Geometry defined by PARTSET. E1 is the shell part set ID. E2 is the offset distance.
PART	Geometry defined by PART. E1 is the shell part ID. E2 is the offset distance.
SEGSET	Geometry defined by SEGSET. E1 is the segment set ID. E2 is the offset distance.
PLANE	Geometry defined by PLANE. E1 is the node ID of a node on the plane. E2 is another node ID off the plane. And vector E2 – E1 is normal to the plane.
CYLINDER	Geometry defined by CYLINDER. E1, E2 are node IDs of the center nodes at two ends. E3, E4 are the radii at those two ends.
BOXCOR	Geometry defined by BOX. E1 is the BOX ID. Please refer to *DEFINE_BOX for details on setting up a box in global coordinate system or *DEFINE_BOX_LOCAL in local coordinate system.
BOXCPT	Geometry defined by BOX. The box is defined using S-ALE control points (CPT). E1 is BOX ID. Please refer to *DEFINE_BOX for details on setting up a box.
ELLIPSOID	Geometry defined by ELLIPSOID. E1 is the node ID of the ellipsoid center node. E2, E3, E4 are the radii along x, y, z directions. E5 is the local coordinate system ID if a local coordinate system is used. Please refer to *DEFINE_COORDINATE_SYSTEM.

Examples:

1. This example uses two *ALE_STRUCTURE_VOLUME_FILLING cards. The first fills all volume in a mesh with AMMG 1. The second fills AMMG2 in a spherical domain. We use the same mesh as the example in the *ALE_STRUCTURED_MESH card manual page.

```
*ALE_STRUCTURED_MESH
$    mshid       pid      nbid      ebid
         1         1    200001    200001
$     nptx      npty      nptz
      1001      1001      1001
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid               ammgto              nsample                           vid
         1                    1
```

```
$     geom
       ALL
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                          vid
         1                   2
$     geom    in/out       node         rx         ry         rz     lcsid
 ELLIPSOID                            0.03                             234
*DEFINE_COORDINATE_NODES
$      cid      nid1      nid2      nid3      flag
       234         2         3         4         1
*NODE
       1   0.0000000e+00   0.0000000e+00   0.0000000e+00
       2   0.0000000e+00   0.0000000e+00   0.0000000e+00
       3   0.1000000e+00   0.0000000e+00   0.0000000e+00
       4   0.0000000e+00   0.1000000e+00   0.0000000e+00
       5   0.1000000e+00   0.1000000e+00   0.1000000e+00
*END
```

Or we could fill every elements with AMMG2 and then switch domain outside of sphere to AMMG1 as follows:

```
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                          vid
         1                   2
$     geom
       ALL
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                          vid
         1                   1
$     geom    in/out       node         rx         ry         rz     lcsid
 ELLIPSOID         1                  0.03                             234
```

If we want to give the sphere filled in by AMMG2 an initial velocity, simply define a *DEFINE_VECTOR card as below and assign its ID to field VID.

```
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                          vid
         1                   2                                               1
$     geom    in/out       node         rx         ry         rz     lcsid
 ELLIPSOID                            0.03                             234
*DEFINE_VECTOR
$      vid        xt        yt        zt
         1      100.      -20.       0.0
```

2. In this model, we fill the whole mesh with AMMG4 first. And then fill AMMG1, AMMG2 and AMMG3 into 3 containers, each defined by a LAG shell part. 4 cards are needed to do that.

```
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                          vid
         1                   4
$     geom
       ALL
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                          vid
         1                   1
```

```
$     geom    in/out       pid    offset
      PART               2001
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                           vid
         1                   2
$     geom    in/out       pid    offset
      PART               2002
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                           vid
         1                   3
$     geom    in/out       pid    offset
      PART               2003
```

The above filling by PART cards assume shell normal pointing inward to the container. If the shell normal points outwards, we need to assign the in/out value to 1. Assuming shell part 2003's normal is pointing outwards, the card need to be set as:

```
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                           vid
         1                   3
$     geom    in/out       pid    offset
      PART         1      2003
```

3. To use BOXCPT, we define a box using S-ALE control point indices.

```
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                           vid
         1                   1
$     geom
       ALL
*ALE_STRUCTURED_MESH_VOLUME_FILLING
$    mshid              ammgto            nsample                           vid
         1                   2
$     geom    in/out     boxid
    BOXCPT                   1
*DEFINE_BOX
$    boxid       xmn       xmx       ymn       ymx       zmn       zmx
         1         8        15         8        15         8        15
*END
```