*ALE_STRUCTURED_MESH_CONTROL_POINTS

Purpose: The purpose of this keyword is to provide spacing information used by the *ALE_STRUCTURED_MESH keyword to generate a 3D structured ALE mesh.

Each instance of the *ALE_STRCUTURED_MESH_CONTROL_POINTS card defines a one-dimensional mesh using control. Each control point consists of a node number (see N, card 2) and of a coordinate (see X, card 2). The first control point *must* be node 1, and the node number of the last point defines the total number of nodes. Between and two control points the mesh is uniform. The *ALE_STRUCTURED_MESH card, in turn, defines a simple three dimensional mesh from the triple product of three *ALE_STRUCTURED_MESH_CONTROL_POINT one-dimensional meshes.

Card 1	1	2	3	4	5	6	7	8
Variable	CPID	Not used	Not used	SFO	Not used	OFFO		
Type	I			F		F		
Default	None			1.		0.		

Point Cards. Put one pair of points per card (2E20.0). Input is terminated at the next keyword ("*") card. *At least two cards are required, one of which, having* N = 1 *is required.*

Card 2	1	2	3	4	5	6	7	8
Variable	N		X		RATIO			
Type	I20		E20.0		E20.0			
Default	none		none		0.0			

VARIABLE	DESCRIPTION
CPID	Control Points ID. A unique number must be specified. This ID is to be referred in the three fields marked up CPIDX, CPIDY, CPIDZ in *ALE_STRUCTURED_MESH.
SFO	Scale factor for ordinate value. This is useful for simple modifications. EQ.0.0: default set to 1.0.

VARIABLE	DESCRIPTION
OFFO	Offset for ordinate values. See Remark 1.
N	Control point node number.
X	Control point position.
RATIO	Ratio for progressive mesh spacing. Progressively larger or smaller mesh will be generated between the control point that has nonzero ratio specified and the control point following it. See remark 2. GT.0.0: mesh size increases; $dl_{n+1} = dl_n * (1 + ratio)$ LT.0.0: mesh size decreases; $dl_{n+1} = dl_n/(1 - ratio)$

Remarks:

1. **Coordinates scaling.** The ordinate values are scaled after the offsets are applied, i.e.:

$$\text{Ordinate value} = \text{SFO} \times (\text{Defined value} + \text{OFFO})$$

2. **Progressive mesh spacing.** The formula used to calculate element length is as follows: $dl_{base} = |x_{end} - x_{start}| * (factor - 1)/(factor^n - 1)$ in which dl_{base} is the smallest base length; x_{start} and x_{end} are the coordinate of the start and end point respectively; $factor = 1 + ratio$ $(ratio > 0)$ or $1/(1 - ratio)$ $(ratio < 0)$; and n is number of elements. Please note here element size either increases by $ratio$ $(ratio > 0)$ or decreases by $-ratio/(1 - ratio)$ $(ratio < 0)$ each time. But the overall effect is the same: starting from the smallest element, each time the element size is increased by $|ratio|$.

Example:

1. This example below generates a regular box mesh. Its size is 0.2 by 0.2 by 0.2. It is generated in a local coordinate system defined by three nodes 2, 3, 4 and originates from node 1.

The local x-axis mesh spacing is defined by control points ID 1001. It has 21 nodes evenly distributed from 0.0 to 0.2. The local y-axis is defined by ID 1002 and has twice the elements of 1001. It has 41 nodes evenly distributed from 0.0 to 0.2. The local z-axis is defined by ID 1003. It has 31 nodes and covers from 0.0 to 0.2. The mesh is two times finer in the region between node 6 and node 26.

```
*ALE_STRUCTURED_MESH
$    mshid      dpid      nbid      ebid
         1         1    200001    200001
$    cpidx     cpidy     cpidz      nid0     lcsid
      1001      1002      1003         1       234
```

```
*DEFINE_COORDINATE_NODES
$      cid      nid1      nid2      nid3
       234         2         3         4
*ALE_STRUCTURED_MESH_CONTROL_POINTS
      1001
$                x1                  x2
                  1                  .0
                 21                  .2
*ALE_STRUCTURED_MESH_CONTROL_POINTS
      1002
$                x1                  x2
                  1                  .0
                 41                  .2
*ALE_STRUCTURED_MESH_CONTROL_POINTS
      1003
$                x1                  x2
                  1                  .0
                  6                 .05
                 26                 .15
                 31                  .2
*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
*END
```

2. This example shows how to generate a progressive larger/smaller mesh spacing. The mesh geometry is the same as the example above. At x-direction the mesh is progressively smaller between node 1 and 8. For these 7 elements, each element is 0.1/1.1=9.09% smaller than its left neighbor. Between node 15 and node 22, the mesh is progressively larger by 10% each time in those 7 elements. The 7 elements in the middle between node 8 and 15 are of equal length.

```
*ALE_STRUCTURED_MESH
$    mshid      dpid      nbid      ebid
         1         1    200001    200001
$    cpidx     cpidy     cpidz      nid0     lcsid
      1001      1002      1003         1       234
*DEFINE_COORDINATE_NODES
$      cid      nid1      nid2      nid3
       234         2         3         4
*ALE_STRUCTURED_MESH_CONTROL_POINTS
      1001
$                x1                  x2                 ratio
                  1                  .0                  -0.1
                  8          0.06666667
                 15          0.13333333                   0.1
                 22                  .2
```