

**\*ALE\_STRUCTURED\_MESH\_CONTROL\_POINTS**

Purpose: Provide spacing information used by the \*ALE\_STRUCTURED\_MESH keyword to generate a 3D structured ALE mesh.

Each instance of the \*ALE\_STRUCTURED\_MESH\_CONTROL\_POINTS card defines a one-dimensional mesh using control points. Each control point consists of a node number (see N in Card 2) and a coordinate (see X in 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 any 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		ICASE	SFO		OFFO		
Type	I		I	F		F		
Default	none		0	1.		0.		

**Point Cards.** Put one pair of points per card. 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	I		F		F				
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.
ICASE	A flag to trigger special logics for a more user-friendly input format for progressive mesh spacing. Please see examples sections below on ICASE usage.

<b>VARIABLE</b>	<b>DESCRIPTION</b>
SFO	Scale factor for ordinate value. This is useful for simple modifications. EQ.0.0: default set to 1.0.
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 Remarks 2 and 3. GT.0.0: mesh size increases $dl_{n+1} = dl_n \times (1 + \text{RATIO})$ LT.0.0: mesh size decreases $dl_{n+1} = dl_n / (1 - \text{RATIO})$

**Remarks:**

1. **Coordinates Scaling.** The ordinate values are scaled after the offsets are applied, that is:

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

2. **Progressive Mesh Spacing.** The formula used to calculate element length is as follows:

$$dl_{\text{base}} = |x_{\text{end}} - x_{\text{start}}| \times (\text{factor} - 1) / (\text{factor}^n - 1)$$

in which  $dl_{\text{base}}$  is the smallest base length;  $x_{\text{start}}$  and  $x_{\text{end}}$  are the coordinates of the start and end points, respectively;  $\text{factor} = 1 + \text{RATIO}$  if  $\text{RATIO} > 0$  or  $1/(1 - \text{RATIO})$  if  $\text{RATIO} < 0$ ; and  $n$  is number of elements. Please note here element size either increases by  $\text{RATIO}$  if  $\text{RATIO} > 0$  or decreases by  $-\text{RATIO}/(1 - \text{RATIO})$  if  $\text{RATIO} < 0$  each time. But the overall effect is the same; that is, starting from the smallest element, each time the element size is increased by  $|\text{RATIO}|$ .

3. **Transition Meshes.** If the mesh has different mesh sizes and transition zones between them (where the mesh size progressively increases or decreases), then the ratios and number of elements for these zones are reported in the `messag` file (look for messages `KEY+1174`). Updated lists of control points are output in the `messag` file for each `*ALE_STRUCTURED_MESH_CONTROL_POINTS` with more than 3 control points.

**Examples:**

1. The 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 nodes 2, 3, and 4 that 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 number of 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 progressively larger/smaller mesh spacing. The mesh geometry is the same as the example above. In the  $x$ -direction the mesh becomes progressively smaller between nodes 1 and 8. For these 7 elements, each element is  $0.1/1.1=9.09\%$  smaller than its left neighbor. Between nodes 15 and node 22, the elements become progressively larger by 10% in comparison to each element's left neighbor. 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

```

3. **ICASE=1.** Say we want to use progressive mesh spacing at certain region. We know the number of elements and the overall length. And we have a “base” element size in mind. However, the generic setup requires “ratio”, which needs an iterative solution of a nth order polynomial equation based on the “base” element size.

ICASE=1 aims to relieve users from this burden. When it is activated, we could input the “base” element size and the value of “ratio” would be internally calculated and then used to generate the S-ALE mesh. The following examples show how this option should be used.

First, to make things clear, we introduce the concept of “region”. A region is defined by 2 consecutive entries in the \*ALE\_STRUCTURED\_MESH\_CONTROL\_POINTS card. For example, the Control\_Points 1001 below has three regions. The first region spans [0, 0.0755] ; the second one [0.0755, 0.1245]; the third one [0.1245,0.2].

The “ratio” field in generic setup, is now marked at “x1”. It is to specify the element size at that location. The example below shows:

- (1) Region 1 is of progressive spacing, and the preferred element size at the 8<sup>th</sup> node is 0.0070.
- (2) In region 2, mesh is evenly distributed.
- (3) Region 3 is of progressive spacing, and the preferred element size at the 8<sup>th</sup> node is 0.0070.

The rule is simple, a region could have totally 0, 1, 2 “x1” specified at its two ends. 1 occurrence means progressive mesh spacing specified. 0 or 2 occurrences implies evenly spaced mesh. Please note, in case of 2 “x1” are given, please make sure that they are of the same value. Otherwise logically it is not correct.

```

*ALE_STRUCTURED_MESH_CONTROL_POINTS
$   CPID      ICASE
      1001      1
$
      x1      x2      x1
      1      .0
      8      0.0755      0.0070
      15     0.1245      0.0070
      22      0.2

```

The above setup will produce a progressive mesh at region 1 and region 3. And the element size is 0.0155 at the 1<sup>st</sup> node; gradually reduced to 0.007 at the 8<sup>th</sup> node; and remains unchanged until he 15<sup>th</sup> node; then gradually increased to 0.0155 at the 22<sup>nd</sup> node.

The other two examples below are self-explanatory and listed for the purpose of showing the versatility of the setup.

```

*ALE_STRUCTURED_MESH_CONTROL_POINTS
  1002                1
$                   x1                x2                x1
                   1                   .0                0.0155
                   8                   0.0755
                   15                  0.1245
                   22                   0.2                0.0155
*ALE_STRUCTURED_MESH_CONTROL_POINTS
  1003                1
$                   x1                x2                x1
                   1                   .0                0.0070
                   11                  0.1
                   21                   0.2

```

4. **ICASE=2.** The idea is to build the progressive spaced mesh starting from a point and extend the mesh to the left and the right. To use this option, we need to first pick a “base node” and put it at a location by specifying its coordinate. This “base node” in the example below is the 8<sup>th</sup> node with a coordinate of 0.0755. And next, we specify the element size at all the nodes listed. Still take the example below, we want element size to be of 0.0155 at the 1<sup>st</sup> node, 0.007 at the 8<sup>th</sup> node, 0.007 at the 15<sup>th</sup> node and 0.0155 at the 22<sup>nd</sup> node.

Internally, the mesh is divided into 3 regions, as discussed in the previous ICASE=1 example. Region 1 and Region 3 are of progressive mesh spacing and in Region 2 mesh is evenly distributed.

```

*ALE_STRUCTURED_MESH_CONTROL_POINTS
$   CPID             ICASE
   1001             1
$   x1              x2              x1
    1               0.0155
    8               0.0755         0.0070
   15               0.0070
   22               0.0155

```

The other two examples below are self-explanatory and listed for the purpose of showing the versatility of the setup.

```

*ALE_STRUCTURED_MESH_CONTROL_POINTS
  1002                1
$                   x1                x2                x1
                   1                   0.0155
                   8                   0.0070
                   15                  0.0070
                   22                   0.2                0.0155

```

# \*ALE

## \*ALE\_STRUCTURED\_MESH\_CONTROL\_POINTS

---

\*ALE\_STRUCTURED\_MESH\_CONTROL\_POINTS

1003		1		
\$	x1		x2	x1
	1			0.0155
	12	0.1		0.0070
	22			0.0155