A Quadratic Pipe Element in LS-DYNA®

Tobias Olsson, Daniel Hilding

DYNAmore Nordic AB

1 Background

Analysis of long piping structures can be challenging due to the enormous number of shell/solid elements that would be required to model a piping structure accurate. In that context a new beam element has been developed that can, if used correctly, reduce the number of elements used in a pipe simulation. Since it is constructed of 3 nodes it is perfect for describing pipe bends, so called elbows.

This document is meant as an introduction and modelling techniques for the elbow element. It is implemented in LS-DYNA® R7.0.0 but improvements are implemented in the coming update of LS-DYNA R7 [1].

2 Theory

The main theory is based on the work done by Almeida [2] The beam is formulated under the plane stress assumption and with thin shell theory. That means that the quotient between the thickness of the tube (t) and the outer radius (a) should be small and the quotient between the radius and the pipe curvature (R) should also be small.

$$\frac{t}{a} \ll 1,\ \frac{a}{R} \ll 1 \tag{1}$$

The basic assumption is that plane sections originally normal to the center line remain plane but not necessarily normal. The following displacement formula holds for a point in the element after deformation

$$u_i(r,s,t) = \sum_{k=1}^{3} h_k(r)u_i^k + \sum_{k=1}^{3} a_k h_k(r)\left(tV_{ti}^k + sV_{si}^k\right),\ i = 1,2,3 \tag{2}$$

Where r, s, t are iso-parametric coordinates, u_i is the displacement at any point in the pipe element, h_k is the interpolation function and u_i^k is the displacement of node k in the current element. The V_{ti}^k and V_{si}^k are the components of the rotated orientation vectors along the t and s directions, and a_k is the outer pipe radius. We calculate $\boldsymbol{V}_s^k$ and $\boldsymbol{V}_t^k$ as the cross product between the nodal rotation increment and the "old" orientation vector.

$$\boldsymbol{V}_s^k = \Delta\boldsymbol{\theta}^k \times \boldsymbol{V}_{s0}^k \tag{3}$$

$$\boldsymbol{V}_t^k = \Delta\boldsymbol{\theta}^k \times \boldsymbol{V}_{t0}^k$$

The current beam displacements assume that the cross section of the pipe does not deform. To include the ovalization to the formulation we introduce a new displacement field as follows

$$w(r,\phi) = \sum_{m=1}^{3}\sum_{k=1}^{3} h_k(r)(c_m^k \sin 2m\phi + d_m^k \cos 2m\phi) \tag{4}$$

Where c_m^k and $d_m^k$are generalized ovalization displacements. The total displacement is calculated as the sum of u and w which give the beam a total of 12 degrees of freedom per node.

Almeida's theory is here enhanced with the possibility to include an inner pressure to for example simulate inner or outer loads such as gas pressure or water pressure due to sub sea placement. The inner pressure works two ways. First it works to stiffen the pipe against bending, i.e., reduces the ovalization displacements. Secondly, it adds stress in the axial and circumferential directions by using simple linear pipe equations. The stresses that are transferred and added to the materials are given by:

Straight pipe

$$\sigma_r = \frac{Pa_m}{2t},\ \sigma_{circ} = \frac{Pa_m}{t} \quad (5)$$

Curved pipe

$$\sigma_r = \frac{Pa_m}{2t},\ \sigma_{circ} = \frac{Pa_m}{2t}\frac{2R-a_m\cos\phi}{R-a_m\cos\phi}, \quad (6)$$

where P is the applied pressure, a_m is the mean radius of the tube, R is the pipe curvature radius and t is the pipe thickness.

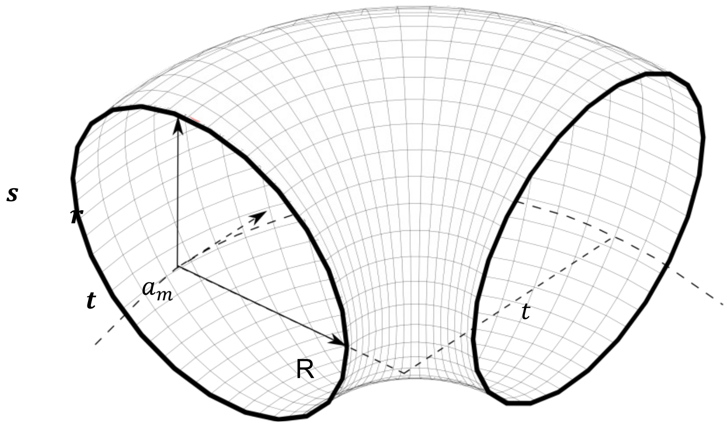

Fig. 1: Illustration of a pipe beam element, $\boldsymbol{r}$, $\boldsymbol{s}$ and $\boldsymbol{t}$ are position vectors, $\boldsymbol{a_m}$ is the mean radius of the tube, R is the pipe curvature radius and t is the wall thickness.

3 Modelling

The pipe element is constructed of three nodes and an orientation node. And the layout of the pipe is to interpolate a quadratic function through the nodes. The bends should be modeled as circular arcs. The orientation vectors are always constructed such that $\boldsymbol{t}$ is perpendicular to the pipe axis and for a curved pipe pointing at the curvature center or for a straight pipe in the same plane as the orientation node and perpendicular to the pipe axis. The curvature center is automatically calculated and it is assumed that the bend is a part of circle. If the pipe is initially curved the orientation node is set to the curvature center. If a straight pipe is used the orientation node should be set to keep continuity in the $\boldsymbol{t}$ direction between elements.

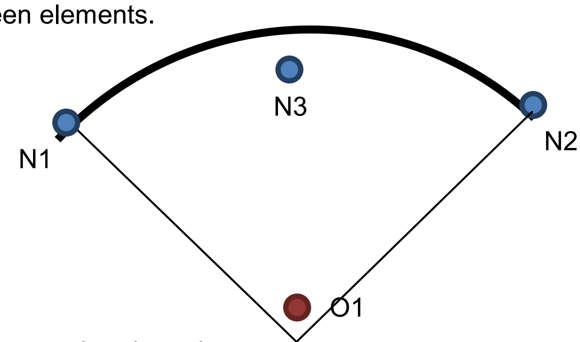

Fig. 2: Pipe beam node orientation.

3.1 Input example (Element)

The input for a pipe element is almost identical as for an ordinary beam. The difference is that the middle node (N3) is also given on card 1 row 2. Note that the orientation node must always be included even though its coordinates are calculated internally for a curved pipe:

```
*ELEMENT_BEAM_ELBOW
$      EID       PID     NODE1     NODE2     ONODE
         1         1        N1        N2        O1
$    NODE3
        N3
```

As a rule of thumbs and for good accuracy it is recommended to use at least 4-6 elements for a 90 degree elbow.

3.2 Input example (Section)

The pipe element is activated by setting the element formulation to 14 in *SECTION_BEAM. Also an integration rule id must be given and the CST parameter should be set to 2. Moreover, the integration rule must be tubular (9).

Physical options such as pressure and elongation effects are also given in the section keyword. The pressure is given at card 1 on row 2, the inclusion of end effects are given at card 3 on row 2. Card 2 on row 2 is for output of the ovalization degrees of freedom, that is, c_k and d_k as an ASCII-file. Doing so it is possible to visualize the ovalization of the pipe by valuate the ovalization displacements $w(r, \phi)$. Below is an example of a section with 1 MPa as internal pressure and both ovalization printing and elongation active.

```
*SECTION_BEAM
$      SID    ELFORM      SHRF   QR/IRID       CST
         1        14       1.0        -1       2.0
$       PR     IOVPR    IPRSTR
    1.0E6          1         1
*INTEGRATION_BEAM
$     IRID       NIP        RA      ICST         K
         1         0         0         9         0
$       D1        D2
       1.0       0.7
```

Also, note the option NEIPB on*DATABASE_EXTENT_BINARY that control the output off the loop stresses. Right now the only option that will work is to set NEIPB to 0 (default) and use the corresponding ASCII-file to fringe plot the loop-stress. All other stresses are of course included in the d3plot file.

3.3 Ovalization degrees of freedom

The extra degrees of freedom are described by scalar nodes that are automatically created during the initialization. Unfortunately that means that the node ids are not known beforehand. However, during the generation of these extra nodes they are echoed to the messag file for easy access for the user. For example, the information can look like this:

```
ELBOW BEAM:       1
n1-n3-n2:               1     2     3
ovalization nodes:  1701  1704  1703
                    1705  1707  1706
```

And it means that elbow beam id 1 that is constructed of nodes 1, 2 and 3 were node 3 is the middle node, have the ovalization degrees saved in nodes 1701 to 1707. The c_1, c_2 and c_3 for node 1, 3 and 2 are stored in 1701, 1704 and 1703, and d_1, d_2 and d_3 are stored in 1705, 1707 and 1706. To simulate a cantilever beam the first node should be constrained in all DOFs. In this case that means nodes 1, 1701 and 1705.

If the IOVPR flag is set, then the ovalization displacements for each element are written to an ASCII file ‘elbwov’. They can be used for further analysis of the pipe. For example the total ovalization of the pipe can be calculated by using the displacement formula above. The format for the ASCII file is as follows (spaces have been removed to fit this page):

```
OVALIZATION D.O.F. WITH PRESSURE: 1.210E+06  (TIME =   1.000000)
 BEAM ID:   1     c1       c2       c3       d1       d2       d3
     NODE 1: 0.35E-4 -0.49E-5  0.16E-6 -0.40E-3 -0.19E-4 -0.15E-5
     NODE 2: 0.46E-4  0.28E-5 -0.77E-7  0.36E-3  0.23E-4  0.52E-4
     NODE 3: 0.11E-3 -0.74E-5  0.14E-6  0.16E-2 -0.84E-4  0.10E-3
```

Note that the ovalization nodes only have translation degrees of freedom. That means that velocity boundary conditions cannot be set.

3.4 Contacts

Due to the extra node in this formulation the beam contacts will not work for curved beams. If a beam contact is used the curved beam will be treated as a linear beam between node 1 and 2. Node to node contacts and node to surface contacts should work as usual but the curved beam between the nodes will not be added to the contact.

4 Examples

In LS-PrePost® 4.1 or newer a new rendering engine is implemented that can visualize the pipes as curved beams, see Fig. 3. All that is needed is that the k-file is used together with the d3plot file and that the CST flag is set to 2.

4.1 Two elements Cantilever beam

The first example is a simple cantilever beam that is constructed with only 2 elbow elements. The purpose is to do a comparison with the standard beam type 1 and the analytical result that is available in Almeida [2].

Fig. 3: Cantilever beam modeled with 2 elbow elements. To the left is the initial geometry and to the right is the deformed state.

The material is linear elastic with a Young’s modulus at 207GPa and Poisson ratio equal to 0.0. The applied torque is 40kNm. The initial straight geometry is deformed by the moment and close to a half-circle is obtained. The same simulation was done with beam type 1 and a comparison between the deformations of the loaded node was done. The result is viewed in Fig. 4 and the simulation with the type 1 beam is not able to complete this test case and is therefore not suitable for this kind of simulations.

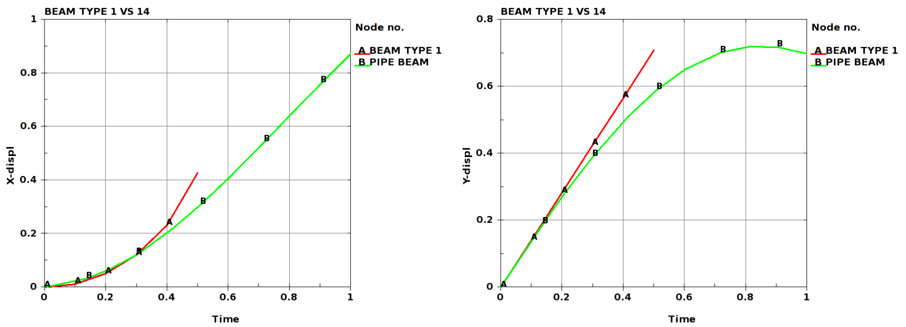

Fig. 4: To the left is the x-displacements for the elbow (B) beam and beam type 1 (A) shown. To the right is the y-displacements for the elbow beam (B) and beam type 1 (A) shown.

An interpretation can be that the type 1 beam have difficulties when the y-displacements become non-linear and as a consequence the simulation is not able to complete to the end. From Almeida [2] an analytical result can be found and a comparison is made in Fig. 5.

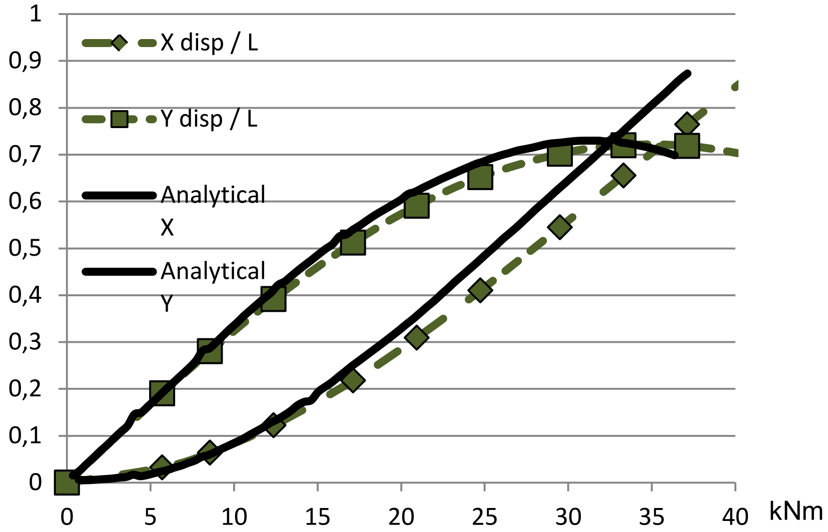

Fig. 5: An comparison of simulated displacements versus analytical.

As can be seen in Fig. 5 a good agreement is obtained even for this coarse 2 element mesh. Note that the type 1 beam was not able to complete the simulation.

4.2 Piping structure

The second example consists of a few pipes that undergo torsional deformations. One end is fixed and a load is applied at the other end of the structure. See Fig. 6. This example is simulated with the simplest elastic material (*MAT_001). A list of all supported materials is given in Appendix A and the complete input deck is appended in Appendix B.

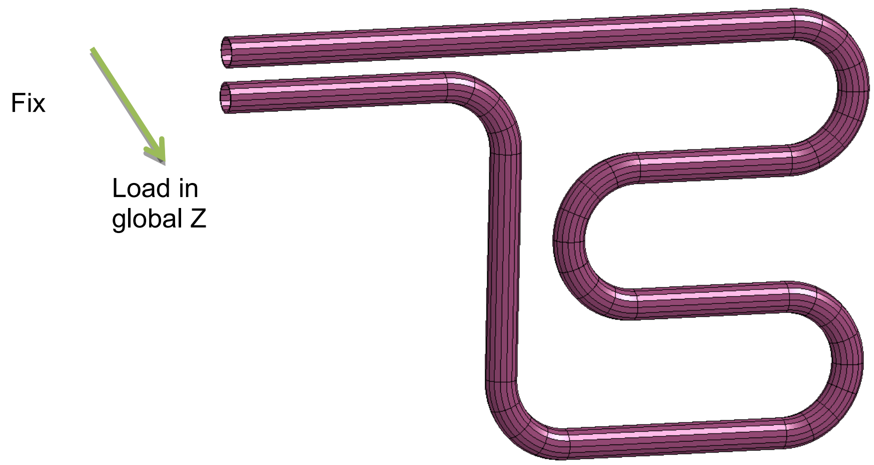

Fig. 6: Initial model. Node 1 is fix and the last node is loaded in the global z-direction.

In Fig. 7 some fringe plots from the above simulation are shown.

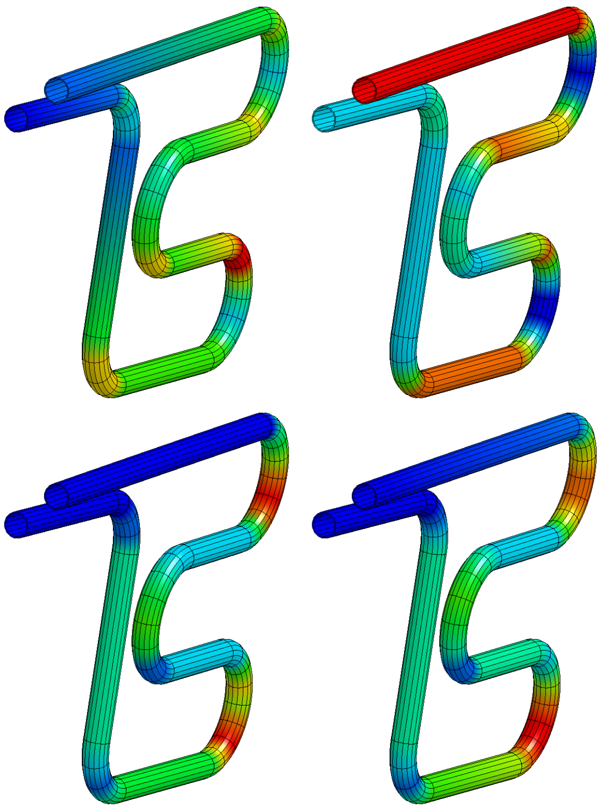

Fig. 7: Different fringe plots. Reading from top left to bottom right: axial-stress, loop-stress, rs-stress and tr-stress.

5 Pre- and postprocessing

Support for pre- and postprocessing of the new element is available in the current, March 2013 version, of LSPrePost 4.1,

6 Summary

A new beam formulation has been developed and implemented in LS-DYNA R7. It is a 3 node beam with 36 degrees of freedom and quadratic interpolation between nodes. It is tailored for the pipeline and offshore industries but can of course be used in other suitable areas as well. It is cost efficient and accurate.

7 References

[1] Hallquist, J. " LS-DYNA R7.0.0 Keyword User's Manual – Volume I", Development version, Livermore Software Technology Corporation, revision 2999, March 29, Livermore, 2013.
[2] Almeida, C.A., "A simple new element for linear and nonlinear analysis of piping systems", PhD Thesis, MIT, 1982.

8 Appendix A

Currently supported materials (early 2013) are materials number 1, 3, 4, 6, 24,153, and 195.

9 Appendix B

The input file that was used for the second example.

```
*KEYWORD
*CONTROL_TERMINATION
$   endtim    endcyc     dtmin    endeng    endmas
     1.000         0       0.0       0.0       0.0
$
*DATABASE_EXTENT_BINARY
10,10,36
0,0,36


*BOUNDARY_PRESCRIBED_MOTION_NODE
      1701         3         2      1000      0.20
*CONTROL_IMPLICIT_GENERAL
1,.1
*CONTROL_IMPLICIT_AUTO
1,100,10,0,0.1
*DEFINE_CURVE
1000,
0.0,0.0
1.0,1.000
4.0,1.000
*DATABASE_BINARY_D3PLOT
$       dt      lcdt
      0.01
$
*NODE
$#   nid               x               y               z       tc       rc
       1             0.0             0.0             0.0        7        7
       2             0.5             0.0             0.0
       3             1.0             0.0             0.0
       5         1.05853        -0.00576
       6         1.11481        -0.02284
       7         1.16667        -0.05056
       8         1.21213        -0.08787
       9         1.24944        -0.13333
      10         1.27716        -0.18519
      11         1.29424        -0.24147
      12         1.30000        -0.30000
     201         1.30000        -0.80000
     202         1.30000        -1.30000
      13         1.30576        -1.35853
      14         1.32284        -1.41481
      15         1.35056        -1.46667
      16         1.38787        -1.51213
      17         1.43333        -1.54944
      18         1.48519        -1.57716
      19         1.54147        -1.59424
      20         1.60000        -1.60000
      22             2.1            -1.6             0.0
      23             2.6            -1.6             0.0
      70         0.00000          -1.000           0.000
```

```
      80         1.00000           -1.30           0.000
      90         1.60000           -1.30           0.000
     100         2.00000            1.30           0.000
    1131          2.6585         -1.5942
    1141          2.7148         -1.5772
    1151          2.7667         -1.5494
    1161          2.8121         -1.5121
    1171          2.8494         -1.4667
    1181          2.8772         -1.4148
    1191          2.8942         -1.3585
    1201          2.9000         -1.3000
    1211          2.8942         -1.2415
    1221          2.8772         -1.1852
    1231          2.8494         -1.1333
    1241          2.8121         -1.0879
    1251          2.7667         -1.0506
    1261          2.7148         -1.0228
    1271          2.6585         -1.0058
    1281          2.6000         -1.0000
    1291          2.2500         -1.0000
    1301          1.9000         -1.0000
    1311          1.8415         -0.9942
    1321          1.7852         -0.9771
    1331          1.7333         -0.9494
    1341          1.6879         -0.9121
    1351          1.6506         -0.8667
    1361          1.6228         -0.8148
    1371          1.6058         -0.7585
    1381          1.6000         -0.7000
    1391          1.6058         -0.6415
    1401          1.6228         -0.5852
    1411          1.6506         -0.5333
    1421          1.6879         -0.4879
    1431          1.7333         -0.4506
    1441          1.7852         -0.4228
    1451          1.8415         -0.4058
    1461          1.9000         -0.4000
    1471          2.2500         -0.4000
    1481          2.6000         -0.4000
    1531          2.6585         -0.3942
    1541          2.7148         -0.3772
    1551          2.7667         -0.3494
    1561          2.8121         -0.3121
    1571          2.8494         -0.2667
    1581          2.8772         -0.2148
    1591          2.8942         -0.1585
    1601          2.9000         -0.1000
    1611          2.8942         -0.0415
    1621          2.8772          0.0148
    1631          2.8494          0.0667
    1641          2.8121          0.1121
    1651          2.7667          0.1494
    1661          2.7148          0.1772
    1671          2.6585          0.1942
    1681          2.6000          .20000
    1691          1.3000          .20000
    1701          0.0000          .20000
*PART
$# title
ELBOW PIPE
$#     pid     secid       mid
        45        45        45
*KEYWORD
*ELEMENT_BEAM_ELBOW
$    eid     pid       n1       n2       n5
      11      45        1        3       70
       2
*ELEMENT_BEAM_ELBOW
$    eid     pid       n1       n2       n5
      12      45        3        6       80
       5
*ELEMENT_BEAM_ELBOW
$    eid     pid       n1       n2       n5
      13      45        6        8       80
       7
*ELEMENT_BEAM_ELBOW
$    eid     pid       n1       n2       n5
```

```
      14      45       8      10      80
       9
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
      15      45      10      12      80
      11
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
       6      45      12     202      90
     201
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
      16      45     202      14      90
      13
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
      17      45      14      16      90
      15
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
      18      45      16      18      90
      17
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
      19      45      18      20      90
      19
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     110      45      20      23     100
      22
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     111      45      23    1141     100
    1131
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     112      45    1141    1161     100
    1151
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     113      45    1161    1181     100
    1171
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     114      45    1181    1201     100
    1191
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     115      45    1201    1221     100
    1211
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     116      45    1221    1241     100
    1231
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     117      45    1241    1261     100
    1251
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     118      45    1261    1281     100
    1271
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     119      45    1281    1301     100
    1291
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     120      45    1301    1321     100
    1311
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     121      45    1321    1341     100
    1331
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     122      45    1341    1361     100
```

```
    1351
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     123      45    1361    1381     100
    1371
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     124      45    1381    1401     100
    1391
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     125      45    1401    1421     100
    1411
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     126      45    1421    1441     100
    1431
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     127      45    1441    1461     100
    1451
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     128      45    1461    1481     100
    1471
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     129      45    1481    1541     100
    1531
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     130      45    1541    1561     100
    1551
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     131      45    1561    1581     100
    1571
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     132      45    1581    1601     100
    1591
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     133      45    1601    1621     100
    1611
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     134      45    1621    1641     100
    1631
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     135      45    1641    1661     100
    1651
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     136      45    1661    1681     100
    1671
*ELEMENT_BEAM_ELBOW
$    eid     pid      n1      n2      n5
     137      45    1681    1701     100
    1691
*SECTION_BEAM
$      sid    elform      shrf   qr/irid       cst
        45        14     1.000        -2       2.0
$       PR     IOVPR    IPRSTR
    12.000         1         0
*INTEGRATION_BEAM
         2         0         0         9         0
     .1375      .125
*MAT_ELASTIC
$      mid        ro         e        pr
        45   7.86e+3   200.E+9      0.28
*END
```