

Basic Finite Elements — One Dimensional Elements



Document Version 1.0 (1/1/2019)

OVERVIEW

This document is intended for use with *1D Elements*© Finite Element Analysis Program by Structural FEA, LLC and ESP Composites, LLC. It provides basic information for 1-dimensional elements (rods, beams, and springs) that are used in 2D space.

1D ELEMENT ADVANTAGES OVER 2D AND 3D ELEMENTS

1D finite elements (beams, rods, springs, etc.) have some advantages over 2D (shell) and 3D (solid) elements.

REQUIRED OUTPUTS: For rods, the axial load is the output. For beams, the axial load, shear load, and moment are the outputs. These outputs can directly be used for classical checks such as Johnson-Euler columns, local buckling, crippling, and plastic bending.

Trying to capture the actual behavior of the checks mentioned above via 2D/3D elements can be complex and time consuming (may also require nonlinear geometry/material analysis). For example, crippling is a phenomenon that is nonlinear in geometry and material. This approach is not usually practical for engineering solutions. Alternatively, if a classical analysis is to be used, a cross section's axial load, shear load, and moment can be extracted from 2D/3D models. For example, MSC Patran has a "Free Body" tool that can be used to determine the cross sections forces/moments for a given cut on a 2D/3D body. However, this requires additional processing. Also, many post-processors do not have this capability.

For many models, the practical solution is to use a 1D element when possible. The required outputs are in a usable manner; there is no "extra" information that may require further processing before performing classical analysis.

SECTION CHANGES: Another advantage of a 1D element is the direct ability to easily change section properties; for a beam element the area and moment of inertia can be modified with simple numeric input changes. However, for a 2D/3D model, the physical geometry may need to be changed (and the underlying mesh) to change the cross sections properties. These changes can be time consuming.

BOUNDARY CONDITIONS: Applying proper boundary conditions to 2D/3D models can sometimes be a challenge:

- may be time consuming for some models
- prone to error for less experienced users
- Poisson effects must be considered
- models can easily be overconstrained
- more difficult to check by other parties
- creating connections between structural members may not be intuitive; for example, see the “PIN FLAGS” section of this manual
- localized “punch” loads or “punch” constraints must be considered as they can affect the stiffness of the model and cause stress concentrations

SOLVE TIME: The solve time for 1D elements is faster than for 2D/3D elements.

ROD ELEMENT

A rod element is a 1D line element that is connected by 2 nodes. In 2D space, a rod element has 2 DOF (degrees of freedom) at each node (two translation DOF). A rod element is a two-force member and only has an axial load.

A rod element *can not* carry a moment. Therefore, it can not carry a load transverse (normal) to its axis. It can only carry an axial load that is in-line with its axis and has an equal and opposite load at each node. A rod element can thought of as a structural member with a pinned joint (spherical joint) at each node.

The inputs for a rod element are:

- 2 nodes with X,Y,Z positions (or just X,Y in 2D space)
- connectivity (defines the 2 node nodes)
- area of the rod's cross section
- elastic modulus in the axis of the rod

The outputs for a rod element are:

- force (either tension or compression) in-line with the element's axis

Truss structures are often modeled as rod elements if the joints are assumed to be pinned. Note that even if the joint is not actually pinned, the moment at the joint is usually “small” for a “well-built” truss structure. An example of a truss structure with rod elements is shown in Figure 1.

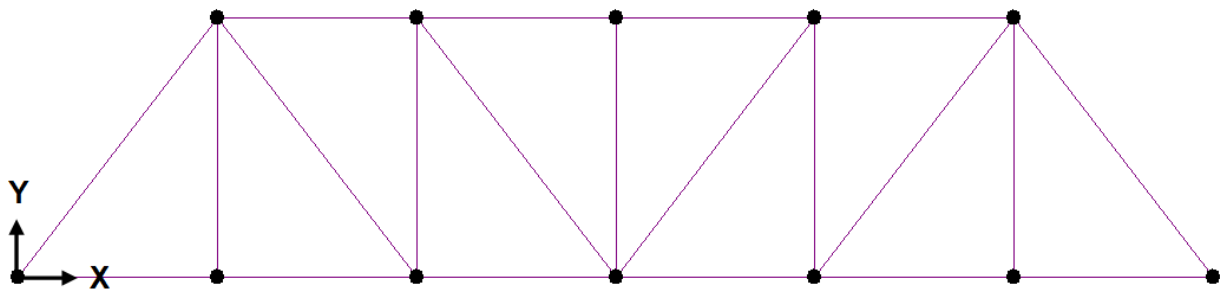


Figure 1

Note that a structural member represented by a rod element should not be discretized (refining of the mesh). For example, on the right side of the truss in Figure 2, there are two rod elements that make up the structural member. Since the element can not carry a moment, this will can an undesirable result.

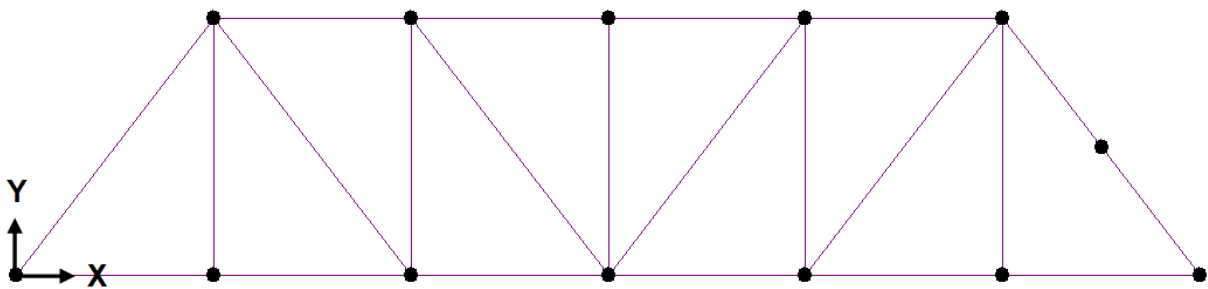


Figure 2

If only the overall deformation of the structure and the axial loads in each member are sought, then the mode in Figure 1 is sufficient. However, what happens if the Eigen solution buckling or natural frequency of each member is sought? In that case, the members must have a refined mesh.

As previously discussed, Figure 2 is not a valid model and will not achieve the desired result. Instead, the members of interest must be beam elements. With a sufficient amount of beam elements for the member of interest, the desired mode shape can be obtained. However, this presents another problem. Beams are, by default, “welded” at the joint. If that is not the intent of the model, this must be addressed. Two ways are presented in the section for beam elements.

ROD ELEMENT – EXAMPLE USES

- truss structures
- mechanically fastened joints (structural members)

BEAM ELEMENT - GENERAL

A beam element is a 1D line element that is connected by 2 nodes. In 2D space, a beam element has 3 DOF (degrees of freedom) at each node (two translation and one rotation).

A beam element *can* carry a moment. Therefore, it can also carry a load transverse (normal) to its axis. By default, beams that are connected to each other are “welded” to each other. Because the connection is “continuous” (carries axial, shear, moment), a structural member to be divided into many beam elements without causing rigid body motion.

The basic inputs for a beam element in 2D space are:

- 2 nodes with X,Y positions
- connectivity (defines the 2 node nodes)
- area of the beam’s cross section
- second moment of inertia corresponding to a moment about the Z-axis
- elastic modulus in the axis of the beam

Additional inputs for a beam element in 2D space are as follows. These options may or may not be present for a given solver, but are supported in the *1D Elements* program, MYSTRAN, and NASTRAN:

- shear deformation coefficient
- pin flags

The outputs for a beam element are:

- axial load
- shear force
- moment

BEAM ELEMENT – DISCRETIZATION – GENERAL

This section discusses discretization, which is the act of subdividing a structural member into various numbers of elements (from 1 to many elements). The required number of elements is a function of the type of analysis being performed. For this section, the loads (axial, shear) and moment outputs for beam elements are discussed.

Not all finite element solvers have the same approach for beam element output. For example, some solvers provide the moment at the integration point, which may *not be at the ends* of the beam element; ABAQUS is one such solver and for the first order beam element there is only **one** moment output per beam element. Loads and boundary conditions can only be applied at the nodes and these locations are often of interest (usually where the max/min shear and moment values are). In order to determine an accurate result, many beam elements are required to approximate the results at the end positions.

Alternatively, some finite element solvers provide the moment at **both** ends of the beam. This is convenient because those are often the locations of interest. Both MYSTRAN and NASTRAN provide the moment at each of the beam (both nodes); *1D Elements* provides these results. Therefore, only a minimum number of beam elements are required. However, the actual number of required elements is a function of the type of analysis required (as discussed next).

Some FEA programs “internally” subdivide the elements created by the user. For example, if the user creates a single beam element, the program may create/solve many more (i.e. 10 elements) in the “background”. This is convenient since it can avoid some discretization issues. Distributed loads can also be applied using this approach. *1D Elements* does not currently support internal subdivision of the elements.

BEAM ELEMENT – DISCRETIZATION – LOADS AND MOMENTS

To determine the **loads (axial and shear) and the moments** for beam elements, a node only needs to exist at the location of a load application or boundary condition (*provided the solver outputs the moments at each node as discussed in the prior section*). Additional node/elements are not required. Consider the model shown in Figure 3.

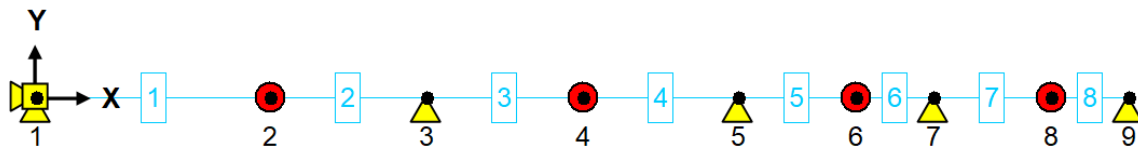


Figure 3

The deformed shape (magnitude is arbitrary) is shown in Figure 4. While the deformed shape is not very representative of the actual shape, the model accurately determines the shear and moment values along the entire span (See Figure 5).

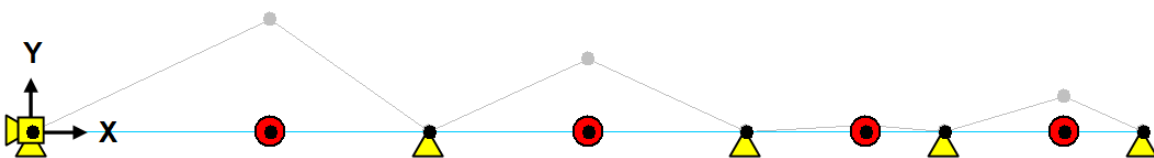


Figure 4

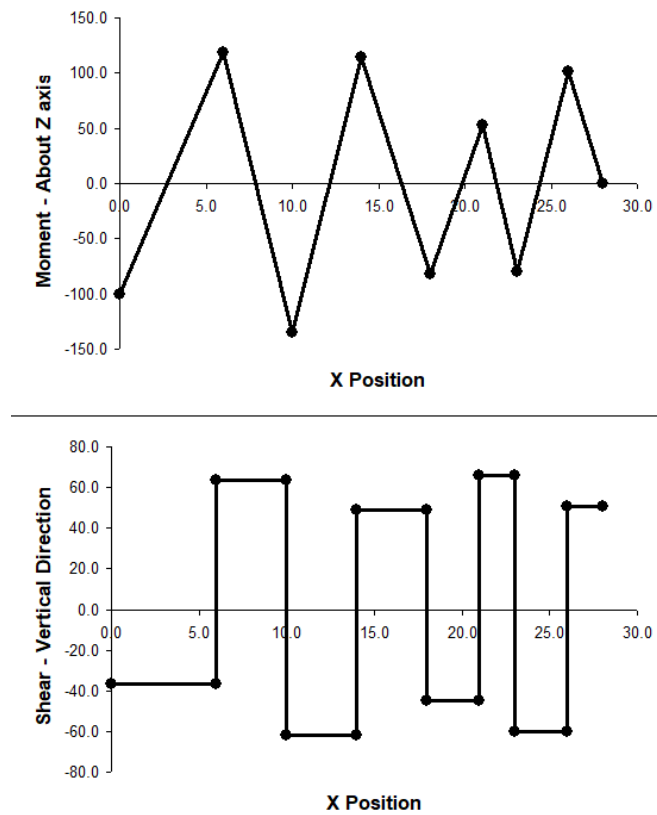


Figure 5

For comparison, a refined mesh is shown in Figure 6. While the deformed shape is more representative of the actual deformed shape, the shear and moment diagrams are identical (See Figure 7).

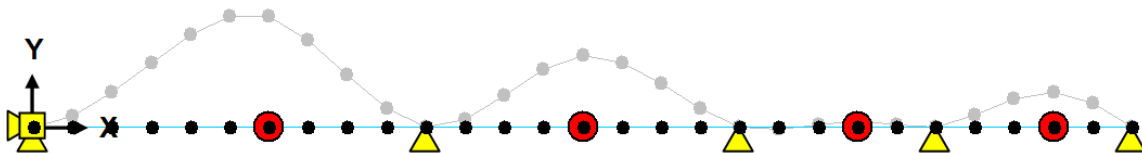


Figure 6

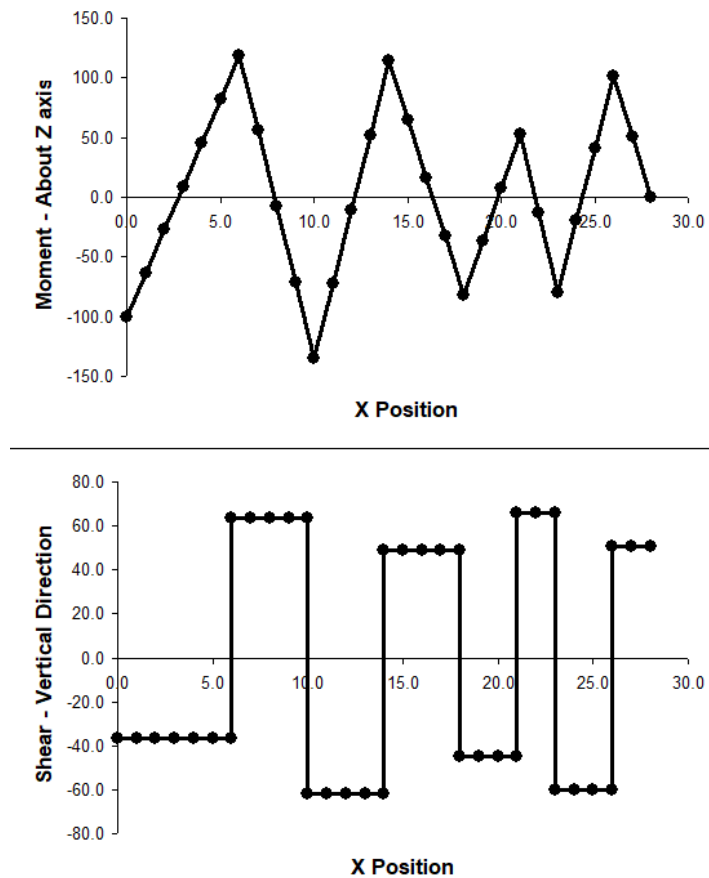


Figure 7

As another example, consider the case the scenario shown in Figure 8. Node 3 (X=20.0) has a rotational constraint (no rotation). Therefore, a step in the moment is expected at this node. A step in the moment would also occur if there were an applied moment at a location other than the far left/right ends. The resulting shear and moment diagrams are shown in Figure 9.

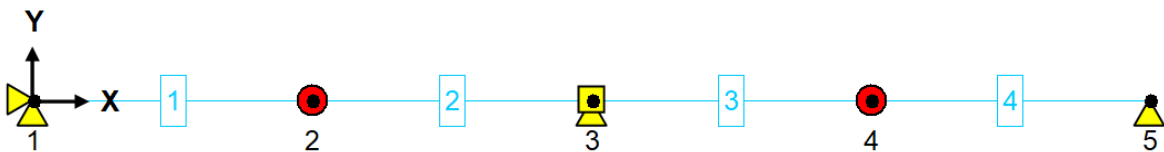


Figure 8

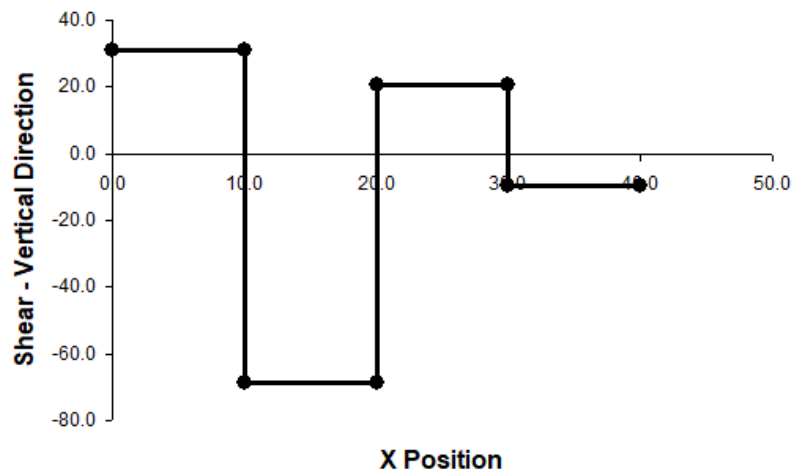
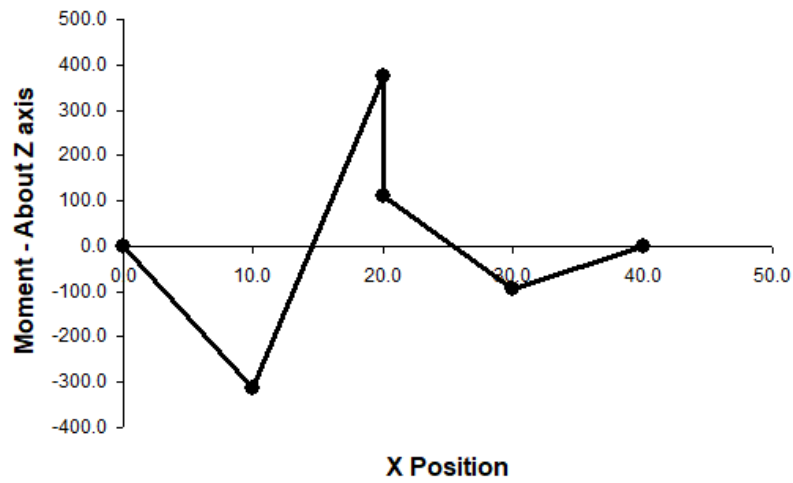


Figure 9

If the structural member is divided into many more beam elements, the result is shown in Figure 10. The shear and moment curves are the same as for Figure 9.

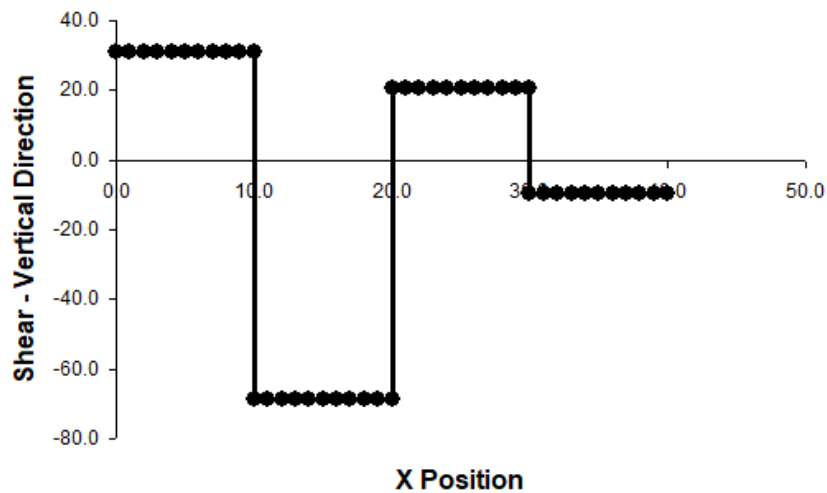
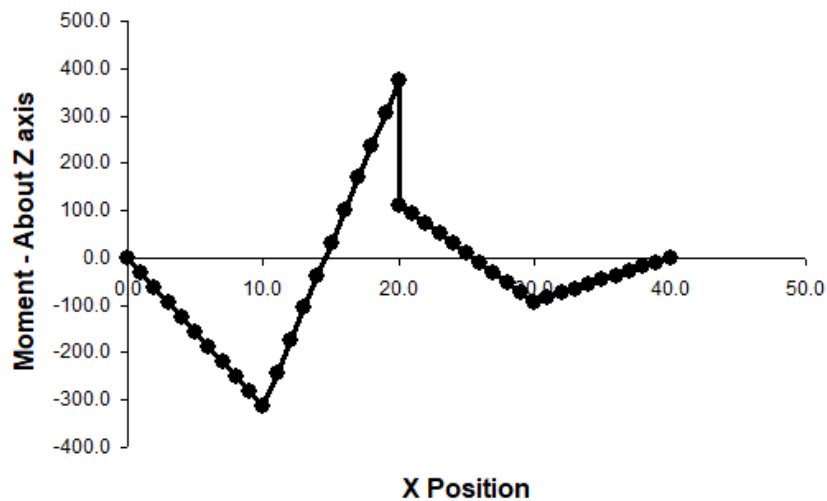


Figure 10

BEAM ELEMENT – DISCRETIZATION – DEFLECTION/BUCKLING/NATURAL FREQUENCY

In the previous section, it was shown that nodes only need to exist at the loads/boundary conditions if the shear loads/moments are the required outputs. However, if an accurate deflected shape is desired, “many” elements are required. For example, Figure 4 does not capture the beam’s curvature, but Figure 6 more accurately represents the actual deformed shape.

If an Eigen solution (buckling or a natural frequency) is desired, an **accurate representation of the deformed shape must be captured**. “Many” elements are required to accomplish this. The exact number of required elements to reach convergence is not known ahead of time. However, provided the deformed shape is accurately captured, then the solution is expected to be accurate.

BEAM ELEMENT – SHEAR DEFORMATION

A beam element has two components for deformation: bending and shear (transverse shear). For long beams, the bending deformation dominates the total deformation and the shear deformation is minor. However, for short beams, the shear deformation can be significant. Also, if a sandwich structure beam (with a soft core) is represented, the transverse shear deformation can be significant even if the beam is not “short”.

The shear deformation is represented by $K*V*L/(A*G)$, where K is the shear coefficient, A is the cross section’s area, V is the internal shear load in the beam element, G is the shear modulus, and L is the length of the beam.

For MYSTRAN, if K is not defined, then the shear stiffness is assumed to be infinite and there is no transverse shear deformation.

For NASTRAN, if K is not defined, then the shear stiffness is assumed to be 1.0. If K is defined to be 0.0, the shear stiffness is infinite and there is no transverse shear deformation.

BEAM ELEMENT – PIN FLAGS

Some solvers support “pin flags”. A pin flag allows a DOF(s) to be released at either end of the beam element. For example, if all of the rotational DOF(s) are released on both ends of the beam, the beam effectively becomes a rod element.

Pin flags can be useful in cases where beam-like behavior is desired for a structural element, but the ends are intended to be pinned. For example, if buckling or natural frequency of a truss structures is desired, many elements should exists between the joints (joints are assumed to be pinned joints). However, as discussed in the “ROD ELEMENT”

section, these elements must be beam elements to prevent rigid body motion. If pin flags are not used, the joint connections will be “welded”. If a pin flag for rotation is used at the joint, the rotation is released. This model will then have the intended behavior.

For example, consider Figure 11. Two point loads are applied in the downward direction (two red dots). On the left side, the beam elements are connected “normally”, without pin flags. Therefore, the moment is transmitted through the top corners, causing the vertical beam elements to bend. On the right side, the rotational DOF is released at each corner via pin flags. Therefore, the vertical members are on in compression (no bending).

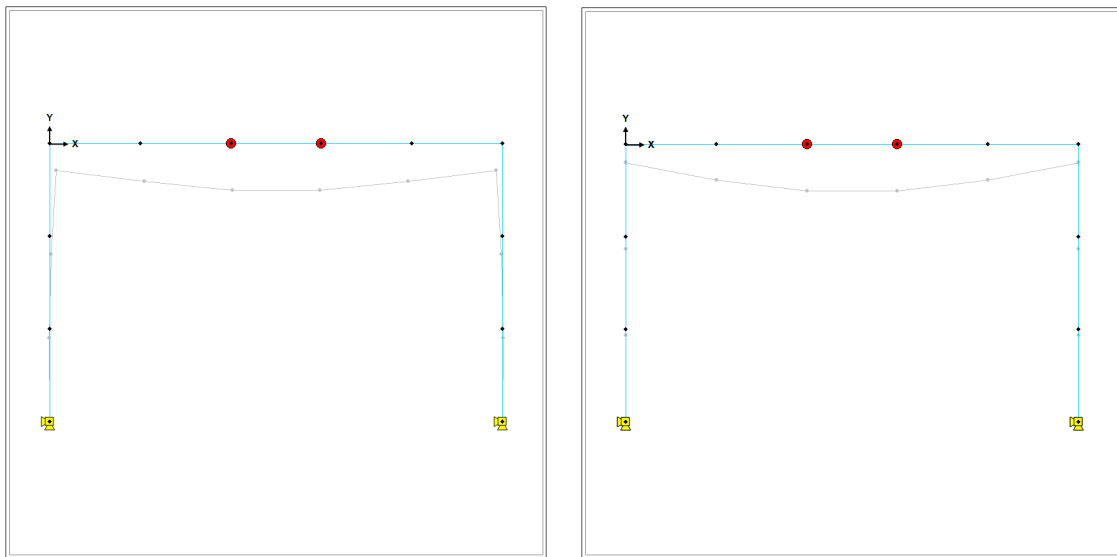


Figure 11

BEAM ELEMENT – EXAMPLE USES

Compared to rods and springs, beam elements are the most flexible, having 3 DOF in 2D space. These DOF can also be released (may be different at each end of the beam) via pin flags. Some example uses are:

- truss structures
- rings
- frames
- brackets
- clips
- arch

It is also possible to model shear joints with beam elements (beam elements can either represent mechanical fasteners or a bondline). Spring elements can also represent mechanical fasteners or a bondline, but can not capture the eccentricity effect. If beam elements are used, the tension load in a fastener or peel stress in a bondline can be determined because the eccentricity effect is recognized. In order to use beam elements, the load-displacement relationship of the beam must be equated to the fastener flexibility or the bondline stiffness.

SPRING ELEMENT

A spring element is a true “line” element, but it usually shown as a line in a pre/post processor. A spring element simply connects two nodes together via a given stiffness and direction.

SPRING ELEMENT – SPECIAL CONNECTION

Some solvers do not have pin flags (See the “PIN FLAGS” section of this manual). In substitute of pin flags, spring elements can be used. This is accomplished by connecting the two nodes (one end of beam A and one end of beam B) together via a spring elements. The spring elements are the DOF that are to “connected” and have a large stiffness.

For example, consider two beam elements in 2D space are intended to be connected with a pinned joint. First, the two beam elements are not directly connected. Next, two spring elements (with zero length) are used (DOF 1 and DOF 2 corresponding to X and Y). The stiffness values are large. After the model is run, verify that the two nodes of the beam have the same displacement, but that the joint is free to rotate (since a spring element for a rotation was not used).

SPRING ELEMENT – EXAMPLE USES

- mechanically fastened joints
- bonded joints
- elastic foundation
- special connections

RIGID BODY MOTION

Valid models do not exhibit rigid body motion (RBM). Rigid body motion is a condition where a very small force causes a theoretically infinite deflection because there is no resistance to the motion.

In 2D space, rigid body motion should be prevented in the X-direction translation, Y-direction translation, and rotation about the Z-axis. At a minimum, at least one node in the model must have a X-direction constraint (or enforced displacement), and at least one node in the model must have a Y-direction constraint (or enforced displacement). The Z-axis rotation may be prevented by constraining at least one node in the Z-axis or constraining multiple nodes in the translation directions such that Z-axis RBM is prevented.

NONLINEAR GEOMETRY

There are various types of nonlinearities supported by FEA programs, such as nonlinear geometry, nonlinear materials, and contact.

Nonlinear geometry occurs when the actual deformation affects the result as the load/displacement continually increases.

An example of nonlinear geometry is nonlinear buckling (general nonlinear buckling also includes the effect of material nonlinearity). If a compressive load is applied to column (and there is also a perturbation load such as a side load), these loads will cause bending. When the loads are small, the bending is small. However, as the load increases, the deformation associated with bending increases. In turn the eccentricity of the compressive load increases the bending, and so on. Nonlinear geometry is required to continually update the bending deformation and associated eccentricity.

Analogous to nonlinear buckling is a beam-column. A beam-column has both bending loads and compressive loads (along the beam's axis). As the compressive load increases, the bending deformation increases and so does the eccentricity. A nonlinear geometry analysis is required to capture this effect.

In the reverse manner, a phenomenon known as “stress stiffening” may occur. An example is the membrane of a drum. If a normal load is applied to the membrane, it becomes more and more difficult to increase the deformation (the load-displacement relationship is not linear).

EIGEN SOLUTION BUCKLING

An Eigen solution can determine the Eigenvalue and Eigenvector for buckling. The load that Eigen solution buckling occurs is the applied load in the FEM multiplied by the Eigenvalue. The Eigenvector is the displaced shape for the given mode. The magnitude of the Eigenvector is arbitrary (the shape itself is the relevant output).

Eigen solution buckling is equivalent to elastic buckling (Euler buckling). It does not address the nonlinear effects of material. For example, short (sometimes called intermediate) columns exhibit plasticity. These columns are known as “Johnson” columns. The classical “Johnson-Euler” buckling solutions (not an aspect of Eigen solution buckling) are used to predict the full range of slenderness ratios (and column length) of buckling behavior for a column. If the column is short (intermediate), then Eigen solution buckling will over-predict the column's capability.

UNITS

Finite element programs do not usually specify the units (they are inherently “unitless”). Instead, the user must input a set of consistent units. Any set of units is possible, provided they are consistent.

Three “base units” and a temperature unit are used for structural FEA problems. For example, the IPS system (English units) utilizes the three base units of inches, pounds (force), seconds. Therefore, the pressure output psi (lbf per square inch) and a moment output is inch-lbf.

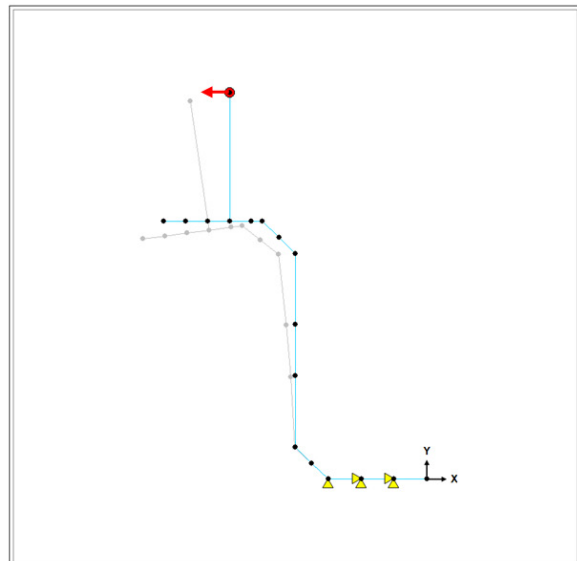
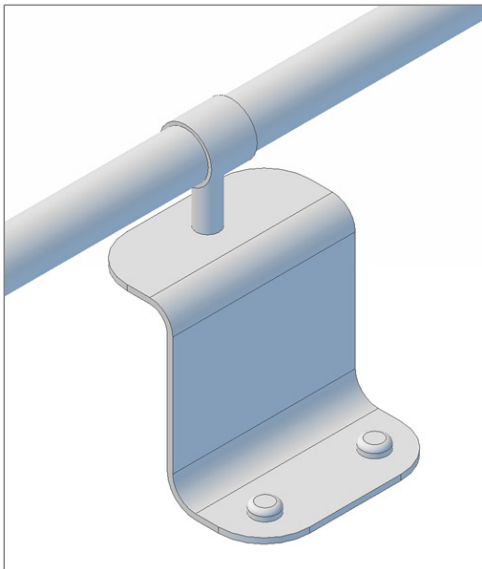
If a Length, Force, Time set of units is used, the mass unit is a function of the length, force, and time. For the IPS system, the mass unit is $\text{lbf}\cdot\text{s}^2/\text{in}$. Therefore, if a consistent set of units is used, the density has units of $\text{lbf}\cdot\text{s}^2/\text{in}^4$. The consistent density unit is required if a natural frequency or dynamic analysis is performed.

A Length, Mass, Time set of units is also possible (though less common). In this case, the Force unit must be determined as a function of length, mass, and time.

PLATE-LIKE BEHAVIOR

A beam element is assumed to be long and slender. This is typical for beams and cross sections such as an I, C (channel), rod, tube, etc. However, plate-like behavior can be also be simulated with beam elements, as shown in the following figure.

If plate-like behavior is simulated, anti-clastic curvature may be resisted (the folds in the following figure prevent anti-clastic curvature). Therefore, the actual bending stiffness is up to $1/(1-\nu^2)$ greater than predicted with classically beam theory, where ν is the Poisson's ratio. The actual increase in stiffness is a function of the plate's aspect ratio. This phenomenon, for both isotropic and composite materials, is further discussed in *"Practical Analysis of Aircraft Composites"*.



NUMERICAL INPUTS

In general, numerical inputs must be “reasonable”. For example, the area and second moments of inertia for a beam element should represent those of a beam element. Though seemingly obvious, this may not always be the case. For example, an element (such as a beam, rod, spring) need not truly represent the actual structural member. Instead, seeing as these elements are simply mathematical representations, they may be used to represent anything the user intends. Elements may be used in creative manners, but in doing so, may run the risk of having “unreasonable” numerical inputs.

For a given element type, if the numerical inputs are excessively large (or small) compared to the “typical” inputs, there may be numerical roundoff errors. These errors may (or may not) be indicated in the solver’s output files; and some solvers are more tolerant/robust than others to the range of acceptable inputs. The user should be aware of this shortcoming and create models that have “reasonable” inputs.

CHECKING MODELS

After a run is complete, the results should always be checked. Some of the considerations are:

- Compare the results to classical solutions. Unless the finite element model is a test case that directly compared to a classical solution (a “hand” calculation), the results will not be identical. However, certain aspects (or portions) of the model may be compared to classical solution. Sometimes only an approximate can be made via the classical solution. Although approximate, it can serve as a valuable way to perform a “sanity check” on the model.
- Displacements should be viewed. Determine if the boundary conditions and forces were applied as intended. Ensure that the general displacement is as expected.
- Ensure that the applied mechanical loads are balanced with the reactions. However, if a temperature delta or enforced displacements are present, the applied mechanical loads may not balance with the reactions.
- View the output file from the solver and observe any warnings, errors, etc.
- For some models, a convergence study should be considered. This means the model should be meshed again with a finer mesh (at least in the areas of interest or areas that affect the model). The results between the original mesh and the refined mesh should be sufficiently similar, which is indicative of convergence.