

# Documentation and Validation of EveryCalc's Beam Tool

Thaddeus Hughes  
hughes.thad@gmail.com  
thaddeus-maximus.github.io

May 30, 2020

## Abstract

Beams are common structures for structural analysis. As such many analytical formulas for specific use cases exist. Complex cases can also be computed by use of static superposition, or the utilization of a finite-element (FE) model. I created a web interface for a rudimentary beam FE model.

## 1 How does FE work?

A finite-element model works generally by solving a matrix equation of the form

$$\{F\} = [K]\{q\}. \quad (1)$$

Where  $F$  is the *load vector*,  $[K]$  is the *stiffness matrix*, and  $q$  is the *displacement vector*. This may be recognized as simply a matrix form of a spring equation  $F = k\delta$ , which it is! The FE model works by splitting a large component into several small springs (*elements*) with endpoints (*nodes*).

There are a few methods to solve this equation. One simple one is to multiply both sides by the matrix inverse (since there is no direct equivalent of division with matrices). This method does not scale well- but it works with the few elements we will need for this calculator.

$$[K]^{-1}\{F\} = [K]^{-1}[K]\{q\} = \{q\} \quad (2)$$

There are many different forms of stiffness, depending on the exact element used.

We will use a 2D beam element derived from *Euler-Bernoulli beam theory*. This element has four degrees of freedom: vertical deflection  $v$  and rotation  $\psi$  at each end node.



Figure 1: A beam with a fixed support, a pinned support, and a force load.

It can be found that the equation derived from euler-bernoulli beam theory is

$$\begin{Bmatrix} F_1 \\ M_1 \\ F_2 \\ M_2 \end{Bmatrix} = \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \begin{Bmatrix} v_1 \\ \psi_1 \\ v_2 \\ \psi_2 \end{Bmatrix} \quad (3)$$

This is just one element though! How do we link the elements together?

## 2 Direct Assembly

One simple way is by *direct assembly*. It can be noticed that (roughly) the matrix looks something like

$$\begin{Bmatrix} F_1 \\ F_2 \end{Bmatrix} = \begin{bmatrix} K & -K \\ -K & K \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} \quad (4)$$

If we had two elements linked together like so:

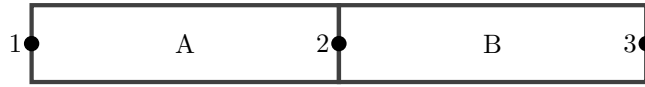


Figure 2: Two connected beams.

These two beams share node 2. That is to say, they both see the forces from node 2, and the deflection at node 2. This leads us to combine the equations for each element as so:

$$\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \end{Bmatrix} = \begin{bmatrix} K_A & -K_A & 0 \\ -K_A & K_A + K_B & -K_B \\ 0 & -K_B & K_B \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} \quad (5)$$

We've linked the nodes together, but we now need to constrain them. These beams are currently floating in space- we need to anchor them otherwise our displacements are meaningless (furthermore, the matrix  $[K]$  would be singular and unsolvable). For example, let's apply the constraint that node 1 is fixed; that is,  $u_1 = 0$ . As a result, the associated column in  $[K]$  (the first column) does not matter. Any loads applied to the node also do not matter since they would be absorbed by the fixed constraint. This allows us to remove columns and rows 1 of the stiffness matrix, degrees of freedom 1, and load 1, changing our equation to:

$$\begin{Bmatrix} F_2 \\ F_3 \end{Bmatrix} = \begin{bmatrix} -K_A & K_A + K_B & -K_B \\ 0 & -K_B & K_B \end{bmatrix} \begin{Bmatrix} u_2 \\ u_3 \end{Bmatrix} \quad (6)$$

At this point we could plug in values  $F_2$  and  $F_3$ , and solve with the matrix inverse method.

### 3 Application to Beams

Let's show how this would work with a beam. Consider the following example:

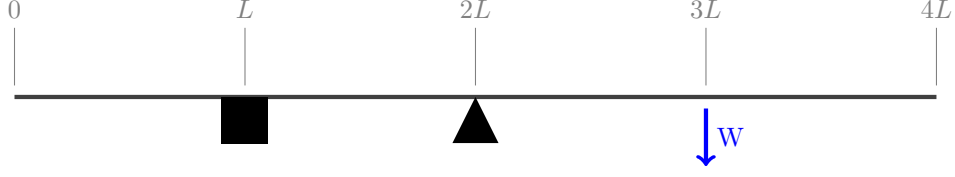


Figure 3: A beam with a fixed support, a pinned support, and a force load.

We can start by creating a general stiffness matrix with the general assembly method:

$$[K] = \begin{bmatrix} 12 & 6L & -12 & 6L & & & & & & \\ 6L & 4L^2 & -6L & 2L^2 & & & & & & \\ -12 & -6L & 24 & 12L & -12 & 6L & & & & \\ 6L & 2L^2 & 12L & 8L^2 & -6L & 2L^2 & & & & \\ & & -12 & -6L & 24 & 12L & -12 & 6L & & \\ & & 6L & 2L^2 & 12L & 8L^2 & -6L & 2L^2 & & \\ & & & & -12 & -6L & 24 & 12L & -12 & 6L \\ & & & & 6L & 2L^2 & 12L & 8L^2 & -6L & 2L^2 \\ & & & & & & -12 & -6L & 12 & -6L \\ & & & & & & 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \quad (7)$$

Zeros have been omitted from matrix to aid in readability.

The load vector is simply

$$F = \begin{Bmatrix} F_1 \\ M_1 \\ F_2 \\ M_2 \\ F_3 \\ M_3 \\ F_4 \\ M_4 \\ F_5 \\ M_5 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix}. \quad (8)$$

Combining this yields the full, unconstrained equation

$$\begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} 12 & 6L & -12 & 6L & & & & & & \\ 6L & 4L^2 & -6L & 2L^2 & & & & & & \\ -12 & -6L & 24 & 0 & -12 & 6L & & & & \\ 6L & 2L^2 & 0 & 8L^2 & -6L & 2L^2 & & & & \\ & & -12 & -6L & 24 & 0 & -12 & 6L & & \\ & & 6L & 2L^2 & 0 & 8L^2 & -6L & 2L^2 & & \\ & & & & -12 & -6L & 24 & 0 & -12 & 6L \\ & & & & 6L & 2L^2 & 0 & 8L^2 & -6L & 2L^2 \\ & & & & & & -12 & -6L & 12 & -6L \\ & & & & & & 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \begin{Bmatrix} v_1 \\ \psi_1 \\ v_2 \\ \psi_2 \\ v_3 \\ \psi_3 \\ v_4 \\ \psi_4 \\ v_5 \\ \psi_5 \end{Bmatrix}. \quad (9)$$

The fixed support at node 2 removes degrees of freedom  $v_2, \psi_2$ . The pinned support at node 3 removes only  $v_3$  (pin still permits rotation).

$$\begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} 12 & 6L & -12 & 6L & & & & & & \\ 6L & 4L^2 & -6L & 2L^2 & & & & & & \\ -12 & -6L & 24 & 0 & 12 & 6L & & & & \\ 6L & 2L^2 & 0 & 8L^2 & -6L & 2L^2 & & & & \\ -12 & -6L & 24 & 0 & -12 & 6L & & & & \\ & 6L & 2L^2 & 0 & 8L^2 & -6L & 2L^2 & & & \\ & & & -12 & -6L & 24 & 0 & -12 & 6L & \\ & & & 6L & 2L^2 & 0 & 8L^2 & -6L & 2L^2 & \\ & & & & & -12 & -6L & 12 & -6L & \\ & & & & & 6L & 2L^2 & -6L & 4L^2 & \end{bmatrix} \begin{Bmatrix} v_1 \\ \psi_1 \\ v_2 \\ \psi_2 \\ v_3 \\ \psi_3 \\ v_4 \\ \psi_4 \\ v_5 \\ \psi_5 \end{Bmatrix}$$

$$\begin{Bmatrix} 0 \\ 0 \\ 0 \\ -W \\ 0 \\ 0 \\ 0 \end{Bmatrix} = \begin{bmatrix} 12 & 6L & & & & & \\ 6L & 4L^2 & & & & & \\ & & 8L^2 & -6L & 2L^2 & & \\ & & -6L & 24 & 0 & -12 & 6L \\ & & 2L^2 & 0 & 8L^2 & -6L & 2L^2 \\ & & & -12 & -6L & 12 & -6L \\ & & & 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \begin{Bmatrix} v_1 \\ \psi_1 \\ \psi_3 \\ v_4 \\ \psi_4 \\ v_5 \\ \psi_5 \end{Bmatrix} \quad (10)$$

At this point, the matrix equation could be solved to achieve the nodal displacement vector. Afterwards, the resulting displacement matrix could be used with the unconstrained stiffness matrix to determine the loads at the constraints.

## 4 Prescribing Displacements

But, what if instead, we wanted to prescribe a displacement rather than a force? This throws a wrench in this whole process... or at least would mean we'd have to go back to the drawing board, and try *matrix partitioning*. Let's consider a different setup, of the form

$$\begin{Bmatrix} \{F_u\} \\ \{F_p\} \end{Bmatrix} = \begin{bmatrix} [K_{u,u}] & [K_{u,p}] \\ [K_{p,u}] & [K_{p,p}] \end{bmatrix} \begin{Bmatrix} \{q_u\} \\ \{q_p\} \end{Bmatrix}, \quad (11)$$

where  $\{F_u\}$  and  $\{F_p\}$  are the portions of the load vector corresponding to the unprescribed and prescribed-displacement portions, respectively. The same holds for  $\{q_u\}$  and  $\{q_p\}$ . There's no *true* sub-vectoring going on here, the brackets are just to denote that these are bits of matrices inside the larger matrices, not just scalar values.

We can expand out first row of the matrix equation to find that

$$\{F_u\} = [K_{u,u}]\{q_u\} + [K_{u,p}]\{q_p\} \quad (12)$$

Solving this yields

$$[K_{u,u}]\{q_u\} = \{F_u\} - [K_{u,p}]\{q_p\} \quad (13)$$

which is of the form  $[M]\{x\} = \{b\}$  (since  $\{q_p\}$  is the prescribed displacements), so can be solved.

After finding the non-prescribed displacements, the second row of the matrix equation can be expanded to

$$\{F_p\} = [K_{p,u}]\{q_u\} + [K_{p,p}]\{q_p\} \quad (14)$$

which can be used to compute the forces required to produce the prescribed displacements.

We would then simply need to re-build the force and displacement vectors, and then we can proceed to post-processing.

## 5 Post Processing and Shape Functions

Usually, a displacement *field* is desired rather than merely the nodal values. Getting deflection at arbitrary points requires the use of *shape functions*. In the derivation of the beam stiffness matrix (which this paper does not cover), particular *shape functions* were used: the Hermite shape functions. These are nondimensional equations that represent the contribution of one *nodal* degree of freedom to the displacement *field* in an element. By combining these displacement fields, we can determine the total displacement field.

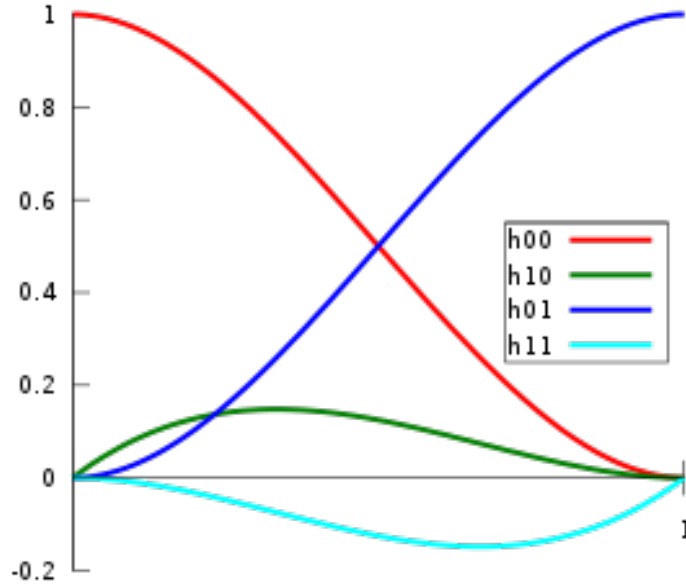


Figure 4: Hermite shape functions

These hermite shape functions can be represented as:

$$N = \begin{Bmatrix} 1/4(1 - \zeta^2(2 + \zeta)) \\ L/8(1 - \zeta)^2(\zeta + 1) \\ 1/4(1 + \zeta)^2(2 - \zeta) \\ L/8(1 + \zeta)^2(\zeta - 1) \end{Bmatrix}^T \quad (15)$$

Where  $L$  is the element length and  $\zeta$  is a nondimensional parameter that is -1 at the left of the element, +1 at the right, and 0 in the center.

$$\zeta = 2\frac{x}{L} - 1 \quad (16)$$

Where  $x$  is the distance from the left of the element.

The functions can be used to find the displacement field  $v$ :

$$v = \{N\}\{q\} = \begin{Bmatrix} 1/4(1 - \zeta^2(2 + \zeta)) \\ L/8(1 - \zeta)^2(\zeta + 1) \\ 1/4(1 + \zeta)^2(2 - \zeta) \\ L/8(1 + \zeta)^2(\zeta - 1) \end{Bmatrix}^T \begin{Bmatrix} v_1 \\ \psi_1 \\ v_2 \\ \psi_2 \end{Bmatrix} \quad (17)$$

This helps us plot the displacement field, but what about the slope, bending moment, and shear force?

The slope  $\theta$  is simply the derivative of the displacement field  $v$  with respect to position  $x$ . The bending moment  $M$  and shear force  $V$  can also be found from Euler-Bernoulli beam theory.

$$\theta = \frac{dv}{dx} \quad (18)$$

$$M = EI \frac{d^2v}{dx^2} \quad (19)$$

$$V = \frac{d}{dx}(EI \frac{d^2v}{dx^2}) = EI \frac{d^3v}{dx^3} \quad (20)$$

Assuming elastic modulus  $E$  and second moment of area  $I$  are constant over the element.

To find these derivatives of  $v$ , we can use the shape functions, ignoring the nodal displacements since they do not vary with  $x$ .

$$\frac{dv}{dx} = \frac{d}{dx}[\{N\}\{q\}] = \frac{d\{N\}}{dx}\{q\} \quad (21)$$

The chain rule can be used to help find these derivatives.

$$\frac{d\{N\}}{dx} = \frac{d\{N\}}{d\zeta} \frac{d\zeta}{dx} \quad (22)$$

$$\frac{d^2\{N\}}{dx^2} = \frac{d^2\{N\}}{d\zeta^2} \left(\frac{d\zeta}{dx}\right)^2 + \frac{d\{N\}}{d\zeta} \frac{d^2\zeta}{dx^2} \quad (23)$$

$$\frac{d^3\{N\}}{dx^3} = \frac{d^3\{N\}}{d\zeta^3} \left(\frac{d\zeta}{dx}\right)^3 + 3 \frac{d^2\{N\}}{d\zeta^2} \frac{d\zeta}{dx} \frac{d^2\zeta}{dx^2} + \frac{d\{N\}}{d\zeta} \frac{d^3\zeta}{dx^3} \quad (24)$$

Looks ugly! But luckily, the higher order derivatives of  $\zeta$  are zero.

$$\frac{d\zeta}{dx} = \frac{2}{L} \quad (25)$$

$$\frac{d^2\zeta}{dx^2} = \frac{d^3\zeta}{dx^3} = 0 \quad (26)$$

$$\frac{d\{N\}}{dx} = \frac{d\{N\}}{d\zeta} \frac{2}{L} \quad (27)$$

$$\frac{d^2\{N\}}{dx^2} = \frac{d^2\{N\}}{d\zeta^2} \left(\frac{2}{L}\right)^2 \quad (28)$$

$$\frac{d^3\{N\}}{dx^3} = \frac{d^3\{N\}}{d\zeta^3} \left(\frac{2}{L}\right)^3 \quad (29)$$

The shape function derivatives are

$$\frac{d\{N\}}{dx} = \begin{Bmatrix} 3/4(\zeta^2 - 1)2/L \\ L/8(3\zeta^2 - 2\zeta - 1)2/L \\ 3/4(\zeta^2 - 1)2/L \\ L/8(3\zeta^2 + 2\zeta - 1)2/L \end{Bmatrix}^T \quad \frac{d^2\{N\}}{dx^2} = \begin{Bmatrix} (6\zeta)/L^2 \\ (3\zeta - 1)/L \\ -(6\zeta)/L^2 \\ (3\zeta + 1)/L \end{Bmatrix}^T \quad \frac{d^3\{N\}}{dx^3} = \begin{Bmatrix} 12/L^3 \\ 6/L^2 \\ -12/L^3 \\ 6/L^2 \end{Bmatrix}^T \quad (30)$$

## 6 Generalization

How does this get generalized to different, arbitrary, and variable scenarios?

- All of the loads and constraints are gathered and binned by type. Each load is given a node id.
- The position along the beam for each load or constraint is stored by node id.
- The node ids are sorted by the order of positions.
- Close nodes (those that are within 1/100 of the beam length) are merged so that the corresponding load/constraints share the same node id. The old node id and position is deleted.
- Beam elements are produced between the first and second node, second and third, and so forth.
- The force matrix is populated with forces
- Direct assembly is performed on the beam elements
- Degrees of freedom corresponding to the node ids where pinned/fixed nodes exist are struck and removed from the stiffness and force matrices.
- The matrix system of equations is solved
- The resulting nodal displacements are used in conjunction with the shape functions to interpolate a displacement field.

## 7 Sanity Check: are the Shape Functions Appropriate?

Usually finite element methods are used as approximations by employing the use of many, many elements to approximate the true displacement field. Beam elements, though, are somewhat unique in that *under certain circumstances* they will model the underlying displacement field exactly. Let's return to Euler-Bernoulli beam theory. This theory states that the displacement field  $q$  can be represented as

$$\frac{d^2}{dx^2} EI \frac{d^2 v}{dx^2} = q, \quad (31)$$

where  $q$  is a distributed load (i.e. with dimensions of force per unit length). Euler-Bernoulli beam theory has several assumptions wrapped up in it, such as the requirement that plane sections remain plane, and no large deflections.

My calculator does not model distributed loads, so  $q$  is always zero. Additionally, the cross-section and elastic modulus remain constant, so  $E$  and  $I$  can be moved outside of the derivative, leaving

$$EI \frac{d^4 v}{dx^4} = 0 \quad (32)$$

$$\frac{d^4 v}{dx^4} = 0 \quad (33)$$

If we integrate both sides repeatedly, we can get the form of  $v$ .

$$\int \frac{d^4 v}{dx^4} dx = \int 0 dx \quad (34)$$

$$\int \frac{d^3 v}{dx^3} dx = \int c_1 dx \quad (35)$$

$$\int \frac{d^2 v}{dx^2} dx = \int c_1 x + c_2 dx \quad (36)$$

$$\int \frac{dv}{dx} dx = \int c_1 x^2 + c_2 x + c_3 dx \quad (37)$$

$$v = c_1 x^3 + c_2 x^2 + c_3 x + c_4 \quad (38)$$

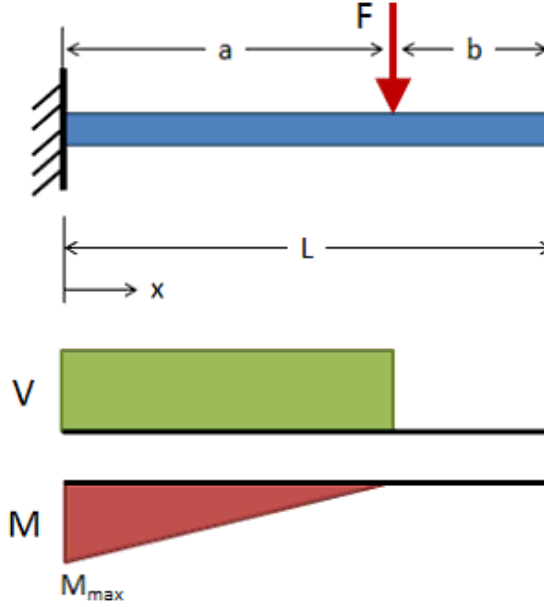
The shape of  $v$  under the assumptions of constant  $E$  and  $I$  with only point loads is a cubic polynomial, a function with four degrees of freedom. What else has four degrees of freedom? Our elements! Using beam elements provides a sufficient quantity of degrees of freedom to provide exact analytical solutions to problems where Euler-Bernoulli beam theory is appropriate, and require point loads and constant cross sections. This may sound restrictive, but encompasses nearly all applications of beams, which is why they are such a powerful modeling tool even in full-blown FE applications.

## 8 Validation Examples

Comparison to analytical solutions is a necessary component when validating any FE model. [MechaniCalc](#) has some good analytical solutions I will compare to.



## 8.1 Cantilevered, Intermediate Load



Let  $E = 69GPa$  (aluminum),  $L = 100mm$ , round bar with  $d = 5mm$ ,  $F = 200N$ ,  $a = 60mm$ .

$$I = \frac{\pi}{64}d^4 = \frac{\pi}{64}(0.100m)^4 = 3.06796 \times 10^{-11}m^4 \quad (39)$$

$$\delta_a = -\frac{Fa^2}{6EI}(3L - a) = -\frac{200N \times (0.06m)^2}{6 \times 69GPa \times 30.6796mm^4}(3 \times 0.06m - 0.06m) = 6.802mm \quad (40)$$

$$\delta_{end} = -\frac{Fa^2}{6EI}(3L - a) = -\frac{200N \times (0.06m)^2}{6 \times 69GPa \times 30.6796mm^4}(3 \times 0.1m - 0.06m) = 13.6048mm \quad (41)$$

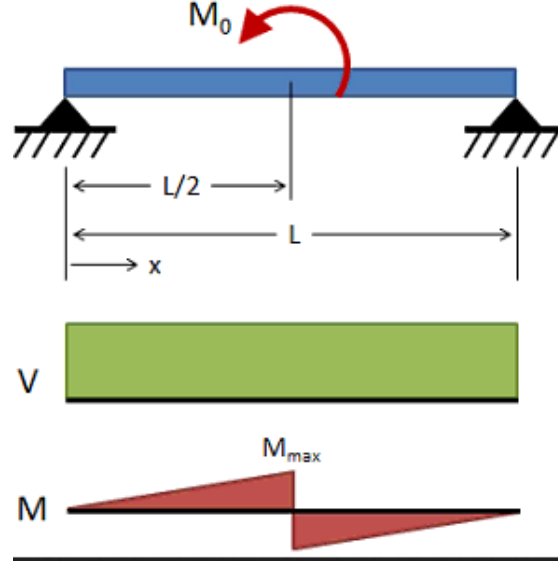
$$\theta_{a/2} = -\frac{F(a/2)^2}{2EI}(2L - a/2) = \frac{200N \times .03m}{2 \times 69GPa \times 30.6796mm^4}(2 \times 0.1m - .03m) = 7.305 \text{ deg} \quad (42)$$

$$V_{0 \rightarrow a} = F = 200 \text{ N} \quad (43)$$

$$M_{x=0} = -F \times a = -200N \times .06m = -12 \text{ Nm} \quad (44)$$

You can plug in the values to the calculator and verify this load case for yourself. I found all of the above numbers to be accurate.

## 8.2 Simply Supported, Center Moment



Let  $E = 27557 \text{ ksi}$  (steel),  $L = 6 \text{ in}$ , box bar with  $w = 0.5 \text{ in}$ ,  $h = 2 \text{ in}$ ,  $M = 1200 \text{ ft} - \text{lb}$ .

$$I = \frac{1}{12}bh^3 = 1/3 \text{ in}^4 \quad (45)$$

$$\delta_{x=1.25 \text{ in}} = \frac{-Mx}{24LEI}(L^2 - 4x^2) = \frac{-12 \text{ ft lb} \cdot 1.25 \text{ in}}{24 \times 6 \text{ in} \cdot 27557 \text{ ksi} \cdot 1/3 \text{ in}^4}((6 \text{ in})^2 - 4(1.25 \text{ in})^2) = -.4048 \text{ thou} \quad (46)$$

$$\theta_1 = \frac{-ML}{24EI} = \frac{-1200 \text{ ft lb} \cdot 6 \text{ in}}{24 \times 27557 \text{ ksi} \cdot 1/3 \text{ in}^4} = -.02246 \text{ deg} \quad (47)$$

$$V = M/L = 1200 \text{ ft} - \text{lb} / 6 \text{ in} = 2400 \text{ lb} \quad (48)$$

$$M_{x=L/2} = M/2 = 1200 \text{ ft} - \text{lb} / 2 = 600 \text{ ft} - \text{lb} \quad (49)$$

You can plug in the values to the calculator and verify this load case for yourself. I found all of the above numbers to be accurate.