

General Truss Solver

Aakash Yadav

Email: me16b001@iittp.ac.in

Indian Institute of Technology Tirupati

May 2019

1 Introduction

A Truss is an important structure type in structural engineering. A Truss is essentially a triangulated system of members that are structured and connected in a way such that they only incur axial force. These members are considered two-force members as the forces are only applied at either end of the member, resulting in either a compression or tension force. They are commonly used as bridge designs, given their ability to efficiently span long distances (Refer figure 1).

In this work, we shall restrain ourselves to planar trusses. A planar truss is one where all members and nodes lie within a two-dimensional plane. The same idea can be extended for solving space trusses i.e. 3D trusses. This problem is solved using an in-house Python code developed by the author (<https://github.com/AakashSYadav/TrussSolver>), which solves both deflections and the reactions at the nodes using the Finite Element Method (FEM) approach [2].

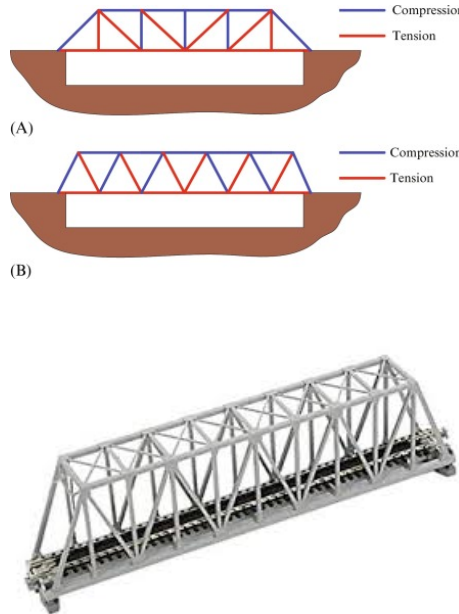


Figure 1: Common applications of Truss structures

In section 2, we bring equations governing the truss, form the required matrix form and solve a sample problem manually by hand. The section will provide insights to boundary conditions at the nodes. The problem solved manually in section 2 can be used as a template forming the generic/softcoded algorithm in section 3. In section 4, we present the results obtained using the inhouse code and compare them with the manual solution. In the final section we present the conclusions. All the algorithms implemented in this work can be found in the appendix (or Attachment).

2 Sample problem and its Manual Solution

We consider a two-element truss problem as shown in Fig. 2, with the nodes being assigned arbitrary “global” numbers from 1 to 3. Since each node can in general move in two directions, there are $3 \times 2 = 6$ total degrees of freedom in the problem. The global stiffness matrix will then be a 6×6 array relating the six displacements to the six externally applied forces. Only one of the displacements is unknown in this case, since all but the vertical displacement of node 2 (degree of freedom number 4) is constrained to be zero. Figure 3 shows the global numbers, and also “local” numbers for each individual element. Using the local numbers, the 4×4 element stiffness matrix of each of the two elements can be evaluated according to equation shown in figure 4, where $c = \cos\theta$ and $s = \sin\theta$

$$\theta = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \quad (1)$$

Here x_1, x_2, y_1, y_2 are the local node numbers for the truss element.

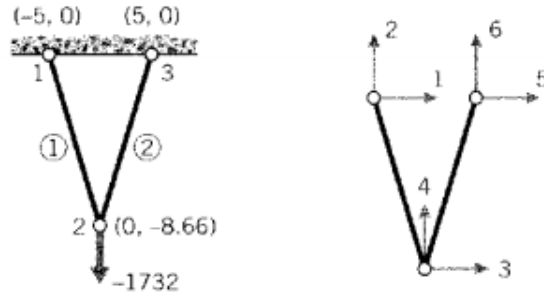


Figure 2: A two membered truss with global nodes

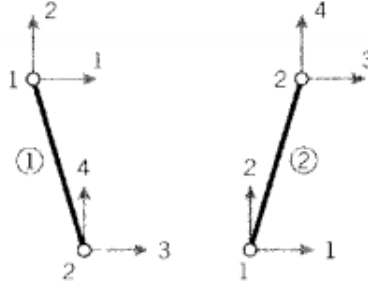


Figure 3: Local nodes for the problem in fig 2

$$\begin{Bmatrix} f_{xi} \\ f_{yi} \\ f_{xj} \\ f_{yj} \end{Bmatrix} = \frac{AE}{L} \begin{bmatrix} c^2 & cs & -c^2 & -cs \\ cs & s^2 & -cs & -s^2 \\ -c^2 & -cs & c^2 & cs \\ -cs & -s^2 & cs & s^2 \end{bmatrix} \begin{Bmatrix} u_i \\ v_i \\ u_j \\ v_j \end{Bmatrix}$$

Figure 4: Equation for the element truss in global coordinates

Using the equation shown in figure 4 we can write the global k matrices (ref fig 5) for the truss elements. We can use the equation 1 to find the angle for each truss element. For the given problem we have rigidity modulus of both elements as $E = 10Mpsi$ and both have area, $A = 0.1in^2$.

Each of the local degrees of freedom for the truss elements can be matched to one of the global degrees of the overall problem and we can form the connectivity matrix (ref fig 6) which can be used for forming the assembled matrix.

$$k^{(1)} = \begin{bmatrix} 25.00 & -43.30 & -25.00 & 43.30 \\ -43.30 & 75.00 & 43.30 & -75.00 \\ -25.00 & 43.30 & 25.00 & -43.30 \\ 43.30 & -75.00 & -43.30 & 75.00 \end{bmatrix} \times 10^3$$

$$k^{(2)} = \begin{bmatrix} 25.00 & 43.30 & -25.00 & -43.30 \\ 43.30 & 75.00 & -43.30 & -75.00 \\ -25.00 & -43.30 & 25.00 & 43.30 \\ -43.30 & -75.00 & 43.30 & 75.00 \end{bmatrix} \times 10^3$$

Figure 5: Global stiffness matrix for truss elements

local	global, element 1	global, element 2
1	1	3
2	2	4
3	3	5
4	4	6

Figure 6: Connectivity matrix

$$K = \begin{bmatrix} k_{11}^{(1)} & k_{12}^{(1)} & k_{13}^{(1)} & k_{14}^{(1)} & 0 & 0 \\ k_{21}^{(1)} & k_{22}^{(1)} & k_{23}^{(1)} & k_{24}^{(1)} & 0 & 0 \\ k_{31}^{(1)} & k_{32}^{(1)} & k_{33}^{(1)} + k_{11}^{(2)} & k_{34}^{(1)} + k_{12}^{(2)} & k_{13}^{(2)} & k_{14}^{(2)} \\ k_{41}^{(1)} & k_{42}^{(1)} & k_{43}^{(1)} + k_{21}^{(2)} & k_{44}^{(1)} + k_{22}^{(2)} & k_{23}^{(2)} & k_{24}^{(2)} \\ 0 & 0 & k_{31}^{(2)} & k_{32}^{(2)} & k_{33}^{(2)} & k_{34}^{(2)} \\ 0 & 0 & k_{41}^{(2)} & k_{42}^{(2)} & k_{43}^{(2)} & k_{44}^{(2)} \end{bmatrix}$$

Figure 7: Assembly of element matrices

Using the connectivity matrix we can write the assembled stiffness matrix as shown in figure 7. Fig 8 shows the final system, it can be solved for u_4 as $u_4 = -0.01155in$.

$$10^3 \begin{bmatrix} 25.0 & -43.3 & -25.0 & 43.3 & 0.0 & 0.00 \\ -43.3 & 75.0 & 43.3 & -75.0 & 0.0 & 0.00 \\ -25.0 & 43.3 & 50.0 & 0.0 & -25.0 & -43.30 \\ 43.3 & -75.0 & 0.0 & 150.0 & -43.3 & -75.00 \\ 0.0 & 0.0 & -25.0 & -43.3 & 25.0 & 43.30 \\ 0.0 & 0.0 & -43.3 & -75.0 & 43.3 & 75.00 \end{bmatrix} \begin{Bmatrix} 0 \\ 0 \\ 0 \\ u_4 \\ 0 \\ 0 \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ f_3 \\ -1732 \\ f_5 \\ f_5 \end{Bmatrix}$$

Figure 8: Matrix equation obtained by using the assembly

3 Methodology and Algorithm

In this section we provide the algorithms which have been used for solving this problem.

No.	Steps
1	Get the nodes of each element truss from the user
2	Calculate angle and length of the trusses
3	Find the inque number of nodes
4	Calculate the global k matrices for the elements
5	Form the connectivity matrix
6	Assemble the matrices using the connectivity matrices
7	Find the unknown parameters

Table 1: Algorithm

4 Results

In this final section we present the results. The results obtained by the program are shown below.

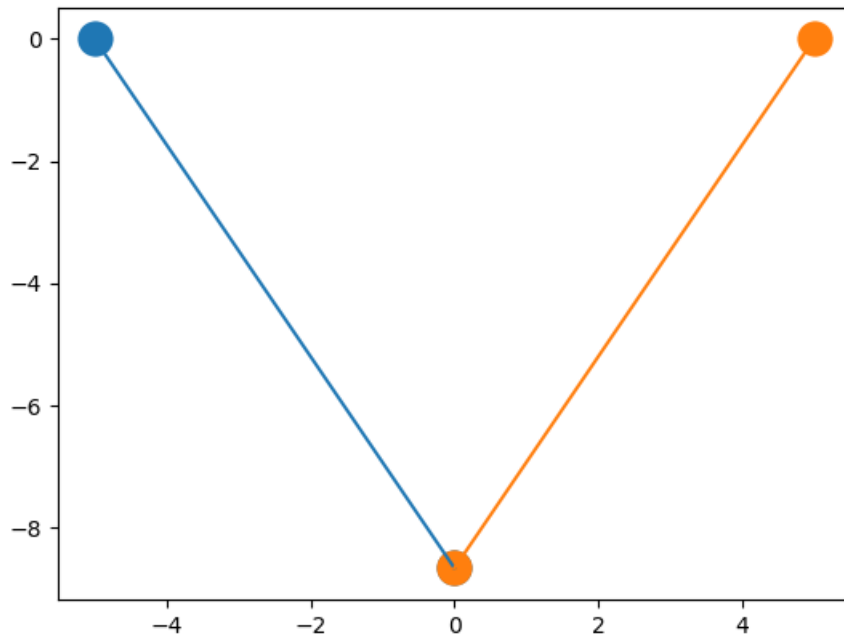


Figure 9: Figure showing the geometry generated by the program

```
aakash@aakash:~/SEM6/FEM/FEM_Project$  
python3 1.py  
[[ 15000.10999637]]  
['-1732']  
([0, 0, 0, -0.11546581994526622, 0, 0])
```

Figure 10: Result obtained by the program

5 Discussion

The results obtained by both the methods are inline with each other, hence we can conclude the validation of the code. There have been an active research in this area to optimize the algorithm and speed up the solution procedure citepCheng1995 [1] [3].

References

- [1] Wei He, Yinqiao Zhu, and Renlong Chen. Brief research on arch hinge of the steel truss arch bridge by contact problem under local stress. *IOP Conference Series: Earth and Environmental Science*, 69:012190, jun 2017.
- [2] J. N. (Junuthula Narasimha) Reddy. *An introduction to the finite element method*. McGraw-Hill series in mechanical engineering. Third edition.. edition, 2006.
- [3] Lingyun Wei, Tianbing Tang, Xianghong Xie, and Wenjie Shen. Truss optimization on shape and sizing with frequency constraints based on parallel genetic algorithm. *Struct. Multidiscip. Optim.*, 43(5):665–682, May 2011.