COE 321K: Arbitrary Any-Dimensional Truss Solver

Ashton Cole

March 3, 2023

1 Introduction

This assignment computes the reaction of an arbitrary any-dimensial truss system to applied forces and constrained displacements within a single Python script. It assumes the joints are pins, that bars are light, and that strains are small and proportional to stresses. It accepts input through files in the form provided in class, and outputs to the terminal and files. The product is applied to an example from class in Section 3.

1.1 Changes

This script now supports trusses of any dimension by using the following global-coordinate local-element stiffness matrix generalization.

$$\mathbf{S} = \begin{bmatrix} c_1^2 & c_1c_2 & \cdots & c_1c_N \\ c_2c_1 & c_2^2 & \cdots & c_2c_N \\ \vdots & \vdots & \ddots & \vdots \\ c_Nc_1 & c_Nc_2 & \cdots & c_N^2 \end{bmatrix}$$

$$\mathbf{K} = \frac{EA}{L}\begin{bmatrix} \mathbf{S} & -\mathbf{S} \\ -\mathbf{S} & \mathbf{S} \end{bmatrix}$$

The script has also been adjusted to accept input files with entries delimited by any amount whitespace, i.e. tabs and spaces.

2 Setup and Execution

The files `trussolverND.py` and `mygui.py` are needed to run the solver. Additionally, the modules `numpy` and `tkinter` must be installed for the matrix operations and user interface, respectively. The module `tkinter` is a part of the Python standard library, but if a user interface is not desired, the method `solve_case()` may be called directly. It requires manually input file paths and

a boolean 6-tuple of whether the user wants to solve for reactionary displacements (required), reactionary forces, internal strains, internal stresses, internal forces, and critical failure elements, respectively. By default, only reactionary displacements are solved.

After this, the main script `trussolverND.py` can simply be interpreted in a Python 3 environment. The user is prompted to select input files and an output directory. These may be selected individually, or automatically with the "Select Directory" button. This presumes that all of the input files are in the directory, named `nodes`, `elements`, `displacements`, and `forces`, and outputs results to the same location. Reactionary forces are output to `reactionary_forces`, reactionary displacements to `reactionary_displacements`, internal strains to `internal_strains`, internal stresses to `internal_stresses`, internal forces to `internal_forces`, and critical elements to `critical`, respectively.

2.1 Arranging Input Files

The nodes input file has the number of nodes and spatial dimensions separated by whitespace on the first line. Each subsequent line represents a node, with a unique ID and spatial coordinates consistent with its dimension separated by whitespace.

The elements input file has the number of elements on the first line. Each subsequent line represents an element, with a unique ID, two referenced node ID's, Young's modulus, and cross-sectional area separated by whitespace. Additionally, if yield, crushing, or buckling analyses are sought, each element must have a positive yield stress, negative crushing stress, and second moment of area appended at the end of the line, separated by whitespace.

The displacements input file has the number of displacements on the first line. Each subsequent line represents a single constraint, with a referenced node ID, dimension starting from 1, and quantity separated by whitespace. Note that this means displacements not aligned with an axis must be decomposed into separate components.

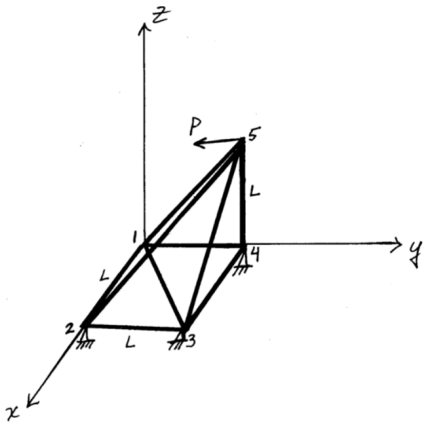
The forces input file has the number of applied forces on the first line. Each subsequent line represents a single applied force, with a referenced node ID, dimension starting from 1, and quantity separated by whitespace. Note that this means forces not aligned with an axis must be decomposed into separate components.

3 Results

To test this program, we consider the simple three-dimensional truss in Figure 1. In this scenario, all nodes but the first and fifth are pre-constrained to no displacements in all dimensions. The reference length L, applied force P, as well as the Young's moduli E and cross-sectional areas A of all of the elements, are assumed to be 1 for the purposes of the program. Because of this, and the

htb!

Figure 1: Simple Three-Dimensional Truss



fact that a truss system is linear, the reactionary displacements in Table 1 can be considered coefficients, where $u_i = C_i \frac{PL}{EA}$.

Table 1: Reactionary Displacements

Node	u	v	w
1	0	0	-8.024579547452817
2			
3			
4			
5	-2.828427124746189	-8.024579547452817	0

4 Analysis

The results seem reasonable. Node 5, being pulled in the negative y direction, moves that way. It also moves some in the negative x direction. This may relate to the particularly long element 25 resisting the compression associated with a pure negative-y translation. Looking at the internal forces, element 25 is in compression, while element 35 experiences tension to a lesser degree. The negative-z translation of node 1, equal to the negative-y translation of node 5, is consistent, given that element 15 experiences no internal forces, and deformations are assumed to be wery small and linear.