Programming Assignment 2
Finite Element Method

Team:

Tejal Ladage 210010026
Dhrutika Kalidindi 210010021
Ashwin Waghmare 210010060

Instructor:
Prof. Nikhil Hegde

CS601: Software Development for Scientific Computing

Indian Institute of Technology Dharwad

November 10, 2023

Contents

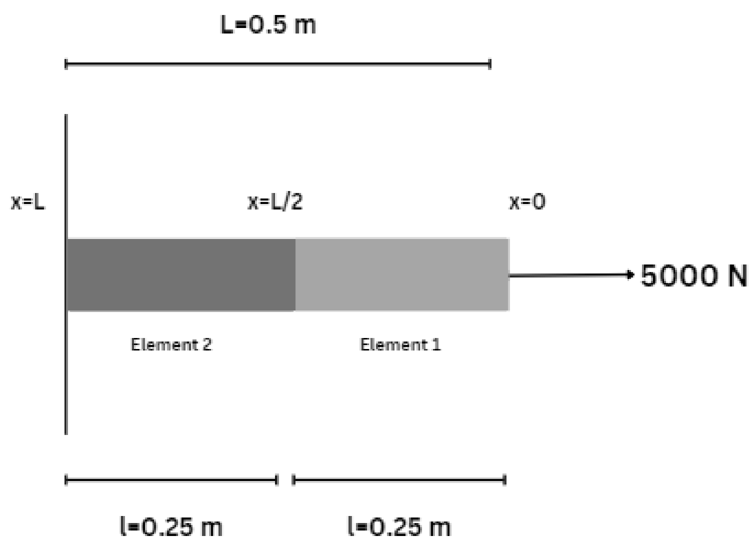
1 Methodology

Given a rod with cross sectional area A(x) and length L. The rod is subjected to a constant load P = 5000N at x = 0. At x = L the rod is fixed. The length of the rod is 0.5 m and the Young's modulus of the material of the rod is 70 GPa. Below are two subproblems:

1.1 Subproblem 1

The cross section of the rod is uniform with area
$A(x) = A = (12.5 * 10^-4)m^2$

Here we are finding displacements at nodes for 2 elements.



Element 1 and element 2 have same cross section, same young's modulus and same length which is equal to half of length of rod.

Element1 stiffness matrix

$$\frac{AE}{l}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}\begin{bmatrix}u1\\ u2\end{bmatrix} = \begin{bmatrix}F1\\ F2\end{bmatrix}$$

u1=Node displacement at $x = 0$
u2=Node displacement at $x = \frac{L}{2}$
F1=External force at $x = 0$

F2=External force at $x = \frac{L}{2}$

Element2 stiffness matrix

$$\frac{AE}{l}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}\begin{bmatrix}u2\\ u3\end{bmatrix} = \begin{bmatrix}F2\\ F3\end{bmatrix}$$

u2=Node displacement at $x = \frac{L}{2}$
u3=Node displacement at $x = L$
F2=External force at $x = \frac{L}{2}$
F3=External force at $x = L$

Combining Elemental Stiffness Matrix(ESM) and Force Vector Matrix of element 1 and element 2 to form **Global Stiffness Matrix(GSM)** and **Combined Force Vector Matrix**.

$$\frac{AE}{l}\begin{bmatrix}1 & -1 & 0\\ -1 & 1+1 & -1\\ 0 & -1 & 1\end{bmatrix}\begin{bmatrix}u1\\ u2\\ u3\end{bmatrix} = \begin{bmatrix}F1\\ F2\\ F3\end{bmatrix}$$

Applying **boundary conditions** and substituting values:
u3=0
F1=5000
F2=0

$$35 * 10^7\begin{bmatrix}1 & -1 & 0\\ -1 & 1+1 & -1\\ 0 & -1 & 1\end{bmatrix}\begin{bmatrix}u1\\ u2\\ 0\end{bmatrix} = \begin{bmatrix}5000\\ 0\\ F3\end{bmatrix}$$

Deleting last row and last column because value of u3(node displacement at x=L) is zero.

$$35 * 10^7\begin{bmatrix}1 & -1\\ -1 & 2\end{bmatrix}\begin{bmatrix}u1\\ u2\end{bmatrix} = \begin{bmatrix}5000\\ 0\end{bmatrix}$$

On solving the matrices we get below equations

$35 * 10^7(u1 - u2) = 5000$
$35 * 10^7(-u1 + 2u2) = 0$

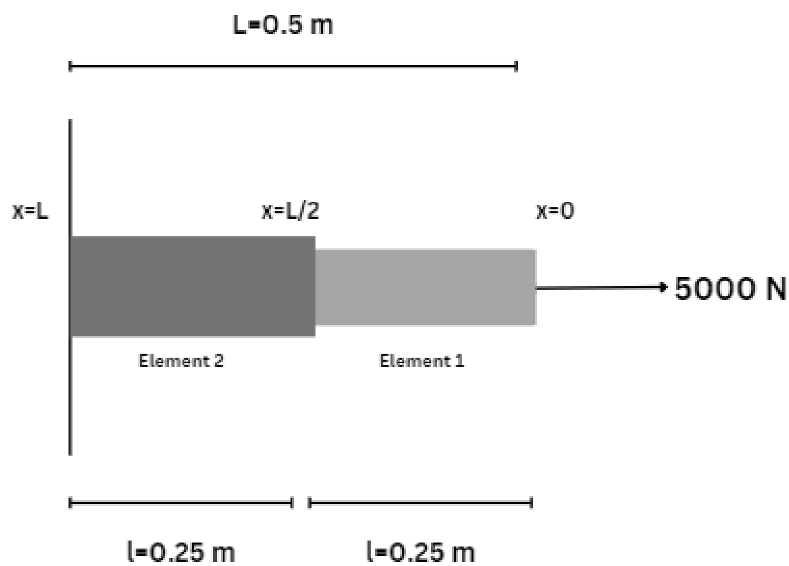
On solving the equations we get

$u1 = 0.2857 * 10^-4$
$u2 = 0.1428 * 10^-4$

1.2 Subproblem 2

The cross sectional area is given by the formula
$A(x) = A0(1 + \frac{x}{L})$
Here the cross section is not uniform, it increases linearly with x.

Here we are finding displacements at nodes for 2 elements.



Element 1 and element 2 have same young's modulus and same length which is equal to half of length of rod but different cross sectional areas.
Element 1 cross sectional area $A1 = A(1 + \frac{0}{L}) = A = 12.5 * 10^-4$
Element 2 cross sectional area $A2 = A(1 + \frac{L/2}{L}) = A * \frac{3}{2} = 18.75 * 10^-7$

Element1 stiffness matrix

$$\frac{A1E}{l} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} u1 \\ u2 \end{bmatrix} = \begin{bmatrix} F1 \\ F2 \end{bmatrix}$$

u1=Node displacement at $x = 0$
u2=Node displacement at $x = \frac{L}{2}$
F1=External force at $x = 0$
F2=External force at $x = \frac{L}{2}$

Element2 stiffness matrix

$$\frac{A2E}{l}\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix}\begin{bmatrix}u2\\ u3\end{bmatrix}=\begin{bmatrix}F2\\ F3\end{bmatrix}$$

u2=Node displacement at $x=\frac{L}{2}$
u3=Node displacement at $x=L$
F2=External force at $x=\frac{L}{2}$
F3=External force at $x=L$

Combining Elemental Stiffness Matrix(ESM) and Force Vector Matrix of element 1 and element 2 to form **Global Stiffness Matrix(GSM)** and **Combined Force Vector Matrix**.

$$\frac{E}{l}\begin{bmatrix}A1 & -A1 & 0\\ -A1 & A1+A2 & -A2\\ 0 & -A2 & A2\end{bmatrix}\begin{bmatrix}u1\\ u2\\ u3\end{bmatrix}=\begin{bmatrix}F1\\ F2\\ F3\end{bmatrix}$$

Applying **boundary conditions** and substituting values:
u3=0
F1=5000
F2=0

$$10^7\begin{bmatrix}35 & -35 & 0\\ -35 & 35+52.5 & -52.5\\ 0 & -52.5 & 52.5\end{bmatrix}\begin{bmatrix}u1\\ u2\\ 0\end{bmatrix}=\begin{bmatrix}5000\\ 0\\ F3\end{bmatrix}$$

Deleting last row and last column because value of u3(node displacement at x=L) is zero.

$$10^7\begin{bmatrix}35 & -35\\ -35 & 87.5\end{bmatrix}\begin{bmatrix}u1\\ u2\end{bmatrix}=\begin{bmatrix}5000\\ 0\end{bmatrix}$$

On solving the matrices we get below equations

$10^7(35u1-35u2)=5000$
$10^7(-35u1+87.5u2)=0$

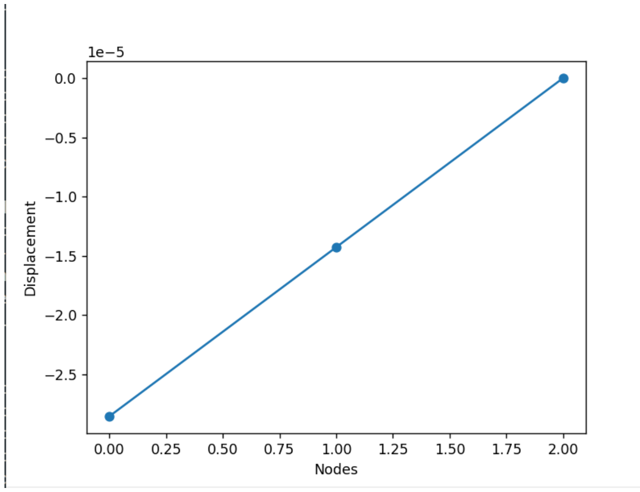
On solving the equations we get

$u1=0.2380*10^-4$
$u2=0.9523*10^-4$

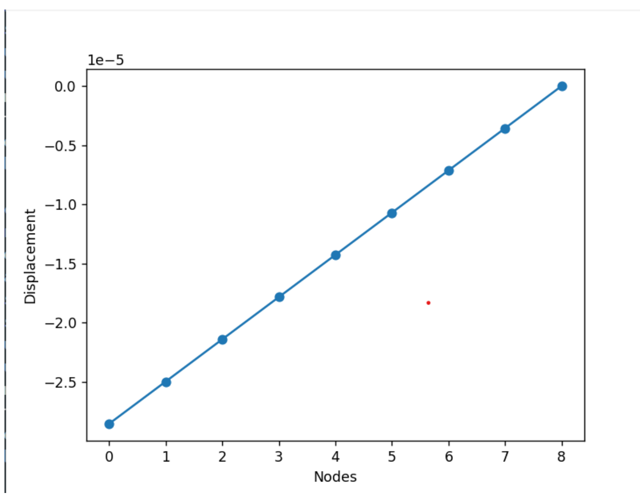
2 Experimental Results Plots

2.1 Subproblem1

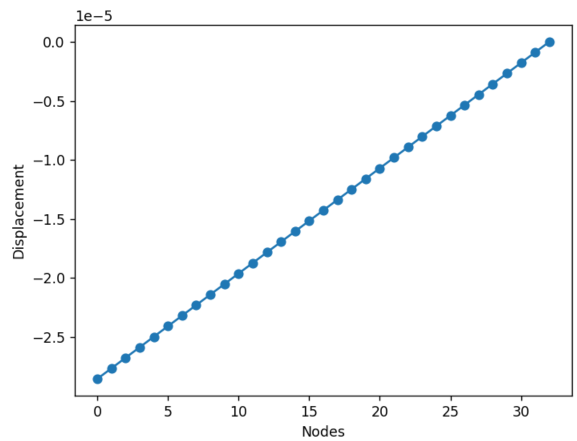
- PROB=1 N=2



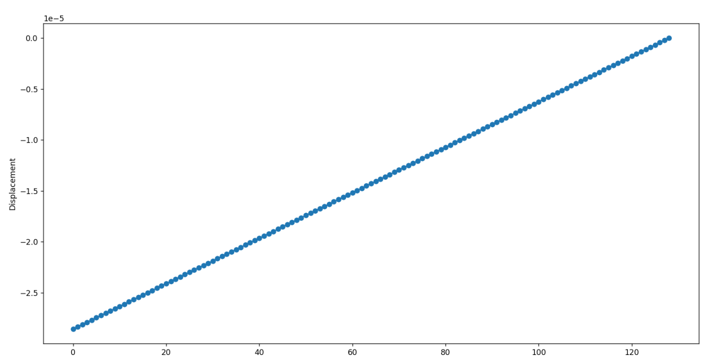
- PROB=1 N=8



- PROB=1 N=32

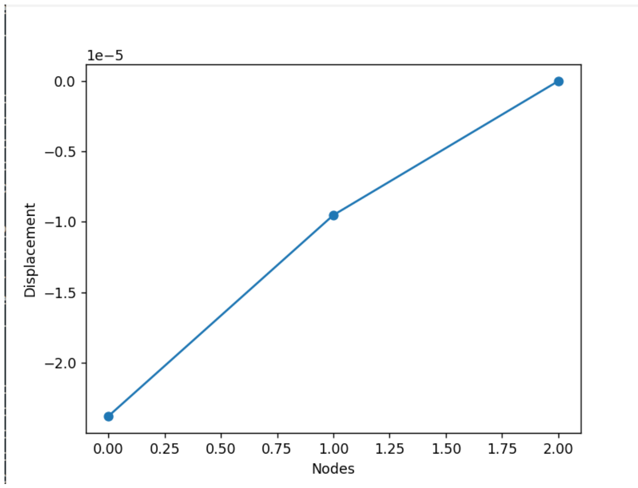


- PROB=1 N=128

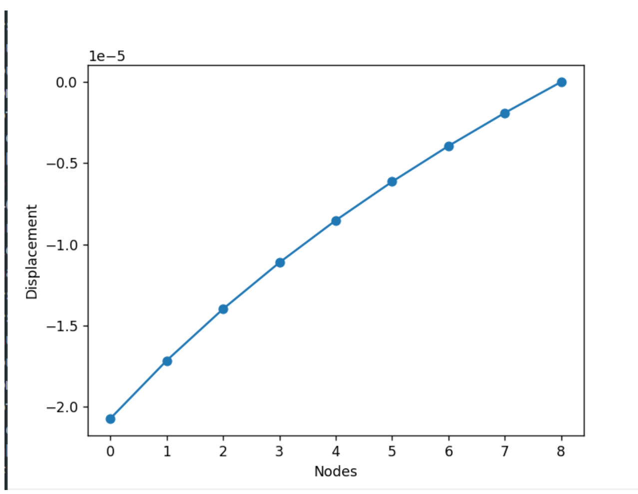


2.2 Subproblem2

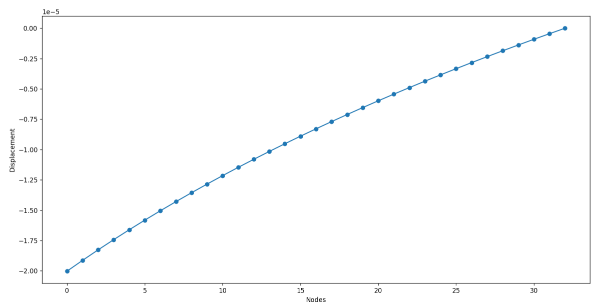
- PROB=2 N=2



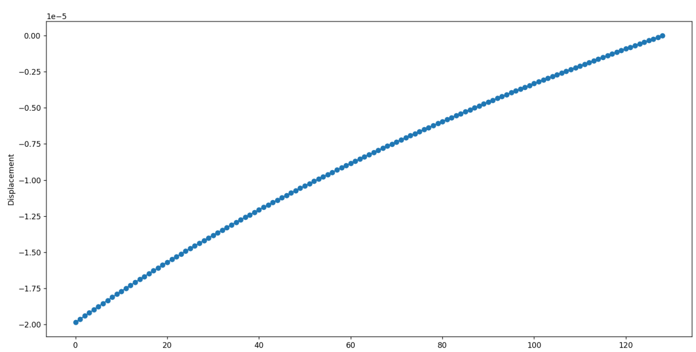
- PROB=2 N=8



- PROB=2 N=32



- PROB=2 N=128



3 Appendix

3.1 Snippets of execution time

3.1.1 Subproblem1

- PROB=1 N=2

```
[cs601user6@hip cs601pa2]$ make PROB=1 N=2
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
g++ -I /home/resiliente/cs601software/eigen-3.3.9 -o main main.cpp fem.cpp
./main 1 2
Number of elements: 2
The reaction force at fixed end= -5000N
Execution time: 3.5076e-05 seconds
```

- PROB=1 N=8

```
[cs601user6@hip cs601pa2]$ make PROB=1 N=8
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
./main 1 8
Number of elements: 8
The reaction force at fixed end= -5000N
Execution time: 9.1758e-05 seconds
```

- PROB=1 N=32

```
[cs601user6@hip cs601pa2]$ make PROB=1 N=32
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
./main 1 32
Number of elements: 32
The reaction force at fixed end= -5000N
Execution time: 0.000865376 seconds
```

- PROB=1 N=128

```
[cs601user6@hip cs601pa2]$ make PROB=1 N=128
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
./main 1 128
Number of elements: 128
The reaction force at fixed end= -5000N
Execution time: 0.0284917 seconds
```

3.1.2 Subproblem2

- PROB=2 N=2

```
[cs601user6@hip cs601pa2]$ make PROB=2 N=2
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
./main 2 2
Number of elements: 2
The reaction force at fixed end= -5000N
Execution time: 4.3863e-05 seconds
```

- PROB=2 N=8

```
[cs601user6@hip cs601pa2]$ make PROB=2 N=8
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
./main 2 8
Number of elements: 8
The reaction force at fixed end= -5000N
Execution time: 9.3097e-05 seconds
```

- PROB=2 N=32

```
[cs601user6@hip cs601pa2]$ make PROB=2 N=32
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
./main 2 32
Number of elements: 32
The reaction force at fixed end= -5000N
Execution time: 0.000867626 seconds
```

- PROB=2 N=128

```
[cs601user6@hip cs601pa2]$ make PROB=2 N=128
Team Members:
1. Kalidindi Dhrutika-210010021
2. Tejal Ladage-210010026
3. Ashwin Waghmare-210010060
./main 2 128
Number of elements: 128
The reaction force at fixed end= -5000N
Execution time: 0.0283756 seconds
```

3.2 Snippets of code

3.2.1 fem.h

```cpp
#ifndef FEM_H
#define FEM_H
#include <vector>
#include <Eigen/Dense>
using namespace Eigen;
class FiniteElementRod{
    double length;    // Length of the rod
    double material_modulus;    // Young's modulus of the material (Pa)
    double load;    // Applied load at x=0 (N)
    int num_elements_;       // Number of elements
    double element_length_;   // Length of each element
    std::vector<double> nodes_;    //// Nodal coordinates
    std::vector<std::pair<int, int>> elements_;   // Element connectivity

   public:
    // Constructor for initializing problem parameters
    FiniteElementRod(double length, double material_modulus, double load);
    // Function to generate the finite element mesh
    void generateMesh(int num_elements);
    // Function to create the local stiffness matrix and assemble it into Global Stiffness Matrix
    // Parameter 'type' specifies uniform or non-uniform area
    MatrixXd assembleStiffnessMatrix(int type);
    // Function to solve the finite element problem and compute displacements
    VectorXd solve(int type);
    // Function to print the displacements at nodal points
    void printDisplacement(VectorXd& displacement);
};
#endif
```

3.2.2 fem.cpp

```cpp
#include"../inc/fem.h"
#include<cstdio>
#include <vector>
#include<iostream>
#include <Eigen/Dense>
using namespace Eigen;

//constructor
FiniteElementRod::FiniteElementRod(double length, double material_modulus, double load)
    : length(length), material_modulus(material_modulus), load(load) {
}


void FiniteElementRod::generateMesh(int num_elements) {
    num_elements_ = num_elements;  // Set the number of elements in the rod
    element_length_ = length / num_elements;   // Calculate the length of each element
    nodes_.resize(num_elements_ + 1);   // Resize the vector to store nodal coordinates
    elements_.resize(num_elements_);   // Resize the vector to store element connectivity
    // Generate nodal coordinates for each node
    for (int i = 0; i <= num_elements_; ++i) {
        nodes_[i] = i * element_length_;
    }
    // Define element connectivity by specifying the pairs of nodes that form each element
    for (int i = 0; i < num_elements_; ++i) {
        elements_[i] = std::make_pair(i, i + 1);
    }
}
```

```cpp
//generating global stiffness matrix and global force vector
//eleminating last row and column from K because u(l)=0
//u=F*K^(-1)
VectorXd FiniteElementRod::solve(int type) {
    generateMesh(num_elements_);
    MatrixXd K = assembleStiffnessMatrix(type);
    VectorXd F(num_elements_);
    F.setZero();
    F(0) = load;
    // std::cout<<K<<"\n";
    MatrixXd K_mod=K.topLeftCorner(K.rows()-1,K.cols()-1);
    MatrixXd K_mod_inv=K_mod.inverse();
    VectorXd displacement=K_mod_inv*F;
    return displacement;
}

//printing the displacements
void FiniteElementRod::printDisplacement(VectorXd& displacement) {
    std::cout << "Displacement at nodal points:" << std::endl;
    for (int i = 0; i < num_elements_; ++i) {
    std::cout << "Node " << i << ": " << displacement[i]*-1 << " m" << std::endl;
    }
     std::cout<< "Node " <<num_elements_<<": "<<0<<" m"<<std::endl;
}
```

```cpp
//generating the global stiffness matrix
MatrixXd FiniteElementRod::assembleStiffnessMatrix(int type) {
    MatrixXd K(num_elements_ + 1, num_elements_ + 1);
    K.setZero();
    //uniform cross-section area
    if(type==1){
       double A =12.5e-4;
           for (int i = 0; i < num_elements_; ++i) {
             int x1 = elements_[i].first;
             int x2 = elements_[i].second;
             double L = nodes_[x2] - nodes_[x1];
         Matrix2d ke;//local stiffness matrix
             ke << 1, -1, -1, 1;
             ke *= A*material_modulus / L;
             K.block(x1, x1, 2, 2) += ke;  //combining local stiffness matrices
          }
    }
    //non-uniform cross-section area
    else{
        for (int i = 0; i < num_elements_; ++i) {
             int x1 = elements_[i].first;
             int x2 = elements_[i].second;
             double L = nodes_[x2] - nodes_[x1];
         double A0=12.5e-4;
         double A=A0*(1+nodes_[x1]/length);
             Matrix2d ke;
             ke << 1, -1, -1, 1;
             ke *= A*material_modulus / L;
             K.block(x1, x1, 2, 2) += ke;
           }
    }
        return K;
}
```

3.2.3 main.cpp

```cpp
#include<iostream>
#include<vector>
#include"../inc/fem.h"
#include <Eigen/Dense>
#include <iomanip>
#include <chrono>
using namespace Eigen;

int main(int argc, char* argv[]){
        // Define the problem parameters
    double length = 0.5; // Length of the rod
    double material_modulus = 70e9; // Young's modulus of the material (Pa)
    double load = 5000; // Constant load at x=0 (N)
    // Create a FiniteElementRod object
    FiniteElementRod rod(length, material_modulus, load);
    int type=std::stoi(argv[1]);  // to specify uniform or non-uniform cross-section area in the rod
    int num_elements=std::stoi(argv[2]);

    auto start = std::chrono::high_resolution_clock::now();

    rod.generateMesh(num_elements);
    VectorXd displacement = rod.solve(type); // generating diaplacements

    auto end = std::chrono::high_resolution_clock::now();
    std::chrono::duration<double> duration = end - start;

    std::cout << "Number of elements: " << num_elements << std::endl;
    rod.printDisplacement(displacement);
    double f;
    if(type==1)
    f=12.5e-4*material_modulus*displacement[num_elements-1]/(length/num_elements);
    else
    f =(12.5e-4*(1 + (static_cast<double>(num_elements - 1) / num_elements)) * material_modulus * displacement[num_elements - 1])/(length / num_elements);
    //f=12.5e-4(1+((num_elements-1)/num_elements))*material_modulus*displacement[num_elements-1]/(length/num_elements);

std::cout<<"The reaction force at fixed end= -"<<f<<"N"<<"\n";
    std::cout << "Execution time: " << duration.count() << " seconds" << std::endl;
}
```

3.2.4 makefile

```
PROB ?= 2
N ?= 2
EXECUTABLE=bin/main

all: team main run
main: src/main.cpp src/fem.cpp
	g++ -I /home/resiliente/cs601software/eigen-3.3.9 -o $(EXECUTABLE) src/main.cpp src/fem.cpp

run: main
	./$(EXECUTABLE) $(PROB) $(N)

team:
	@echo "Team Members:"
	@echo "1. Kalidindi Dhrutika-210010021"
	@echo "2. Tejal Ladage-210010026"
	@echo "3. Ashwin Waghmare-210010060"
clean:
	rm -f $(EXECUTABLE)
git:
	git status
	git add .
	git commit -m "auto git"
	git push
	git tag -a -f cs601pa2submission -m "Turnin PA1"
	git push -f --tags

.PHONY: all run team clean
```