A PROJECT REPORT

on

# PLANE STRAIN PROBLEM: ANALYSIS OF CANTILEVER RETAINING WALL USING CST ELEMENT

for the course of

Finite Element Methods in Civil Engineering

CE 723A

By

(Group C)

MALIK FAISAL NISSAR (16807379)

NUPUR SAXENA (19203269)

ADIDAM KAMESWARA RAMA KRISHANA (20103013)

RUPSAGAR CHATTERJEE (20103098)

Under the guidance of

PROF. SAMIT RAY CHAUDHURI

IIT Kanpur

Civil Engineering Department

Date of submission: 03/12/2020

CIVIL ENGINEERING DEPARTMENT

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

KANPUR-208016

# ACKNOWLEDGMENT

The success and final outcome of this project required a lot of guidance and assistance and we are extremely privileged to have got this all along the completion of our project.

We respect and thank Dr. Samit Ray Chaudhuri, Professor (Structural Engineering Department), for providing us the opportunity to do this project and giving us all support and guidance which made us complete the project duly. We are extremely thankful to him for providing such a nice support and guidance, although he had busy schedule managing the corporate affairs.

We perceive this opportunity as an important milestone in our career development. This has provided us a lot of information as well as enhanced our skills.

Sincerely,

Malik Faisal Nissar (16807379)

Nupur Saxena (19203269)

Adidam Kameswara Rama Krishana (20103013)

Rupsagar Chatterjee (20103098)

# ABSTRACT

In theory of elasticity, many problems can be modelled as two dimensional ones. Usually three dimensional problems are complicated and often involves cumbersome modelling and computation. Hence, the problems that can be modelled as two dimensional enable easier approach for obtaining solutions. Two dimensional problems can be plane stress and plane strain problems. If there are no strains out of the plane i.e. the strains perpendicular to the plane being considered, or any components therefore, are all zero, then this condition is called the plane strain condition

In this project report, a cantilever retaining wall which is a plane strain element has been analysed using finite element analysis method. The wall was discretized with CST elements and then using finite element equations, analysed for finding the deflections in MATLAB. Thereupon, the obtained results are compared with the analytical ones. Mesh sensitivity is also performed in this project report.

# TABLE OF CONTENTS

# 1. INTRODUCTION

In theory of elasticity, two dimensional problems can be plane stress and plane strain problems. Usually three dimensional problems are complicated and often involves cumbersome modelling and computation. Hence, the problems that can be modelled as two dimensional enable easier approach for obtaining solutions. Both the plane stress and the plane strain conditions can be modelled using 2D plane elements. 2D Planar Elements are defined by at least 3 nodes in a two-dimensional plane (x-y plane). These elements can be connected at common nodes and/or along common edges.

If there are no strains out of the plane i.e. the strains perpendicular to the plane being considered, or any components therefore, are all zero, then this condition is called the plane strain condition. The strains in z direction is as given in Equation 1.

$$\varepsilon_z = \gamma_{xz} = \gamma_{yz} = 0 \tag{1}$$

This happens in long bodies with constant cross sectional area and loads acting in x or y directions only with a constant value along the z-direction. Only a unit thickness of these structures is considered for analysis because all unit thicknesses except near the ends of the structure behave identically.

In this project report, a cantilever retaining wall which is a plane strain element has been analysed using finite element analysis method. The wall was discretized with CST elements and then through finite element equations, analysed for finding the deflections in MATLAB. Thereupon, the obtained results are compared with the analytical ones. Mesh sensitivity is also performed in this project report.

## 2. PROBLEM STATEMENT

The cantilever retaining wall as shown in Figure 1, is backfilled with tropical cohesion less lateritic earth fill, having a unit weight, ρ, of 18 kN/m3 and an internal angle of friction, ϕ, of 26 degrees. The allowable bearing pressure of the soil is 150 kN/m3, the coefficient of friction is 0.5, and the unit weight of reinforced concrete is 24 kN/m3. The pressure exerted is active earth pressure. The cantilever retaining wall is a plane strain element. It is discretized with CST elements and then analysed by finite element analysis method for finding the deflections in MATLAB. Also, then the obtained results are compared with the analytical ones. Grade of Concrete: 25 MPa and Poisson’s Ratio: 0.1

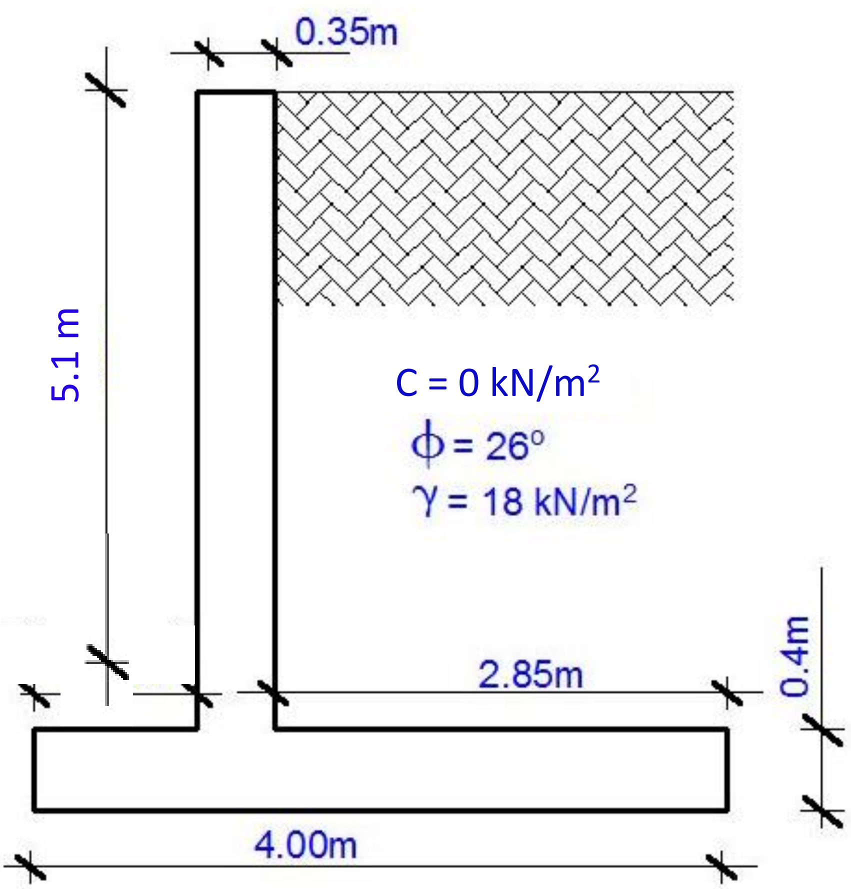



Fig. 1 Cantilever Retaining Wall

# 3. THEORY

## 3.1. 2-D STATE OF STRAIN IN PLANE STRAIN CONDITION

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix}$$

Where,

$$\{\varepsilon_x\} = \frac{\partial u}{\partial x}\,; \qquad \{\varepsilon_y\} = \frac{\partial v}{\partial y}\,; \qquad \{\gamma_{xy}\} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \tag{2}$$

u and v are the displacements in x and y directions respectively.

## 3.2. STRESS-STRAIN RELATIONSHIP FOR ISOTROPIC MATERIALS (IN PLANE STRAIN CONDITION)

The stress strain relationship is given by Equation 3.

$$\{\sigma\} = [D]\{\varepsilon\} \tag{3}$$

Where, {σ} is the stress, {Ɛ} is the strain and the matrix $D$ is called the stress-strain matrix or the constitutive matrix.

$$[D] = \frac{E}{(1+\mu)(1-2\mu)}\begin{bmatrix} 1-\mu & \mu & 0 \\ \mu & 1-\mu & 0 \\ 0 & 0 & \dfrac{1-2\mu}{2} \end{bmatrix} \tag{4}$$

Where, $E$ is the modulus of Elasticity and $v$ is the Poisson's ratio.

## 3.3. CONSTANT STRAIN TRIANGULAR ELEMENTS (CST)

This is a basic 2D finite element. It's a 3 nodded triangular element. Its derivation is the simplest among all the 2D elements. The strain remains constant throughout the element; hence the name CST. The formulation for the CST can most feasibly be achieved through the principle of minimum potential energy.

## 3.4. PROCEDURE

The basic procedure adopted for the analysis are as described below:

- <u>Discretization</u>

The domain of the problem is discretized into a set of triangular elements. Each element is defined by nodes i, j and m. To begin with, an initial coarse discretization of the geometry is done consisting of some desired number of elements. Subsequently the mesh is made finer by increasing the number of elements in the discretized geometry.

Each node in the discretized geometry has two degrees of freedom: displacements in the x and y directions. Let $u_i$ and $v_i$ represent the displacement components of node i in the x and y directions respectively.

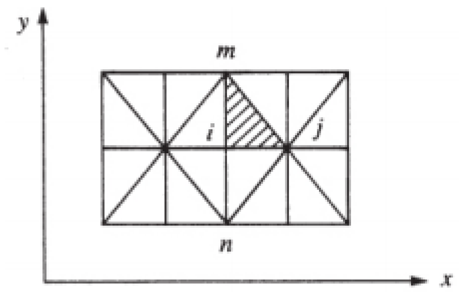
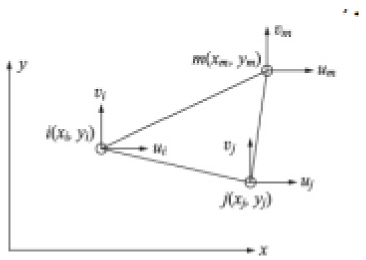



Fig. 2 Discretization

The nodal displacements for an element with nodes i, j, and m are:



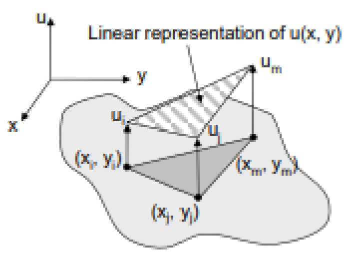
$$\{d\} = \begin{Bmatrix} d_i \\ d_j \\ d_m \end{Bmatrix}; \quad \{d\} = \begin{Bmatrix} u_i \\ v_i \\ u_j \\ v_j \\ u_m \\ v_m \end{Bmatrix} \tag{5}$$

Fig. 3 Nodal Displacements

Where the nodes are ordered counter clockwise around the element.

- <u>Element Stiffness Matrix and Force vector</u>

To generate element stiffness matrix and force vector, a linear displacement function for each triangular element is selected as defined below:




$$\{\varphi_i\} = \begin{Bmatrix} u\ (x, y) \\ v\ (x, y) \end{Bmatrix} = \begin{Bmatrix} a_1 + \ a_2 x + a_3 y \\ a_4 + \ a_5 x + a_6 y \end{Bmatrix} \tag{6}$$

Fig. 4 Linear Representation of u (x, y)

To obtain the values for a's nodal points are substituted into the above equations.

$$\begin{aligned} u_i &= \ a_1 + a_2 x_i + a_3 y_i \qquad v_i = \ a_4 + a_5 x_i + a_6 y_i \\ u_j &= \ a_1 + a_2 x_j + a_3 y_j \qquad v_i = \ a_4 + a_5 x_j + a_6 y_j \\ u_m &= \ a_1 + a_2 x_m + a_3 y_m \qquad v_i = \ a_4 + a_5 x_m + a_6 y_m \end{aligned} \tag{7}$$

- *Assembling the Stiffness Matrix and Force vector, Comparison with the Analytical results, Mesh sensitivity Analysis*

After obtaining the elemental stiffness matrix, it was assembled. And thereupon, tip deflection was calculated by finite element method and also by analytical calculation, followed by sensitivity analysis.

# 4. ANALYSIS AND RESULTS

## 4.1. DISCRETIZATION

Initially the retaining wall is discretized by 6 vertical and 11 horizontal lines in such a way so as to form a set of 50 triangular elements. After analysing the problem with this coarse mesh, the mesh size is progressively refined to add more nodes and elements into the discretized geometry. The geometry is chosen to be discretized by the following sets of vertical and horizontal lines: [6,11,16,21,26,31,36,41,46,51,56] and [11,21,31,41,51,61,71,81,91,101,111] subjected to the condition of relative approximate percentage error for tip deflection falling below the allowable value. For higher level of solution accuracy greater number of such divisions are required. Some of the discretized geometries are shown in figures below.

The Y axis is taken in the upward vertical direction while the X axis is taken along the horizontal direction towards right.

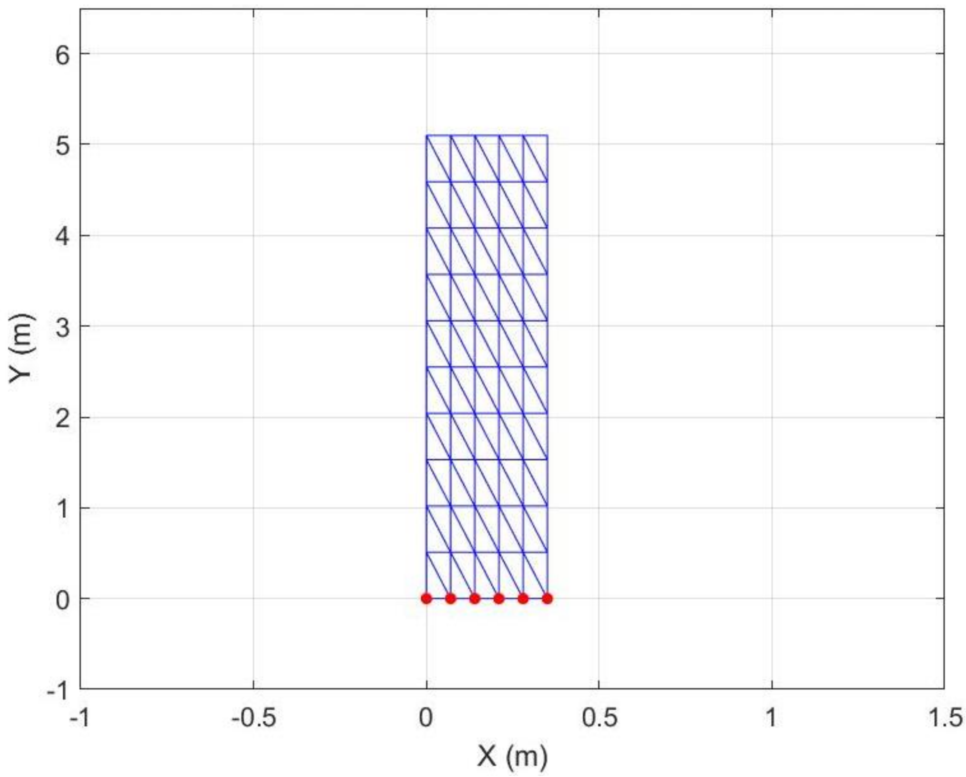



Fig. 5 Discretized geometry with 6 vertical lines and 11 horizontal lines

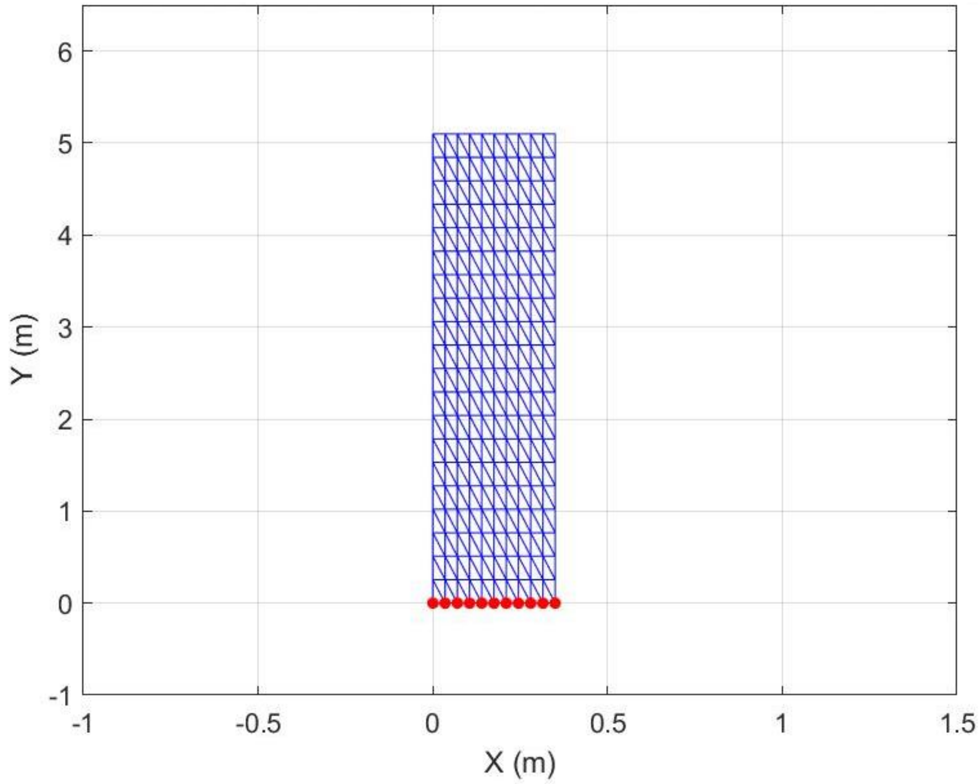



Fig. 6 Discretized geometry with 11 vertical lines and 21 horizontal lines

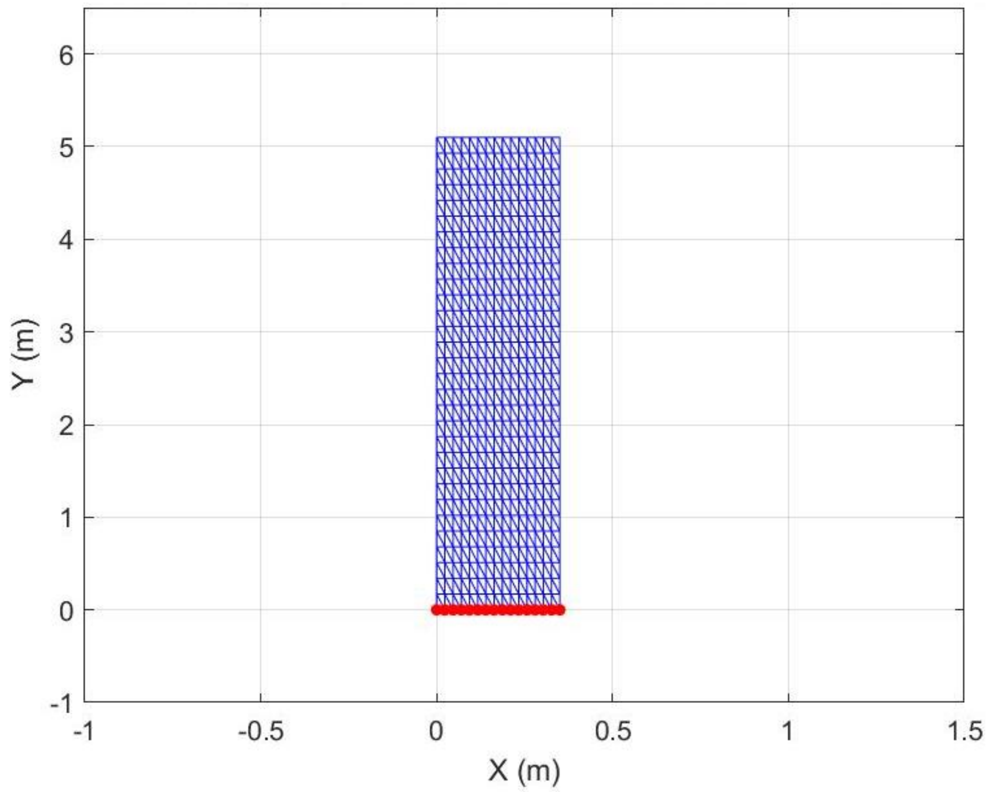



Fig. 7 Discretized geometry with 16 vertical lines and 21 horizontal lines

## 4.2. BOUNDARY CONDITIONS

In the discretized geometry, the boundary conditions for the problem are imposed at the respective nodes. The retaining wall at the bottom end having a fixed end support with the slab both horizontal and vertical degrees of freedom of the corresponding element nodes are restrained.

## 4.3. ELEMENT LEVEL CALCULATION

In the discretized geometry, the element connectivity matrix is evaluated. The elements are numbered sequentially along a horizontal starting from the left most bottom triangular element. The degrees of freedom associated with a node i are taken as (2*i-1) in the horizontal direction and 2*i in the vertical direction. The area and the strain displacement matrix for each element are evaluated from the resulting nodal coordinates of the discretization. The element stiffness matrices are subsequently calculated which is of size 6x6. The elements are assumed to be to uniform and unit thickness.

## 4.4. EVALUATION OF FORCE VECTOR

The traction forces acting on the right side of the retaining wall due to the active earth pressure of the cohesionless soil is of trapezoidal distribution across the element side on which it is acting. The exerted traction is assumed to be composed of a triangular force distribution superimposed on a rectangular one. The consistent traction vector on element nodes is evaluated by equally distributing the forces of the rectangular load distribution to both nodes and taking 2:1 proportion of the triangular load across the element side. The body force is only due to the gravity force acting on the element in the vertically downward direction. These forces are taken as $1/3^{rd}$ of the element weight under gravity in the Y axis degrees of freedom. The resultant consistent force vector is the sum of the body force vector and the traction force vector.

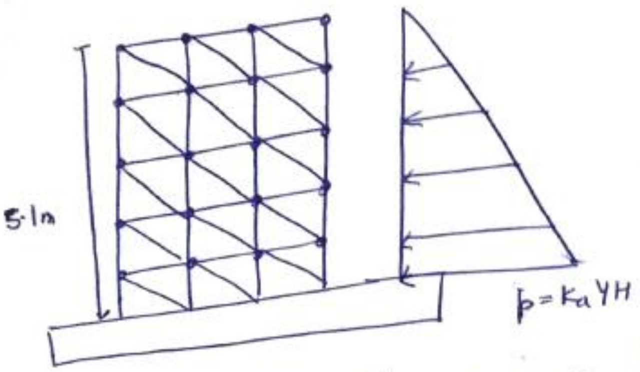
## 4.5. ASSEMBLY

The element stiffness matrices are assembled into the global system stiffness matrix by summing the stiffness contribution at a node from all the surrounding elements of the node. The element stiffness coefficients are mapped from the local nodal degrees of freedom to the global degrees of freedom through the mapping of the associated element nodes with their associated degrees of freedom.

## 4.6. SOLUTION

The boundary conditions are imposed on the global force vector and the global system stiffness matrix to give simultaneous equations in the unknown degrees of freedom. These are solved using standard subroutine available in MATLAB to find the unknown nodal displacement vector.

## 4.7. COMPARISION OF DEFLECTION WITH ANALYTICAL RESULTS

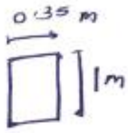



$$K_a = \frac{1 - \sin 26^\circ}{1 + \sin 26^\circ} = 0.3905$$

$$p = 0.3905 \times 10^3 \times 18 \times 5.1$$
$$= 35847.9 \ \text{N/m}^2 = w$$

$$E = 5000\sqrt{f_{ck}}$$
$$= 2.5 \times 10^{10} \ \text{N/m}^2$$

$$I = \frac{bd^3}{12} = \frac{1 \times 0.35^3}{12}$$




Theoretical cantilever deflection

$$= \frac{wl^4}{30EI} = \frac{35847.9 \times 5.1^4}{30 \times 2.5 \times 10^{10} \times 0.35^3/12}$$

$$= 9.05 \times 10^{-3} \ m$$

## 4.8. MESH SENSITIVITY

A coarser mesh size results in a stiff system thereby giving lower values of tip deflection. As the geometry is discretized into finer mesh sizes the result is found to converge. The iterations with refined mesh sizes are performed till the relative approximate percentage error does not fall below the acceptable limit as defined by the problem statement. The mesh sensitivity for the problem is shown below. It is observed from the figure, that with increase in the number of CST elements, that is with a finer mesh size, the absolutes value of tip defection rapidly increases and slowly converges to a constant value.

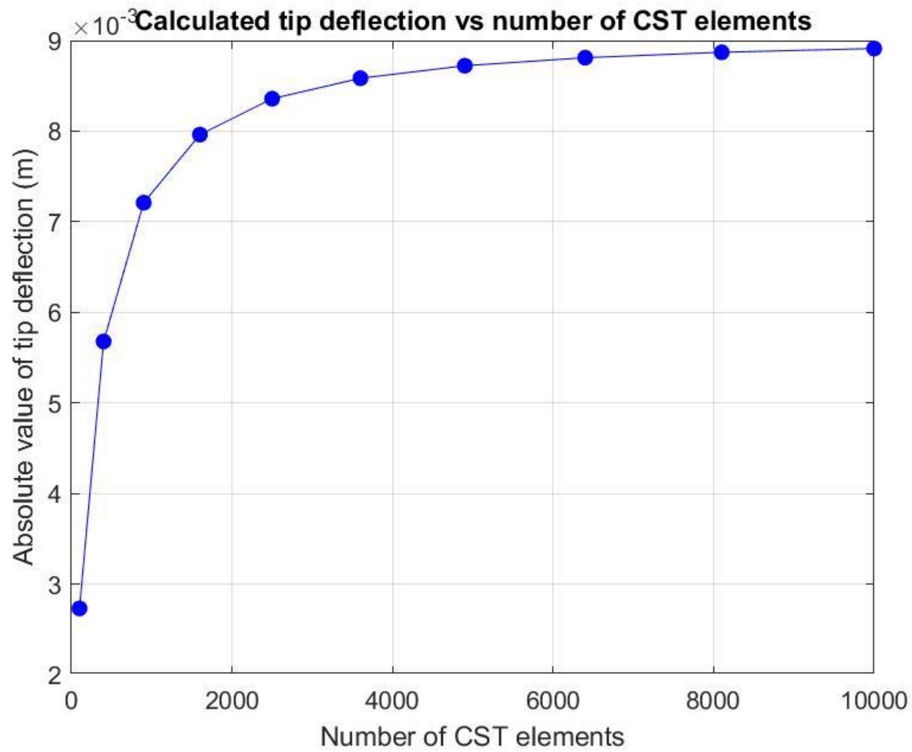



Fig. 8 Calculated tip deflection vs number of CST elements

Also, with finer mesh sizes the relative approximate error diminishes and ultimately falls below 0.5% with 10000 elements.

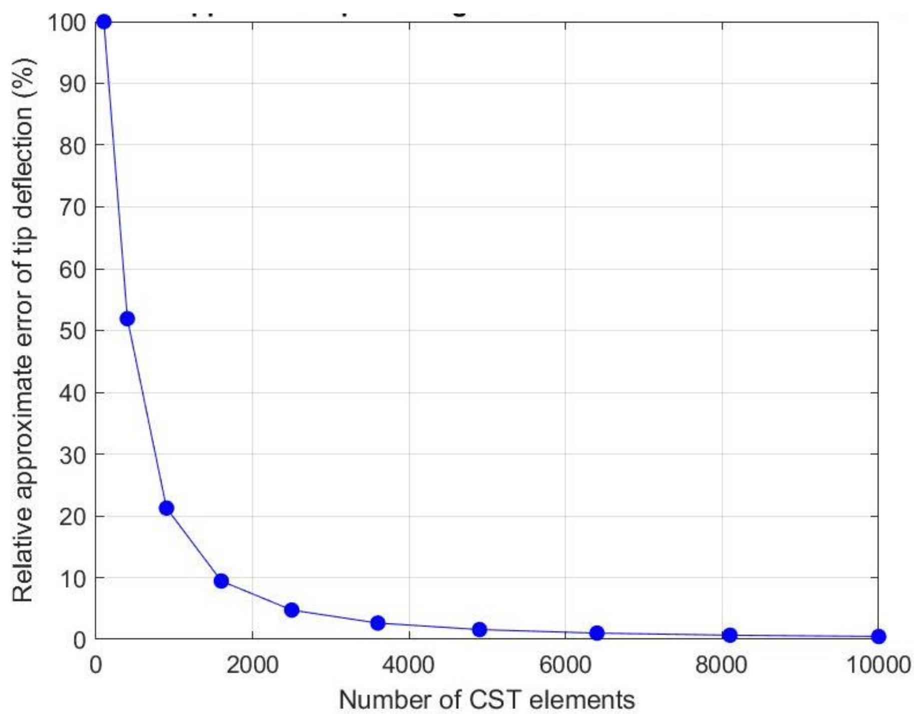



Fig. 9 Relative approximate percentage error vs number of CST elements

## 5. CONCLUSION

- A theoretical tip deflection of the given problem can be calculated by approximating the retaining wall subjected to the active earth pressure of the cohesion less soil to Euler Bernoulli cantilever beam with a uniformly varying load with maximum intensity at the fixed end. The theoretical deflection in such case is given by:

$$\delta = \frac{wL^4}{30EI}$$

Where w = maximum load intensity at fixed end

L = length of the cantilever

E = Young's modulus of beam material

I = Second area moment of inertia

- By substituting the above data from the problem, the tip deflection is obtained as $9.0502*10^{-3}$. In this theoretical approximation, the forces due to body weight are not taken into consideration.
- The obtained value of tip deflection from computer analysis with CST elements is $8.9121*10^{-3}$ with 0.5% approximate relative percentage error as defined in the problem.
- It can be seen that the obtained finite element result is in close agreement with the idealized theoretical result. The result can be made further accurate with a lower error limit at the cost of computational effort.
- However, beyond a certain limit refining the mesh size may introduce undesirable truncation error.

## 6. MATLAB CODES

The plane strain problem is analysed with the following MATLAB codes.

ProjectMainScript.m

```
% MAXIMUM ALLOWABLE RELATIVE APPROXIMATE PERCENTAGE ERROR
EaMax = 0.5; % in percentage (%)

% DISCRETIZATION (USER INPUT - ALL VALUES MUST BE GREATER THAN ONE)
VerLineWall = [6 11 16 21 26 31 36 41 46 51 56]; % make divisions along
width of slab
HorLineWall = [11 21 31 41 51 61 71 81 91 101 111]; % make divisions
along height of slab

NumOfIter = numel(VerLineWall); % maximum number of iterations intended
MeshSensitivity = zeros(NumOfIter,3); % array containing mesh
sensitivity details

for i = 1:NumOfIter
    % PERFORM PLANE STRAIN ANALYSIS WITH CST ELEMENTS
    Output = CSTAnalysis(VerLineWall(i),HorLineWall(i));
```

```
    % STORING CURRENT ITERATION MESH SIZE, TIP DEFLECTION, RELATIVE
APPROXIMATE ERROR
    if i==1
        Ea = 100; % in percentage (%)
    else
        Ea = abs(1-MeshSensitivity(i-1,2)/abs(Output.Disp(end-1)))*100;
% in percentage (%)
    end
    MeshSensitivity(i,:) = [Output.NumOfElements,abs(Output.Disp(end-
1)),Ea];

    % PLOT MESH AND BOUNDARY CONDITIONS (COMMENT THIS TO SPEED UP CODE
EXECUTION)
%     SavePlotMeshBC(VerLineWall(i),HorLineWall(i),Output,'on')

    % CHECK WHETHER RELATIVE APPROXIMATE ERROR IS LESS THAN MAXIMUM
ALLOWABLE RELATIVE APPROXIMATE ERROR
    if Ea<=EaMax
        NumOfIter = i;
        break;
    end
end

% PLOT MESH SENSITIVITY
SavePlotMeshSensitivity(MeshSensitivity(1:NumOfIter,:),'on')
```

## CSTAnalysis.m

```
function Output = CSTAnalysis(VerLineWall,HorLineWall)

% LOAD PROBLEM DATA
ProblemData

% DISCRETIZED GEOMETRY DATA
NumOfNodes = VerLineWall*HorLineWall; % total number of nodes of wall
NumOfElements = 2*(VerLineWall-1)*(HorLineWall-1); % total number of
elements of wall

% NODAL COORDINATES OF WALL ELEMENTS
x = 0:WallWidth/(VerLineWall-1):WallWidth;
y = 0:WallHeight/(HorLineWall-1):WallHeight;
x = x'.*ones(VerLineWall,HorLineWall);
y = y.*ones(VerLineWall,HorLineWall);
NodeXY = [x(:),y(:)];

% BOUNDARY CONDITIONS
BC = false(2*NumOfNodes,1);
BC(1:2*VerLineWall) = true;
BCDOFMap = logical(~BC'.*~BC);
```

```
% INITIALIZATION
ElementNodes = zeros(NumOfElements,3); % array containing ID of element
nodes
ElementDOF = zeros(NumOfElements,6); % array containing associated DOF
of each member
A2 = zeros(NumOfElements,1); % vector containing twice of element areas
B = cell(NumOfElements,1); % cell containing element strain
displacement matrices
Ke = cell(NumOfElements,1); % cell containing element stiffness
matrices
Kg = zeros(2*NumOfNodes); % global system stiffnes matrix
Fb = zeros(2*NumOfNodes,1); % nodal body force vector
Ft = zeros(2*NumOfNodes,1); % nodal traction force vector
Disp = zeros(2*NumOfNodes,1); % nodal displacment vector
K = zeros(sum(~BC)); % system stiffness matrix in the unrestrained DOFs

% ELEMENT CONNECTIVITY MATRIX
index = 0;
for i = 1:NumOfElements
    if mod(i,2)~=0 % nodes for odd numbered element
        ElementNodes(i,:) = (i+1)/2+index+[0 1 VerLineWall];
    else % nodes for even numbered element
        ElementNodes(i,:) = i/2+index+[1 VerLineWall VerLineWall+1];
    end
    if mod(i,2*(VerLineWall-1))==0
        index = index+1;
    end
end

% ELEMENT LEVEL CALCULATION
for i = 1:NumOfElements
    % ELEMENT DOFs
    for k = 1:numel(ElementNodes(i,:))
        ElementDOF(i,2*k-[1 0]) = 2*ElementNodes(i,k)-[1 0]; % DOFs
associated with each element
    end

    % TWICE OF ELEMENT AREA (POSITIVE VALUE IF NODES ARE TAKEN IN
CLOCKWISE SENSE)
    A2(i) = det([ones(3,1),NodeXY(ElementNodes(i,:),:)]); % twice of
element area

    % ELEMENT STRAIN DISPLACEMENT MATRIX
    B{i} = StrainDisp(NodeXY(ElementNodes(i,:),:),A2(i));

    % ELEMENT STIFFNESS MATRIX
    Ke{i} = B{i}'*D*B{i}*Thickness*abs(A2(i))/2;

    % ASSEMBLING ELEMENT STIFFNESS MATRIX INTO GLOBAL STIFFNESS MATRIX
    for j = 1:numel(ElementDOF(i,:))
        for k = 1:numel(ElementDOF(i,:))
            Kg(ElementDOF(i,k),ElementDOF(i,j)) =
Kg(ElementDOF(i,k),ElementDOF(i,j))+Ke{i}(k,j);
        end
    end

    % CONTRIBUTION OF ELEMENT BODY FORCES AT EACH NODE
```

```
    Fb(ElementDOF(i,:)) =
Fb(ElementDOF(i,:))+Thickness*abs(A2(i))/2/3*GammaConcrete*[0 -1 0 -1 0
-1]';
end

% CALCULATION FOR TRACTION FROM RIGHT SIDE OF WALL
for i = 2*(VerLineWall-1):2*(VerLineWall-1):NumOfElements
    Limits = NodeXY(ElementNodes(i,[1 3]),2);
    TractionValue =
arrayfun(TractionWallRight,Limits)*Thickness*(Limits(2)-Limits(1));
    RectLoad = TractionValue(2);
    TriLoad = (TractionValue(1)-TractionValue(2));
    Ft(ElementDOF(i,[1 5])) = Ft(ElementDOF(i,[1 5]))+RectLoad*[1/2
1/2]'+TriLoad*[2/3 1/3]';
end

% LOAD VECTOR
F = Fb+Ft;

% IMPOSING BOUNDARY CONDITIONS
K(:) = Kg(BCDOFMap);
Load = F(~BC);

% SOLVING FOR UNRESTRAINED DOFs
Disp(~BC) = K\Load;

% OUTPUT STRUCTURE
Output.NumOfNodes = NumOfNodes;
Output.NumOfElements = NumOfElements;
Output.NodeXY = NodeXY;
Output.BC = BC;
Output.ElementNodes = ElementNodes;
Output.Ke = Ke;
Output.Kg = Kg;
Output.F = F;
Output.Disp = Disp;

end
```

## ProblemData.m

```
% GEOMETRY DETAILS
WallHeight = 5.1; % in m
WallWidth = 0.35; % in m
Thickness = 1; % unit thickness

% CONCRETE DETAILS
E = 5000*sqrt(25)*1e6; % in N/m^3
nu = 0.1; % Poisson's ratio
GammaConcrete = 24e3; % N/m^3

% SOIL (COHESIONLESS C = 0)
Phi = 26; % in degree
GammaSoil = 18e3; % N/m^3

% EARTH PRESSURE CALCULATION
Ka = (1-sin(Phi/180*pi))/(1+sin(Phi/180*pi)); % coefficient of active
earth pressure
```

```
TractionWallRight = @(y)-Ka*GammaSoil*(WallHeight-y); % traction
variation on right face

% CONSTITUTIVE MATRIX
D = E/(1+nu)/(1-2*nu)*[1-nu nu 0;nu 1-nu 0;0 0 (1-2*nu)/2]; %
constitutive relation for plane strain problem
```

## StrainDisp.m

```
function B = StrainDisp(XY,A2)
    B = zeros(3,6);
    index = [1 2 3 1 2];
    for j = 1:3
        B(1,2*j-1) = XY(index(j+1),2)-XY(index(j+2),2);
        B(2,2*j) = XY(index(j+2),1)-XY(index(j+1),1);

        B(3,2*j-1) = XY(index(j+2),1)-XY(index(j+1),1);
        B(3,2*j) = XY(index(j+1),2)-XY(index(j+2),2);
    end
    B = B/A2;
end
```

## SavePlotMeshBC.m

```
function SavePlotMeshBC(VerLineWall,HorLineWall,Output,DisplayPlot)

MeshFig = figure('visible','off','WindowState','maximized');
Axis1 = gca;
hold on

% PLOT MESH
for j = 1:Output.NumOfElements
    plot(Axis1,Output.NodeXY(Output.ElementNodes(j,[1 2 3
1]),1),Output.NodeXY(Output.ElementNodes(j,[1 2 3 1]),2),'-b')
end

% PLOT BOUNDARY CONDITION
for j = 1:Output.NumOfNodes
    if Output.BC(2*j-1)&&Output.BC(2*j)

plot(Axis1,Output.NodeXY(j,1),Output.NodeXY(j,2),'.r','MarkerSize',15)
    end
end

title(['Discretized geometry with ',num2str(VerLineWall),' vertical
lines and ',num2str(HorLineWall),' horizontal lines'])
xlabel('X (m)')
ylabel('Y (m)')
Axis1.XLim = [-1 1.5];
Axis1.YLim = [-1 6.5];
grid on
box on
saveas(MeshFig,['GeneratedMesh',num2str(Output.NumOfElements),'.jpg'])
MeshFig.Visible = DisplayPlot;

end
```

## SavePlotMeshSensitivity.m

```
function SavePlotMeshSensitivity(MeshSensitivity,DisplayPlot)

DeflectionFig = figure('visible','off','WindowState','maximized');
plot(MeshSensitivity(:,1),MeshSensitivity(:,2),'ob-
','MarkerFaceColor','b')
title('Calculated tip deflection vs number of CST elements')
xlabel('Number of CST elements')
ylabel('Absolute value of tip deflection (m)')
grid on
box on
saveas(DeflectionFig,['DeflectionConvergenceFig
',num2str(MeshSensitivity(:,1)'),'.jpg'])

ErrorFig = figure('visible','off','WindowState','maximized');
plot(MeshSensitivity(:,1),MeshSensitivity(:,3),'ob-
','MarkerFaceColor','b')
title('Relative approximate percentage error vs number of CST
elements')
xlabel('Number of CST elements')
ylabel('Relative approximate error of tip deflection (%)')
grid on
box on
saveas(ErrorFig,['ErrorFig ',num2str(MeshSensitivity(:,1)'),'.jpg'])


DeflectionFig.Visible = DisplayPlot;
ErrorFig.Visible = DisplayPlot;

end
```