23PHY114
Class Notes

Adithya Nair

Contents

Chapter 1

Solids

1.1 Moment Of Area

The moment of inertia is used to help find the ”resistance” to the force, given a specific axis or direction.

1.2 Resisting Force And Moments From Supports

There are three main kinds of supports -

1. Pin/Hinge - Fixes linear motion but leaves rotation free.
2. Roller - Fixes rotation but leaves motion free.
3. Clamp - Fixes both linear and rotational motion.

Take this case, with a bunch of forces being applied to the given load. There are three main things we need to find for this figure.

1. The resultant force acting on this bar fixed to a hinge.
2. The support reaction force and moment.
3. The moment on the object (maximum)

The way to approach the problem is as always,

1. Free Body Diagram first.
2. Assuming $\Sigma f = 0$, because the object has no acceleration currently, because it is fixed.
3. Assuming $\Sigma M = 0$

4. The point at which the resultant force is applied is found by,

$$\frac{\int |r| dm}{\int dm}$$

1.3 Derivation Of The Uniaxial Formula

1.4 For Uniaxial Deformation

Main Subroutines For Uniaxial Deformation

1.4.1 Finding The Local Stiffness Matrix

```
function stiffnessLocal = localStiffnessGenerator(E,A,l,theta);
    stiffnessConstant = E*A/l;
    R = [cos(theta) -sin(theta); sin(theta) cos(theta)];
    stiffnessMatrix = [stiffnessConstant 0 -stiffnessConstant 0;
    zeros(1,4);-stiffnessConstant 0 stiffnessConstant 0; zeros(1,4)
    ];
    R4 = [R zeros(2,2); zeros(2,2) R];
    stiffnessLocal = R4*stiffnessMatrix*R4';
end
```

octave

1.4.2 Converting The Local Stiffness To A Global Stiffness Matrix

```
   function stiffnessLocalGlobal = local2Global(stiffnessLocal,
    node1,node2,nodeCount)
    stiffnessLocalGlobal = zeros(2*nodeCount,2*nodeCount);
    i = 2*node1 - 1;
    j = 2*node2 - 1;
    stiffnessLocalGlobal(i:(i+1),i:(i+1)) = stiffnessLocal(1:2,1:2)
    ;
    stiffnessLocalGlobal(i:(i+1),j:(j+1)) = stiffnessLocal(1:2,3:4)
    ;
    stiffnessLocalGlobal(j:(j+1),i:(i+1)) = stiffnessLocal(3:4,1:2)
    ;
    stiffnessLocalGlobal(j:(j+1),j:(j+1)) = stiffnessLocal(3:4,3:4)
    ;
end
```

octave

1.4.3 Main Loop Through Evaluating The Global Stiffness Matrix

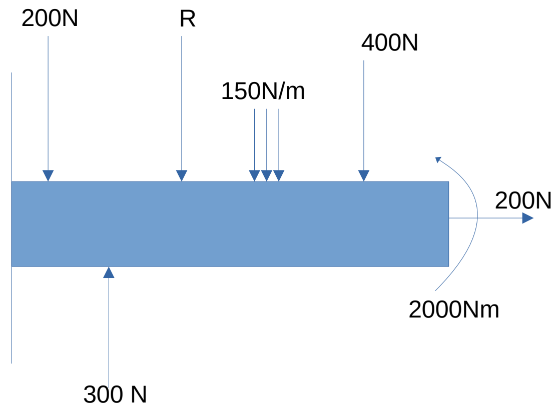
```
   A = 0; theta = 0; l = 0; stiffnessLocal = zeros(4,4);
    nodeAxialForces = zeros(nodeCount,1);
for element = 1:5 % For first three bars
    A = areaVector(element);
    theta = angleVector(element);
    l = lengthVector(element);
    stiffnessLocal = localStiffnessGenerator(E,A,l,theta);
    nodeCounter = element*2 - 1;
    stiffnessLocalGlobal = local2Global(stiffnessLocal,nodeVector(
    nodeCounter),nodeVector(nodeCounter+1),nodeCount);
    stiffnessLocal
    stiffnessLocalGlobal
    stiffnessGlobal += stiffnessLocalGlobal;
end
```

octave

1.4.4 Applying Boundary Conditions

```
selectedVector = [3 5:end]
forceEval = forceVector(selectedVector);
displacementEval = displacementVector(selectedVector);
stiffnessEval = stiffnessGlobal(selectedVector);
displacementEval = stiffnessEval\forceEval;
```

octave


200N
R
400N
150N/m
200N
2000Nm
300 N