Open System for Earthquake Engineering Simulation
Pacific Earthquake Engineering Research Center

OpenSees User & Developer Workshop

OpenSees section and beam-column element Commands:

August 21, 2003

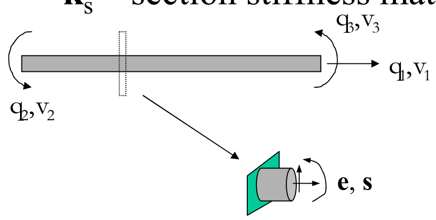
Michael H. Scott

Overview

- Beam-column sections
- Uniaxial materials for fiber sections
- Fiber section patch and layer commands
- Beam-column elements
- Geometric transformations
- Example commands

What is a section?

- Defines the stress resultant force-deformation response at a cross section of a beam-column element
- Matrix-vector notation
  - **e** – section deformation vector
  - **s** – section force vector
  - $\mathbf{k}_s$ – section stiffness matrix



Types of sections

- Elastic – defined by material and geometric constants
- Fiber – section is discretized into smaller regions for which the material response is integrated to give stress resultant behavior
  - Examples: steel, reinforced concrete
- Resultant – general nonlinear description of force-deformation response, e.g. moment-curvature

Elastic section command

- Useful for patch tests of the nonlinear beam-column elements (nonlinearBeamColumn, beamWithHinges, dispBeamColumn, etc.)
- Allows nonlinear beam-column elements to be used for elastic analysis

```
section Elastic $tag $E $A $I
```

tag – integer identifier unique among all sections

E – elastic modulus

A – cross section area

I – second moment of cross section area

Fiber section command

- Created from previously defined uniaxial material models and the section discretization commands: `patch`, `layer`, and `fiber`.

```
section Fiber $tag {
   patch …
   layer …
   fiber …
}
```

Uniaxial Material Models

- Before getting into the fiber section command, brief review of uniaxial materials
- Uniaxial materials define the stress-strain response at a point on a beam-column section
- For beam-column sections mainly steel and concrete models in OpenSees
- If they don't suit your needs, you can code up your own … that's why we made it open source!
- Other uniaxial material models available for other uses in OpenSees, e.g. SDOF systems

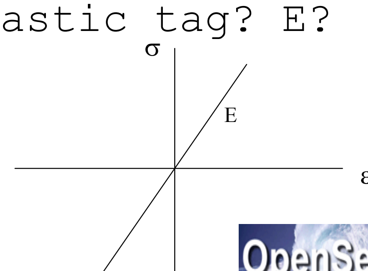
Elastic Material

- One-dimensional elasticity
- Useful for tracking down problems in your model, i.e., if your model doesn't run with all elastic fibers there's something seriously wrong, such as rigid body motion

```
uniaxialMaterial Elastic tag? E?
```

tag – integer identifier unique among all uniaxial materials

E – elastic modulus

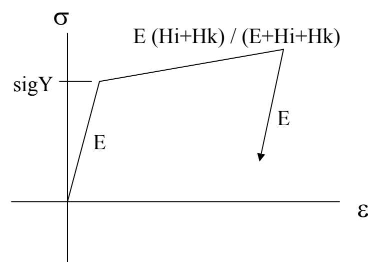


Bilinear Hardening Material

- One-dimensional return map algorithm
- Combined kinematic and isotropic hardening capable of representing steel behavior

```
uniaxialMaterial Hardening tag? E?
  sigmaY? Hiso? Hkin?
```

tag – integer identifier unique among all uniaxial materials
E – elastic modulus
sigmaY – yield stress
Hiso – isotropic hardening modulus
Hkin – kinematic hardening modulus



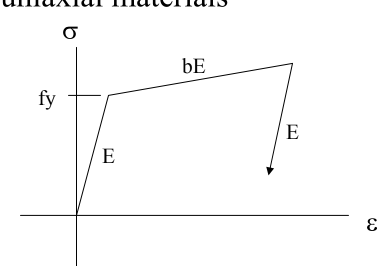
Note: reduces to EPP for Hiso = Hkin = 0

Steel01 Material

- Bilinear model with kinematic hardening

```
uniaxialMaterial Steel01 tag? fy?
  E? b?
```

tag – integer identifier unique among all uniaxial materials
fy – yield stress
E – elastic modulus
b – hardening ratio



- Other steel material models
  - Steel02 – Menegotto-Pinto

  See documentation

Concrete01 Material

- Simple uniaxial concrete model with Kent-Park backbone, linear unloading/reloading, and no tension strength

```
uniaxialMaterial Concrete01 tag? fc? epsc?
  fu? epsu?
```

tag – integer identifier unique among all uniaxial materials

fc – compressive strength

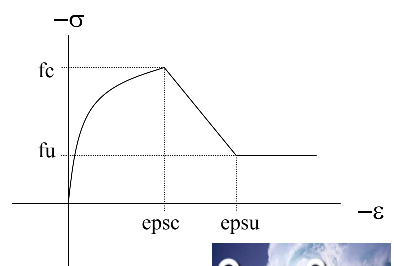
epsc – strain at fc

fu – ultimate strength

epsu – strain at fu

- Other concrete material models
  - Concrete02 (tension strength)
  - Concrete03 (nonlinear unloading)

  See documentation

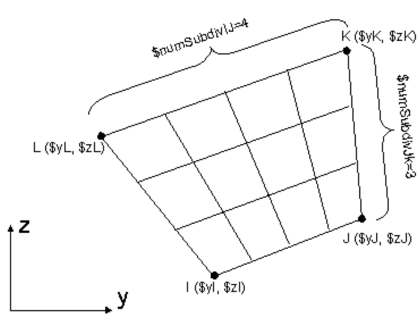


The patch command

- Quadrilateral and circular patches available

```
patch quadr $matTag $nfIJ $nfJK
  $yI $zI $yJ $zJ $yK $zK $yL $zL
```

matTag – integer identifier for previously defined uniaxial material



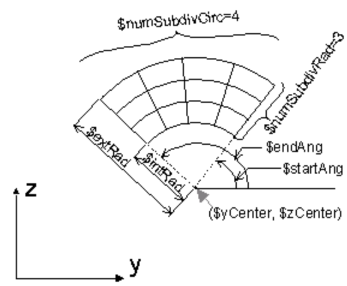
Quadrilateral Patch

Points I, J, K, L must be defined in counter-clockwise order around section!

Note: bending is about the section z-axis in 2D

The patch command (cont.)

```
patch circ $matTag $nfCirc $nfRad $y $z
  $intRad $extRad <$startAng $endAng>
```



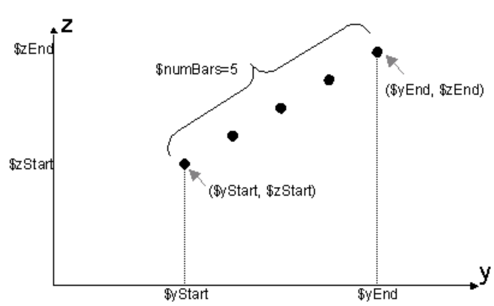
Circularl Patch

If `startAng` and `endAng` are not specified, a full circle is generated

Note: bending is about the section z-axis in 2D

The layer command

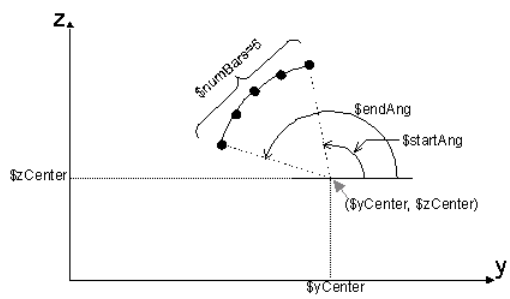
```
layer straight $matTag $numBars
  $barArea $yStart $zStart $yEnd $zEnd
```



Straight reinforcing layer

The layer command (cont.)

```
layer circ $matTag $numBars $barArea $y
  $z $radius <$startAng $endAng>
```



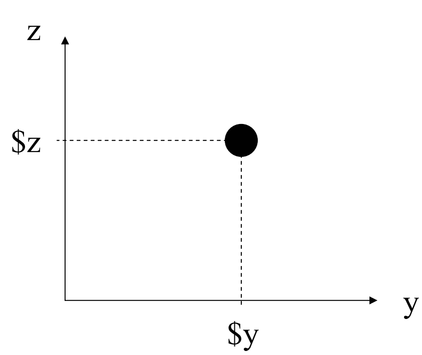
Circular reinforcing layer

If `startAng` and `endAng` are not specified, a full circle is generated

The fiber command

- Define a single fiber by location and area

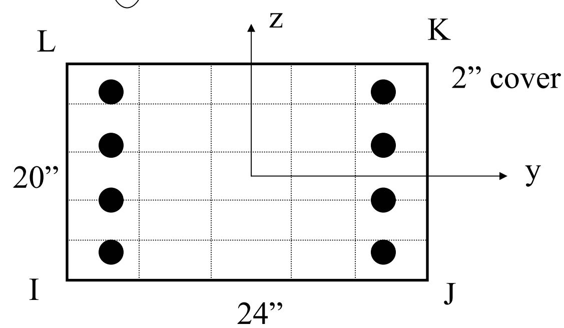
```
fiber $y $z $area $matTag
```



Example Fiber Section Creation

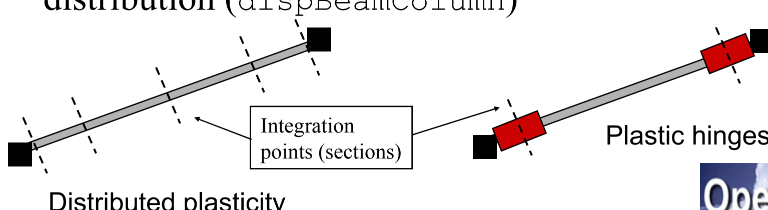
```
uniaxialMaterial Concrete01 1 -5.0 -0.002 -2.0 -0.01
uniaxialMaterial Steel01 2 60 30000 0.02

section Fiber 1 {
  patch quadr 1 5 5  -12 -10  12 -10  12 10  -12 10
  layer straight 2 4 1.0 -10 -8  -10 8
  layer straight 2 4 1.0  10 -8   10 8
}
```



Nonlinear beam-column elements

- Force based elements
  - Distributed plasticity (`nonlinearBeamColumn`)
  - Plastic hinge regions with elastic interior (`beamWithHinges`)
- Displacement based element
  - Distributed plasticity with linear curvature distribution (`dispBeamColumn`)



The nonlinearBeamColumn command

```
element nonlinearBeamColumn $tag $ndI
  $ndJ $numSec $secTag $transfTag
```

tag – integer identifier unique among all elements

ndI – integer identifier for node I

ndJ – integer identifier for node J

numSec – number of sections (Gauss-Lobatto integration)

secTag – integer identifier for sections (assumed prismatic)

transfTag – integer identifier for geometric transformation

The dispBeamColumn command

- Exactly the same inputs as nonlinearBeamColumn, just change the name!

```
element dispBeamColumn $tag $ndI $ndJ
  $numSec $secTag $transfTag
```

tag – integer identifier unique among all elements

ndI – integer identifier for node I

ndJ – integer identifier for node J

numSec – number of sections (Gauss-Legendre integration)

secTag – integer identifier for sections (assumed prismatic)

transfTag – integer identifier for geometric transformation

The beamWithHinges command

```
element beamWithHinges $tag $ndI $ndJ
  $secTagI $lpI $secTagJ $lpJ $E $A $I
  $transfTag
```

tag – integer identifier unique among all elements

ndI – integer identifier for node I

ndJ – integer identifier for node J

secTagI(J) – integer identifier for section at node I (J)

lpI(J) – hinge length for section at node I (J)

E, A, I – elastic properties of element interior

transfTag – integer identifier for geometric transformation

The geomTransf command

- Defines transformation of element forces and stiffness from basic to global system
- Three types available: `Linear, PDelta, Corotational`

```
geomTransf $type $tag <$vecXZ>
```

type – transformation type (`Linear, PDelta, Corotational`)

tag – integer identifier unique among all geometric transformations

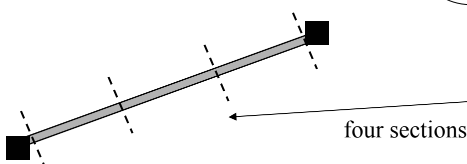
vecXZ – vector defining element orientation (for 3D only); defines direction of section z-axis in space; three values, e.g. 0 0 1

Example element definition

- Using previously defined fiber section with tag 1 and nodes with tags 1 and 2

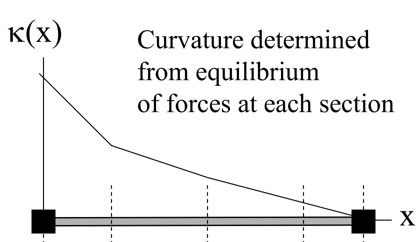
```
geomTransf Linear 1

#          tag ndI ndJ numSec secTag transfTag
element nonlinearBeamColumn 1 1 2 4 1 1
```

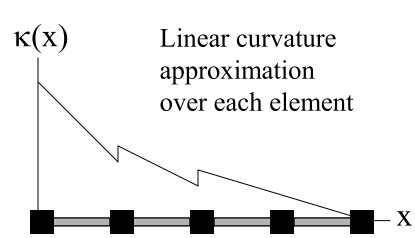


Modeling Rule of Thumb

- For distributed plasticity, only need one force-based element (`nonlinearBeamColumn`) per member, as opposed to several displacement-based elements (`dispBeamColumn`) to capture nonlinear material behavior



Force-based



Disp-based