



OpenSees

Open System for Earthquake Engineering Simulation
Pacific Earthquake Engineering Research Center



Structural Modeling with Examples

Vesna Terzic
UC Berkeley

OpenSees Days 2012

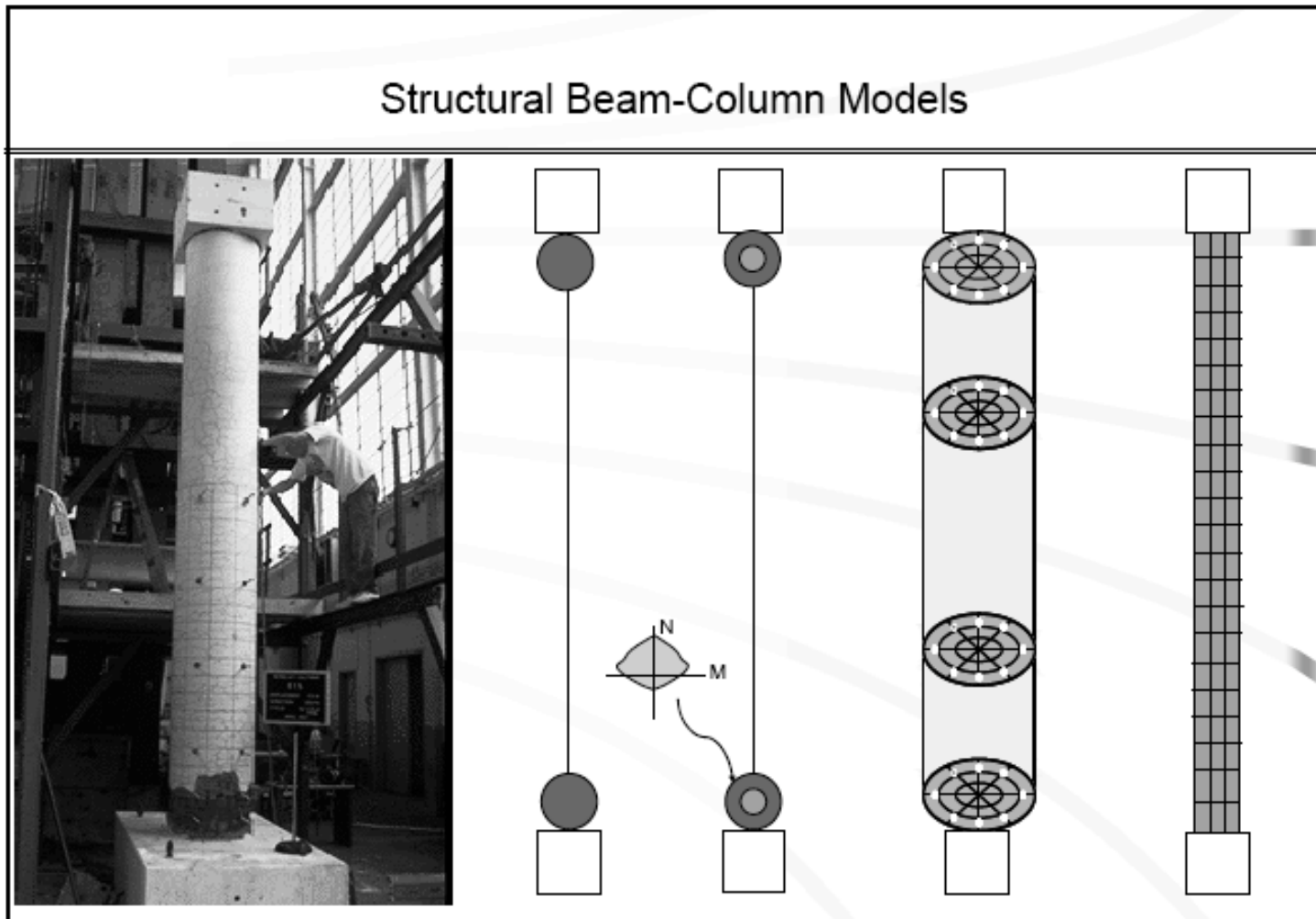
Agenda

- **Nonlinear** modeling using **frame elements with fiber section**
- Creating a fiber section
- Background on nonlinear frame elements of OpenSees
- Examples & Applications
- Q&A with participants

Elements and materials

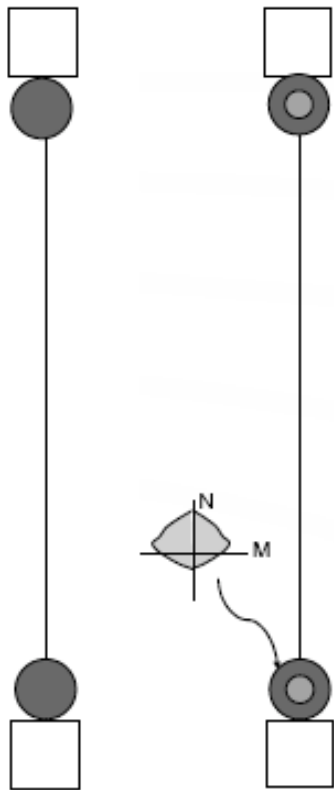
- Elements:
 - Force-based elements (FBE)
 - Displacement-based elements (DBE)
 - Beam with hinges elements (BWH)
- Materials:
 - Reinforced Concrete
 - Steel

Nonlinear Beam Column Modeling



F. Filippou

Concentrated Plasticity Models



Advantages:

- Simple
- Good for Interface Effects (bar pullout, shear sliding)

Disadvantages:

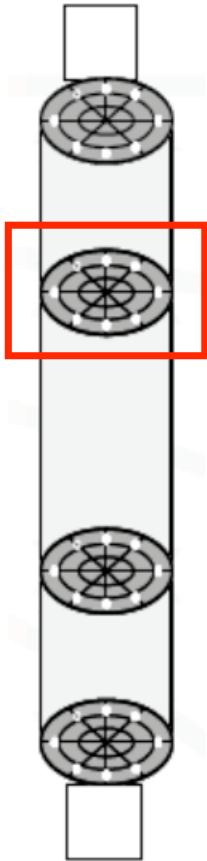
- Requires moment-rotation relationship
- Moment-rotation relationship of element is related to Plastic Hinge Length
- Can only be used if axial load is constant

Distributed Plasticity Models



- Contrary to concentrated plasticity models, force-based element (FBE) and displacement-based element (DBE) permit spread of plasticity along the element (distributed plasticity models).
- Distributed plasticity models allow yielding to occur at any location along the element, which is especially important in the presence of distributed element loads (girders with high gravity loads).
- If yielding is confined to element ends, beam with hinges element (BWH) that permits spread of plasticity within plastic hinge region is a good alternative to FBE and DBE (especially if strain-softening response is expected)

Introduction



- In presence of **axial loads** in the members, fiber section is recommended to be used as it accounts for **PMM interaction**.
- It is suggested not to use more than 5 integration points.

Creating a fiber section in OpenSees

Each Fiber Section is composed of Fibers, with each fiber containing:

1. Uniaxial Material,
2. Area assigned to a fiber
3. Location of the fiber in local (y,z) system

Command:

```
section Fiber $secTag {  
  fiber...  
  patch...  
  layer...  
  ...  
}
```

Arguments:

| | |
|-----------------|---|
| \$secTag | unique tag among FiberSections |
| fiber... | command to generate a single fiber. |
| patch... | command to generate a number of fibers over a geometric cross-section |
| layer... | command to generate a row of fibers along a geometric-arc |

Fiber command

Creates the single fiber and adds it to the enclosing Fiber Section

fiber SyLoc \$zLoc SA \$matTag

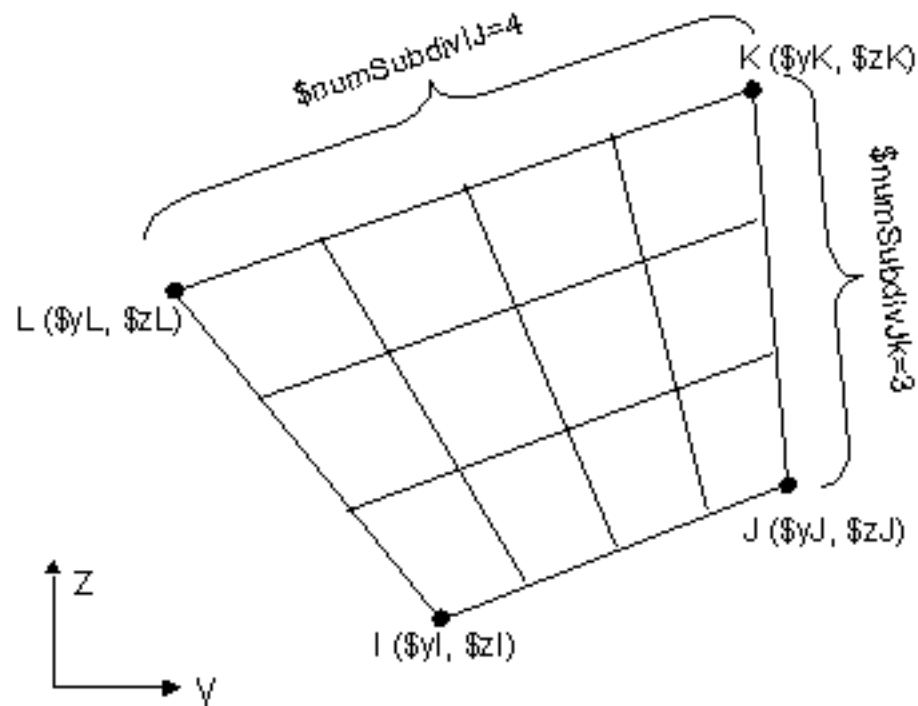
| | |
|-----------------|--|
| SyLoc | y coordinate of the fiber in the section (local coordinate system) |
| \$zLoc | z coordinate of the fiber in the section (local coordinate system) |
| SA | area of the fiber. |
| \$matTag | material tag associated with this fiber. |

Patch command

Generates a number of fibers over a cross-sectional area

Quadrilateral shaped patch:

```
patch quad $matTag $numSubdivIJ $numSubdivJK $yI $zI $yJ $zJ $yK $zK $yL $zL
```



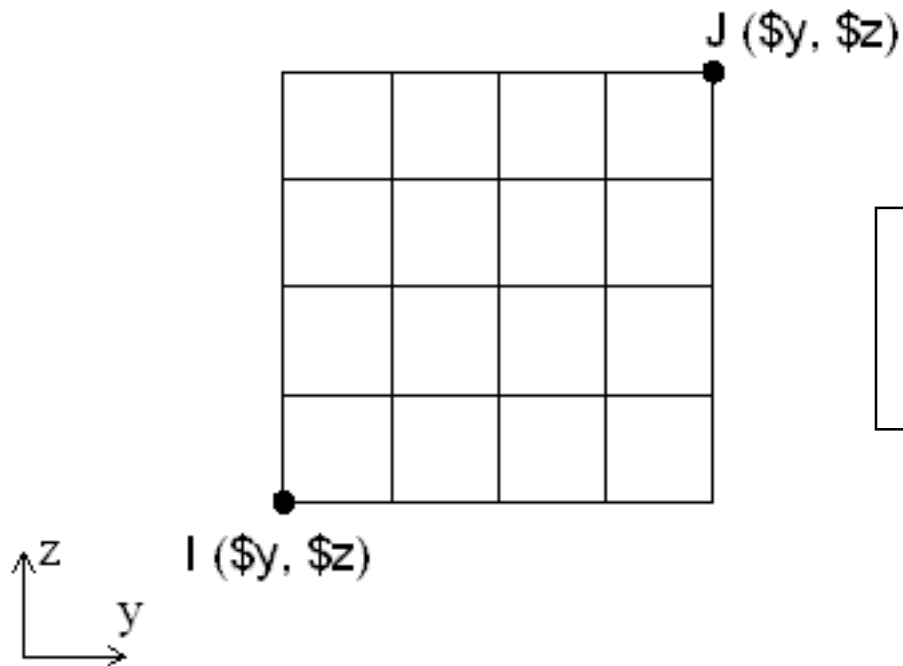
Vertices I J K L
need to be defined in
COUNTER CLOCKWISE
sequence

Patch command

Generates a number of fibers over a cross-sectional area

Rectangular patch:

```
patch rect $matTag $numSubdivY $numSubdivZ $yI $zI $yJ $zJ
```



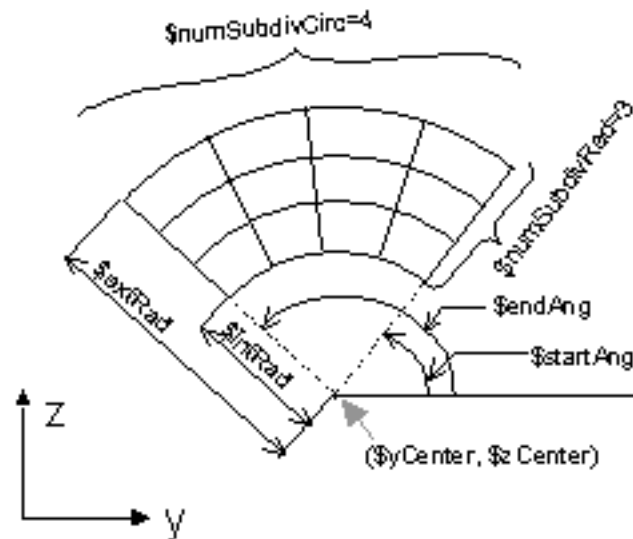
Vertex I is the bottom-left point and the vertex J is the top-right point relative to local coordinates

Patch command

Generates a number of fibers over a cross-sectional area

Circular patch:

```
patch circ $matTag $numSubdivCirc $numSubdivRad $yCenter $zCenter $intRad $extRad <$startAng $endAng>
```

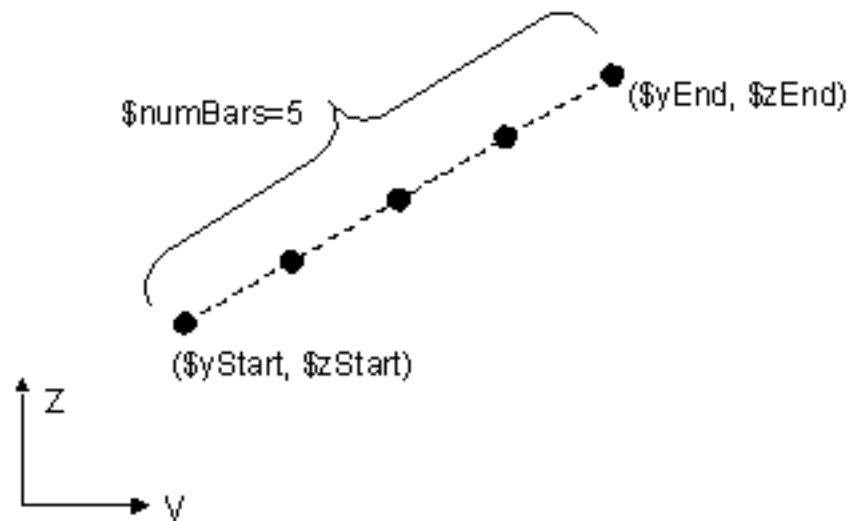


Layer command

Generates a number of fibers along a *line* or a *circular arc*

Straight layer:

```
layer straight $matTag $numFiber $areaFiber $yStart $zStart $yEnd $zEnd
```

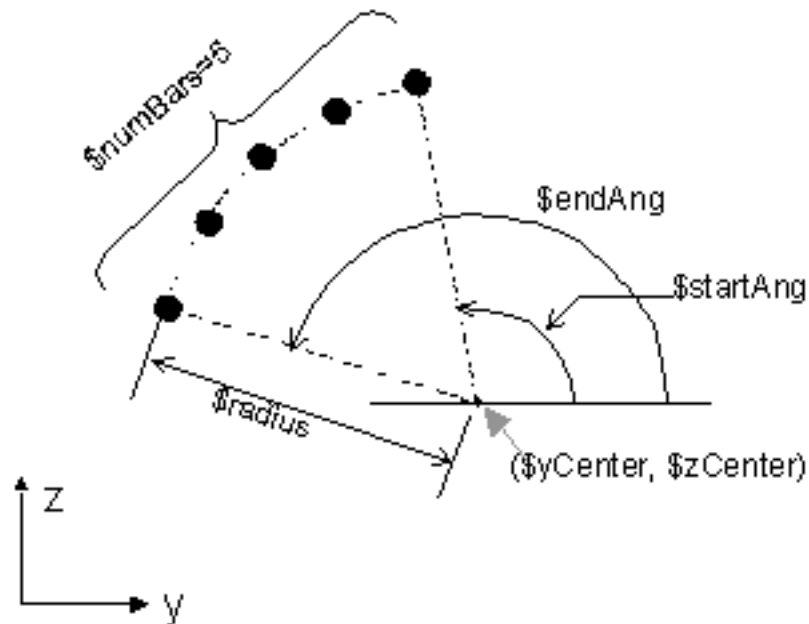


Layer command

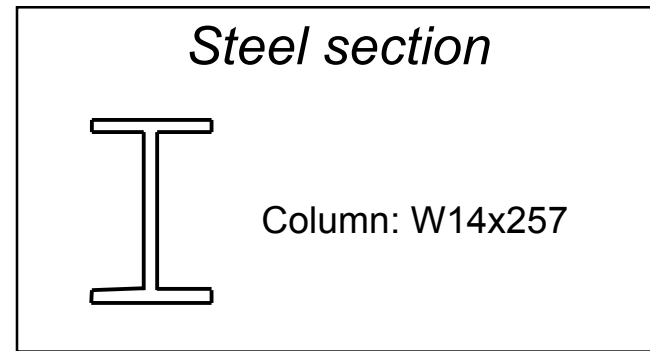
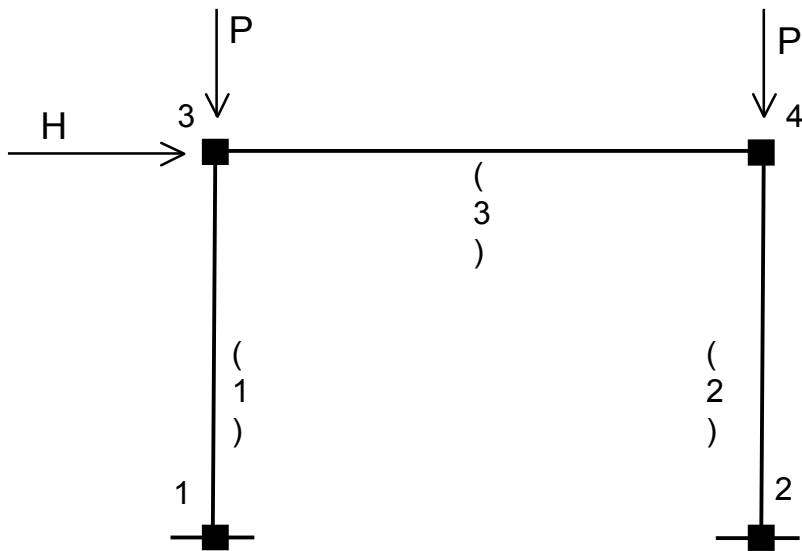
Generates a number of fibers along a *line* or a *circular arc*

Circular layer:

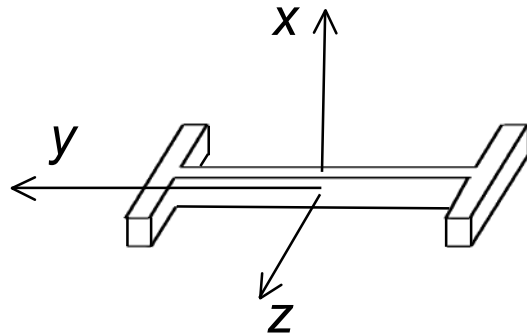
```
layer circ $matTag $numFiber $areaFiber $yCenter $zCenter $radius <$startAng $endAng>
```



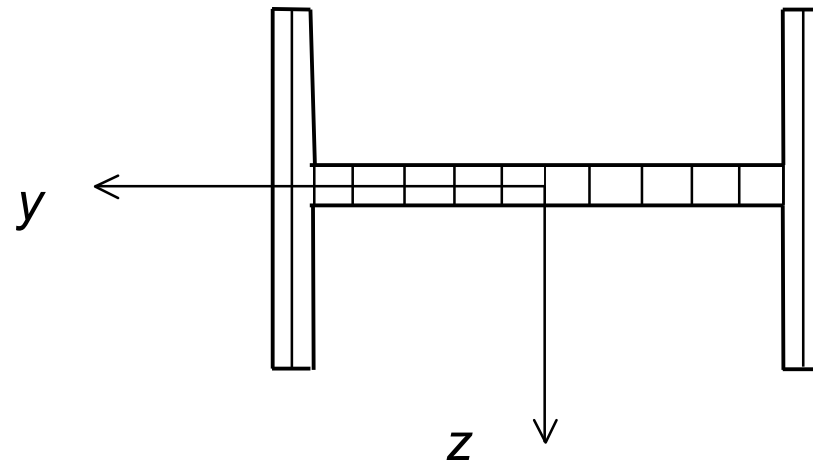
Example: *Steel cross-section – unidirectional loading*



Select Local Axes



Section discretization



Example: Steel cross-section – unidirectional loading

input variables

set d 16.4; # nominal depth

set tw 1.18; # web thickness

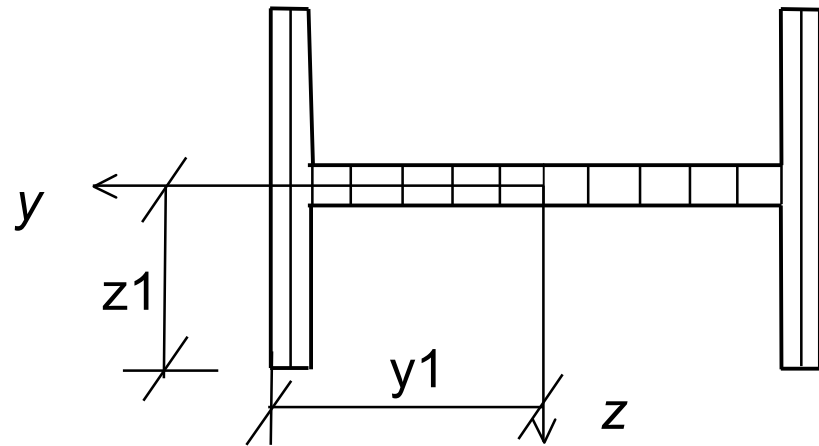
set bf 16.0; # flange width

set tf 1.89; # flange thickness

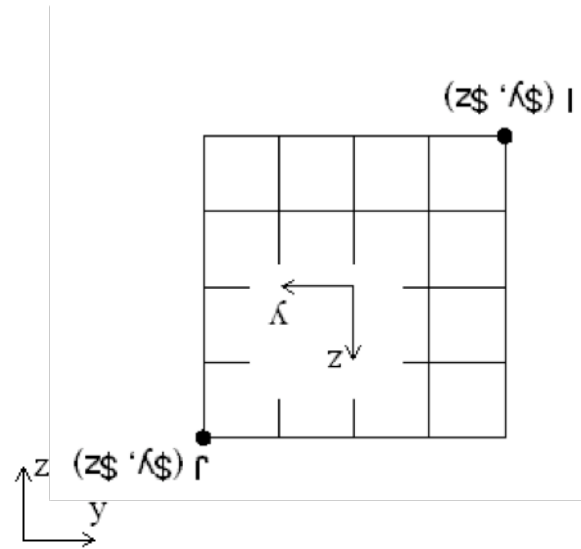
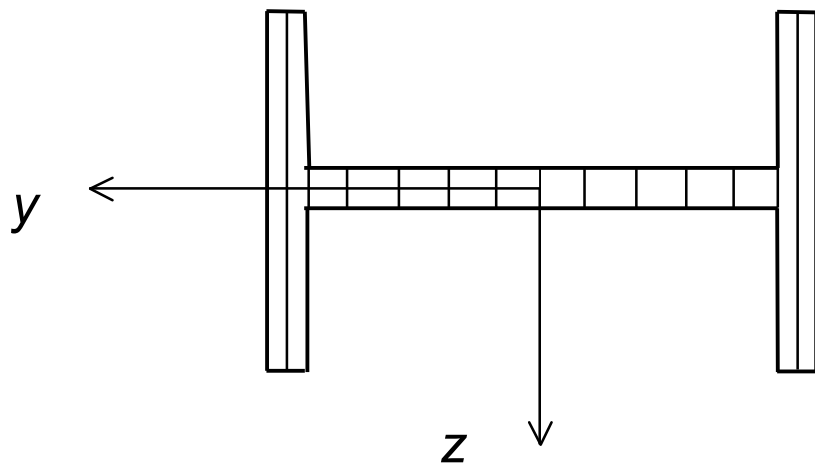
derived variables

set y1 [expr \$d/2.0]

set z1 [expr \$bf/2.0]

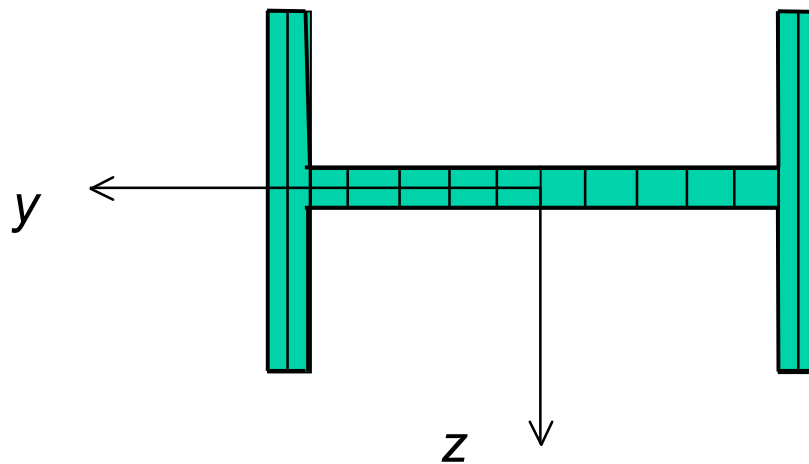


Example: *Steel cross-section – unidirectional loading*

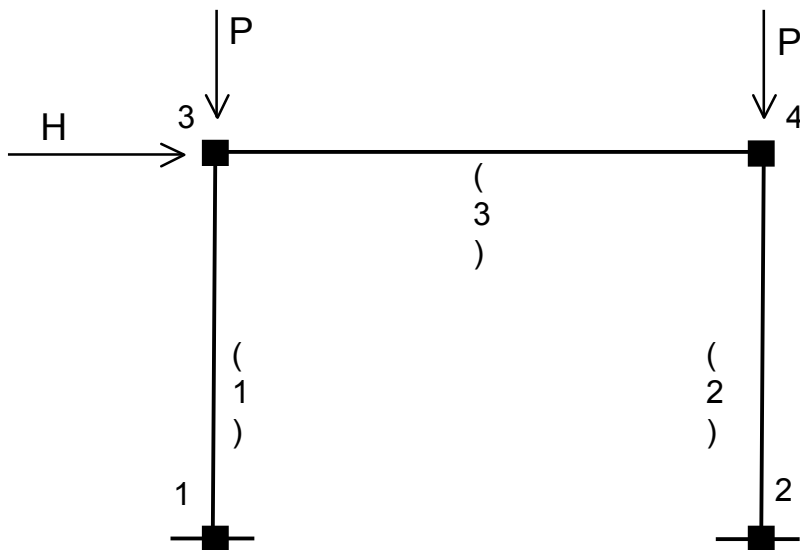


Example: *Steel cross-section – unidirectional loading*

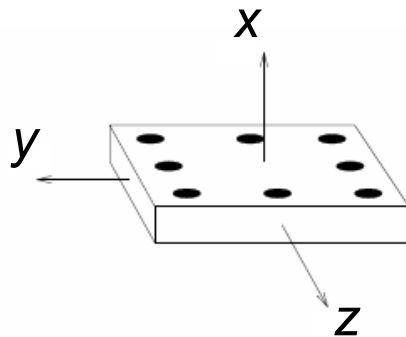
```
section Fiber 1 {  
  # Create the flange fibers (left & right)  
  patch rect $matID 2 1 [expr $y1-$tf] [expr -$z1] $y1 $z1  
  patch rect $matID 2 1 [expr -$y1] [expr -$z1] [expr $tf-$y1] $z1  
  
  # Create the web fibers  
  patch rect $matID 10 1 [expr $tf-$y1] [expr -$tw/2.] [expr $y1-$tf] [expr  
$tw/2.]  
}
```



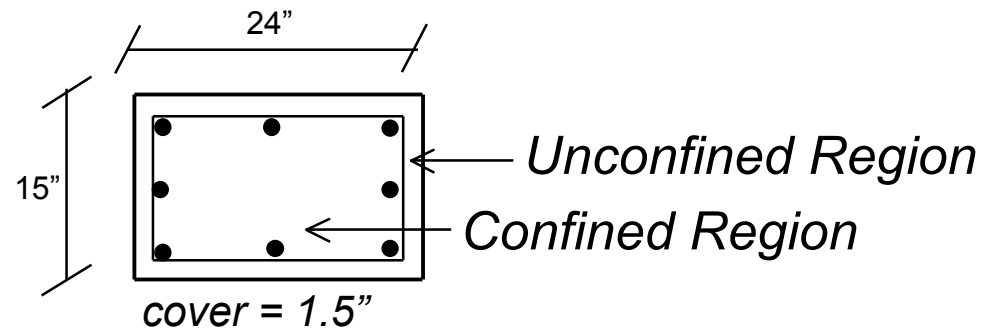
Example: *RC rectangular cross-section – unidirectional loading*



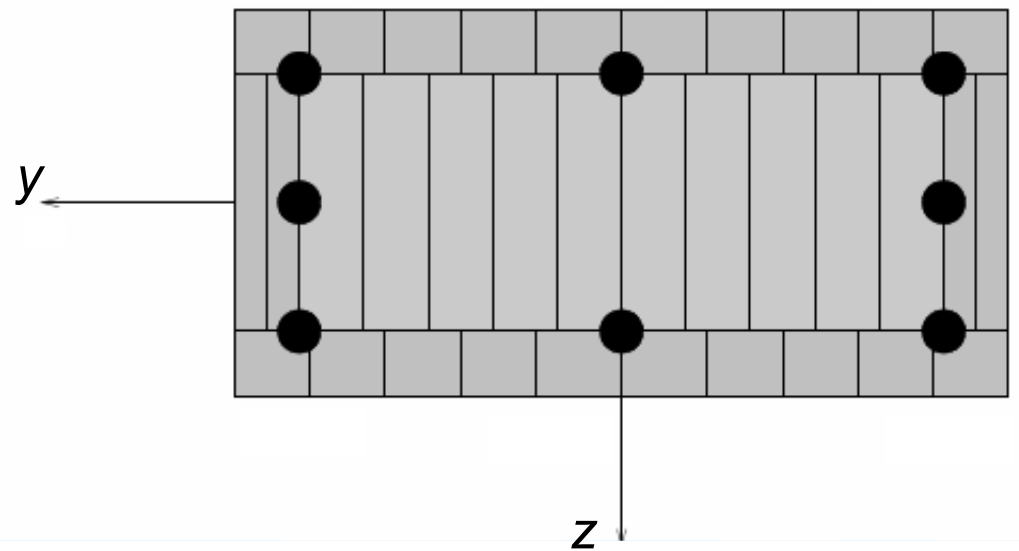
Select Local Axes



Column cross-section

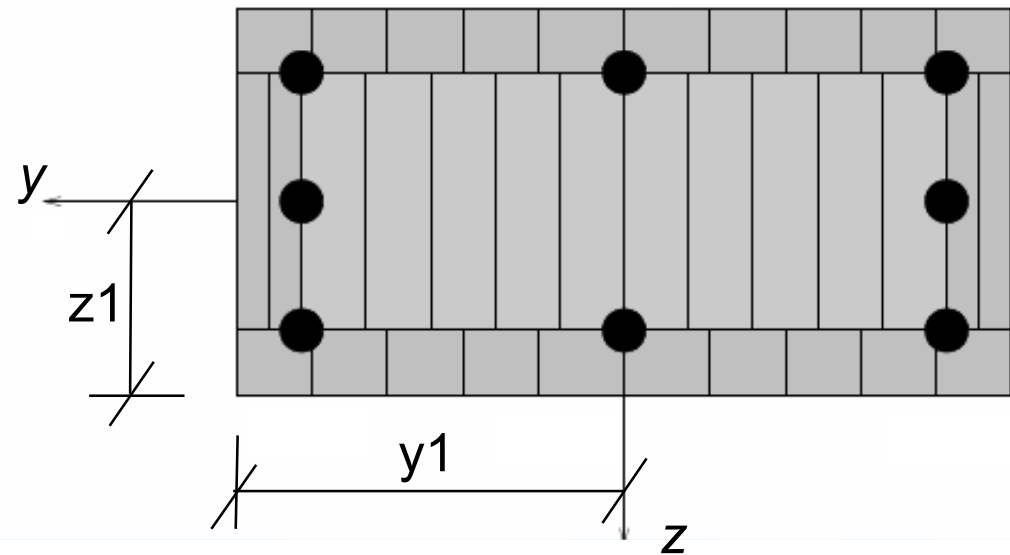


Section discretization



Example: *RC rectangular cross-section – unidirectional loading*

```
# input variables
set b 15.
set h 24.
cover 1.5
set As 0.6
set $nfCore_y 10
set $nfCoverS_y 10
set $nfCoverTB_y 2
set $nf_z 1
set $numBarsTB 3
set $numBarsM 2
```

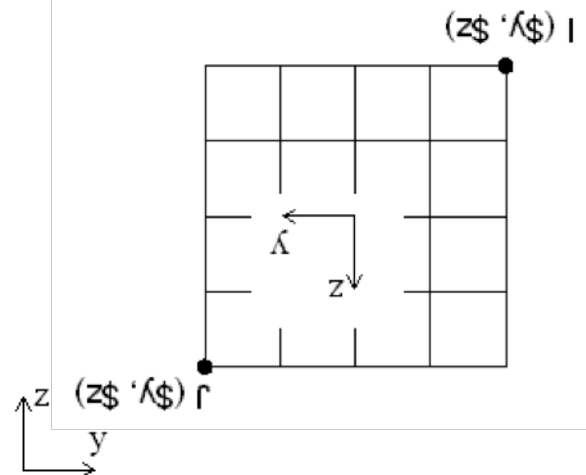
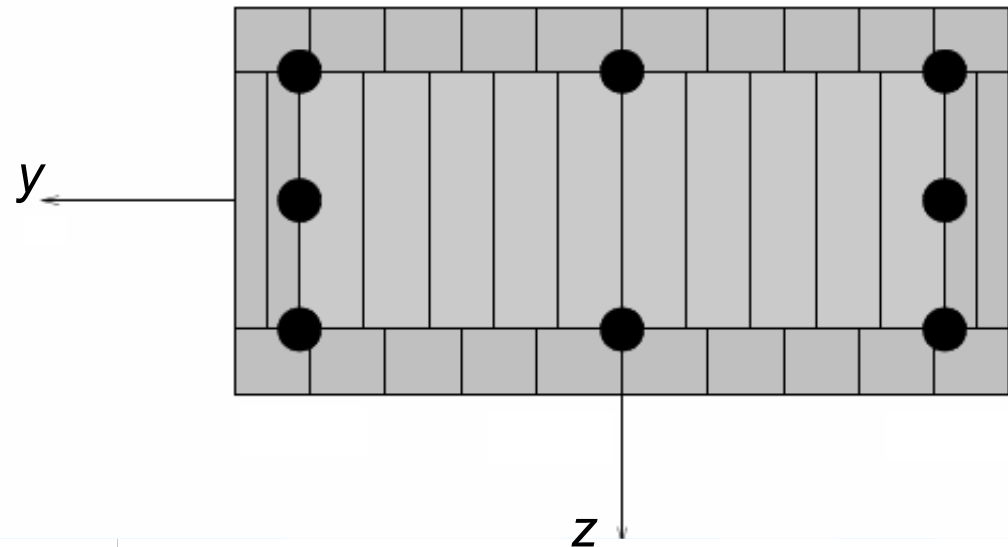


```
# derived variables
set y1 [expr $h/2.0]
set z1 [expr $b/2.0]
```

Example: *RC rectangular cross-section – unidirectional loading*

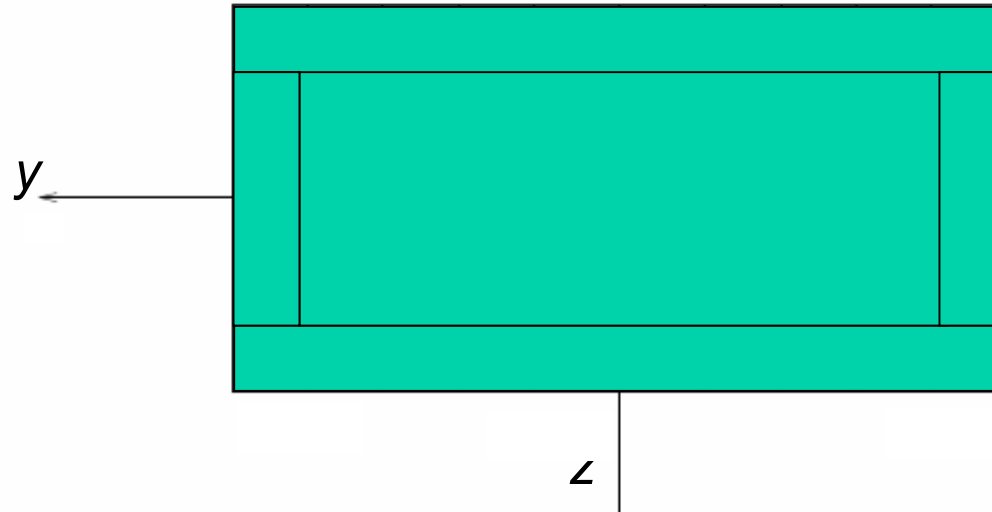
```
# input variables
set b 15.
set h 24.
cover 1.5
set As 0.6
set $nfCore_y 10
set $nfCoverS_y 10
set $nfCoverTB_y 2
set $nf_z 1
```

```
# derived variables
set y1 [expr $h/2.0]
set z1 [expr $b/2.0]
```



Example: *RC rectangular cross-section – unidirectional loading*

```
section Fiber 1 {  
  # Create the concrete core fibers  
  patch rect $coreID $nfCore_y $nf_z [expr $cover-$y1] [expr $cover-$z1] [expr $y1-$cover] [expr  
  $z1-$cover]  
  
  # Create the concrete cover fibers (top, bottom, left, right)  
  patch rect $coverID $nfCoverS_y $nf_z [expr -$y1] [expr $z1-$cover] $y1 $z1  
  patch rect $coverID $nfCoverS_y $nf_z [expr -$y1] [expr -$z1] $y1 [expr $cover-$  
  $z1]  
  patch rect $coverID $nfCoverTB_y $nf_z [expr -$y1] [expr $cover-$z1] [expr $cover-$y1] [expr  
  $z1-$cover]  
  patch rect $coverID $nfCoverTB_y $nf_z [expr $y1-$cover] [expr $cover-$z1] $y1 [expr  
  $z1-$cover]
```



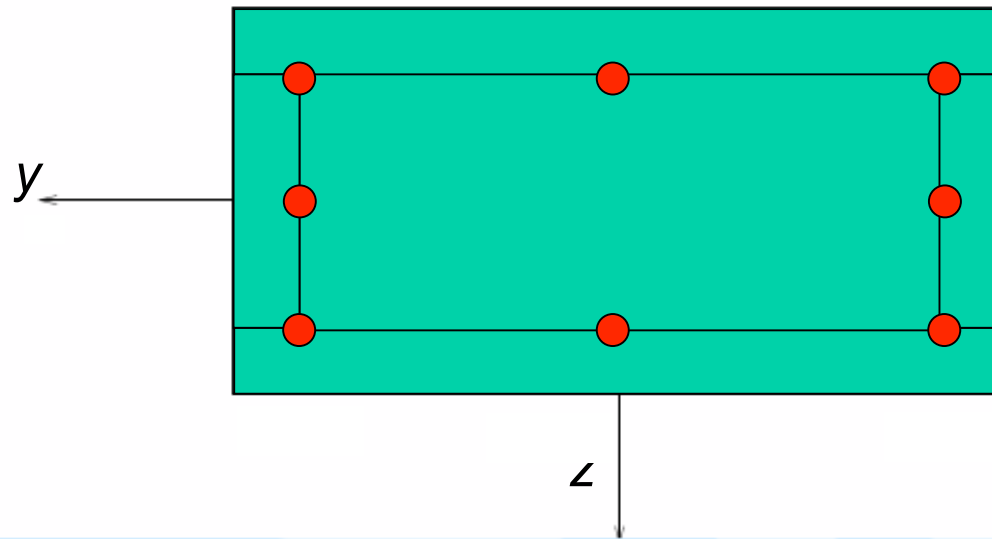
Example: *RC rectangular cross-section – unidirectional loading*

Create the reinforcing fibers (left, right, middle)

```
layer straight $steelID $numBarsTB $As [expr $y1-$cover] [expr $z1-$cover] [expr $y1-$cover] [expr $cover-$z1]
```

```
layer straight $steelID $numBarsTB $As [expr $cover-$y1] [expr $z1-$cover] [expr $cover-$y1] [expr $cover-$z1]
```

```
} layer straight $steelID $numBarsM $As 0. [expr $z1-$cover] 0. [expr $cover-$z1]
```



Nonlinear Elements – FBE & DBE

- OpenSees commands for defining FBE and DBE have the same arguments:

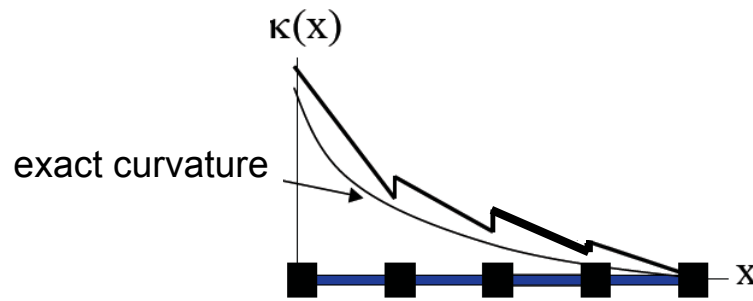
```
element forceBeamColumn $eleTag $iNode $jNode $numIntgrPts $secTag $transfTag
```

```
element displacementBeamColumn $eleTag $iNode $jNode $numIntgrPts $secTag $transfTag
```

- However, to achieve a comparable level of accuracy a beam-column element needs to be modeled differently using these two elements.
- In order to enhance understanding of these two elements and to assure their correct application I will present the theory of these two elements and demonstrate their application on two examples.

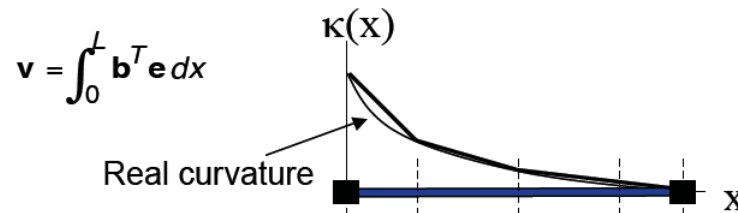
Displacement-based element

- The displacement-based approach follows standard finite element procedures where we interpolate section deformations from an approximate displacement field (constant axial deformation and linear curvature distribution are enforced along the element length, exact only for prismatic linear elastic elements) then use the PVD to form the element equilibrium relationship.
- Use the PVD to form the element equilibrium relationship (“weak equilibrium”)
- Mesh refinement of the element is needed to represent higher order distributions of deformations.



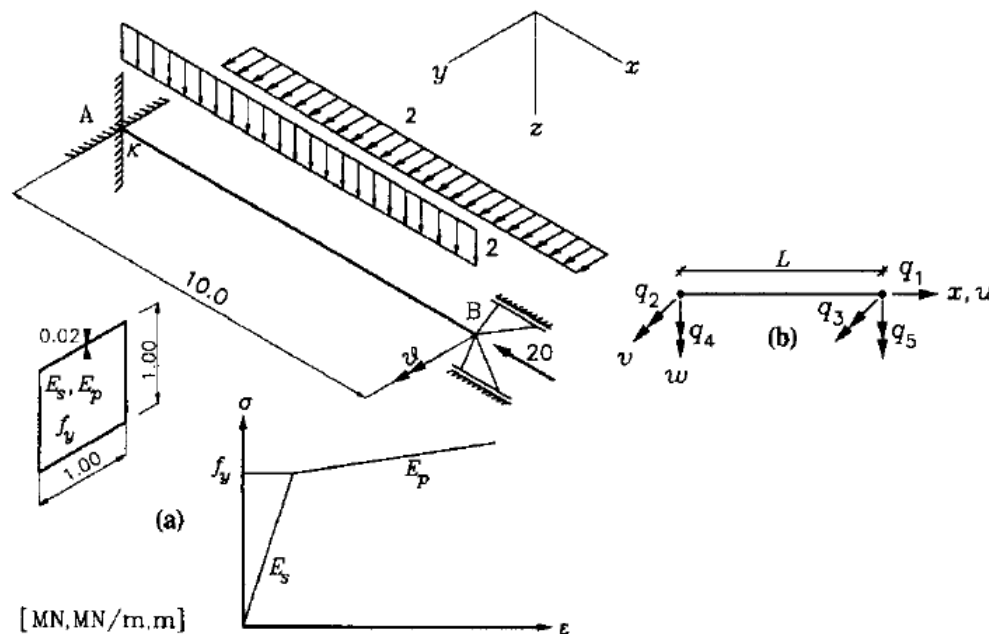
Force-based element

- The force-based approach relies on the availability of an exact equilibrium solution within the basic system of a beam-column element. Equilibrium between element and section forces is exact, which holds in the range of constitutive nonlinearity.
- Section forces are determined from the basic forces by interpolation within the basic system.
 - Interpolation comes from static equilibrium and provides constant axial force and linear distribution of bending moment in the absence of distributed element loads.
- PVF is used to formulate compatibility between section and element deformations:



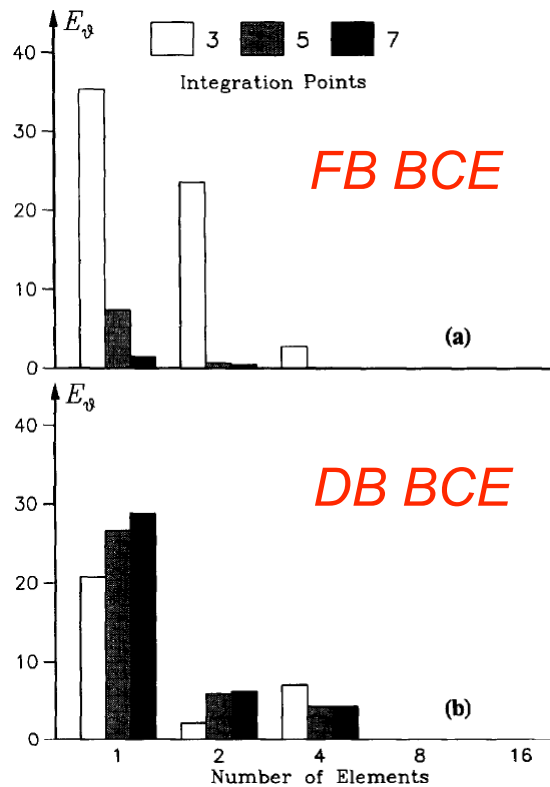
Example 1 – Steel Beam

Neuenhofer, A., and F. C. Filippou, (1997). "Evaluation of Nonlinear Frame Finite Element Models." *Journal of Structural Engineering*, 123(7): 958-966.

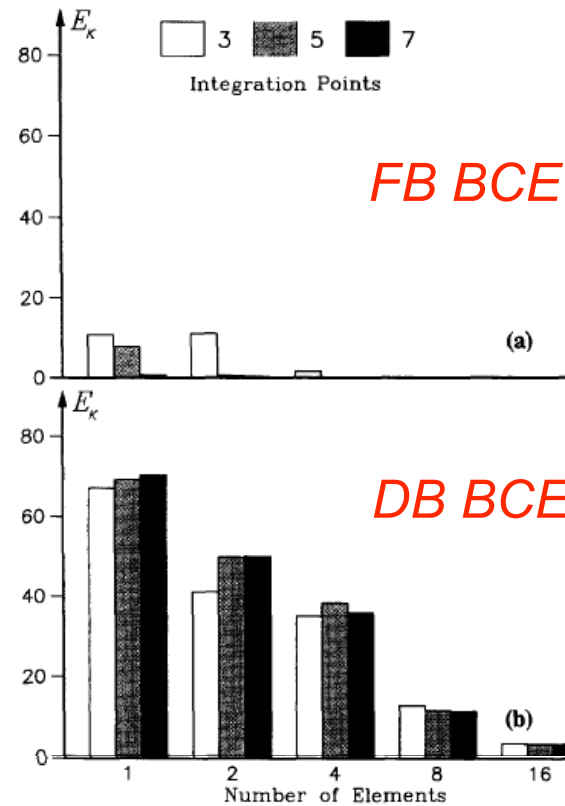


Example 1 – Results

Rotation error (node B)
Global response

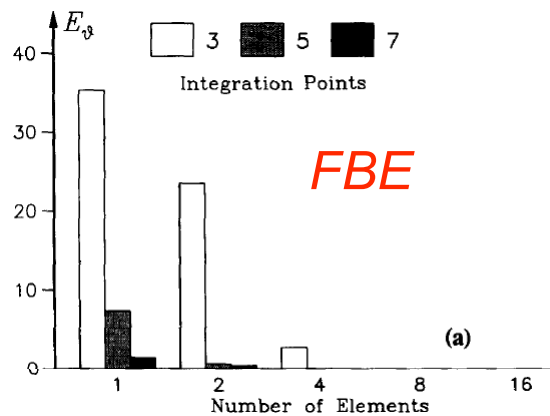


Curvature error (node A)
Local response

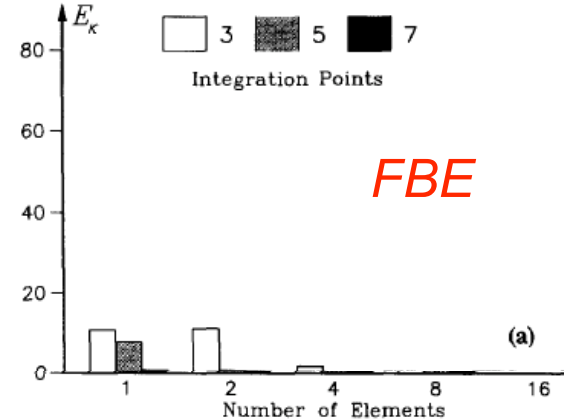


Example 1 – FBE

Rotation error (node B)
Global response



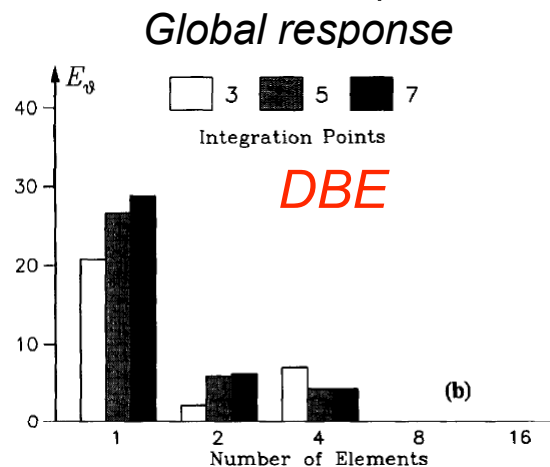
Curvature error (node A)
Local response



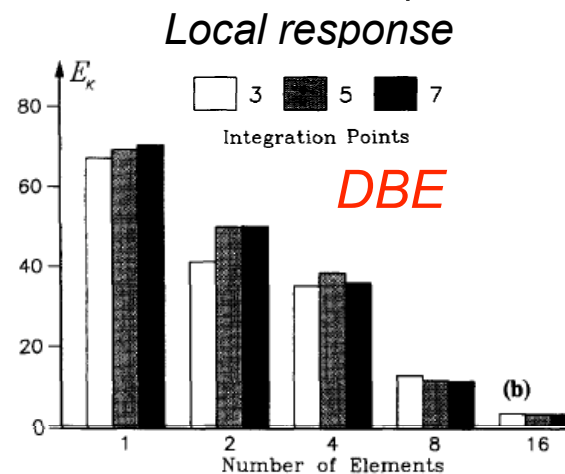
- Accuracy of the solution can be improved by either increasing the NIP (preferable from a computational standpoint) or the number of elements. This is due to the fact that FBE uses the exact force interpolation functions.
- An error less than 2% is obtained for both global and local response quantities with only one element and 7 IPs.

Example 1 – DBE

Rotation error (node B)



Curvature error (node A)



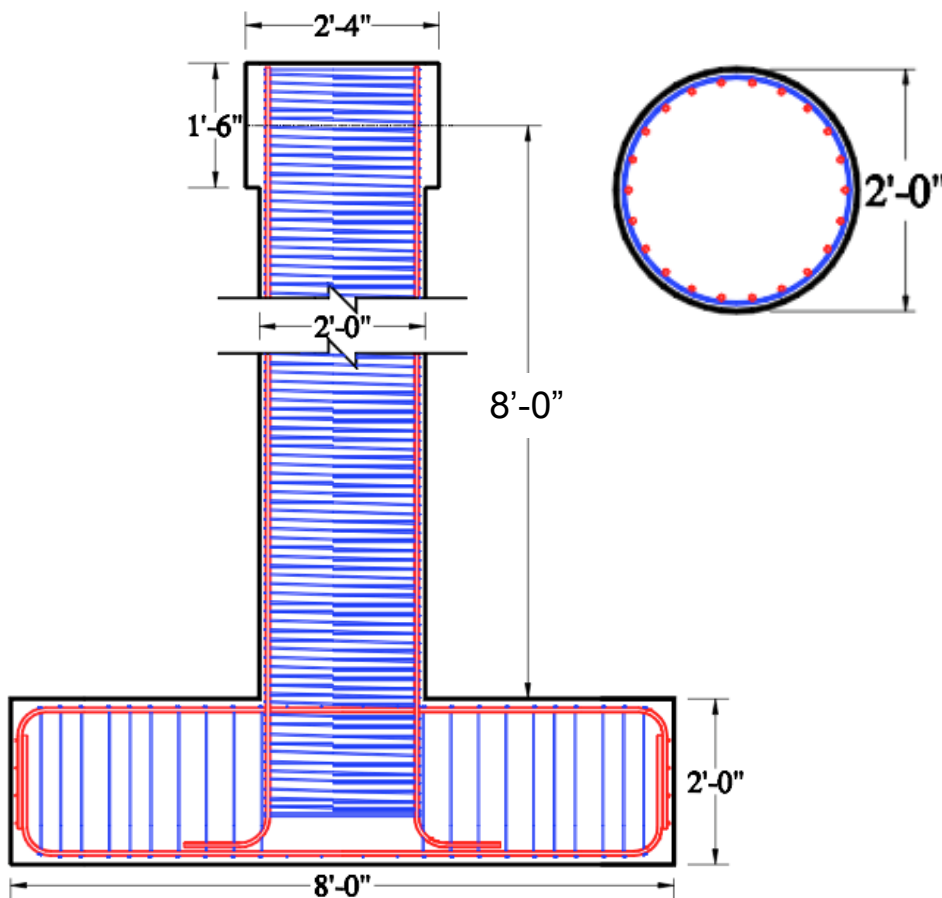
- Accuracy of the solution can only be improved by increasing the number of elements (not by increasing the NIPs). This is due to the fact that DBE uses displacement interpolation functions that approximate the exact solution.
- 8 elements are required to reduce the rotation error to ~0, and 16 elements are required to reduce the curvature error to 3%.

Example 1 - Summary

- Accuracy of the solution can be improved:
 - for FBE, by either increasing the NIPs (preferable from a computational standpoint) or the number of elements,
 - for DBE, only by increasing the number of elements.
- In case of FBE, both local and global quantities converge fast with increasing NIPs.
- In case of DBE, higher derivatives converge slower to the exact solution and thus, accurate determination of local response quantities (e.g., curvature) requires a finer finite-element mesh than the accurate determination of global response quantities (e.g., rotations).

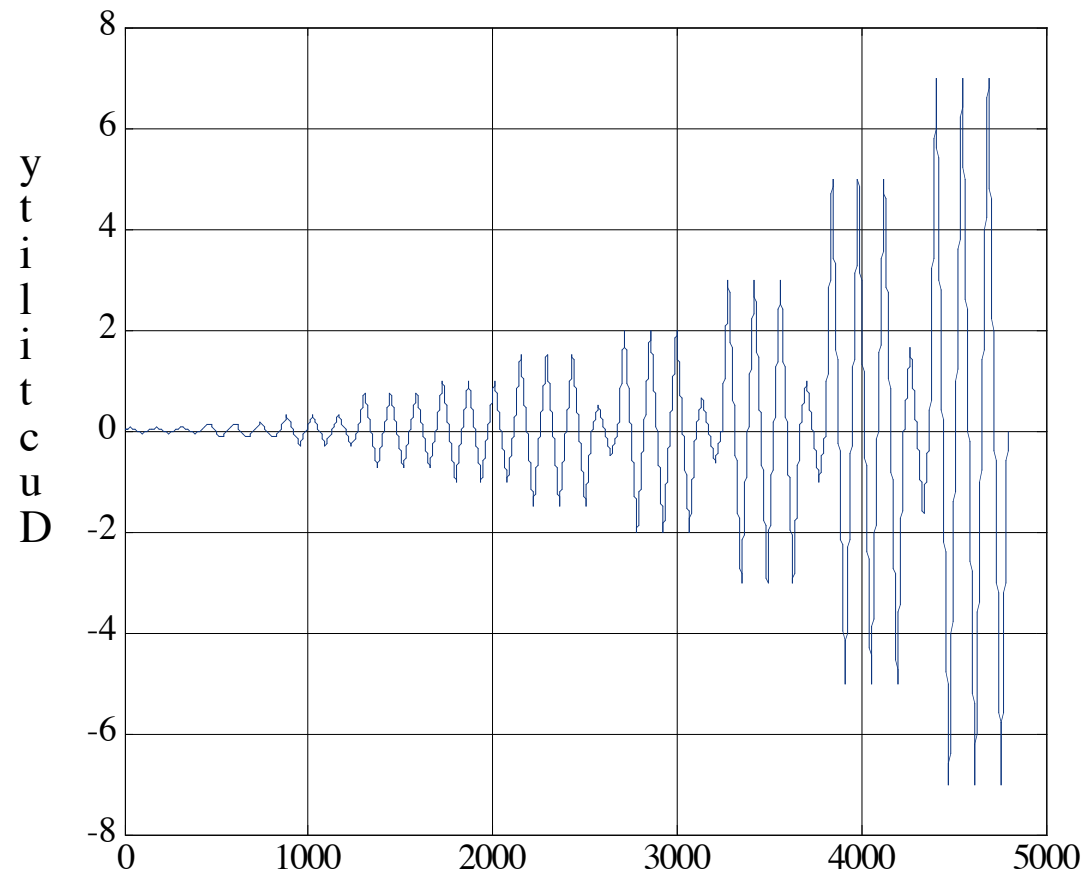
Example 2 – Bridge Column

- Bridge column (Lehman & Moehle, PEER 1998/01 (Column 415))



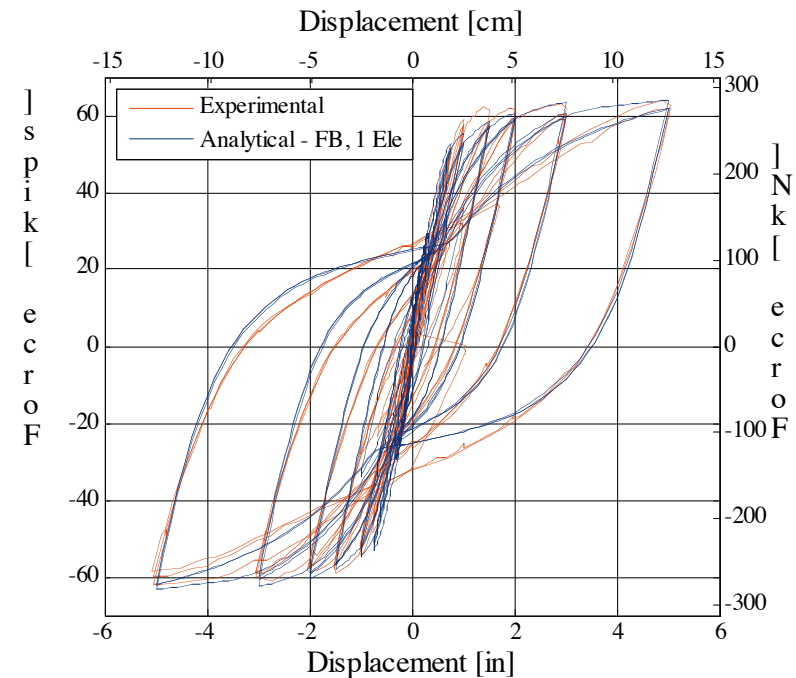
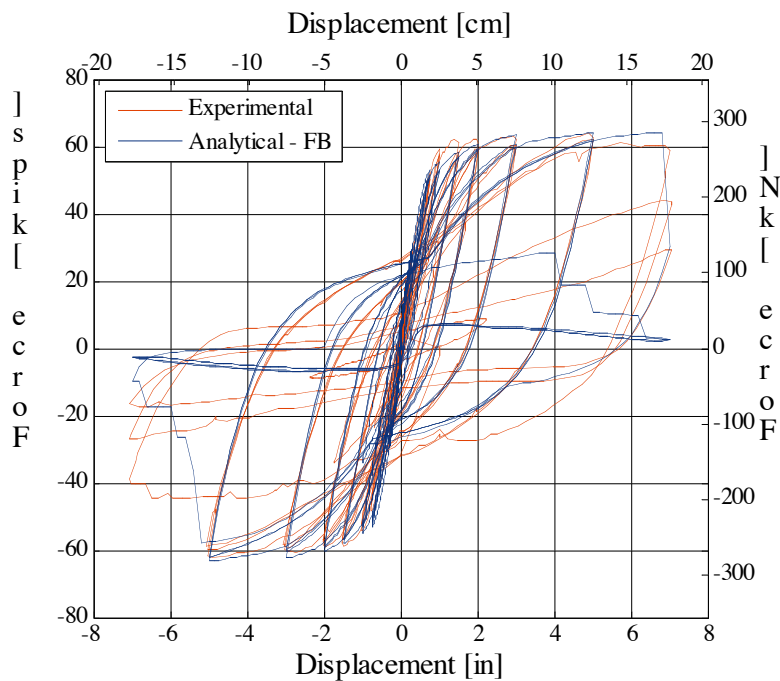
| | |
|----------------------------|--|
| Diameter | 24 in. |
| Height | 96 in. |
| Longitudinal Reinforcement | 22#5 Gr60 ($\rho_l=1.5\%$) $f_y=70\text{ksi}$ |
| Transverse Reinforcement | #2@1.25 in. ($\rho_t=0.7\%$) $f_y=96.6\text{ ksi}$ |
| Concrete | $f_c'=4.4\text{ ksi}$ |

Example 2 - Loading protocol



Example 2 – Model calibration

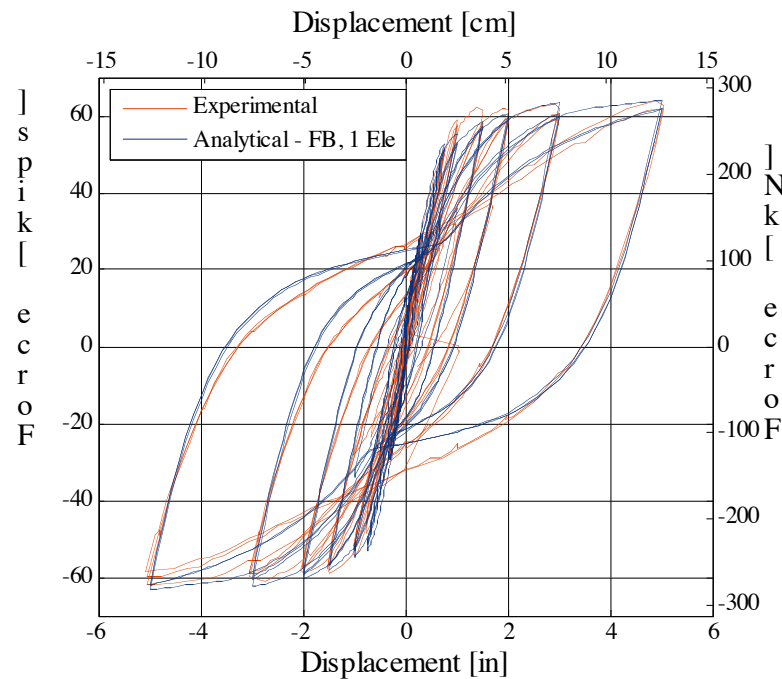
- The column model is calibrated using **force-based element** with 5 integration points. To provide better accuracy of local strains NIPs is chosen such that integration weight of the end node is close to the plastic hinge length.



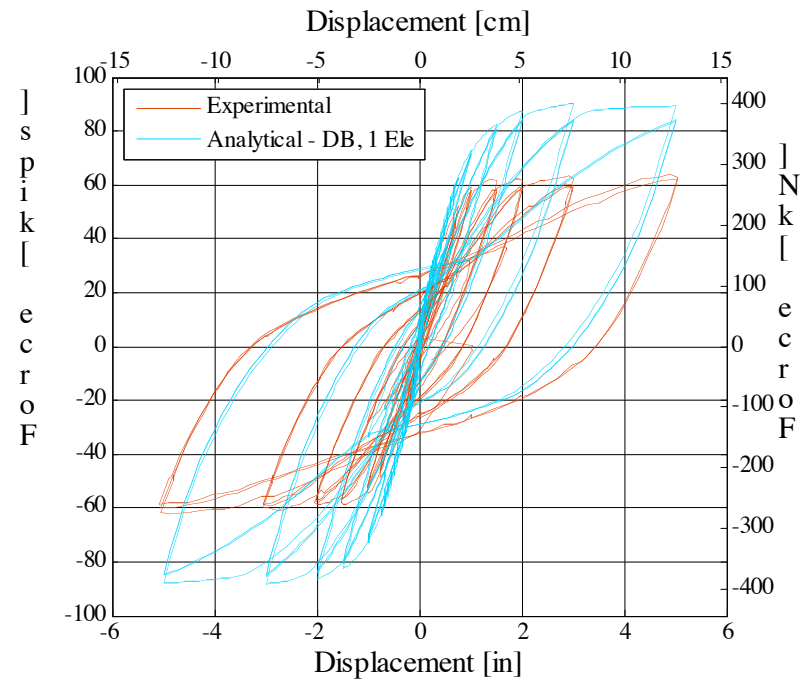
Example 2 – FBE vs. DBE

- The response will change significantly by replacing the force-based beam-column element with the displacement-based beam-column element.

FBE vs. Experiment (NIP = 5)



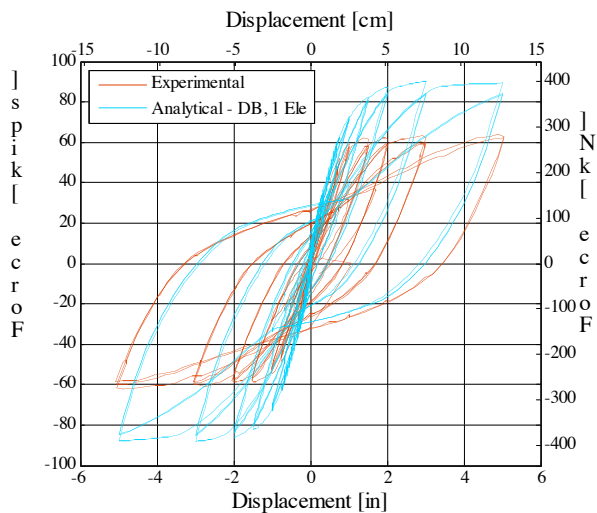
DBE vs. Experiment (NIP = 3)



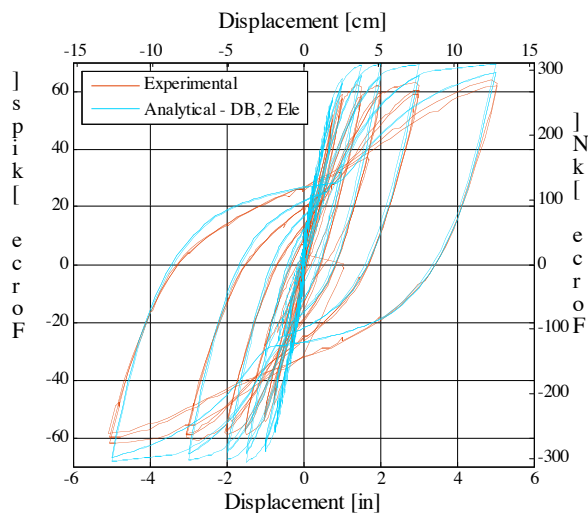
Example 2 - DBE

- With the increase of number of DBE the analytical prediction better matches the measured response of the column.

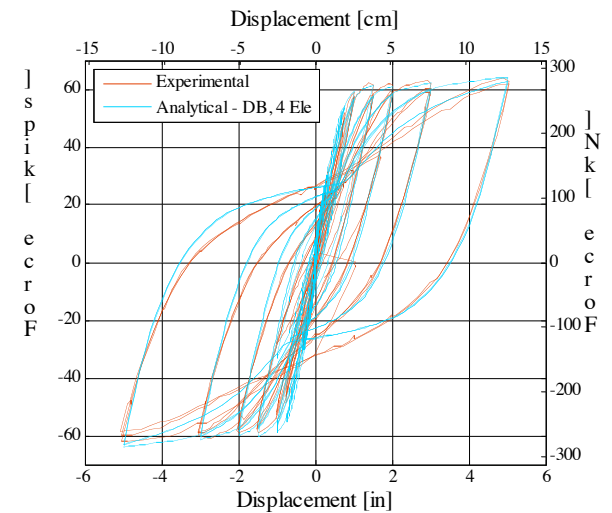
DBE vs. Experiment
Column modeled with 1 element



DBE vs. Experiment
Column modeled with 2 elements



DBE vs. Experiment
Column modeled with 4 elements

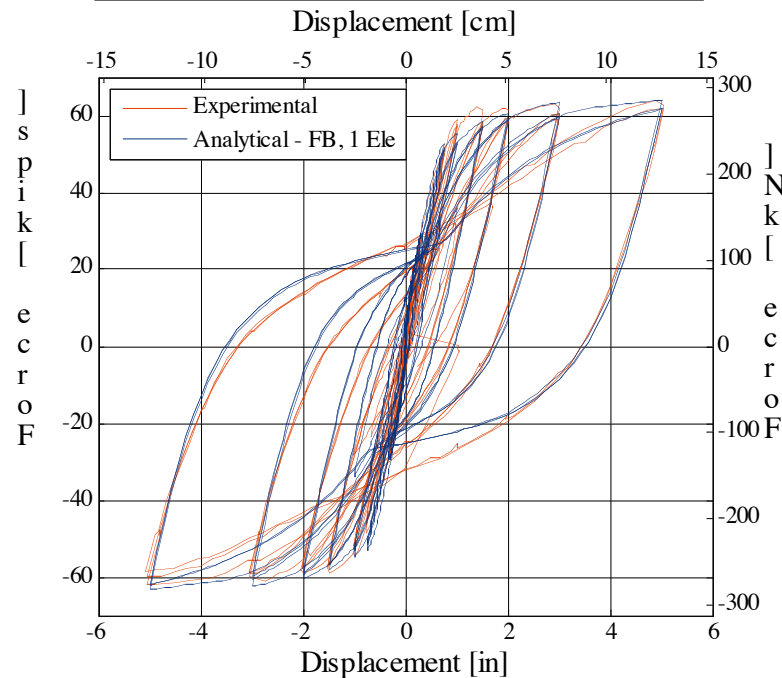


Example 2 - Summary

- To match the measured column response, the column had to be modeled with either **1 FBE** or **4 DBE**.
- Local response quantities could not be compared due to the lack of experimental data. However, it is advisable to use more than 4 DBE when predicting local response quantities.

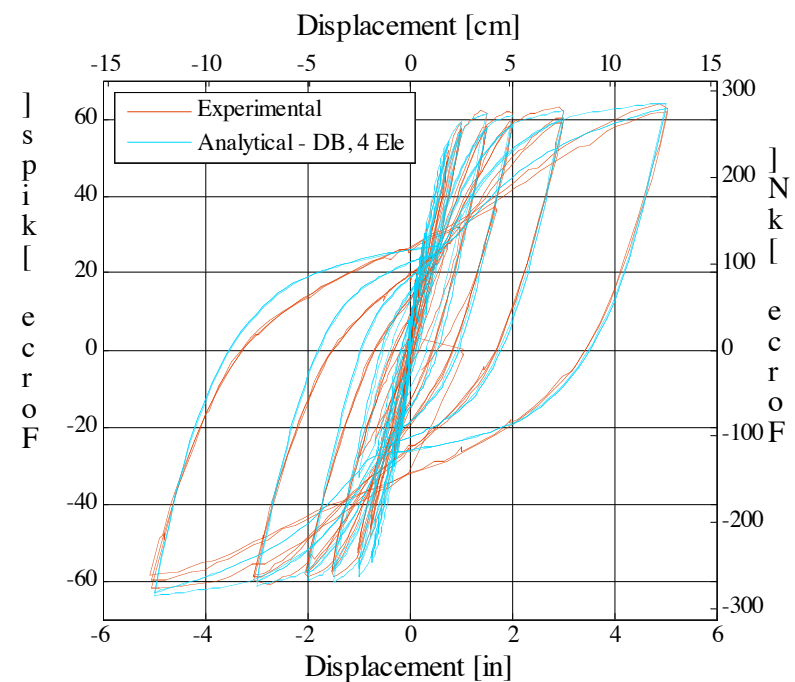
FBE vs. Experiment

Column modeled with 1 element

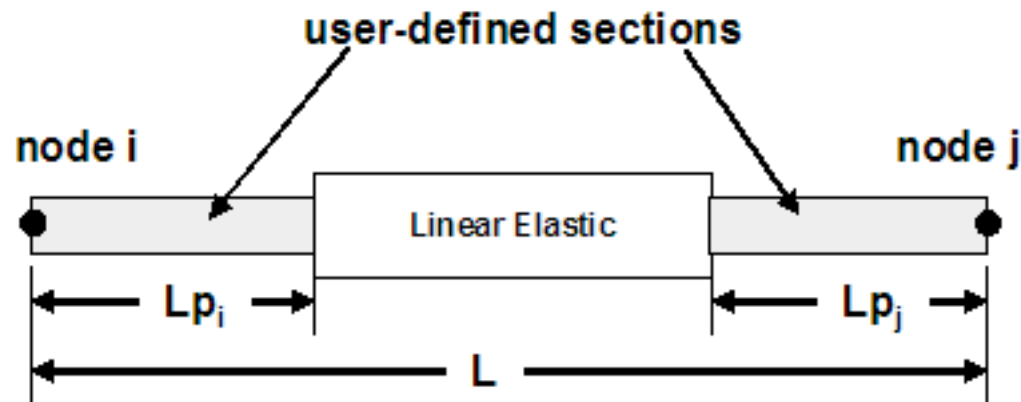


DBE vs. Experiment

Column modeled with 4 elements



Beam with Hinges element (BWH)



- It is a force-based element with plastic hinges localized at element ends; middle portion stays elastic
- Integration points are localized in the hinge regions (two integration points per hinge)
- Integration method is Modified Gauss-Radau

For 2D:

```
element beamWithHinges SeleTag SiNode SjNode $secTagI $Lpi $secTagJ $Lpj SE SA Siz $stransfTag <-mass $massDens> <-iter $maxIters $tol>
```

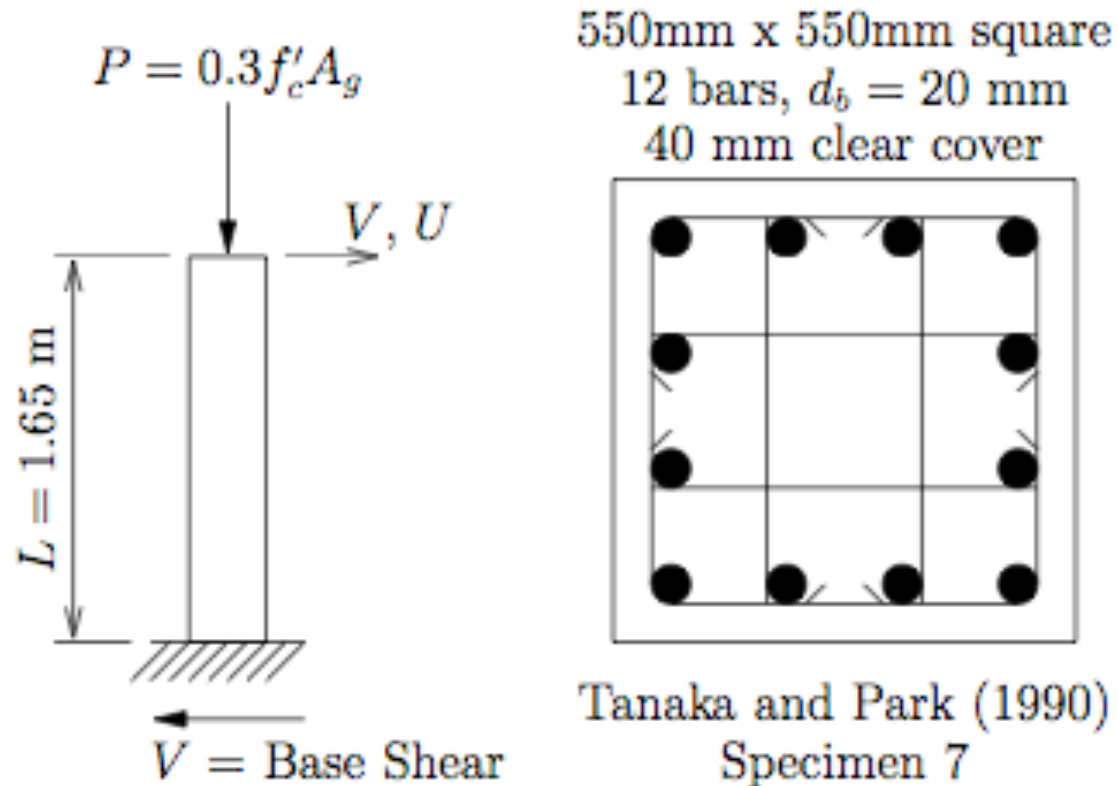
For 3D:

```
element beamWithHinges SeleTag SiNode SjNode $secTagI $Lpi $secTagJ $Lpj SE SA Siz $ly $G $J $stransfTag <-mass $massDens> <-iter $maxIters $tol>
```

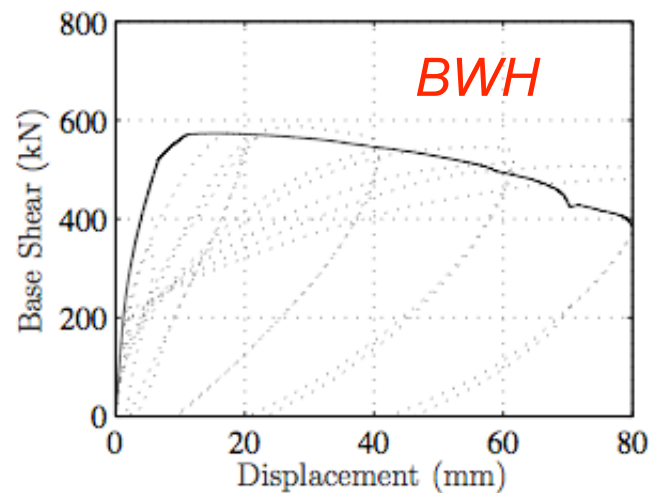
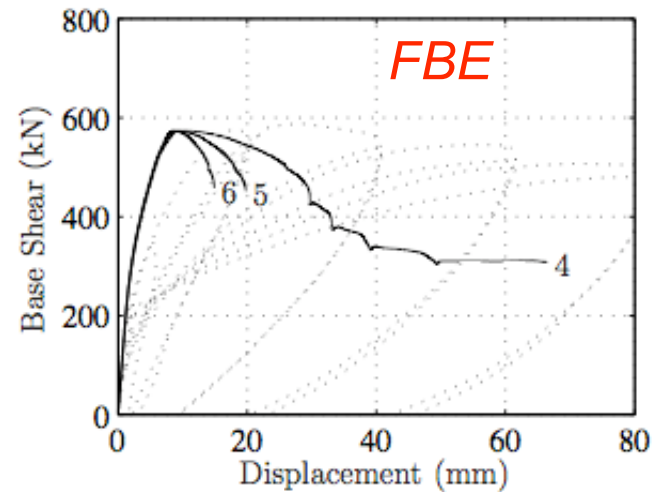
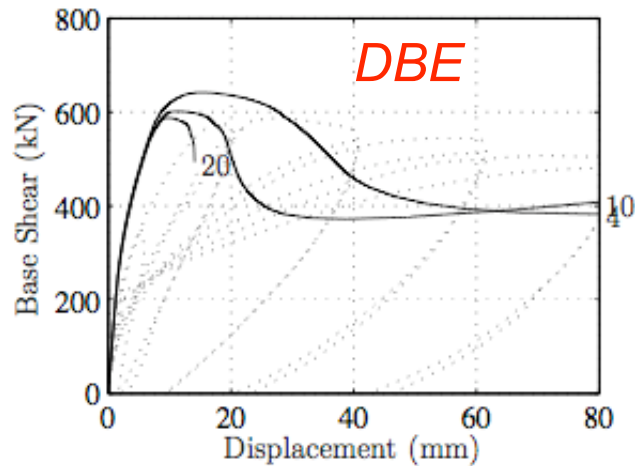
Characteristics of the BWH

- Nonlinear behavior is confined to the integration points at the element ends.
- The user only needs to specify the length of each hinge.
- Captures largest bending moment at the ends.
- Represents linear curvature distributions exactly.
- Objective in capturing strain-softening response
 - FBE localizes deformation at an IP
 - DBE localizes deformation within a single element

Example (from M.H.Scott)



Comparison of response using different type of elements



Questions?
