

Apply Earthquake in Any Direction

By: Mohammad Foroughi

www.opensees.ir

opensees.ir@gmail.com

In this example we want to apply earthquake in any direction to a 3D elastic frame by using *UniformExcitation* command in OpenSEES software. Presented method is usable for any other 3D structure. In OpenSEES by *UniformExcitation* command we can apply acceleration in global directions. But it is not possible to apply acceleration in a direction other than global directions. To gain this end we have to rotate model through an introduced angle. So model geometry is dependent on that angle. The intended frame is a one bay two story elastic frame as in fig 1.

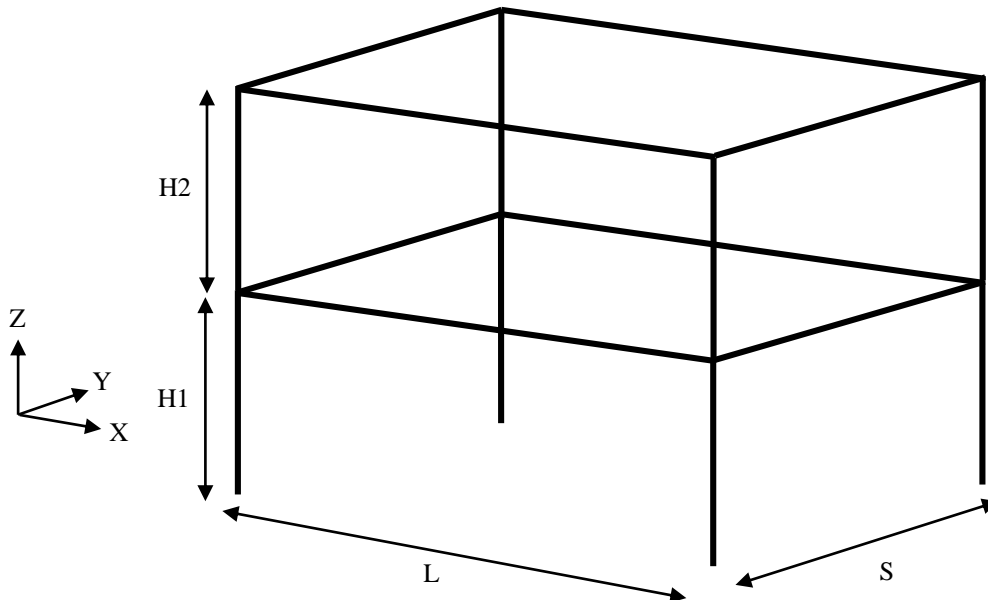


Fig 1: schematic view of the frame

In this frame we tried to introduce parameters for all of geometric components. Parameters L and S are used for bay length in parallel with global X and Y directions, respectively (not rotated frame). H1 and H2 are story heights.

Model Dimension:

As it is a 3D frame model it has 3 dimension with 6 degrees of freedom at each node. So the *model basic* command is as follow:

```
wipe ;  
model BasicBuilder -ndm 3 -ndf 6 ;
```

Define Parameters:

Parameters of this model are as follow:

```
# Define Parameters  
set L 7.0 ;
```

```

set S 4.0 ;
set H1 3.0 ;
set H2 3.0 ;

set Alpha 30.0 ;
set Pi 3.14 ;
set g 9.81 ;
# Loads (Kg/m2)
set DL 500.0
set LL 200.0
set ang [expr int($Alpha)] ;
set FileName "OutPut_$ang" ;
file mkdir $FileName ;

```

Parameters L, S, H1 and H2 are geometric parameters and defined with *set* command. Note that units that are used in this model are N, m and sec. Alpha is the angle between longitudinal direction of the frame and global X direction in degrees. DL and LL are distributed dead and live load of the floors in kg/m2. In last two lines we create a folder that contain the rotation angle in its name.

Define Nodes:

This step is one of the important steps in this model procedure as their coordinates have to define dependent to rotation angle. In fig 2 you can see location of base nodes as function of rotation angle. Other nodes of first and second floor are defined such base nodes with deferent Z coordinates. Global X and Y directions are shown un fig 2. We define nodes 1 to 4 as function of rotation angle, considering node 1 to be origin of the global coordinates. We define new parameter D to be diameter of the rectangular shape plan, also beta and theta are new angels that are shown in fig 2.

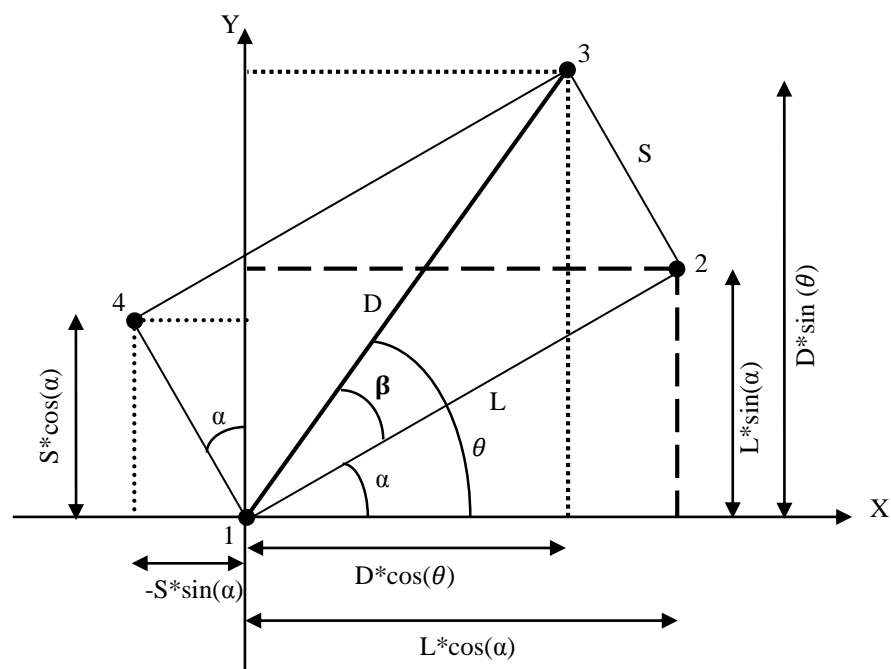


Fig 2: Define Base Nodes

As angels have to define in radian we create new parameter, AlphaRad, to make degrees to radian for next calculations. All nodes of the frame are defined as follow:

```
# Define Nodes
set AlphaRad [expr $Alpha*$Pi/180.0] ;
set Beta [expr atan($S/$L)] ;
set Teta [expr $AlphaRad+$Beta] ;
set D [expr [expr pow((pow($S,2.0)+pow($L,2.0)),0.5)]] ;

# On Ground
#   Tag   X                               Y                               Z
node 1    0.0                               0.0                               0.0
;
node 2    [expr $L*cos($AlphaRad)]         [expr $L*sin($AlphaRad)]
0.0;
node 3    [expr $D*cos($Teta)]             [expr $D*sin($Teta)]
0.0;
node 4    [expr -$S*sin($AlphaRad)]        [expr $S*cos($AlphaRad)]
0.0;

# First Floor
#   Tag   X                               Y                               Z
node 101  0.0                               0.0                               $H1
;
node 102  [expr $L*cos($AlphaRad)]         [expr $L*sin($AlphaRad)]
$H1;
node 103  [expr $D*cos($Teta)]             [expr $D*sin($Teta)]
$H1;
node 104  [expr -$S*sin($AlphaRad)]        [expr $S*cos($AlphaRad)]
$H1;

# Second Floor
#   Tag   X                               Y                               Z
node 201  0.0                               0.0                               [expr $H1+$H2];
node 202  [expr $L*cos($AlphaRad)]         [expr $L*sin($AlphaRad)]
[expr $H1+$H2] ;
node 203  [expr $D*cos($Teta)]             [expr $D*sin($Teta)]
[expr $H1+$H2] ;
node 204  [expr -$S*sin($AlphaRad)]        [expr $S*cos($AlphaRad)]
[expr $H1+$H2] ;
```

Boundary Conditions:

In this frame all base nodes are fix to represent fix support as follow:

```
# Define Constrains
fix 1 1 1 1 1 1 1;
fix 2 1 1 1 1 1 1;
fix 3 1 1 1 1 1 1;
fix 4 1 1 1 1 1 1;
```

Define Elements:

In this step we define elastic elements of the frame. The manner of numbering elements is shown in fig 3 and fig 4:

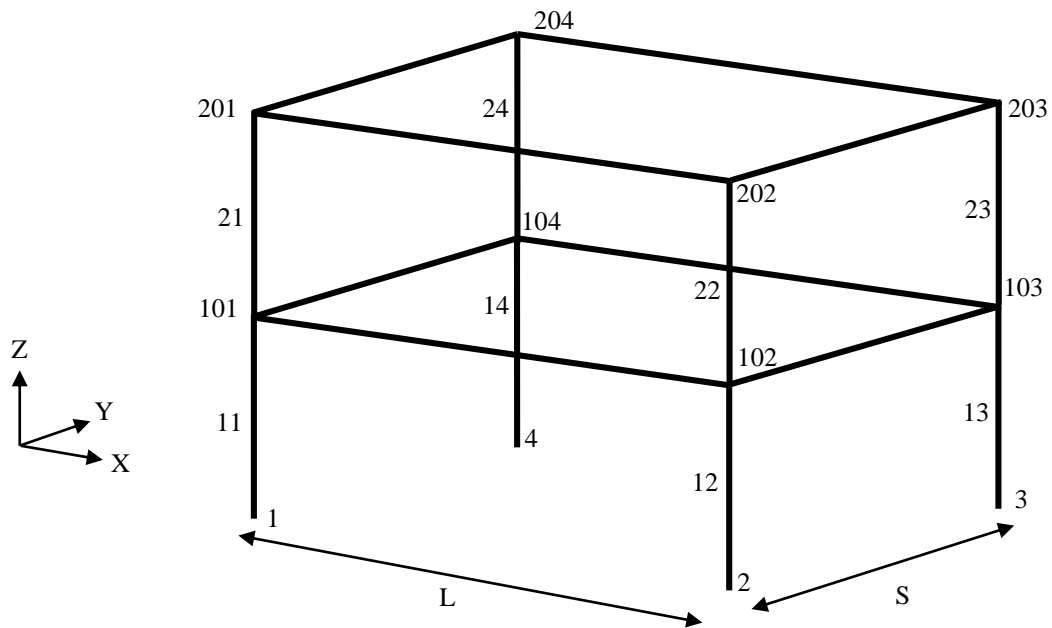


Fig 3: Column Tags

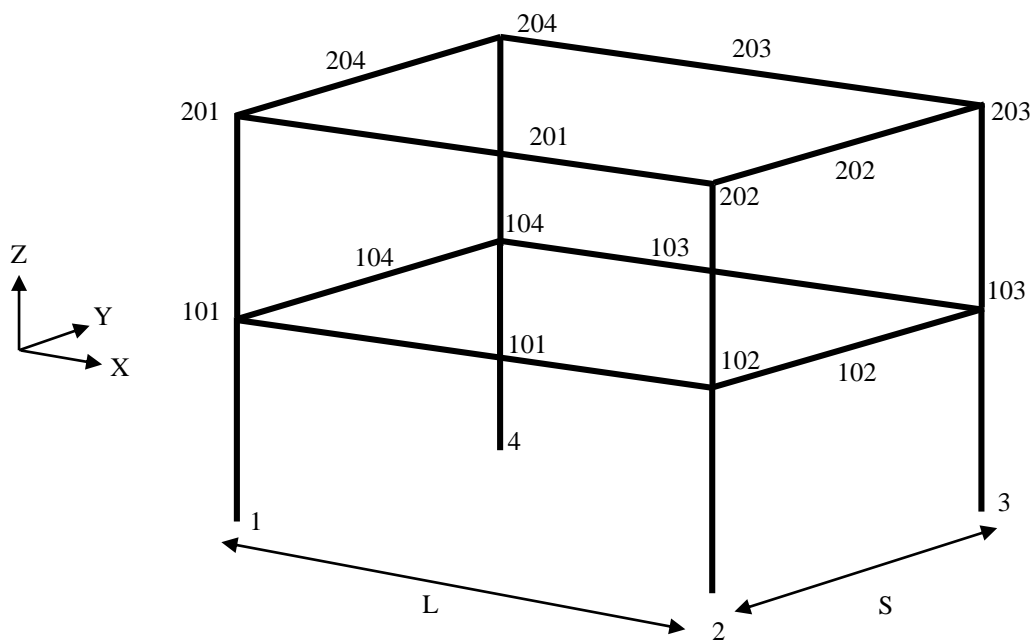


Fig 4: Beam Tags

Before define elements we have to define geometric transformations of elements. We define a Linear transformation for beams and a PDelta transformation for columns. To calculate components of the *geomTransf* command pay attention to fig 5. Geometric transformations are defined as follow:

```
# Define Elements
geomTransf PDelta 1 [expr -cos($AlphaRad)] [expr -
sin($AlphaRad)] 0;
```

```
geomTransf Linear 2 0 0 1 ;
```

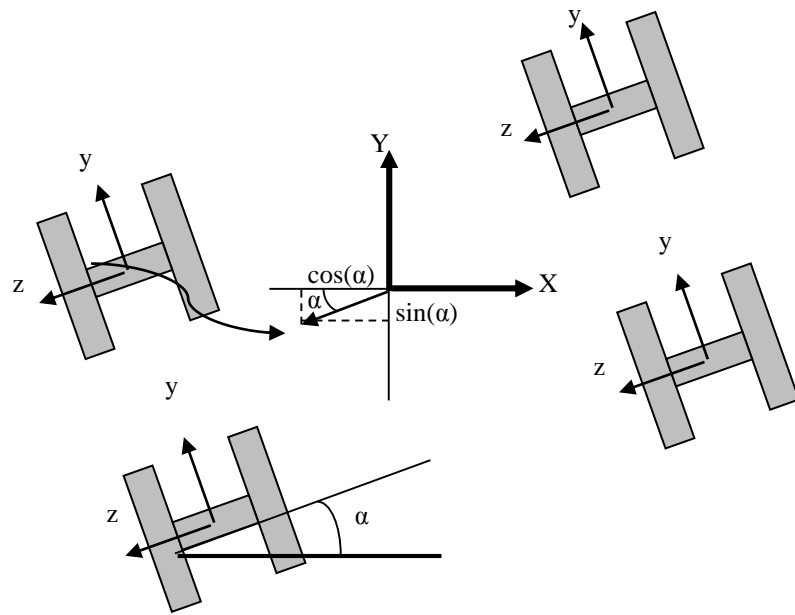


Fig 5: Column Local Coordinate

As it is shown from fig 5 geometric transformation of the columns is related to rotation angle. For beams we consider that the local weak axis is z, so to define transformation components, in all rotation angle local z axis is in the vertical (global Z) direction, and its transformation command is not related to the rotation angle. Fig 6 emphasis this point.

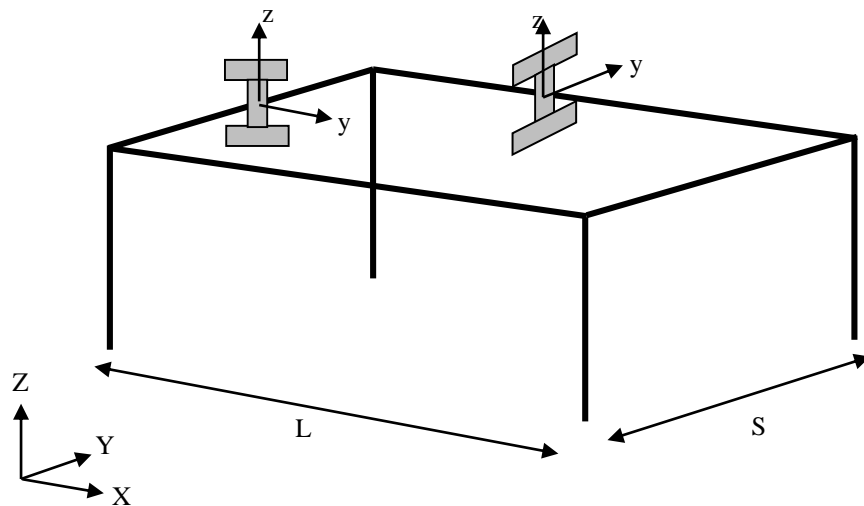


Fig 6: Beams Local Coordinate

We use IPE270 for beams and IPB300 for columns. Now we can define elements of the model as follow:

```
# IPE270
set Iy_B 5790.0E-8 ;
set Iz_B 420.0E-8 ;
set J_B [expr $Iz_B+$Iy_B] ;
set A_B 45.9E-4 ;
```

```
# IPB300
```

```

set Iy_C 25170.0E-8 ;
set Iz_C 8560.0E-8 ;
set J_C [expr $Iz_C+$Iy_C] ;
set A_C 194.0E-4 ;

```

```

set RSteel 7850.0 ;
set E 2.001E11 ;
set G [expr $E/(2.0*(1.0+0.3))] ;

```

```
# Columns
```

```
set mCol [expr $A_C*$RSteel] ;
```

```
# First Floor
```

```

#
# Tag iNode jNode Area E G J Iy
Iz Trans Mass
element elasticBeamColumn 11 1 101 $A_C $E $G $J_C $Iy_C
$Iz_C 1 -mass $mCol ;
element elasticBeamColumn 12 2 102 $A_C $E $G $J_C $Iy_C
$Iz_C 1 -mass $mCol ;
element elasticBeamColumn 13 3 103 $A_C $E $G $J_C $Iy_C
$Iz_C 1 -mass $mCol ;
element elasticBeamColumn 14 4 104 $A_C $E $G $J_C $Iy_C
$Iz_C 1 -mass $mCol ;

```

```
# Second Floor
```

```

element elasticBeamColumn 21 101 201 $A_C $E $G $J_C
$Iy_C $Iz_C 1 -mass $mCol ;
element elasticBeamColumn 22 102 202 $A_C $E $G $J_C
$Iy_C $Iz_C 1 -mass $mCol ;
element elasticBeamColumn 23 103 203 $A_C $E $G $J_C
$Iy_C $Iz_C 1 -mass $mCol ;
element elasticBeamColumn 24 104 204 $A_C $E $G $J_C
$Iy_C $Iz_C 1 -mass $mCol ;

```

```
# Beams
```

```

set mass [expr ($DL+(0.2*$LL))*($L/2.0)] ;
set mSt1 [expr $mass+($A_B*$RSteel)] ;
set mSt2 [expr $A_B*$RSteel] ;

```

```
# First Floor
```

```

#
# Tag iNode jNode Area E G J Iy
Iz Trans Mass
element elasticBeamColumn 101 101 102 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt2 ;
element elasticBeamColumn 102 102 103 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt1 ;
element elasticBeamColumn 103 103 104 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt2 ;
element elasticBeamColumn 104 104 101 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt1 ;

```

```

# Second Floor
element elasticBeamColumn 201 201 202 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt2 ;
element elasticBeamColumn 202 202 203 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt1 ;
element elasticBeamColumn 203 203 204 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt2 ;
element elasticBeamColumn 204 204 201 $A_B $E $G $J_B
$Iy_B $Iz_B 2 -mass $mSt1 ;

puts "End of Define Model Geometry"

```

For define column mass we consider a distributed mass, proportional to its section area. But for calculating beam mass, we consider mass of the floor and add it to elements mass.

Modal Analysis:

In this step we form an Eigen analysis to get principal frequency and period of the frame. As nature of the frame is unchanged by changing the rotation angle, it can use to test model geometry. In next lines we calculate Eigen values of the model:

```

# Define Eigen Parameters
set a [eigen 3] ;
set W12 [lindex $a 0] ;
set W22 [lindex $a 1] ;
set W1 [expr pow($W12,0.5)] ;
set W2 [expr pow($W22,0.5)] ;
set T1 [expr 2.0*$Pi/$W1] ;
set T2 [expr 2.0*$Pi/$W2] ;

puts "*****"
puts "W1=$W1 Rad/Sec"
puts "T1=$T1 Sec"
puts "*****"
puts "W2=$W2 Rad/Sec"
puts "T2=$T2 Sec"

```

Gravity Loading:

In this step we define distributed gravity loads on beams.

```

# Define Loads
set WzFloor [expr ($DL+$LL)*($L/2.0)*$g] ;

pattern Plain 1 Linear { ;
eleLoad -ele 102 104 202 204 -type -beamUniform 0.0 -$WzFloor
} ;

```

Static Analysis:

After defining gravity loads we have to perform a static analysis to apply gravitational loads to the model. This analysis procedure is as follow:

```

# Static Analysis

```

```
wipeAnalysis ;
constraints Transformation ;
numberer RCM ;
system SparseGeneral ;
test EnergyIncr 1e-7 25 0 ;
algorithm ModifiedNewton ;
integrator LoadControl 0.01 ;
analysis Static ;
analyze 100 ;
```

```
puts "End of Static Analysis"
```

To ensure that static analysis have been performed successfully we set a sentence to appear after analysis by *puts* command.

Define Recorders:

Now we define required recorders as follow:

```
# Define Recorders
recorder Node -file $FileName/Base1.txt -node 1 -dof 1 reaction;
recorder Node -file $FileName/Base2.txt -node 2 -dof 1 reaction;
recorder Node -file $FileName/Base3.txt -node 3 -dof 1 reaction;
recorder Node -file $FileName/Base4.txt -node 4 -dof 1 reaction;

recorder Node -file $FileName/Disp201.txt -node 201 -dof 1 disp;
recorder Node -file $FileName/Disp202.txt -node 202 -dof 1 disp;
recorder Node -file $FileName/Disp203.txt -node 203 -dof 1 disp;
recorder Node -file $FileName/Disp204.txt -node 204 -dof 1 disp;
```

Seismic Loading:

In this step we define an acceleration to apply through a transient analysis to the model.

```
# Define Acceleration
set dtIn 0.02;
set accelSeries "Series -dt $dtIn -filePath accel.txt -factor $g ";
```

```
#
pattern UniformExcitation 2 1 -accel $accelSeries ;
```

Acceleration file with name accel.txt is next to the model and we make it to a time series. Then by UniformExcitation command we define direction of the earthquake to be global X direction. You see we define X direction, but as Alpha changes at the beginning of the model, direction of the earthquake will change.

Time History Analysis:

Now we perform a transient analysis to apply seismic load to the frame.

```
# Time History Analysis
set NInput 2686;
set ndt 2 ;
set dtAnalysis [expr $dtIn/$ndt] ;
```



```

set NAnalysis [expr $NInput*$ndt] ;

wipeAnalysis ;
constraints Transformation ;
numberer RCM ;
system SparseGeneral ;
test EnergyIncr 1e-7 25 0 ;
algorithm ModifiedNewton ;
integrator Newmark 0.5 0.25 ;
analysis Transient ;
analyze $NAnalysis $dtAnalysis ;

puts "End of Time History Analysis"

```

In the above commands parameter NInput is number of acceleration points in the accel.txt file, ndt is a number to divide steps to reduce length of analysis steps, dtAnalysis is the analysis time step and NAnalysis is the number of analysis steps.

RUN OpenSEES:

Now modeling is finished and we can RUN the model. After RUN model a window is opened as follow:

```

C:\Windows\system32\cmd.exe
Pacific Earthquake Engineering Research Center -- 2.4.6 (rev 6058)
(c) Copyright 1999-2013 The Regents of the University of California
All Rights Reserved
(Copyright and Disclaimer @ http://www.berkeley.edu/OpenSees/copyright.html)

End of Define Model Geometry
*****
M1=15.996905950839368 Rad/Sec
T1=0.3925759155738791 Sec
*****
M2=16.66343602221038 Rad/Sec
T2=0.3768730526152731 Sec
End of Static Analysis
End of Time History Analysis
-----
End of script <E:\OpenSEES_IR\Book\Asli\sorushnia\Final\14\Codes\OpenSEES_Book_E\
12.tcl> reached. Press any key to continue

```

Fig 7: RUN window

Results:

We change model direction from 0.0 to 90.0 degrees and compare results, such as max node displacement and max column shear between this rotation angles. Fig 8 and 9 presents this results:

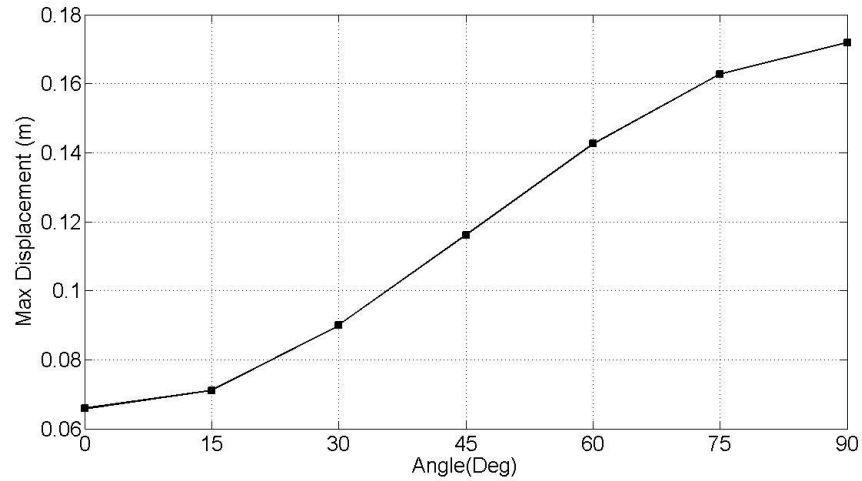


Fig 8: Maximum Node Displacement (Node 201)

As you can see maximum displacement increased by increase of rotation angle. In case of 90 degrees, seismic load applies to the weak direction of the frame and its max displacement is the largest displacement through other cases. In case of 0.0 degree, seismic load acts on strength direction of the frame and its maximum displacement is the minimum value through other cases.

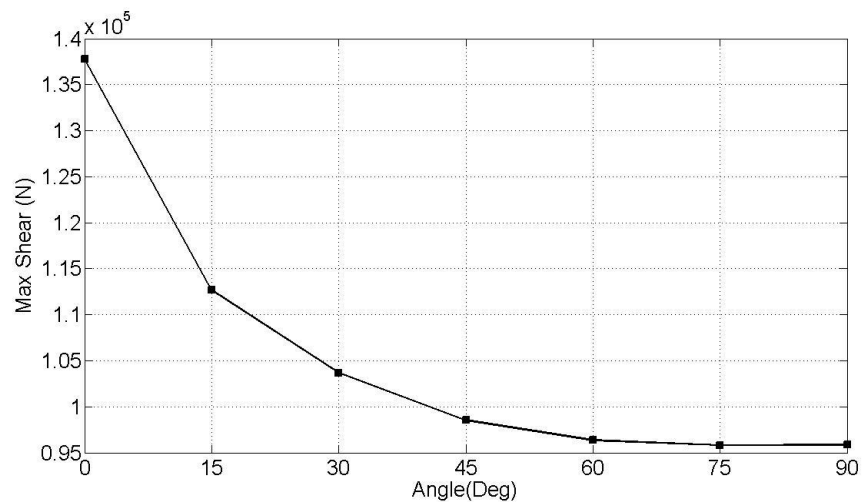


Fig 9: Maximum Column Shear (Column 11)

Fig 9 compare column shear between rotation angle cases from 0.0 to 90.0 degrees. As it is shown maximum shear of the column decrease by increase of rotation angle.