OpenSees Navigator

Tony Yang

Assistant Professor

Department of Civil Engineering

University of British Columbia, Vancouver, Canada

Andreas Schellenberg

Project Engineer

Rutherford & Chekene Structural and Geotechnical Engineers

Eiji Kohama

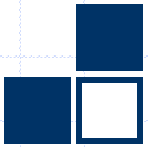
Head of Group

Earthquake and Structural Dynamics Group

Port and Airport Research Institute

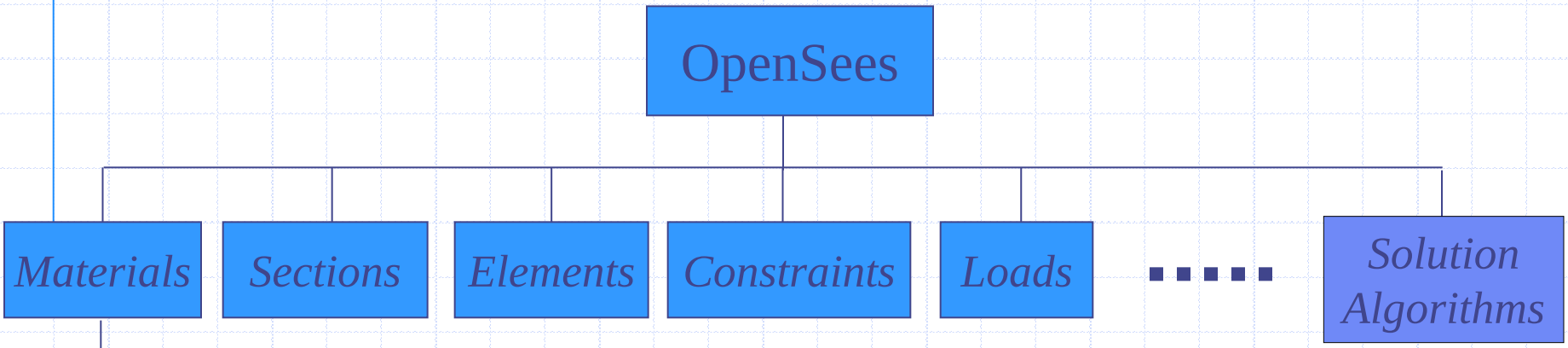
What is OpenSees?

- **O**pen **S**ystem for **E**arthquake **E**ngineering **S**imulation.
- A software *framework* for developing sequential and parallel, grid-enabled finite element applications in earthquake engineering.
- It is open source. Hence everybody can download and contribute to the code.
- Mostly used as a research tool, but it is fairly stable for regular use.



OpenSees

- A robust finite element software.

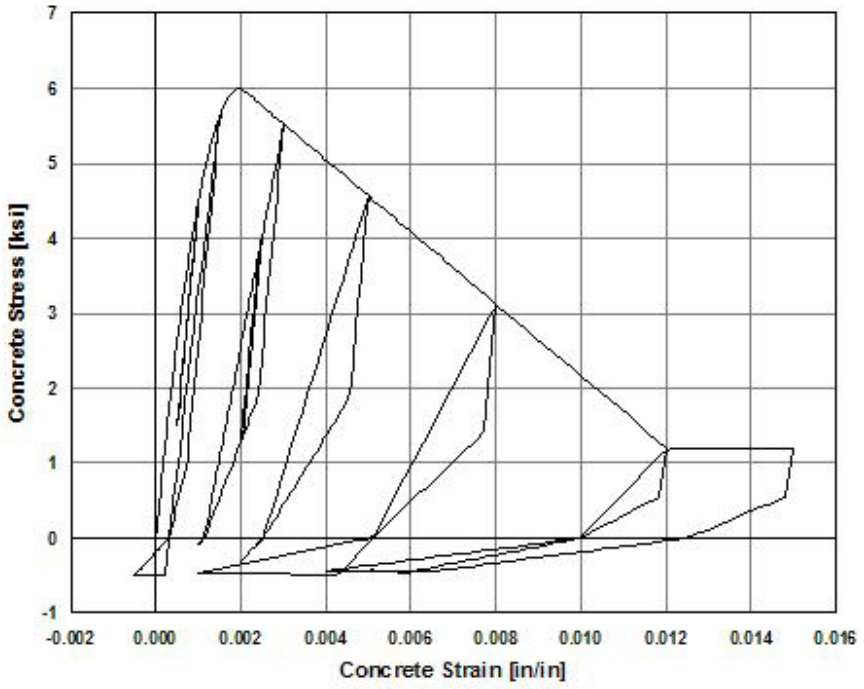


Uniaxial material

-Elastic, -BoucWen, -Steel02, -Concrete02, -Gap, -Fatigue, -Hardening, -Hysteretic, -Viscous, -MinMax, -Parallel, -Series, ...

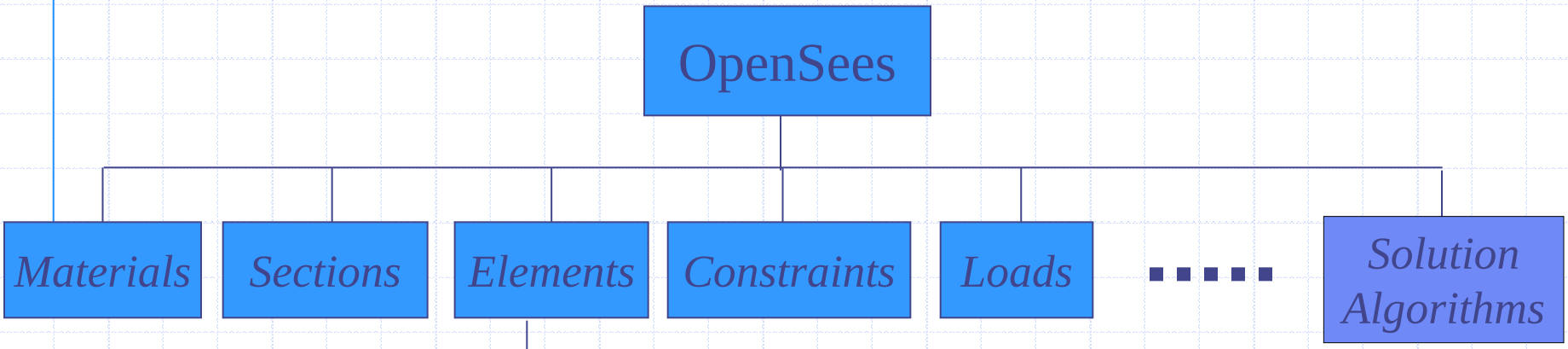
nD material

-PlaneStress
-PressureDependMultiYield
-Template3DElastoPlastic,...



OpenSees

- A robust finite element software.



Line element

-ZeroLength, -Truss, -ElasticBeamColumn,
-NonlinearBeamColumn (force, displacement),
-BeamWithHinges,...

Area element

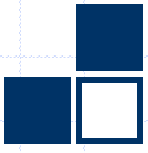
-Quad (std, bbar, enhanced, u-p),
-Shell, -BbarQuad,...

Solid element

-Brick (std, bbar, 20node, u-p, u-p-U)

Running OpenSees

- Tcl (tool command language) programming language
  - It is a string based command language.
  - Variables and variable substitution.
  - Expression evaluation.
  - Basic control structures (if , while, for, foreach).
  - Procedures (subroutine).
- Requires some time and learning effort to use it.



TCL scripting models

- Difficult to visualize the model.
- Prone to error.
- Difficult to review the structural response.

```
1.  wipe;
2.  source Units.tcl;                    # define units
3.  source ParamList.tcl;                # load up parameter values (Model information)
4.  source GMFiles.tcl;                  # load up ground-motion filenames
5.  foreach Xframe $iXframe Hcol $iHcol Lcol $iLcol Lbeam $iLbeam
                GIblc $iGIblc GrhoCol $iGrhoCol GPcol $iGPcol GMfact $iGMfact {
6.      source Static.tcl;               # load procedure for static analysis
7.      source Dynamic.tcl;              # load procedure for dynamic analysis
8.      puts FRAME$Xframe.....FRAME$Xframe.....
9.      puts STATIC_ANALYSIS
10.     Static $Xframe $Hcol $Lcol $Lbeam $GIblc $GrhoCol $GPcol $GMfact ;
11.     puts DYNAMIC_ANALYSIS
12.     foreach GroundFile $iGroundFile {
13.         puts GroundMotion$GroundFile
14.         Dynamic $Xframe $Hcol $Lcol $Lbeam $GIblc $GrhoCol $GPcol $GMfact $GroundFile;
15.     }
16. }
```

FRAME

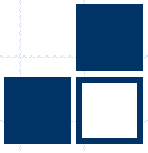
GROUND MOTION

OpenSees Navigator

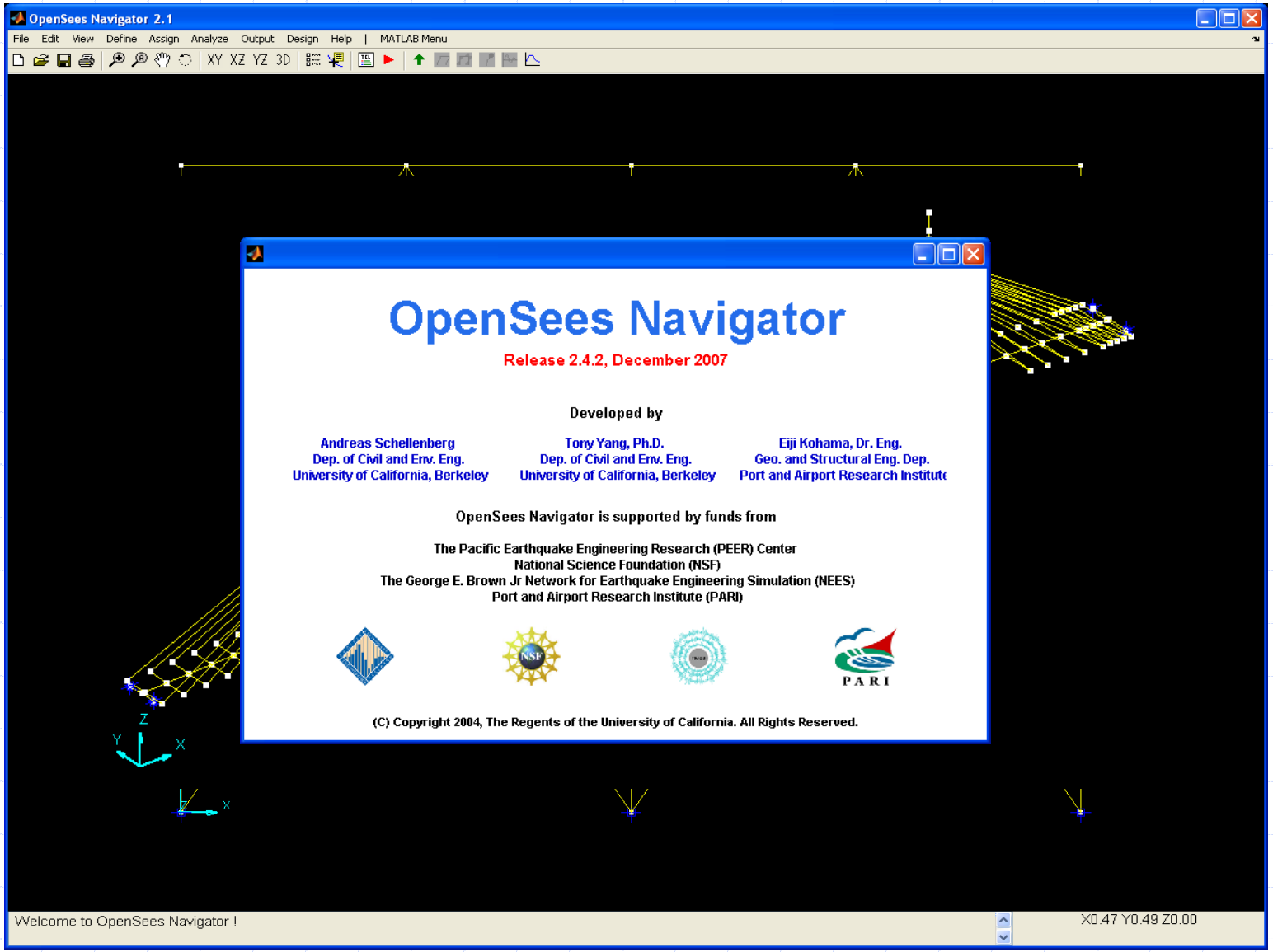
- Graphical user interface (GUI) for OpenSees.
- Pre- and post- processing for OpenSees.
- Design toolboxes: NSP, PBEE, AISC design checks, AISC database, response spectra for linear and bilinear systems, system identification and signal filtering.
- Can be used as a standalone application or coupled with MATLAB for development purpose.
- Being used by researchers from Asia, north and south America and Europe.
  https://www.google.com/analytics/reporting/maps?id=20238794&pdr=20100201-20100303&cmp=average#lts=1267723466859

Motivations

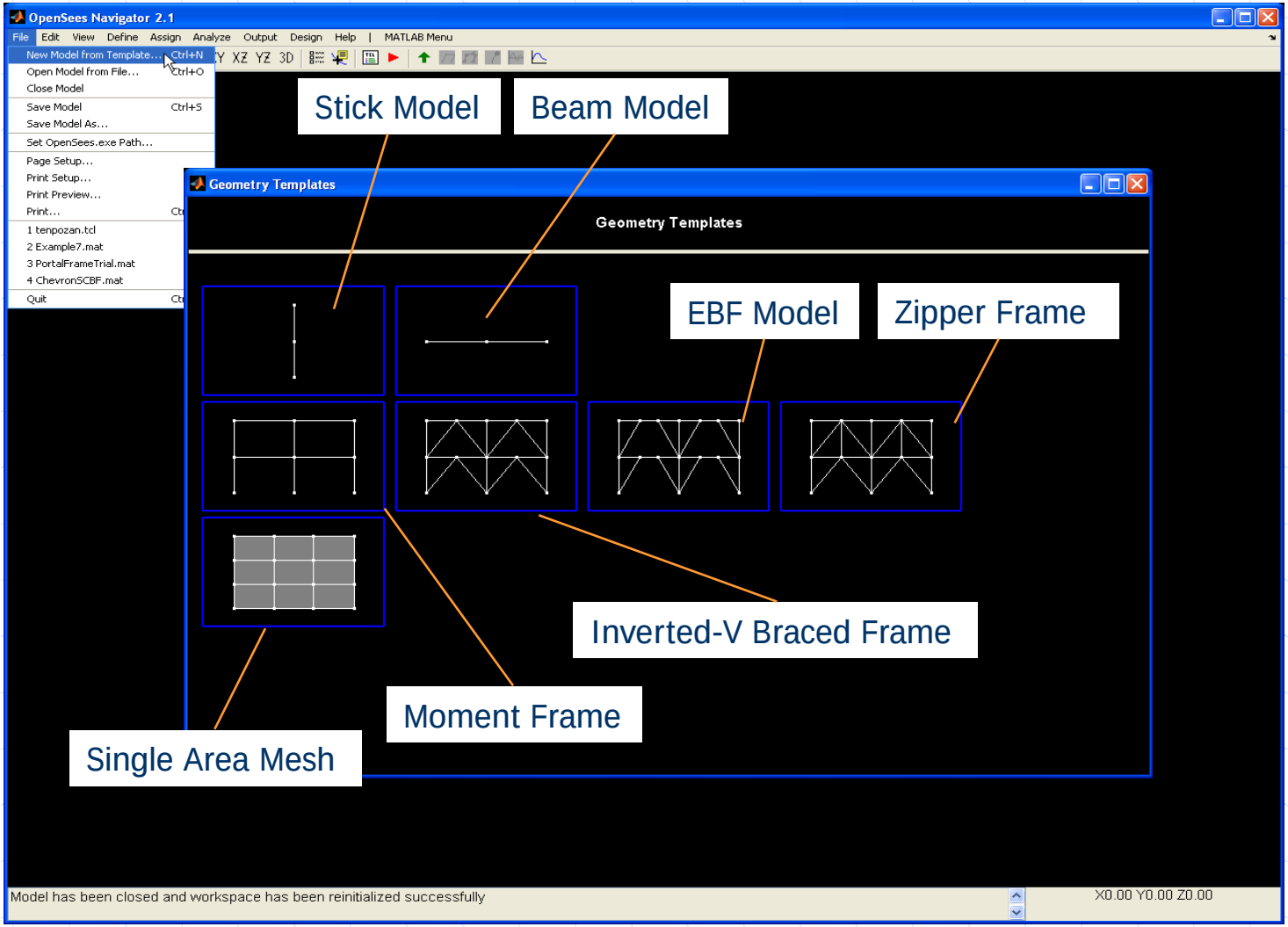
- Replace the TCL text input with graphical input.
- This will provide many robust plotting algorithms and it is very effective in generating the plots for engineering applications.
- OpenSees Navigator will create the OpenSees model and graphically display the results before, during or after a test.
- Flexible to use and requires no programming skill.



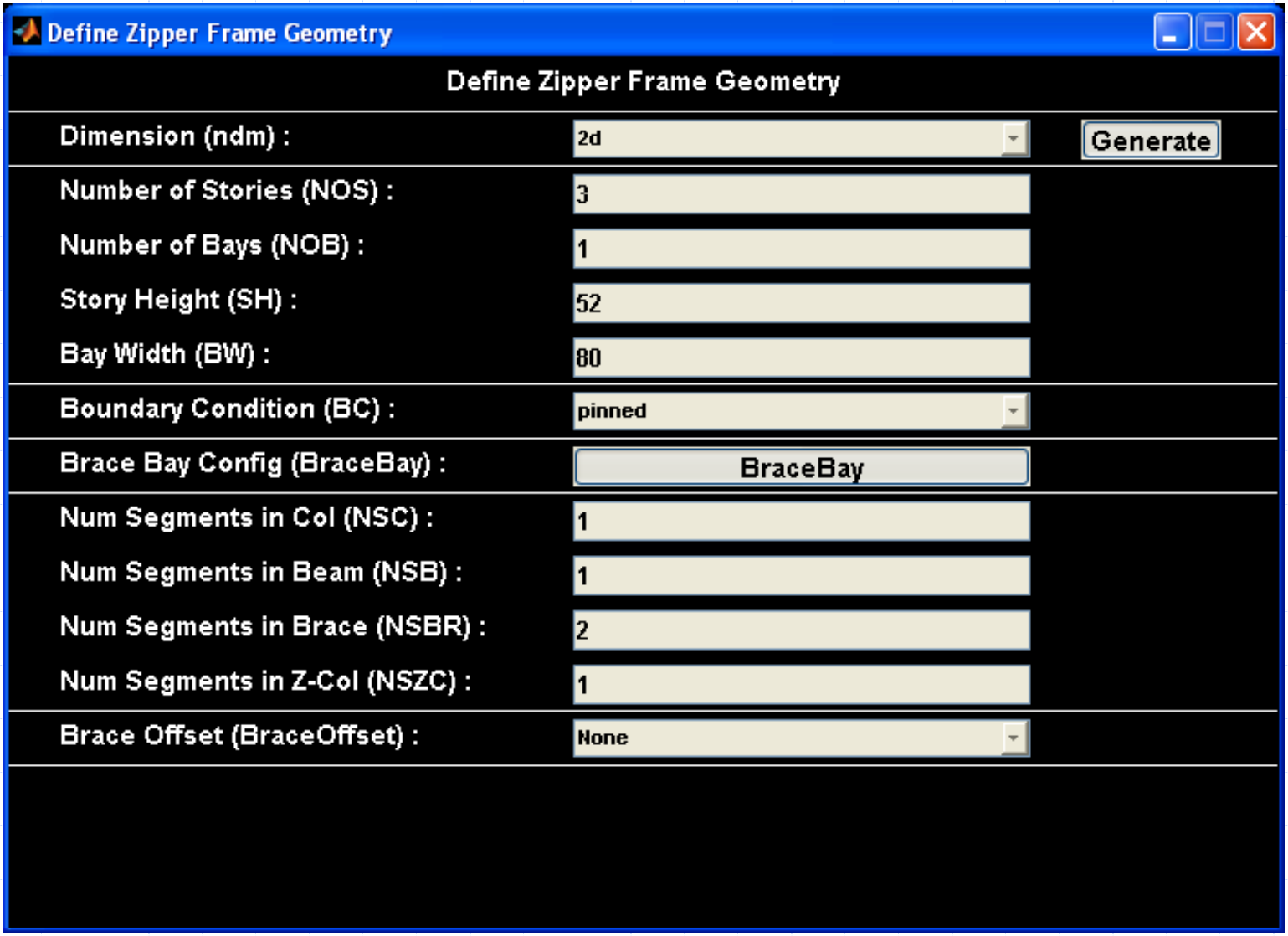
OpenSees Navigator



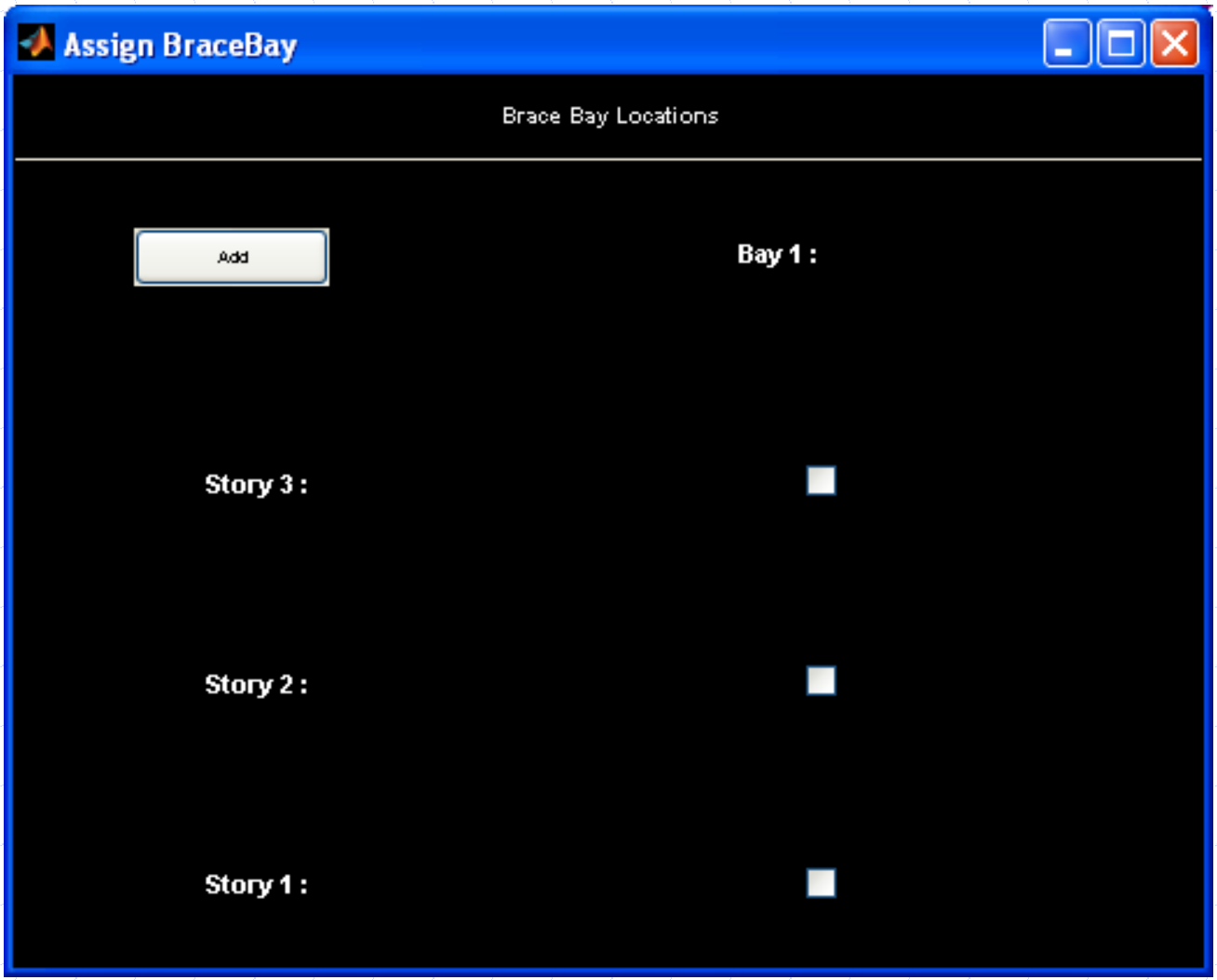
Define geometry: new model template



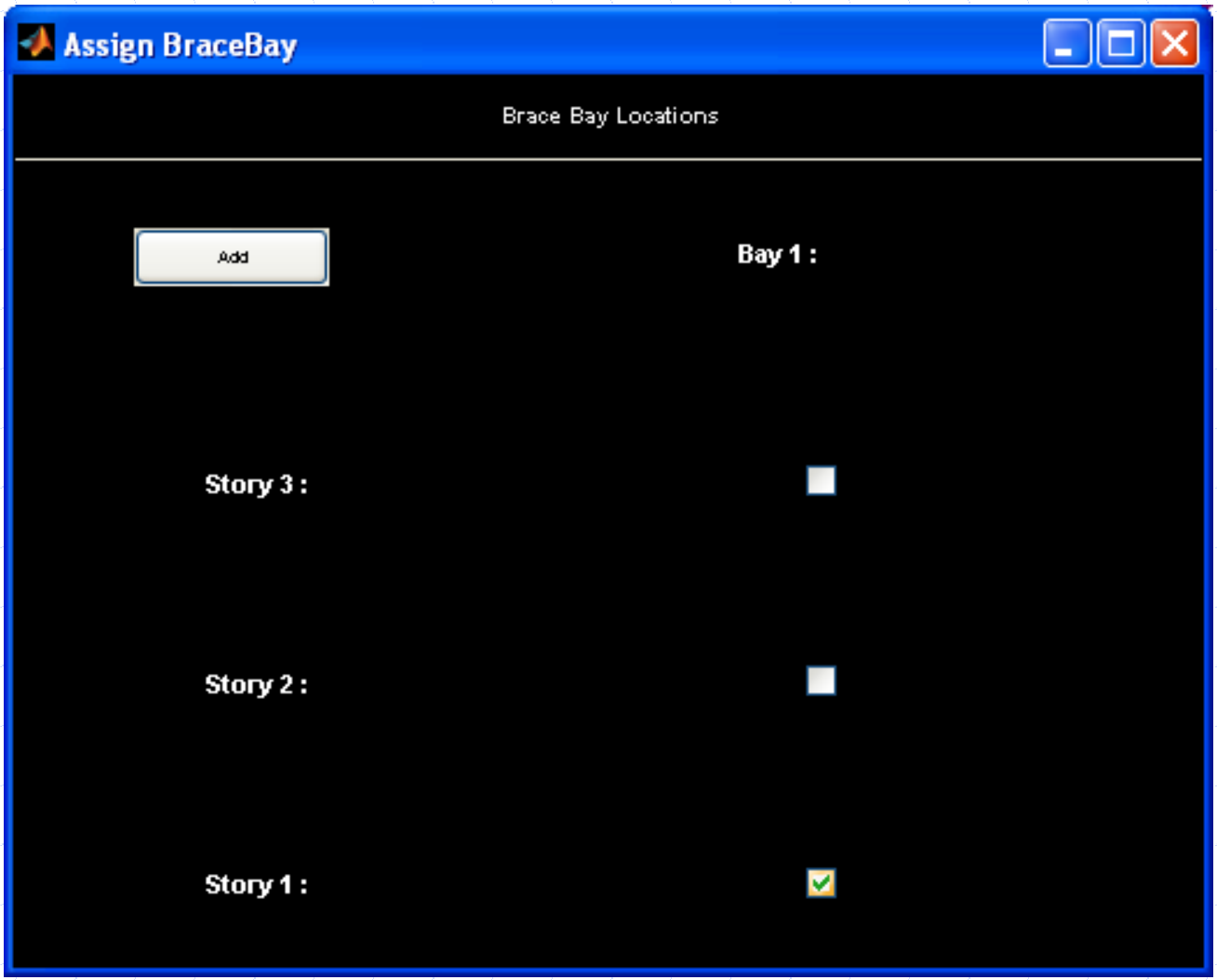
Define geometry: Zipper braced frame



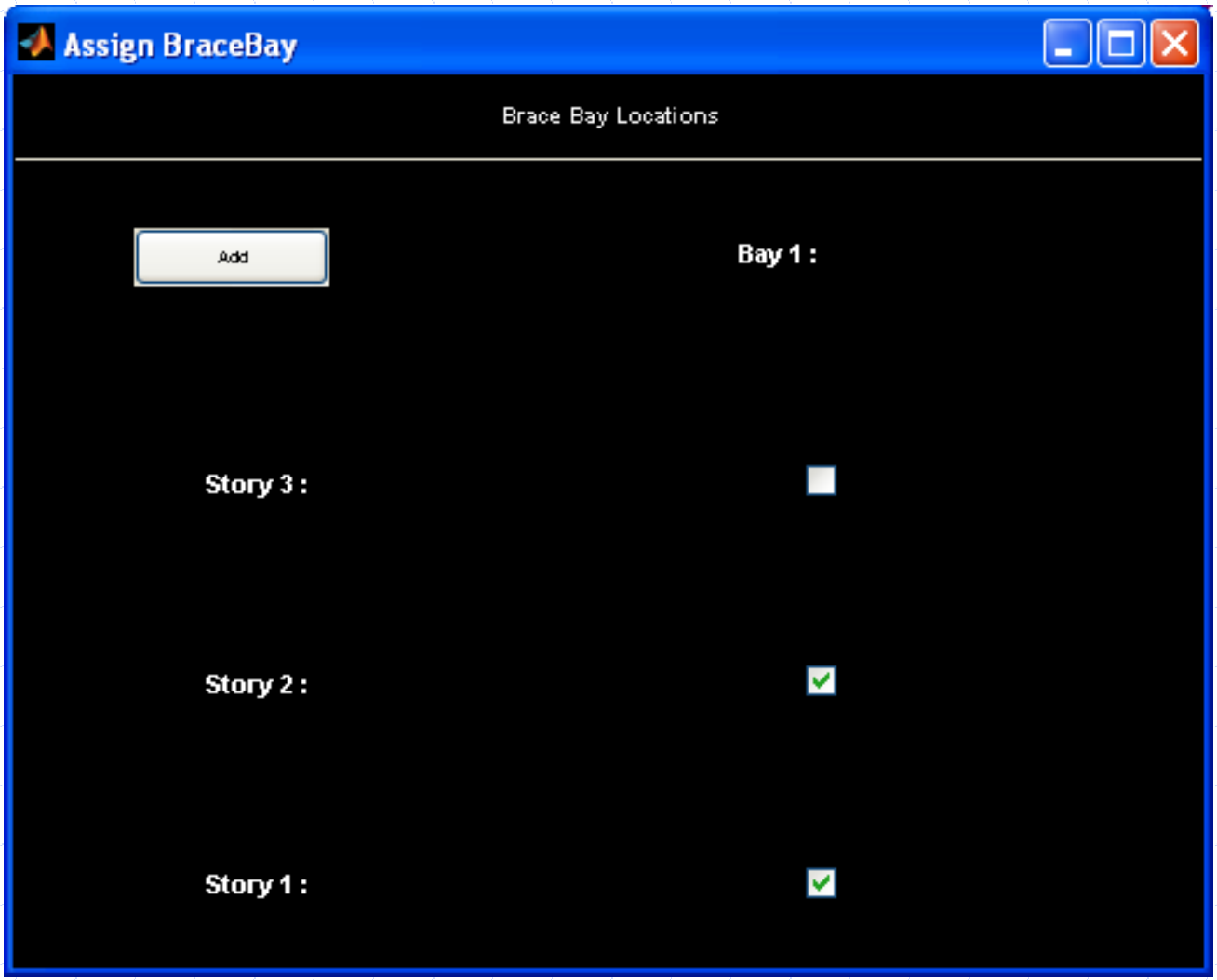
Define geometry: Zipper braced frame



Define geometry: Zipper braced frame

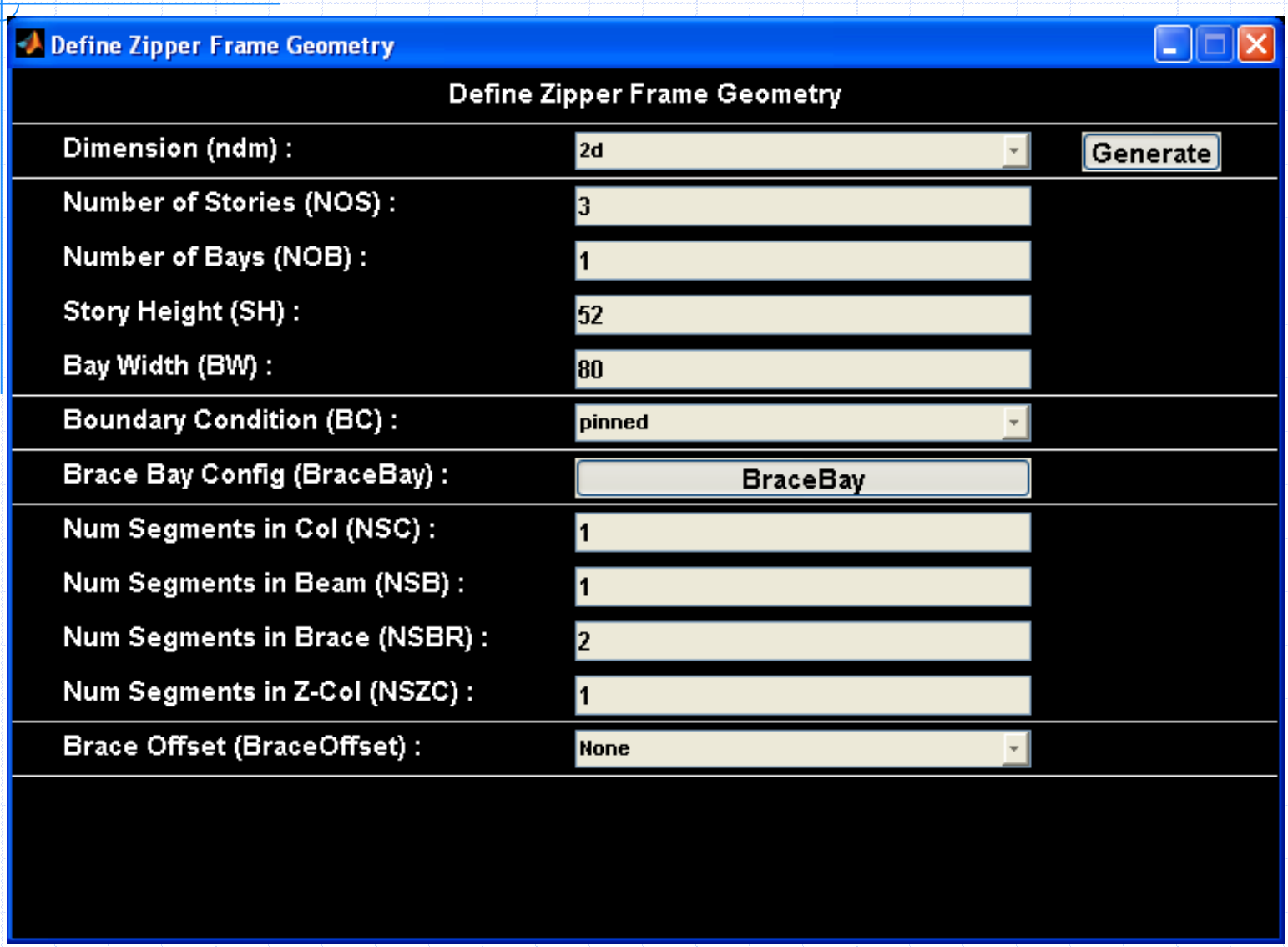


Define geometry: Zipper braced frame

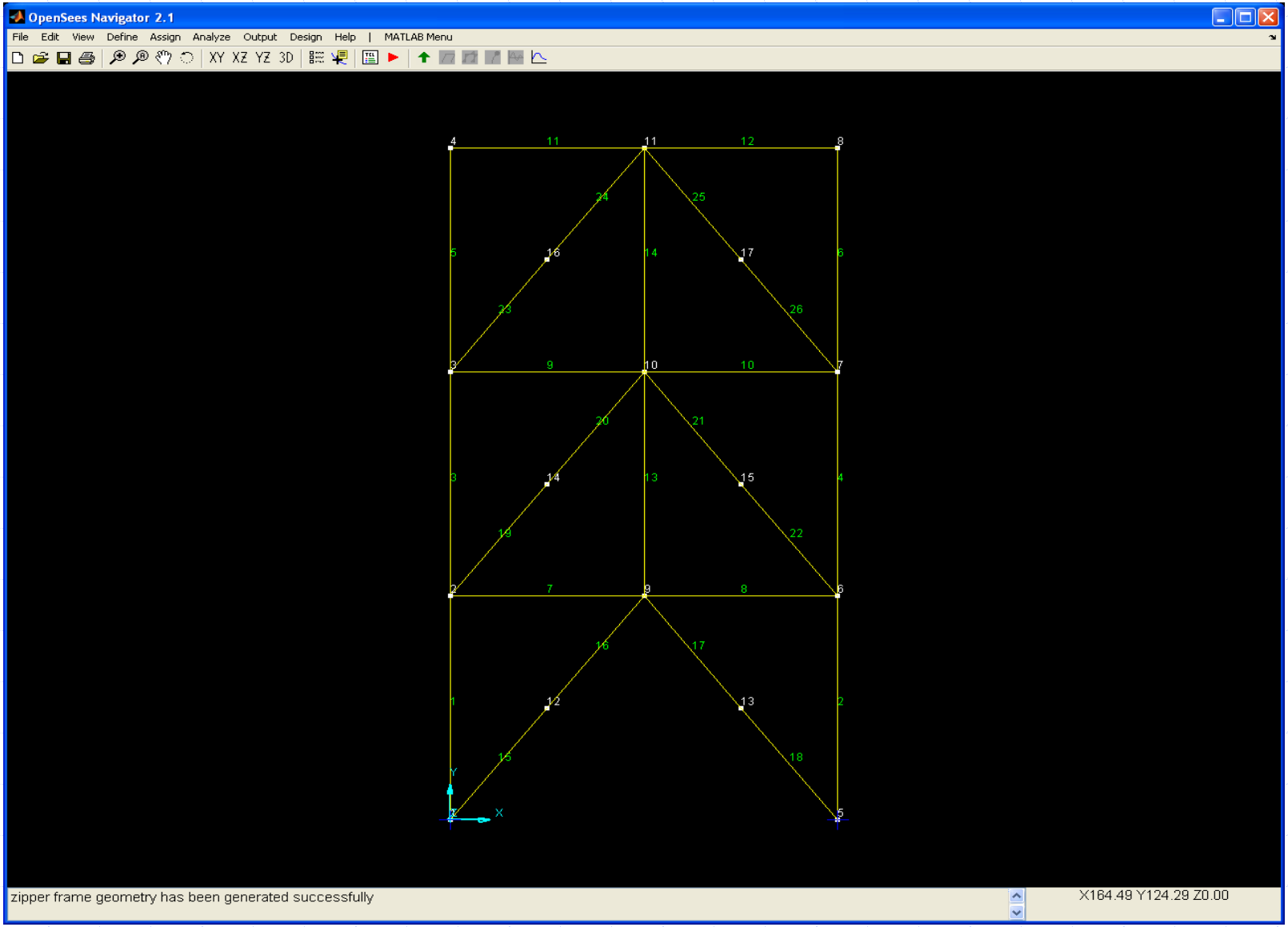


Define geometry: Zipper braced frame

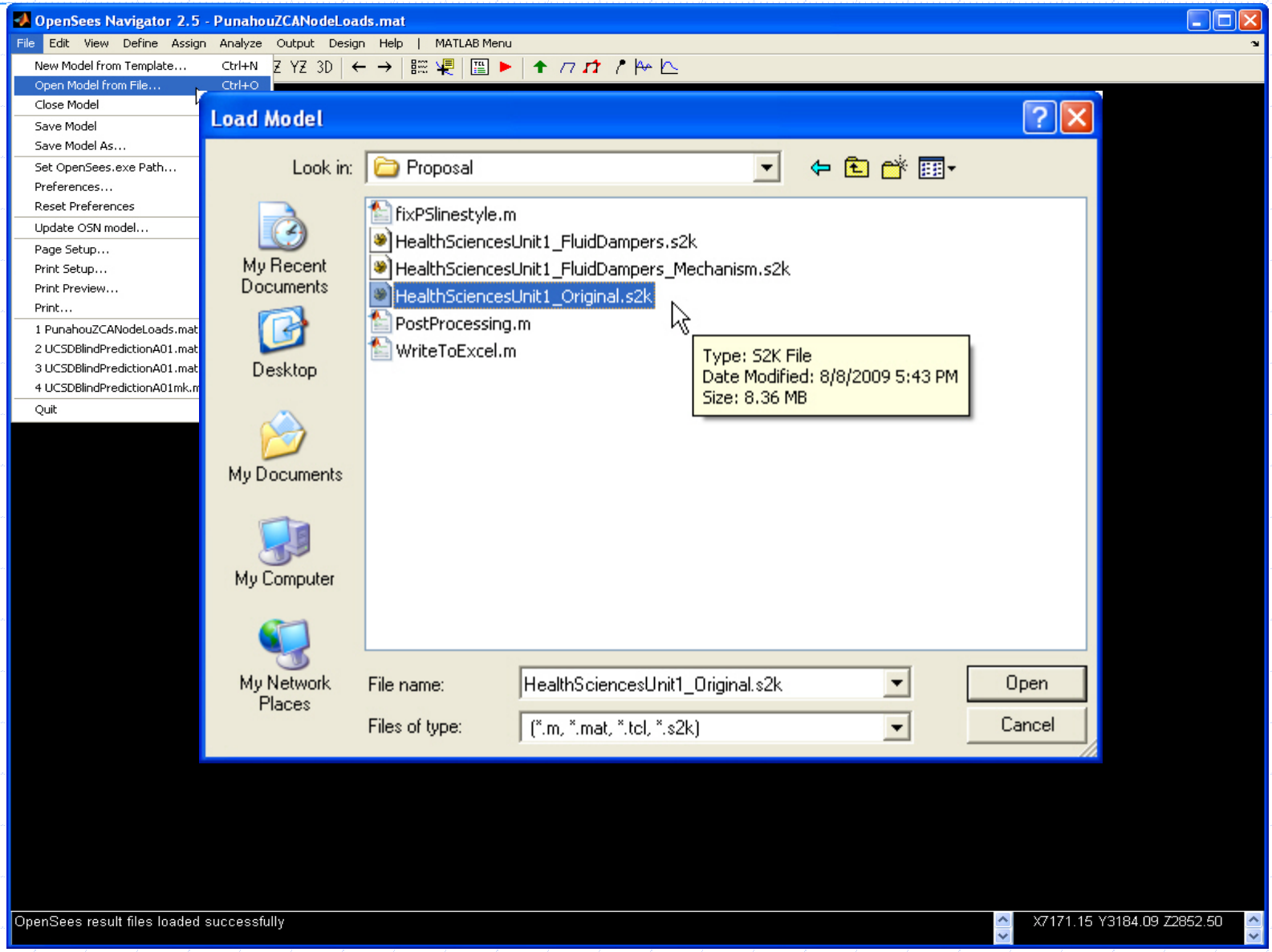
Define geometry: Zipper braced frame



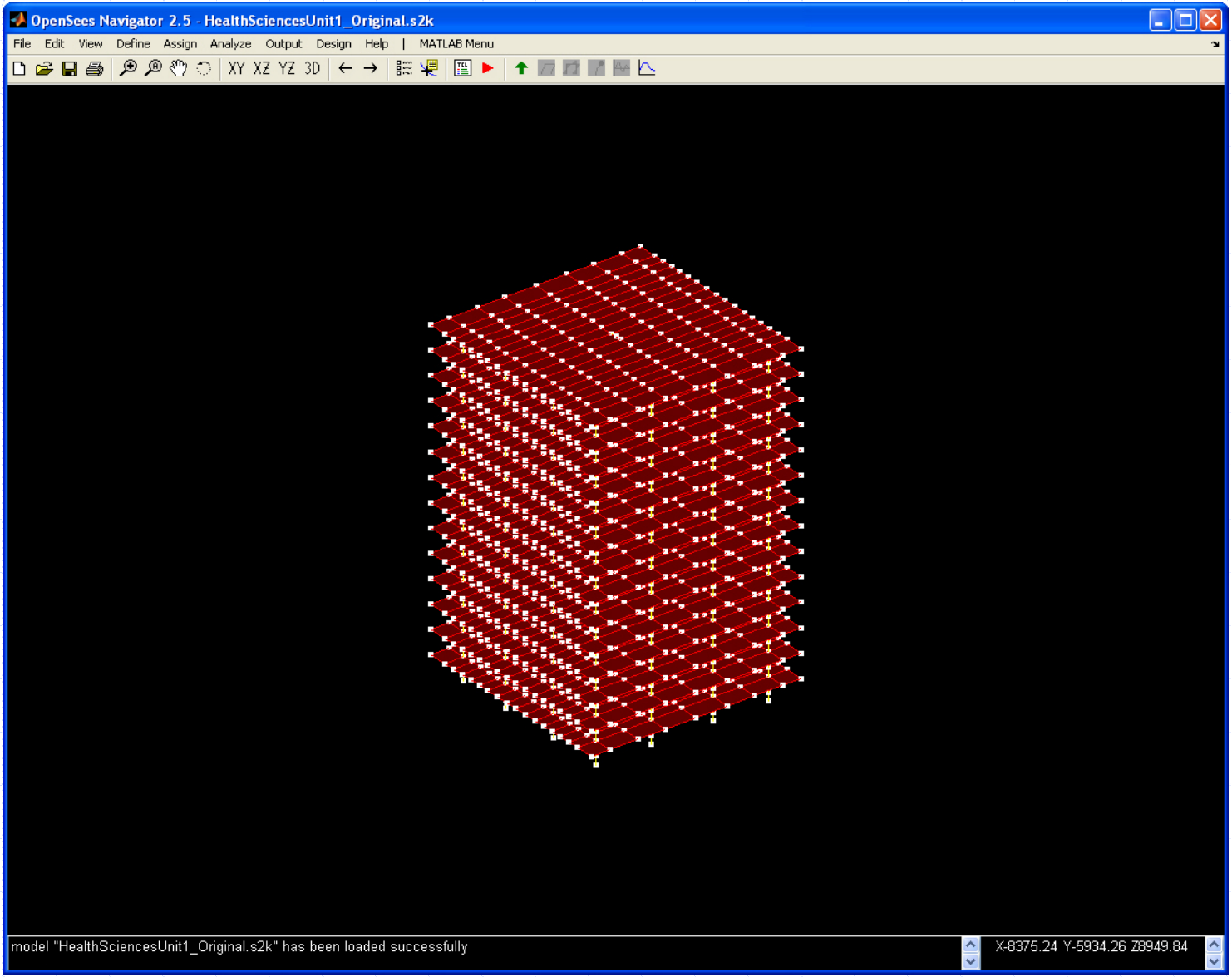
View geometry: display



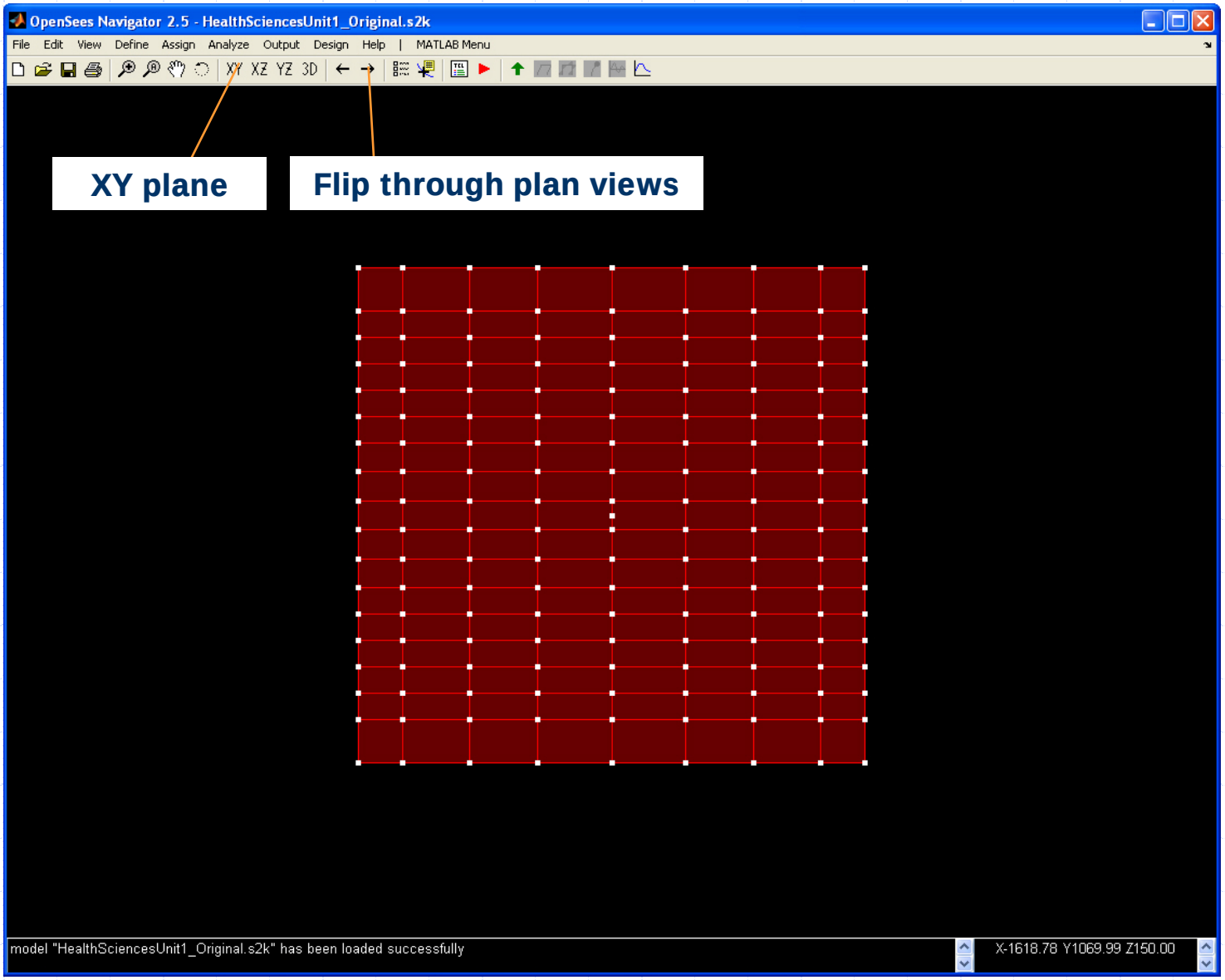
Define geometry: *.m, *.mat, *.tcl, *.s2K



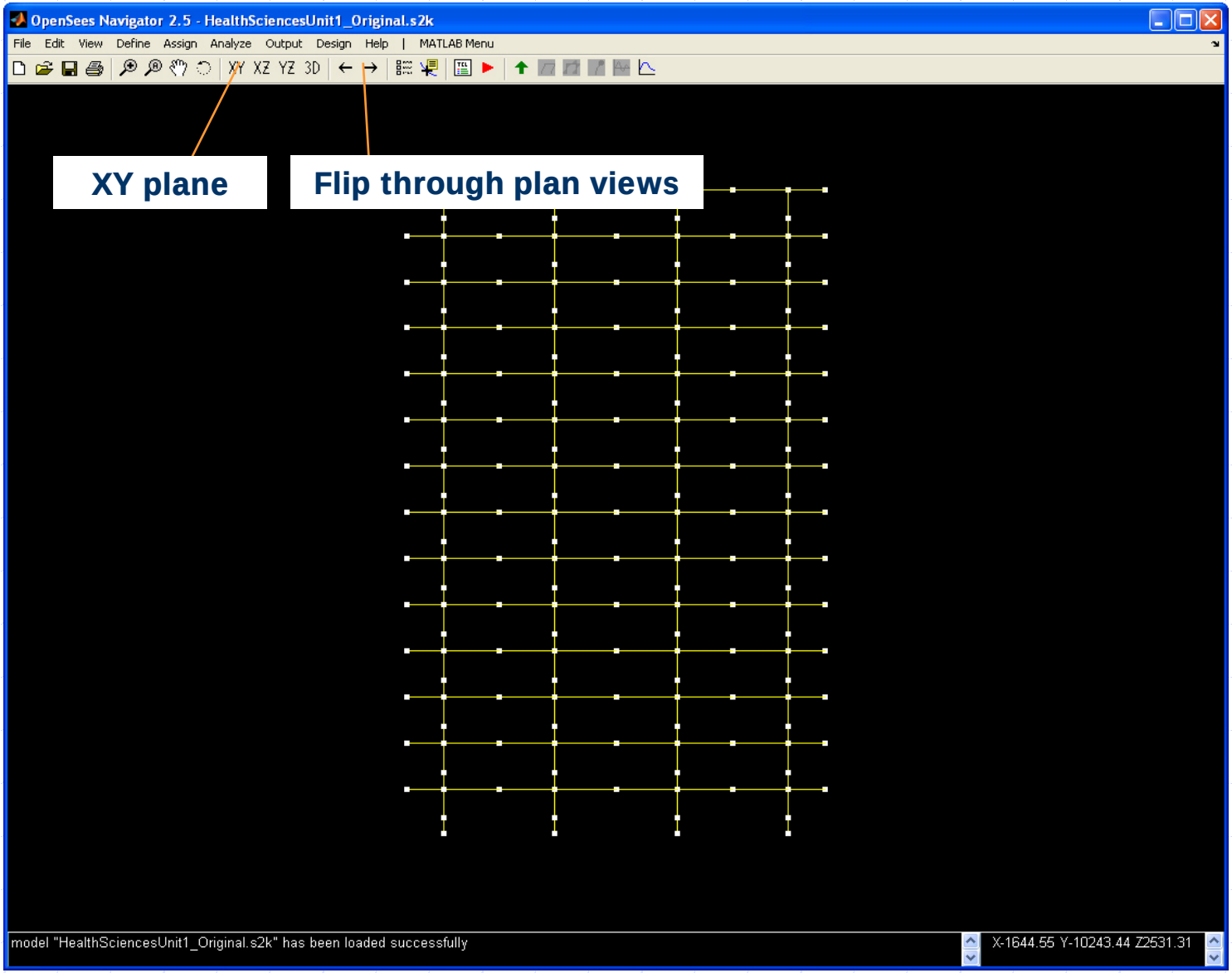
Define geometry: import from SAP2000



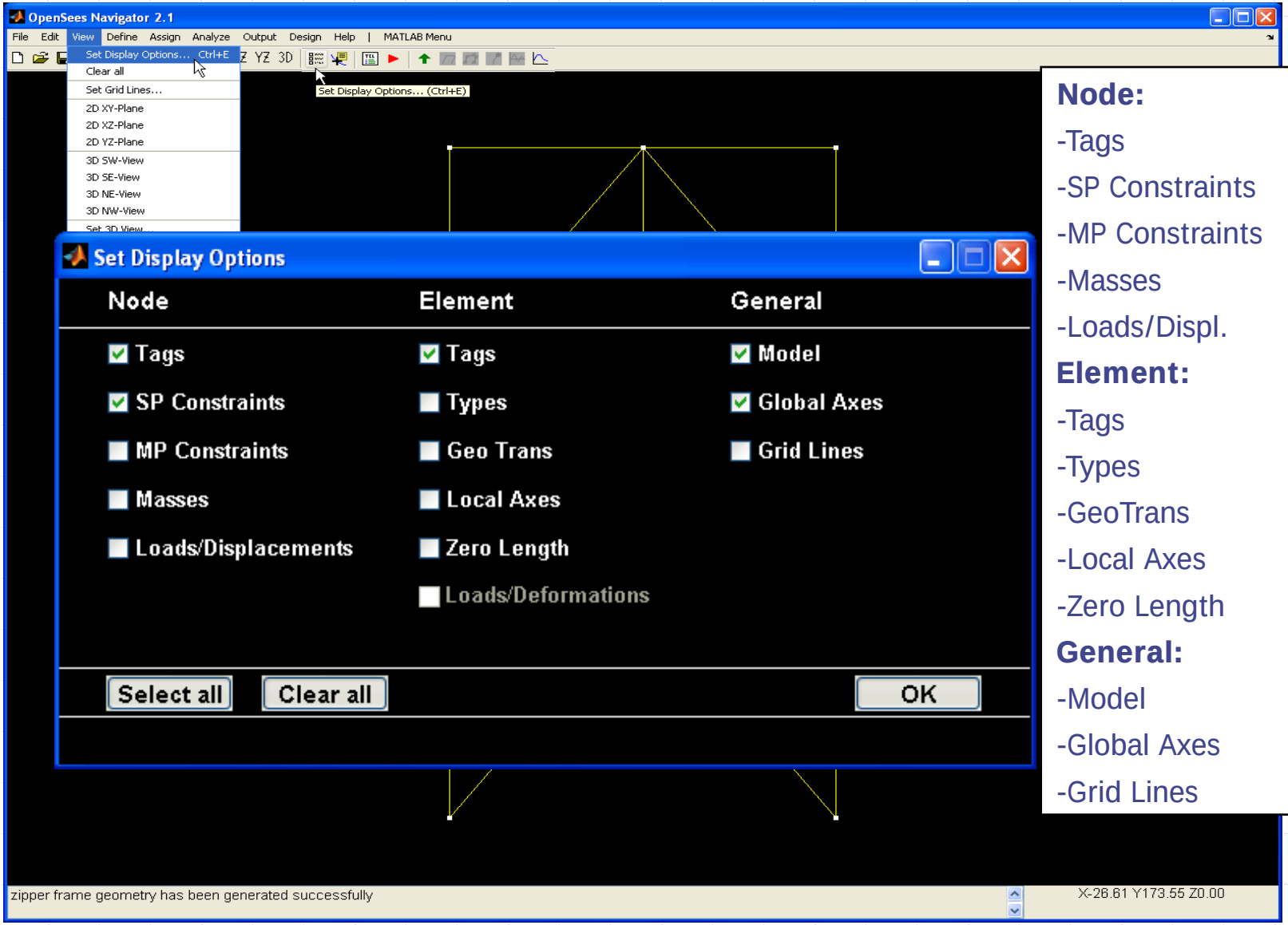
Define geometry: Plan



Define geometry: plan



View geometry: set display options



Node:

-Tags

-SP Constraints

-MP Constraints

-Masses

-Loads/Displ.

Element:

-Tags

-Types

-GeoTrans

-Local Axes

-Zero Length

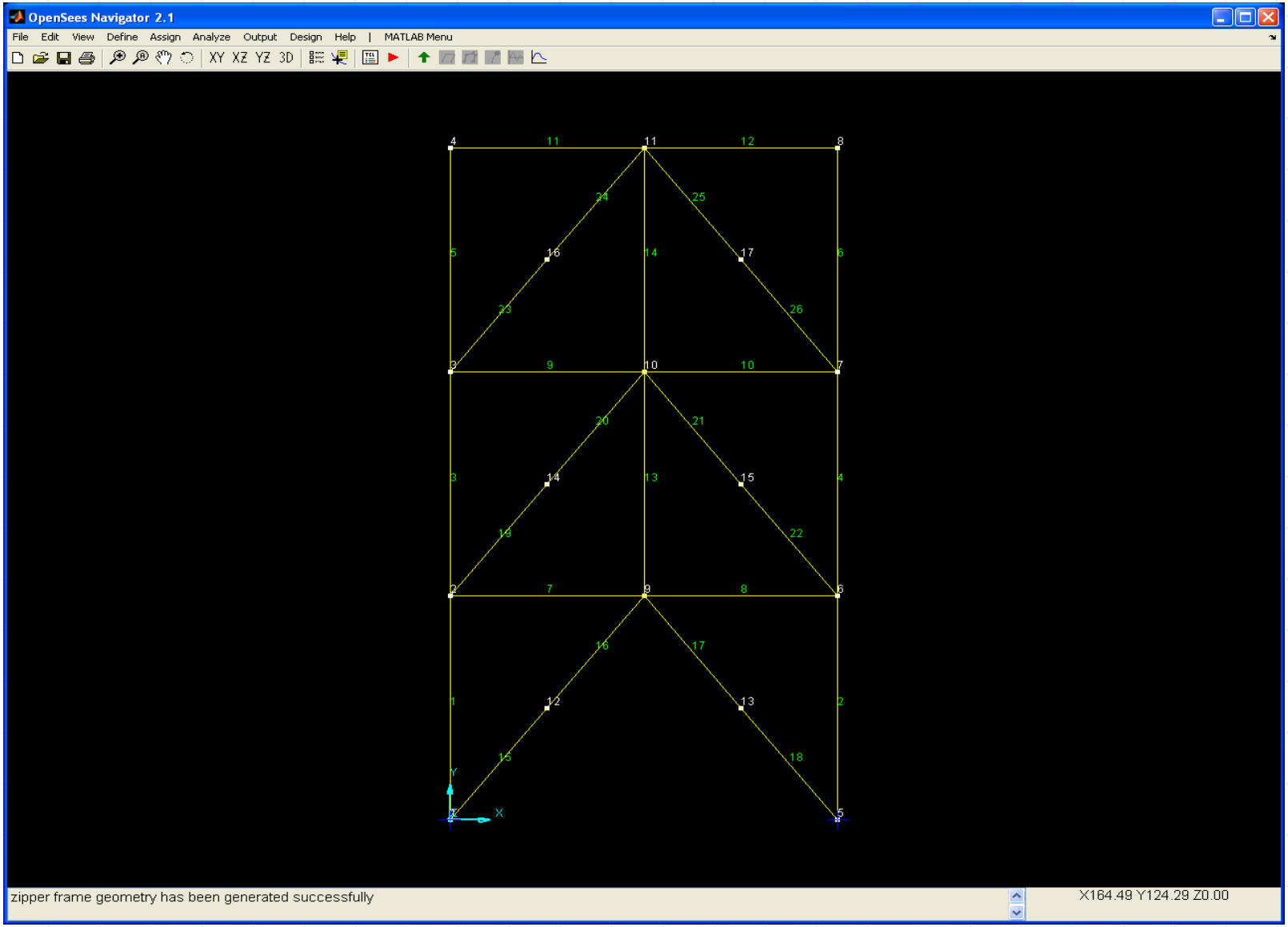
General:

-Model

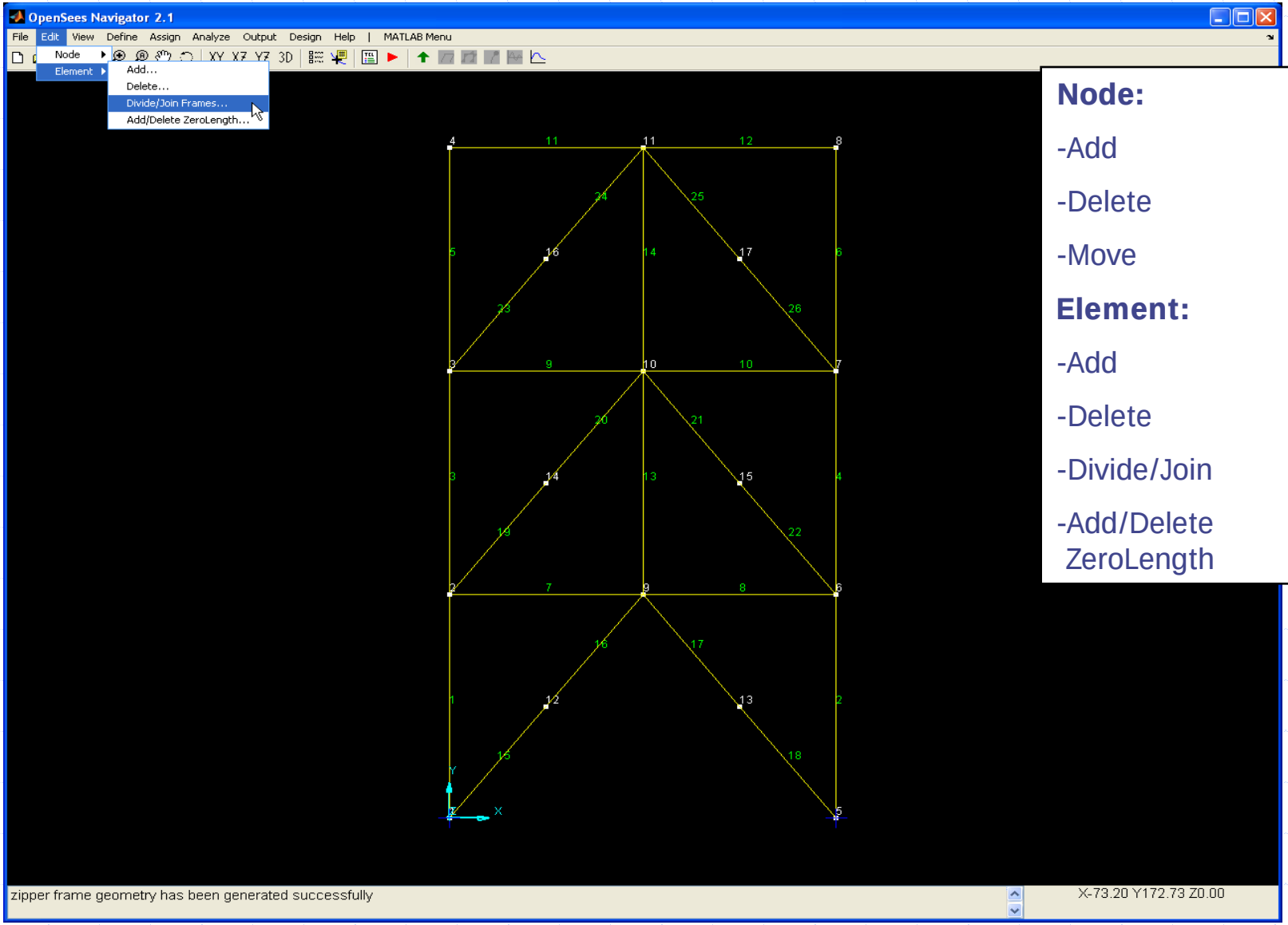
-Global Axes

-Grid Lines

View geometry: display



Edit geometry



Node:

-Add

-Delete

-Move

Element:

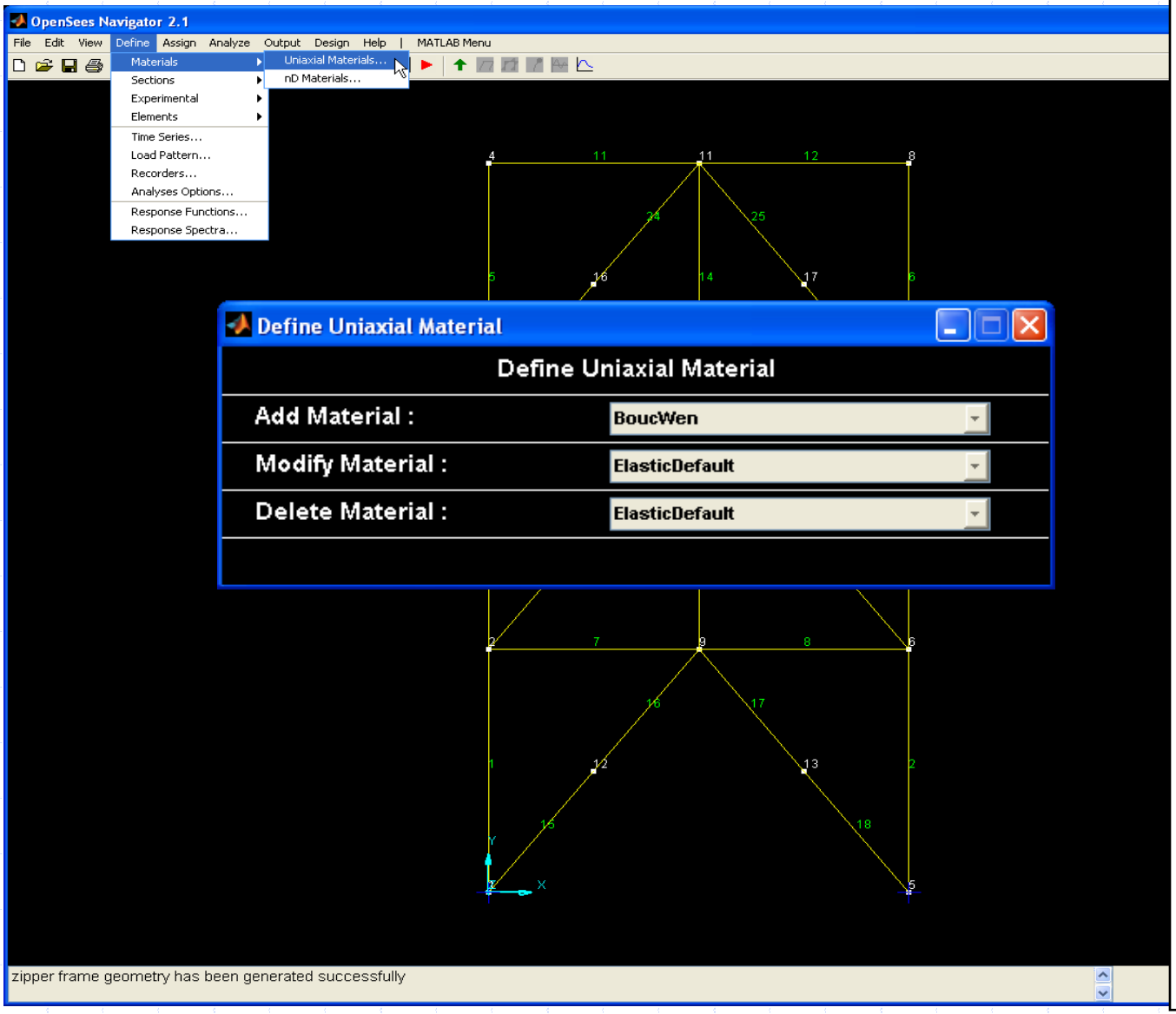
-Add

-Delete

-Divide/Join

-Add/Delete ZeroLength

Define material: uniaxial materials



Templates:

-BoucWen

-Concrete01

-Concrete02

-Concrete03

-Elastic

-ElasticNoTension

-ElasticPP

-ElasticPPGap

-Fatigue

-Hardening

-Hysteretic

-MinMax

-Parallel

-Series

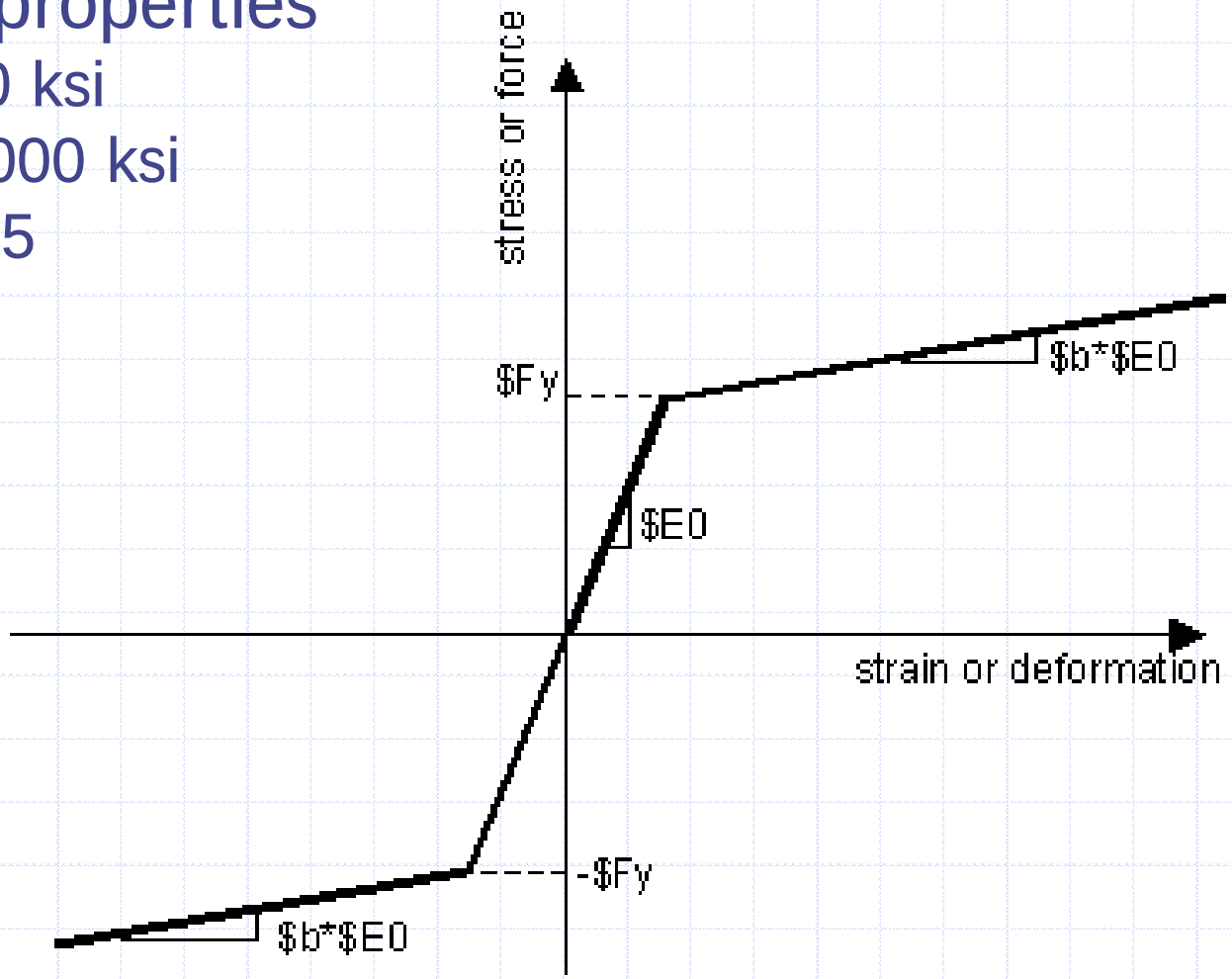
-Steel01

-Steel02

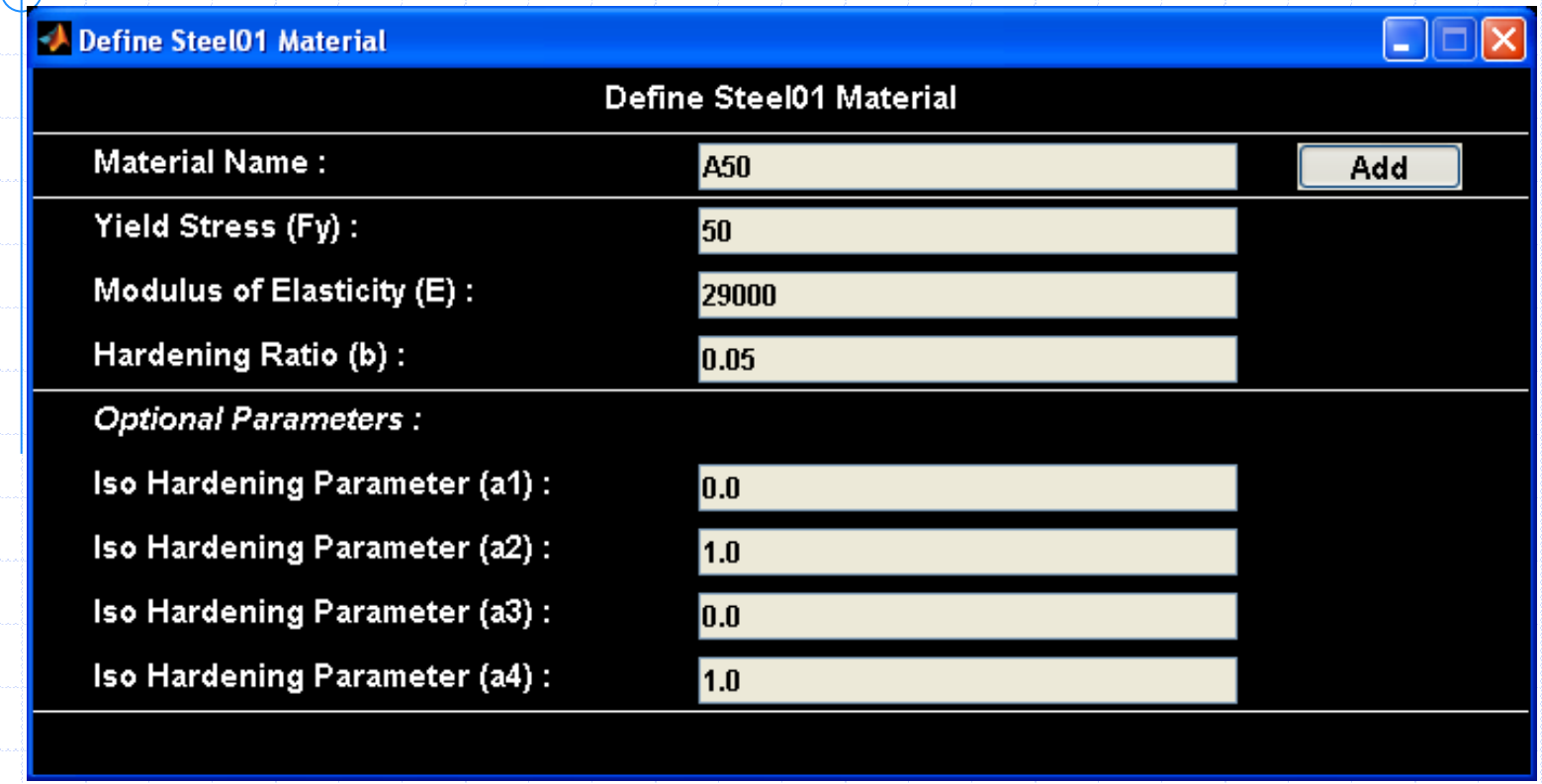
-Viscous

Define uniaxial material: Steel01

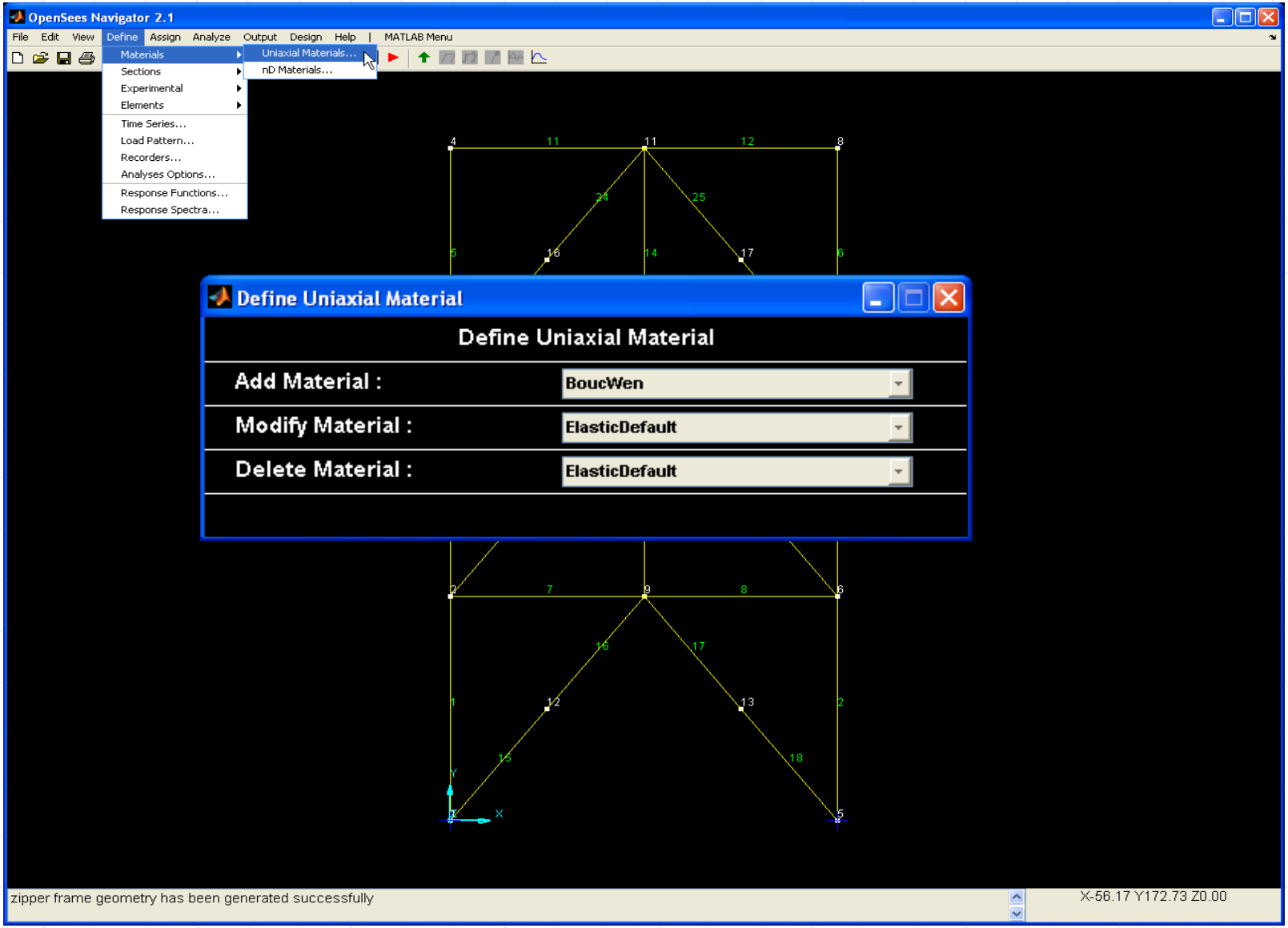
- Material properties
  - Fy = 50 ksi
  - E = 29000 ksi
  - b = 0.05



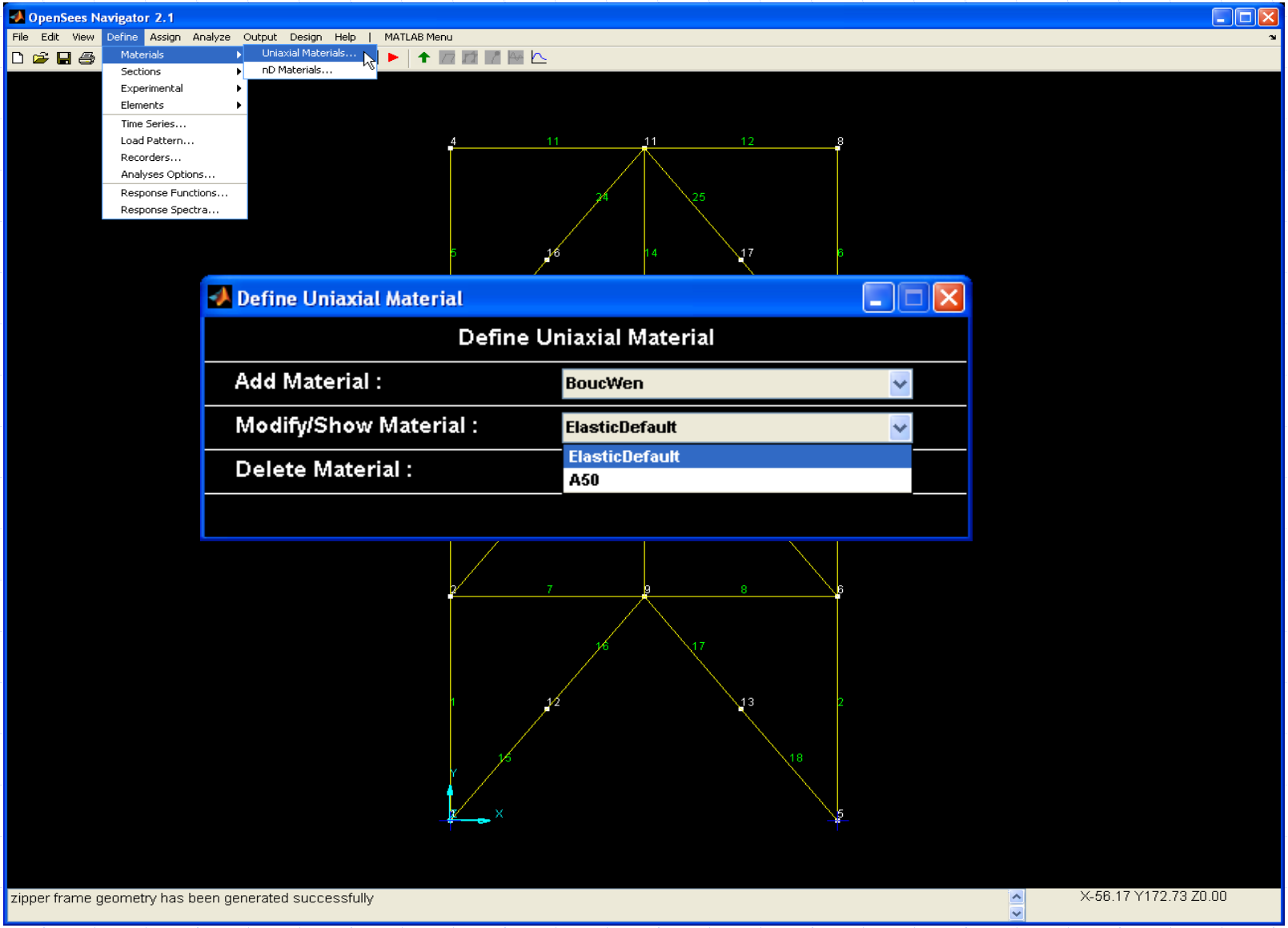
Define uniaxial material: Steel01



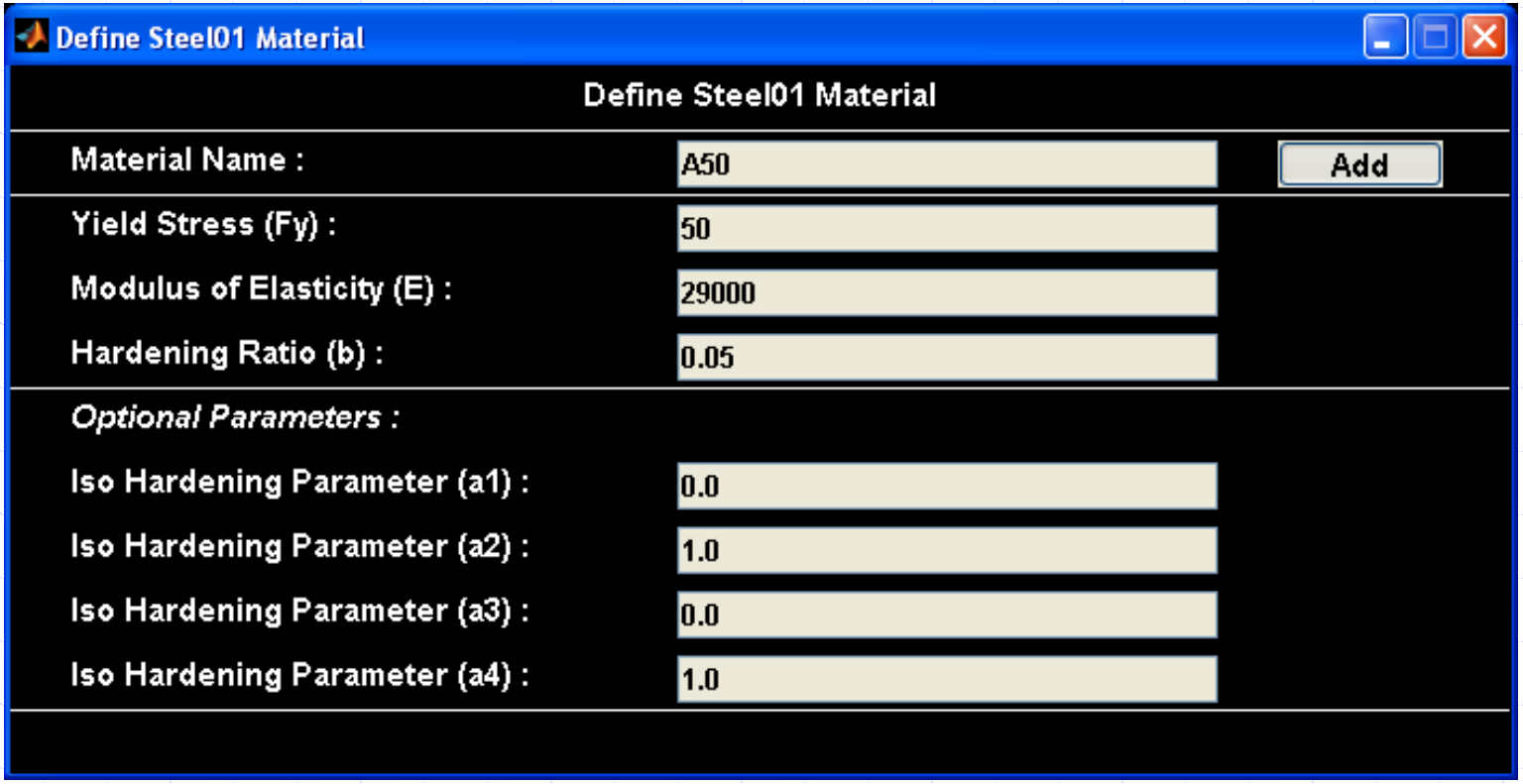
Define material: uniaxial materials



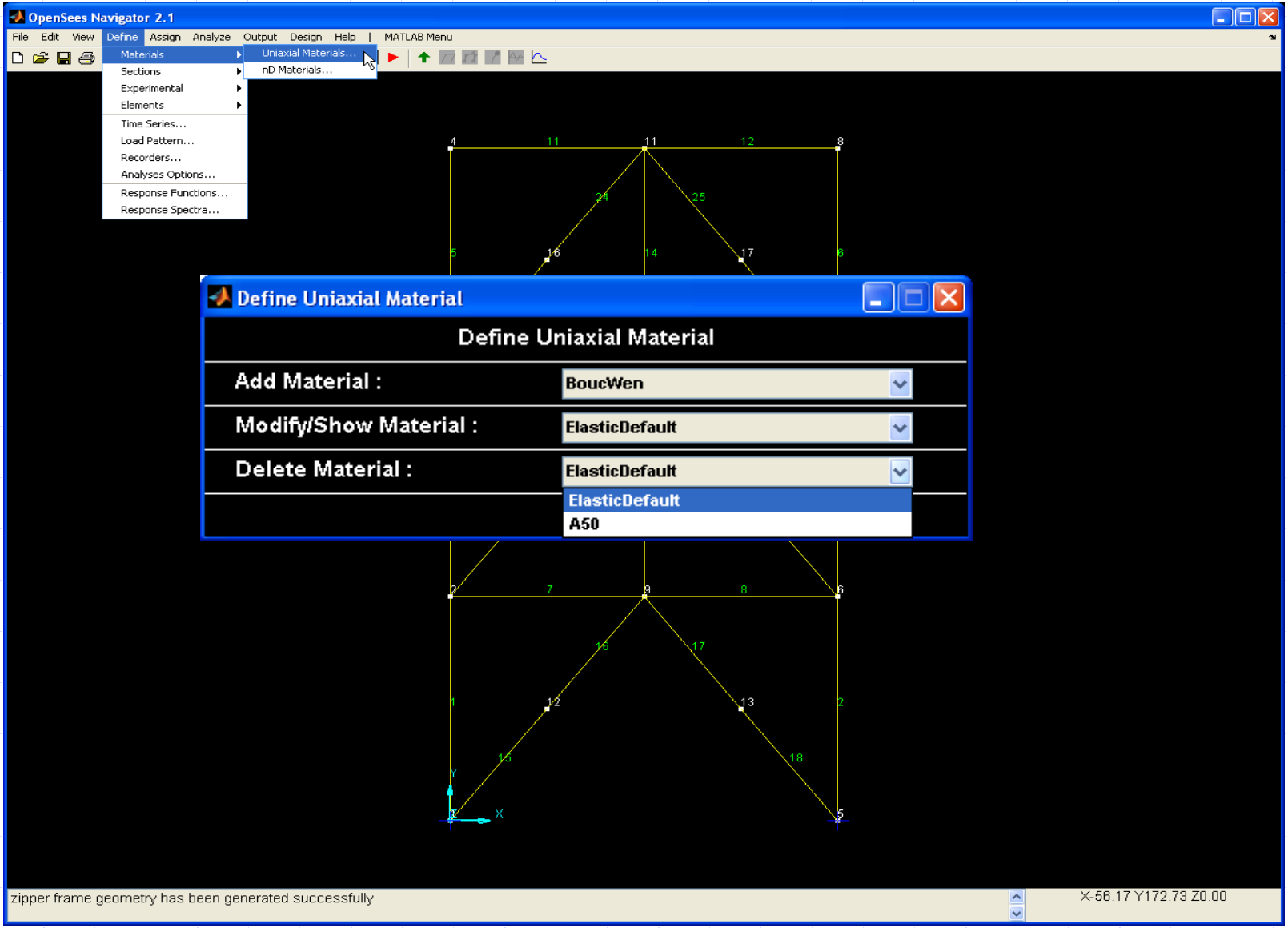
Define material: uniaxial materials



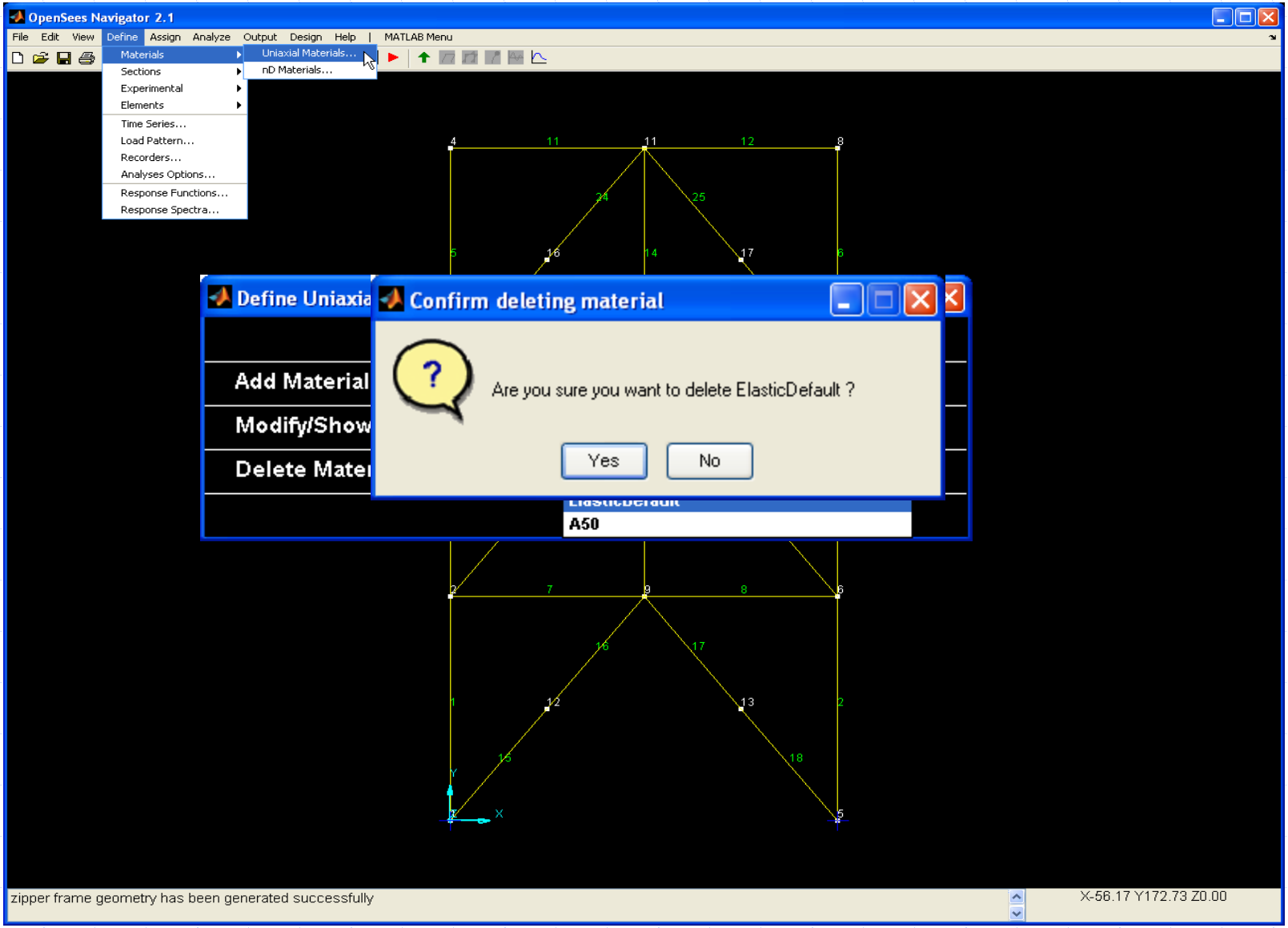
Define uniaxial material: Steel01



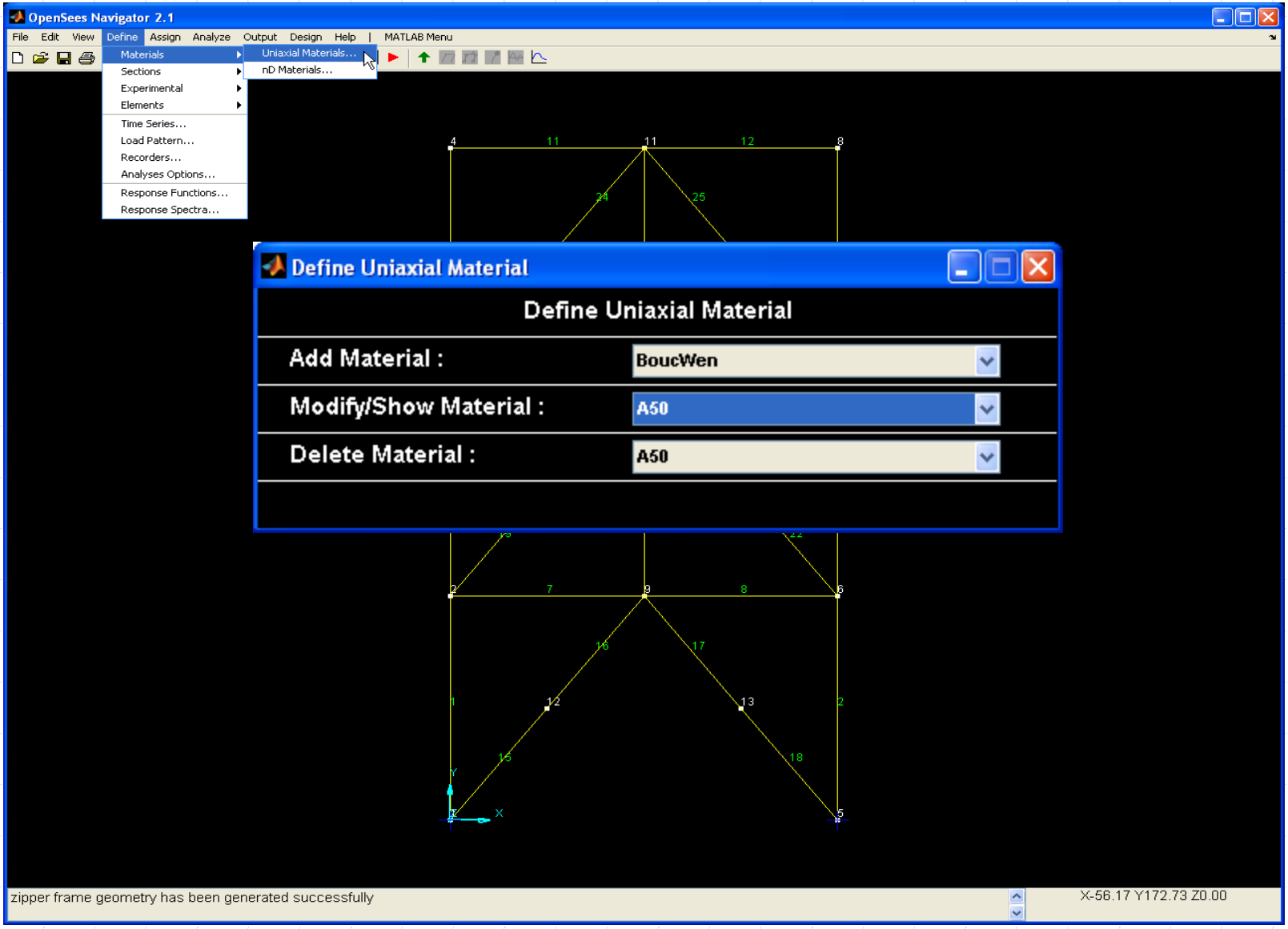
Define material: uniaxial materials



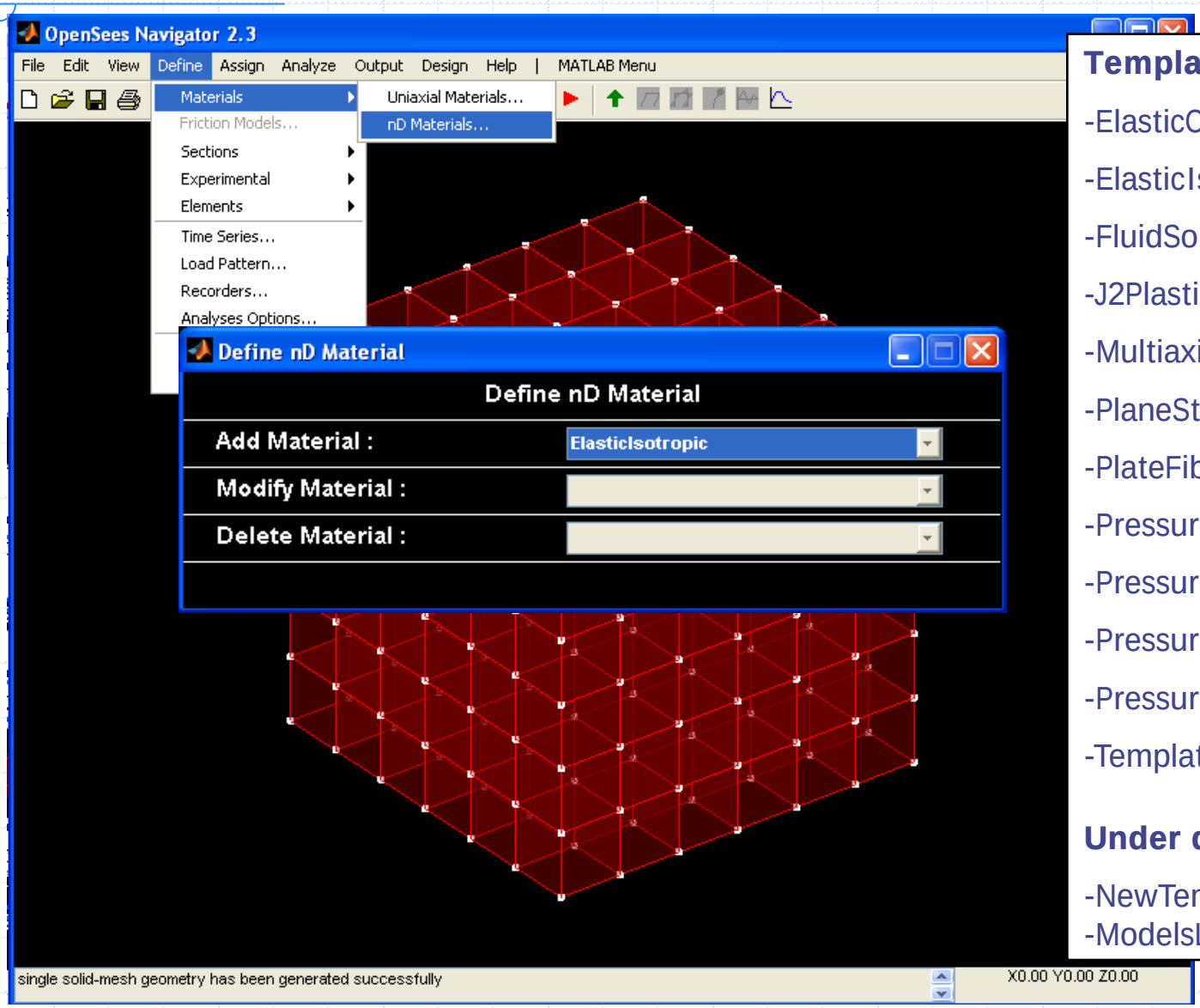
Define material: uniaxial materials



Define material: uniaxial materials



Define material: nD materials



Templates:

-ElasticCrossAnisotropic3D

-ElasticIsotropic

-FluidSolidPorous

-J2Plasticity

-MultiaxialCyclicPlasticity

-PlaneStress

-PlateFiber

-PressureDependMultiYield

-PressureDependMultiYield02

-PressureDependentElastic3D

-PressureIndependMultiYield

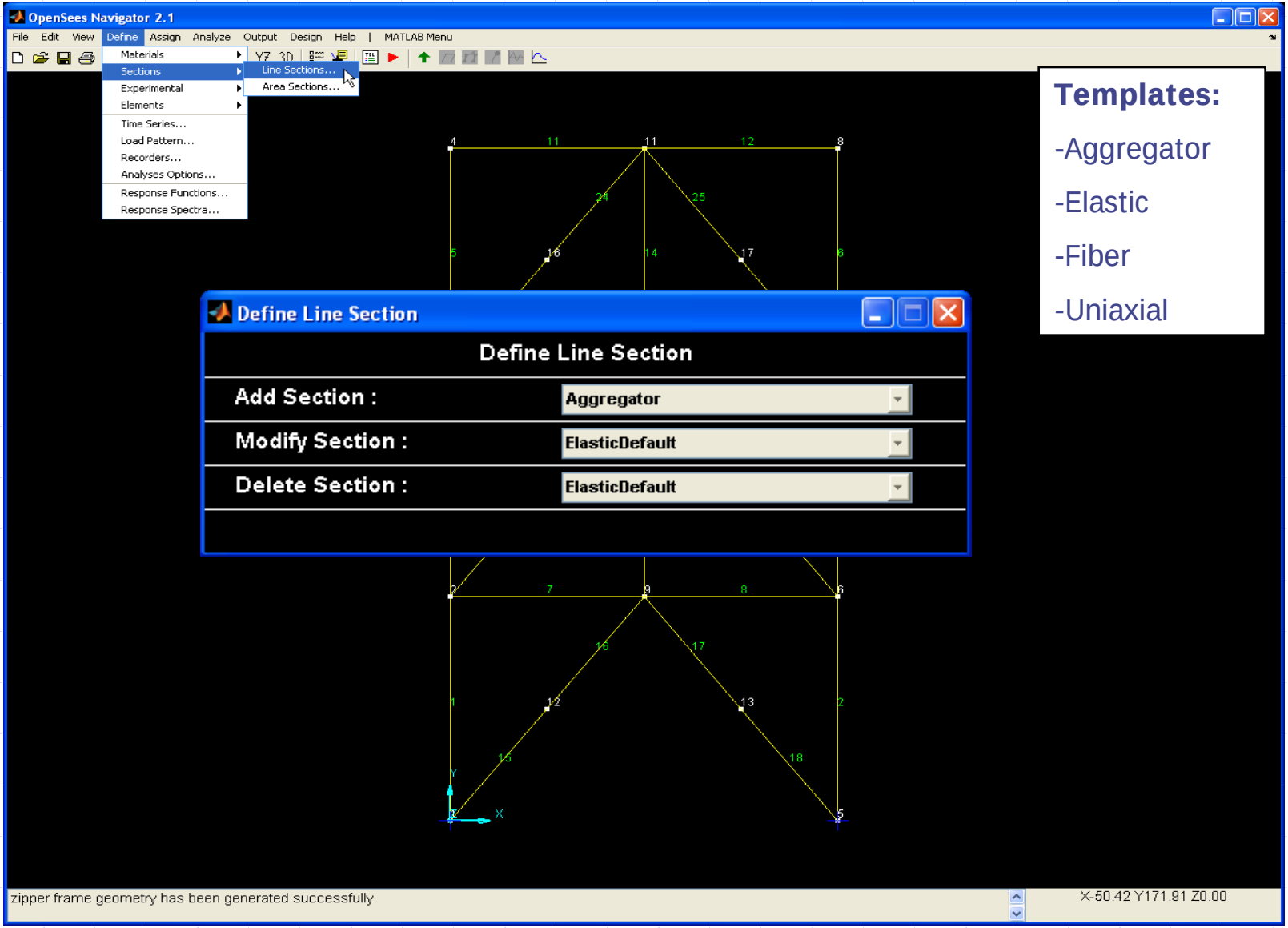
-Template3DElastoPlastic

Under development:

-NewTemplate3DElastoPlastic

-ModelsLargeDeformation

Define section: line sections



Templates:

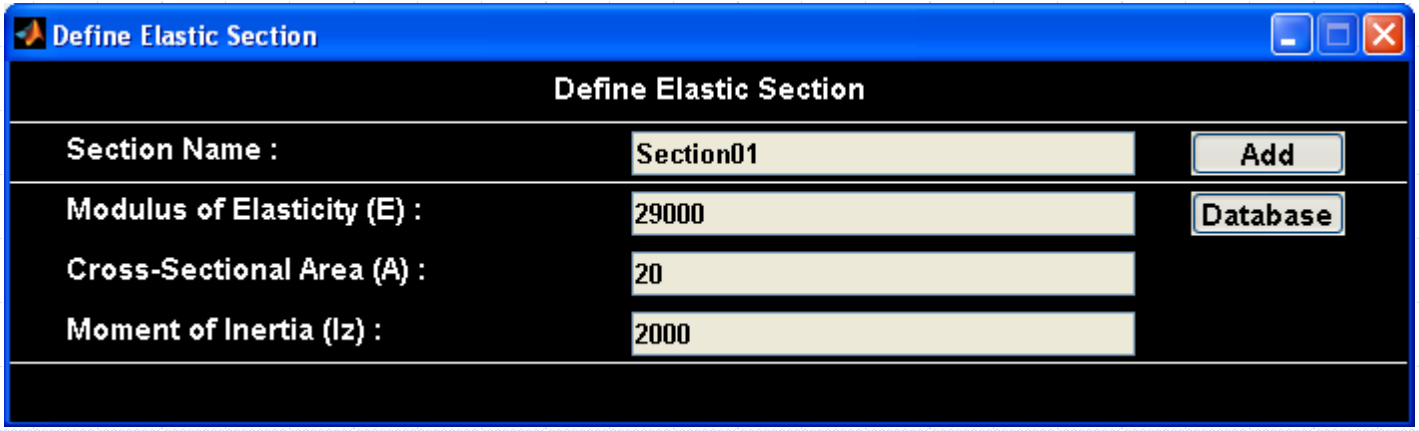
-Aggregator

-Elastic

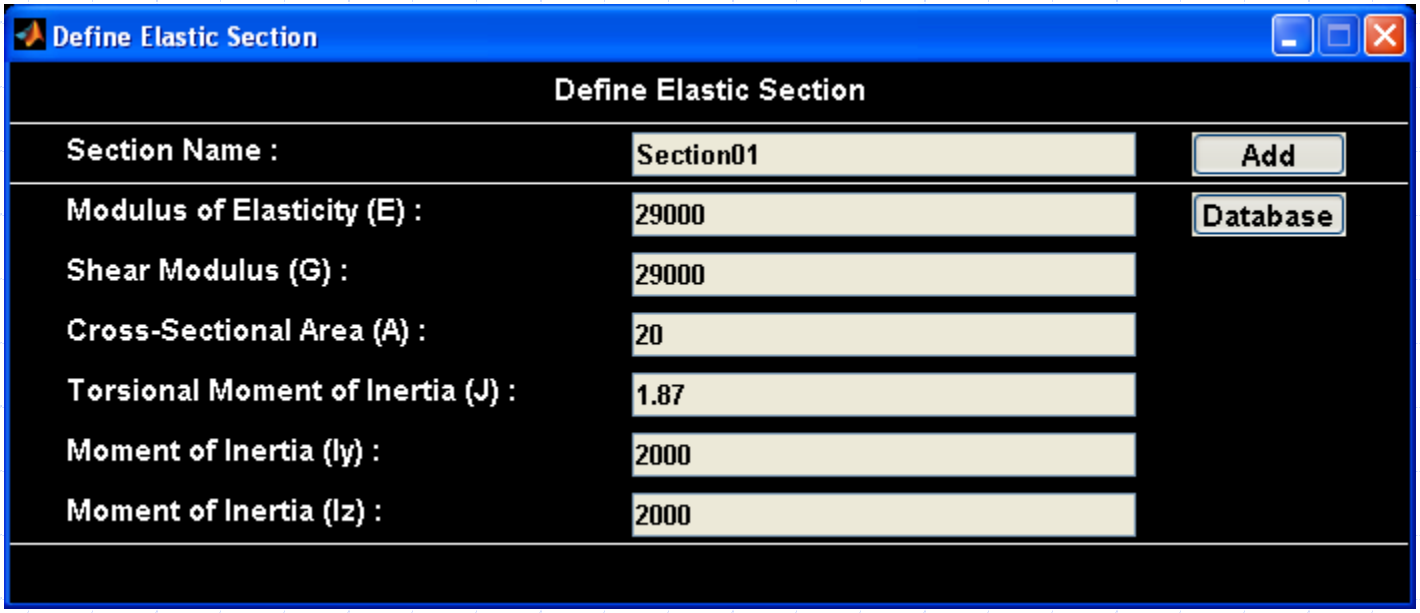
-Fiber

-Uniaxial

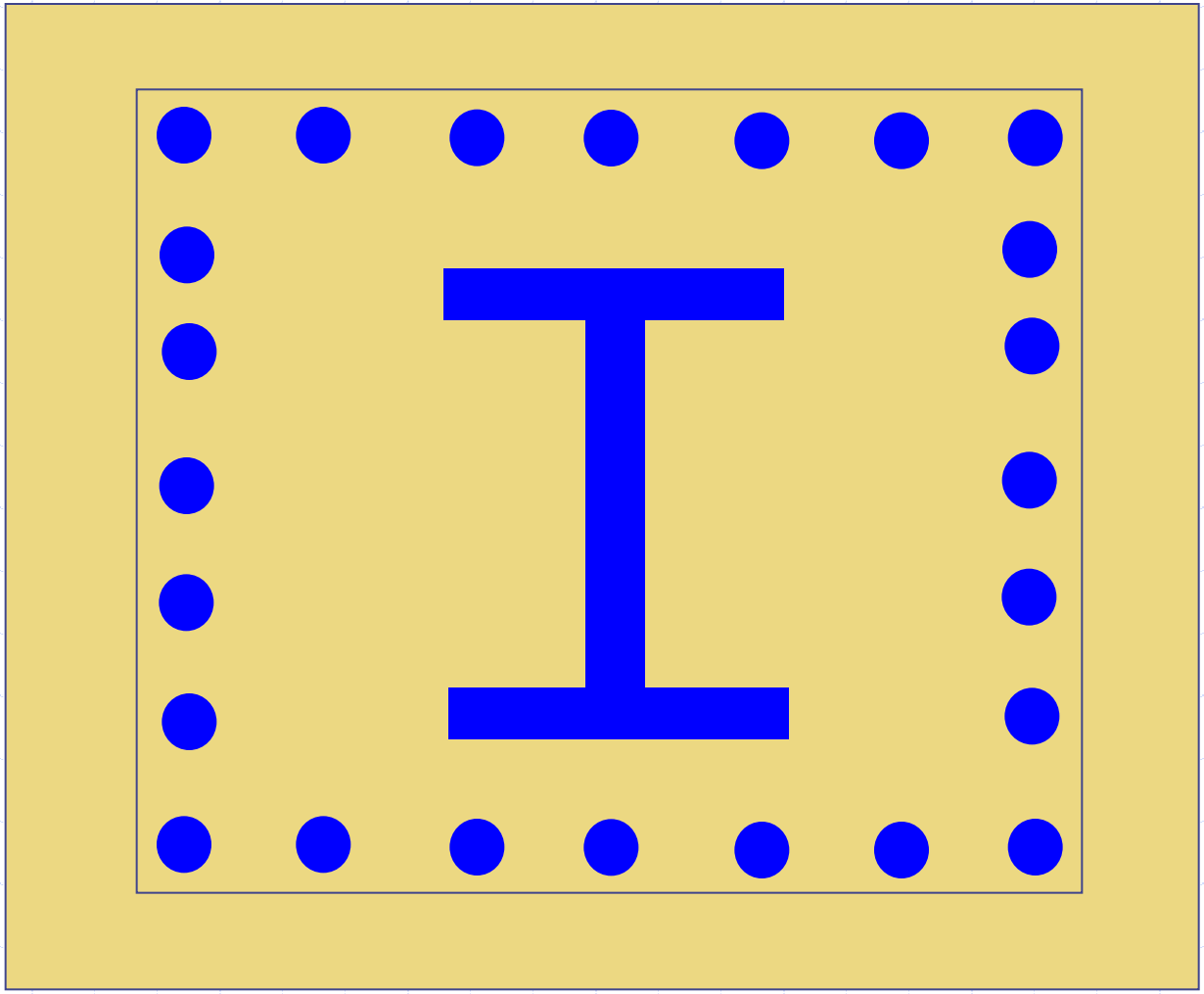
Define line section: elastic section



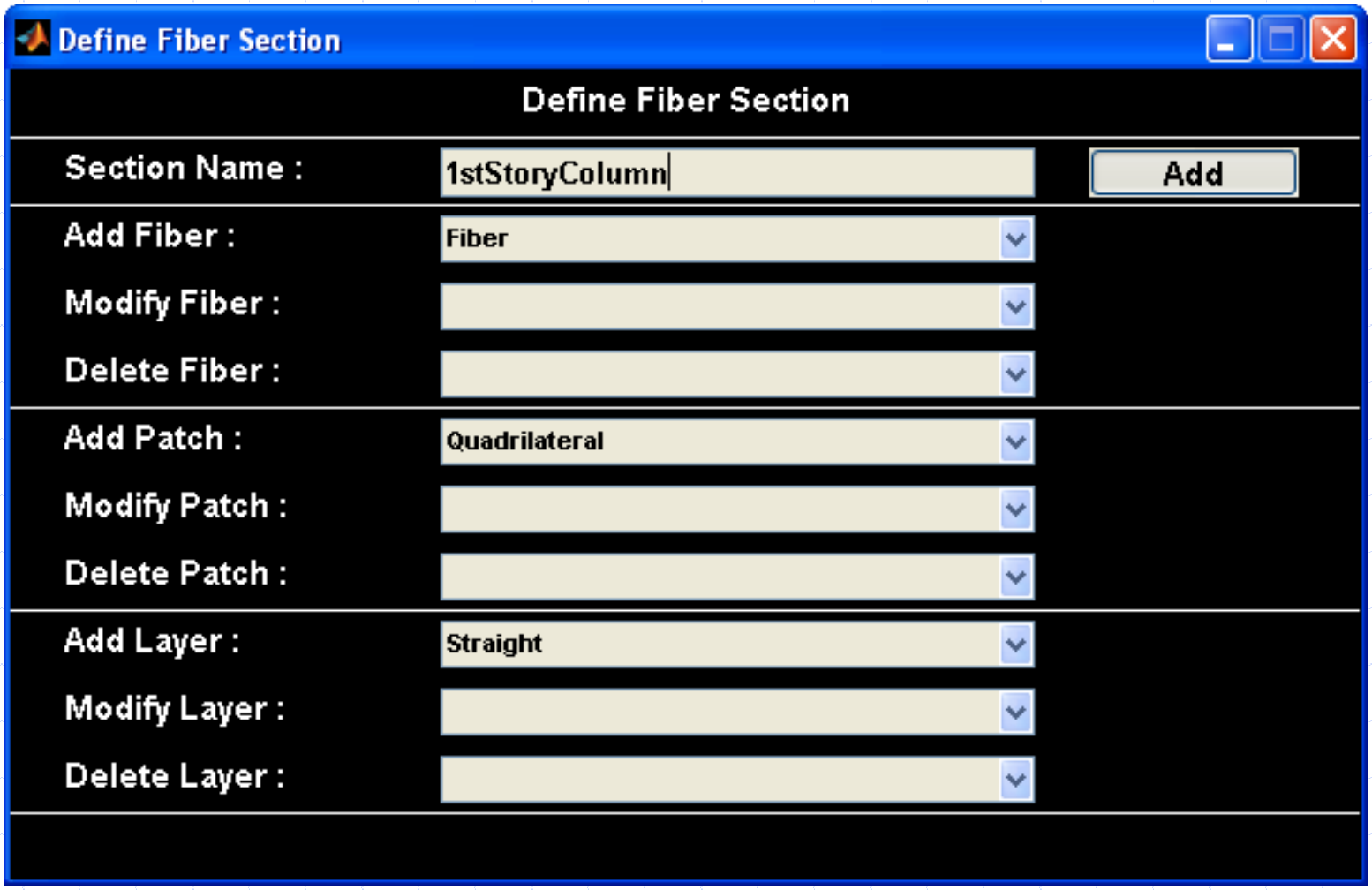
If the model is 3D



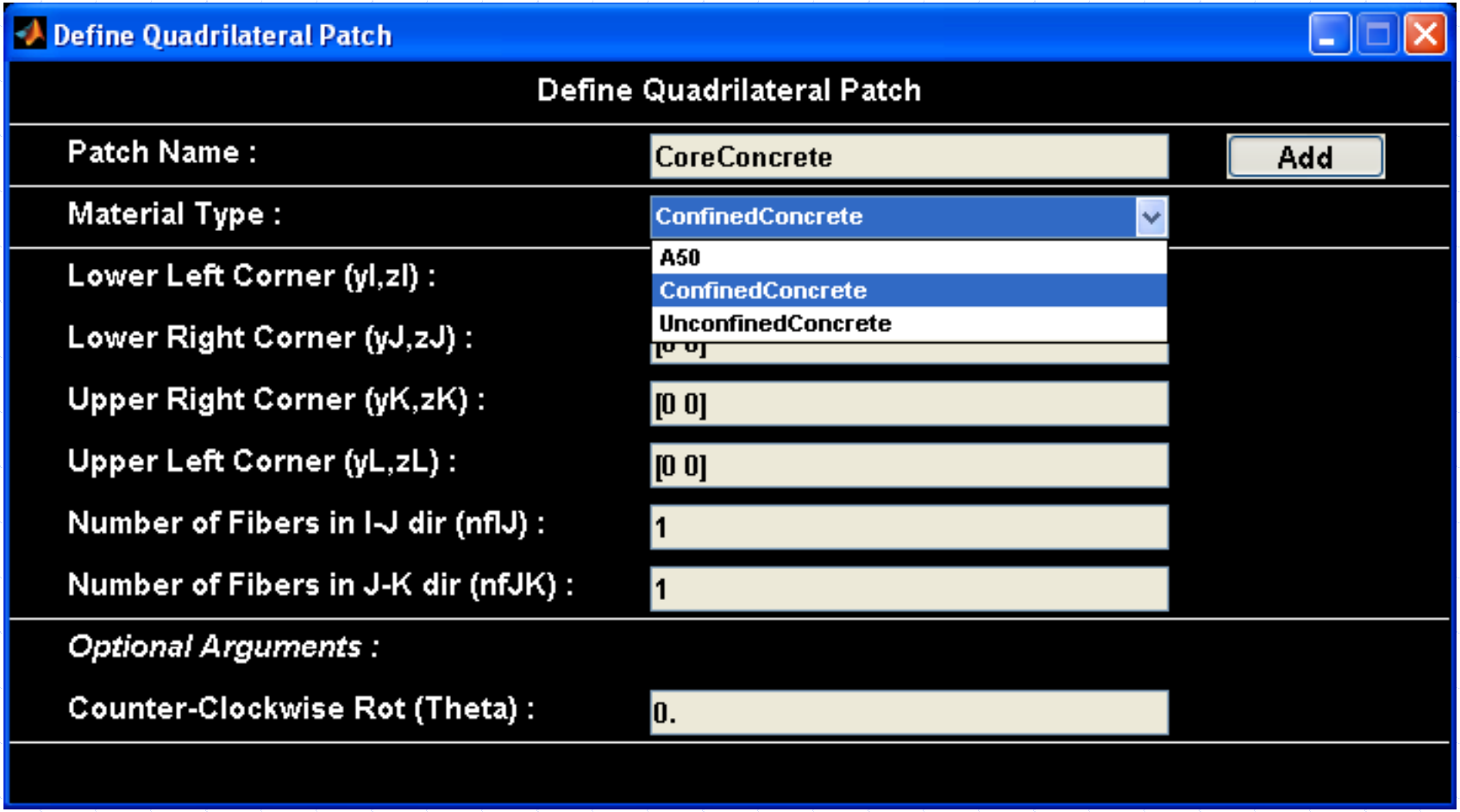
Define fiber section: Composite patch



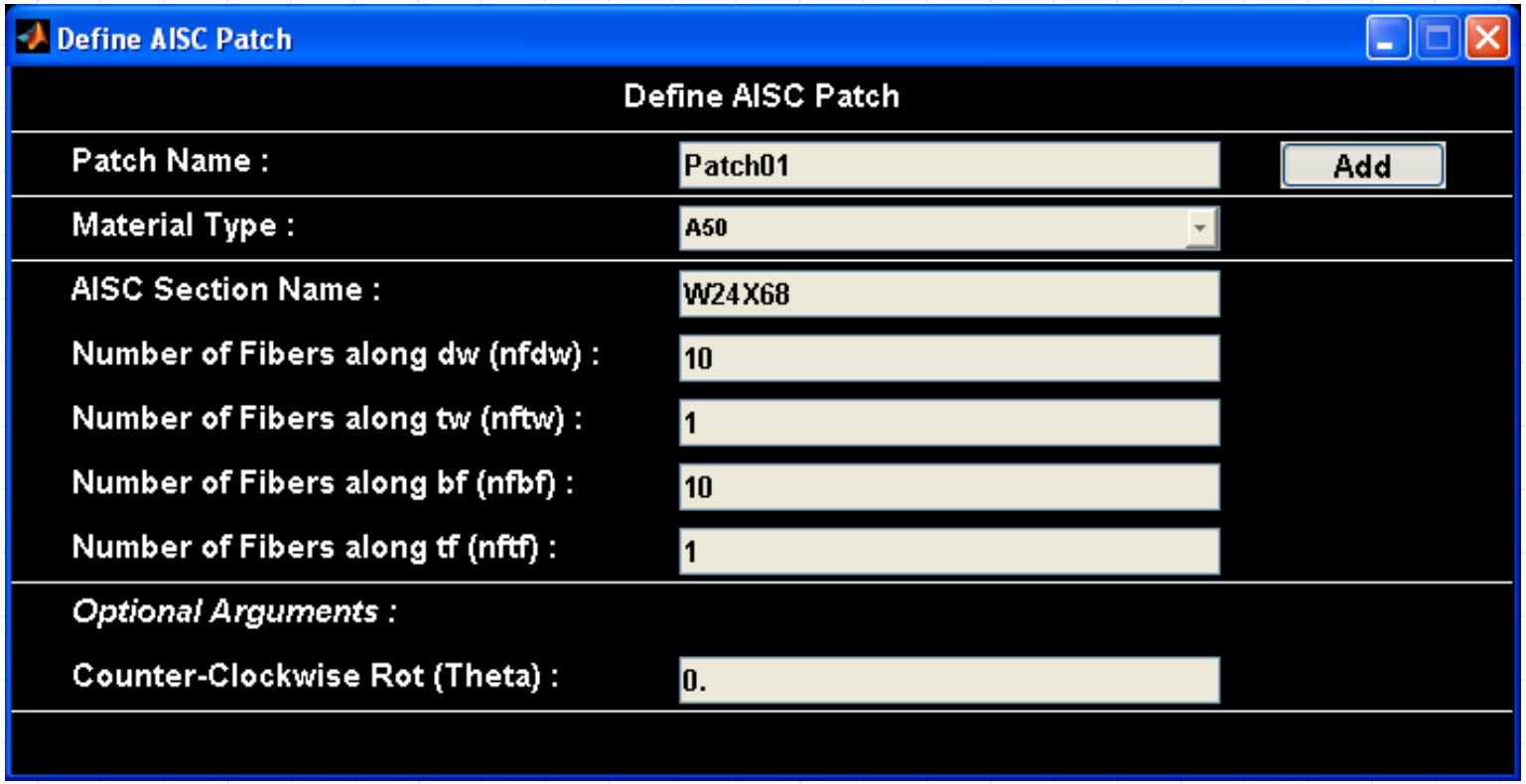
Define line section: fiber section



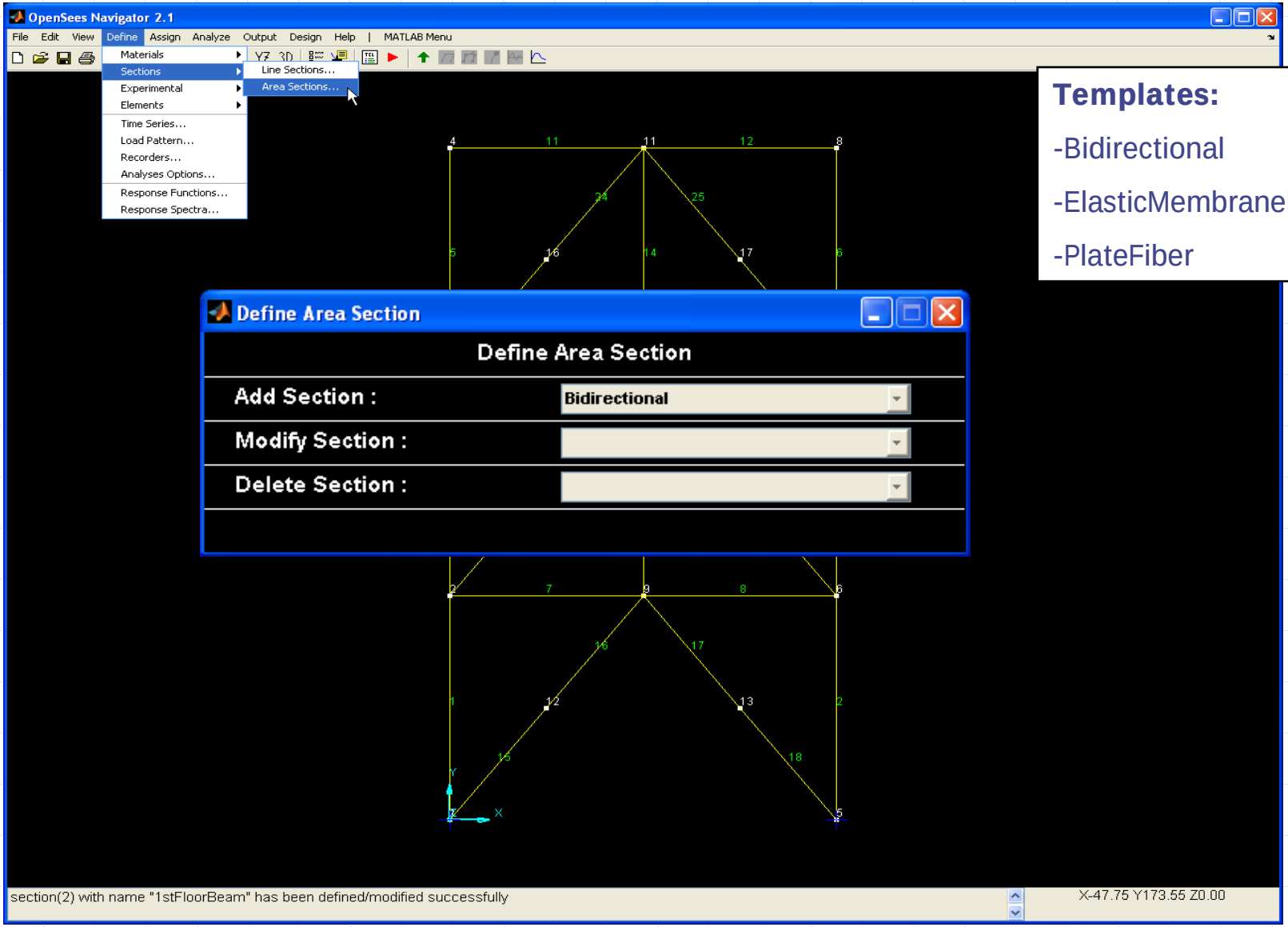
Define line section: quadrilateral patch



Define fiber section: AISC patch



Define section: area section



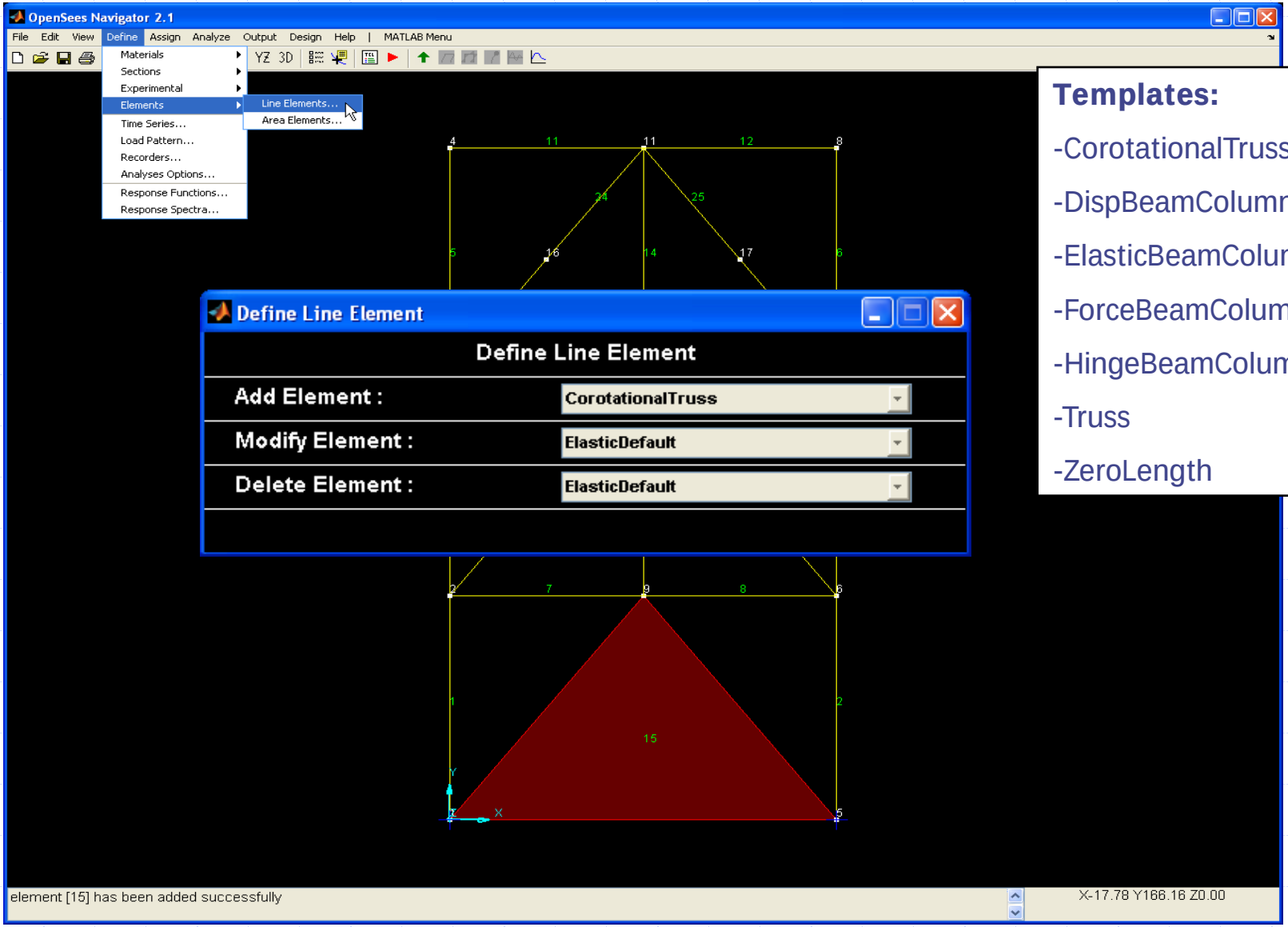
Templates:

-Bidirectional

-ElasticMembranePlate

-PlateFiber

Define element: line element



Templates:

-CorotationalTruss

-DispBeamColumn

-ElasticBeamColumn

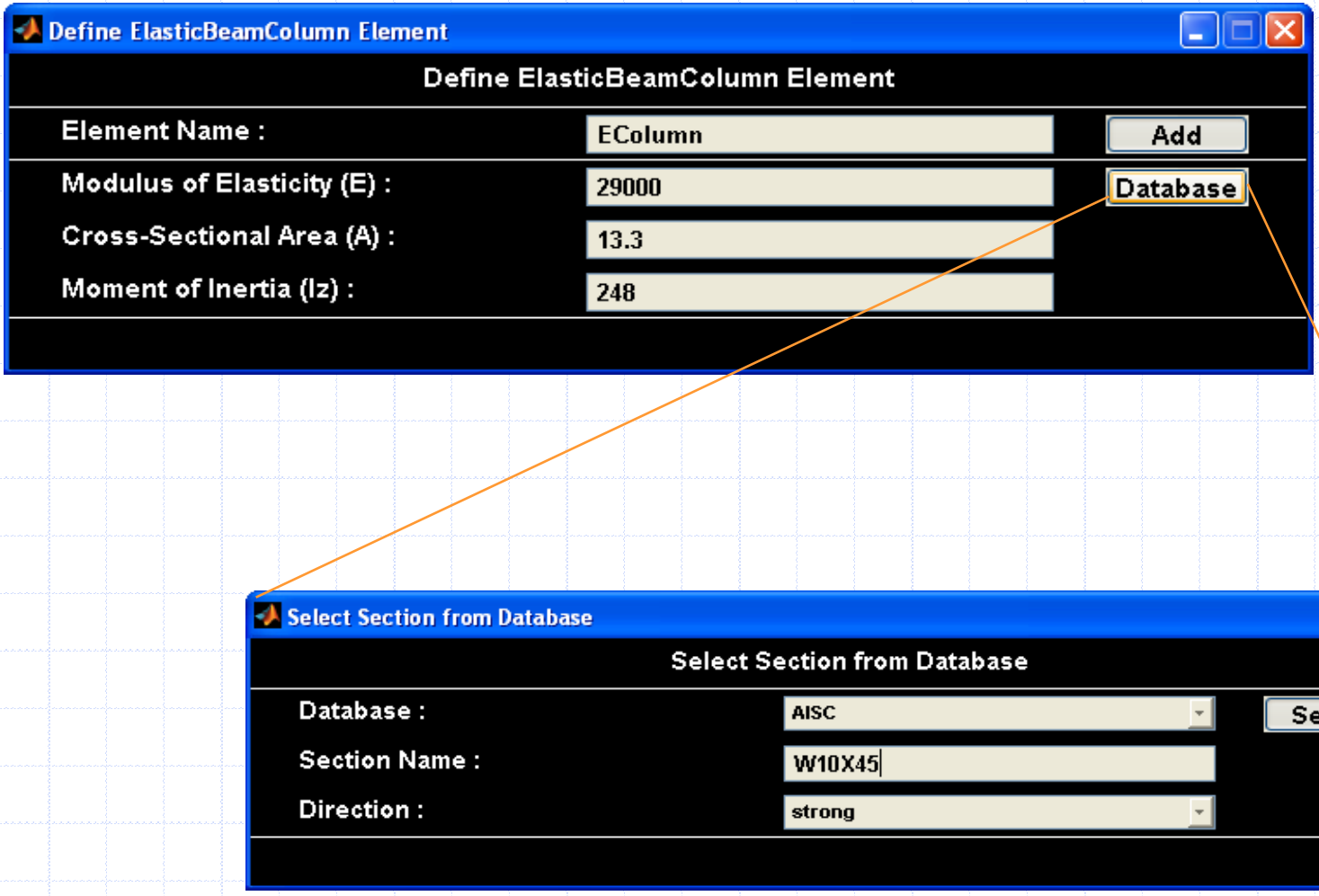
-ForceBeamColumn

-HingeBeamColumn

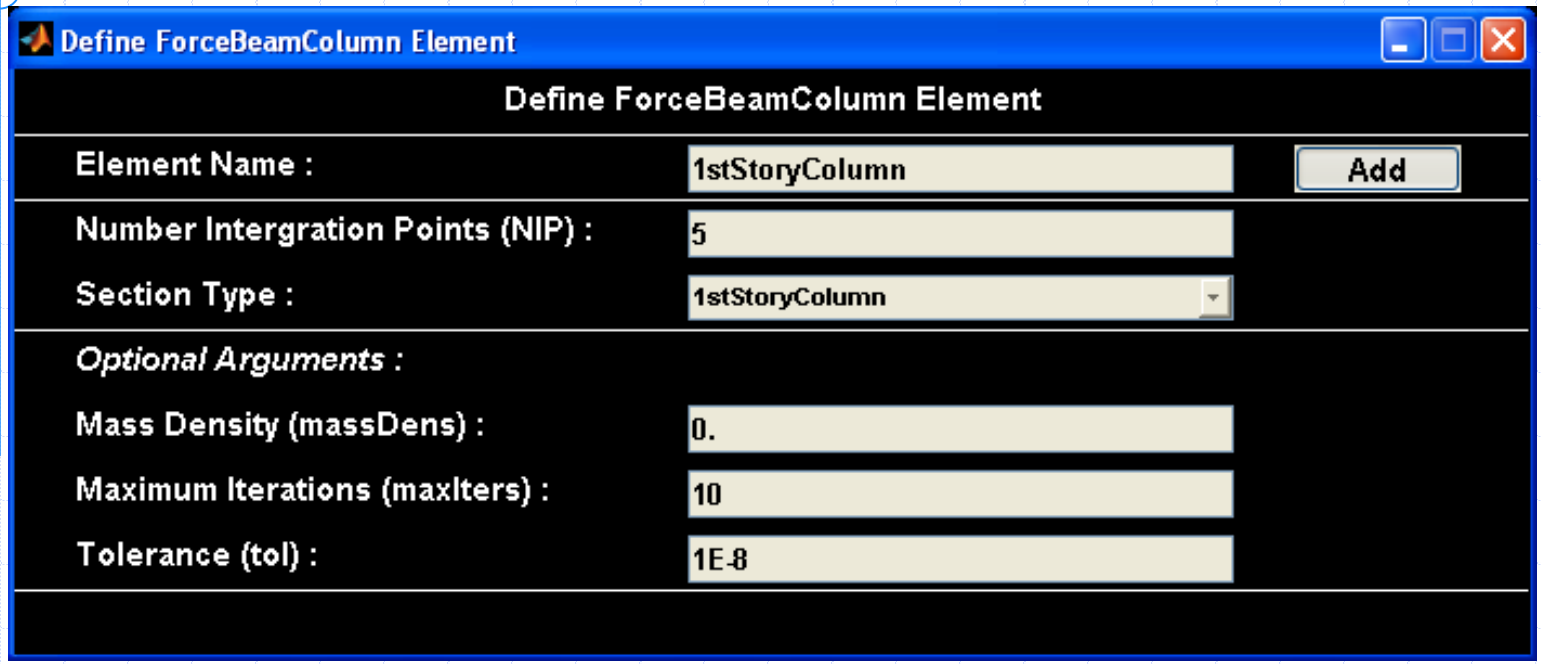
-Truss

-ZeroLength

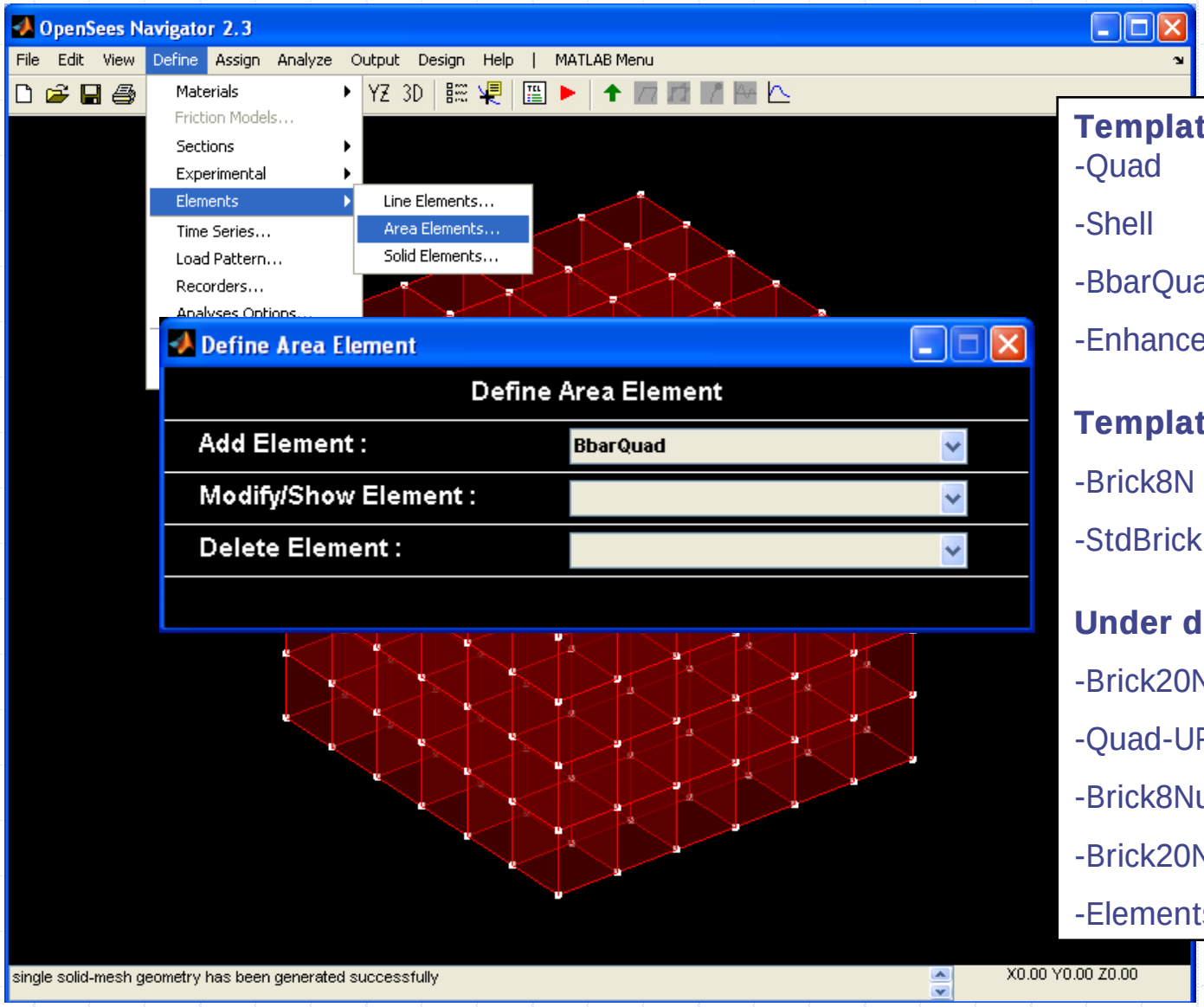
Define line element: ElasticBeamColumn



Define line element: ForceBeamColumn



Define element: area and solid elements



Templates (area):

-Quad

-Shell

-BbarQuad

-EnhancedQuad

Templates (solid):

-Brick8N

-StdBrick

Under development:

-Brick20N

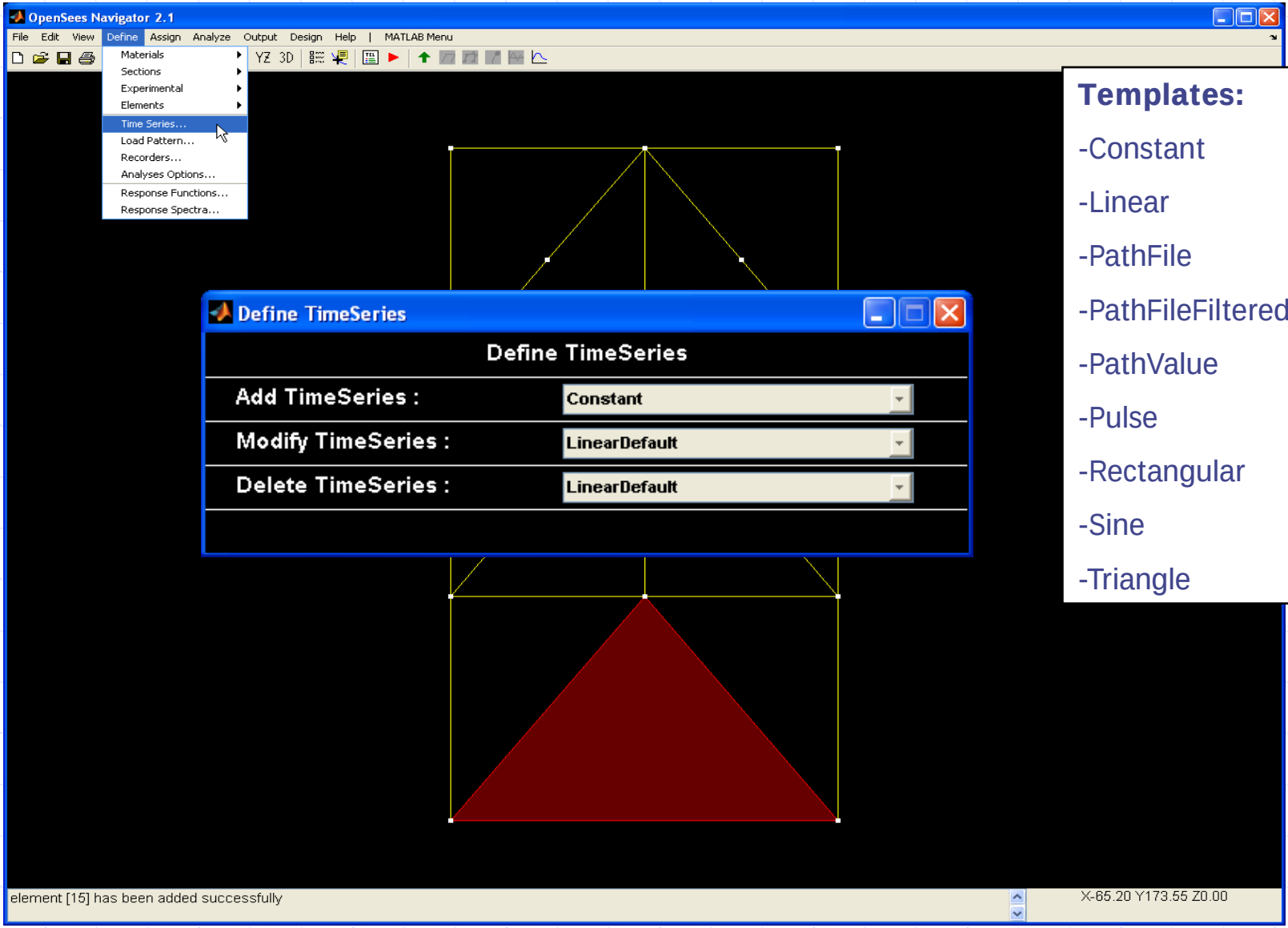
-Quad-UP

-Brick8Nu-p-U

-Brick20Nu-p-U

-ElementsLargeDeformation

Define TimeSeries:



Templates:

-Constant

-Linear

-PathFile

-PathFileFiltered

-PathValue

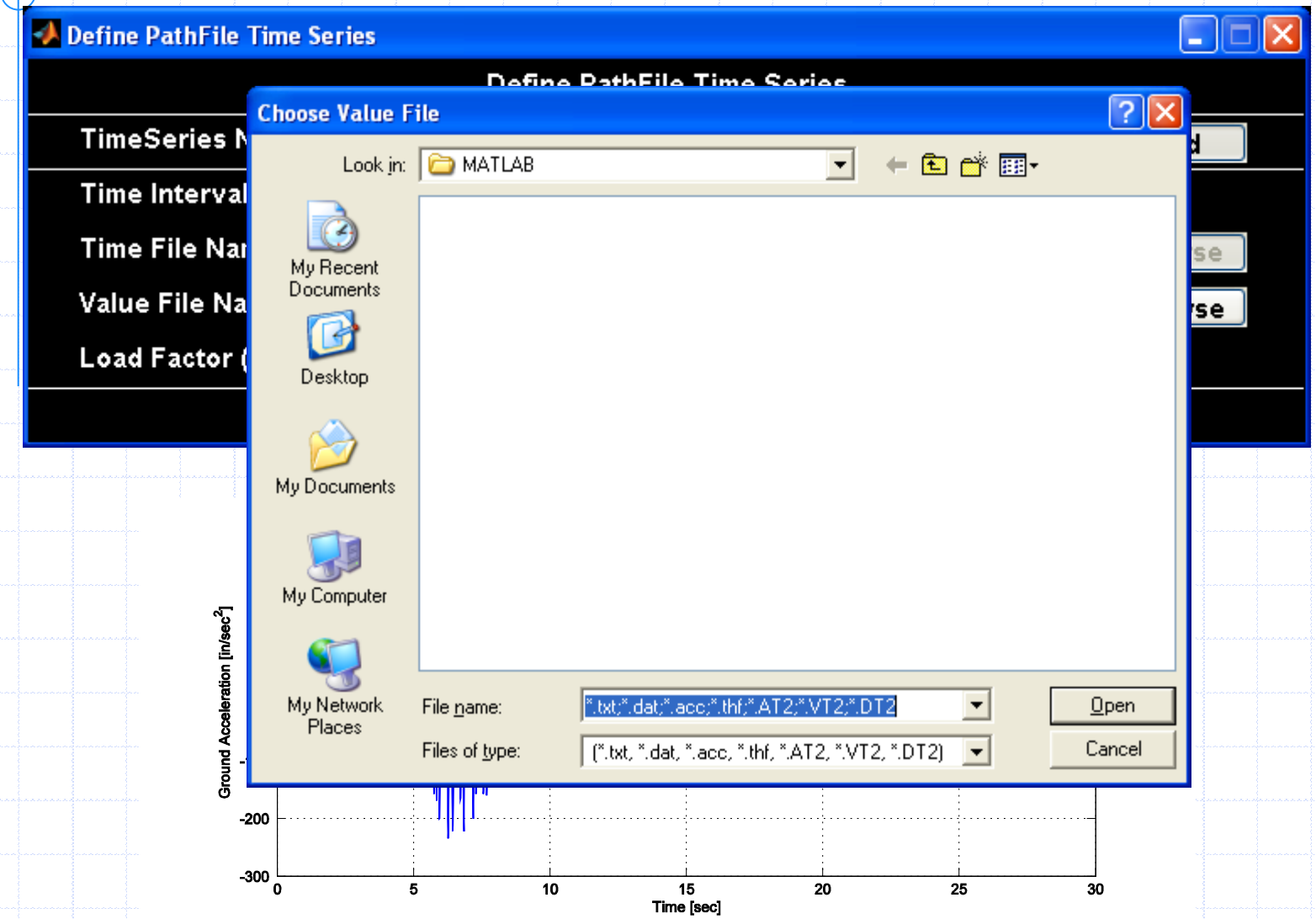
-Pulse

-Rectangular

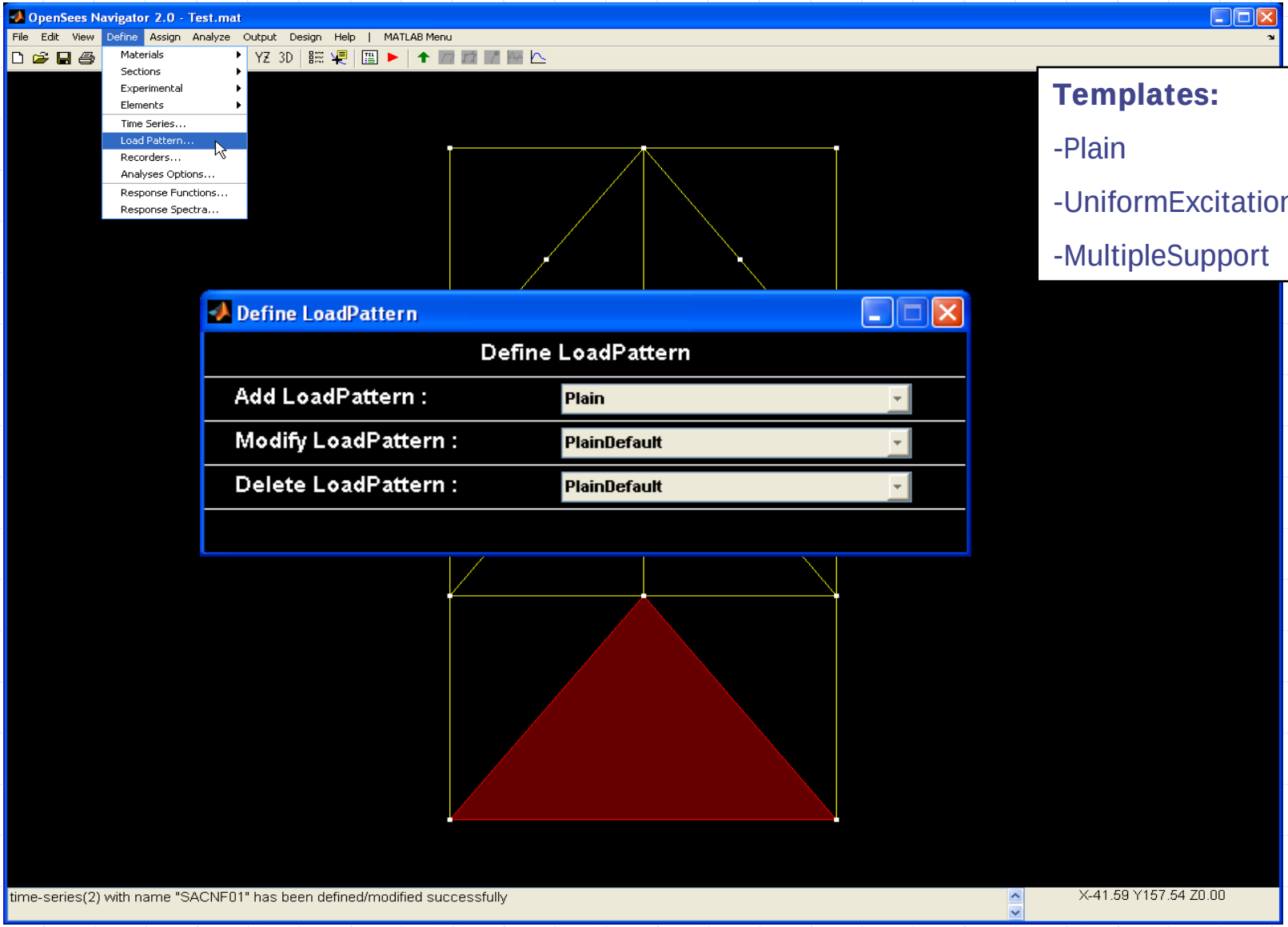
-Sine

-Triangle

Define TimeSeries: PathFile



Define LoadPattern:



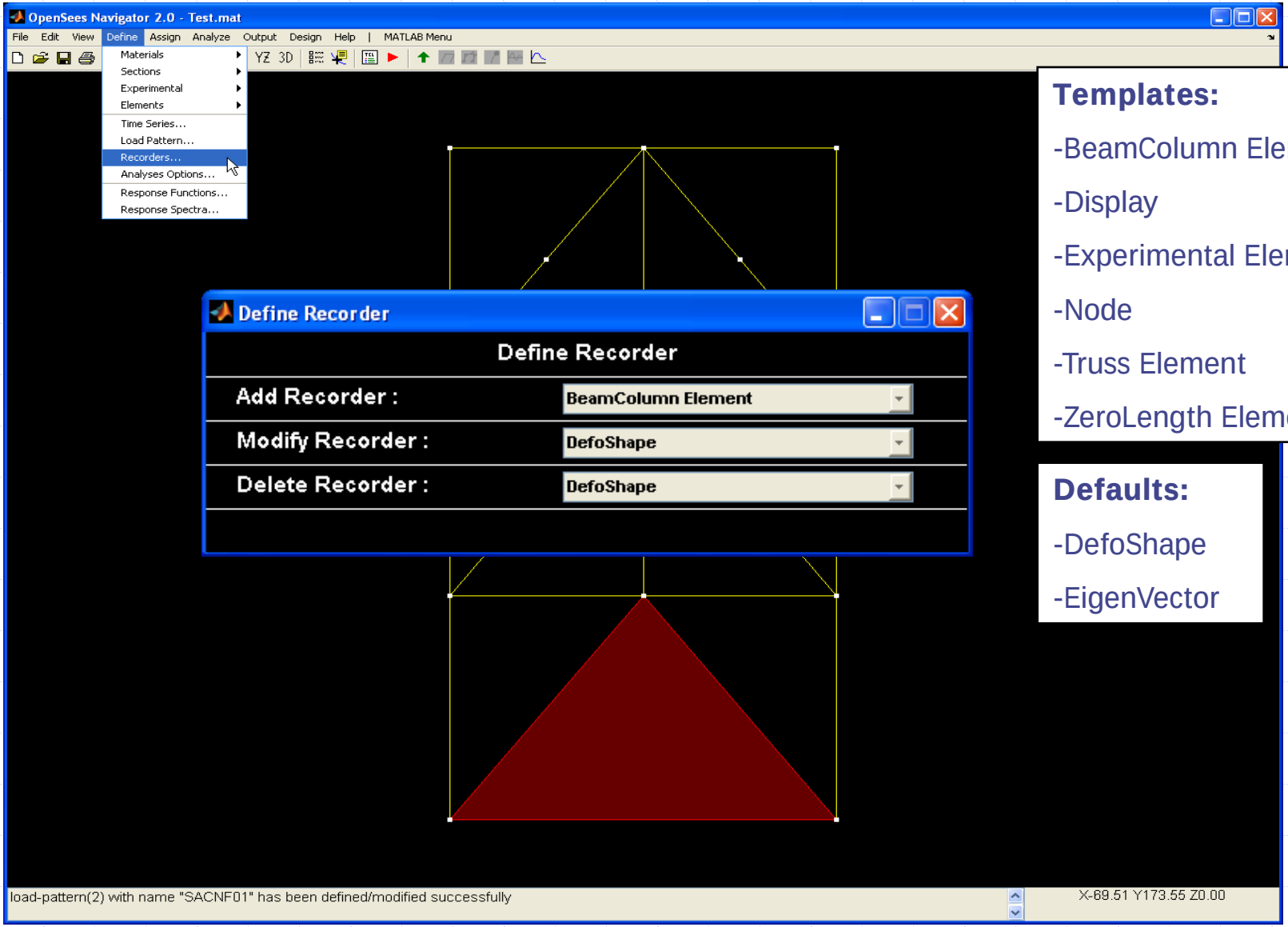
Templates:

-Plain

-UniformExcitation

-MultipleSupport

Define recorder



Templates:

-BeamColumn Element

-Display

-Experimental Element

-Node

-Truss Element

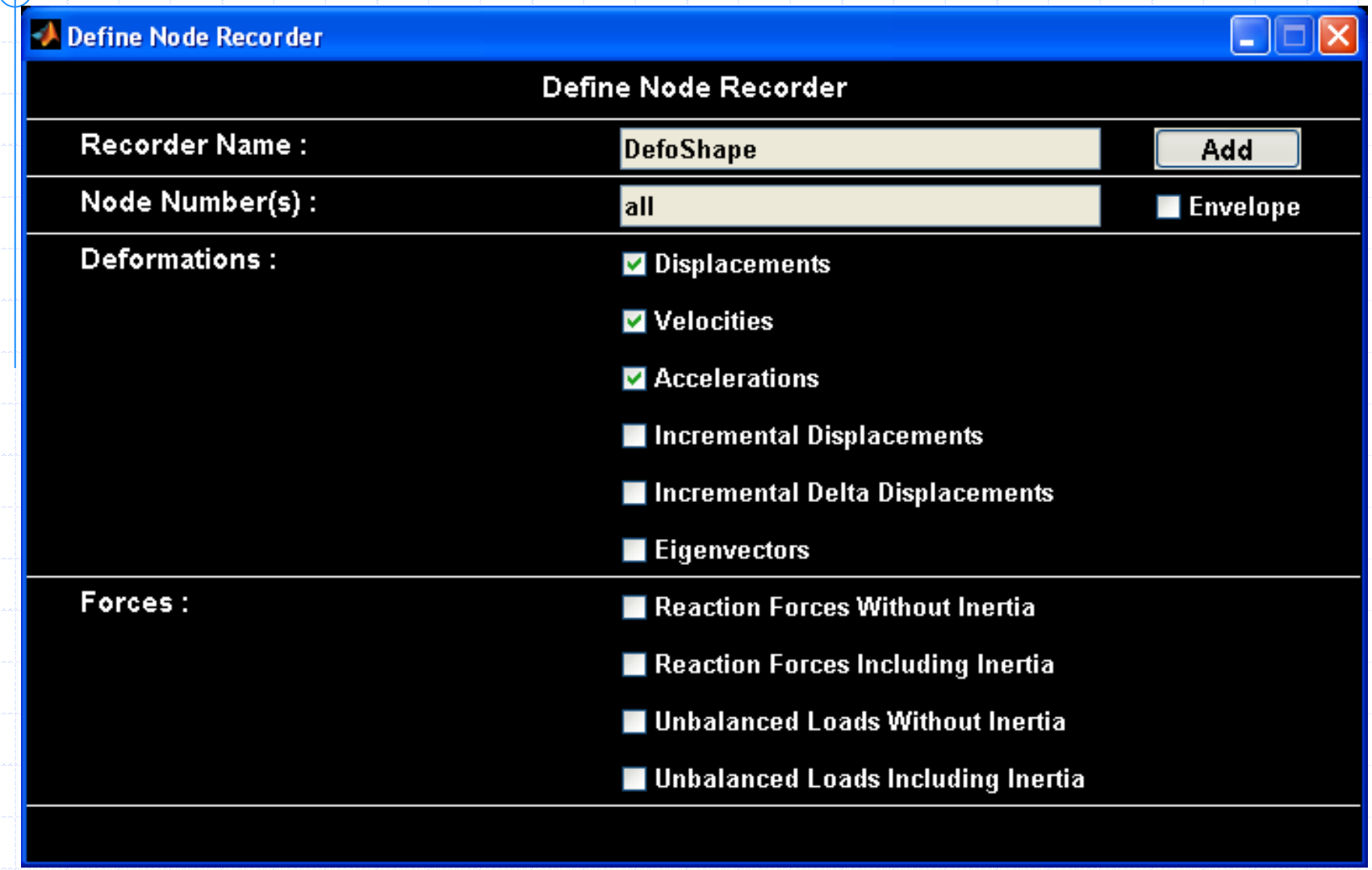
-ZeroLength Element

Defaults:

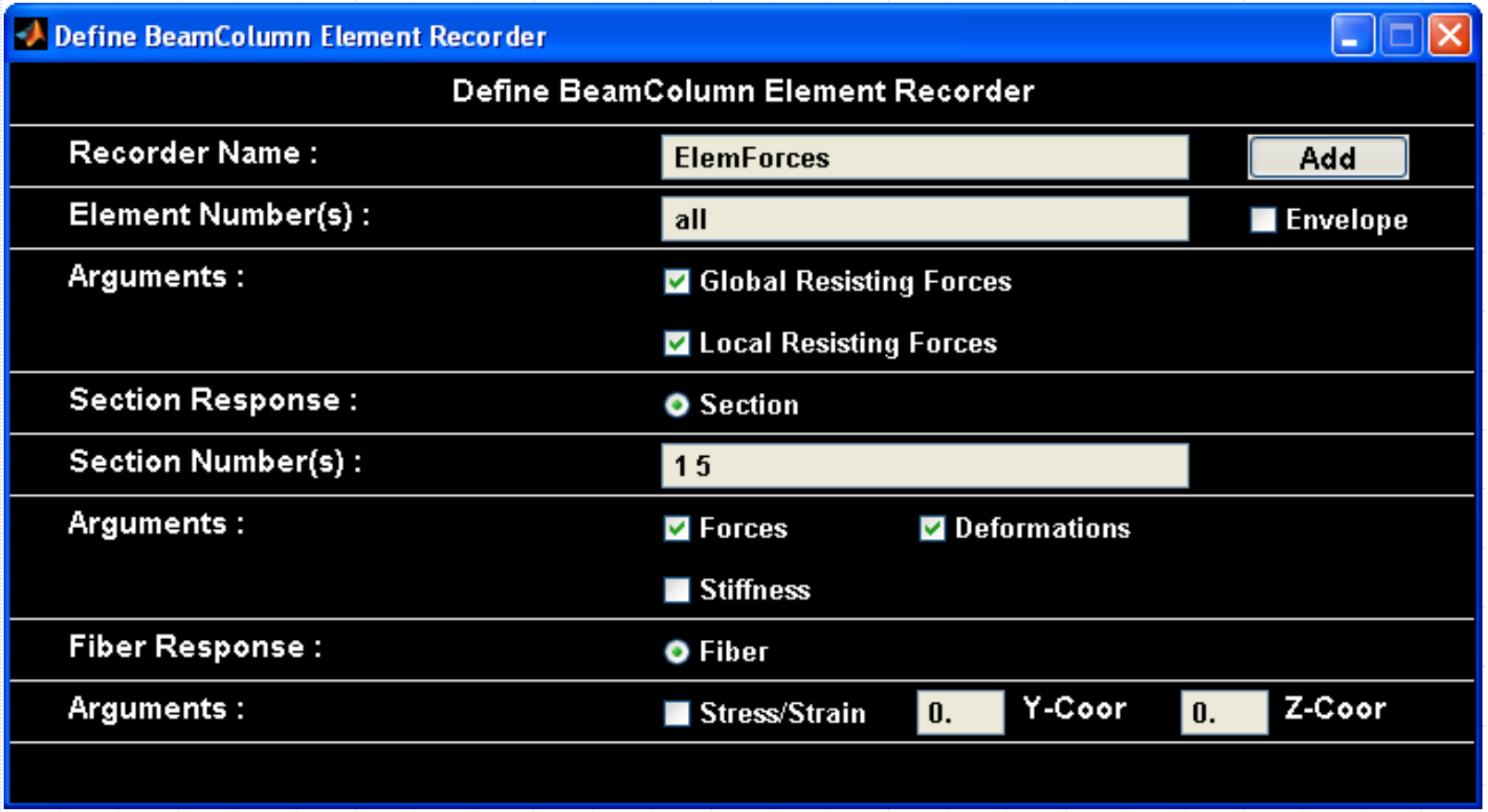
-DefoShape

-EigenVector

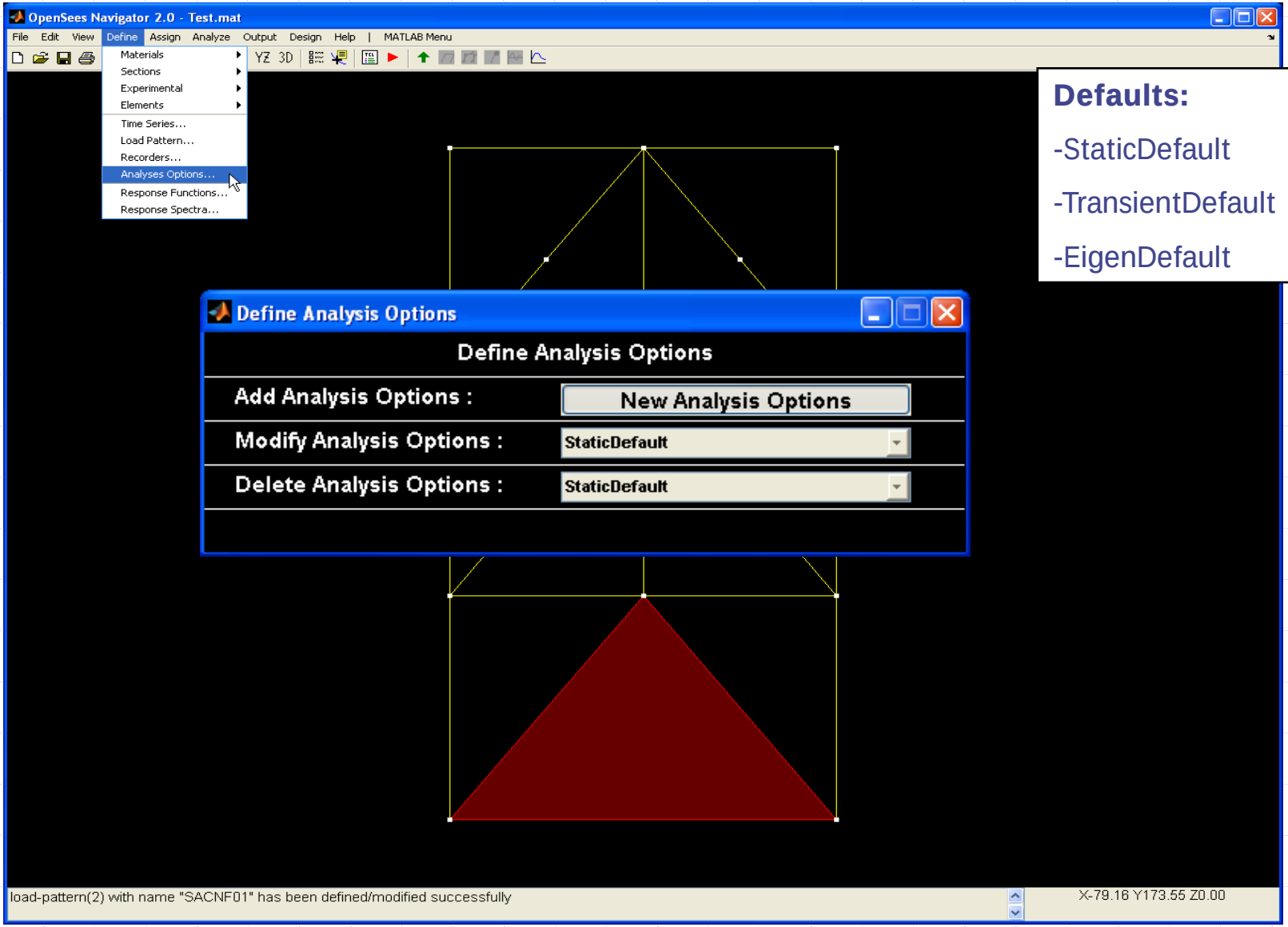
Define recorder: node recorder



Define recorder: BeamColumn recorder



Define analysis options



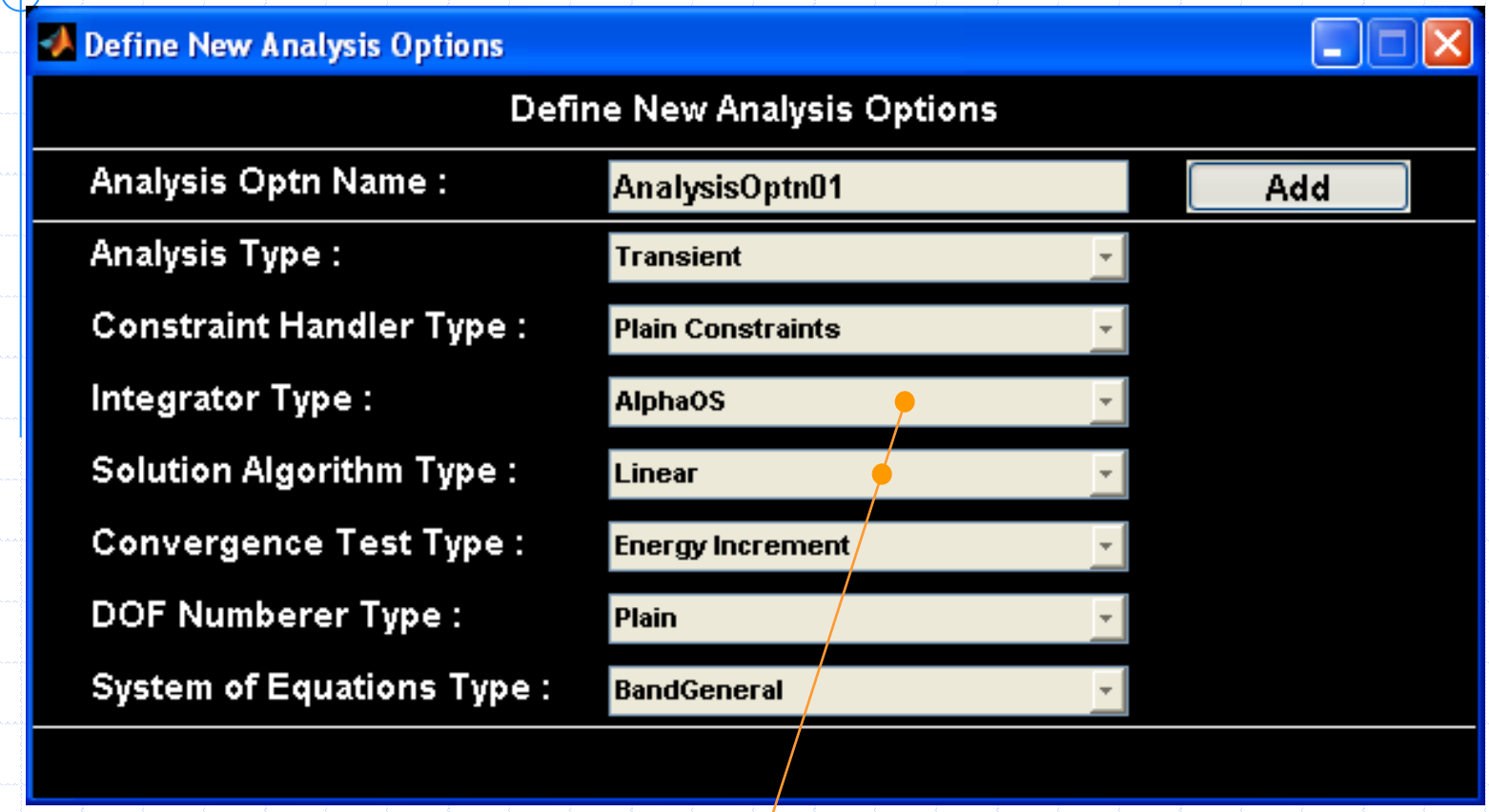
Defaults:

-StaticDefault

-TransientDefault

-EigenDefault

Define analysis options: new analysis



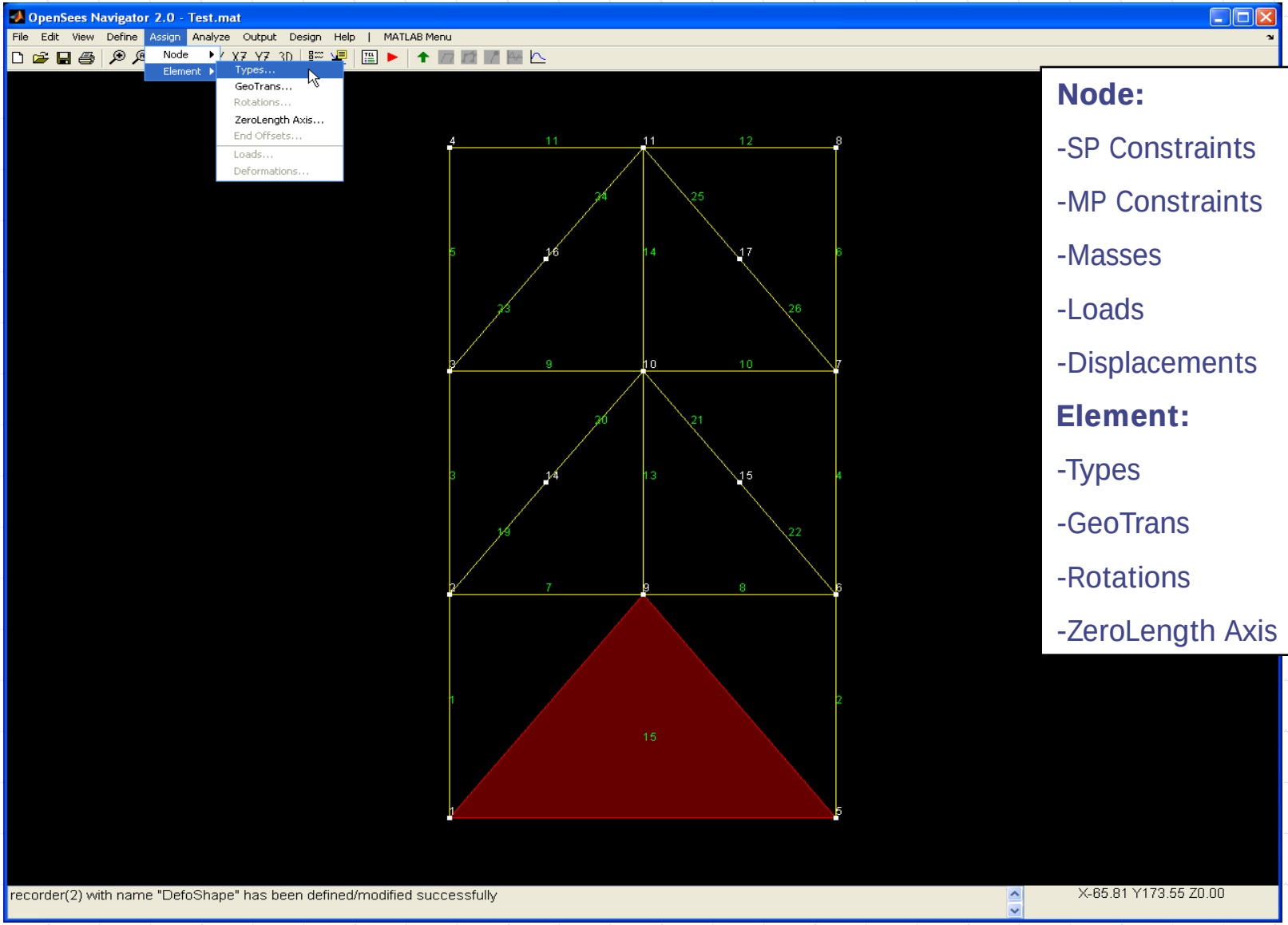
Integrator Type:

For example use AlphaOS Method for Hybrid Simulation

Solution Algorithm:

The AlphaOS Method requires a Linear solution algorithm

Assign menu



Node:

-SP Constraints

-MP Constraints

-Masses

-Loads

-Displacements

Element:

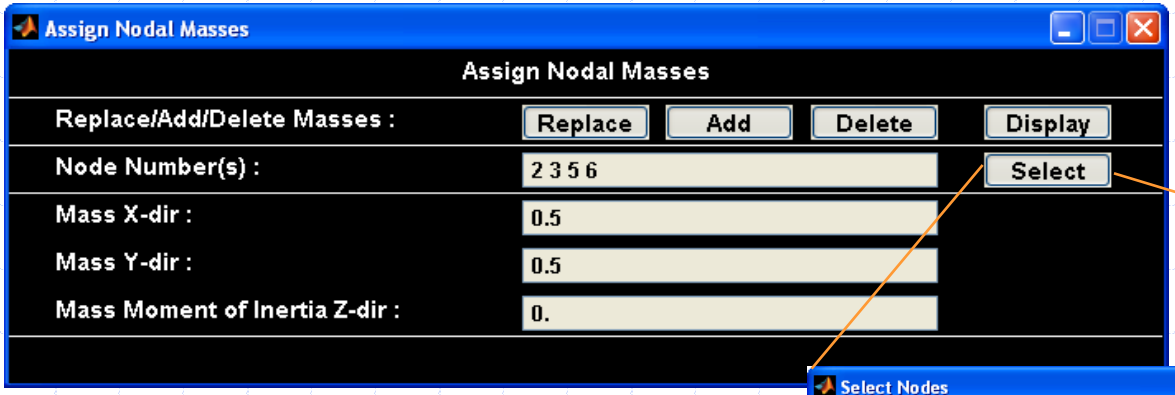
-Types

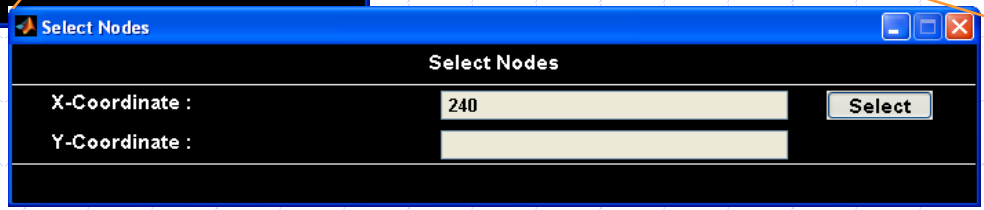
-GeoTrans

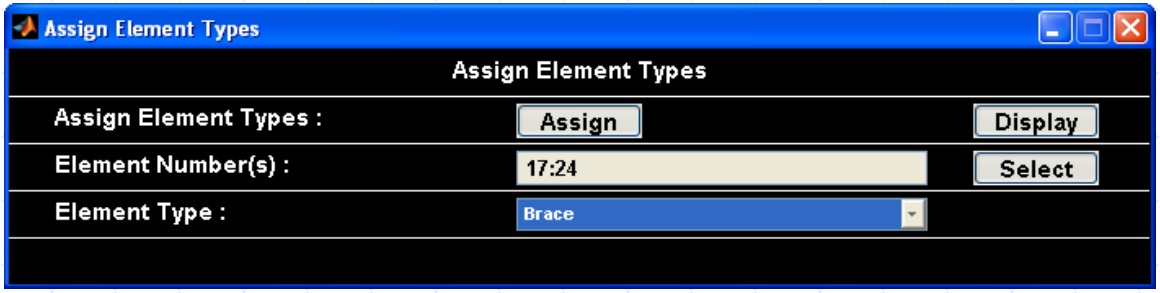
-Rotations

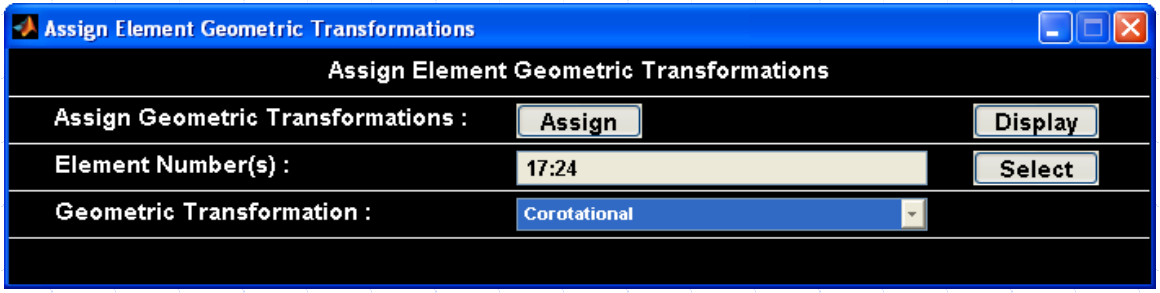
-ZeroLength Axis

Assign menu

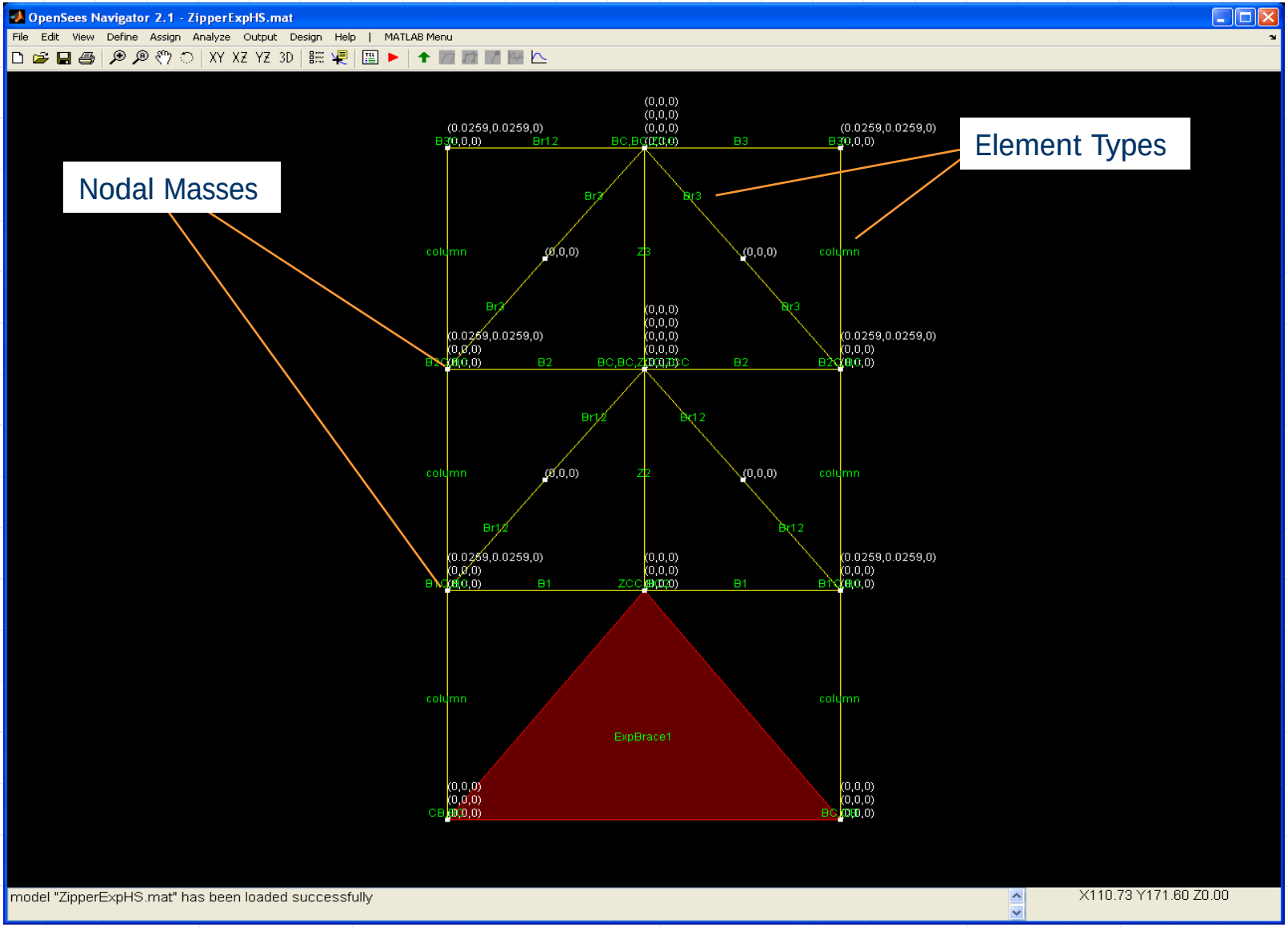

Assign Nodal Masses
Assign Nodal Masses
Replace/Add/Delete Masses :
Replace
Add
Delete
Display
Node Number(s) :
2 3 5 6
Select
Mass X-dir :
0.5
Mass Y-dir :
0.5
Mass Moment of Inertia Z-dir :
0.


Select Nodes
Select Nodes
X-Coordinate :
240
Select
Y-Coordinate :

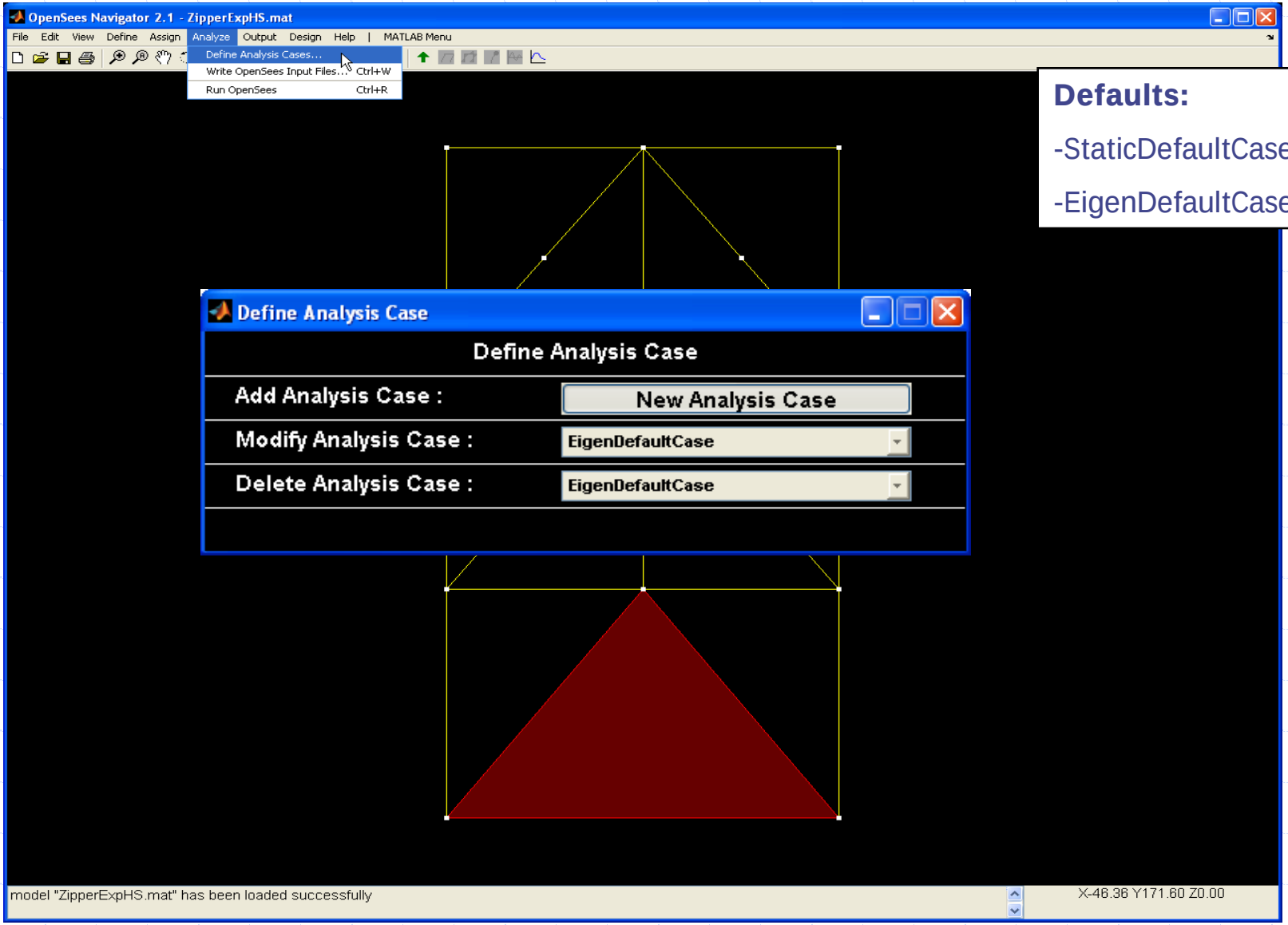

Assign Element Types
Assign Element Types
Assign Element Types :
Assign
Display
Element Number(s) :
17:24
Select
Element Type :
Brace


Assign Element Geometric Transformations
Assign Element Geometric Transformations
Assign Geometric Transformations :
Assign
Display
Element Number(s) :
17:24
Select
Geometric Transformation :
Corotational

Display assigned properties



Define analysis case

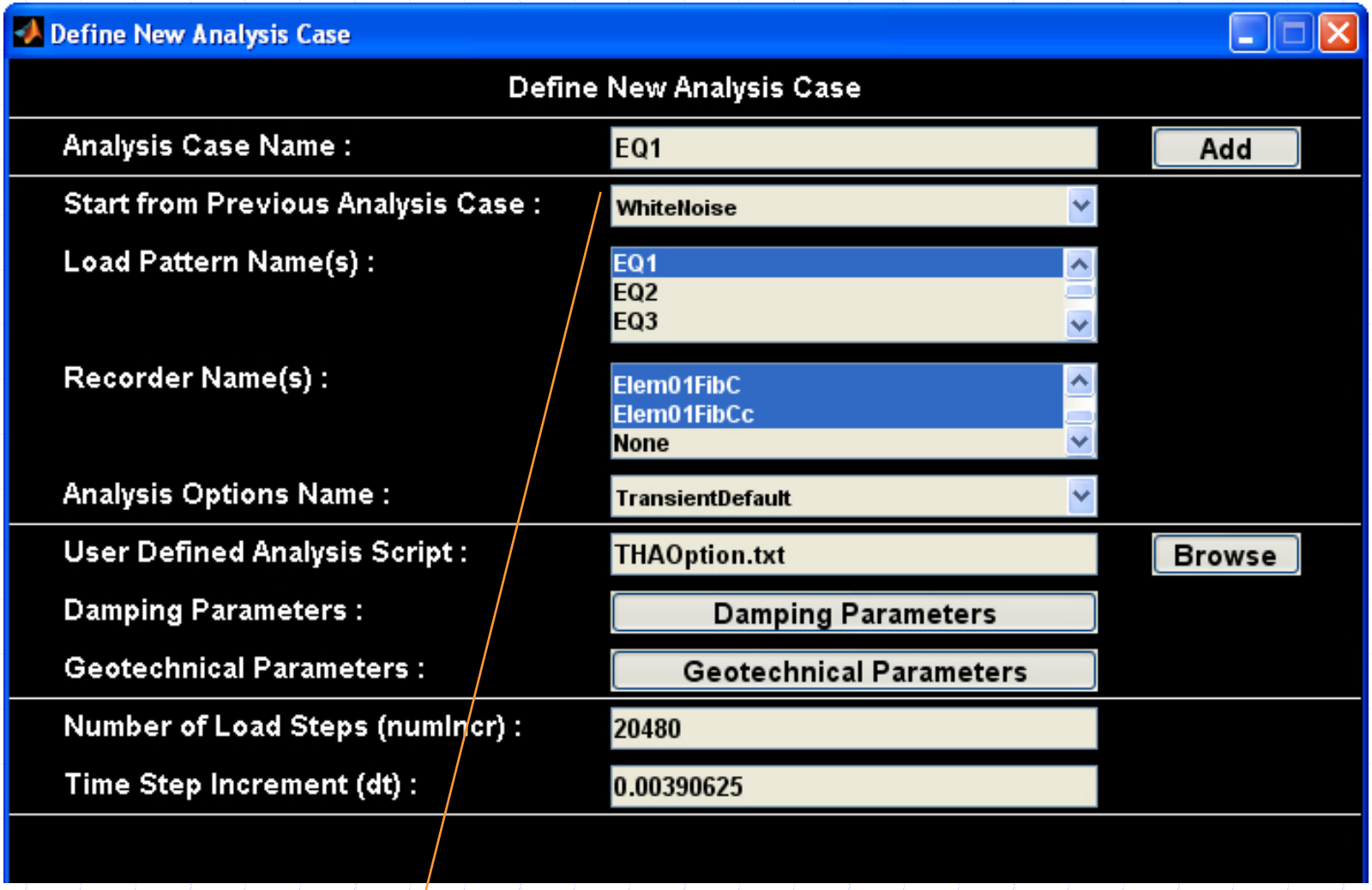


Defaults:

-StaticDefaultCase

-EigenDefaultCase

Define analysis case: new analysis case

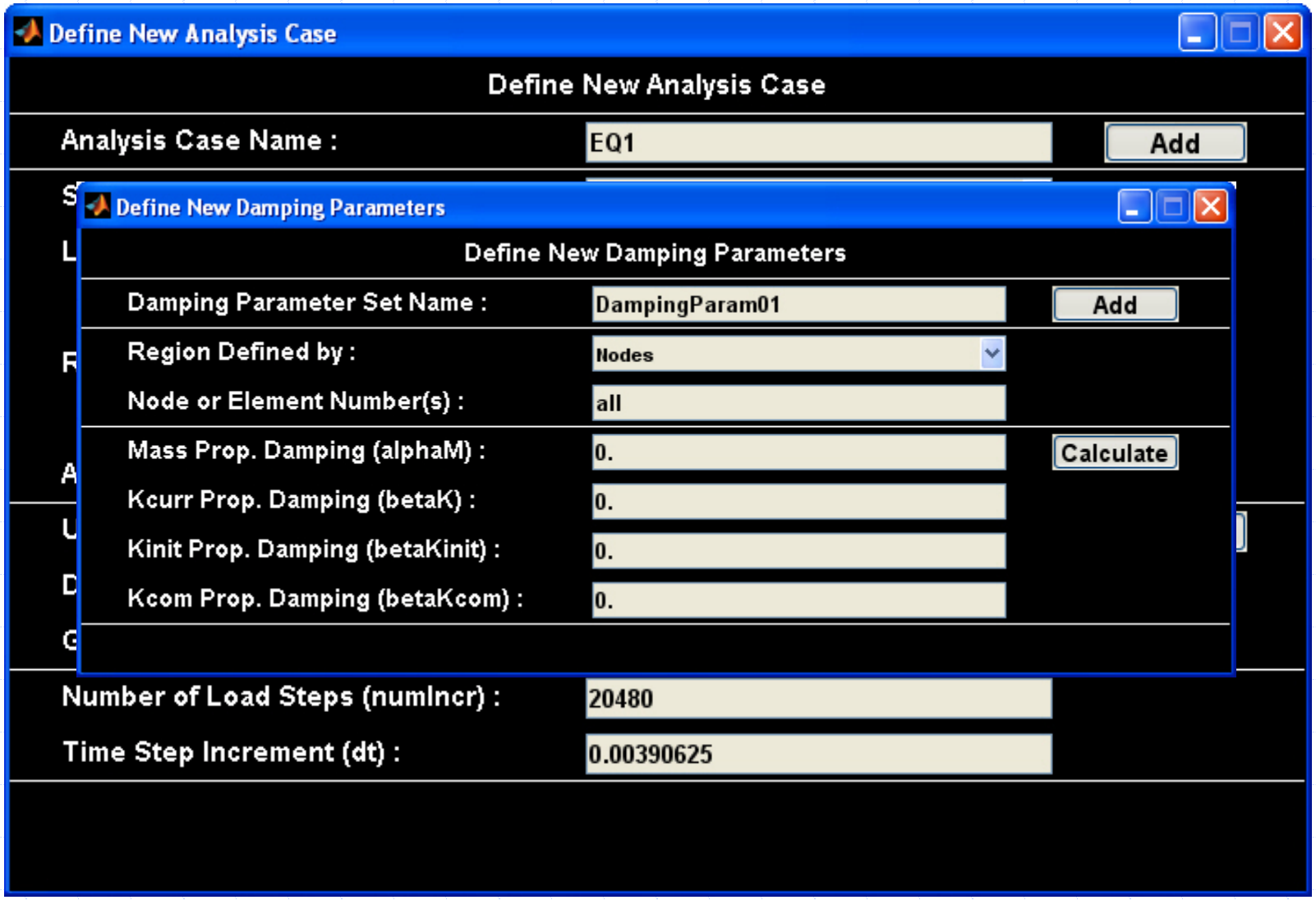


For Example:
Run the Time-History Analysis after the gravity load

User Defined Analysis Script

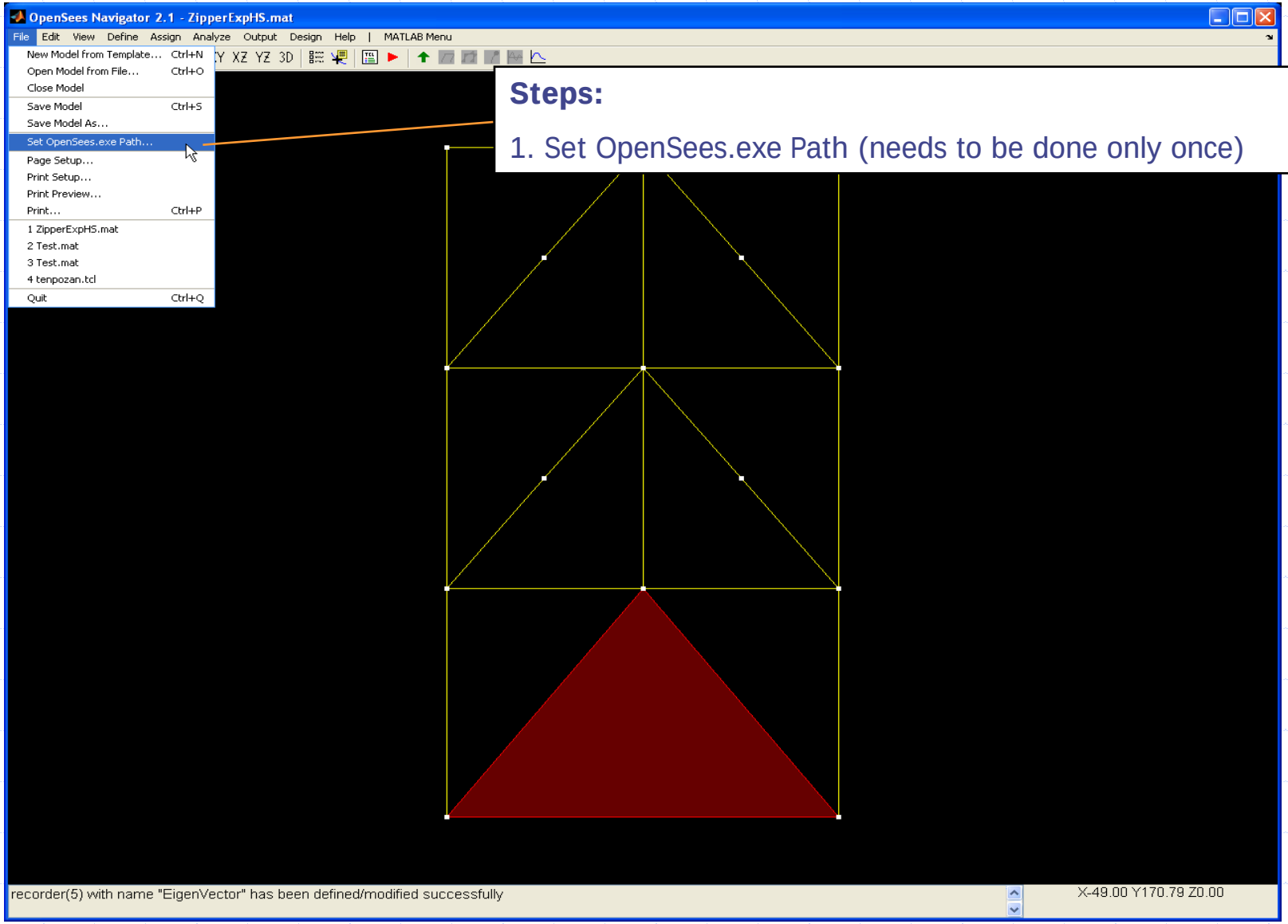
```
# set the test parameters
set testType NormDispIncr
set testTol 1.0e-8;
set testIter 25;
test  $testType $testTol $testIter

# set the algorithm parameters
set algoType KrylovNewton
algorithm  $algoType

set ok 0;
set tFinal [expr $numSteps * $dt]
set tCurrent [getTime]

while {$ok == 0 && $tCurrent < $tFinal} {
   if {fmod($tCurrent,1) < 1.0E-16} {
          puts "$i $tCurrent"
       }
       set ok [analyze 1 $dt]

   if {$ok != 0} {
      puts " "
      puts [format "KrylovNewton failed (time = %1.3e), try Newton" $tCurrent]
            algorithm Newton
      test $testType $testTol $testIter 0
      set ok [analyze 1 $dt]
      algorithm $algoType
   }

   if {$ok != 0} {
      puts " "
            puts [format "Newton failed (time = %1.3e), try Newton  w/ iniCurrent" $tCurrent]
            algorithm Newton -initialCurrent
      test $testType $testTol $testIter 0
      set ok [analyze 1 $dt]
      algorithm $algoType
   }

     if {$ok != 0} {
      puts " "
            puts [format "Newton  w/ iniCurrent failed (time = %1.3e), try Newton  w/ ini" $tCurrent]
            algorithm Newton -initial
      test $testType $testTol [expr 500 * $testIter] 0
      set ok [analyze 1 $dt]
      algorithm $algoType
      test $testType $testTol $testIter 0
```

Define analysis case: new analysis case

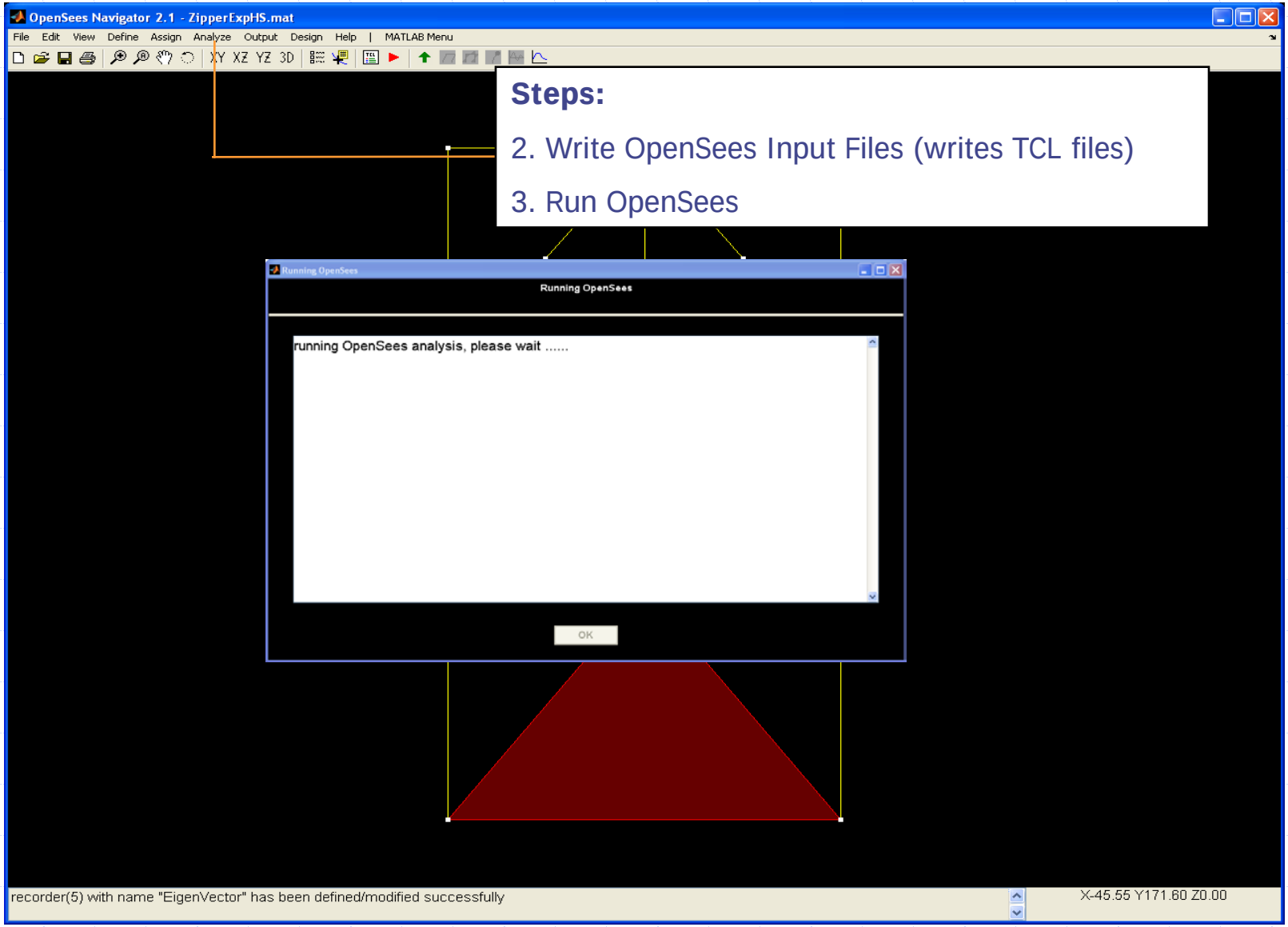


Define analysis case: new analysis case

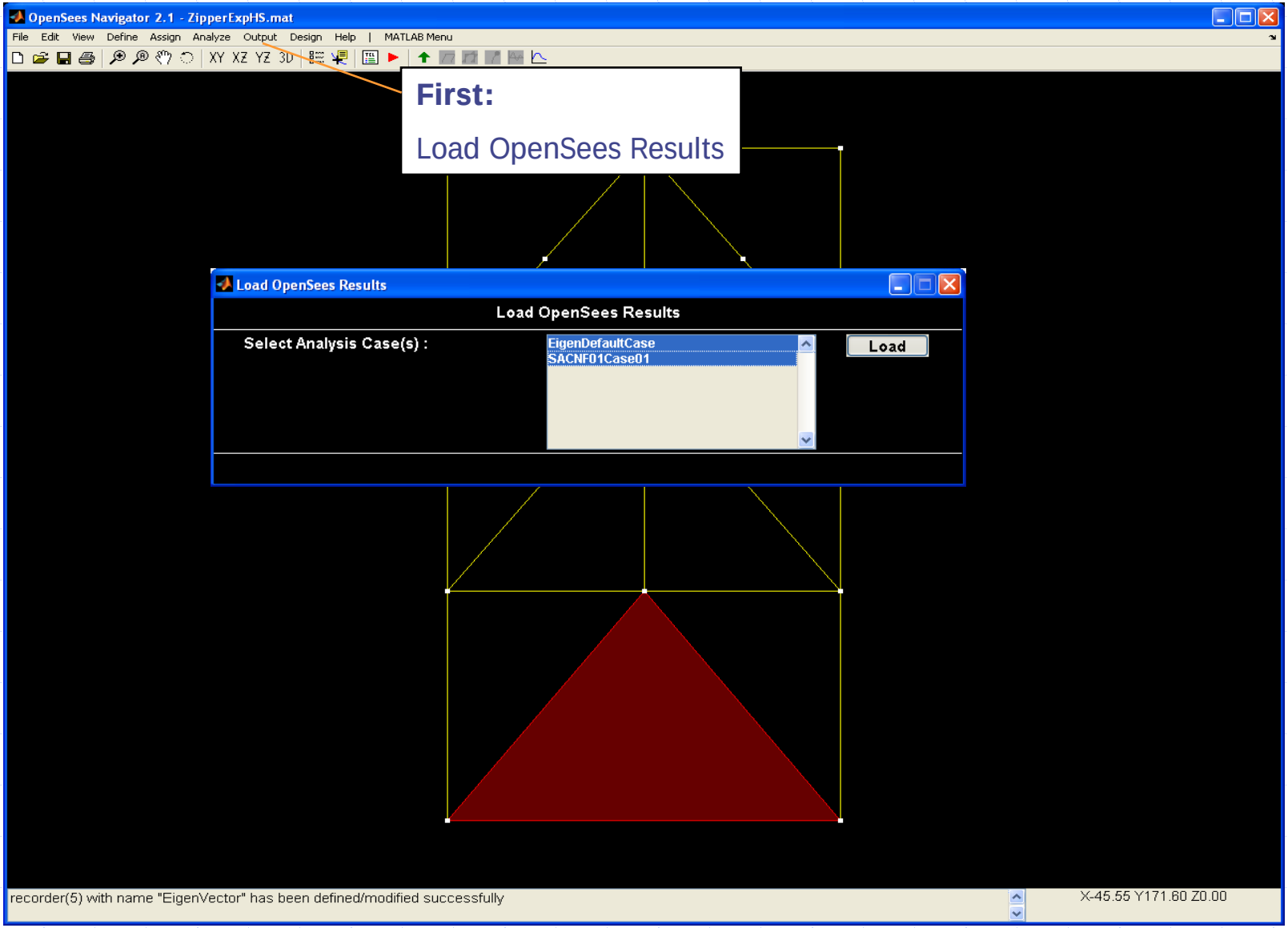
Run OpenSees: set OpenSees.exe path



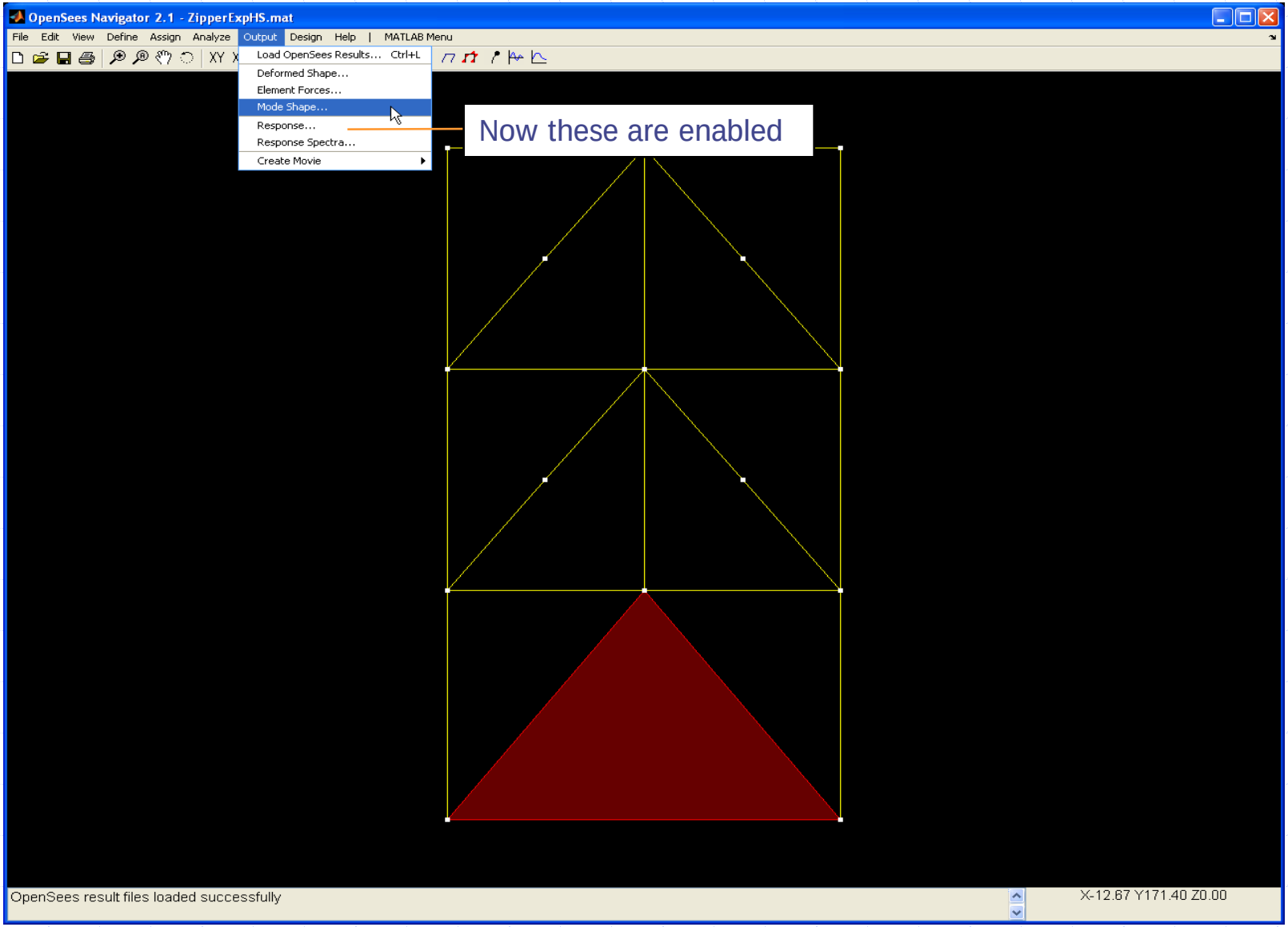
Run OpenSees: write TCL files



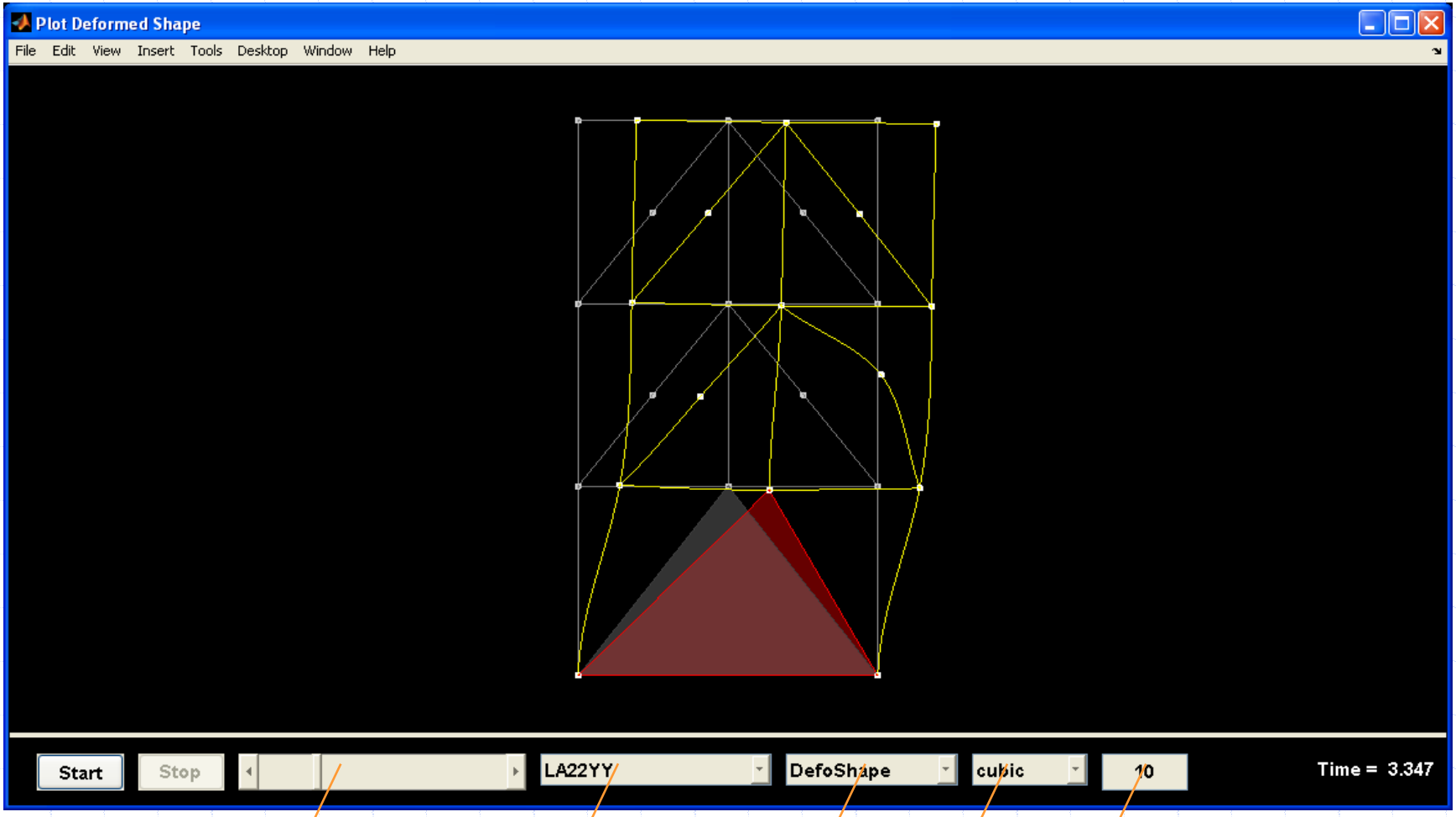
Post processing: load results



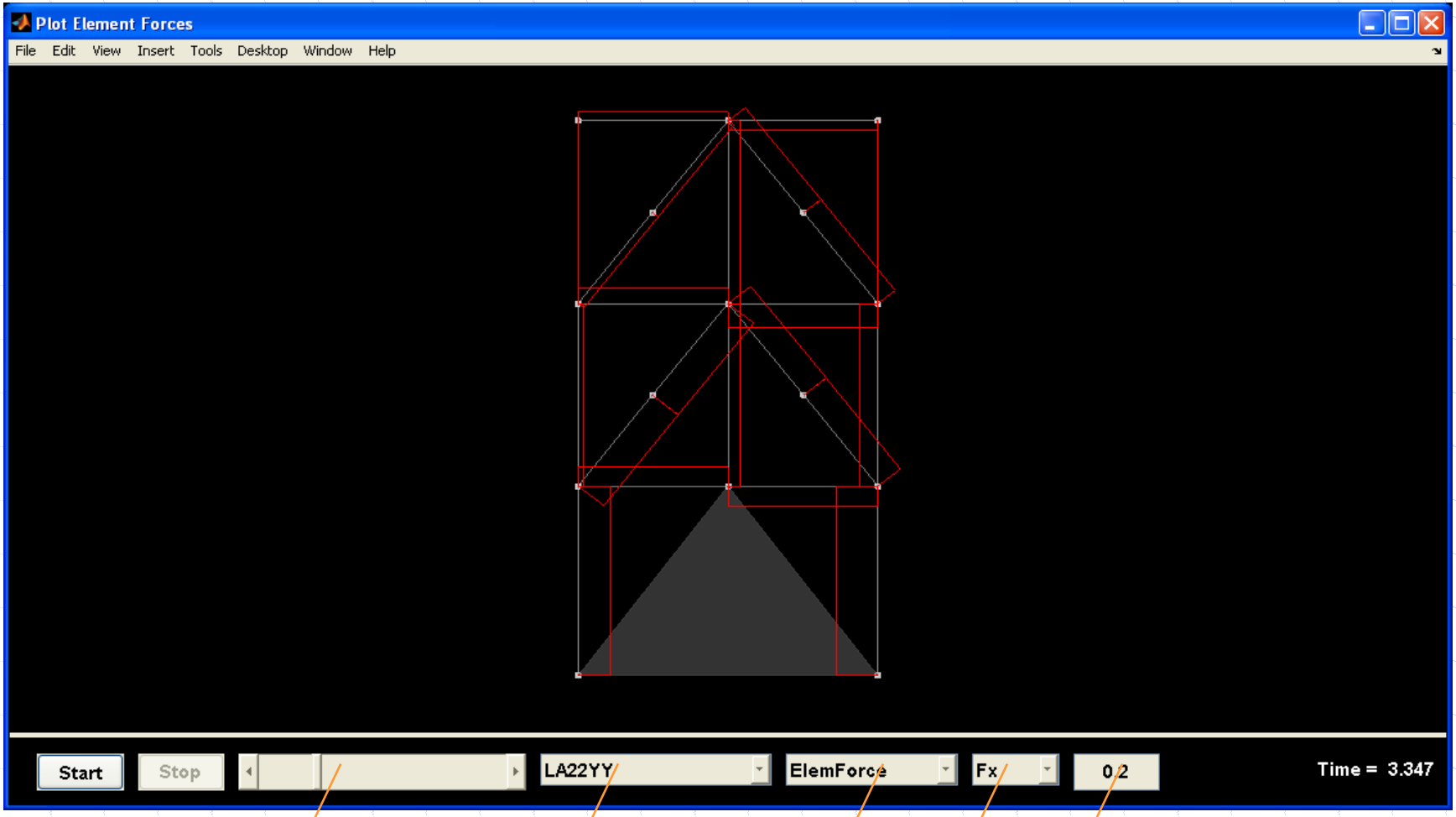
Post processing: output



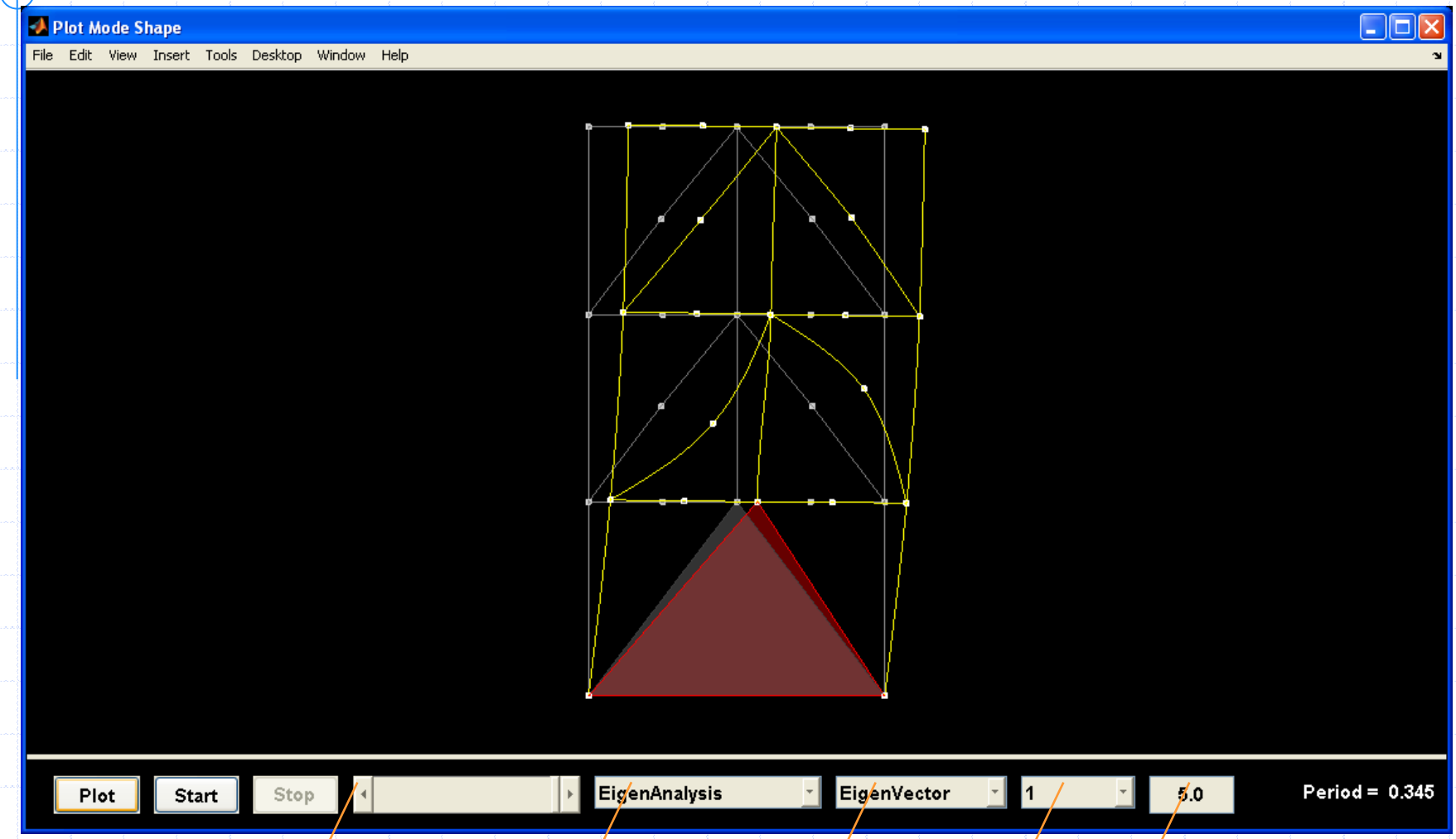
Post processing: plot deformed shape



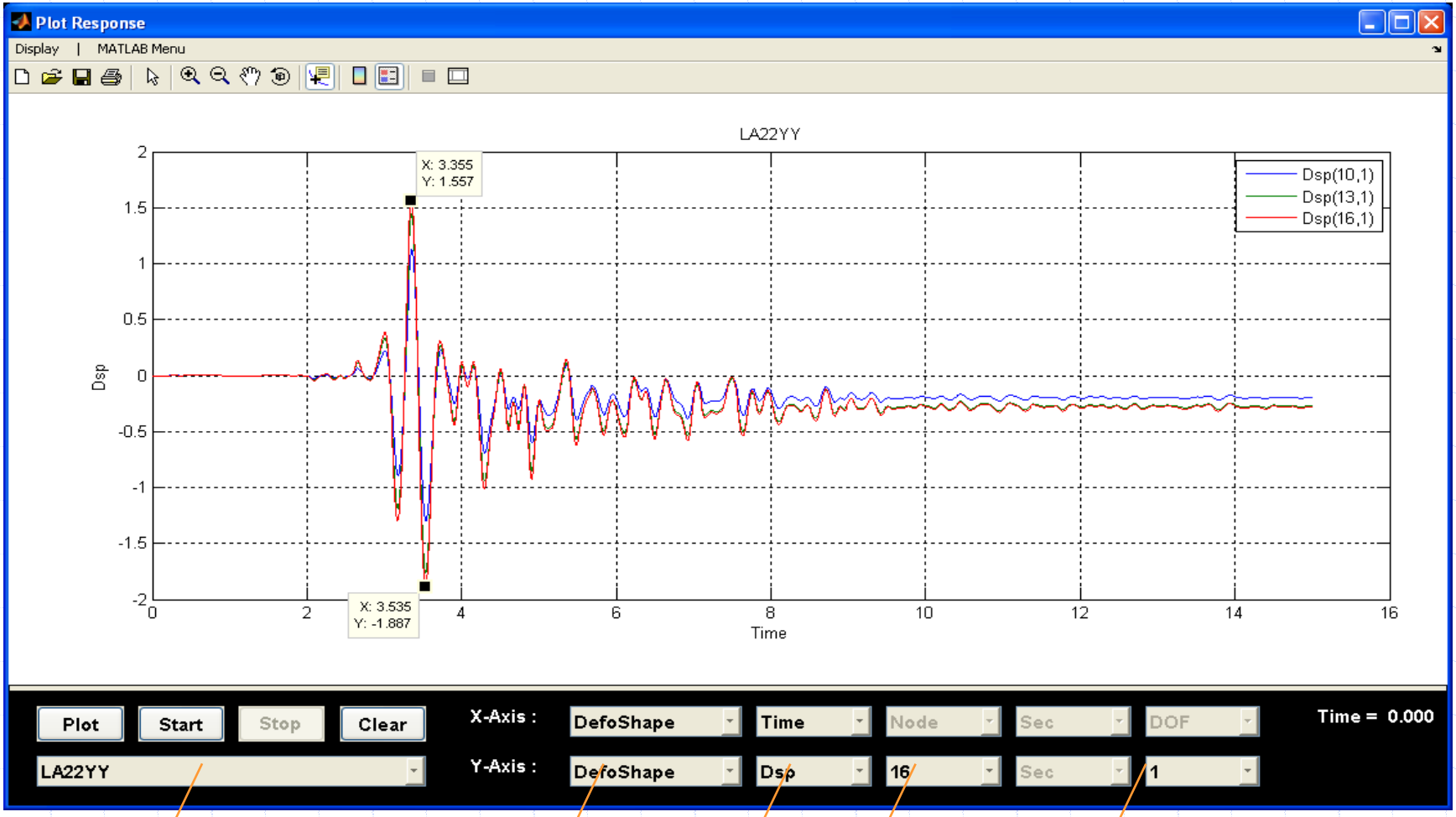
Post processing: plot element forces



Post processing: plot mode shape



Post processing: plot response histories



AnalysisCase

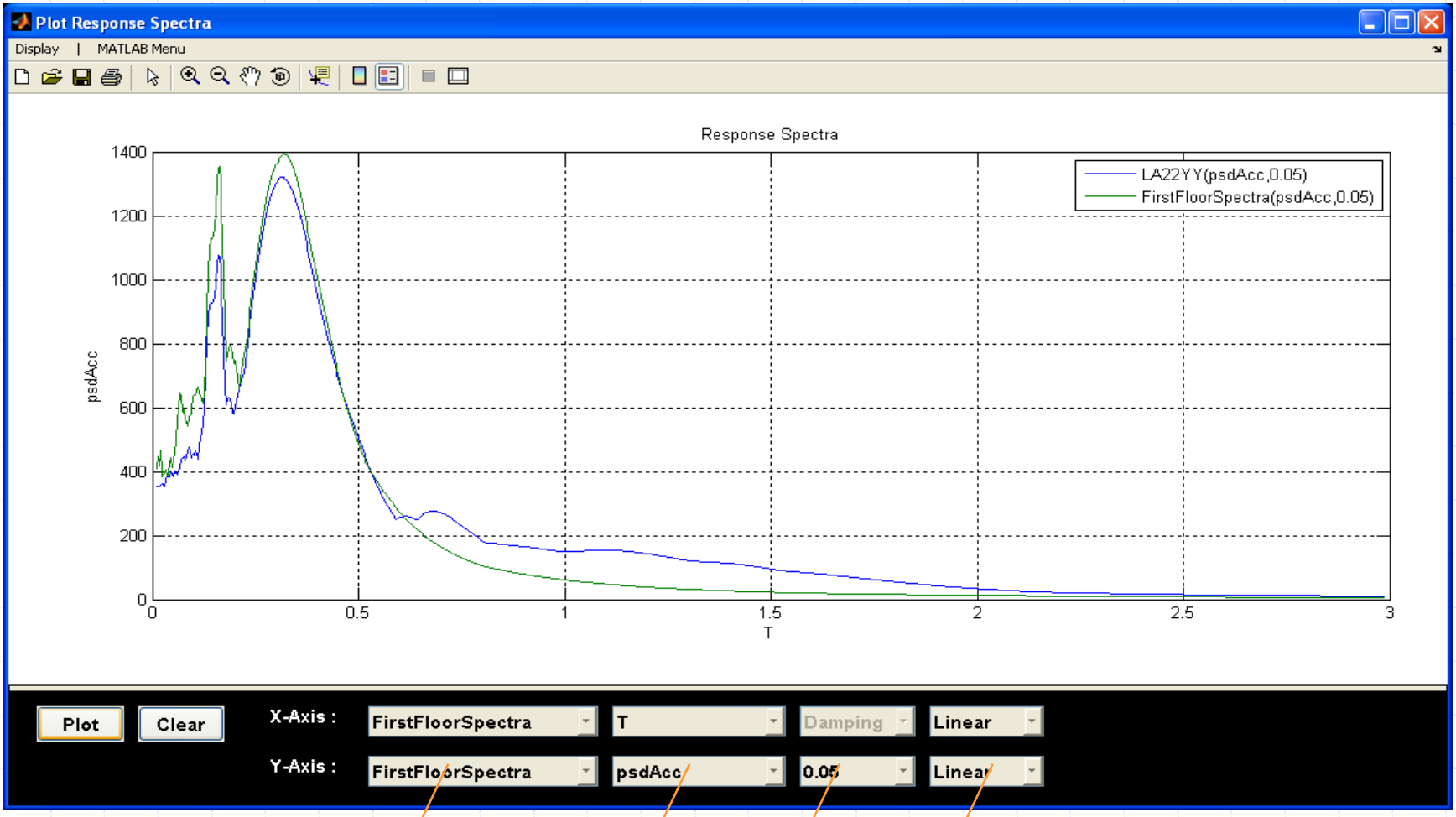
Recorder

Parameter

Node/Element

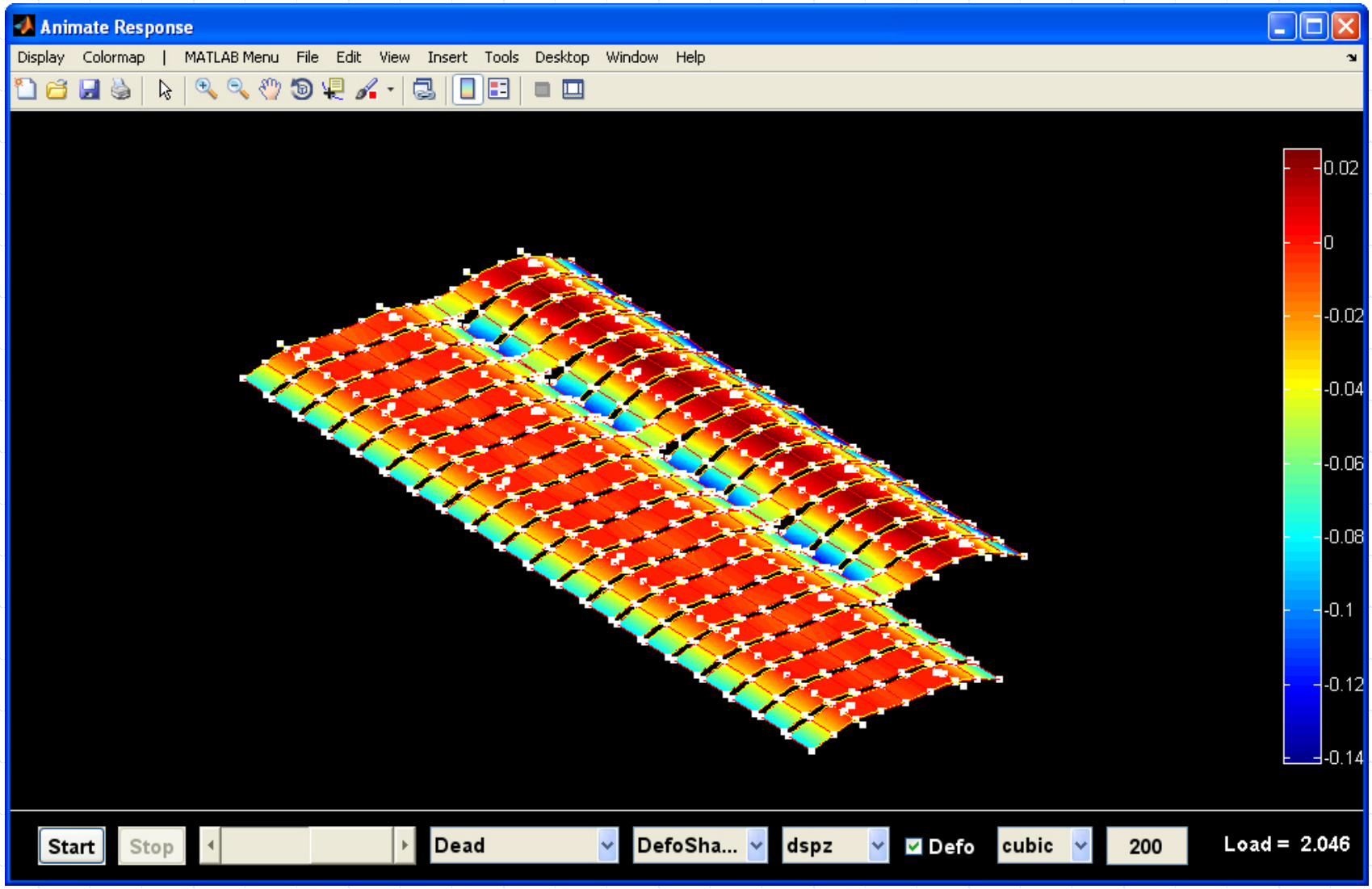
DOF

Post processing: plot response spectra

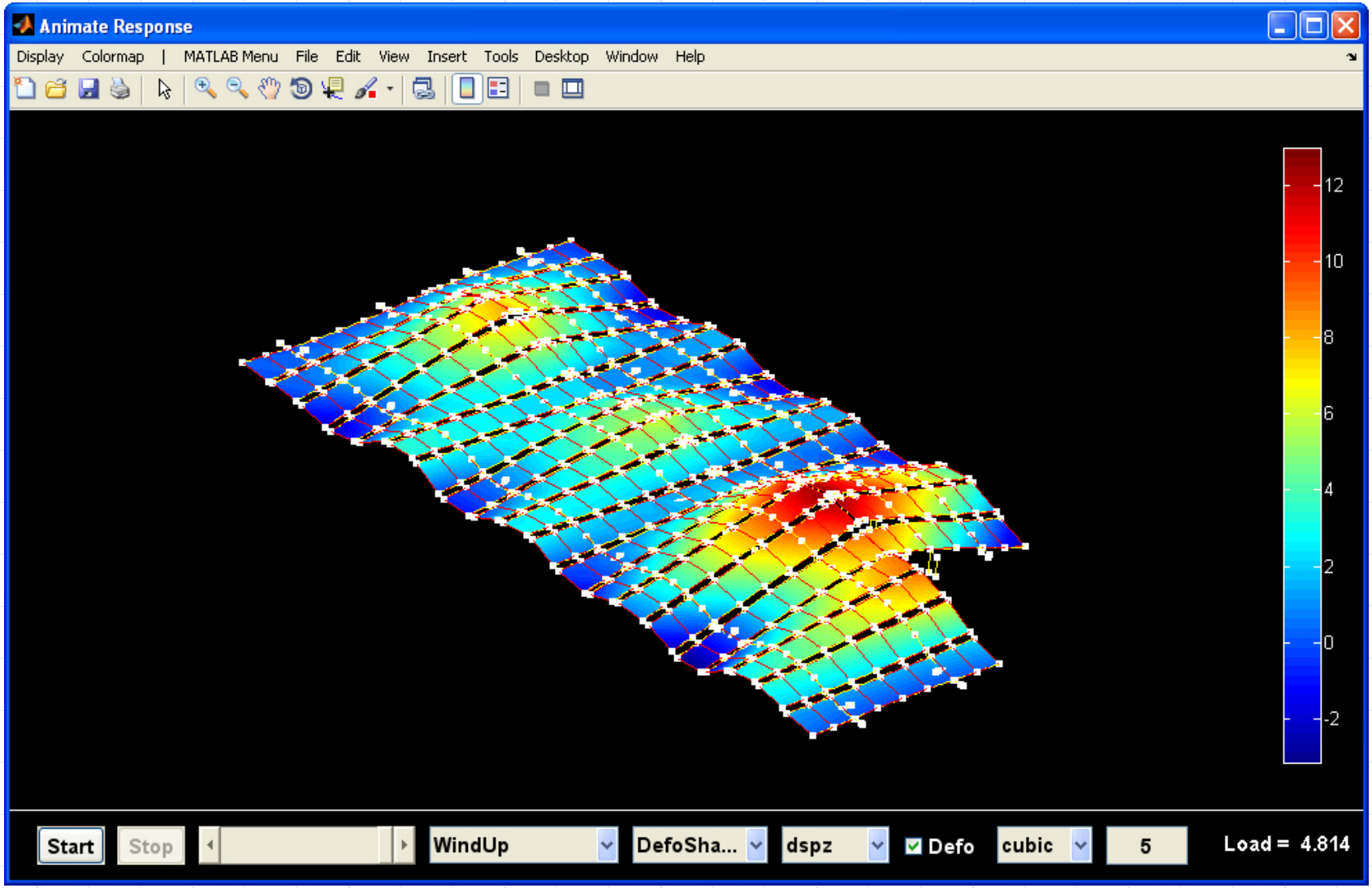


Spectra Name Response Damping Axis Scale

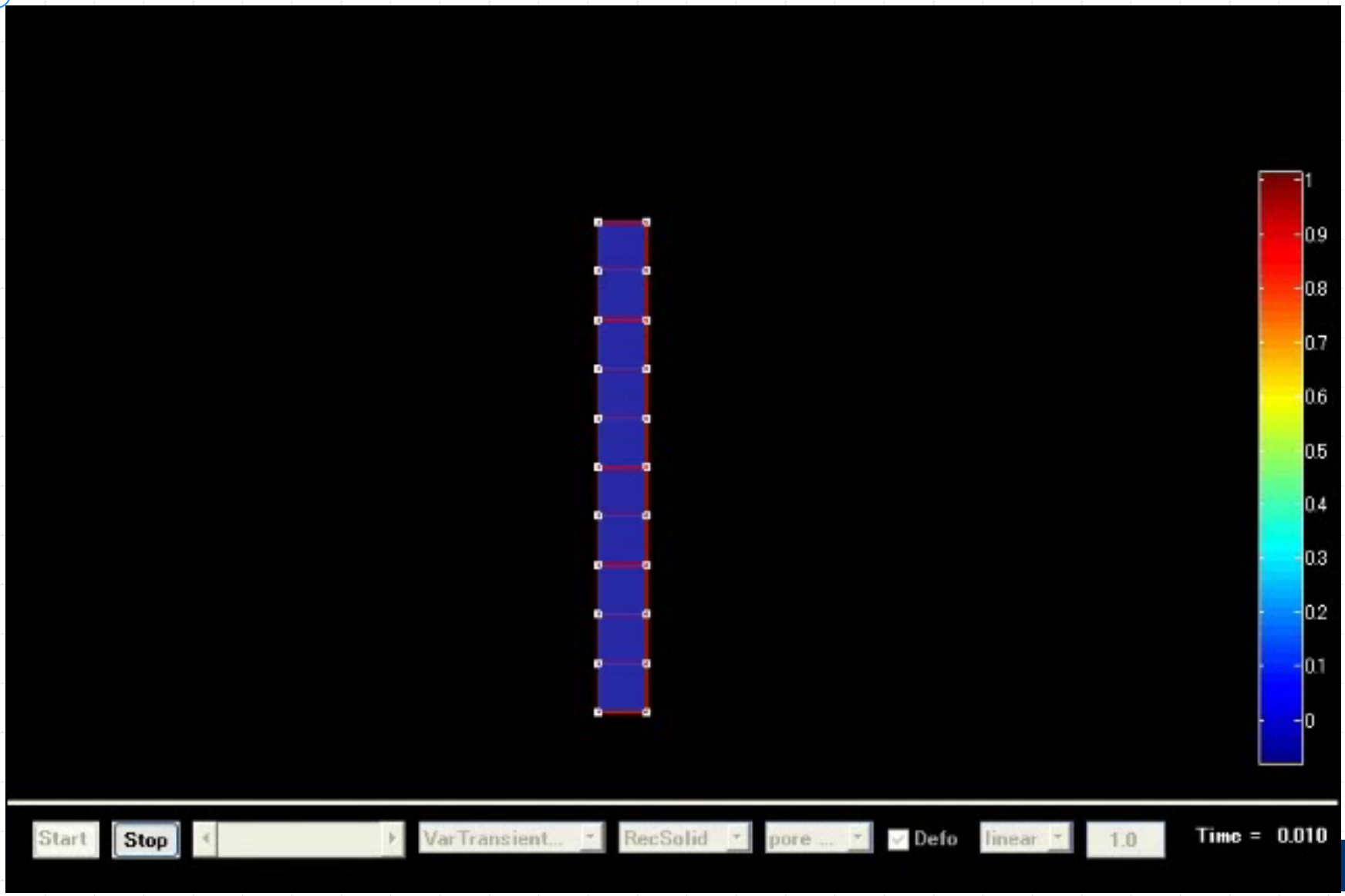
Post processing: animate response



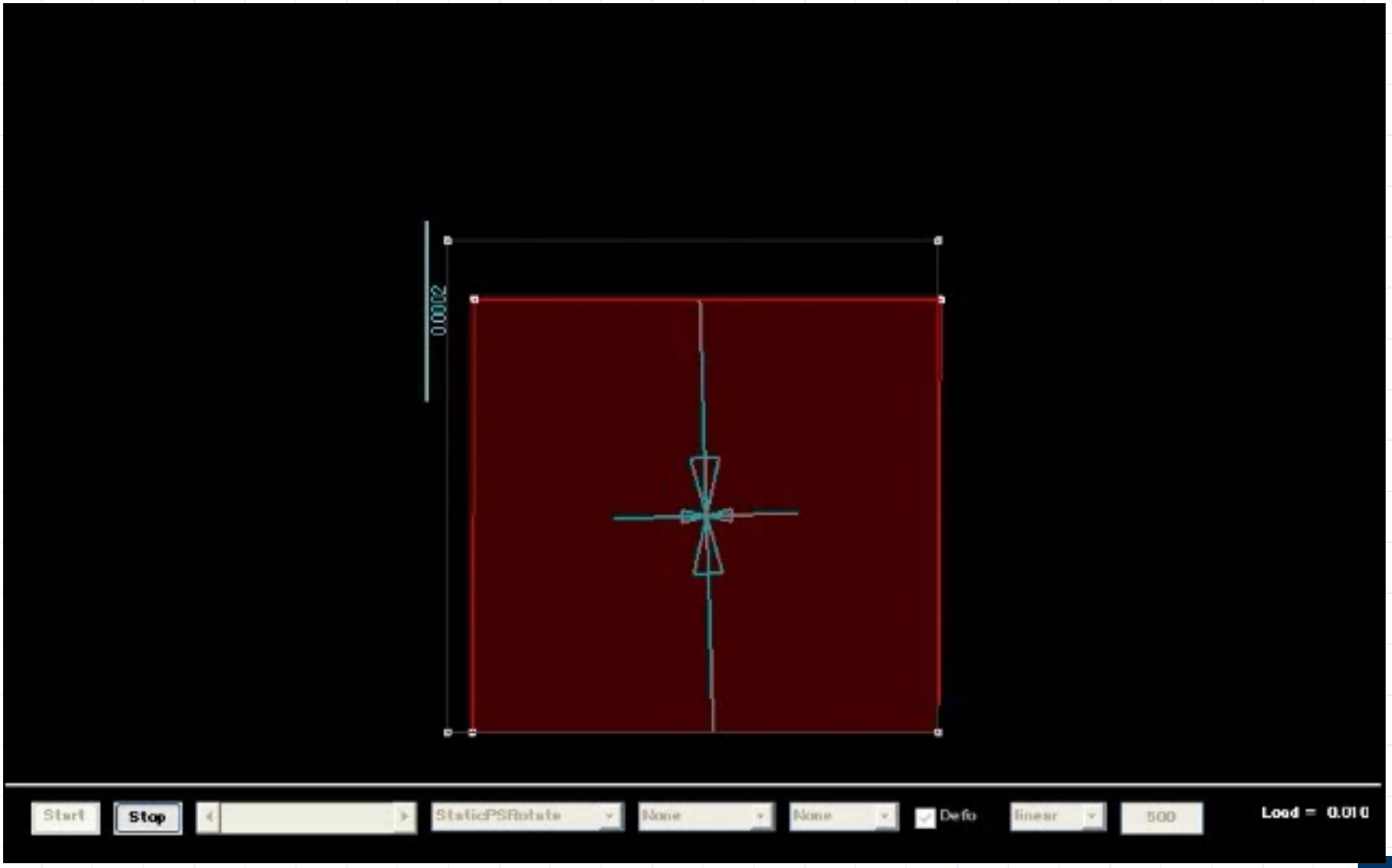
Post processing: animate response



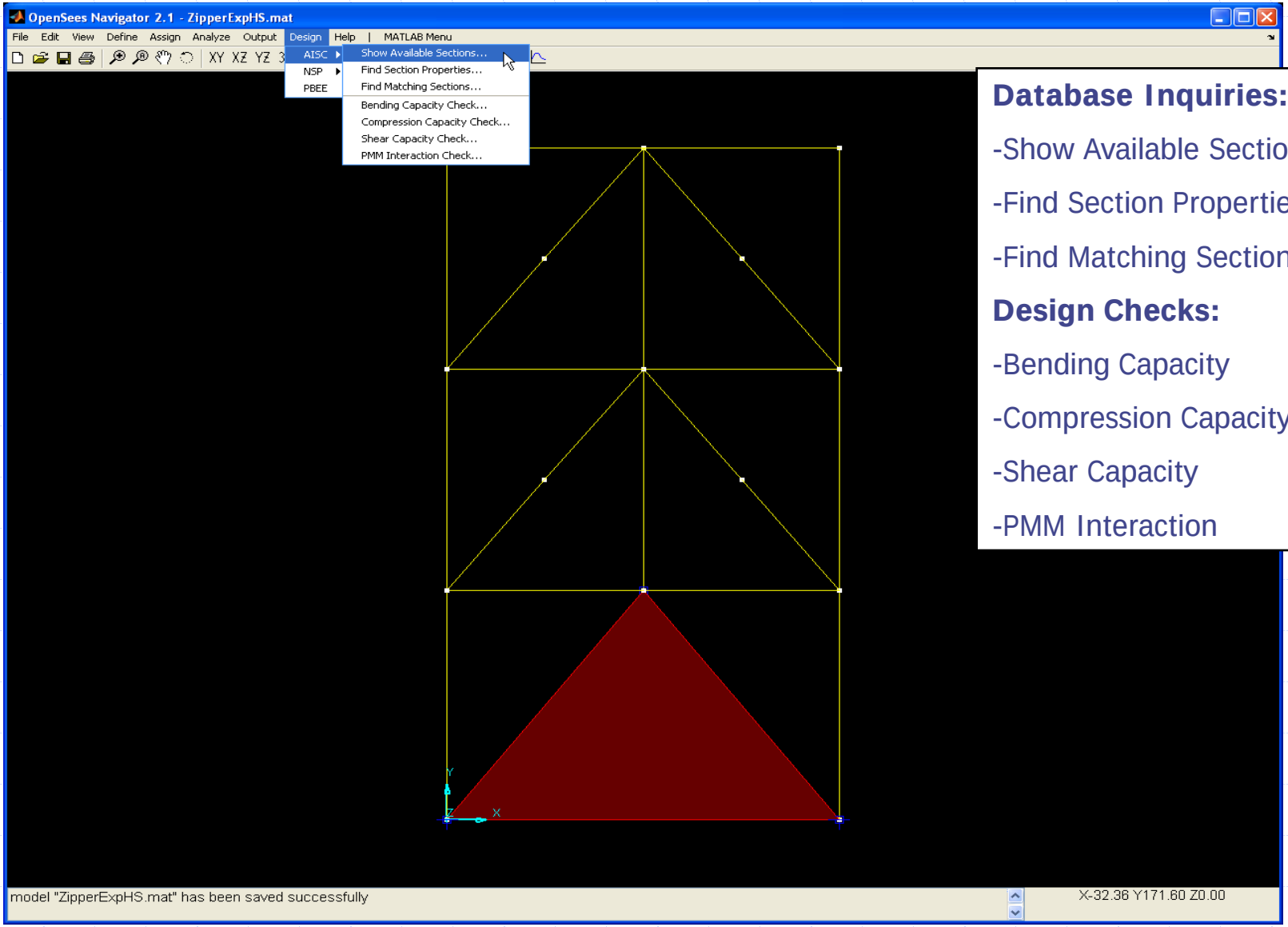
Post processing: animate response



Post processing: principal σ and ε



Design: AISC design toolbox



Database Inquiries:

-Show Available Sections

-Find Section Properties

-Find Matching Sections

Design Checks:

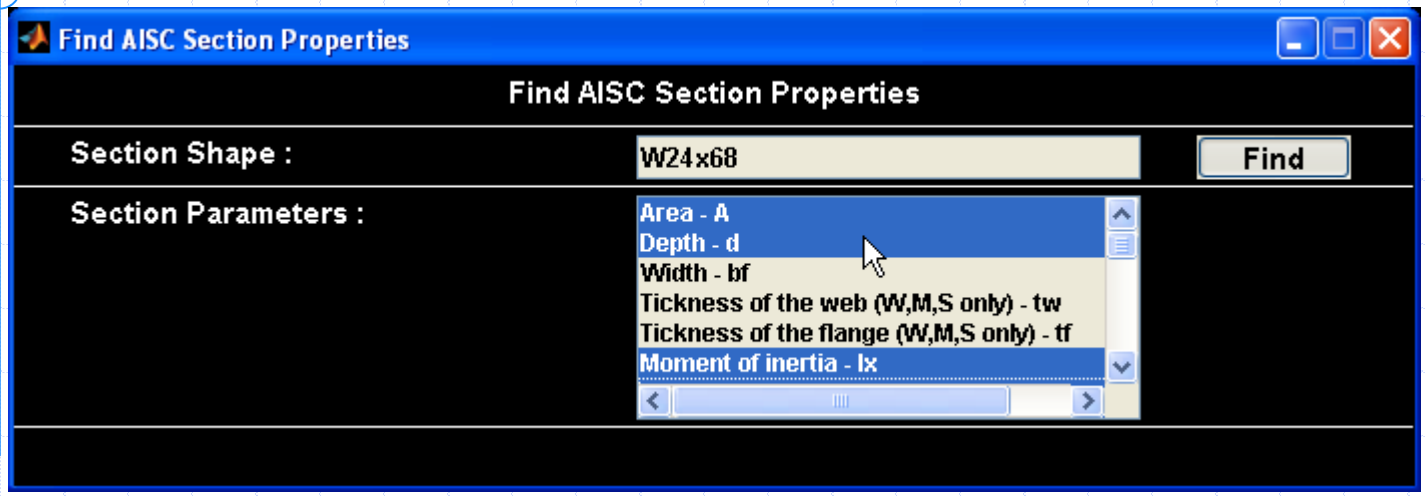
-Bending Capacity

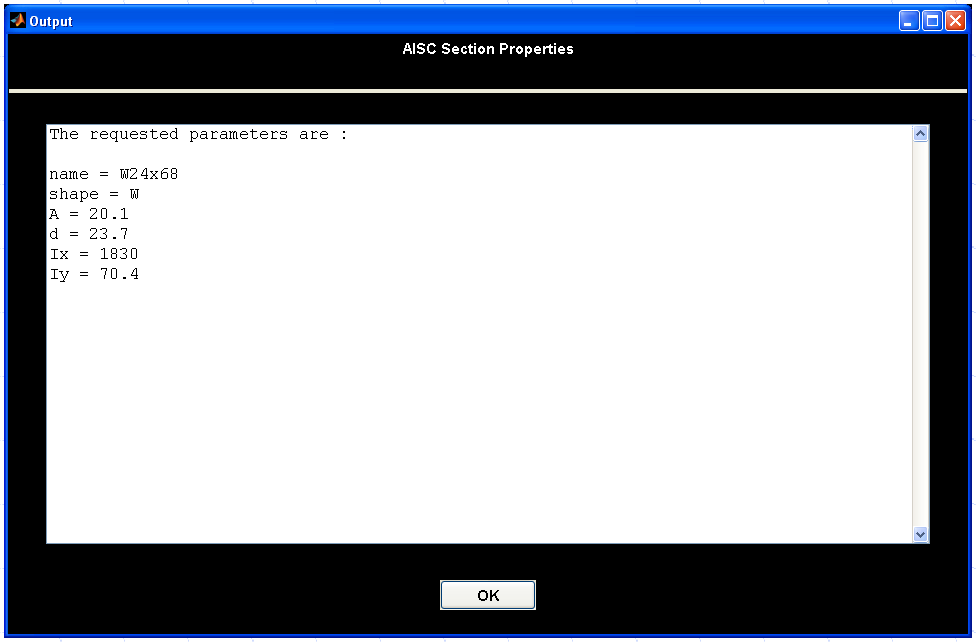
-Compression Capacity

-Shear Capacity

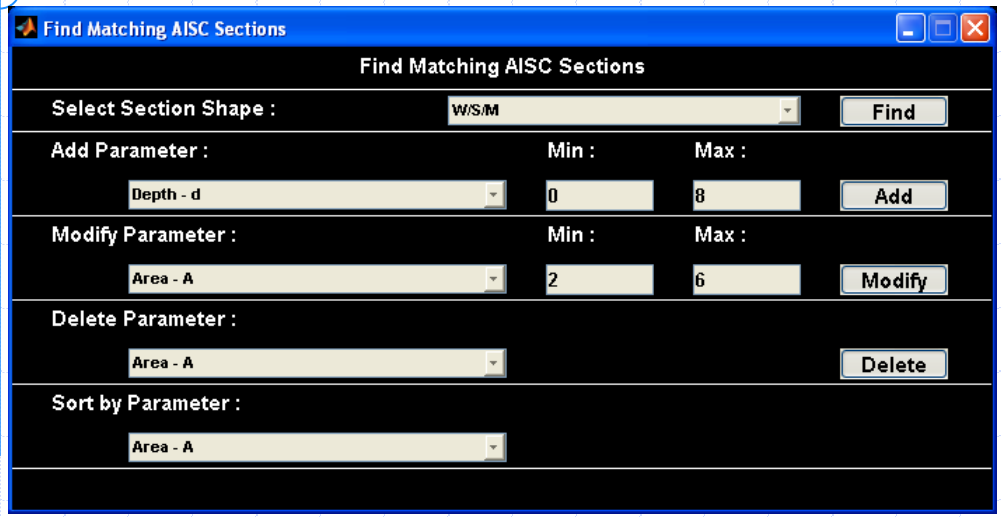
-PMM Interaction

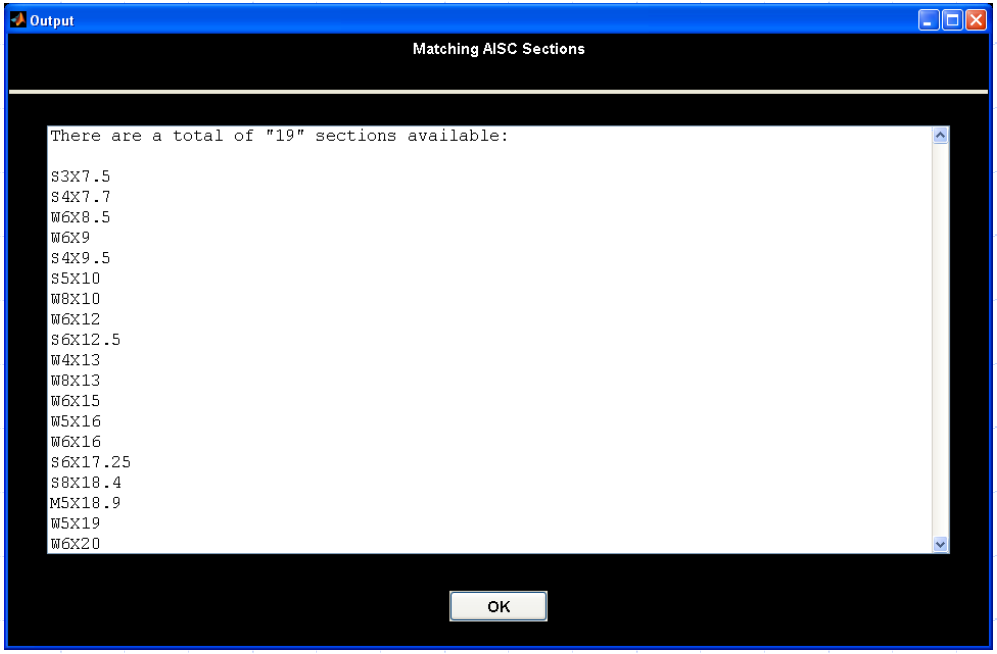
AISC design toolbox: section properties



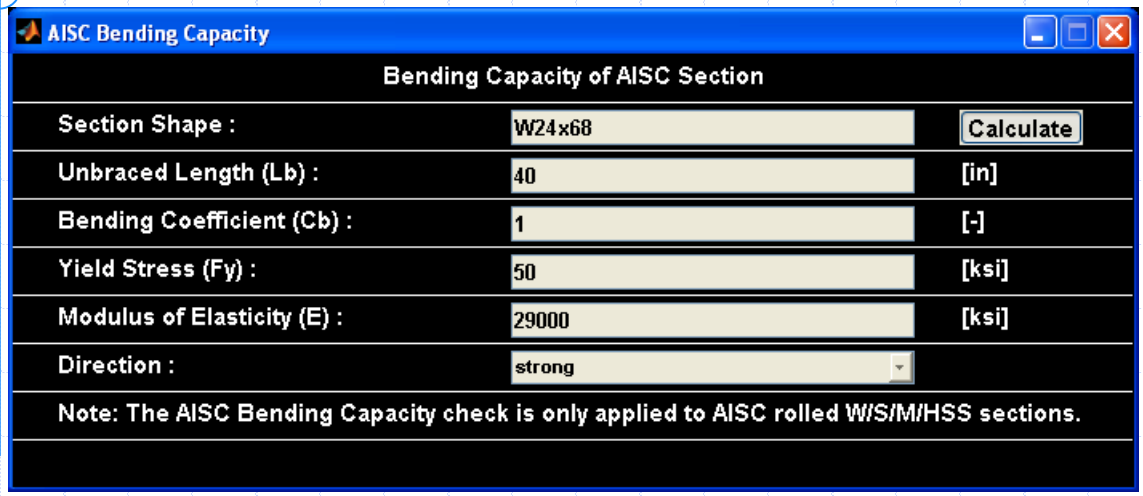


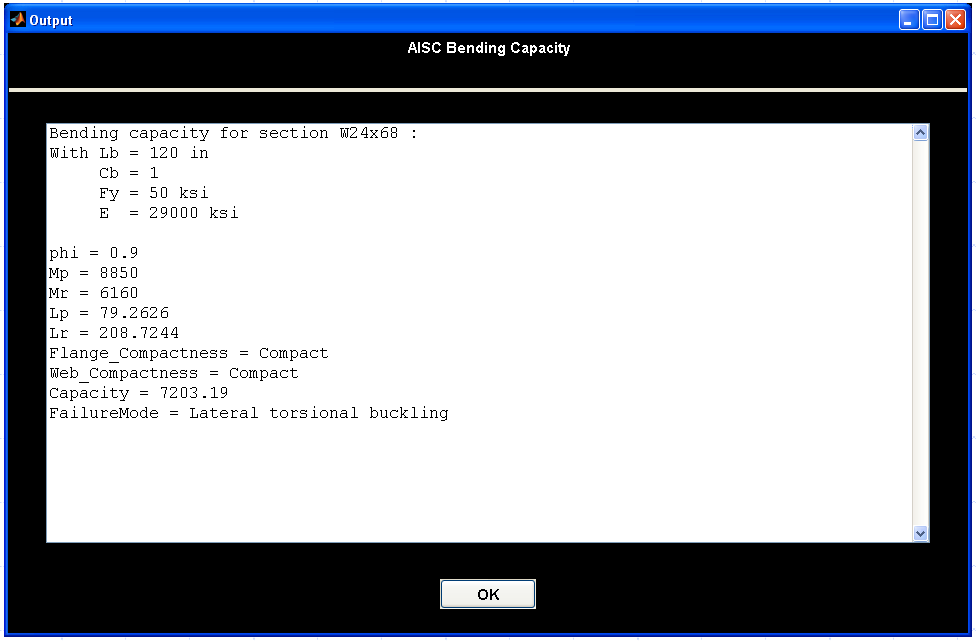
AISC design toolbox: matching sections



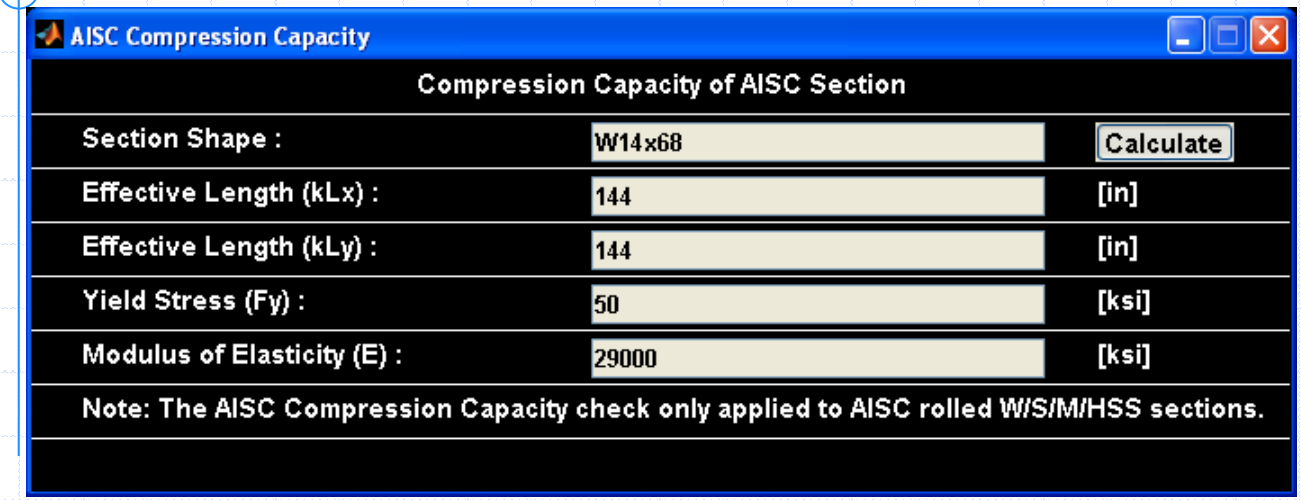


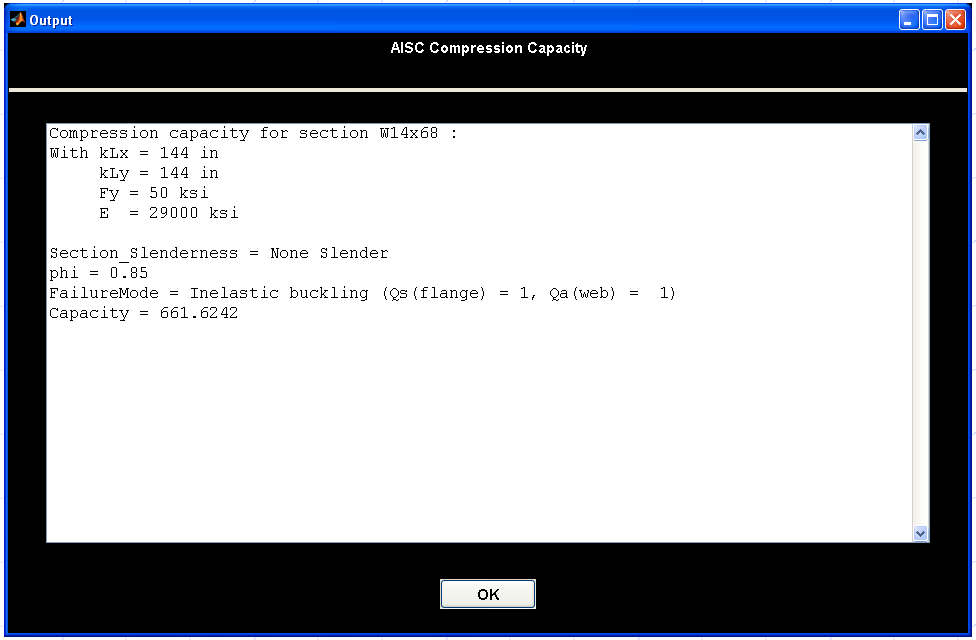
AISC design toolbox: bending capacity



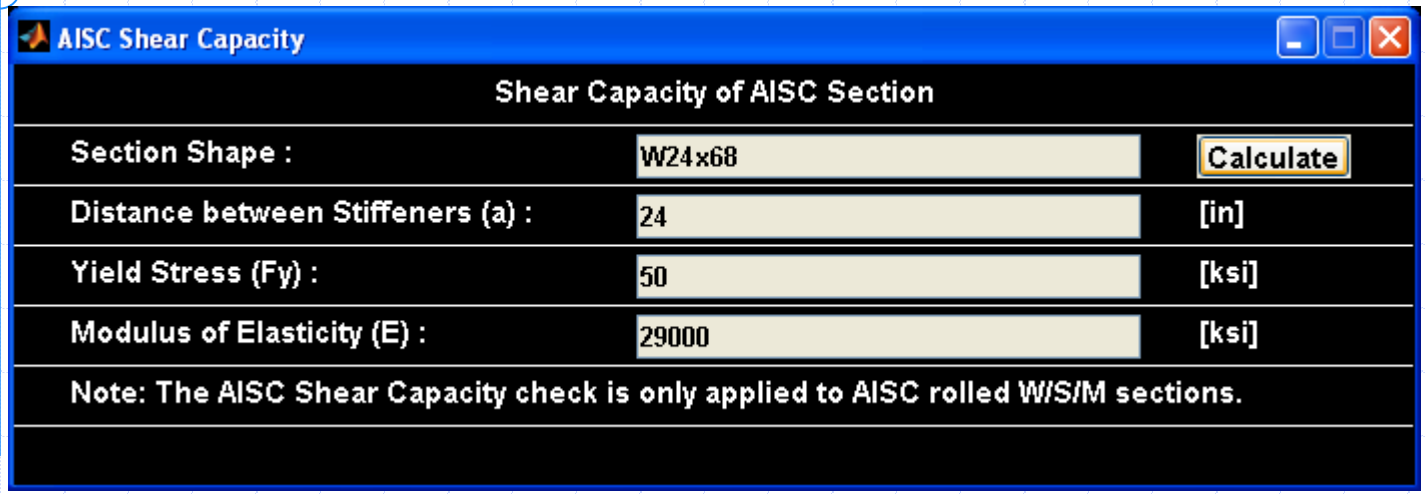


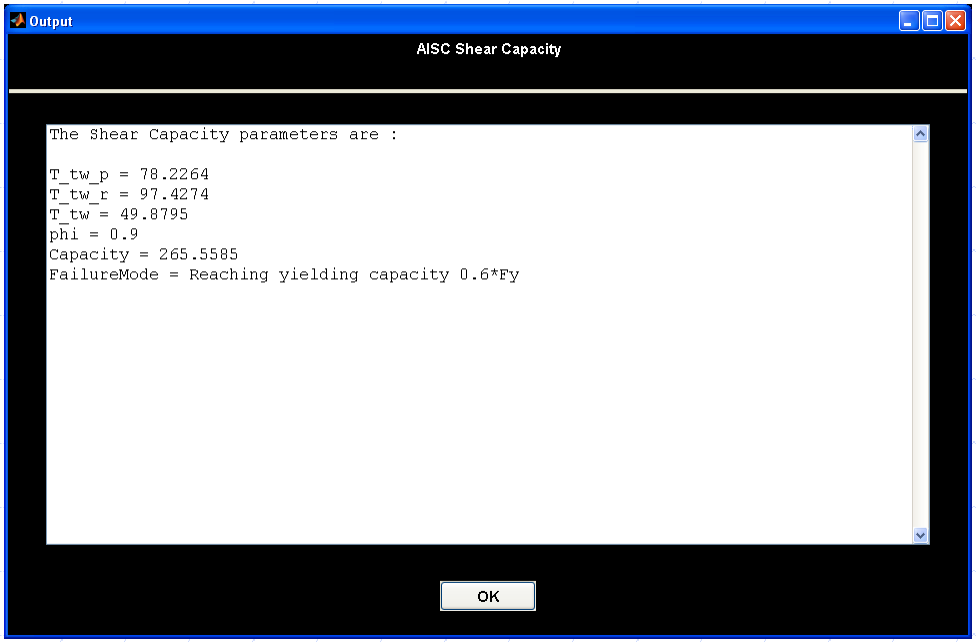
AISC design toolbox: compression cap.



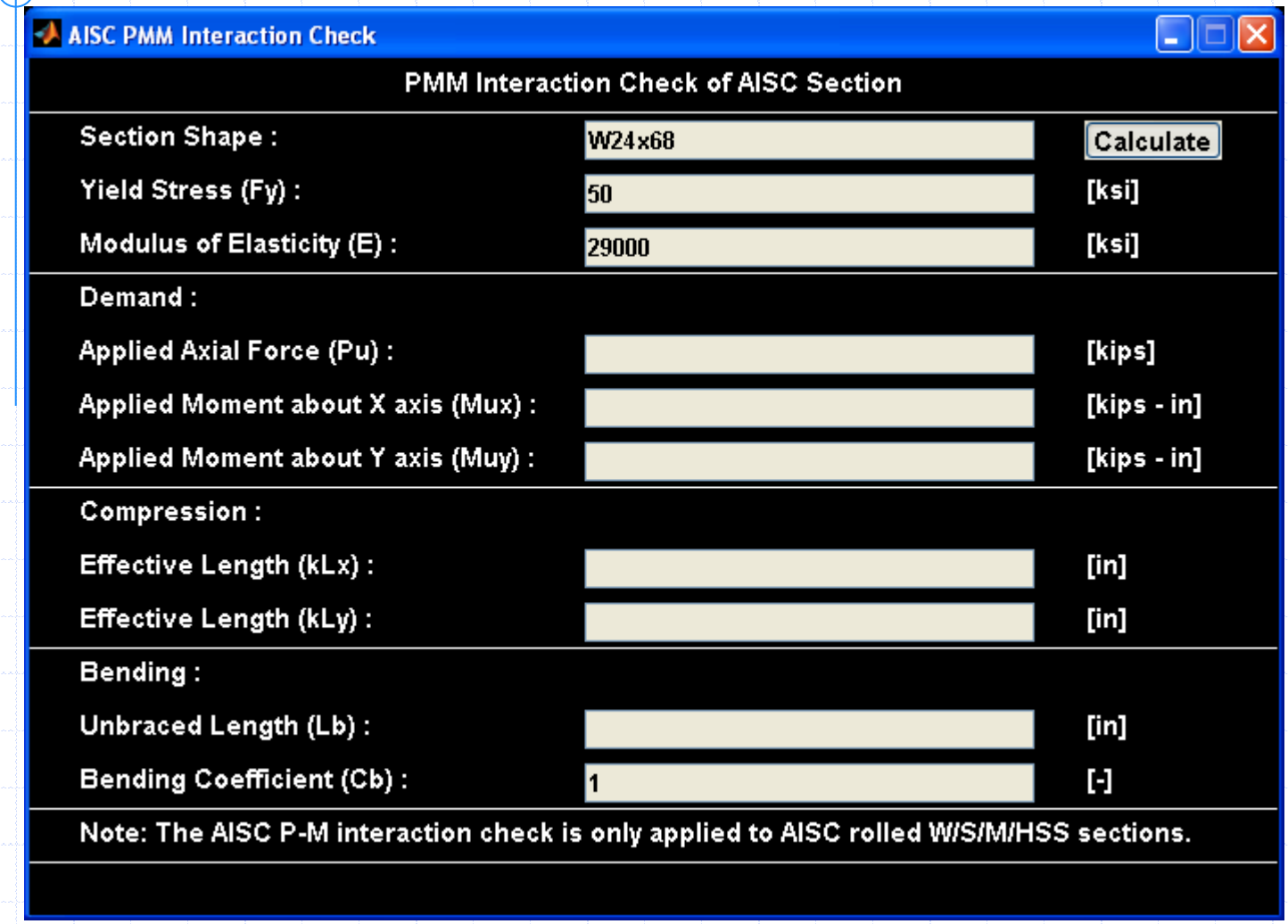


AISC design toolbox: shear capacity

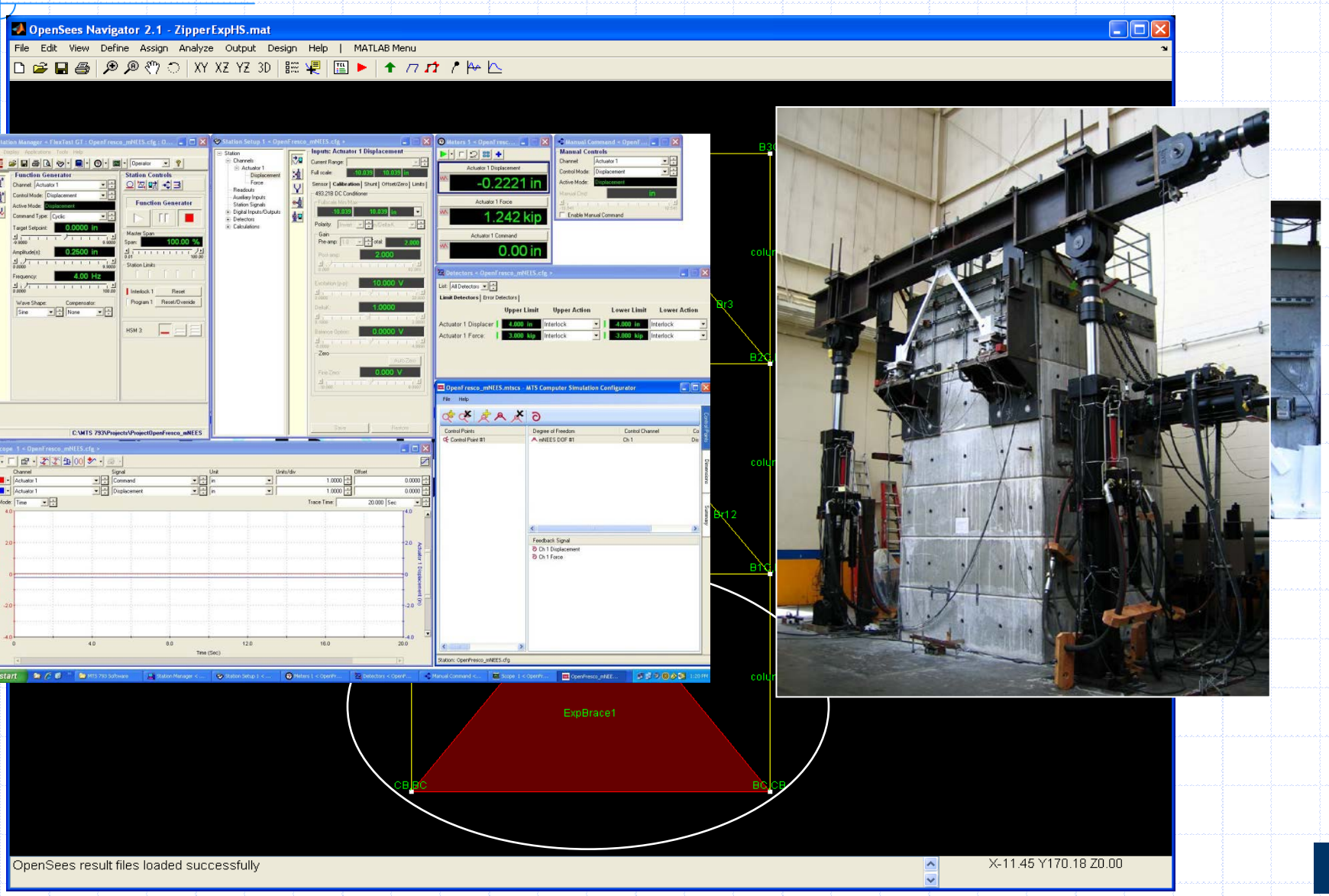




AISC design toolbox: PMM interaction



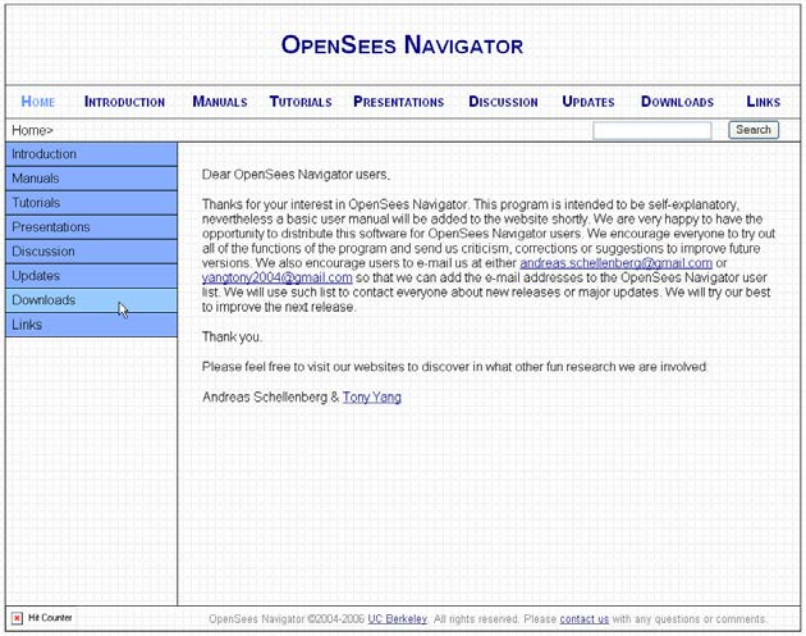
OpenSees Navigator – Hybrid simulation

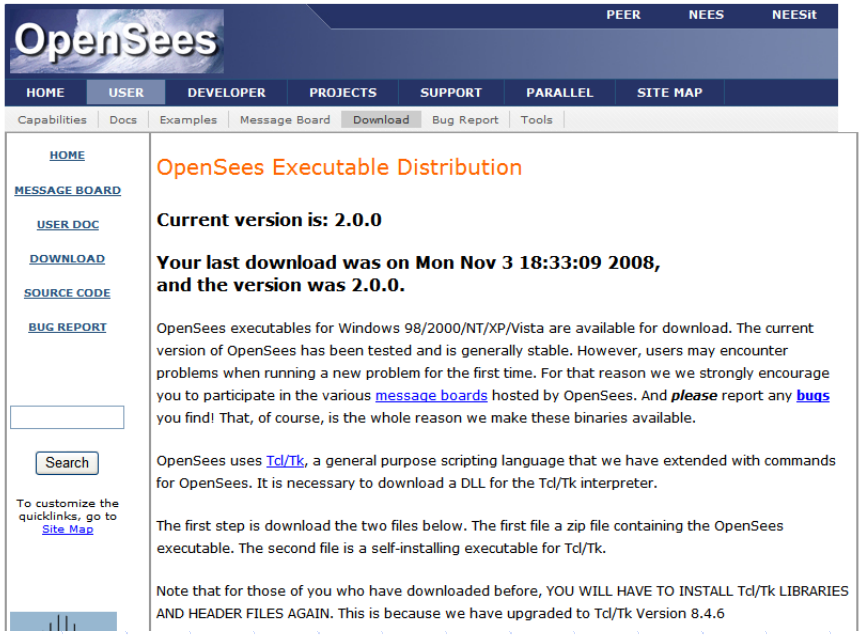


Summary and conclusions

- OpenSees
  - A powerful software framework which is capable for conducting sequential and parallel, grid-enabled finite element applications.
- Hybrid simulation interface (OpenFresco)
- OpenSees Navigator
  - Flexible and user friendly graphical user interface.
  - Great tool to visualize structural behavior.
  - Easy way to study material, section, element or system behavior.
  - Many design toolboxes: NSP, PBEE, AISC design checks, AISC database, response spectra for linear and bilinear systems and signal filtering.
- Both MATLAB Pcode and self-executable versions are available

Websites:





Download information:

OpenSees 2.2 is available at

http://opensees.berkeley.edu

OpenSees Navigator 2.5 is available at

http://peer.berkeley.edu/OpenSeesNavigator

Thank you for your attention!

Contact information:
Tony Yang, yang@civil.ubc.ca,
http://peer.berkeley.edu/~yang

Andreas Schellenberg, andreas.schellenberg@gmail.com

Eiji Kohama, ekohama@gmail.com