16th World Conference on Earthquake Engineering, 16WCEE 2017

Santiago Chile, January 9th to 13th 2017

Paper N°3024

Registration Code: S-W1464753446

SEISMICAE – A COMPUTER-AIDED TOOL FOR PERFORMANCE AND RISK ASSESSMENT OF BUILDING STRUCTURES USING OPENSEES

S. Mazzoni[(1)]

[(1)] Consultant & Project Scientist, PEER, University of California Berkeley, mazzoni@berkeley.edu

Abstract

OpenSees is a numerical-simulation framework capable of modeling large structural and geotechnical systems as well as easily manage large number of simulations of such structures. Structural engineers in the profession, however, have been reluctant to use it because of its lack of an integrated graphical user interface (GUI). While visualization via a GUI is critical in building and validating a numerical model, it limits the efficiency of building complex models. SeismiCAE is Computer-Aided Engineering tool with a user interface (UI) developed to address these limitations by integrating a graphical interface (useful for visualization) with a scripting interface for numerical simulation in OpenSees. The scripting interface is critical in enabling the user to include programming features into the building of the model and analysis procedures. The programming features, such as logical statements and array manipulation, allow more control in the generation of multiple variation of a model, ease in modifying the model, and efficiency in running large numbers of simulations. In addition, SeismiCAE includes a library of commands that simplify the model input by using a format that is more intuitive to an engineer than the basic OpenSees commands. Because SeismiCAE runs OpenSees in the background, model-building and simulation runs can be executed interactively or via scripting models. The structural response can be visualized in real time during the analysis. The post-processing features of SeismiCAE also allow interactive visualization of results. Because of its ability to handle large numbers of simulations, a risk-analysis module is currently being developed for SeismiCAE to handle Risk-Assessment analyses.

Keywords: Numerical Simulation; OpenSees; SeismiCAE; OpenSees GUI

1. Introduction

OpenSees is a numerical-simulation framework capable of modeling large structural and geotechnical systems as well as easily manage large number of simulations of such structures [1]. Structural engineers in the profession, however, have been reluctant to use it because of its lack of an integrated graphical user interface (GUI). While visualization via a GUI is critical in building and validating a numerical model, it limits the efficiency of building complex models.

SeismiCAE is Computer-Aided Engineering tool with a user interface (UI) developed to address these limitations by integrating a graphical interface (useful for visualization) with a scripting interface for numerical simulation in OpenSees. The scripting interface is critical in enabling the user to include programming features into the building of the model and analysis procedures. The programming features, such as logical statements and array manipulation, allow more control in the generation of multiple variation of a model, ease in modifying the model, and efficiency in running large numbers of simulations.

In addition, SeismiCAE includes a library of commands that simplify the model input by using a format that is more intuitive to an engineer than the basic OpenSees commands. Because SeismiCAE runs OpenSees in the background, model-building and simulation runs can be executed interactively or via scripting models. The structural response can be visualized in real time during the analysis. The post-processing features of SeismiCAE also allow interactive visualization of results. Because of its ability to handle large numbers of simulations, a risk-analysis module is currently being developed for SeismiCAE to handle Risk-Assessment analyses.

2. SeismiCAE Features

SeismiCAE is a library of Tcl procedures (commands) used to build a database for numerical simulation of building frames. Structural materials, sections, elements, models, analyses, loads and load combinations created in this database. Structural models can be Elevations, Plans, or 3DModels, which combine elevations and plans. Procedures are included to perform the numerical simulation using OpenSees. The OpenSees recorder data is post-processed into more accessible format.

There following are the main objectives to the SeismiCAE program:

- Generate numerical-simulation input in a manner consistent with architectural/structural drawings
- Interchangeable User Interface: scripting and graphical
- Programmable input file via Tcl script commands
- Direct integration with OpenSees – run OpenSees real-time
- Create a database of all structure and simulation data
- Maintain flexibility and power of OpenSees while keeping it simple
- Be able to run on a number of platforms, just as OpenSees can

SeismicCAE relies on two types of User Interface: the scripting interface and the graphical viewer. The following are the key features of the scripting interface:

- Generate building-model data
    - Materials
    - Sections
    - Element Types
    - Analysis Models

- Loads (Gravity & Lateral)
- Load Combinations
- Models
  - Elevations
  - Grids
  - 3D Frames

- Generate analysis-model data
- Generate loading and load-combination data
- Generate OpenSees model of building
- Perform OpenSees numerical simulations
- Post-Process OpenSees recorder output into formatted data
- Generate OpenSees input files

The graphical user interface has the same capabilities as the scripting interface, with the exception of the ability to program the input by using logical statements. The following are the key features of the viewer:

- Graphical User Interface (GUI)
- generate and/or visualize ALL input graphically
- Save ALL input into script
- Perform numerical simulations using OpenSees interactively
- Visualize OpenSees simulation real-time
- Visualize simulation results interactively
- Export simulation results

3. SeismiCAE Language

By using the procedure features of TCL, SesimiCAE specific commands have been developed to enable an efficient and effective language to build models, define analyses, run these analyses, post-process the results, and display these results. These commands can be integrated with the TCL scripting language, which uses arrays and logical functions, to create an efficient and expandable input file.

Examples of these additional commands are shown in the figures 1-4 below.

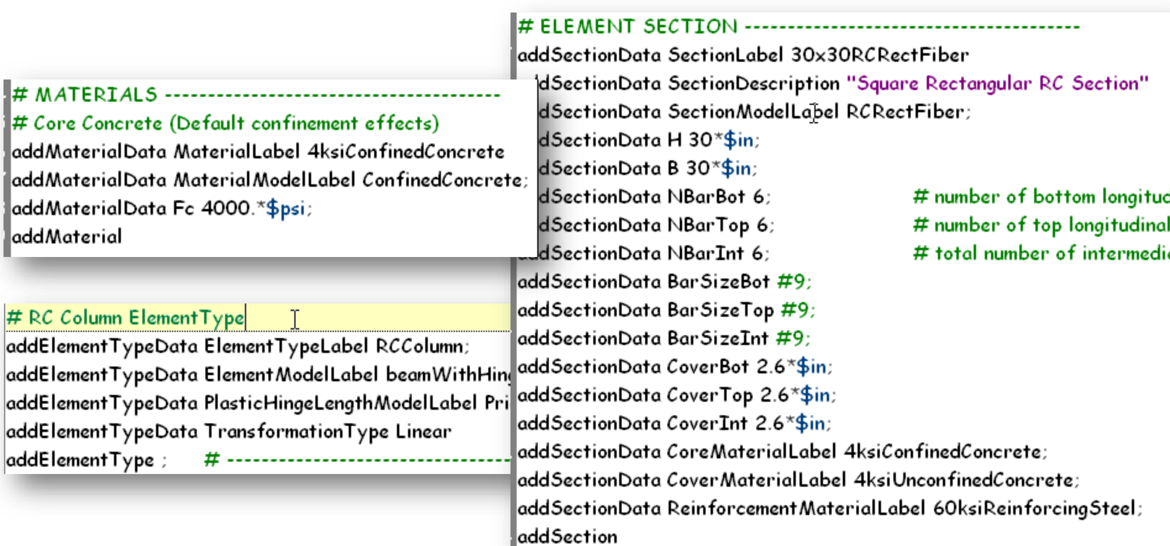
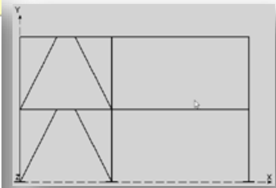


Fig 1. Material, Section & Element-Building Commands

```
# Elevation
addModelData ModelLabel RCTestFrame2Story2BayBbraced
addModelData ModelDescription "RC MRF, 2-Story, 2-Bay"
addModelData ModelTypeLabel Elevation
addModelData -Geometry Height 16*\$ft StoryRange "1 2"
addModelData -Geometry Width 20*\$ft Bay 1
addModelData -Geometry Width 30*\$ft Bay 2
addModelData -Columns  SectionLabel 30x30RCRectangularFiber ColumnLineRange "1 3" StoryRange "1 2" Orient Rotated
addModelData -Beams  SectionLabel 30x60RCRectangularFiber BayRange "1 2" FloorRange "2 3"
addModelData -ChevronBraces SectionLabel W12x16 BayRange "1 1" Story "1 2"  Eccentricity 4.*\$ft
addModelData SupportBC fix
addModelData OutOfPlaneSupportBC pin
addModelData RigidFloor Off
addModelData JointOffsetsSwitch on
addModelData TributaryWidth 5.*\$ft
addModelData -GravityLoad LoadLabel DL1  FloorRange "2 3" DistributedLoad 100.*\$psf
addModelData -GravityLoad LoadLabel LL1  FloorRange "2 3" DistributedLoad 74.*\$psf
addModel
```



```
# Plan
addModelData ModelLabel 2x2Floor
addModelData ModelDescription "1-Bay by 1-Bay Floor Plan"
addModelData ModelTypeLabel Plan
addModelData iVerticalGridLineLabel "0.0 10.0 25.0"
addModelData iHorizontalGridLineLabel "A B C"
addModelData -HorizontalBayWidth Bay 1 Width 20*\$ft
addModelData -HorizontalBayWidth Bay 2 Width 30*\$ft
addModelData -VerticalBayWidth BayRange "1 2" Width 5*\$ft
addModel
```

```
# 3D Frame
addModelData ModelLabel 3DBuildingFrameRC
addModelData ModelDescription "Let's try It"
addModelData ModelTypeLabel 3DFrame
addModelData PlanModelLabel 2x2Floor
addModelData RigidFloor Off
addModelData -addElevation ElevationModelLabel RCTestFrame2Story2BayA iGridLineLabel "0.0"
addModelData -addElevation ElevationModelLabel RCTestFrame2Story2BayA iGridLineLabel "10.0" TributaryWidth 12.5*\$ft
addModelData -addElevation ElevationModelLabel RCTestFrame2Story2BayA iGridLineLabel "25.0"  TributaryWidth 7.5*\$ft
addModelData -addElevation ElevationModelLabel RCTestFrame2Story2BayBbraced iGridLineLabel "A C" TributaryWidth 2.5*\$ft
addModelData -addElevation ElevationModelLabel RCTestFrame2Story2BayBbraced iGridLineLabel "B" TributaryWidth 5.*\$ft
addModel
```

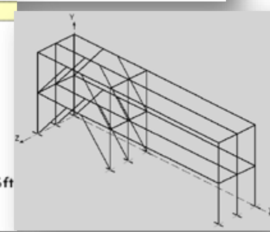


Fig. 2. Example of SeismiCAE language to build models

```
# Gravity Analyses
addAnalysisModelData AnalysisModelLabel GravityAnalysis;
addAnalysisModelData AnalysisTypeLabel LoadControl;
addAnalysisModelData Tolerance 1e-8;
addAnalysisModelData Nstep 12;
addAnalysisModel;
```

```
# Pushover Analyses
addAnalysisModelData AnalysisModelLabel PushoverAnalysisVerySmallSteps
addAnalysisModelData AnalysisTypeLabel DisplacementHistory
addAnalysisModelData DisplacementIncrement 0.01*\$in
addAnalysisModel                                    # --------------------------

addAnalysisModelData AnalysisModelLabel PushoverAnalysisSmallSteps
addAnalysisModelData AnalysisTypeLabel DisplacementHistory
addAnalysisModelData DisplacementIncrement 0.1*\$in
addAnalysisModel                                    # --------------------------
```

```
# Dynamic Analyses
addAnalysisModelData AnalysisModelLabel ShortDynamicTimeHistoryAnalysis;
addAnalysisModelData AnalysisTypeLabel TimeHistory
addAnalysisModelData Tolerance 1e-6;
addAnalysisModelData DtAnalysis 0.1*\$sec
addAnalysisModelData TmaxAnalysis 10*\$sec
addAnalysisModel;                             # ---------
```

Fig. 3. Defining Analysis Models

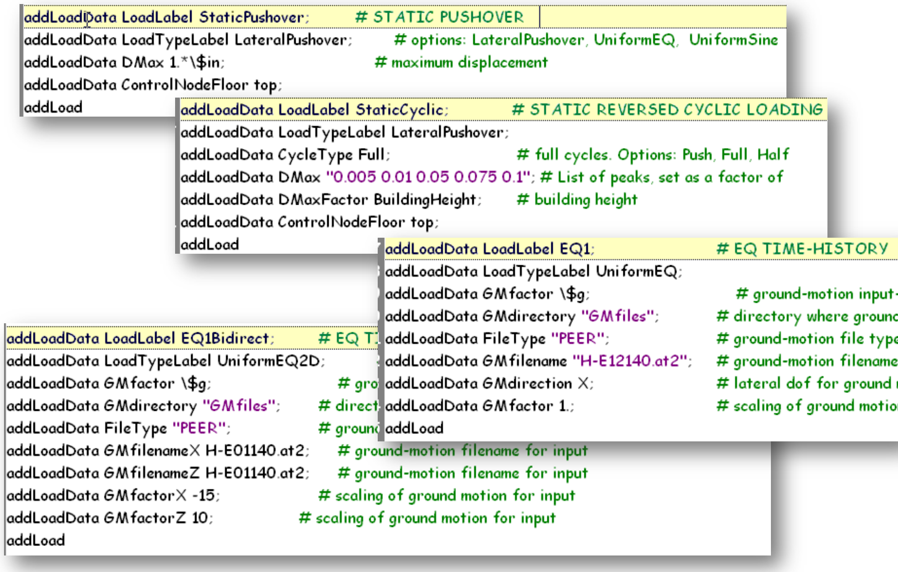


Fig. 4 Defining Lateral Loads

4. SeismiCAE Viewer

All the model and analysis-building commands that can be defined through a script can also be defined using the graphical interface. All the input created via the viewer can be saved and modified in a text script. Figures 5-7 show how materials and sections are defined in the viewer. The graphics of the cross-section geometry and material and section response are updated real-time by calling OpenSees to run in the background.

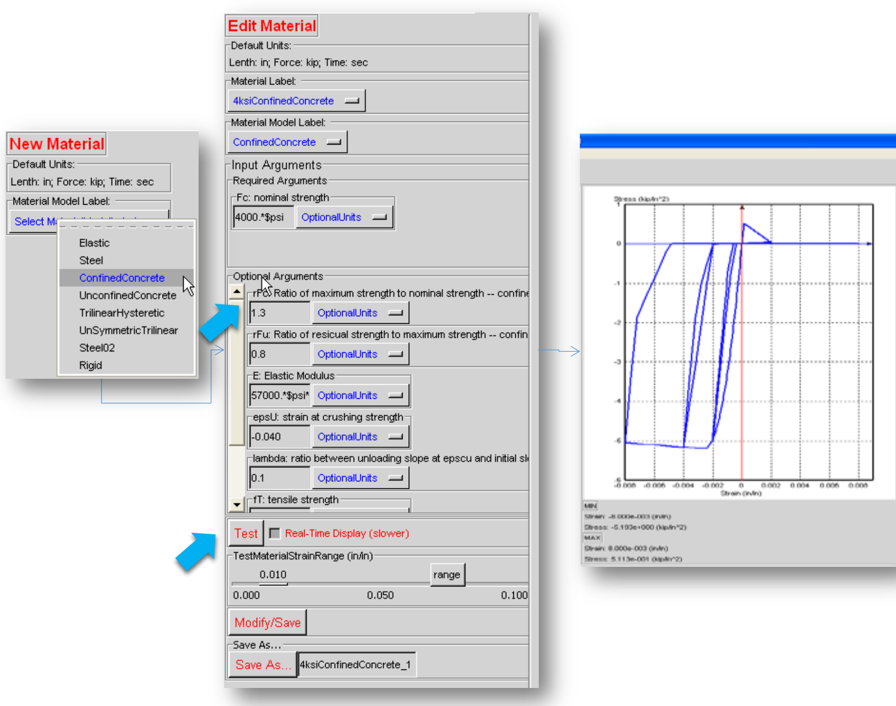


Fig. 5. Defining Materials in Viewer

- Elastic
  - Elastic
  - ElasticNoTension
- Multilinear
  - Steel*
  - Bilinear & Multilinear
    - symmetric & not, define Ks or points
  - Steel01
  - Steel02
- Concrete
  - Confined Concrete*
  - Unconfined Concrete
  - Concrete01,02,03,04,06,07
- Additional
  - Viscous
  - SelfCentering
- Combined
  - Series
  - Parallel
- nDMaterial
  - Elastic Isotropic
  - Etc.

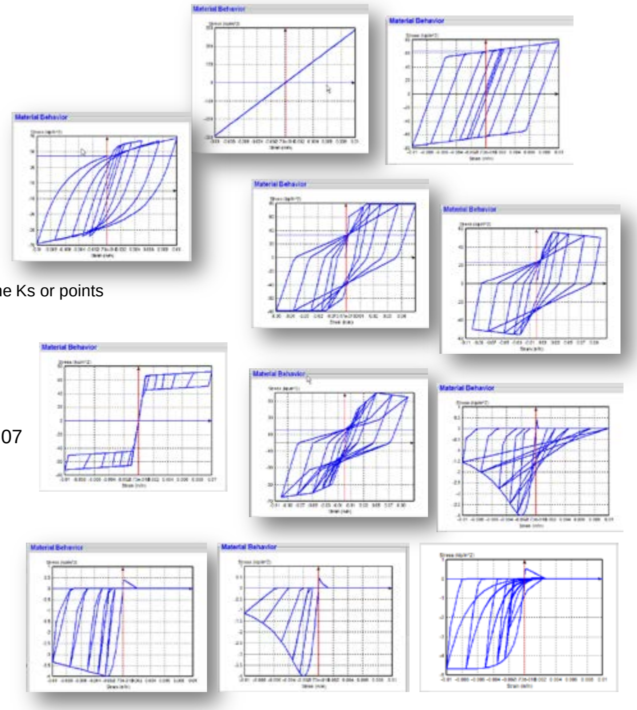


Fig. 6 Library of OpenSees Materials

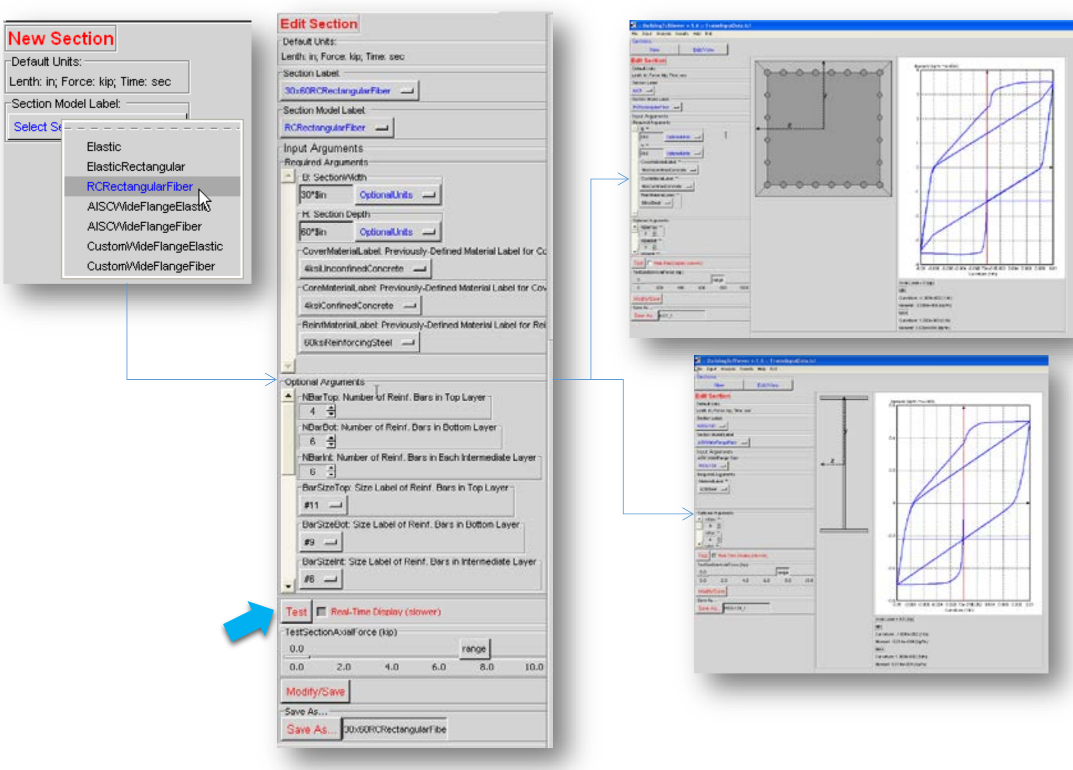


Figure 7. Defining Cross Section in Viewer

Models are built by first defining all the elevations and a basic grid structure, as shown in Figures 8 and 9. The 3D models are then built by assembling the elevations on a grid, as shown in Figure 10.

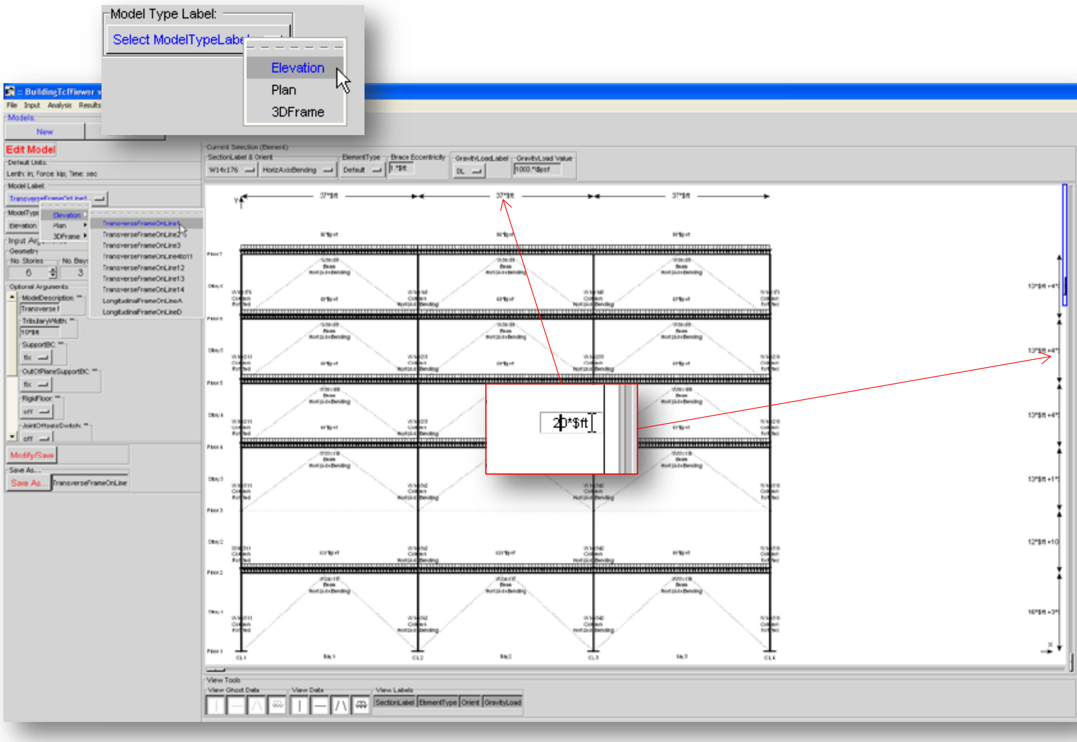


Fig. 8. Defining Elevation in Viewer

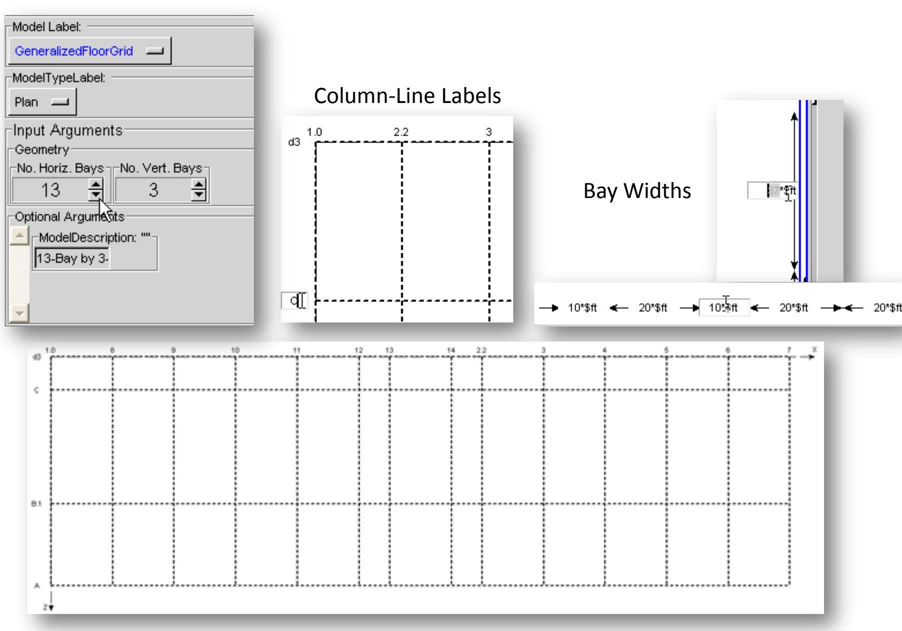


Fig. 9. Defining Generic Grid in Viewer

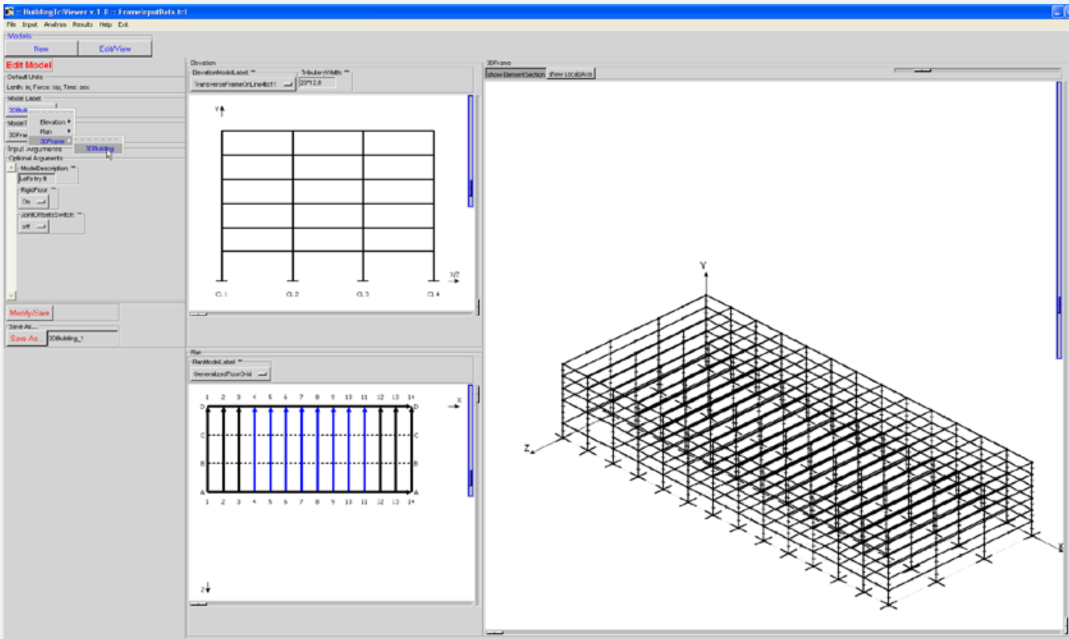


Fig. 10. Defining 3D Model in Viewer

Even though it aims to improve the simplicity and efficiency of building a model, running the analyses and postprocessing, SeismiCAE has all the flexibility of OpenSees. While only one input is necessary to define an analysis, the modeler can modify all default values. The default values provide a guideline for the user. An example of this flexibility is in the definition of the analyses models, as shown in Figure 11.

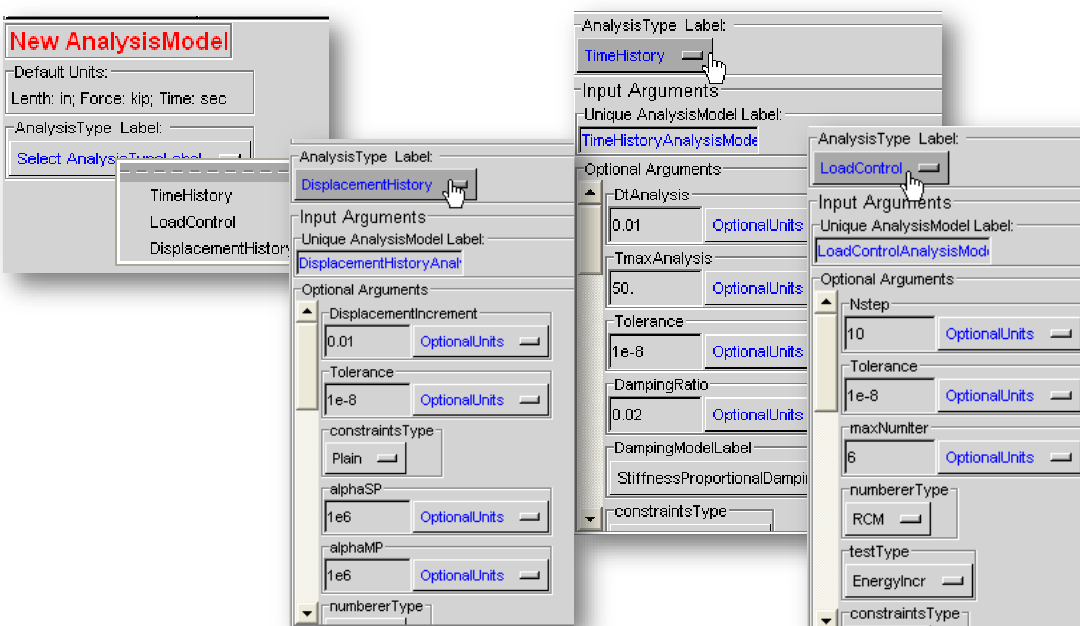


Figure 11. Defining the Analysis Model

5. Running the Analysis

Once a library of models and load combinations has been defined, the user selects all models and all load combinations to be run consecutively, as shown in Figure 12. The run may be started via a batch-file command, or through the Viewer. The modeler has the option to visualize, in real-time, the model as it is run by OpenSees, as shown in Figure 13.

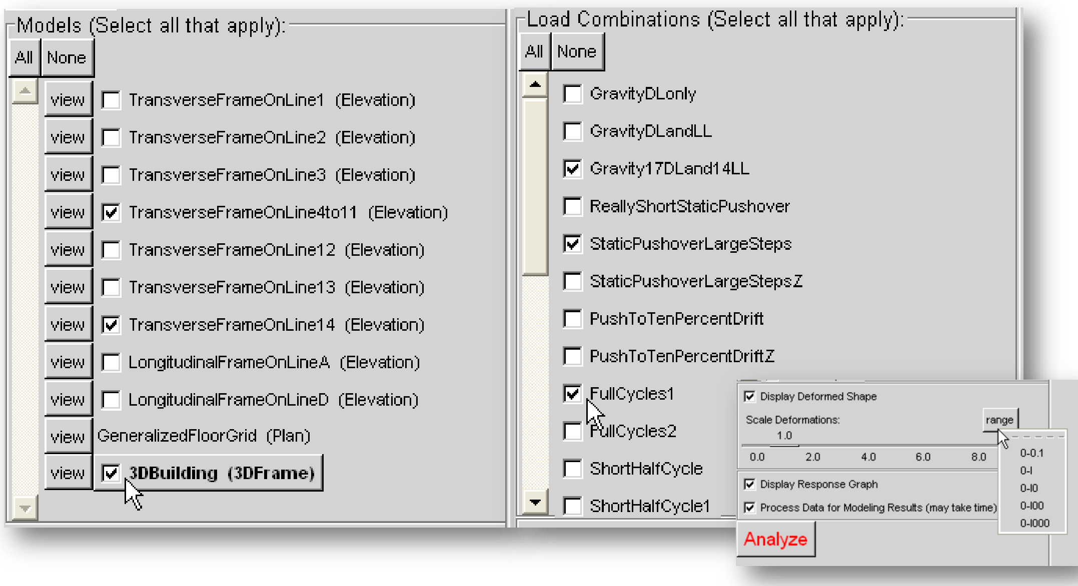


Figure 12. Running Analyses in OpenSees

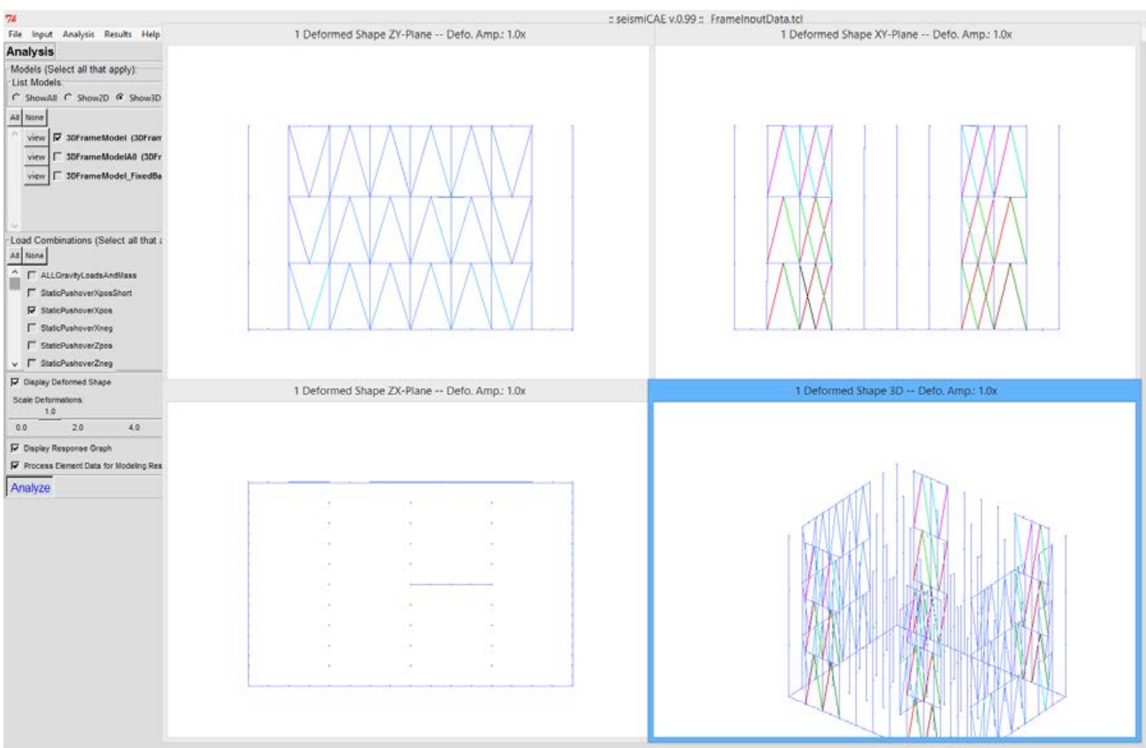


Fig 13 Real-Time Visualization of OpenSees Analysis

6. Analysis Results

One of the most powerful and useful features of SeismiCAE is its post-processing capability. The OpenSees-recorder data is processed so that it can be visualized in the viewer, as shown in Figures 14, or uploaded to a database program. The viewer is useful in evaluating the results of an individual analysis. The database processing is useful in combining the results from many analyses.

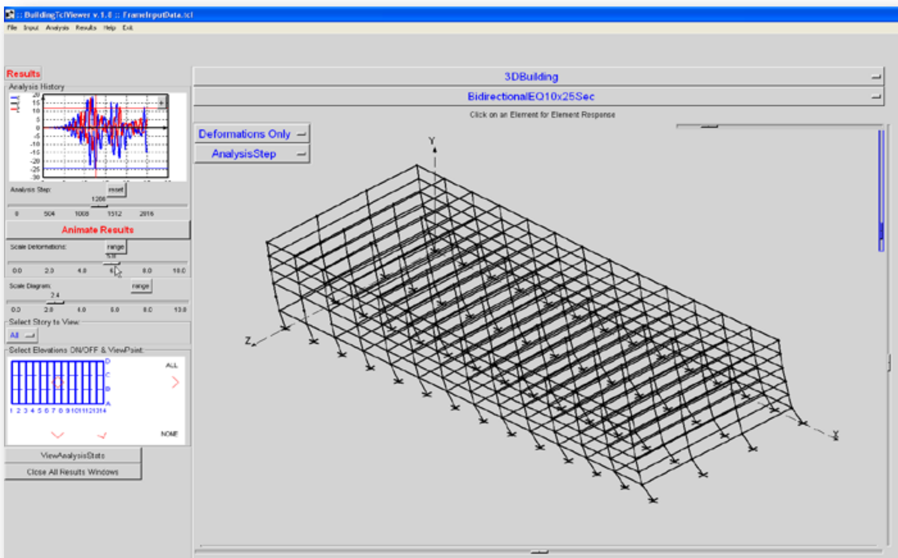


Fig. 14 Analysis-Results Viewer

In the post-processing viewer the user can visualize the deformations at the structural level, as shown in Figure 14 and 15, as well as at the material-fiber level, as shown in Figure 16.

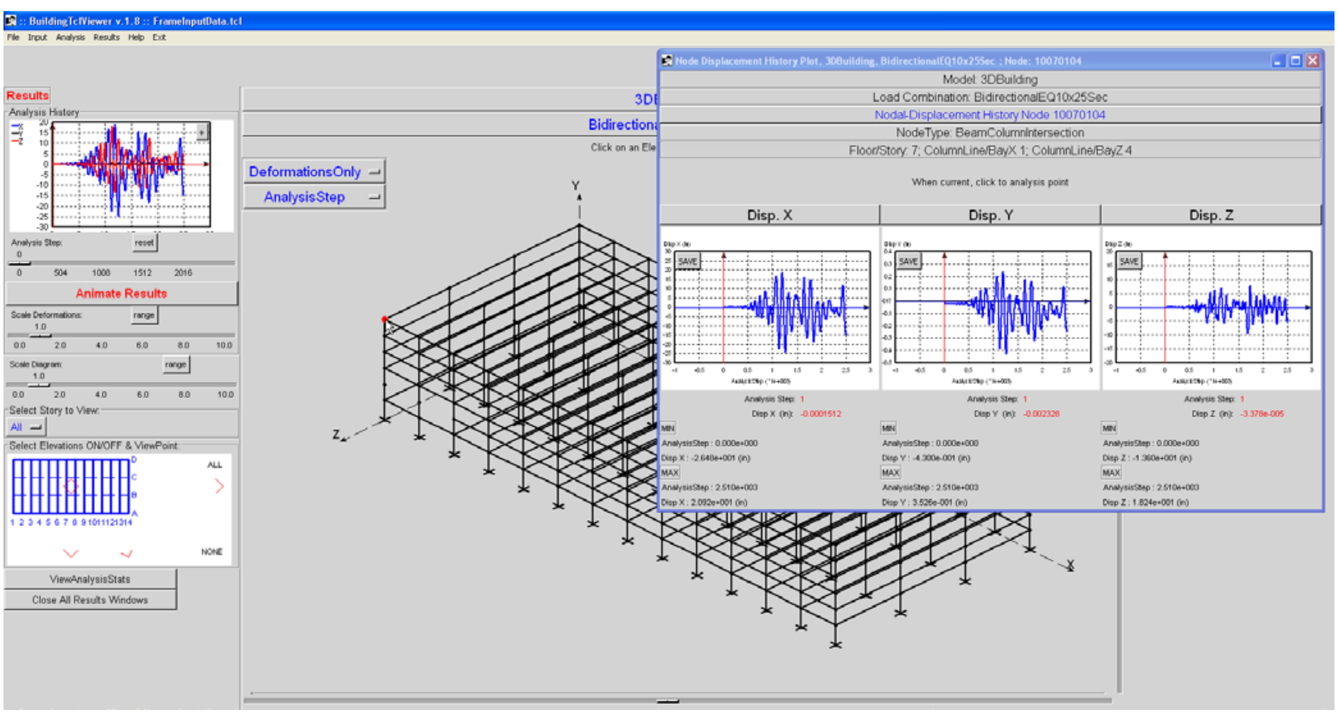


Fig. 15. Nodal Displacements

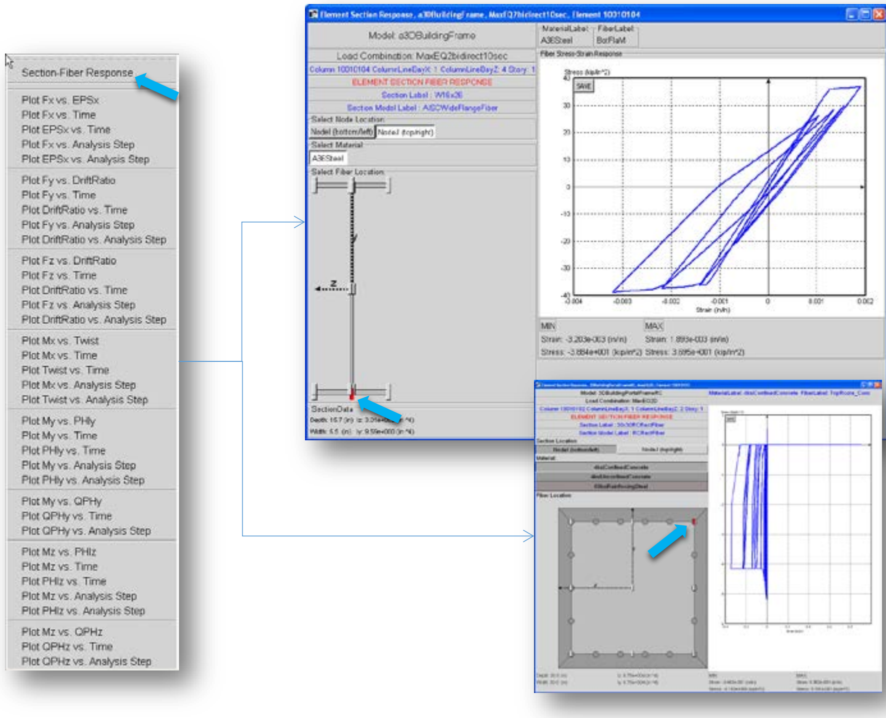


Fig. 16 Material-Fiber Response

All force and deformation diagrams are also available in the viewer, as shown in Figure 17.

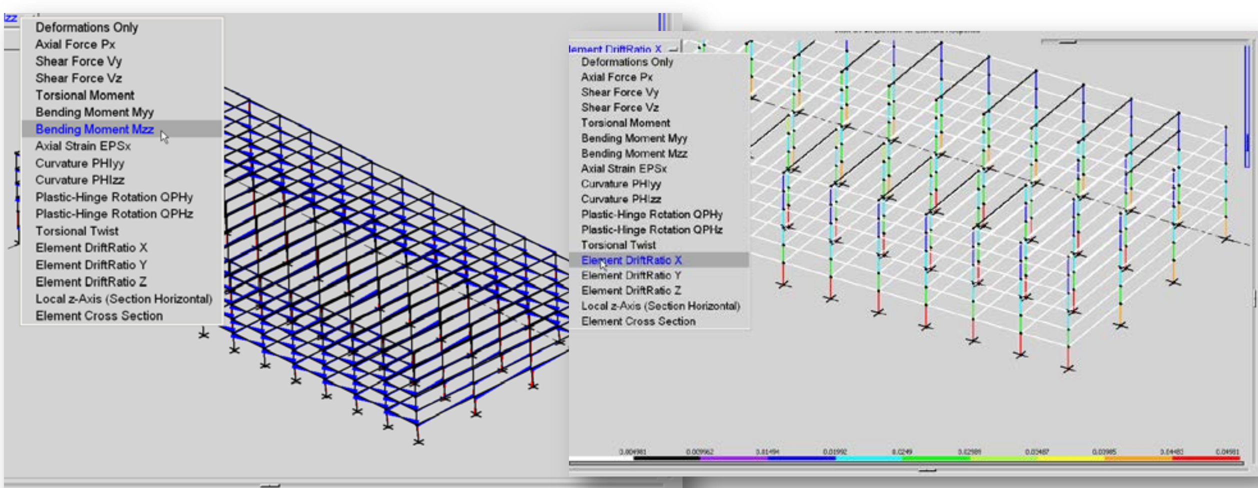


Fig. 17. Element-Response Diagrams

7. Example Application

SeismiCAE was used to model the base rocking of a 10-story RC frame with core wall for the ATC-83 project on Soil-Structure Interaction, as shown in Figure 18 [2].

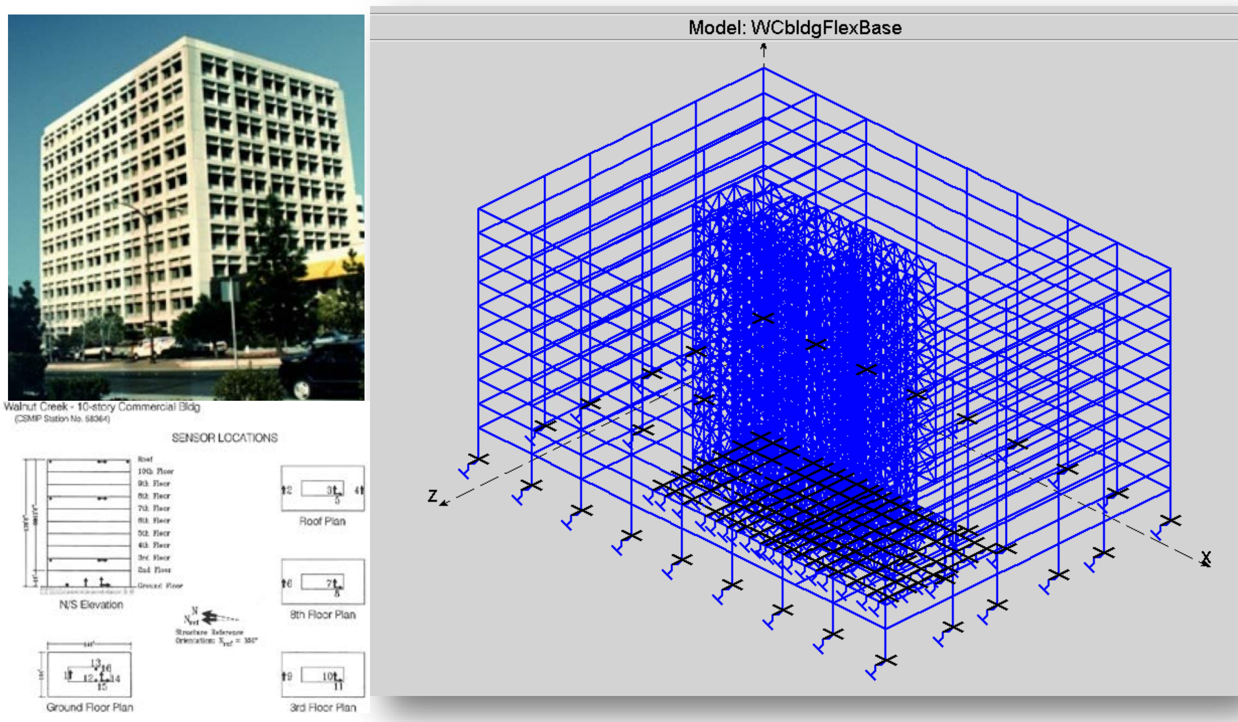


Fig. 18. Example Application

8. Conclusions

SeismiCAE is a powerful tool that integrates a scripting interface with a graphical user interface to run real-time numerical simulations using OpenSees. SeismiCAE is not a pre- and post-processor, it is a completely new scripting and graphical interface which optimes the efficiency of building the OpenSees models, running the analyses in real time, post-processing, and visualizing the results.

9. References

[1] McKenna F, "Object-oriented finite element programming: frameworks for analysis, algorithms and parallel computing." University of California, Berkeley 1997.

[2] NIST GCR 12-917-21. “Soil-Structure Interaction for Building Structures.” NEHRP Consultants Joint Venture A partnership of the Applied Technology Council and the Consortium of Universities for Research in Earthquake Engineering. September 2012