Dynamic Behavior Prediction of a Reinforced Concrete Building
Using Finite Element Analysis with OpenSees

by

K.E. Woiceshyn

Supervisor: Oh-Sung Kwon

December 2011
Abstract

Earthquakes are a rare but significant cause of destruction to infrastructure and threat to human well-being. Since most of the damage is sustained by structures designed by civil engineers, it is important to design these structures to be resistant to such events. In order to do this, it is necessary to understand how the structure will behave under a certain earthquake excitation. Structural prediction combines the latest advances in structural analysis and numerical modeling, and there are several software programs for this purpose. A leading group of earthquake engineering researchers has produced an open-source program, OpenSees, to simulate structural behavior under dynamic excitation, incorporating advanced material models and nonlinear dynamic analysis. The Pacific Earthquake Engineering Research Center recently held a contest for the blind prediction of a test structure under earthquake excitation. The goal of this project was to assess the performance of the OpenSees program by using it to create an entry for the contest, and then analyzing to see which parameters affected the results of the modeling most sensitively. Although it did not win, the model produced reasonable results given the magnitude of the earthquake excitations, and also given a sample video of the shake table test on which this contest was based. The sensitivity analysis showed that the results depended most highly on the properties of the concrete as a material. Ultimately, the exact behavior replication of inelastic structures is still a future endeavor; however, researchers are currently finding many clues to help direct them in the right direction.
Acknowledgements
Table of Contents
Abstract .......................................................................................................................... ii
Acknowledgements ...................................................................................................... iii
List of Figures ............................................................................................................... vi
List of Equations ......................................................................................................... vii
1 Introduction .................................................................................................................. 1
  1.1 Objective ............................................................................................................... 2
2 Background .................................................................................................................. 2
  2.1 The Contest ........................................................................................................... 2
  2.2 Given Data ............................................................................................................. 3
  2.3 OpenSees .............................................................................................................. 5
3 Method ......................................................................................................................... 6
  3.1 Approach ............................................................................................................... 6
  3.2 Model ..................................................................................................................... 7
    3.2.1 Concrete ........................................................................................................... 8
    3.2.2 Steel ............................................................................................................... 9
    3.2.3 Sections ......................................................................................................... 10
    3.2.4 Masses and Gravity Loads ............................................................................ 10
    3.2.5 Eigenvalue Analysis ...................................................................................... 12
    3.2.6 Damping ....................................................................................................... 14
    3.2.7 Gravity Load Analysis .................................................................................. 14
    3.2.8 Dynamic Analysis ......................................................................................... 15
4 Results .......................................................................................................................... 16
  4.1 Output Data .......................................................................................................... 18
5 Sensitivity Analysis ..................................................................................................... 19
6 Potential Limiting Aspects of OpenSees in Modeling Real Structures ....................... 25
  6.1 Moment Resisting Frames (MRF) ......................................................................... 25
  6.2 Confined concrete ................................................................................................. 26
  6.3 From Linear to Nonlinear (Case Studies) .............................................................. 26
    6.3.1 Other Experiments in Nonlinear Structural Modeling .............................. 28
7 Summary ....................................................................................................................... 30
8 Conclusion .................................................................................................................... 31
References ....................................................................................................................... 33
<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Key Construction Drawings</td>
<td>35</td>
</tr>
<tr>
<td>B</td>
<td>Mass and Weight Table of Building</td>
<td>38</td>
</tr>
<tr>
<td>C</td>
<td>Supporting Calculations for Mass Allocation</td>
<td>39</td>
</tr>
<tr>
<td>D</td>
<td>Results for Contest Submission</td>
<td>42</td>
</tr>
<tr>
<td>E</td>
<td>Ground Motion Data</td>
<td>43</td>
</tr>
<tr>
<td>F</td>
<td>Source Code</td>
<td>44</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1: Floor Plan ................................................................. 3
Figure 2: Elevation Views ........................................................................ 4
Figure 3: Stress-Strain of Steel Test Bar ......................................................... 4
Figure 4: Stress-Strain of Concrete Specimen ................................................ 5
Figure 5: Initial Node Layout of Model .......................................................... 8
Figure 6: Stress-strain curve of Concrete02 material ......................................... 8
Figure 7: Stress-Strain curve for Steel02 material .............................................. 9
Figure 8: Floor Layout Illustrating Tributary Areas ......................................... 11
Figure 9: Displacement Spectrum from SeismoSignal ...................................... 13
Figure 10: New Node Arrangement ............................................................... 17
Figure 11: Spiky Data in Roof Acceleration .................................................... 18
Figure 12: Horizontal Displacement at Roof Level ......................................... 18
Figure 13: Acceleration at Roof Level ............................................................ 19
Figure 14: Reaction Forces at Column Bases. Red = Left outer column; Blue = Right outer column ......................................................... 19
Figure 15: Axial Strains in Outer Columns. Red = Left; Blue = Right .......... 19
Figure 16: Krylov-Newton/Broyden algorithm vs. Modified Newton algorithm .......................................................... 20
Figure 17: Max and min displacements with varied Kfc .................................. 21
Figure 18: Max and min displacements with varied damping ratio ........................ 22
Figure 19: Max displacement with varied tensile strength ............................... 22
Figure 20: Max displacement with varied peak strain ...................................... 22
Figure 21: Max displacement with varied ultimate strain .................................. 23
Figure 22: Max and min displacements with varied confinement factor .......... 23
Figure 23: Max and min displacement with variance of peak strain ................... 24
Figure 24: Max and min displacements with variance of tensile strength ........ 24
Figure 25: Max and min displacements with lambda variance ........................ 24
Figure 26: A snapshot of some of the steel layout in the girder ........................ 25
Figure 27: Contest Results for Floor Displacement ........................................ 26
Figure 28: Contest Results for Story Drift Angle ............................................ 27
Figure 29: Contest Results for Floor Acceleration ......................................... 27
Figure 30: Contest Results for Max Curvature .............................................. 28
Figure 31: Floor plan of RC Building ........................................................... 35
Figure 32: Elevation view of RC Building, Frame direction ............................. 35
Figure 33: Elevation view of RC Building, Wall direction and Center Columns .................................................................................. 36
Figure 34: Elevation view of both buildings on shake table ............................. 36
Figure 35: Detail of Column and Girder Reinforcement .................................. 37
Figure 36: Ground Motion for Kobe 25% ....................................................... 43
Figure 37: Ground Motion for Kobe 50% ....................................................... 43
Figure 38: Ground Motion for Kobe 100% ..................................................... 43
Figure 39: Ground Motion for Takatori 40% .................................................. 43
Figure 40: Ground Motion for Takatori 60% .................................................. 43
## List of Equations

<table>
<thead>
<tr>
<th>Eqn.</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eqn. 1</td>
<td>13</td>
</tr>
<tr>
<td>Eqn. 2</td>
<td>13</td>
</tr>
<tr>
<td>Eqn. 3</td>
<td>14</td>
</tr>
<tr>
<td>Eqn. 4</td>
<td>14</td>
</tr>
<tr>
<td>Eqn. 5</td>
<td>14</td>
</tr>
<tr>
<td>Eqn. 6</td>
<td>14</td>
</tr>
<tr>
<td>Eqn. 7</td>
<td>17</td>
</tr>
</tbody>
</table>
1 Introduction

The occurrence of a large earthquake is becoming almost a yearly event somewhere in the world. The largest earthquakes cause many billions of dollars in damage as well as loss of human life. For example, in March 2011, Japan sustained one of the most severe earthquakes in history, killing at least 15,000 people and causing over 300 billion (USD) in damage [1]. The majority of this damage is sustained by buildings and other public infrastructure. Earthquake engineering, the study and prediction of the dynamic behavior of structures in response to earthquake forces, and the design of these structures to best minimize adverse effects, is a relatively new and exciting field of research.

An important area of earthquake engineering is the prediction of the dynamic behavior of a structure subject to a known loading pattern. The Pacific Earthquake Engineering Research Center at UC Berkeley (PEER) recently held the “2011 Blind Prediction Contest” for the modeling and behavior prediction of a specific concrete building subject to a specific earthquake loading pattern. This includes predicting maximum accelerations and displacements experienced at each floor of the structure, as well as damage sustained by certain structural components. These are important data to determine because the contents of a building (people, equipment, etc.) are highly sensitive to displacements and accelerations.

Accurately predicting the response of a nonlinear structure is, at the time of this writing, not an exact science. PEER held a similar contest in 2010 for the behavioral prediction of a single, cylindrical reinforced concrete bridge pier subject to dynamic ground loading. From the contest results, even the winning entries differed significantly from the experimental data, and in a few instances, there was up to a 50% difference between the winning predicted value and the experimental value\(^2\). The results of this and a similar contest are discussed in Section 6.3.

At the time of this writing, several commercial software programs exist for the purpose of earthquake modeling of structures, the leading players being SAP2000\(^3\) and SeismoStruct\(^4\), and more general finite element analysis programs such as ANSYS\(^5\). These programs are great for the design of earthquake-resistant structures according to various structural code clauses, and according to the principles of

\(^1\) [http://peer.berkeley.edu/prediction_contest_2011/](http://peer.berkeley.edu/prediction_contest_2011/)
\(^2\) [http://nisee2.berkeley.edu/peer/prediction_contest/?page_id=25](http://nisee2.berkeley.edu/peer/prediction_contest/?page_id=25)
\(^5\) [http://www.ansys.com/](http://www.ansys.com/)
performance based earthquake engineering (PBEE), which is focused more on designing structures to meet a variety of societal and economic needs [2]. What many of them do lack, however, is a framework for performing accurate simulations of pre-existing structures, and the ability to modify many of the input and output parameters individually. Thus, of more interest to this project are the programs which are ongoing projects of various research groups at universities, and thus are often open-source, ongoing collaborative projects, and are available free of charge. The most well-known of these programs among the earthquake engineering community is OpenSees, the brainchild of a key research team at PEER6.

1.1 Objective
The goal of this project was to successfully model a full-scale, inelastic structure subject to earthquake excitation, and to investigate the modeling parameters that most sensitively affect the response of the structure. This involved creating an entry for the 2011 Blind Prediction Contest using the OpenSees program. In this process, several assumptions were made in order to reduce the aspects of the real structure into a set of parameters that could be processed by OpenSees. Because of these reductions, some of the parameters were more sensitive than others. Post-contest sensitivity analysis was performed to determine the individual sensitivities of various parameters. This was important toward the making of an assessment of the strengths and weaknesses of OpenSees, as well as an overall assessment of its accuracy and applicability to the assessment of real structures.

2 Background
2.1 The Contest
In 2010, the National Research Institute for Earth Science and Disaster Prevention (NIED) of Japan performed a simultaneous shake table test of two full-scale, four story concrete buildings. The two buildings were nearly identical in size, dimension, and mass; however, one used conventional reinforced concrete (RC), while the other featured post-tensioned (PT) floor girders and columns. Both were constructed and prepared according to the Japanese building code, and were intended to represent realistic buildings as accurately as possible [3].

The tests were performed by input of a simulated ground motion via the shake table, using the time-history patterns of two real previous earthquakes at various levels of intensity. Instrumentation was

---

6 http://opensees.berkeley.edu/
implemented at several points on each building to record displacements, accelerations, reaction forces, and element strains.

PEER used these tests and the building designs as the subjects of the contest, requiring that the recorded results from the tests be withheld from the public until the contest closed. The contest was open to all students, researchers, and professionals. A successful submission to the contest consisted of a computer model of one of the buildings, as well as a table of predicted values for floor displacements, accelerations, axial forces and strains in the columns. No restrictions were placed on how or by what means the model was to be created, and the winners would be those who obtained numerical predictions closest to the measured results.\(^7\)

### 2.2 Given Data

The two buildings were fully detailed in a series of construction drawings [4]. Each building was two by four bays, with four stories above ground. The two buildings were designed to be of identical geometry and dimension as much as possible. In the short direction (y-direction), shear walls are included at each end of the buildings, in between the corner columns. Each building has six square columns, four at the corners, and two in the middle of the long side. At each floor, all columns and the shear walls are connected by rectangular girders. These girders are interlaced by rectangular beams which support a flat floor slab. The floor plan of the building is shown in Fig.1, and the elevation views and dimensions are shown in Fig.2.

\(^7\) [http://peer.berkeley.edu/prediction_contest_2011/?page_id=19](http://peer.berkeley.edu/prediction_contest_2011/?page_id=19)
Further data were given as to the details of each structural member, including dimensions and reinforcement details. For the RC structure, this involved specification of the longitudinal and transverse reinforcing steel [Appendix A, fig. 20]. For the PT structure, this involved dimensions, details, and geometry of the tendons and ducts. Detailed drawings were provided for the configuration of all reinforcing steel and post-tensioning tendons and ducts. Total weights of various components of the structures were given, as well as dead loads, and total weights of each floor [Appendix B].

Specific numerical values were given for peak compressive and yield stresses of the concrete and reinforcing steel, respectively. A document summarizing the results of tensile tests on various sample bars of reinforcing steel and prestressing tendon were provided\(^8\). For each sample, a stress-strain plot was provided showing yield stress values and rupture strains, a sample of which is seen in Fig.3. A spreadsheet was provided summarizing cylinder compression tests on 27 concrete cylinder samples of unspecified size\(^9\). Stress and strain values were given until failure for each sample. Plotted, this data leads to a familiar, apparently parabolic concrete stress strain curve, an example of which is shown in Fig.4.

The buildings were tested with a series of five separate earthquake ground motion patterns. Two different earthquakes were used, JMA-Kobe and Takatori. The tests involved using 25%, 50%, and 100% intensity of the Kobe pattern, followed by 40% and then 60% of the Takatori pattern. Ground motion data was provided in a series of text files each over duration of approximately 52 seconds, with a time step of 0.005 seconds. Acceleration values were provided in \( \text{m/s}^2 \). Graphical representations of the critical segments of each of the five patterns are shown in Appendix E. The patterns were applied to the buildings in series, so that the damage accumulated from one test would carry over into the next test.

A video recording of one of the test patterns (Takatori 60%) was provided\(^{10}\). It shows the buildings shaking considerably, but this may be due to the motion of the table, which is unseen in the video. Including the motion of the table, the amplitude of vibration at roof level seems to be less than one-half meter, and neither of the buildings appears to undergo significant permanent deformation or collapse. It is clear that reinforcing steel undergoes yielding, and that there is some rotation at the beam-column joints, as some concrete can be seen flaking off at these locations. Rupture of reinforcing steel is unlikely, as no catastrophic failure was witnessed. The test shown in the video is not the most severe of the ground motions; however, it occurs after the most severe in the sequence, the Kobe pattern at 100%. Thus, any damage sustained during the Kobe 100% would be notable in the video.

### 2.3 OpenSees

OpenSees is an ongoing collaborative project started at PEER. It is an object-oriented, finite element analysis platform for modeling structures under static or dynamic loads. It is open-source, and thus available to download free of charge. It is primarily written in C++. OpenSees currently lacks a graphical user interface. Instead, the user operates the program by entering commands written in the Tool Command Language (Tcl). Long scripts in Tcl are usually written out and then called in to the

\(^{10}\) [http://peer.berkeley.edu/prediction_contest_2011/?page_id=177](http://peer.berkeley.edu/prediction_contest_2011/?page_id=177)
OpenSees program. The script used in this project is shown in Appendix F. An OpenSees User Wikipedia is available online\textsuperscript{11}, which serves as a user manual.

3 Method

3.1 Approach

The Blind Prediction Contest encouraged but did not require the modeling of the buildings in three dimensions. Three separate contest categories were created, one for modeling the RC building in the moment frame direction, one for modeling the RC building in the shear wall direction, and one for modeling the PT building in the shear wall direction. It was assumed that once the entrant chose which category to work in, that the effects of ground motion in the perpendicular direction could be ignored, leaving a two-dimensional problem, making the task much easier.

Due to the scope of this project and the author’s limited exposure to dynamic structural analysis, as well as the fact that many high-caliber academics and professionals would be submitting contest entries, it was decided that the simplest possible case should be considered. This would also lead to more meaningful analysis and interpretation, without getting too lost in complexity.

Of the two buildings, an analysis of the PT building might have yielded more accurate predictions, because, due to the post tensioning, the concrete would have exhibited less cracking and nonlinear behavior. Its behavior would have been more close to elastic than that of the RC building. However, modeling the PT building in OpenSees would have posed a significantly greater challenge, due not only to the greater number of components and detail in the structural drawings, but to the approximation of the prestressing tendons as applied forces on the structure, as OpenSees currently lacks a specific module for the direct implementation of prestressed members.

Of the two directions, the shear wall direction would have posed the additional challenge of modeling the shear resistance of the wall. According to Professor Michael Collins at the University of Toronto, flexure is currently a “solved problem,” whereas shear is not\textsuperscript{12}. Given that there exists no exact solution for predicting shear behavior as there is for flexure, the author felt that this challenge would be best left to a later, separate study. Also, OpenSees currently lacks the capacity to directly model transverse reinforcement, requiring even more approximations.

\textsuperscript{11} http://opensees.berkeley.edu/wiki/index.php/OpenSees_User
\textsuperscript{12} Anecdotally noted by the author when taking the course “CIV517: Prestressed Concrete”
Given that for the PT building the only choice given was to model the shear wall direction, it was decided that the easiest would be to model the RC building in the frame direction, minimizing the shear considerations required, and understanding that most of the damage would occur due to flexure at the joints.

Given that the RC is approximately symmetrical about the long axis, it was decided that modeling half of the building would represent behavior of the whole building reasonably well. A model of half the building could be reduced from three to two dimensions, as all of the columns and girders in the frame (long) direction lie in a single plane. The loads and weights of the floors would have to be calculated and distributed on this two-dimensional structure. Although the shear wall sits in the middle of the long axis, its ability to resist moment in the frame direction, it was decided, was minimal, due to its small dimension (250mm) in this direction. Thus, its effects on moment resistance were chosen to be ignored.

### 3.2 Model

The procedure for performing a test in OpenSees involves two stages: building the model, and running the analysis of the model. In order to build the model correctly in OpenSees, the contest input data had to be reviewed thoroughly. Initially, the structure was divided into nodes and linear elements based on the cross sections defined in the construction drawings. Each member with uniform section properties and reinforcement was treated as one element. In the drawings, the columns in the first floor have additional longitudinal reinforcement in the lower half [Appendix A, fig. 20], meaning that the first story columns were made of two elements each. All other columns have uniform section, so these were modeled as single elements. Each girder has additional longitudinal reinforcement at its outer two thirds, meaning that each girder was made of three elements.

To maintain order, the nodes were defined in a coordinate system, with the smallest division in either a girder or column designated as one unit. Thus, the entire structure, which consisted of two bays of 7.2 m each and four stories of 3.0 m each, was defined as 6 units long and 8 units tall, given that the smallest column division was half of a column, and the smallest girder division was one third of a girder. The nodes were labeled, viewing the structure in the frame direction, as per Fig. 5. This was an easy method of definition because only the unit lengths had to be defined in each axis, and the rest of the structure could then be defined in multiples of these units.
OpenSees is unitless, thus everything must be defined in terms of a base unit, maintaining internal consistency in the model. Basic units of Newtons for force, metric tons for mass, and millimeters for length were assumed. This was chosen because masses dealt with were to be large and small units of length were needed to define section details.

Once the geometrical positions of the nodes were defined, the bases of the three columns were fixed against displacements and rotation using the \textit{fix} command in OpenSees.

3.2.1 Concrete

The OpenSees uniaxial material definition \textit{Concrete02}\textsuperscript{13} incorporates linear softening after peak stresses are reached, in both compression and tension. Peak stress in tension ($f_t$) is reached linearly at a defined stiffness, and, after the peak, stress reduces along the tensile softening slope ($E_{ts}$) until reaching zero [fig. 6]. In compression, stress increases with strain linearly with slope $E_c$ until it begins to curve toward the pre-defined peak stress ($f'_c$) and strain at peak stress point ($e_{c0}$). Linear stress decline then occurs until it reaches crushing stress ($f_u$) at crushing strain ($e_a$). An unloading slope, \textit{lambda}, is also defined.

The $f'_c$ was given in the construction drawings as 27 MPa. The spreadsheet of cylinder tests showed peak stresses were too varied to be trustworthy. These graphs, however, proved useful in determining the elastic modulus ($E_c$) of concrete. The slopes of several of these were calculated in Microsoft Excel.

\textsuperscript{13} \url{http://opensees.berkeley.edu/wiki/index.php/Concrete02_Material_--_Linear_Tension_Softening}
yielding an average value of 31200 MPa, which, from the author’s experience in concrete design courses, seems like a reasonable value. The strain at maximum stress was defined as twice \( f'_{c} \) divided by \( E_{c} \), as given in Fig. 6, resulting in a value of approximately 1.7E-03. The \( f_u \) was defined as a fraction of \( f'_{c} \), for which a value of 0.2 was used, based on various example models in the OpenSees User Wiki\(^{14}\). The \( e_u \) was defined as -0.01, again based on the example models. This value is large compared to what is typically used in design (-0.003) and represents significantly more plastic deformation than is likely realistic. With limited guidelines for selection, this parameter was anticipated to be a potential source of uncertainty and to later be assessed in a sensitivity analysis. The value of \( \lambda \) was defined as 0.1 of \( E_{c} \), a default value again as per the examples.

For the confined concrete, as per a method outline in a paper by Mander et al. [10], the value of \( f'_{c} \) was multiplied by a factor of 1.3 to represent the increased strength of confined concrete. The \( f_t \) was defined as 0.14*\( f'_{c} \), which yielded a value of 3.78 MPa, which is somewhat higher than what is typically used in design (≈ 2 MPa). However, this allows for a more elastic behavior, which, like the exaggerated estimate of crushing strain, encourages more elastic behavior of the building, and increases the chances of the analysis converging. The \( E_{ts} \) was defined, by default, to be 0.002*\( f_t \), again, a default value based on the examples. Two separate \texttt{uniaxialMaterial} objects were created, one with the properties of confined concrete, and the other with properties of unconfined concrete.

3.2.2 Steel

The OpenSees uniaxial material object \texttt{Steel02}\(^{15}\) is represented by an approximate bi-linear stress-strain curve. This requires the definition of Young’s modulus \((E)\) and yield stress \((f_y)\) of the steel [fig. 7]. Three other parameters are required to define the transition between the elastic and plastic behavior as a small curve, rather than a sharp corner.

For the rebar steel, the values given in the construction drawings were used directly. For the main longitudinal reinforcement of the girders and columns, D22 bars were used, whose \( f_y \) was given as

\(^{14}\) http://opensees.berkeley.edu/wiki/index.php/OpenSees_Example_5._2D_Frame,_3-story_3-bay,_Reinforced-Concrete_Section_%26_Steel_W-Section
\(^{15}\) http://opensees.berkeley.edu/wiki/index.php/Steel02_Material_-_Giuffr%C3%A9-Menegotto-Pinto_Model_with_Isotropic_Strain_Hardening
345 MPa. In the steel stress strain curves given in the contest data, which were results of tests, show a fairly linear elastic behavior until around 350 MPa whereupon the curve nearly flattens off [Fig. 3]. Thus, the value of 345 is satisfactory. The only other type of bar used in the RC building is the D10, which serves as skin reinforcement in the girders. The $f_r$ is given as 295 MPa, which is satisfactorily in accord with the values in the graphs. An $E$ of 200000 MPa was chosen, as this is the most commonly used value for steel in structural engineering, according to the author’s experience from coursework. The values controlling the transition from elastic to plastic were left at their default values, as there was no reason to change them.

3.2.3 Sections
The most common way of defining a heterogeneous (i.e. reinforced concrete) section in OpenSees is by using the RC Section Builder module developed by Dr. Mazzoni at PEER\textsuperscript{16} [end of Appendix F]. This module creates a rectangular cross section and divides it into a grid of “fibers,” the number of which is specified by the user. Each fiber behaves uniaxially in the direction of the element longitudinal axis, according to the properties of the pre-defined material used for each fiber. Thus, each individual element can only undergo uniform lengthening or shortening. The default number of fibers seemed to be 20 in each direction, and this was the number used in the model. The longitudinal bars were placed as additional fibers at their appropriate positions in the section, corresponding to the diagrams in the construction drawings. All of the concrete fibers inside of the outermost rectangle formed by the reinforcing bars were defined as confined concrete, whereas all concrete outside of this outermost rectangle was defined as unconfined concrete.

The elements were then defined as displacement-based beam/column elements, using the appropriate sections as previously defined. Displacement-based means that the material properties of the element are defined by displacements of the element, and the model accounts for the spread of plasticity effects in the fibers. Five integration points were chosen unanimously for all elements, regardless of the length. The columns were defined to account for P-Delta effects using the $PDelta$ coordinate transformation option, which is more realistic than not accounting for these effects.

3.2.4 Masses and Gravity Loads
One of the largest challenges of creating a 2-D model of a 3-D structure was to accurately represent all of the dead loads and self-weight of the structure. In OpenSees, the mass of the structure must be

\footnotesize{\textsuperscript{16} http://opensees.berkeley.edu/wiki/images/f/f8/BuildRCRectSection.tcl}
allocated at the nodes. The real structure is far more complex, with the mass being distributed throughout the members, with variations due to different material densities. The masses of the different components of the RC structure are given in the PEER Report [3] [Appendix B], however, the values are given in totals per floor.

In order to allocate all self-weights, loads, and masses to the columns and the girders in the x-direction (C1, C2, and G1 in Fig. 8), the floor needed to be divided into tributary areas. The load from the slab was assumed to go into the beams (B) and then into the girders at the ends of the beams (G1) and then into the columns. Although not considered in the analysis, it was assumed that the shear wall would act as a column and thus carry load downward. It was given its own tributary area. The tributary area for the corner columns (C1) was taken directly from a calculation example in the PEER Report [3, p. 25]. For half of the floor, as considered in the figure, this involved leaving a 1.7 m x 2.3 m rectangle tributary area for the shear wall. The girder connecting C1 and C2 was split at the middle, with the load on one half going to C1 and the load on the other half going to C2. Thus, the tributary areas for C1 and C2 could be drawn as shown in Fig. 8.

Figure 8: Floor Layout Illustrating Tributary Areas

As an example of the process, the total slab for the roof floor is given as 44.1 tons. For half of the floor as considered in the model, this equates to 22.05 tons. The portion of this mass that is allocated to Column 1 is determined by multiplying 22.05 t by the tributary area for C1 divided by the total area of half the floor. The resulting number is treated as a point mass which is applied at the top of C1. The process can be repeated similarly for C2. For the beams and girders, the total masses of these at a given
floor is divided by the total length of beams and girders in that floor (accounting also for the differences in cross section) to obtain a mass per unit length. Then, the total length of girder and beam within the tributary area of a column is calculated, and, given the mass per unit length, it is possible to calculate the total mass of beams and girders allocated to each column. Thus, as with the slab, this mass is defined as a point mass at the column. Masses of other components of the structure are defined in a similar fashion according to tributary areas, and then defined as point loads at the nodes. The column mass per unit length can be determined by dividing the value in the table by the length of column in each floor (the halves of each column above and below a floor are allocated to the floor). Column masses are placed at the nodes using this half above and half below approximation. A detailed calculation of all of the nodal masses is presented in Appendix C. After considering all of the structural components defined in the table, all of the allocated point masses are added up, and these values are then defined at the respective nodes in OpenSees using the mass command\(^ {17}\).

The RC structure supports a variety of loads from equipment and machinery, primarily on the roof. Also, stair and railing loads must be considered. The PEER Report [3, p. 16] defines the quantity of each of these loads on each floor, but does not describe their placement or concentration on the floor. Without this further information, all of these additional loads were assumed to be uniformly distributed over the floor. Railing and parapet loads were assumed to be line loads along the perimeter of the floor. Stair loads were allocated to the middle column, as from pictures of the structure it appears that the stairs are mounted at the middle column [Appendix A, fig. 19].

The OpenSees model also requires the gravity loads to be defined as loads per unit length along the elements. Thus, in addition to defining point masses, all of the components contributing to these masses had to be defined as line loads along the girders in the x direction (G1 in Fig. 8). The calculations for this were similar to those used to define the point masses, however, instead of allocating all of the mass within the tributary area to a point, it was multiplied by the gravitational constant (to make it a load) and divided over the length of girder G1 within that tributary area.

3.2.5 Eigenvalue Analysis

In order to determine the response of the structure to ground motion, the natural period of the structure must be determined. OpenSees, having calculated the mass and stiffness matrices, calculates the eigenvalues ($\lambda$) from taking determinant of the stiffness matrix multiplied by a diagonal matrix of $\lambda$.

\(^ {17}\) http://opensees.berkeley.edu/wiki/index.php/Mass_Command
values, equal to zero, and solving for \( \lambda \) values [eqn. 1]. This is done using the *eigen* command. The lambda values represent the eigenvalues of the different vibration modes, and taking the square roots of them leads to the fundamental angular frequencies (\( \omega \)) of each mode. The fundamental periods are then calculated by taking the inverse of the \( \omega \) values multiplied by \( 2\pi \) [eqn. 2].

\[
(K - \lambda I) \Phi = 0 \quad \text{(Eqn. – 1)}
\]
\[
\omega = \sqrt{\lambda} = \frac{2\pi}{T} \quad \text{(Eqn. – 2)}
\]

OpenSees repeats this procedure throughout the analysis due to the changing stiffness values associated with accumulating earthquake damage. Thus, it can be observed that the natural period increases throughout the test.

For the model, the eigenvalues were determined for two modes of vibration. The first mode is a simple, back-and-forth sway of the whole building, whereas the second mode would include a pivot somewhere in the middle of the building. From the video recordings, it was assumed that most of the vibration would occur in the first mode, as the buildings in the video appeared to sway back and forth uniformly. Thus, the eigenvalue of the first mode and the natural period determined from it would be the most critical value. OpenSees evaluated this period to be approximately 0.4 seconds at the beginning of the test.

An additional program, SeismoSignal\(^{18}\), was used to assist in determining the period. SeismoSignal takes in ground motion data from a file and determines the Fourier and power spectra of the data. It also plots the elastic response spectra for a given damping value, which shows the maximum displacement values expected for an elastic structure for different values of


![Figure 9: Displacement Spectrum from SeismoSignal](image-url)
natural period [fig. 9]. Given that from the video, the displacement of the building did not appear to be greater than 0.2 or 0.3m, it was fairly evident that a period value of 0.4 was reasonable, as this is for an approximate elastic structure.

3.2.6 Damping
OpenSees uses the Rayleigh method [5, p.199] to construct a damping matrix for the structure. This is done by calculating a coefficient for each of the mass and stiffness matrices. The damping matrix is then the sum of the mass and stiffness matrices multiplied by their respective coefficients [eqn. 3]. For a transient analysis, the Rayleigh process takes into account an initial and current stiffness matrix, to re-adjust the damping matrix in accordance with accumulated structural damage, much like is done in the eigen analysis\(^\text{19}\). Typically, the damping ratio is calculated based on the chosen coefficients [eqn. 4]. One of the methods used in an example analysis\(^\text{20}\), and the one used in this project was to calculate the alpha and beta coefficients based on the natural frequencies of the first and second modes of vibration, as per [eqn. 5,6].

\[
[C] = \alpha [M] + \beta [K] \quad \text{(Eqn. – 3)}
\]

\[
\xi = \frac{1}{2} \left( \frac{\alpha}{\omega_i} + \beta \omega_i \right) \quad \text{(Eqn. – 4)}
\]

\[
\alpha = \xi \left( \frac{2 \omega_1 \omega_2}{\omega_1 + \omega_2} \right) \quad \text{(Eqn. – 5)}
\]

\[
\beta = \frac{2 \xi}{\omega_1 + \omega_2} \quad \text{(Eqn. – 6)}
\]

The damping ratio of a large, complex structure is difficult to determine, and is often done so from empirical tests. Many approximate formulas exist, based on research, for estimating the damping ratio based on a given number of properties of the structure [11]. The most common approximation for reinforced concrete structures is given as 5% [8]. This was the value used in the tests.

3.2.7 Gravity Load Analysis
The gravity loads in OpenSees need to be applied to the structure before the dynamic analysis can be executed. This was done using a load-controlled static analysis, applying the gravity loads in 10 steps over a total time of 10 seconds, increasing from zero to full loads in a ramp function. Newton’s

\(^\text{19}\) http://opensees.berkeley.edu/wiki/index.php/Eigen_Command
\(^\text{20}\) http://opensees.berkeley.edu/wiki/index.php/OpenSees_Example_5_2D_Frame_3story_3-bay_Reinforced-Concrete_Section_%26_Steel_W-Section
algorithm was used with a displacement increment test with a convergence tolerance of 10E-06. After the analysis, the gravity loads were maintained constant using the loadConst command, and the time counter was reset to zero for the dynamic analysis.

3.2.8 Dynamic Analysis

In order to perform a dynamic analysis in OpenSees, the user must elect from several options as to how the analysis is to be carried out. First, the method by which the analysis handles the constraints and boundary conditions must be chosen using the constraints command\textsuperscript{21}. In the model, properties of individual elements are defined in a local coordinate system and later transferred to the global coordinate system of the model. Thus, the Transformation constraint handler was selected.

Next, the way in which the degrees of freedom at individual nodes are mapped to the system of equations in the analysis needs to be chosen using the numberer command. Given the size and complexity of the model, and the need for optimizing computation time, an option which uses the reverse Cuthill-McKee algorithm\textsuperscript{[7]} to order the matrix equations was selected.

Next, the way in which the system of equations is constructed must be chosen with the system command. For basic models, the BandGeneral option may be selected. This option is used for matrices with a banded profile, meaning that all of the nonzero values exist along diagonals in a “band” in the middle, and all values outside this band are zero. Since the model had larger, sparse matrices, the UMFPack solver\textsuperscript{[8]} for large, sparse linear systems was selected.

A convergence test may be chosen using the test command. This checks to see if convergence has been reached in the linear SOE after each iteration step in the analysis, by finding the solution vector of the matrix equation. The displacement increment test uses the norm of the solution vector of the left side of the matrix equation to test for convergence, and this is equal to the displacement increments applied to the structure. For the model, a displacement increment convergence test was used with a tolerance of 10\textsuperscript{-6}, same as for the gravity load analysis.

To solve the nonlinear equations, a solution algorithm must be chosen. The most commonly used option in OpenSees is the Newton-Rhapson method, or the modified Newton-Rhapson method. The latter was

\textsuperscript{21} http://opensees.berkeley.edu/wiki/index.php/Analysis_Commands
selected for the model. This method involves taking tangents of a function iteratively at the proximity of a root in order to narrow in on the root of the function\textsuperscript{22}.

Next, a method of integration must be chosen to move the analysis from one time step to the next using the integrator command. This depends on whether the analysis is static or transient. For static problems, such as applying gravity load, the LoadControl option is the most common, which applies the load in a series of incremental steps defined by the user. For dynamic problems, such as ground motion analysis, the Newmark method is the most common. The Newmark method uses residuals of the momentum equation to predict the state of a system at a future time step based on its state at the current time step using the Taylor formulation. This is often done by approximating the acceleration as an average of or linear interpolation from the previous two data points [9]. It allows the user to define two parameters, gamma and beta, usually configured for either the average acceleration method or the linear acceleration method. For the model, the Newmark integrator was used with the gamma and beta parameters defined as 0.5 and 0.25, representing the average acceleration method.

Finally, the type of analysis must be selected as either static or transient. Static analysis used for cases in which the loads do not vary over time. Transient analysis is used for any time-varying loads, such as cyclic loading or ground motion.

The dynamic analysis was performed by applying all five ground motions to the structure in series, in the same order that they were applied to the structure in the experiment, so that the accumulated damage in the building could best be accounted for. A few seconds of idle time was left in between the ground motions, to allow for any residual vibrations to come to a rest. The analysis was performed at the same time step as the data in the input files, because there was no reason to use a different one.

4 Results

Running a clean analysis of the ground motions proved to be a veritable challenge. Many tests were run before the analysis was able to reach the end without failing to converge. Because the cause of this non-convergence was unclear, trial-and-error methods were required to tackle this problem. The failure did seem to occur consistently in the Kobe earthquake at 100% intensity (Kobe100), which was the most intense of the five patterns. This makes sense in that the large ground accelerations would produce the largest deformations and nonlinear behavior, making failure more likely.

\textsuperscript{22} http://opensees.berkeley.edu/wiki/index.php/Newton_Algorithm
Before modifying any of the model properties, it was decided to test some of the analysis methods if the problem lied therein. Some of the other options for system, algorithm, and integrator were tried, without success. The tolerance for the convergence test was increased by an order and later two of magnitude, without making a difference in the results. The test always failed at the same time step, around 15.5 seconds into the Kobe100 test.

Later, it was decided to create more nodes to allow for more discretization of the elements. This follows from the basic principles of finite element analysis, which say that more, smaller elements should be created in areas with higher anticipated stress concentration [12]. The columns initially were rather long as single elements, at 3m each. They were subsequently divided into 3 elements of 1 m length each. The new node configuration can be seen in Fig. 17. This did not solve the problem.

Using the SeismoSignal program, for the Kobe100 ground motion it was seen in the displacement graph that a permanent lateral drift was produced toward the end of the data. SeismoSignal has the ability to provide baseline corrected data from the initial ground motion data. This means re-centering the data if the original data produces more acceleration in one direction than the other. All data in the tests used herein was the baseline corrected data, however, this did not solve the convergence problem.

Initially, the base of the building was not modeled, because it was assumed to be completely fixed to the shake table, and would not affect the response. However, the author made the error of assuming OpenSees would treat the problem the same way. Adding a massive base to a structure increases the mass, and thus decreases natural frequency, as seen in Eqn. 7. This applies to elastic structures, however, with small displacements, can approximate nonlinear structures.

\[
\omega^2 = \frac{k}{m} \quad \text{(Eqn. – 7)}
\]

The base of the building was thus incorporated into the model as point masses at the base nodes (which were significantly larger than the other node masses). Also, additional base girders were added

![Figure 10: New Node Arrangement](image)
connecting the three base nodes, and a gravity load was applied to these girders [Appendix C]. When tested with the five ground motions in series, the test reached completion, giving a full set of output data. Later, however, when the order of the ground motions was changed, the test failed, indicating that the Kobe100 pattern in combination with this specific structure is really testing Opensees’ computational limits.

The raw output data was graphed in MATLAB. It was immediately visible that after the Kobe 100 pattern, the structure had a permanent lateral displacement that continued throughout the following two patterns [fig. 19]. This indicates that the structure had experienced a permanent deformation. This was eliminated using MATLAB by subtracting the value of this base displacement from all of the data points occurring after the displacements, in order to re-center the graph.

Some of the data values in the displacement output file produced very sharp spikes, sharp enough to represent impossible behavior of the structure. These had to be discounted as noise when choosing the maximum and minimum displacements [fig. 18].

4.1 Output Data
Graphs of the raw output data are presented below, for the patterns of Kobe25, Kobe50, Kobe100, Taka40, and Taka60, respectively. These values were chosen after aberrant spikes were discounted in MATLAB. The values given and the graphs shown for displacement and acceleration are for the roof level. For the values for the other stories, see Appendix D.

Figure 12: Horizontal Displacement at Roof Level
Here it is visible that the red and blue patterns are almost perfect inverses of each other. This indicates that as the building was oscillating from side to side, pivot occurred about the middle column, alternating tension and compression on the two outer columns.

Here it is visible that the left column experienced more permanent deformation than the right column after the Kobe100 pattern. Large strains begin to occur during the Kobe100 pattern, where plastic deformation and damage take place, and continue throughout the Takatori patterns. Some spikes are high enough to not be visible in the graph.

5 Sensitivity Analysis

One initial (and very costly) assumption that was made was that the experimental results of the shake table test would be published soon after the contest closed. At the time of this writing, these results
were unfortunately not yet available. Having these results would have proved of enormous use in seeing exactly where OpenSees model differed from the real structure.

In lieu of this error, it is still possible to assess the accuracy of OpenSees by testing how variations in a few key parameters affect the output.

A good starting point is to see how the different choices of equation system, numberer, solution algorithm, and integrator affect the results. OpenSees provides a handful of choices for each of these. Thus, a sensitivity analysis would involve repeating tests with each of the different options, but changing only one at a time.

First, modifications to the choice of system were made while maintaining all other parameters identical. The three systems tried, UMFPack, SparseGEN, and SparseSYM, all yielded exactly identical output data for roof displacement and acceleration.

Modifications to the algorithm were made, choosing Krylov-Newton and the Broyden algorithms, in addition to the Modified Newton algorithm used for the contest entry. The Krylov-Newton and Broyden yielded very similar results, which were slightly different from the original test. The Krylov-Newton and the Broyden algorithm both produced a maximum negative displacement 6% larger than the Modified-Newton, and a maximum positive displacement equal to that of the Modified-Newton. A graph comparing the Krylov with the original test in the region of maximum displacements is shown in Fig. 23. Overall, choice of algorithm has only slight effect on the results, without significantly modifying maximum and minimum values.

Next, choices of integrator were modified. The Hilber-Hughes-Taylor method with an alpha parameter of 0.5 was tried, and this test failed to converge. The Newmark method was tried with a beta parameter of 1/6, representing the linear acceleration method. This test failed similarly. With these two tests,
decreasing the parameters takes the test out of the “unconditionally stable” range, which explains instability in the test. This goes to show that the Kobe100 test pushes the limits for OpenSees capacities of analysis, and that slight modifications of the integrator parameters make the difference between a successful test and failure.

A significant parameter in the model was the confinement factor \((K_{fc})\) for confined concrete, and, as discussed earlier, the result of several approximations. Several values of \(K_{fc}\) were tried, ranging from 1.0 to 2.0. Some tests passed, many failed without explanation. No systematic mechanism of failure was evident. Tests using values of 1.7, 1.9, and 2.0 passed, while tests using values of 1.4, 1.5, and 1.6 failed. Tests with values of 1.2, 1.29, and 1.37 passed, whereas tests with 1.25, 1.35, and 1.27 failed. The results of the tests that did pass produced a fairly linear result that is shown in Fig. 24. On the horizontal axis is the factor by which \(K_{fc}\) was multiplied by for each test. On the vertical axes, the maximum displacement in the positive direction in the left figure (Max), and the maximum displacement in the negative direction is shown in the right figure (max). The displacement in the positive direction shows a less linear relationship (smaller \(R^2\)). This is likely because the larger displacement induce more nonlinear behavior.

![Figure 17: Max and min displacements with varied Kfc](image)

A test was then performed ignoring second order (P-Delta) effects. This test failed to converge, interestingly enough, despite the fact that P-Delta effects increase the overturning effects on the columns.

Next, the damping ratio was varied. Tests with values lower than 5% all failed. Five tests were successfully completed, with values ranging from 6% to 10%. The result was quite linear, with decreased maximum displacement with increased damping [fig. 25].
A very high susceptibility to test failure was noted with the tensile strength \((f_{ts})\) parameter. A test of \(f_{ts} = 0.145 f'c\) (vs. the original of \(f_{ts} = 0.14 f'c\)) failed to converge. A test of \(f_{ts} = 0.141 f'c\) passed, but showing minimal differences in displacement. A test of \(f_{ts} = 0.143 f'c\) failed, but \(f_{ts} = 0.142 f'c\) passed. Differences from original were minimal. With only three values, creating a sensitivity graph was too difficult, and an attempt at such led to \(R^2\) values of approximately 0.7 when attempting to do a linear fit [fig]. What could be seen vaguely, however, was that increased tensile strength caused a decrease in maximum displacements, which makes sense, as a higher resistance in the elastic range prevents the concrete from cracking and plastic deformation.

Next, tests were done varying the strain at peak stress \((\varepsilon_{p1})\) parameter. These were highly susceptible to unexplained failure similar to the \(Kfc\) tests. Only four successful tests were recorded before the author gave up on trying random values and getting non-convergence. These four ranged from 100% to 120% of the original value. With four points, making a sensitivity graph was pointless, as the \(R^2\) values were too low [fig. 20]. A vague pattern was seen that an increased \(\varepsilon_{p1}\) resulted in increased maximum displacements. This makes sense, because if the concrete deforms more before it reaches peak stress, more displacement will occur without additional damage to the concrete.
Tests were done varying the strain at ultimate stress (\(\epsilon_{su}\)) parameter. Three tests were completed before it was obvious that it didn’t have much of a significant effect on the results [fig. 21]. This is likely because the concrete probably did not often reach the range of crushing, and thus its effects on the test were minimal.

Given the high rate of failure with the Kobe100 pattern, it was decided to repeat and continue the sensitivity analysis with the Takatori60 pattern, with the hopes that less intensity would lead to more meaningful results.

With the Takatori60 pattern, all of the tests passed. From the stress-strain recorders, it is seen that the stresses in the steel don’t even reach 100MPa, and the strains don’t even reach 1E-03. Thus, the steel isn’t anywhere near yielding (345Mpa, 1.7E-03), and the behavior is far closer to linear elastic than it is with the Kobe100 test.

Tests performed modifying \(Kfc\) showed a highly linear relationship with respect to positive displacement, but an almost parabolic relationship with respect to negative displacement [fig. 22].

Tests with \(\epsilon_{st}\) showed a strong linear trend with displacements in both directions [fig. 23]. Tests of \(f_{ts}\) also showed some linearity, but, similar to \(Kfc\), the negative direction was somewhat parabolic [fig. 24]. They at least show that increasing tensile strength reduced maximum displacements, which is good.
Four other parameters were tested. The Max and min graphs for the \( \text{eps}_u \) parameter had slopes of opposite sign, meaning that increasing the parameter increased the maximum displacement in one direction but decreased it in another. This kind of asymmetric effect can be discounted, because these parameters should either increase the building’s resistance against lateral displacements or not, in order to give a meaningful result. A similar effect was noted with the \( K_{res} \) (ratio of ultimate to peak stress) and the \( E_{ts} \) (tensile softening slope) parameters, and thus their graphs have not been included. The last test, which involved modifying the unloading slope (\( \lambda \)), produced graphs with same-direction slopes, although very weak ones [fig. 25].
6 Potential Limiting Aspects of OpenSees in Modeling Real Structures

6.1 Moment Resisting Frames (MRF)

A MRF is a rigid frame which does not contain a significant shear resisting component, such as a shear wall. Its members are connected at joints, which must be adequately strengthened to resist rotation. The RC building in the PEER contest is made of reinforced concrete frames without diagonal or cross-bracing, consisting of girders supporting the floor slab and columns. Under lateral loading, the horizontal girders are unlikely to sustain significant bending forces, because the rotational damage is likely to be concentrated at the joints. Similarly, the columns are unlikely to fail anywhere in mid-length, because of the forces being transferred to the joints. Thus, the behavior and damage sustained at the column-girder joints is likely the most critical part of the analysis.

In the RC building, the exterior joints are reinforced such that the longitudinal steel of the girder is developed into the column and then hooked at 90 degrees [fig. 26]. Additional longitudinal bars with these hooks are provided in the end one-third of the girder. At the interior joints, additional bars are added to the girder in the end one-third of its span passing through the column to the girder on the other side [fig. 26]. Aside from this additional steel, the joints are not dimensioned in any way to resist additional moments, such as by adding extra concrete.

![Figure 26: A snapshot of some of the steel layout in the girder](image)

The most common type of damage occurring at these joints are diagonal cracks in the concrete at the opening part of the joint, as well as spalling of the concrete due to crushing on the closing part of the joint. The modes of failure for various types of joints are outlined neatly in a paper by Uma and Prasad at the India Institute of Technology [12].

\[^{23}[4]\]
OpenSees needs to be able to model the resistance of these joints, as well as deformations, accurately in order to produce a reliable test. Spalling of concrete and plastic deformation of steel at the joints are difficult to model numerically, making this these aspects likely to be large culprits in any deviation of the predicted from the numerical results. These behaviors depend highly on material parameters, and thus a sensitivity analysis of some of these parameters is performed after the contest entry was complete.

6.2 Confined concrete

One shortfall of OpenSees is that it can only model elements (beams and columns) in the longitudinal direction, and thus cannot incorporate the placements of transverse reinforcement in the form of stirrups. Thus arises the need for an approach to approximate the properties of the concrete that would have been confined by the stirrups. Also, the lack of stirrups significantly alters the member’s response to shear. Luckily, with the moment frame direction, the primary modes of failure will be flexure of the columns and rotation at the joints.

An effective technique developed by Mander et al. [10] involves calculating a coefficient by which the compressive strength of the concrete is multiplied if it is confined. This takes into account the type and configuration of the reinforcement. The author performed these calculations in a spreadsheet and found the coefficient to be around 1.3. Calculation of this parameter, however, has many variables, and thus is used in the sensitivity analysis performed after the contest entry was complete.

6.3 From Linear to Nonlinear (Case Studies)

As can be seen from many other attempts to predict nonlinear behavior of structures, it is anything but a trivial task. In 2009, E-Defense of Japan held a similar contest\(^2\), but for the prediction of a steel building instead of a concrete one. The building was a 4-story moment frame building, 2 by 2 bays. Three dampers were installed per floor. Contest entries were evaluated by calculating the errors from different values calculated in each contest entry.

\(^2\) [http://www.blind-analysis.jp/index_e.html](http://www.blind-analysis.jp/index_e.html)
entry, and points were awarded based on the lowest root mean square (RMS) errors for the various values. The entry with the highest number of points was elected winner. The results of the contest are published on the website, with graphs displaying the measured results versus the top three entries.

For example, in the analysis category labeled “2D Analysis with viscous damper,” the values of the winning entry do not match the experimental values exactly [13]. Predicted floor displacements differ from measured values by up to approximately 15% at the upper stories [Fig. 27]. Predicted accelerations differ by over 40% from the measured values at upper stories [Fig. 28]. Story drift angle values differed by over 30% from the measured values [Fig 29]. This clearly illustrates the uncertainty in predicting the behavior of a steel structure, which should, in theory, approximate linear elastic behavior far more closely than a concrete structure, given the material properties of steel.

In 2010, PEER held a blind prediction contest as a precursor to the 2011 contest, but somewhat simpler: the behavioral prediction of a single reinforced concrete column subject to earthquake load. The column, cylindrical, 1.2 m diameter and 7.2 m tall, was intended to represent a single bridge pier. A 220,000 kg concrete block was freely supported atop the column in order to provide large inertial forces. The column was reinforced by longitudinal bars distributed around the perimeter, and these were enclosed by circular hoops. The column was subjected to a series of
six ground motion patterns of varying intensity, similar to the 2011 contest [14]. The results of the experiment are published on the website, as well as graphical plots showing the experimental results compared to the top few winning entries.

For example, for max and min horizontal displacements, it can be seen from the graphs that the best entry different from the experiment by as much as approximately 40%, while one of the winning entries differed by over 60% [Fig. 30, 31]. Predicted accelerations, bending moments, and shear values were somewhat closer to measured, but by looking at the graphs that the results are still quite scattered. Predicted results for curvature have even larger discrepancies [Fig. 32].

Given some of these figures, it is almost believable that such a contest could be won by pure luck, and as such, the evaluation procedure must be extremely well-designed to produce a fair assessment.

6.3.1 Other Experiments in Nonlinear Structural Modeling
As evidenced in literature, many capable, experience researchers have performed experiments in predicting nonlinear behavior of structures, using OpenSees or other similar programs. In a test performed by Xin, Shi, and Baojian [15] on a simple frame structure of high strength concrete reinforced with high strength steel rebar, the results indicated that OpenSees performed well at the beginning of the test, but later, once damage began to accumulate in the structure, the results became more deviant in comparison with the experimental measurements. They found that the degradation in stiffness that occurred over the test was not modeled well in OpenSees, because bond slip occurred in the real

---

25 http://nisee2.berkeley.edu/peer/prediction_contest/?page_id=25
structure. Their conclusion was that OpenSees performs well for small ground vibrations, but for larger earthquakes, it lacks accuracy in modeling more extensive damage.

In a reinforced concrete building test by Dolsek using OpenSees [2], the prediction of displacement in lower stories of the structures was far more accurate than predicting displacements in higher stories, due to the larger displacements and thus higher nonlinearity at the upper stories.

In another RC building test by Sadjadi et al. using IDARC2D software [16], severe yielding was noted in the beam column joints at the first floor. Their conclusions were that the strong-column-weak-beam assumption is valid, meaning that plastic hinges form far more readily in beams than in columns. They also concluded that choosing the parameters for the model was extremely difficult, as the material properties themselves are always somewhat unpredictable.

Tests performed by Hemmati [17] on structures of varying numbers of stories further supports the hypothesis that predictions are less reliable with higher number of stories. These results show that with a 2-story structure, a 5% difference between predicted and experimental is common. However, with a 10-story structure, this difference can be around 20%.

In a study of steel-concrete composite frame structures by Zona et al.[18], it was found that the most critical aspect of numerically predicting the behavior of these structures was to account for the shear deformations of the joints, and that the boundary conditions of joint deformation made almost all of the difference in the results. They also found that proper modeling of damping is also essential.

Frank McKenna, one of OpenSees’ key developers, in a paper outlining the features of OpenSees [19], admits that significant improvements are still required for its use in accurate prediction for performance based design of structures. He attributes much of the uncertainty to the difficulty in choosing the model parameters, and to the fact that real materials behave far more unpredictably than those used in the models.
7 Summary

The results produced by the sensitivity analysis are uniformly messy. The Kobe100 pattern is clearly the most critical for evaluating nonlinear behavior. In the output file of the sequence test, the building displacements are very small in the two patterns leading up to the Kobe100 pattern. In the tests thereafter, the building displacements continue to be large; however, this is likely due to permanent damage and plastic deformations caused during the Kobe100 part. Tests performed with the Takatori patterns alone yield much smaller displacements than when they are performed after the Kobe100 test.

The sensitivity analysis using the Kobe100 pattern was difficult and very time consuming, due to the high probability of a test failing, and the fact that there was no discernible way discovered for choosing the parameters so that the test would pass. In these situations, the analysis is running near the limits of tolerance of convergence. If the original test used for the contest entry passed, it is expected that setting the parameters too far off from these original values might cause test failure. This is why very small modifications were attempted first before making larger ones. However, this hypothesis did not hold true, because the incidences of failure seemed to be arbitrary in relation to the parameter values. This is one indication that analysis in the nonlinear range causes some strange and unexpected behavior in the software. An in-depth investigation into the exact causes of this non-convergence is advanced beyond the scope of this report.

When comparing the sensitivity tests with the Kobe100 and the Taka60 patterns, the Taka60 tests produce far cleaner results, given that behavior remains more within the linear elastic range. The results of the Kobe100 tests are sufficient enough only to make vague conclusions and suggestions, although these conclusions are largely correct. The fact that the various choices of system and algorithm didn’t make much of a difference on the results helps buttress their validity in the way that various methods for solving the same problem lead to the same answer. Of all the parameters, the confinement factor $K_{fc}$ seems to have the highest sensitivity for the Kobe100 test, given the slope of 60.7 and 60.3 in Fig. 17. The fact that these slopes are of similar value shows that the displacement effect is similar in both directions of displacement. However, it must be taken into account that the first graph has a lower correlation coefficient of 0.91 for the linear fit. The second most sensitive parameter for the Kobe100 test is the damping ratio, as can be seen from the high correlation coefficients of 0.99 in Fig. 18. The damping ratio, however, is not a primary parameter, as its value depends on material and dimension properties of the structure. For the other three parameters, as shown in Figs. 19-21, it is difficult to say anything as to their effect given the shapes of the graphs.
For the Taka60 test, the results fail to represent a large portion of the nonlinear behavior that occurred during the Kobe100 pattern, and thus their sensitivity results cannot be applied ad eundem to nonlinear behavior. However, the results make sense with respect to structural principles, in that increasing $Kfc$ and $f$ generally decreased maximum displacements, while increasing peak strain generally increased them.

8 Conclusion

The real challenge that this project alludes to is the understanding of exactly why the combination of processes executed in OpenSees proceeds exactly as it does. This could be the subject of a far more rigorous, advanced study. Overall, these results show that OpenSees is a valid tool for predicting structural behavior, and that its features and possibilities continue to grow through the innovation of many collaborators. As earthquake engineering continues to grow as a field of research it will become more evident exactly why structures behave as they do. The current gap between an experimental test and a computer-simulated model will continue to close, and many of the unexplained aspects of nonlinear behavior will be drawn to attention. Much advancement has been made in the last few decades, however, with the implementation of seismic resistance clauses in various design codes, and with the implementation of dampers and base isolation devices. These last two have added enormous safety to the buildings in which they have been implemented, by reducing the occurrence of large structural deformations. This direction of advancement may also hold enormous future promise in that with sufficient control systems, buildings could be designed to actively counter the ground motion produced by earthquakes, and this would indeed reduce the relevance of predicting the behavior of static structures.
References


Appendix A: Key Construction Drawings

Figure 31: Floor plan of RC Building

Figure 32: Elevation view of RC Building, Frame direction

26 All images taken from [4]
Figure 33: Elevation view of RC Building, Wall direction and Center Columns

Figure 34: Elevation view of both buildings on shake table
Figure 35: Detail of Column and Girder Reinforcement

<table>
<thead>
<tr>
<th>Section</th>
<th>C1</th>
<th>C2</th>
</tr>
</thead>
<tbody>
<tr>
<td>B x D</td>
<td>500 x 500</td>
<td>500 x 500</td>
</tr>
<tr>
<td>Rebar</td>
<td>8-D22</td>
<td>10-D22</td>
</tr>
<tr>
<td>Hoop</td>
<td>2.2-D10@100</td>
<td>2.2-D10@100</td>
</tr>
<tr>
<td>Joint</td>
<td>2.2-D10@140</td>
<td>2.2-D10@140</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Location</th>
<th>End</th>
<th>Center</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4Fl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3Fl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2Fl.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1Fl.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>G1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>End</td>
</tr>
<tr>
<td>RFl.</td>
<td></td>
</tr>
<tr>
<td>4Fl.</td>
<td></td>
</tr>
<tr>
<td>3Fl.</td>
<td></td>
</tr>
<tr>
<td>2Fl.</td>
<td></td>
</tr>
<tr>
<td>1Fl.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>G1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>End</td>
</tr>
<tr>
<td>RFl.</td>
<td></td>
</tr>
<tr>
<td>4Fl.</td>
<td></td>
</tr>
<tr>
<td>3Fl.</td>
<td></td>
</tr>
<tr>
<td>2Fl.</td>
<td></td>
</tr>
<tr>
<td>1Fl.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Section</th>
<th>G1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>End</td>
</tr>
<tr>
<td>RFl.</td>
<td></td>
</tr>
<tr>
<td>4Fl.</td>
<td></td>
</tr>
<tr>
<td>3Fl.</td>
<td></td>
</tr>
<tr>
<td>2Fl.</td>
<td></td>
</tr>
<tr>
<td>1Fl.</td>
<td></td>
</tr>
</tbody>
</table>
## Appendix B: Mass and Weight Table of Building

Table 2.1  Weight of RC specimen.

<table>
<thead>
<tr>
<th>Structural</th>
<th>RC</th>
<th>2.4</th>
<th>t/m²</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RFL</td>
<td>4FL</td>
<td>3FL</td>
</tr>
<tr>
<td>RC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Column</td>
<td>5.4</td>
<td>10.8</td>
<td>10.8</td>
</tr>
<tr>
<td>Girder</td>
<td>16.4</td>
<td>16.4</td>
<td>16.4</td>
</tr>
<tr>
<td>Wall</td>
<td>4.1</td>
<td>8.1</td>
<td>8.1</td>
</tr>
<tr>
<td>Slab</td>
<td>44.1</td>
<td>43.7</td>
<td>43.3</td>
</tr>
<tr>
<td>Beam</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
</tr>
<tr>
<td>Parapet</td>
<td>5.3</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Steel</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Temp. Girder</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Sum [t]</td>
<td>83.3</td>
<td>87.0</td>
<td>86.6</td>
</tr>
</tbody>
</table>

### Non-Structural

<table>
<thead>
<tr>
<th>Steel</th>
<th>Stair</th>
<th>Measurement</th>
<th>Handrail</th>
<th>Machine on the slab</th>
<th>under the slab</th>
<th>RC Base</th>
<th>Ceiling under the slab</th>
<th>Sum [kg]</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>330</td>
<td>360</td>
<td>360</td>
<td>360</td>
<td>0</td>
<td></td>
<td></td>
<td>12040</td>
<td></td>
</tr>
<tr>
<td>Steel</td>
<td>3000</td>
<td>1750</td>
<td>1690</td>
<td>1690</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Handrail</td>
<td>244</td>
<td>271</td>
<td>271</td>
<td>271</td>
<td>197</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Machine</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>on the slab</td>
<td>4633</td>
<td>180</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>under the slab</td>
<td>495</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RC Base</td>
<td>6042</td>
<td>346</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ceiling</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>under the slab</td>
<td>296</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sum</td>
<td>[kg]</td>
<td>12040</td>
<td>4157</td>
<td>2381</td>
<td>2321</td>
<td></td>
<td>1887</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Total

<table>
<thead>
<tr>
<th>RFL</th>
<th>4FL</th>
<th>3FL</th>
<th>2FL</th>
<th>Base</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>95.3</td>
<td>91.2</td>
<td>89.0</td>
<td>88.5</td>
</tr>
</tbody>
</table>

Whole Building [t] 602.4

---

27 Source: [3, p.16]
Appendix C: Supporting Calculations for Mass Allocation

Total slab weight for RFL = 44.1 t (Appendix B)

➔ For half of the floor, that’s 22.05 t

Tributary area for column 1, Ac1 = 4.4m x 5m – (1.7m x 2.3m) – (0.9m x 1.4m) = 16.83 m²

Tributary area for column 2, Ac2 = (5m – 0.9m) x 7.2m = 29.52 m²

Total slab area for half the floor = (16m x 5m) – (0.9m x 10m) – (1.7 x 2.3 x 2) = 63.18 m²

Slab weight carried by column C1 = (16.83/63.18) x 22.05 t = 5.87 t

Slab weight carried by column C2 = (29.52/63.18) x 22.05 t = 10.30 t

Total weight of girders per floor = 16.4 t (, p. 16)

➔ For half the floor, that’s 8.2 t

Volume of G1 = 0.6m x 0.3m x (7.2m – 0.5m) = 1.206 m³

G2 = 0.3m x 0.3m x (3.6m – 0.25 – 1.25) = 0.189 m³

G3 = 0.3m x 0.4m x (3.6m – 0.25) = 0.402 m³

Total girder volume per floor = 2 x G1 + 2*G2 + G3 = 3.192 m³

Girder weight per volume = 8.2 t / 3.192 m³ = 2.57 t/m³

Girder weight carried by C1 = 0.5 * G1 + 0.5 * G2

= (0.5*1.206*2.57) + (0.5*0.189*2.57) = 1.792 t

Girder weight carried by C2 = G1 + G3

= (1.206*2.57) + (0.402*2.57) = 4.13 t

Total weight of beams per floor = 8.0 t (, p.16)

➔ For half the floor, that’s 4 t

Total beam length per half floor = 6* (3.6m – 0.25m) = 20.1m

Beam weight per length = 4t / 20.1m = 0.199 t/m

Beam weight carried by C1 = 1.5*B = 1.5 * (3.6m – 0.25m) * 0.199t/m = 1 t

Beam weight carried by C2 = 3*B = 3* (3.6m – 0.25m) * 0.199t/m = 2 t
Considering the parapet:

Assume the weight is distributed evenly around the perimeter

Total parapet weight on roof = 5.3 t (p. 16)

➔ For half the roof, that’s 2.65 t

Outer perimeter of half of roof = 5m + 5m + 16m + 2*0.9m = 27.8m

Outer edge of tributary area for C2 = 7.2 m

Outer edge of tributary area for C1 = (27.8m – 7.2)/2 = 10.3m

Parapet weight carried by C1 = (10.3m/27.8m) * 2.65 t = 0.982 t

Parapet weight carried by C2 = (7.2m/27.8m) * 2.65 t = 0.686 t

We’ll assume all load form stairway gets taken by C2, as it’s placed along the edge of C2’s tributary area

Stair weight carried by C2 = 0.330t/2 = 0.165 t

For the ceiling weight under RFl, we’ll assume it behaves like the slab, and its load distribution is per tributary area

From Appendix B, total load = 0.296t / 2 = 0.148t

Ceiling weight carried by column C1 = (16.83/71) x 0.148 t = 0.039 t

Ceiling weight carried by column C2 = (29.52/71) x 0.148 t = 0.069 t

For other, more poorly defined loads, such as machines or measuring equipment, I will make the super simple assumption of dividing their masses evenly among the 3 columns.

From Appendix B, total mass of machine, base, and measurement for RFl = ( 4.633 + 0.495 + 6.042 + 0)/2 = 5.85 t

So, weight supported by each column = 5.85t / 3 = 1.862 t

From Appendix B, column mass per floor for RFL = 5.4 t

For each node that’s 5.4/6 = 0.9 t

For total node mass at top of C1, we get 5.23 + 1.792 + 1 + 0.982 + 0.035 + 1.862 + 0.9 = **12.494t**

For total node mass at top of C2, we get 9.17 + 4.13 + 2 + 0.686 + 0.165 + 0.062 + 1.862 + 0.9 = **20.147t**
For the rest of the floors, the calculations are the same, except modifying the values accordingly in the table Appendix B.

The following tables present the masses allocated at each node, and the gravity loads on each girder, obtained through similar calculations:

Masses allocated to top-of-column nodes at each floor:

<table>
<thead>
<tr>
<th>Mass (t)</th>
<th>Roof</th>
<th>4Fl</th>
<th>3Fl</th>
<th>2Fl</th>
<th>Foundation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column 1</td>
<td>12.494</td>
<td>11.050</td>
<td>10.700</td>
<td>10.624</td>
<td>26.289</td>
</tr>
<tr>
<td>Column 2</td>
<td>20.147</td>
<td>18.942</td>
<td>18.553</td>
<td>18.426</td>
<td>58.215</td>
</tr>
</tbody>
</table>

Gravity loads assigned to x-direction girders at each floor:

<table>
<thead>
<tr>
<th>Load (N/mm)</th>
<th>Roof</th>
<th>4Fl</th>
<th>3Fl</th>
<th>2Fl</th>
<th>Foundation</th>
<th>Column (vertical)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girder</td>
<td>35.187</td>
<td>34.179</td>
<td>33.172</td>
<td>32.910</td>
<td>85.149</td>
<td>5.886</td>
</tr>
</tbody>
</table>
Appendix D: Results for Contest Submission

Floor Masses (kg):

<table>
<thead>
<tr>
<th>Level</th>
<th>Mass</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roof</td>
<td>90269.0</td>
</tr>
<tr>
<td>4th Floor</td>
<td>82086.0</td>
</tr>
<tr>
<td>3rd Floor</td>
<td>79101.0</td>
</tr>
<tr>
<td>2nd Floor</td>
<td>79350.0</td>
</tr>
</tbody>
</table>

NOTE: Values should be reported in absolute terms

1 Relative floor displacement (mm) at:

<table>
<thead>
<tr>
<th>Level</th>
<th>EQ1</th>
<th>EQ3</th>
<th>EQ4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>Roof</td>
<td>15.9</td>
<td>239.2</td>
</tr>
<tr>
<td>maximum</td>
<td>Roof</td>
<td>14.4</td>
<td>161.2</td>
</tr>
<tr>
<td>Maximum</td>
<td>4th Floor</td>
<td>13.1</td>
<td>222.7</td>
</tr>
<tr>
<td>maximum</td>
<td>4th Floor</td>
<td>11.9</td>
<td>148.6</td>
</tr>
<tr>
<td>Maximum</td>
<td>3rd Floor</td>
<td>8.7</td>
<td>158.2</td>
</tr>
<tr>
<td>maximum</td>
<td>3rd Floor</td>
<td>7.9</td>
<td>113.6</td>
</tr>
<tr>
<td>Maximum</td>
<td>2nd Floor</td>
<td>3.5</td>
<td>71.0</td>
</tr>
<tr>
<td>maximum</td>
<td>2nd Floor</td>
<td>3.2</td>
<td>55.2</td>
</tr>
</tbody>
</table>

2 Absolute floor accelerations (g) at:

<table>
<thead>
<tr>
<th>Level</th>
<th>EQ1</th>
<th>EQ3</th>
<th>EQ4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>Roof</td>
<td>0.367</td>
<td>1.165</td>
</tr>
<tr>
<td>maximum</td>
<td>Roof</td>
<td>0.305</td>
<td>0.972</td>
</tr>
<tr>
<td>Maximum</td>
<td>4th Floor</td>
<td>0.287</td>
<td>1.203</td>
</tr>
<tr>
<td>maximum</td>
<td>4th Floor</td>
<td>0.254</td>
<td>0.947</td>
</tr>
<tr>
<td>Maximum</td>
<td>3rd Floor</td>
<td>0.178</td>
<td>0.893</td>
</tr>
<tr>
<td>maximum</td>
<td>3rd Floor</td>
<td>0.166</td>
<td>0.684</td>
</tr>
<tr>
<td>Maximum</td>
<td>2nd Floor</td>
<td>0.067</td>
<td>0.706</td>
</tr>
<tr>
<td>maximum</td>
<td>2nd Floor</td>
<td>0.066</td>
<td>0.515</td>
</tr>
</tbody>
</table>

3 Maximum axial force (kN) in the columns:

<table>
<thead>
<tr>
<th>Level</th>
<th>EQ1</th>
<th>EQ3</th>
<th>EQ4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>1st Floor</td>
<td>1685.9</td>
<td>1681.0</td>
</tr>
<tr>
<td>maximum</td>
<td>1st Floor</td>
<td>697.5</td>
<td>401.0</td>
</tr>
</tbody>
</table>

4 Average axial strain at the columns over 530 mm from the base:

<table>
<thead>
<tr>
<th>Level</th>
<th>EQ1</th>
<th>EQ3</th>
<th>EQ4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum</td>
<td>96.7</td>
<td>3545.5</td>
<td>606.0</td>
</tr>
<tr>
<td>maximum</td>
<td>85.6</td>
<td>130.1</td>
<td>80.8</td>
</tr>
</tbody>
</table>

5 Mode of column failure: Flexural failure
Appendix E: Ground Motion Data

Figure 36: Ground Motion for Kobe 25%

Figure 37: Ground Motion for Kobe 50%

Figure 38: Ground Motion for Kobe 100%

Figure 39: Ground Motion for Takatori 40%

Figure 40: Ground Motion for Takatori 60%
Appendix F: Source Code

# MAIN FILE

wipe all; # clear memory of all past model definitions
model BasicBuilder -ndm 2 -ndf 3; # Define the model builder, ndm=#dimension, ndf=#dofs
set dataDir Data; # set up name of data directory (you can remove this)
file mkdir $dataDir; # create data directory

# Define Units
# basic units of 1 shall be N, t, mm, s, MPa, sec
set Negligible 1e-10; # small number to avoid problems with zero
set M 1000000; # Mega

# Define Materials
# REINFORCED CONCRETE PARAMETERS
# nominal concrete compressive strength
set fc -27; # MPa, CONCRETE Compressive Strength (+Tension, -Compression)

# confined concrete
set Kfc 1.3; # ratio of confined to unconfined concrete strength
set Kres 0.2; # ratio of residual/ultimate to maximum stress
set fc1C [expr $Kfc*$fc]; # MPa, CONFINED concrete (mander model), maximum stress
set EcC 31200; # MPa, Concrete Elastic Modulus
set eps1C [expr 2.*fc1C/$EcC]; # strain at maximum stress
set fc2C [expr $Kres*fc1C]; # MPa, ultimate stress
set eps2C [expr 20*eps1C]; # strain at ultimate stress
set lambda 0.1; # ratio between unloading slope at $eps2 and initial slope $Ec

# unconfined concrete
set fc1U $fc; # UNCONFINED concrete (todeschini parabolic model), maximum stress
set EcU $EcC; # MPa, Concrete Elastic Modulus
set eps1U [expr 2.*fc1U/$EcU]; # strain at maximum stress
set fc2U [expr $Kres*fc1U]; # ultimate stress
set eps2U -0.01; # strain at ultimate stress

# tensile-strength properties
set ftC [expr -0.14*fc1C]; # tensile strength +tension
set ftU [expr -0.14*fc1U]; # tensile strength +tension
set Ets [expr ftU/0.002]; # tension softening stiffness

# set up library of materials
set IDconcCore 1
set IDconcCover 2
set IDconcCoreE 5
set IDconcCoverE 6
uniaxialMaterial Concrete02 $IDconcCore $fc1C $eps1C $fc2C $eps2C $lambda $ftC $Ets; # Core concrete (confined)
uniaxialMaterial Concrete02 $IDconcCover $fc1U $eps1U $fc2U $eps2U $lambda $ftU $Ets; # Cover concrete (unconfined)

# REINFORCING STEEL PARAMETERS
set Fy1 345; # STEEL yield stress D22, 345 MPa
set Fy2 295; # STEEL yield stress D10, 295 MPa
set Es 200000.; # modulus of steel, 200000 MPa
set Bs 0.01; # strain-hardening ratio
set R0 18; # control the transition from elastic to plastic branches
set cR1 0.925; # control the transition from elastic to plastic branches
set cR2 0.15; # control the transition from elastic to plastic branches
set IDSteel 3
set IDSteel2 4
uniaxialMaterial Steel02 $IDSteel $Fy1 $Es $Bs $R0 $cR1 $cR2
uniaxialMaterial Steel02 $IDSteel2 $Fy2 $Es $Bs $R0 $cR1 $cR2

source BuildRCrectSection.tcl; # procedure for defining RC fiber section, see below

# Define Geometry and Nodes

# define GEOMETRY

set LCol [expr 1000]; # column height (mm) divided by 3
set LBeam [expr 2400]; # beam length (mm) divided by 3, division of reinforcement for girders

# calculate all possible coordinates for nodes:
set X1 0.;
set X2 [expr $X1 + $LBeam];
set X3 [expr $X2 + $LBeam];
set X4 [expr $X3 + $LBeam];
set X5 [expr $X4 + $LBeam];
set X6 [expr $X5 + $LBeam];
set X7 [expr $X6 + $LBeam];
set Y1 0.;
set Y2 [expr $Y1 + $LCol];
set Y3 [expr $Y2 + $LCol];
set Y4 [expr $Y3 + $LCol];
set Y5 [expr $Y4 + $LCol];
set Y6 [expr $Y5 + $LCol];
set Y7 [expr $Y6 + $LCol];
set Y8 [expr $Y7 + $LCol];
set Y9 [expr $Y8 + $LCol];
set Y10 [expr $Y9 + $LCol];
set Y11 [expr $Y10 + $LCol];
set Y12 [expr $Y11 + $LCol];
set Y13 [expr $Y12 + $LCol];

# define coordinates of all 55 nodes
node 11 $X1 $Y1; # foundation isnt divided into 3, define 3 nodes for bottom layer
node 14 $X4 $Y1;
node 17 $X7 $Y1;
node 21 $X1 $Y2; # column below first floor, no girder
node 24 $X4 $Y2;
node 27 $X7 $Y2;
node 31 $X1 $Y3;
node 34 $X4 $Y3;
node 37 $X7 $Y3;
node 41 $X1 $Y4; # floor 2, 7 nodes for girder
node 42 $X2 $Y4;
node 43 $X3 $Y4;
node 44 $X4 $Y4;
node 45 $X5$ $Y4$;
node 46 $X6$ $Y4$;
node 47 $X7$ $Y4$;
node 51 $X1$ $Y5$; # column below 3FL, no girder
node 54 $X4$ $Y5$;
node 57 $X7$ $Y5$;
node 61 $X1$ $Y6$;
node 64 $X4$ $Y6$;
node 67 $X7$ $Y6$;
node 71 $X1$ $Y7$; # floor 3, 7 nodes
node 72 $X2$ $Y7$;
node 73 $X3$ $Y7$;
node 74 $X4$ $Y7$;
node 75 $X5$ $Y7$;
node 76 $X6$ $Y7$;
node 77 $X7$ $Y7$;
node 81 $X1$ $Y8$; # column below 4FL, no girder
node 84 $X4$ $Y8$;
node 87 $X7$ $Y8$;
node 91 $X1$ $Y9$;
node 94 $X4$ $Y9$;
node 97 $X7$ $Y9$;
node 101 $X1$ $Y10$; # floor 4, 7 nodes
node 102 $X2$ $Y10$;
node 103 $X3$ $Y10$;
node 104 $X4$ $Y10$;
node 105 $X5$ $Y10$;
node 106 $X6$ $Y10$;
node 107 $X7$ $Y10$;
node 111 $X1$ $Y11$; # column below RFL, no girder
node 114 $X4$ $Y11$;
node 117 $X7$ $Y11$;
node 121 $X1$ $Y12$;
node 124 $X4$ $Y12$;
node 127 $X7$ $Y12$;
node 131 $X1$ $Y13$; # roof girder, 7 nodes
node 132 $X2$ $Y13$;
node 133 $X3$ $Y13$;
node 134 $X4$ $Y13$;
node 135 $X5$ $Y13$;
node 136 $X6$ $Y13$;
node 137 $X7$ $Y13$;

# BOUNDARY CONDITIONS
fix 11 1 1 1; # fix column bases against translation and rotation
fix 14 1 1 1;
fix 17 1 1 1;

# Define Sections
# define section tags:
set ColSecCenterTag 1; # interior column
set ColSecEndsTag 2; # exterior columns
set BeamSecCenterTag 3; # center of girder, all floors
set BeamSecEnd4FlTag 4; # ends of girder, floor 4 and roof
set BeamSecEnd3FlTag 5; # ends of girder, floor 2
set BeamSecEnd2FlTag 6; # ends of girder, floor 1
# Section Properties:
set HCol [expr 0.5*sqrt($M)]; # square-Column width
set BCol $HCol
set HBream [expr 0.6*sqrt($M)]; # Beam depth -- perpendicular to bending axis
set BBream [expr 0.3*sqrt($M)]; # Beam width -- parallel to bending axis

# FIBER SECTION properties
# Column section geometry:
set cover 50; # cover to be used for all members for now
set numBarsHorCol1 3; # number of longitudinal-reinforcement bars on top/bottom layer
set numBarsHorCol2 4; # number of longitudinal-reinforcement bars on top/bottom layer for column 2
set numBarsIntCol 2; # TOTAL number of reinforcing bars on the intermediate layers
set barAreaCol 387; # longitudinal-reinforcement bar area of a D22, only one used in columns, in m^2

# Girder section geometry:
set numBarsBotBeam 3; # number of longitudinal-reinforcement bars on bottom and top of center girder
set numBarsTopBeam1 4; # number of longitudinal-reinforcement bars on top layer of end girder floor 2
set numBarsTopBeam2 5; # number of longitudinal-reinforcement bars on top layer of end girder floor 3
set numBarsTopBeam3 6; # number of longitudinal-reinforcement bars on top layer of end girder floor 4 and roof
set numBarsIntBeam 2; # TOTAL number of reinforcing bars on the intermediate layers
set barAreaBeam 71; # longitudinal-reinforcement bar area of D10, used for skin reinf in girder
set nfCoreY 20; # number of fibers in the core patch in the y direction
set nfCoreZ 20; # number of fibers in the core patch in the z direction
set nfCoverY 5; # number of fibers in the cover patches with long sides in the y direction
set nfCoverZ 5; # number of fibers in the cover patches with long sides in the z direction

# rectangular section with one layer of steel evenly distributed around the perimeter and a confined core.
BuildRCrectSection $ColSecCenterTag $HCol $BCol $cover $cover $IDconcCore $IDconcCover $IDSteel $IDSteel $numBarsHorCol12 $barAreaCol $numBarsHorCol2 $barAreaCol $numBarsIntCol $barAreaCol $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
BuildRCrectSection $ColSecEndsTag $HCol $BCol $cover $cover $IDconcCore $IDconcCover $IDSteel $IDSteel $numBarsBotBeam $barAreaCol $numBarsBotBeam $barAreaCol $numBarsIntBeam $barAreaCol $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
BuildRCrectSection $BeamSecCenterTag $HBeam $BBeam $cover $cover $IDconcCore $IDconcCover $IDSteel $IDSteel $numBarsTopBeam1 $barAreaCol $numBarsBotBeam $barAreaCol $numBarsIntBeam $barAreaCol $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
BuildRCrectSection $BeamSecEnd4FlTag $HBeam $BBeam $cover $cover $IDconcCore $IDconcCover $IDSteel $IDSteel $numBarsTopBeam2 $barAreaCol $numBarsBotBeam $barAreaCol $numBarsIntBeam $barAreaCol $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
BuildRCrectSection $BeamSecEnd3FlTag $HBeam $BBeam $cover $cover $IDconcCore $IDconcCover $IDSteel $IDSteel $numBarsTopBeam3 $barAreaCol $numBarsBotBeam $barAreaCol $numBarsIntBeam $barAreaCol $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
BuildRCrectSection $BeamSecEnd2FlTag $HBeam $BBeam $cover $cover $IDconcCore $IDconcCover $IDSteel $IDSteel $numBarsTopBeam4 $barAreaCol $numBarsBotBeam $barAreaCol $numBarsIntBeam $barAreaCol $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ

# Define Elements
# set up geometric transformations of element
# separate columns and beams, in case of P-Delta analysis for columns
set IDColTransf 1; # all columns
set IDBeamTransf 2; # all beams
set ColTransfType PDelta; # options, Linear PDelta Corotational
gemTransf $ColTransfType $IDColTransf ; # only columns can have PDelta effects (gravity effects)
gemTransf Linear $IDBeamTransf

# Define Beam-Column Elements
set np 5; # number of Gauss integration points for nonlinear curvature distribution-- np=2 for linear distribution ok
element dispBeamColumn 111 11 21 $np $ColSecCenterTag $IDColTransf; # level 1-2
element dispBeamColumn 112 14 24 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 113 17 27 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 121 21 31 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 122 24 34 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 123 27 37 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 131 31 41 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 132 34 44 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 133 37 47 $np $ColSecCenterTag $IDColTransf;
          # level 2-3
element dispBeamColumn 141 41 51 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 142 44 54 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 143 47 57 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 151 51 61 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 152 54 64 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 153 57 67 $np $ColSecCenterTag $IDColTransf;
          # level 3-4
element dispBeamColumn 161 61 71 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 162 64 74 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 163 67 77 $np $ColSecCenterTag $IDColTransf;
          # level 4-R
element dispBeamColumn 171 71 81 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 172 74 84 $np $ColSecCenterTag $IDColTransf;
element dispBeamColumn 173 77 87 $np $ColSecCenterTag $IDColTransf;
          #BEAMS
element dispBeamColumn 211 41 42 $np $BeamSecEnd2FlTag $IDBeamTransf; # level 2
element dispBeamColumn 212 42 43 $np $BeamSecCenterTag $IDBeamTransf;
          # level 3
element dispBeamColumn 221 71 72 $np $BeamSecEnd3FlTag $IDBeamTransf;
element dispBeamColumn 224 74 75 $np $BeamSecEnd3FlTag $IDBeamTransf;
element dispBeamColumn 225 75 76 $np $BeamSecCenterTag $IDBeamTransf;
element dispBeamColumn 226 76 77 $np $BeamSecEnd3FlTag $IDBeamTransf;

# level 4

element dispBeamColumn 231 101 102 $np $BeamSecEnd4FlTag $IDBeamTransf;  
element dispBeamColumn 232 102 103 $np $BeamSecCenterTag $IDBeamTransf;  
element dispBeamColumn 233 103 104 $np $BeamSecEnd4FlTag $IDBeamTransf;  
element dispBeamColumn 234 104 105 $np $BeamSecEnd4FlTag $IDBeamTransf;  
element dispBeamColumn 235 105 106 $np $BeamSecCenterTag $IDBeamTransf;  
element dispBeamColumn 236 106 107 $np $BeamSecEnd4FlTag $IDBeamTransf;  

# level R

element dispBeamColumn 241 131 132 $np $BeamSecEnd4FlTag $IDBeamTransf;  
element dispBeamColumn 242 132 133 $np $BeamSecCenterTag $IDBeamTransf;  
element dispBeamColumn 243 133 134 $np $BeamSecEnd4FlTag $IDBeamTransf;  
element dispBeamColumn 244 134 135 $np $BeamSecEnd4FlTag $IDBeamTransf;  
element dispBeamColumn 245 135 136 $np $BeamSecCenterTag $IDBeamTransf;  
element dispBeamColumn 246 136 137 $np $BeamSecEnd4FlTag $IDBeamTransf;  

element dispBeamColumn 247 11 14 $np $BeamSecEnd4FlTag $IDBeamTransf;  
element dispBeamColumn 248 14 17 $np $BeamSecEnd4FlTag $IDBeamTransf;  

# Define Node Masses

set QdlBeamRFl 35.18657;  # dead load distributed along beam, N/mm.  
set QdlBeam4Fl 34.1791;  # dead load distributed along beam4fl.  
set QdlBeam3Fl 33.17164;  # dead load distributed along beam3fl.  
set QdlBeam2Fl 32.91045;  # dead load distributed along beam2fl.  
set QdlBeamBFl 85.14925;  # dead load distributed along beam2fl.  

set QdlCol 5.886;  # column section weight per length

set nodeMassRFlC1 12.49367;  # in t  
set nodeMassRFlC2 20.14714;  
set nodeMass4FlC1 11.05013;  
set nodeMass4FlC2 18.9427;  
set nodeMass3FlC1 10.70086;  
set nodeMass3FlC2 18.55326;  
set nodeMass2FlC1 10.62426;  
set nodeMass2FlC2 18.42645;  
set nodeMassBFlC1 26.28896;  
set nodeMassBFlC2 58.21492;  

# assign masses to the nodes that the columns are connected to
mass 131 $nodeMassRF1C1 $Negligible $Negligible;  # level R  
mass 134 $nodeMassRF1C2 $Negligible $Negligible;  
mass 137 $nodeMassRF1C1 $Negligible $Negligible;  
mass 101 $nodeMass4FlC1 $Negligible $Negligible;  # level 4  
mass 104 $nodeMass4FlC2 $Negligible $Negligible;  
mass 107 $nodeMass4FlC1 $Negligible $Negligible;  
mass 71 $nodeMass3FlC1 $Negligible $Negligible;  # level 3  
mass 74 $nodeMass3FlC2 $Negligible $Negligible;  
mass 77 $nodeMass3FlC1 $Negligible $Negligible;  
mass 41 $nodeMass2FlC1 $Negligible $Negligible;  # level 2  
mass 44 $nodeMass2FlC2 $Negligible $Negligible;  
mass 47 $nodeMass2FlC1 $Negligible $Negligible;  
mass 11 $nodeMassBF1C1 $Negligible $Negligible;  # level B  
mass 14 $nodeMassBF1C2 $Negligible $Negligible;  
mass 17 $nodeMassBF1C1 $Negligible $Negligible;  

# Define Recorders

# Define Node Masses

set QdlBeamRFl 35.18657;  # dead load distributed along beam, N/mm.  
set QdlBeam4Fl 34.1791;  # dead load distributed along beam4fl.  
set QdlBeam3Fl 33.17164;  # dead load distributed along beam3fl.  
set QdlBeam2Fl 32.91045;  # dead load distributed along beam2fl.  
set QdlBeamBFl 85.14925;  # dead load distributed along beam2fl.  

set QdlCol 5.886;  # column section weight per length

set nodeMassRFlC1 12.49367;  # in t  
set nodeMassRFlC2 20.14714;  
set nodeMass4FlC1 11.05013;  
set nodeMass4FlC2 18.9427;  
set nodeMass3FlC1 10.70086;  
set nodeMass3FlC2 18.55326;  
set nodeMass2FlC1 10.62426;  
set nodeMass2FlC2 18.42645;  
set nodeMassBFlC1 26.28896;  
set nodeMassBFlC2 58.21492;  

# assign masses to the nodes that the columns are connected to
mass 131 $nodeMassRF1C1 $Negligible $Negligible;  # level R  
mass 134 $nodeMassRF1C2 $Negligible $Negligible;  
mass 137 $nodeMassRF1C1 $Negligible $Negligible;  
mass 101 $nodeMass4FlC1 $Negligible $Negligible;  # level 4  
mass 104 $nodeMass4FlC2 $Negligible $Negligible;  
mass 107 $nodeMass4FlC1 $Negligible $Negligible;  
mass 71 $nodeMass3FlC1 $Negligible $Negligible;  # level 3  
mass 74 $nodeMass3FlC2 $Negligible $Negligible;  
mass 77 $nodeMass3FlC1 $Negligible $Negligible;  
mass 41 $nodeMass2FlC1 $Negligible $Negligible;  # level 2  
mass 44 $nodeMass2FlC2 $Negligible $Negligible;  
mass 47 $nodeMass2FlC1 $Negligible $Negligible;  
mass 11 $nodeMassBF1C1 $Negligible $Negligible;  # level B  
mass 14 $nodeMassBF1C2 $Negligible $Negligible;  
mass 17 $nodeMassBF1C1 $Negligible $Negligible;  

# Define Recorders
# Define Gravity Loads and Analysis

```
puts "T1 = $T1 s"
puts "T2 = $T2 s"
```

# Define Gravity Loads and Analysis

```
# GRAVITY LOADS # define gravity load applied to beams and columns -- eleLoad
# applies loads in local coordinate axis
pattern Plain 101 Linear {
  eleLoad -ele 211 212 213 214 215 216 -type -beamUniform -$QdlBeam2Fl;
  # beams level 2 (in -ydirection)
  eleLoad -ele 221 222 223 224 225 226 -type -beamUniform -$QdlBeam3Fl;
  eleLoad -ele 231 232 233 234 235 236 -type -beamUniform -$QdlBeam4Fl;
  eleLoad -ele 241 242 243 244 245 246 -type -beamUniform -$QdlBeamRF1;
  eleLoad -ele 251 252 -type -beamUniform -$QdlBeamBF1;
  eleLoad -ele 111 112 113 121 122 123 131 132 133 -type -beamUniform $Negligible - $QdlCol; #QdlCol; columns level 1-2 (in -direction)
  eleLoad -ele 141 142 143 151 152 153 161 162 163 -type -beamUniform $Negligible - $QdlCol;
  eleLoad -ele 171 172 173 181 182 183 191 192 193 -type -beamUniform $Negligible - $QdlCol;
  eleLoad -ele 1101 1102 1103 1111 1112 1113 1121 1122 1123 -type -beamUniform $Negligible - $QdlCol;
}
```

# Gravity-analysis parameters -- load-controlled static analysis

```
set Tol 1.0e-6; # convergence tolerance for test
constraints Plain; # how it handles boundary conditions
numberer RCM; # renumber dof's to minimize band-width (optimization), if you want system BandGeneral; # how to store and solve the system of equations in the analysis
(large model: try UmfPack)
```
test NormDispIncr $Tol 6; # determine if convergence has been achieved at the end of an iteration step
algorithm Newton; # use Newton's solution algorithm: updates tangent stiffness at every iteration
set NstepGravity 10; # apply gravity in 10 steps
set DGravity [expr 1./$NstepGravity]; # first load increment;
integrator LoadControl $DGravity; # determine the next time step for an analysis
analysis Static; # define type of analysis static or transient
analyze $NstepGravity; # apply gravity

# ----------------------------------------------- maintain constant gravity loads
loadConst -time 0.0
puts "Model Built"

#######################################################
# ANALYSIS
#######################################################

#DisplayModel2D DeformedShape 5;
# ----------- define & apply damping
# RAYLEIGH damping parameters, Where to put M/K-prop damping, switches
(http://opensees.berkeley.edu/OpenSees/manuals/usermanual/1099.htm)
# D=$alphaM*M + $betaKcurr*Kcurrent + $betaKcomm*KlastCommit + $beatKinit*Kinitial
set xDamp 0.05; # damping ratio
set MpropSwitch 1.0;
set KcurrSwitch 0.0;
set KcommSwitch 1.0;
set KinitSwitch 0.0;
set alphaM [expr $MpropSwitch*$xDamp*(2*$omegaI*$omegaJ)/($omegaI+$omegaJ)]; # M-prop. damping; D = alphaM*M
set betaKcurr [expr $KcurrSwitch*2.*$xDamp/($omegaI+$omegaJ)]; # current-K; +beatKcurr*KCurrent
set betaKcomm [expr $KcommSwitch*2.*$xDamp/($omegaI+$omegaJ)]; # last-committed K; +betaKcomm*KlastCommit
set betaKinit [expr $KinitSwitch*2.*$xDamp/($omegaI+$omegaJ)]; # initial-K; +beatKinit*Kini
rayleigh $alphaM $betaKcurr $betaKinit $betaKcomm; # RAYLEIGH damping

# define ground motion parameters
set patternID 1; # load pattern ID
set GMdirection 1; # ground motion direction (1 = x)
set GMfile "GMSeries.txt"; # ground motion filename, unit: mm/sec2
set dt 0.005; # timestep of input GM file
set Scalefact 1000; # ground motion scaling factor, multiply by 1000 because we are dealing in mm, and GM file is in m

# define the acceleration series for the ground motion
# syntax: "Series -dt $timestep_of_record -filePath $filename_with_acc_history -factor $scale_record_by_this_amount
set accelSeries "Series -dt $dt -filePath $GMfile -factor $Scalefact";

# create load pattern: apply acceleration to all fixed nodes with UniformExcitation
# command: pattern UniformExcitation $patternID $GMdir -accel $timeSeriesID
pattern UniformExcitation $patternID $GMdirection -accel $accelSeries;

# define dynamic analysis parameters
set dt_analysis 0.005; # timestep of analysis
wipeAnalysis; # destroy all components of the Analysis object, i.e. any
objects created with system, numberer, constraints, integrator, algorithm, and
analysis commands
constraints Transformation; # how it handles boundary conditions
numberer RCM; # renumber dof's to minimize band-width (optimization)
system UmfPack; # how to store and solve the system of equations in the analysis
test NormDispIncr $Tol 50 5; # determine if convergence has been achieved at the
end of an iteration step
algorithm ModifiedNewton; # use Newton's solution algorithm: updates tangent
stiffness at every iteration
integrator Newmark 0.5 0.25; # uses Newmark's average acceleration method to
compute the time history
analysis Transient; # type of analysis: transient or static
set NumSteps 10248; # number of steps in analysis

# perform the dynamic analysis and display whether analysis was successful
set ok [analyze $NumSteps $dt_analysis]; # ok = 0 if analysis was completed
if {$ok == 0} {
puts "Dynamic analysis complete";
} else {
puts "Dynamic analysis did not converge";
} puts "Ground Motion Done. End Time: [getTime]"

# FILE: BuildRCrectSection.tcl---------------------------------------------

###############################################################################
# Build fiber rectangular RC section, 1 steel layer top, 1 bot, 1 skin, confined
core
# Define a procedure which generates a rectangular reinforced concrete section
# with one layer of steel at the top & bottom, skin reinforcement and a
# confined core.
# by: Silvia Mazzoni, 2006; adapted from Michael H. Scott, 2003
# Formal arguments
# id - tag for the section that is generated by this procedure
# HSec - depth of section, along local-y axis
# BSec - width of section, along local-x axis
# cH - distance from section boundary to neutral axis of reinforcement
# cB - distance from section boundary to side of reinforcement
# coreID - material tag for the core patch
# coverID - material tag for the cover patches
# steelID - material tag for the reinforcing steel
# steelID2 - material tag for second reinforcing steel
# numBarsTop - number of reinforcing bars in the top layer
# numBarsBot - number of reinforcing bars in the bottom layer
# numBarsIntTot - TOTAL number of reinforcing bars on the intermediate layers,
symmetric
# about X axis and 2 bars per layer-- needs to be an even integer
# barAreaTop - cross-sectional area of each reinforcing bar in top layer
# barAreaBot - cross-sectional area of each reinforcing bar in bottom layer
# barAreaInt - cross-sectional area of each reinforcing bar in intermediate layer
# nfCoreY - number of fibers in the core patch in the y direction
# nfCoreX - number of fibers in the core patch in the X direction

52
# nfCoverY - number of fibers in the cover patches with long sides in the y direction
# nfCoverX - number of fibers in the cover patches with long sides in the X direction
#
# Notes
# The core concrete ends at the NA of the reinforcement
# The center of the section is at (0,0) in the local axis system
#

proc BuildRCrectSection {id HSec BSec coverH coverB coreID coverID steelID steelID2 numBarsTop barAreaTop numBarsBot barAreaBot numBarsIntTot barAreaInt nfCoreY nfCoreX nfCoverY nfCoverX} {

# BuildRCrectSection $id $HSec $BSec $coverH $coverB $coreID $coverID $steelID $steelID2 $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreX $nfCoverY $nfCoverX

################################################

set coverY [expr $HSec/2.0]; # The distance from the section x-axis to the edge of
# the cover concrete -- outer edge of cover concrete
set coverX [expr $BSec/2.0]; # The distance from the section y-axis to the edge of
# the cover concrete -- outer edge of cover concrete
set coreY [expr $coverY-$coverH]; # The distance from the section x-axis to the
# edge of the core concrete -- edge of the core concrete/inner edge of cover concrete
set coreX [expr $coverX-$coverB]; # The distance from the section y-axis to the
# edge of the core concrete -- edge of the core concrete/inner edge of cover concrete

set numBarsInt [expr $numBarsIntTot/2]; # number of intermediate bars per side

# Define the fiber section
section fiberSec $id {

# Define the core patch
patch quadr $coreID $nfCoreX $nfCoreY $coreY $coreX $coreY $coreX $coreY $coreX

# Define the four cover patches
patch quadr $coverID 2 $nfCoverY $coverY $coverX $coreY $coreX $coreY $coreX $coreY $coverX

patch quadr $coverID 2 $nfCoverX $coverY $coverX $coreY $coreX $coreY $coreX $coreY $coreX

patch quadr $coverID $nfCoverX 2 $coverY $coreY $coreX $coreY $coreX $coverY $coverX $coreY $coreX

patch quadr $coverID $nfCoverX 2 $coverY $coreY $coreX $coreY $coreX $coverY $coverX $coreY $coverX

# define reinforcing layers
layer straight $steelID2 $numBarsInt $barAreaInt $coreY $coreX $coreY $coreX $coreY $coreX $coreY $coreX $coreX

# intermediate skin reinf. +X
layer straight $steelID2 $numBarsInt $barAreaInt $coreY $coreX $coreY $coreX $coreY $coreX $coreY $coreX $coreX

# intermediate skin reinf. -X
layer straight $steelID $numBarsTop $barAreaTop $coreY $coreX $coreY $coreX $coreY $coreX $coreY $coreX $coreX

# top layer reinforcement
layer straight $steelID $numBarsBot $barAreaBot $coreY $coreX $coreY $coreX $coreY $coreX $coreY $coreX $coreX

# bottom layer reinforcement
}

# end of fibersection definition

}; # end of procedure

}; # end of procedure