جامعة بيرزيت

BIRZEIT UNIVERSITY

DERIVATION OF RESPONSE MODIFICATION FACTOR (R) FOR REINFORCED CONCRETE BUILDINGS IN PALESTINE

BY

MOHAMMAD SAFI

Master’s Thesis

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Civil Engineering
in the Department of Civil and Environmental Engineering,
Birzeit University, 2022

Birzeit, Palestine

Adviser:

Dr.-Eng. Ghada Karaki

APPROVAL OF THE THESIS:

DERIVATION OF RESPONSE MODIFICATION FACTOR (R) FOR REINFORCED CONCRETE BUILDINGS IN PALESTINE

Submitted by Mohammad Safi in partial fulfillment of the requirements for the degree of Master of Science in Civil Engineering, Birzeit University,

This Thesis was defended successfully on 16/7/2022 and approved by:

Defense committee members	**Signature**
Assoc. Prof. Dr. Ghada Karaki Supervisor and Head of the Department, Civil Engineering, Birzeit University	……………………………
Assoc. Prof. Dr. Khalil Qatu Civil Engineering, Birzeit University	……………………………
Assoc. Prof. Dr. Munther Ibrahim Civil Engineering, An-Najah National University University	……………………………

AUTHOR DECLARATION

I declare that this Master's Thesis is my original work and does not comprise any formerly published material or written by another person except where due reference has been made in the text and documented in the references list.

Signed:	Date:
Mohammad Safi	16/07/2022

ACKNOWLEDGMENTS

I would like to express my appreciation and thanks to my supervisor, Dr. Ghada Karaki, who provided me with the opportunity to work with her. She has been of immense help, advising and guiding me during my studies. I also acknowledge the support and encouragement of my supervisor, who put me on the right track, making me understand strengths and weaknesses trying to convert the weakness to strength in the field I am interested in.

I would further like to thank all the Department of Civil and Environmental Engineering teachers for their commitment to teach me during various academic courses that I took at Birzeit University.

DEDICATION

To my parents, sister-Rawan, brothers- Mohanad, Qusai, the rest of the family, friends and who have stood behind me. To my people in Palestine, I dedicate this work….

ABSTRACT

The response modification factor (R) is an essential seismic design parameter and also a performance level indicator. It is a force reduction factor that considers the inelastic deformations formed in buildings due to a seismic action that affects the design forces of the structures. The R factors provided in the US and the EU standards give inaccurate representation of the local Palestinian structural systems, which may lead to significant variability and inconsistency in the seismic forces, design of the structures, and consequently, a change in both behavior and performance buildings. This occurs due to traditional construction practices that enforce other components such as infill walls to contribute to the lateral resisting system. Therefore, there is a dire need to investigate the realistic R factor for different structural systems in Palestine. This research assesses the non-linear seismic behavior of different building systems by evaluating the R factor for each building model using non-linear analytical tools such as the OpenSees program. Line framing combinations are used in this study (2D-analysis) as each line frame type is analyzed alone to identify the controlled line framing with minimum R-factor. A set of twelve two-dimensional building models are developed and analyzed using a non-linear static pushover procedure to compute the R factor. Moreover, the coefficient method (CM) estimates the performance level at a particular seismic demand. It was found that the R factor recommended by the ASCE/SEI 7-16, Euro code (EC8), and the Egyptian code (ECP-201) were unconservative and overestimated the R factor for most of the prototype frame models that consider the current construction practices. Structural deficiencies and the height of structure were found to affect the performance of buildings, and decrease the R factor as well. Building structures with stone-concrete or masonry-concrete infills increased the strength and stiffness of the frame system. However, it reduced the frame system's ductility and its plastic strain energy. Consequently, the R factor was affected. Furthermore, the research proved that most frame models were inadequate and failed to meet the required performance level by the conventional standards, which was the life safety performance level under the local seismic demands. Recommendations are presented in this study to enhance the performance and behavior of building structures in Palestine. Lastly, this thesis paves the way for further research on evaluating R-factor and performing seismic assessment of structural buildings in Palestine, considering the limitations that have been faced in this study.

Keywords: Reinforce concrete structures, Moment-resisting frames, Response modification factor, Pushover analysis, OpenSees, Seismic performance assessment.

TABLE OF CONTENT

CHAPTER TITLE **PAGE**

CHAPTER 1. INTRODUCTION

CHAPTER 2. SEISMICITY IN PALESTINE AND SURROUNDING COUNTRIES

CHAPTER 3. RESPONSE MODIFICATION FACTOR AND ANALYSIS METHODS

CHAPTER 4. TYPE OF BUILDING SYSTEMS AND THEIR DEFICIENCIES

CHAPTER 5. SELECTION OF STRUCTURAL BUILDING MODELS

CHAPTER 6. NUMERICAL MODELING AND ANALYSIS

CHAPTER 7. EVALUATION OF R FACTORS OF PROTOTYPE FRAME SYSTEMS

CHAPTER 8. PERFORMANCE ASSESSMENT FOR STRUCTURAL BUILDINGS

LIST OF FIGURES

FIGURE **PAGE**

LIST OF TABLES

TABLE **PAGE**

CHAPTER 1: INTRODUCTION

1.1 Research Motivation:

Earthquakes are one of the most destructive natural hazards that can cause significant damage to the structures and infrastructure components, which impacts the country's economic and social aspects. Therefore, earthquake engineering is an established interdisciplinary branch of engineering that aims to estimate the seismic impacts on structures in terms of damage and loss in life and property. In the past decades, the seismic design philosophy for buildings developed many procedures to model and determine the seismic loads. Determining seismic loads is an essential step for a proper seismic design of structures. Force-based design procedures are among the most common procedures used in the conventional seismic design codes as SEI/ASCE7-16.

The Equivalent Lateral Force Method (ELF) is the main static analysis procedure, and the Response Spectrum Method (MRS) is the more common dynamic analysis procedure. Both methods are found in seismic codes and guidelines and are popular for their simplicity compared with non-linear time history analysis. These procedures are within the linear analysis, so the inelastic behavior (non-linearity) of the structures under a seismic action is accounted for by response modification factor. The response modification factor considers the inelastic deformations formed in buildings due to a seismic action that affects the design forces of the structures. The response modification factor represents the system's capability of dissipating energy through inelastic deformations, and it depends on the type of building's structural system, regularity, and construction material. Moreover, this factor is used to reduce the elastic seismic forces into their respective inelastic forces that must be resisted by the structures, which is extremely important to evaluate the lateral capacity of the structural systems.

The response modification factor (R) is found for example, in US standards, such as in Uniform Building Code UBC-1997, and SEI/ASCE7-16, and in European Code (Eurocode 8) as the behavior factor (q). The values of this factor are based on the designed lateral resisting system (skeleton structure) and vary among these codes and standards. This indicates that this factor is determined by investigating the performance of the local building stocks during past earthquakes.

However, countries that do not have seismic codes or standards also use R-factors from other standards.

In Palestine, a developing country, buildings are designed according to the R-factors from the US and the EU codes and standards, and this may lead to significant variability in the seismic forces, design of the structures, and consequently, a change in both behavior and performance of buildings due to the fact that the construction technology in Palestine is insufficient to build based on any seismic guidelines. The traditional construction practice of buildings also enforces other components such as infills to contribute to the lateral resisting system. In addition, the region is within a low to moderate seismic zone. On average, one major earthquake happens every century, and one hundred earthquakes occur almost every year due to the movement of the tectonic plate along the Dead Sea fault. Higher seismicity is expected along the fault that gets lower as one moves away from the fault. The seismicity and strong motion records of Palestine are different from the ones in the US and EU. Therefore, local earthquakes may cause different structural damage, and consequently, the behavior needs further investigation.

Based on the reasons mentioned above, there is a need to examine the overall performance of the buildings in Palestine by evaluating the response modification factor (R) for the building systems.

1.2 Research Objectives:

This research aims to assess the non-linear seismic behavior of different building systems in Palestine by evaluating the response modification factor (R) for each building frame model using non-linear analytical tools. The non-linear static pushover analysis procedure is utilized in this research to find the response modification factor R. Moreover, the Response modification factor R is taken as a performance indicator of structures to determine if the system has an acceptable performance level under the current construction technology using performance and damage assessments for specific performance objectives. Lastly, a comparison between the calculated response modification factor and the values from the seismic codes of practice is performed.

1.3 Framework of Thesis:

The following chapters are discussed in this thesis:

Chapter 2: Seismicity in Palestine and Surrounding Countries

This chapter discusses the nature of tectonic plates in Palestine, historical records of earthquakes in Palestine, the different seismic hazards and their effect, the adapted Seismic Codes in Palestine, and seismic hazard maps used in Palestine and in the Region.

Chapter 3: Response Modification Factor and Analysis Methods

An Introduction to the response modification factor (R) concept is presented. An intensive Literature review on the response modification factor R is performed, and the methods for evaluating the response modification factor R are also discussed. Recommended values of the R factor from the adopted seismic codes are presented. Moreover, the definition of the coefficient method (CM) is introduced.

Chapter 4: Type of Building Systems and their Deficiencies

Various types of building systems in Palestine are described in this chapter. Shortcomings in Construction Practices are also discussed. Furthermore, a brief literature review on the effect of structural deficiencies on the R-value. Furthermore, the modeling of infill walls is presented and discussed.

Chapter 5: Selection of Structural Building Models

A description of reinforced concrete moment resisting frame structures in Palestine is introduced. The selection of prototype frame models and their structural detailing is also presented. An overview of material properties and loads used in constriction practices is presented.

Chapter 6: Numerical Modeling and Analysis

A detailed description of the 2-D analytical models using OpenSees software is discussed in this chapter. The constitutive material model parameters and beam-column joint input parameters are also defined. The geometrical and structural data used for system modeling, modeling criteria, and model verification are discussed. Furthermore, validation of the behavior of the infills model is presented.

Chapter 7: Evaluation of R Factors of Prototype Frame Systems

The analysis results from the non-linear static pushover method for each reinforced concrete moment-resisting frame (RC-MRF) Prototype are presented in this chapter. The calculated R factors for each prototype frame model are discussed and compared with the recommended values from SEI/ASCE 7-16, Euro code (EC8), and the Egyptian code (ECP-201).

Chapter 8: Performance Assessment for Structural Buildings

This chapter presents an introduction to performance-based seismic assessment, then discusses performance point determination based on the coefficient method (CM). Performance and damage evaluation for each prototype frame model under local seismic demand is also investigated and discussed.

CHAPTER 2: SEISMICITY IN PALESTINE AND SURROUNDING COUNTRIES

2.1 The Nature of Tectonic Plates in Palestine and Countries around:

The seismic activity in the Eastern Mediterranean Region, especially Palestine, is directly affected and controlled by the geodynamic processes along the Dead Sea Transform (DST). The Dead Sea Transform is a fault between the Arabian and the Sinai tectonic plates. The relative motion of the two plates causes a left lateral displacement, which creates a collision zone from the Red Sea to the Taurus-Zagrous shown in Fig 2-1. The Dead Sea Transform started to form in the mid-Miocene, accommodating approximately 105 km northwards displacement of the Arabian Plate (Fruend, 1968).

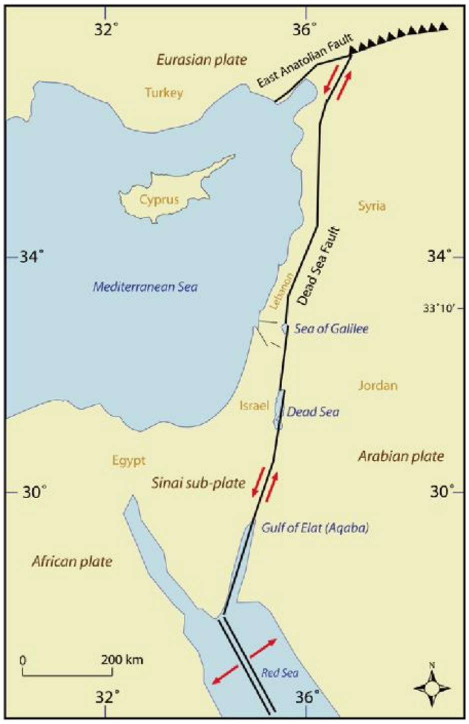


Fig 2-1: The Dead Sea Transform Fault (Garfunkel et al., 2014).

2.2 Seismic Hazard in the Region:

Palestine has a long history with earthquakes, mainly because of the Dead Sea fault along the two plates in the Eastern Mediterranean Region. The historical records show that major earthquakes have caused severe damage in buildings and losses in human lives, and these destructive earthquakes happened in the Jordan-Dead Sea region (Al-Tarazi, 1999). According to Shapira

(Shapira, 1983) and Abou Karaki (Abou Karaki, 1987), most of the recorded earthquakes have a magnitude ranging between 1.0 to 6.5 on the local magnitude scale (ML). Most historical earthquakes are concentrated along the central Dead Sea fault, as shown in Fig 2-2. According to Al-Dabbeek (Al-Dabbeek, 2010), the region is expected to have a major earthquake of magnitude more than six at any time in the near future, and this earthquake will have an epicenter in the north of the dead sea. This is based on the data of an earthquake that occurred in 1927 (6.25 magnitude with an epicenter some 15 km north of the Dead Sea). However, the author states that some other studies in the region predict that a possible earthquake could have an epicenter in the southern part of the dead sea.

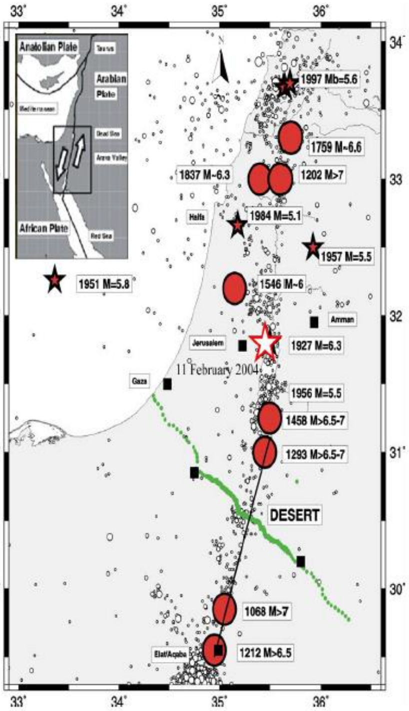


Fig 2-2: Seismicity Map of the Dead Sea Transform Region for the Period 1000-2007 (Ambraseys et al., 1994; Al Dabbeek, 2010).

2.3 Adopted Seismic Codes in Palestine:

As part of the conducted interviews discussed in detail in Chapter 4, an investigation of the most common and adopted design codes in Palestine was conducted. Consequently, most of the design offices adopt the ACI-314 code for structural design of buildings, while others use the Jordanian code. However, when it comes to the seismic design of structures in Palestine, the seismic forces induced by an earthquake are estimated using Equivalent Static Force Procedure which determines the total design base shear in a given direction. In contrast, the Response Spectrum Method estimating the seismic forces using design response spectrum is not well understood and is randomly used. These seismic forces are calculated according to either UBC 97 (the Uniform Building code, last edition issued in 1997) or SEI/ASCE7-16 (American Society of Civil Engineers). The following sections explain the procedures in UBC-97 and SEI/ASCE7-16.

2.3.1 UBC 1997:

The data required to estimate the seismic loads in the Unified Building Code are shown in Table 2-1.

The procedure for determining the base shear based on UBC 97 is as follows:

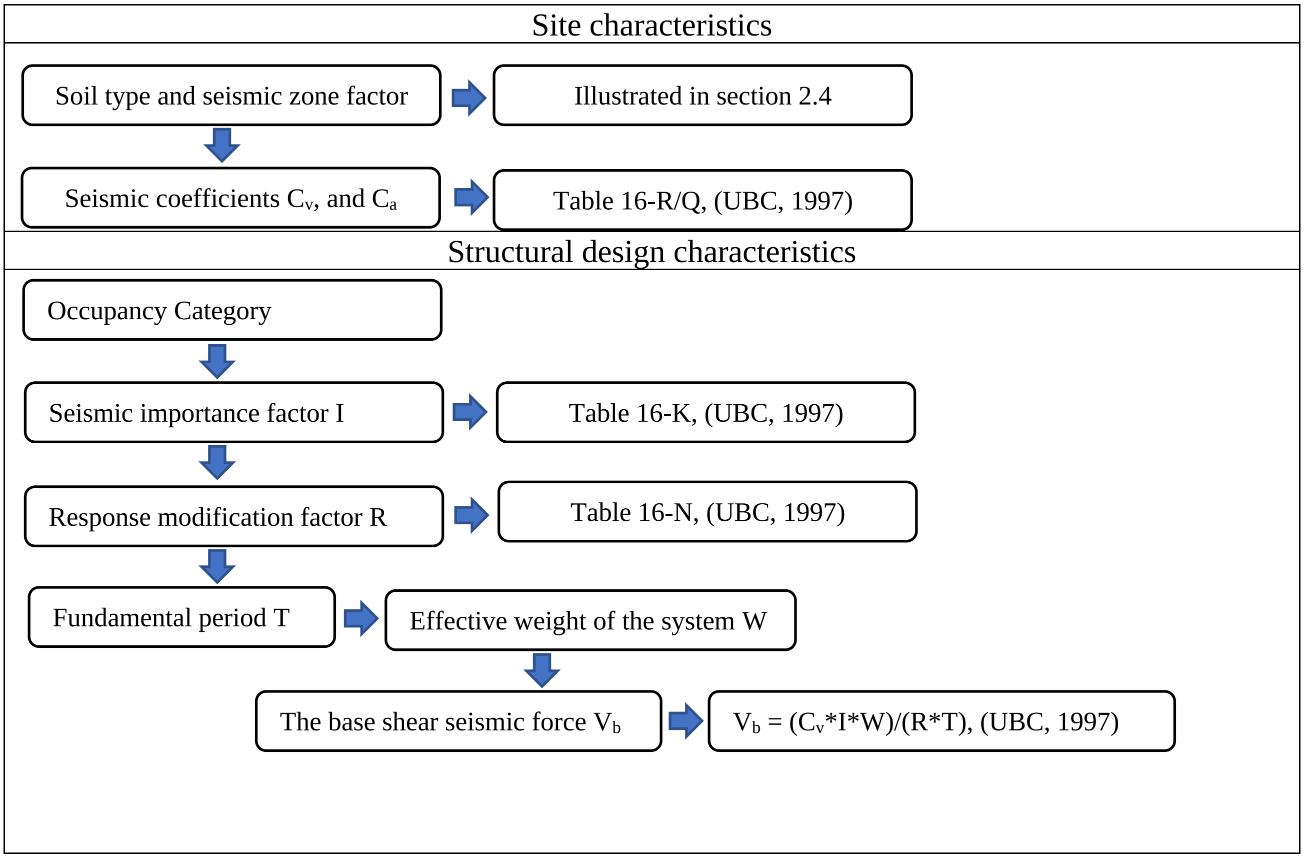


Table 2-1: Data required to estimate the seismic loads in UBC 1997.

Parameter	Abbreviation
Seismic importance factor	I
Soil profile type	S
Seismic zone factor	Z
Seismic coefficient	C_v
Seismic coefficient	C_a
Response modification factor	R
Fundamental period	T
Effective weight of the system	W
The base shear seismic force	V_b

However, the design response spectrum can be constructed according to UBC (UBC, 1997) using the parameters shown in Fig 2-3, which describes the relationship between acceleration spectrum versus time (periods).

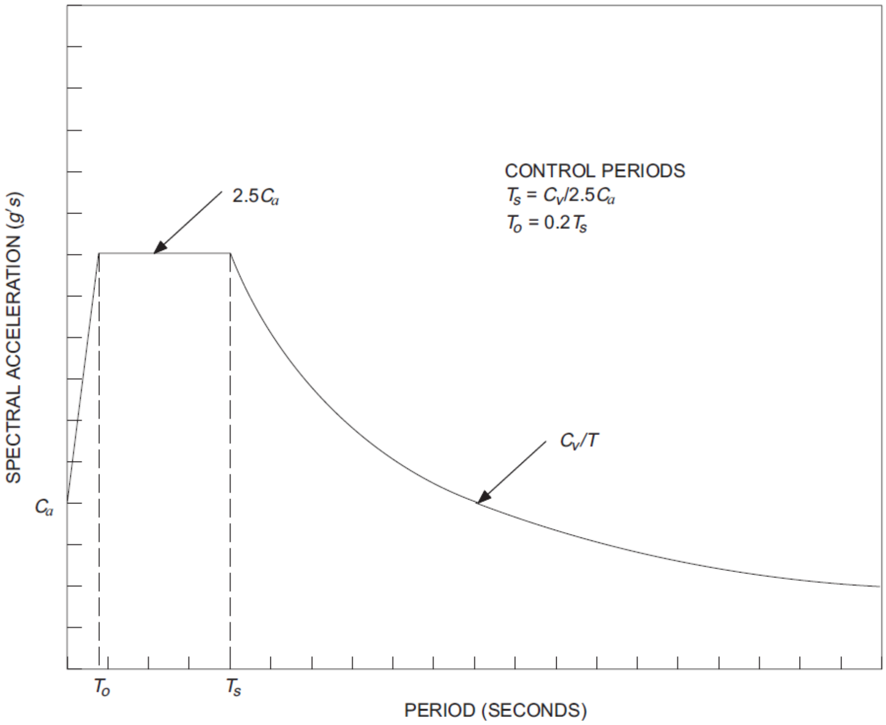


Fig 2-3: Design Response Spectra (UBC, 1997).

2.3.2 SEI/ASCE7-16:

The data required to estimate the seismic loads in the American Society of Civil Engineers Standard are shown in Table 2-2.

The procedure for determining the base shear based on SEI/ASCE7-16 is as follows:

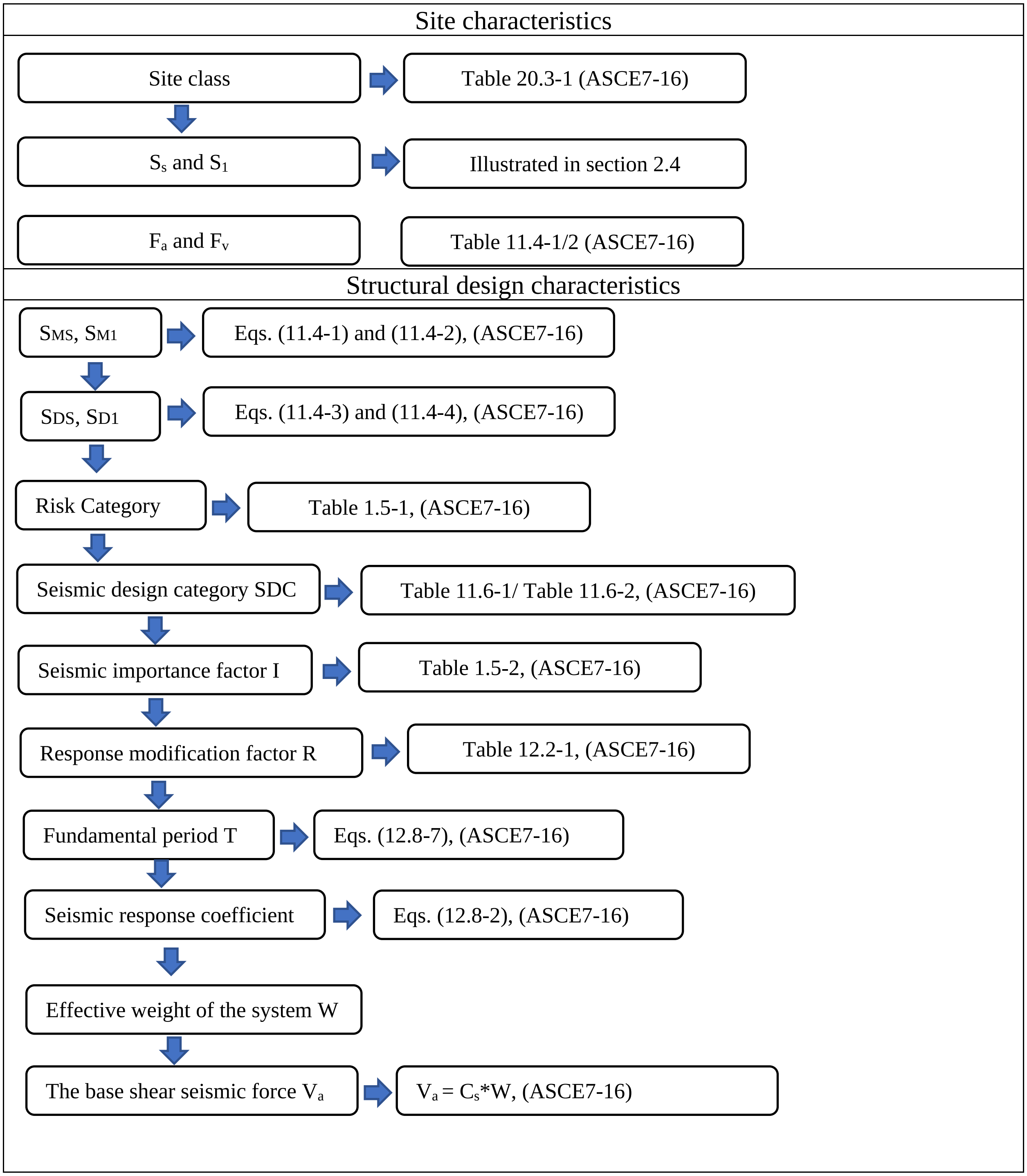


Table 2-2: Data required to estimate the seismic loads in SEI/ASCE7-16.

Parameter	Abbreviation
Seismic importance factor	I
Spectral accelerations for short periods and 1 second periods	S_S, S_1
Site coefficient	F_a
Site coefficient	F_v
Maximum spectral accelerations at short periods and 1 second period	S_{MS}, S_{M1}
Design spectral accelerations at short periods and 1 second period	S_{DS}, S_{D1}
Response modification factor	R
Seismic response coefficient	C_s
Effective weight of the system	W
The base shear seismic force	V_a

Moreover, the design response spectrum can be constructed according to SEI/ASCE7-16 using the parameters shown in Fig 2-4.

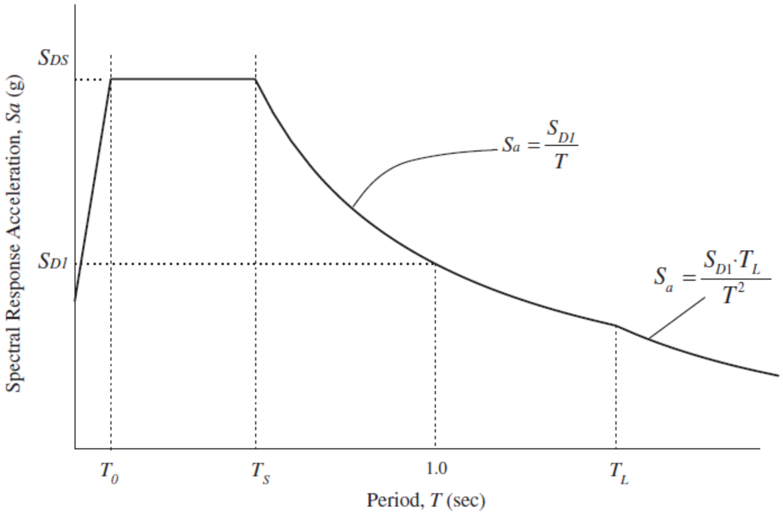


Fig 2-4: Design Response Spectra (SEI/ASCE7-16).

These procedures are adopted in the design offices in Palestine. However, the response modification factor R in both standards is similar. Therefore, the procedure according to SEI/ASCE7-16 is used to evaluate and assess the performance of the chosen models, which represent the existing buildings in Palestine.

2.4 Seismic Hazard Maps used in Palestine and in the Region:

The UBC 97 code procedure to estimate the seismic base shear uses the seismic zone factor for Palestine to represent the seismic hazard for the area, as described in section 2.3.1. This factor divides the area into four different seismic zones (1, 2A, 2B, 3); each seismic zone has a peak horizontal ground acceleration value, as shown in Fig 2-5. In contrast, the seismic hazard map in Fig 2-6 is implemented to estimate the base shear according to SEI/ASCE7-16 procedure.

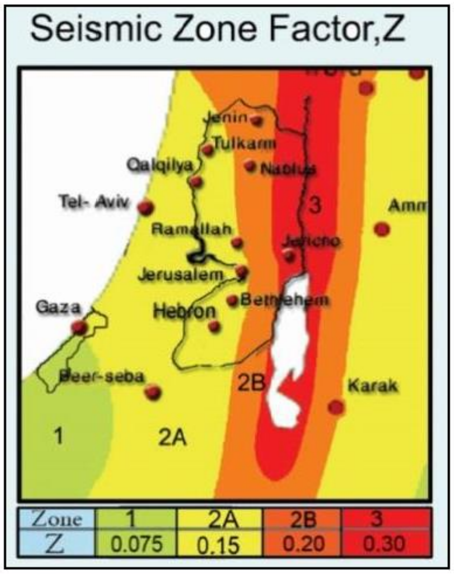


Fig 2-5: Seismic zone factor for Palestine (Source An-Najah National University, Boore et al., 1997).

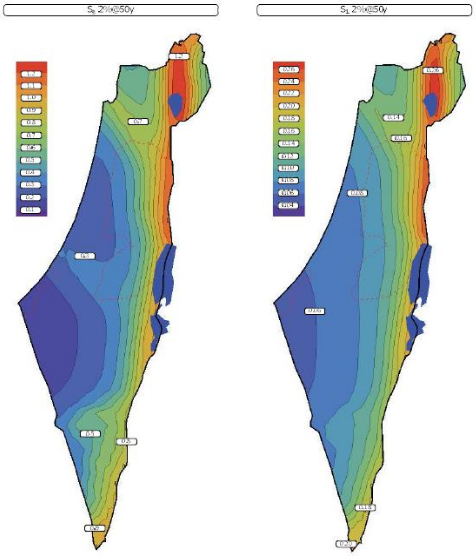


Fig 2-6: Seismic Hazard Map for Historical Palestine (IL Standard SI 413).

CHAPTER 3: RESPONSE MODIFICATION FACTOR AND ANALYSIS METHODS

3.1 Introduction to the Concept of Response Modification Factor:

The definition of the response modification factor (R) for the seismic design of structures was carried out first through (Wu et al., 1989; Hanson et al., 1993). The authors focused on designing energy dissipation systems by looking at the displacement response. The results were implemented in Uniform Building Code (UBC, 1994). Then, the proposals from (Newmark and Hall, 1982) were implemented in the (UBC, 1997) and (FEMA-273, 1997). According to the IBC (IBC, 2000), the value of R should be used to reduce the seismic forces of structures, and it was established based on the performance of different structural systems in previous strong earthquakes. In contrast, the National Earthquake Hazards Reduction Program (NEHRP, 2000) proposed the R coefficient to explain the ductility, over strength, and energy dissipation in the soil-foundation system.

3.2 Literature Review on the Response Modification Factor:

Many studies have been investigating response modification factors (R) for the seismic design of structures. (Miranda, 1994) investigated the response modification factor R coefficients, described as a strength reduction factor (R_μ). Miranda (Miranda, 1994) found that the R factor is primarily a function of displacement ductility (μ) and the structure's natural period (T). (Daza, 2010) defined the relationship between the R factor and the minimum strength of the building using the pushover analysis method. Furthermore, (Mitchell D and Paultre P, 1994) and (Izadinia M, 2012) summarized that non-linear static (conventional pushover or adaptive pushover analysis methods), pseudo-static, or dynamic analysis methods can be used to evaluate the value of the response modification factor.

Moreover, the literature included multiple research work on the numerical derivation of response modification factor R for reinforced concrete structures. (Adeel, 2010) evaluated the response modification factor R for the reinforced concrete moment resisting frames in Pakistan. It was found that the US seismic codes' recommended values of the response modification factors are unconservative and overestimate R values for a selection of ground motion records. (Apurba et al., 2013) investigated the actual values of response modification factors for realistic reinforced concrete buildings designed and detailed according to Indian standards. These calculated factors were compared with the values suggested in the national design code. The authors concluded that the recommended values of R in the Indian standards are higher than the actual values of R, which is considered unsafe.

In contrast, (Manar et al., 2021) studied the response modification factor for reinforced concrete frames with non-uniform spans and heights. The authors then compared the calculated values with the recommended values in the Egyptian code (ECP-201) of practice. It was found that the recommended values of R in the code are conservative and consequently not economical. Furthermore, the non-uniformity in span lengths significantly affects the R factor.

(Sharifi et al., 2018) derived the response modification factor R for both ordinary and special-moment resisting RC-frame structures designed based on the limit state design. The study results indicated that the response modification factor R is significantly affected by the number of stories, bays number, and the maximum imposed lateral displacement during a pushover analysis. Moreover, (Sharifi et al., 2018) compared the values of the obtained R-factor with the corresponding values prescribed in current seismic design codes. The authors concluded that the design R-factors of both ordinary and special-moment resisting RC-frame structures should be taken as 3 and 7, respectively, to achieve a conservative and safe design. (Brahmavrathan et al., 2016) evaluated the response modification factor R for irregular ordinary and special-moment resisting RC-frame structure. The study revealed that the actual value of R was found to be decreasing while the number of stories of frame structures increased. Furthermore, the actual value of R was less than the value assumed in the design process. Following the authors, a certain percentage of reduction for the response modification factor R has to be considered in irregular frame structures.

(Louzai et al., 2015) assessed the value of seismic behavior factors for reinforced concrete frame structures using a comparative analysis between non-linear pushover and incremental dynamic analysis. The structural models were designed according to the reinforced concrete code BAEL 91 and Algerian seismic code RPA 99/Version 2003. The study showed that the adopted value of the seismic behavior factor from the seismic design code RPA 99/Version 2003 was overestimated. Moreover, the value of the seismic behavior factor tended to decrease for larger number of stories and when non-linear static pushover analysis is performed. However, non-linear incremental dynamic analysis increased the value of the seismic behavior factor for a higher number of stories. This may indicate that the value of the seismic behavior factor does not depend on the structure's height. (Patel et al., 2017) determined the response modification factor R for both ordinary and special-moment resisting RC-frame structures through a non-linear pushover analysis. According to Patel et al. (Patel et al., 2017), the response modification factor R for both ordinary and special-moment resisting RC-frame structures is affected by the seismic zone. Furthermore, the authors concluded that the recommended value of response modification factor in the Indian seismic code (IS-1893, 2002) is on the conservative side.

(Abdi et al., 2017) studied the effect of implementing viscous damper devices in RC structures on the value of response modification factor R. Non-linear pushover analysis was performed to evaluate R-factor. The study results show that the value of R-factor is higher in the case of reinforced concrete structures with damper devices. Furthermore, it was found that the response modification factor is highly affected by the number of dampers and the height of the buildings. (Wang, 2014) investigated the influence of high mode effects on the ductility reduction factor in the case of MDOF shear-type structures. It was found that the modification factor is mainly affected by the fundamental period and ductility.

(Jinkoo et al., 2005) evaluated the response modification factor for both special and ordinary-concentric braced frames by performing a non-linear pushover analysis on the model with different stories and span lengths. It was found that response modification factors are smaller than the recommended ones by design codes except for the low-rise special concentric braced frames. (Mwafy, 2002) assessed and calibrated the force reduction factors adopted in current seismic codes. The author concluded that the force reduction factors adopted by the Eurocode (EC8) design code are over-conservative and can be increased in the case of regular frame structures that are subjected to low ground acceleration. (Junaid et al., 2020) studied the behavior of low-rise reinforced concrete frames retrofitted with steel haunches. The results showed that the response modification factor could be used in a force-based design and assessing haunch-retrofitted reinforced-concrete frame structures. (Borzi et al., 2000) illustrated that there is no influence of the earthquakes data set (magnitude, distance, and site characterization) on values of force reduction factors and found that the force reduction factor is only influenced by the shape of the hysteretic model, and it showed low sensitivity to strong-motion characteristics.

(Chaulagain et al., 2014) investigated the actual response modification factor R for irregular RC buildings in Kathmandu valley, Nepal, through a non-linear static pushover analysis. The authors concluded that geometrical configuration and material strength influence the actual R factor. Furthermore, it was found that the computed R-factor is less than the recommended value in the (IS 1893. 2002). (Ma'moun, 2020) evaluated the response modification factor R for reinforced concrete structures with shear walls. The results of the study showed that the openings in the 2D RC frames with shear walls affect the value of the response modification factor. Likewise, (Badrashi, 2016) computed the response modification factor for special-moment resisting frame buildings in Pakistan using both experimental and numerical work. It was found that the response modification factor for RC special moment resisting frames is 8.5. In comparison, the response modification factor value ranged from 4.0 to 6.1 in the case of special-moment resisting frames with construction deficiencies. Moreover, the span length and building height affect the response modification factor.

(Pandit et al., 2020) studied the effect of using masonry infilled walls on the response modification factor of existing RC-buildings. The authors evaluated the structural response in terms of natural period, base shear, strength, ductility, stiffness, and response modification factor R. and it was found that the inclusion of infills decreased the time period of the structure by 40-60%. The presence of infills increased the base shear of the structure from 1.2 to 2.2 and reduced the displacement ductility factor by 35-45%. In most cases, the response modification factor R values of existing masonry infilled RC buildings were lower than the bare RC buildings. The results confirmed that buildings with infill walls significantly influence the R-factor of structures. In contrast, (Patel and Vasanwala, 2019) assessed the response modification factor of un-reinforced masonry infilled RC buildings. The study results showed that including infill walls decreased the ductility factor, while the overstrength factor was found to be increased. Furthermore, the authors concluded that the overall response modification factor R increased due to the substantial increase in the values of overstrength factor.

(Shendkar et al., 2018) investigated the effect of semi-interlocked masonry (SIM) and unreinforced masonry (URM) infill walls on the response modification factor of RC frame structures. The authors revealed that the base shear of the SIM infilled frame is higher than the base shear in the case of URM infilled walls. The ductility factor is less in both SIM and URM infilled RC frame structures compared with bare RC frames. However, the over-strength factor increases in the case of SIM and URM infilled panels. In addition. The R-factor is sensitive to material and geometric configuration. (Ko et al., 2008) examined the seismic behavior of a low-rise RC moment-resisting frame with masonry infill walls. The study results indicated that the inclusion of the masonry infill walls increases the structure's global stiffness and decreases the natural periods. The structure may also be subjected to significant damages if there is no consideration of the infill walls in the design.

As noticed from the above research works, there is an active debate regarding the derivation of R values and their influence on seismic design and behavior. Most research works agree that the R-factor is affected by the structure's height, irregularity, damping of frame system, and the existing openings in the structure. Moreover, the inclusion of infill walls decreases the structure's period and ductility and increases the frame systems' base shear. Therefore, in this research, the response modification factor (R) for each building frame model is evaluated to assess the non-linear seismic behavior of different building systems in Palestine, and it is expected to have similar outcomes to the literature mentioned above review with a value of response modification factor around 3.

3.3 Evaluation of the Response Modification Factor:

The response modification factor R is necessary to design earthquake load-resisting elements. The R factor was discussed in (ATC-3-06, 1978) for the first time. It was evaluated based on the performance observed of buildings during past earthquakes. Moreover, according to (ATC-19, 1995), the R accounts for over strength, ductility, and redundancy. Response modification factor R can be calculated by the product of three parameters that influence the response of structures during earthquakes (ATC-19):

$$R = R_0 R_\mu R_r$$

Where:

$R_0 = Over\ strength\ factor$

$R_\mu = Ductility\ factor$

$R_r = Redundancy\ factor$

Evaluation of these factors is based on running a non-linear static pushover analysis and developing each building category's capacity curve, as shown in Fig 3-1. The capacity curve is the relationship between base shear and roof displacement and can be obtained from a non-linear static analysis (pushover analysis). This type of analysis depends on incremental lateral load applied to the building until the building collapses or reaches the top target displacement, representing the structure's top displacement under an earthquake. Based on the results of pushover analysis, response modification factor R can be calculated and evaluated.

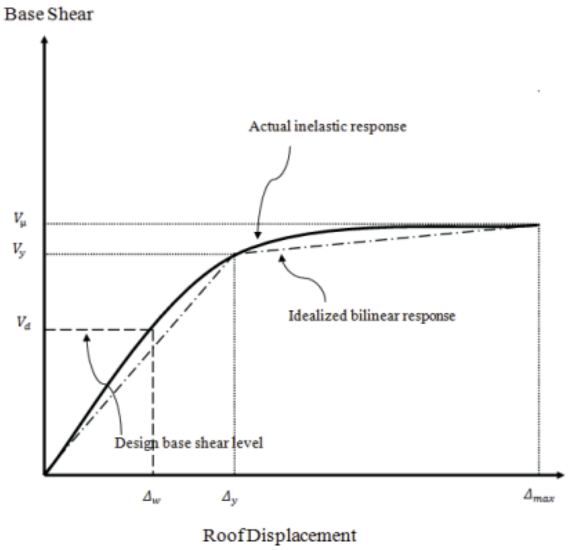


Fig 3-1: Ideal and the Actual Inelastic Response (Capacity Curve).

3.3.1 Ductility Factor (R_μ):

According to Abdi et al. (Abdi et al., 2018), the Ductility ratio (μ) can be calculated at the element, story, and system levels. The terms curvature, strain, and rotational ductility ratio are expressed at the element level. However, the term displacement ductility ratio is used at the story level. The ductility factor (R_μ) is calculated based on the displacement ductility ratio (μ). (Newmark and Hall, 1982) studied the response modification factor R resulting from ductility and found that $R\mu$ is sensitive to the natural period of structures. The authors developed a chart with five periods range to estimate the ductility factor (R_μ), as shown in Fig 3-2. Moreover, Equations (1)-(5) estimate the ductility ratio for different structures' natural periods.

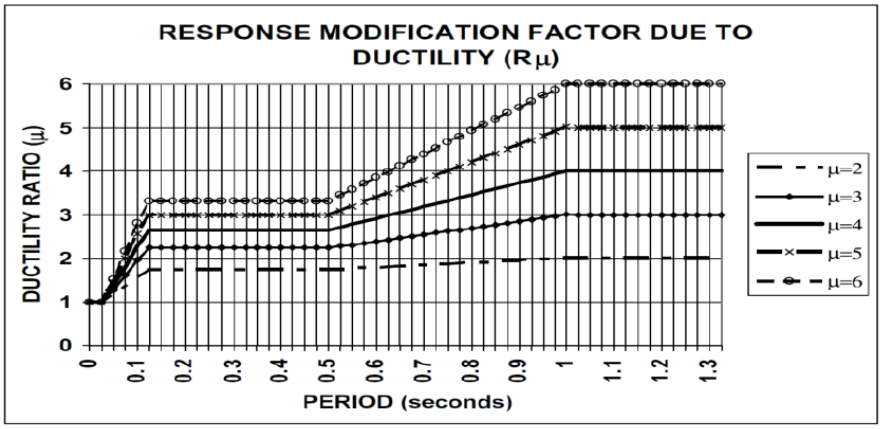


Fig 3-2: $R_{\mu} - T - \mu$ charts (Newmark & Hall, 1982).

Periods ≤ 0.03 sec:

$$R_u = 1.0 \tag{1}$$

Periods 0.03 <t < 0.12 sec:

$$R_u = 1 + \frac{(T - 0.03) * ((2 * U - 1)^{0.5} - 1)}{0.09} \tag{2}$$

Periods 0.12 ≤ T ≤ 0.5 sec:

$$R_u = (2 * u - 1)^{0.5} \tag{3}$$

Periods 0.5<T <1.0 sec:

$$R_u = (2 * u - 1)^{0.5} + 2 * (T - 0.5) * (u - (2 * u - 1)^{0.5} \tag{4}$$

Periods T ≥ 1.0 sec:

$$R_u = u \tag{5}$$

3.3.2 Over Strength Factor (R_0):

The actual strength of the structures generally exceeds the design strength due to the overall design simplifications, actual construction practices, and the differences between the real and nominal material strength. Therefore, an overstrength factor should be included to avoid any local and global behavior change. Based on the capacity curves developed from the non-linear static analysis, the maximum base shear (V_0) and the designed base shear (V_d) can be found. Then, the overstrength factor (R_0) can be calculated using Equation (6).

$$R_s = \frac{V_0}{V_d} \tag{6}$$

3.3.3 Redundancy Factor ($\mathrm{R_r}$):

(ATC-19, 1995) defines redundancy as "beyond what is essential or naturally excessive" in the perspective of structural engineering. Redundancy is to take into consideration the reliability of the seismic frame systems that use multiple vertical framing lines in each main direction of the structure. A redundant seismic framing system is built of multiple vertical lines of framing that are designed and detailed to transfer seismic-induced inertial forces to the foundation. This kind of redundancy in the system is active redundancy. However, standby redundancy is when members stay inactive in the typical cases and become active when one of the active components fails. The $redundancy\ factor\ (R_r)$ can be taken from Table 3-1 (ATC-19, 1995).

Table 3-1: $Redundancy\ factor\ (R_r)$, (ATC-19 (1995).

Line of Vertical Seismic Framing	Redundancy Factor $\mathrm{R_r}$
2	**0.71**
3	**0.86**
4	**1.00**

3.4 The R Factor in the Adoptive Seismic Codes and Neighboring Country Codes

The concept of response modification factor has been introduced in many seismic codes and standards of countries with seismic hazards, indicating its significance in seismic analysis and design. The following section presents an overview of the response modification factor and its recommended values in the seismic codes of the U.S (SEI/ASCE7-16, and UBC 1997), Europe (Euro code 8), and Egypt (Egyptian code ECP-201).

3.4.1 U. S (SEI/ASCE7-16, UBC 1997)

In the U.S codes and standards, the response modification factor is defined for reinforced concrete moment resisting frames based on the level of ductility. For instance, an ordinary moment-resisting frame relates to a low ductility. On the other hand, a special moment-resisting frame indicates a high ductility. The recommended values of the response modification factor R in the Uniform Building Code (UBC, 1997) and SEI/ASCE7-16 are shown in Table 3-2.

Table 3-2: Recommended Values of R-Factor in both SEI/ASCE7-16 and UBC-1997.

Building System	R-Factor	
	SEI/ASCE7-16	UBC-1997
Special reinforced concrete moment frames	**8**	**8.5**
Intermediate reinforced concrete moment frames	**5**	**5.5**
Ordinary reinforced concrete moment frames	**3**	**3.5**
Special reinforced concrete shear walls	**6**	**5.5**
Ordinary reinforced concrete shear walls	**5**	**4.5**

3.4.2 Europe (Euro Code 8)

The concept of response modification factor in the Euro code (EC8) is based on reducing the elastic spectral demand into their corresponding strength design level. This factor is called the behavior factor “q”, which depends on the level of ductility, stiffness regularity, and building strength. The recommended values of the behavior factor q for different structural systems are shown in Table 3-3.

Table 3-3: Recommended Values of Behavior Factor (q) in Euro Code (EC8).

Building System	Behavior Factor (q)
Frame System	**5**
Wall system with coupled walls	**5**
Wall system with uncoupled walls	**4**

3.4.3 Egypt (Egyptian Code ECP-201)

The response modification factor for reinforced concrete structures in the Egyptian Code (ECP-201) is also based on the ductility level. However, high ductility is indicated as sufficient ductility, and low ductility is defined as non-sufficient ductility. The recommended values of the response modification factor R for the RC moment-resisting frame system are shown in Table 3-4.

Table 3-4: Recommended Values of Response Modification Factor R in Egyptian Code (ECP-201).

Building System	R-Factor
Reinforced concrete moment frames with sufficient ductility	**7**
Reinforced concrete moment frames with non-sufficient ductility	**5**

It can be seen from the above mentioned that there is variability in the values of response modification factor R for each structural system, which indicates that this factor is determined by investigating the performance of the local behavior of structures during past earthquakes. Therefore, to avoid inconsistency and significant variability in the seismic performance of structures in Palestine, the R-factor should be evaluated based on local conditions and structural parameters.

3.5 Analysis Methods:

The non-linear static pushover analysis is opted to derive and conduct each frame system's capacity curve. This method can provide much characteristic information that cannot be acquired from the elastic static or dynamic analysis, namely; realistic force and deformation demands for each inelastic deformed element, the effect of the deterioration of element's strength on the overall structural stability, identification of the member's yielding and hinge formation which helps in damage assessment of the structural system, and monitor the progress of the capacity curve of the structure. Consequently, this method is adopted because the previously mentioned information is crucial to accomplishing the goals of this research (M. Seifi. et al., 2008).

The capacity curve derived from the non-linear static pushover analysis is a relationship between the base shear and roof displacement. This type of analysis depends on incremental lateral load applied to the building until the building collapses or reaches the top target displacement, representing the structure's maximum displacement under an earthquake, as shown in Fig 3-1. The selection of the lateral load pattern is more critical in the case of performance assessment due to the fact that the load pattern should deform the structure the same way an earthquake does. According to Krawinkler et al. (Krawinkler et al., 1998), the inverted triangular is the conventional pattern shown in Fig 3-3. The adaptive load shape pattern is more essential when a single mode shape does not control the structure's response.

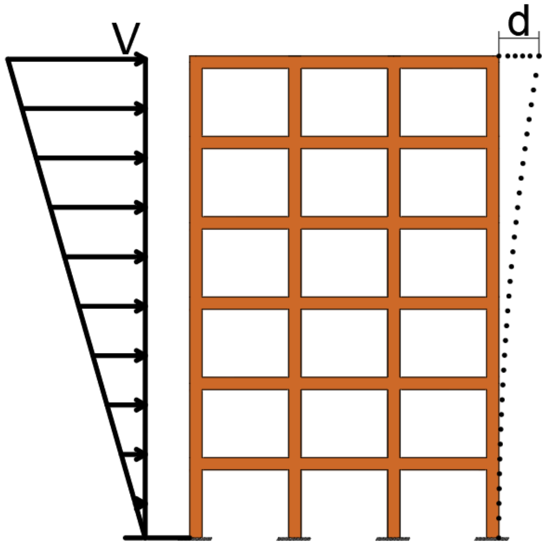


Fig 3-3: Conventional Lateral Load Distribution.

3.6 Determination of the Performance Point: Coefficient Method.

The behavior of the structural systems is further examined by carrying out a performance-based evaluation of buildings. The performance-based method is a relationship between a seismic event and a structural ability to resist that event. The assessment is done by determining the performance of a frame system and comparing it with the required performance objectives. Furthermore, the structure's response should meet the selected acceptance criteria adopted by SEI/ASCE 41-17. According to the SEI/ASCE 41-17 standard, the seismic performance-based method (Coefficient method) principles can be applied to determine the performance point for each frame system.

The displacement coefficient method (CM) is a widely used procedure to estimate the target displacement in a non-linear static pushover procedure (NSP). This method is defined in the FEMA-356 and adopted in the SEI/ASCE 41-17 standard. The coefficient method (CM) utilizes a displacement modification procedure to calculate the target displacement for a linearly elastic single degree of freedom system (SDOFs) using several coefficients. Moreover, the original structural and equivalent SDOF systems have the same period and damping. The target displacement in SEI/ASCE 41-17 (CM) is computed from Equation (7).

$$\delta_t = C_0 C_1 C_2 S_a \frac{T_e^2}{4\pi^2} \mathrm{g} \tag{7}$$

Where S_a relates to the response spectrum acceleration at the fundamental period and 5% damping ratio. It is computed using Equations (8) and (9), as illustrated in Fig 3-4. The coefficient C_0 refers to the spectral displacement of an equivalent SDOF system to the elastic displacement of a multi-degree-of-freedom system at the roof control node.

$$S_{x1} = F_v S_1 \tag{8}$$

$$B_1 = 4/[5.6 - In(100 * damping\ ratio)] \tag{9}$$

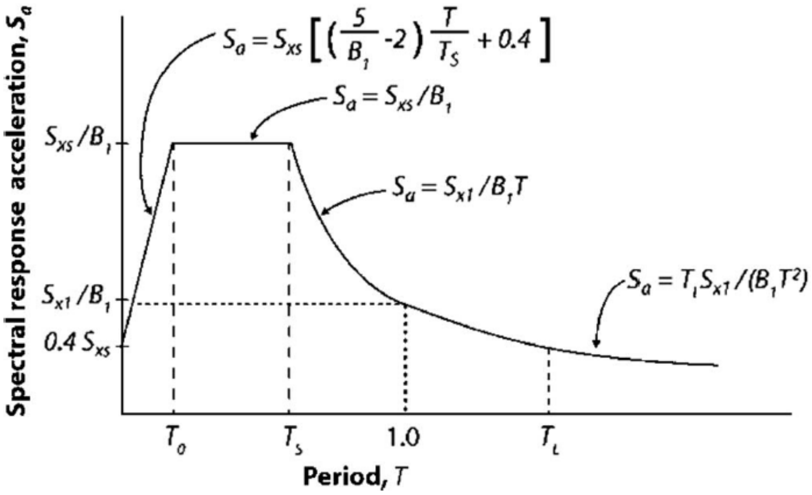


Fig 3-4: General Horizontal Response Spectrum (SEI/ASCE 41-17)

The coefficient C_1 represents the expected maximum inelastic displacements to the linear elastic displacements, given by Equation (10). In contrast, coefficient C_2 takes into account the effect of stiffness degradation and strength deterioration on the maximum displacement. For periods less than 0.7s, C_2 is calculated based on Equation (11). The effective fundamental period (T_e) is evaluated based on the idealized force-displacement curve shown in Fig 3-5. Equation (12) defines the effective fundamental period.

$$C_1 = 1.0; \qquad T_e > 1.0s, \tag{10}$$

$$1.0 + \frac{\mu_{strength} - 1}{aT_e^2} \qquad T_e < 1.0s,$$

$$C_2 = 1.0 + \frac{1}{800}\left(\frac{\mu_{strength}-1}{T_e}\right)^2 \tag{11}$$

$$T_e = T_i\left(\frac{Ki}{Ke}\right)^{0.5} \tag{12}$$

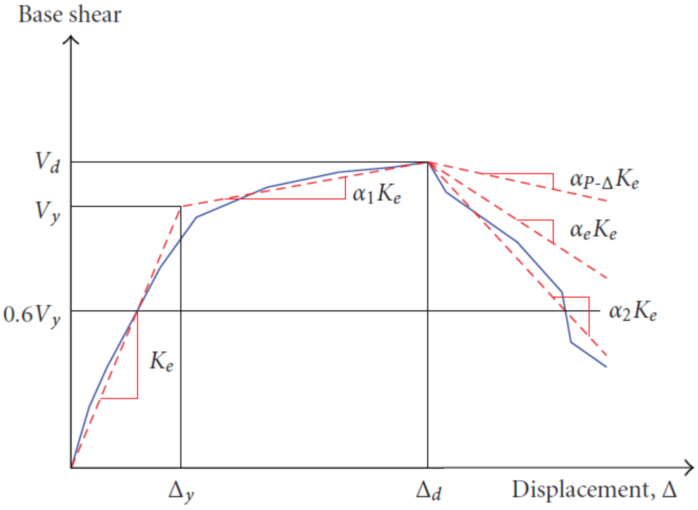


Fig 3-5: Idealized force-deformation curve in SEI/ASCE 41-17 (Goel, 2011)

Furthermore, the target displacement should satisfy the selected performance level. And referring to the FEMA 356, there are three performance levels described in Table 3-5. The chosen performance level is based on the conducted interviews (see section 4.1), and it shows that most of the Palestinian buildings are commercial-residential buildings. This means that buildings are occupied mainly by people. Therefore, the target performance level is set to be Life Safety (LS) to protect and save human lives. The life safety performance level is defined in the SEI/ASCE 41-17 as 0.75 times the deformation at point E shown in Fig 3-6.

Table 3-5: Structural Performance levels (FEMA, 356).

Performance Levels	Description
Immediate Occupancy IO	The structure retains most pre-earthquake strength and stiffness, and the system remains safe to occupy.
Life Safety LS	Significant damage happens to the structural components. However, the elements can resist partial or total collapse, and injuries may occur, but the possibility of life-threatening injuries is low.
Collapse Prevention CP	Substantial damage to the structural components and significant degradation in the stiffness and strength of structural elements occurs. Due to the structural element's damage, life-threatening injuries may occur.

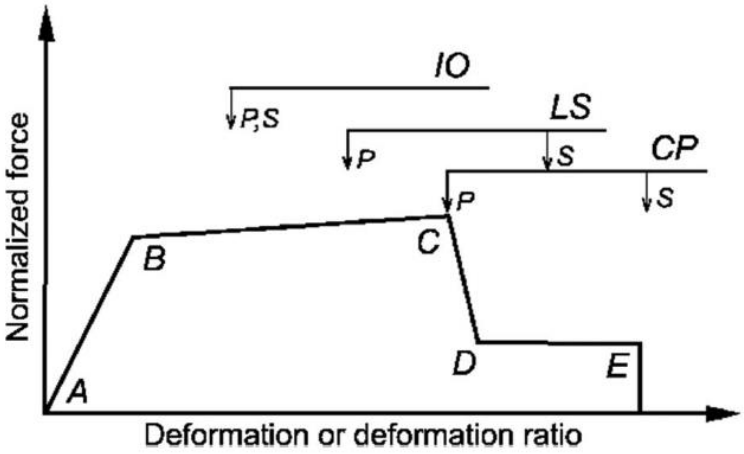


Fig 3-6: Acceptance Criterion Illustration (SEI/ASCE 41-17)

Therefore, on the condition that the performance point is below the life safety performance level, the model performance satisfies the acceptance criteria. The shorter the distance between the performance point and the life safety condition indicates a more efficient design. This procedure is applied to each analyzed frame system.

CHAPTER 4: TYPE OF BUILDING SYSTEMS AND THEIR DEFICIENCIES

4.1 Types of Building Systems:

The data regarding the building categories and their representation percentages are discussed in this chapter. The data were collected through semi-structured interviews. The interviews were personal interviews held with highly skilled/active engineers, contractors, and construction managers. A copy of the interview form and the questions that have been discussed is presented in Appendix 1. Based on these interviews, it was found that reinforced concrete frame buildings are the most common building category in Palestine. Reinforced concrete shear-walls are often used around the stairwells and elevator cases, their poor positioning does not support the frame system effectively for the building to be a shear-wall system or dual system. A study done by (Antonella et al., 2016) agreed with this classification, and it was used to guide and verify the data collection. According to Antonella et al. (Antonella et al., 2016), there are three main types of structures in Nablus city which can be as the following: masonry-concrete wall buildings, RC frame buildings, and RC shear wall buildings. The collected data was compatible with the data from (Antonella et al., 2016). Therefore, it was used for further work in this research.

The interviews were used to determine the current construction practice to build an accurate prototype for each building category. The results of the interviews showed that the majority of buildings have a number of floors ranging from 4 to 6 and from 6 to 9 floors. The inter-storey height in most buildings is 3.0 meters, and the ground floor is 6 meters in the case of commercial-residential buildings (the ground floor is commercial, and the rest are residential). The number of bays ranges from 3 to 6 with a typical span width of 4 meters. Building structures are commonly in regular shape in both horizontal and vertical planes. However, geometric irregularities exist and are presented in building structures due to the region's lack of urban land-use regulations.

Furthermore, most RC buildings have external stone-concrete infill walls with thicknesses ranging from 25 to 35cm, and this type of infill wall consists of stone, plain concrete, insulation boards, and concrete masonry, as shown in Fig 4-1. The Interviewees confirmed that some RC buildings can have concrete masonry infill walls with 20cm concrete masonry and plastering. This type of infill wall can reach 25cm in thickness. These infills are constructed based on the method illustrated by (Halahla, 2019). The author stated that the construction of infill walls precedes the construction of the flooring system and is the oldest method and most popular in many residential and commercial buildings. This method can affect the interaction between the infills and other structural components. Generally, these infill walls provide a cheap thermal and noise insulation solution. On the other hand, the infills used may add to the lateral stiffness of the frame line, and this may cause an increase in the global stiffness and strength of the frame system. In contrast, the loss in ductility is enormous, and according to Adeel. (Adeel, 2010), this behavior may result in the failure of the infills and creates severe problems in the whole structure.

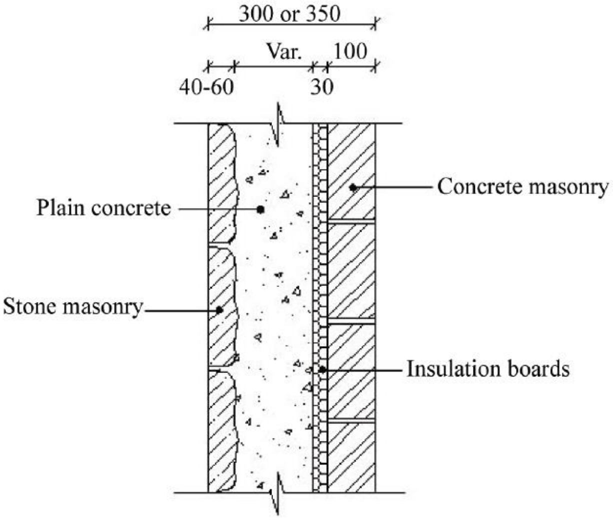


Fig 4-1: Typical Cross-Section of the Stone-Concrete Infill Panel, Dimension in mm (Al-Nimry et al., 2015).

4.2 Deficiencies in Construction Practices:

Based on the interviews, most engineers agreed that most buildings have a structural deficiency in the beam-column joint area, as no transverse reinforcement exists. Furthermore, inadequate development length in beams, columns, and beam-column joints was recognized as a deficiency in the current construction practice. The interviewed engineers also commented on the shear reinforcement and the spacing of the stirrups being in some cases inadequate for new and existing buildings.

4.3 Structural Deficiencies effect on R-value:

The ductility of the frame system mainly depends on the ductility of the beams, columns, and beam-column joints. Any deficiencies in these elements will affect the performance of the entire system. One of the significant deficiencies is the lack of transverse reinforcement in the beam-column joints. The beam-column joint behavior is affected by the inadequate shear capacity and reinforcing bars-concrete bond-slip. The absence of shear reinforcement leads to significant shear deformation in the joint area, restricting the flexural capacities for the beams and columns joining together (Pampanin et al., 2002).

Therefore, many researchers investigated the behavior of the beam-column joint to determine its effect on the lateral response of structures. (Park, 2002) summarized the results of simulated seismic load tests on RC beam-column joints. It was found that the seismic performance of an interior beam-column joint in RC moment resisting frames is poor, especially without transverse reinforcement in the joint cores. This refers to the diagonal tension cracking and bond-slip. Furthermore, the behavior of exterior beam-column joints is influenced mainly by the direction of the beam bar hooks in which way they are bent and anchored in the confined core. The test results showed that the nominal horizontal joint shear stress is less in case bar hooks are bent out of the joint core, which indicates that the seismic performance is more vulnerable. (Pampanin et al., 2002) also studied the seismic behavior of RC beam-column joints designed for gravity loads only. The experimental tests on the beam-column joints with typical structural deficiencies in the Italian construction practice between the '50s and '70s showed a significant vulnerability of the joint area under simulated seismic loads.

Furthermore, the behavior of beam-column joints has been investigated analytically. Many researchers have proposed joint models for use in simulating lateral response in the case of earthquake loadings. (Youssef and Ghobarah, 2001) proposed a joint model which consists of two diagonal translational springs linked to the opposite corners of the joint area, with 12 translational springs located at the interface of the joint area to cover all the modes of inelastic behavior such as bond-slip and concrete crushing. This model demands a large number of translational springs, and each one needs a separate constitutive model. (Lowes and Altoontash, 2003) proposed a joint model with four nodes and 12 degrees of freedom (DOF), representing three types of inelastic mechanisms of beam-column joints under cyclic loading. According to the authors, the bonding-slip response of beam-column longitudinal reinforcement is modeled as eight zero-length translational springs. Moreover, a zero-length rotational spring is used to represent the joint's shear deformations, and four zero-length shear springs are used to simulate the interface-shear deformations.

(Shin and LaFave, 2004) proposed the joint model as rigid elements located along the edges of the joint area with rotational springs linking the rigid elements and two rotational springs located at the interface of the beam joint to take into account the bond-slip behavior of the beam longitudinal reinforcement. The constitutive parameters for bond-slip deformation were determined. According to the authors, the proposed joint model incorporates the behavior of RC beam-column joints of ductile moment-resisting frames designed and detailed based on current codes. (Mitra et al., 2007) modified the joint model of (Lowes and Altoontash, 2003) to improve the overall joint response mechanisms and anchorage zone response. The study showed that the modified model represented well the stiffness and strength parameters for joints.

Based on these models, many studies were published showing the effect of the beam-column joints' behavior on the global performance of structures subjected to cyclic loads. (Shafaei et al., 2014) investigated the effect of joint flexibility on the lateral response of RC frames. The study showed that the analytical results of a modified joint element accurately predict the behavior of beam-column joints by taking into account the detailing of reinforcement in the joint region, the slip of the beam longitudinal reinforcement in the joint, and shear deformation of the joint area. Moreover, RC structures with deficient beam-column joints are vulnerable, especially under severe earthquakes. (Rajeev et al., 2020) investigated the behavior of RC external beam-column joints subjected to dynamic loads using analytical models and experimental tests. The authors concluded that any additional transverse reinforcement decreases the development of shear cracks in the joint area. (Park et al., 2013) proposed a multi-linear backbone curve for beam-column joint, which was used to assess the seismic performance of non-ductile RC buildings. OpenSees program was utilized to perform the non-linear dynamic analysis based on the proposed backbone curve. The study showed the importance of joint flexibility in assessing the seismic performance of non-ductile reinforced concrete buildings. Furthermore, the derived backbone curve reflected the actual behavior of unreinforced beam-column joints compared with the rigid joints.

(Ahmad et al., 2021) performed a non-linear dynamic analysis on reinforced concrete moment resisting frames with/without beam-column joint detailing to evaluate the significance of joint detailing on the performance of the building under dynamic loads. It was found that the frames with shear reinforcement in beam-column joints have a collapse margin ratio (CMR) equal to 11% higher than the acceptable. However, frames without shear reinforcement in beam-column joints have a CMR equal to 29% less than the other frames with proper joint detailing. Thus, the beam-column joint detailing dramatically affects the dynamic behavior of RC moment-resisting frames.

(K. Ramanjaneyulu et al., 2013) compared the seismic performance of an exterior reinforced concrete beam-column joint according to Eurocode (EC8) and Indian Standard (IS) codes. The study showed that structures designed for gravity loads only are vulnerable to medium earthquake intensities. Moreover, structures designed according to ductile detailing provisions in the Indian Standard (IS) have a better performance than the structures designed based on the ductile detailing provisions in the Eurocode (EC8). Therefore, the authors concluded that the ductile detailing provisions influence the seismic performance of reinforced concrete structures in different standards. (Abdelwahed, 2019) also investigated the behavior of RC beam-column joints in the case of seismic loading. OpenSees program was utilized to perform the analysis. The joint models were selected from the OpenSees library to account for the expected shear deformation and the bar slip. The results showed that the lack of shear reinforcement in beam-column joints reduces the ultimate joint capacity.

Another significant deficiency is the length of longitudinal bars in both members and beam-column joints. This deficiency can affect the bond-slip failure criteria between reinforced bars and concrete. Many researchers highlight the importance of incorporating the bond-slip between longitudinal bars and concrete. (Calvi et al., 2002) experimentally assessed the damage and collapse of the beam-column joints in RC frames. It was concluded that the inadequacy of beam-column joint detailing, such as the lack of end-hook anchorage in beams and smooth bars, may cause strength degradation leading to a brittle failure mechanism. (Goksu et al., 2014) investigated the effect of the lap splices on the lateral behavior of reinforced concrete members. The study showed that the presence of the hooks in lap splices decreased the negative effect of poorly lap splices on RC member performance. (Eshghi et al., 2008) investigated the cyclic behavior of a slender reinforced concrete column with lap splices through experimental work. The data were verified using the plastic hinge method. The study confirms that the absence of well lap splices in the plastic hinge region significantly affects the member's ductility. Moreover, the bond-slip effect in the splice region on the lateral deformation is substantial and should not be ignored in the analysis.

Many researchers proposed analytical models to include the bond-slippage effect on the response of reinforced concrete elements. In addition to the studies mentioned above, (Ning et al., 2016) proposed a model for beam-column joints under dynamic loading. The bond-slip mechanism of longitudinal reinforcement and the shear deformation mechanism in the joint concrete core were modeled as eight springs and a one-panel zone. The model was calibrated using experimental data for non-seismically detailed joints. (D'Amato et al., 2012) developed a modified steel bar model taking the bond-slip of longitudinal bars into account. According to the authors, the model can predict axial slip deformations with accurate results. (Mergos et al., 2012) built a beam-column model for the seismic analysis of reinforced concrete structures. The model components are two gradual spread inelasticity elements to encounter inelastic flexure and shear response. The model also includes two rotational springs at the end of the member to represent the anchorage slip effect. The study showed that the model could capture the hysteretic response and the type of failure in the member.

(Wei-Chih Lin, 2013) studied the effects of insufficient lap splices in deficient reinforced concrete frames on the system performance. Both experimental and analytical approaches were used to find the response of structures. The authors concluded that in the case of laboratory tests, the presence of a 25% deficient lap splices length does not affect the structures' overall response. However, the analytical models using Opensees program results were not sufficient to catch local deformations such as end rotations.

Furthermore, it can be observed that the inclusion of structural deficiencies (insufficient beam and column shear reinforcement, short anchorage length in longitudinal beam bars, and inadequate beam-column joint shear reinforcement) in the analysis of frame structures influences the seismic performance of reinforced concrete structures by reducing the ultimate system capacity and ductility. Therefore, structural deficiencies shall be considered in modeling and analysis stages to accurately predict the performance of building structures and, consequently, the response modification factor.

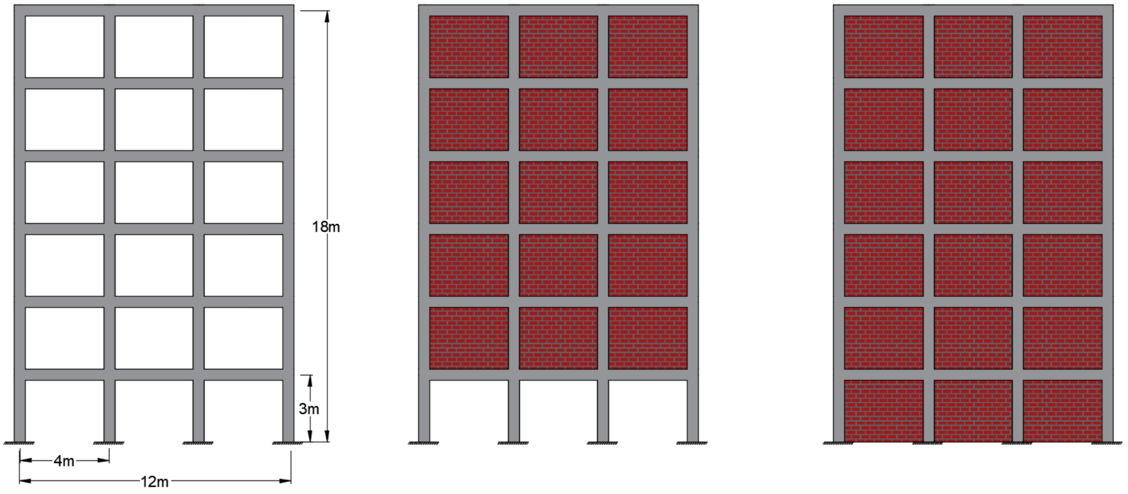
CHAPTER 5: SELECTION OF STRUCTURAL BUILDING MODELS

5.1 Description of the Case Study Buildings:

In this study, in order to represent structural building systems in Palestine, semi-structured interviews were conducted and illustrated in section 4.1. The common characteristics of the buildings are identified. The geometrical data selected for RC-frames are also summarized in Table 5-1. Based on the research argument, the performance of the structural buildings is controlled by the line framing with a minimum R-factor (less ductile line frame system) (ASCE 41-17). line framing combinations are used in this study (2D-analysis) as each line frame type is analyzed alone. The external line framing is considered in the analysis due to the high stiffness of infills. Therefore, twelve 2D RC-moment resisting frames (MRFs) were chosen as typical RC-frames in Palestine. Two prototype RC-MRFs are three bay-six storey (3B6S) as bare frame and bare frame with structural deficiencies. Four prototypes RC-MRFs three bay-six storey (3B6S) are used to investigate the effect of the infill walls (stone-concrete and concrete masonry) on the global response, as shown in Fig 5-1. In contrast, two prototype RC-MRFs are three bay-nine storey (3B9S)) as bare frame and bare frame with structural deficiencies. The other four 3B9S frames are to consider the effect of infill walls on the structural performance, as shown in Fig 5-2.

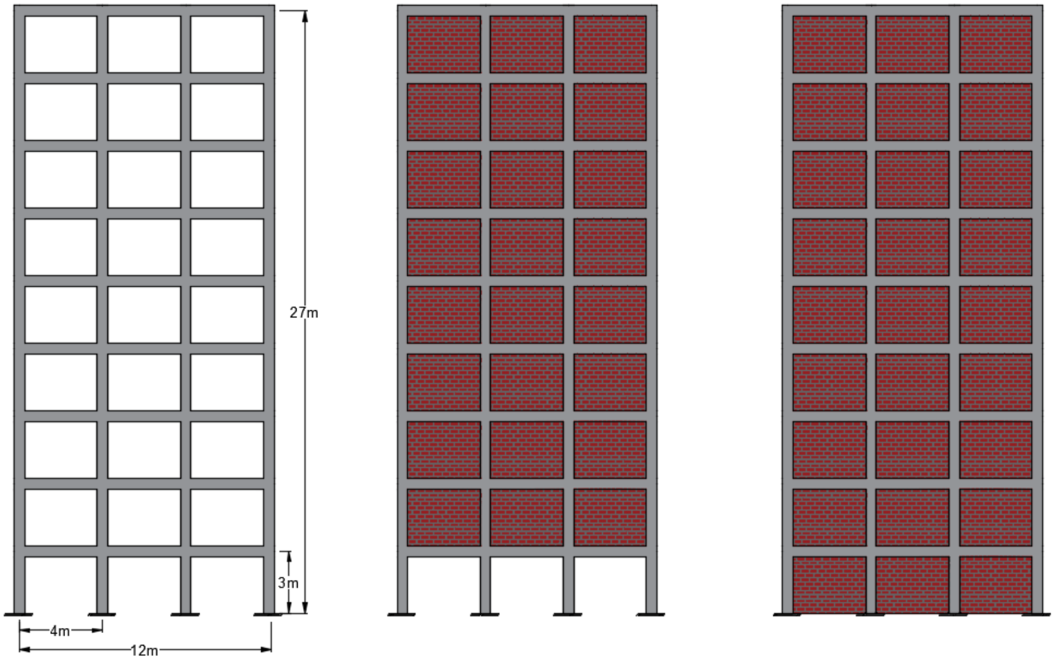
Table 5-1: Selection of Model Characteristics.

Category	Common characteristics	Selected characteristics
Construction material	Reinforced concrete	Reinforced concrete
Structural system	MRFs	MRFs
Number of stories	4-6, 6-9	6 and 9
Number of bays	3-6	3
Plan regularity	Regular	Regular
Plan symmetry	Symmetrical	Symmetrical
Elevation regularity	Regular	Regular
Occupancy	Residential and commercial	Residential and commercial
Floor area (m^2)	100-500	100-500



a) b) c)

Fig 5-1: Elevation of three bay-six storey (3B6S), a) bare frame/with deficiencies b) infilled frame without ground infills (stone-concrete/concrete masonry) c) infilled frame (stone-concrete/concrete masonry)



a) b) c)

Fig 5-2: Elevation of three bay-nine storey (3B9S), a) bare frame/with deficiencies b) infilled frame without ground infills (stone-concrete/concrete masonry) c) infilled frame (stone-concrete/concrete masonry)

Moreover, these prototype RC-MRFs are based on typical existing buildings constructed in Palestine and are assumed to be designed according to SEI/ASCE7-16 and ACI 314 as intermediate moment-resisting frames. The plan of typical buildings in Fig 5-3 shows multiple line framing of three bay-six and nine stories. Two typical line framing were chosen to represent the three bay-six storey (3B6S) and the three bay-nine storey (3B9S). It can be seen from the figure mentioned earlier that most of the structure's slabs are ribbed slabs with hidden beams or flat slabs. The absence of drop-beams pressing to include the column strip to represent the beams in the modeling and analysis. Fig 5-4 shows cross-sections and reinforcement detailing of typical beams and columns. Full structural detailing for 3B6S and 3B9S ductile and non-ductile frame systems is shown in Appendix 16.

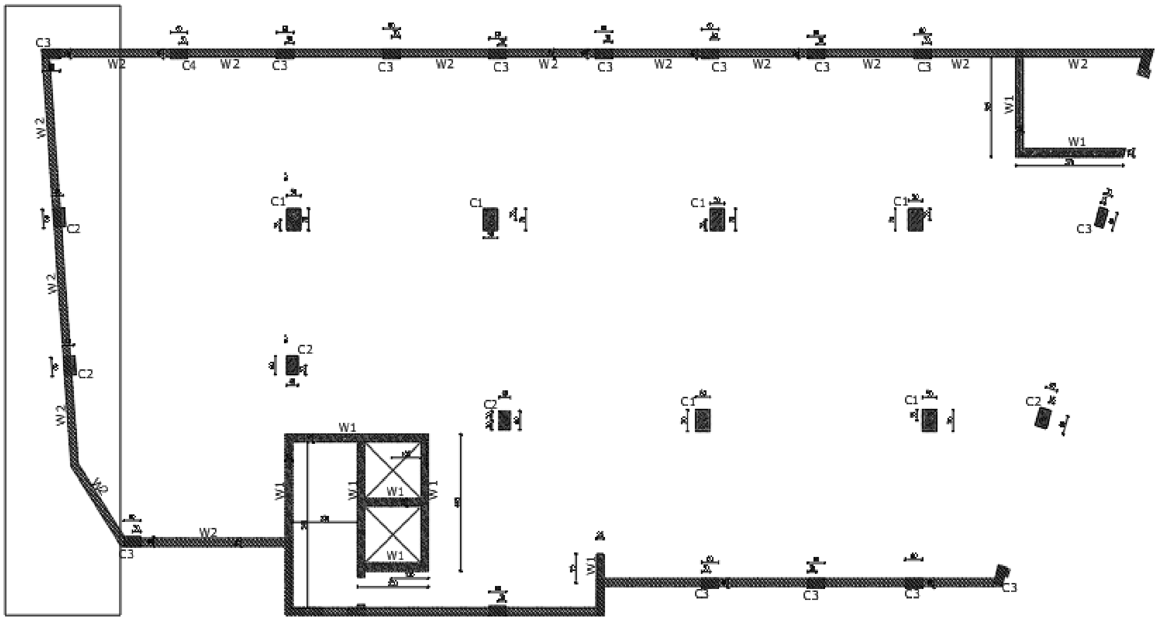
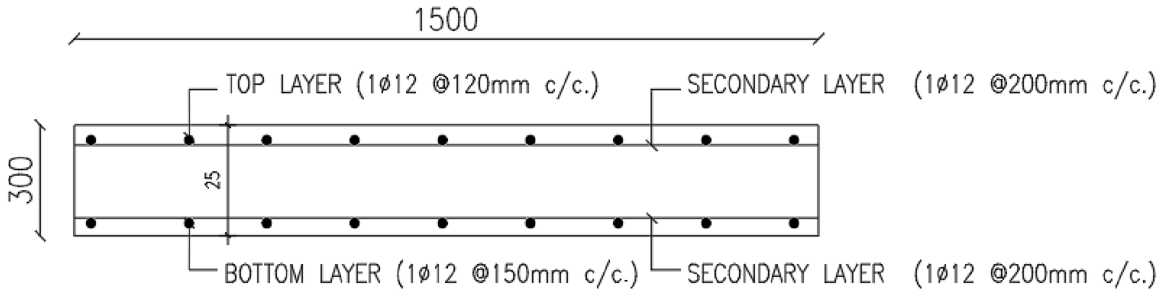
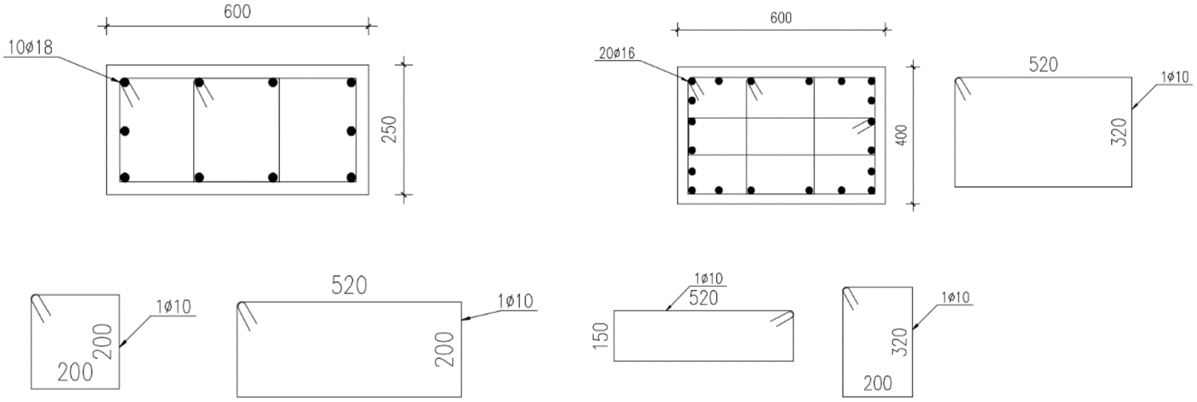


Fig 5-3: Floor plan of an existing prototype three-bay-six and nine-storey structure.



Cross-section and reinforcement detailing (B1)



Cross-section of columns (C1 and C2)

Fig 5-4: Cross-section of the beam (B1) and column (C1) for three bay-six storey. Beam (B1) and Column(C2) are for three bay-nine storey structures.

5.2 Material Properties:

Concrete and steel materials are the main materials used in building structures in Palestine. Based on the interviews (see section 4.1), the concrete has a wide range of design cylinder compressive strength that is used in designing reinforced concrete moment-resisting frames. A typical concrete cylinder compressive strength of 28 MPa is taken for the purpose of this research. In contrast, the steel reinforcement used in reinforced concrete moment resisting frames usually has a yield strength of 420 MPa. Modulus of elasticity in most cases is taken as 200000 MPa. The reinforcement of beam and column bars ranges from 12 mm diameter to 25 mm diameter.

5.3 Gravity Loads used in Construction Practice:

Most of the reinforced concrete moment resisting frames in Palestine are designed for dead and live loads (gravity loads) according to both SEI/ASCE7-16 and UBC 1997 standards. In this research, each prototype model is assigned to a superimposed dead load of 4.5 kN/m2 and a live load of 3 kN/m^2. The weight of infill walls is added as a line load along the beam's length, which equals 20 kN/m. The self-weight of beams and columns based on tributary area is considered with a concrete density equal to 24 kN/m^3. All the assigned loads are converted to concentrated loads and placed at the nodes where the lump masses of the floors exist to avoid convergence problems that may occur in the program.

CHAPTER 6: NUMERICAL MODELING AND ANALYSIS

6.1 Modeling Criteria:

Numerical models for the examined structural systems are developed using an open-source finite element program to investigate the performance of the buildings considering the identified deficiencies. OpenSees is used for this purpose; it is a program designed especially for non-linear seismic analysis. It is utilized to perform a non-linear static pushover analysis for different models based on the collected data. The software has a library with an extensive number of material and element model definitions that capture the non-linear behavior of sections under bending, shear, axial forces and their interactions. The models are created in detail to represent the structural elements and their deficiencies. This level of detailing will provide adequate accuracy for the analysis results to better understand the overall behavior of the building. The structural elements beams, columns, and shear walls are defined as non-linear elements with fiber sections to model the frame elements. These elements are discretized to model longitudinal bars and concrete fiber sections see Fig 6-1.

Furthermore, uniaxial material models predefined in OpenSees are utilized to define the constitutive behavior of concrete and steel materials. According to Mazzoni et al. (Mazzoni et al. 2006), the section's force–deformation relationship is evaluated through numerical integration of the non-linear uniaxial material constitutive behavior of the fibers. The force-displacement behavior of the element is obtained by numerical integration of the section force–deformation behavior along the length of the element. The constitutive material model defined as concrete 02 is used to model confined and unconfined concrete, see Fig 6-2a. This model considers both compression and tensile strength of the concrete material (Yassin, 1994). Steel 02 is taken as a constitutive material model for steel reinforcement see Fig 6-2b, (Filippou, 1983). The distributed plasticity model is used to model the nonlinearities along the beam and column elements.

Moreover, non-structural elements such as infills (masonry walls and concrete stone walls) are modeled using simplified macro-models such the equivalent strut model to include the effect of the infills on the in-plane behavior of the structures. The beam-column joint is modeled according to Lowes and Altoontash (Lowes and Altoontash, 2003). The authors proposed a joint model with four nodes and 12 degrees of freedom (DOF), representing three types of inelastic mechanisms of beam-column joints under cyclic loading. According to the authors, the bonding-slip response of beam-column longitudinal reinforcement is modeled as eight zero-length translational springs. Furthermore, a zero-length rotational spring is used to represent the joint's shear deformations, and four zero-length shear springs are used to simulate the interface-shear deformations see Fig 6-3.

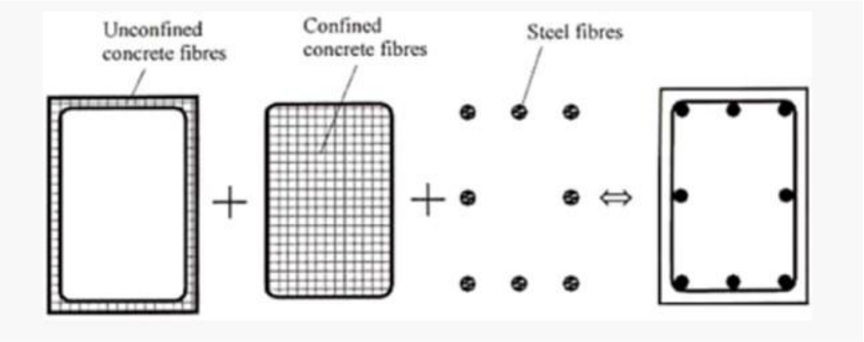
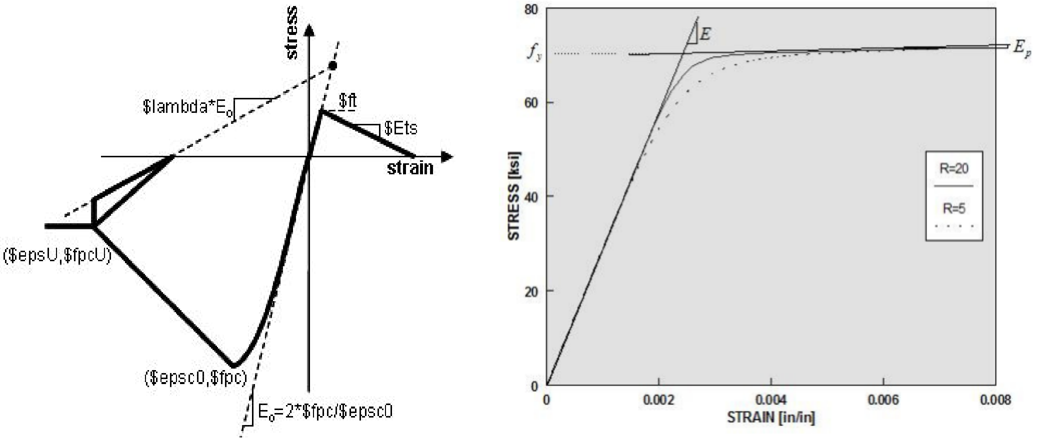


Fig 6-1: Modeling Technique Utilized in OpenSees; Fiber-Section Discretization Approach (Pahlavan et al., 2015).



a)

b)

Fig 6-2: a) Stress-Strain Relationship Assumed for the Concrete Material; Confined Concrete, and Unconfined Concrete (Mazzoni, 2006). b) Stress-Strain Relationship Assumed for the Steel Reinforcement (Mazzoni, 2006).

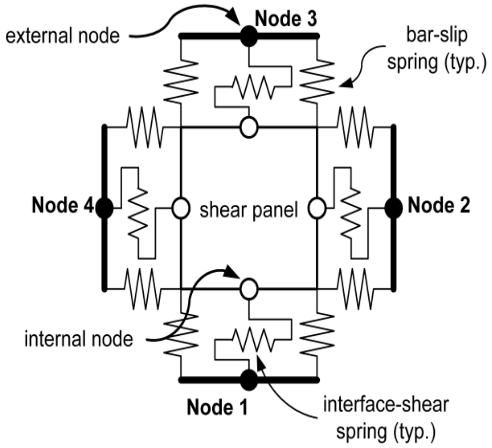


Fig 6-3: Beam-Column Joint Model (Mazzoni, 2006).

6.2 Constitutive Material Model Parameters:

6.2.1 Concrete:

Unconfined and confined concrete are modeled using the Concrete02 constitutive model, which considers the linear tension softening of concrete. The parameters required to define Concrete02 are compressive strength (f_{pc}), strain at compressive strength (ep_{sc0}), ultimate strength (f_{pcu}), strain at ultimate strength (ep_{su}), tensile strength (f_t), tension softening stiffness (E_{ts}), the ratio between unloading slope at (ep_{scu}) and the initial slope (E_o). The initial gradient (E_o) is automatically computed as $2*f_{pc}/ep_{sc0}$.

In the case of unconfined concrete, the compressive strength is the same as the design value, while the strain at maximum strength (ε'_c) is assumed to be 0.002. The strain at the ultimate stress equals $2(\varepsilon'_c)$, and the residual stress is assumed as 20% of the compressive stress.

Furthermore, the parameters of the confined concrete are determined based on a study by (Mander et al., 1988) to take into account the confinement due to transverse reinforcement. The derived equations in the study are based on a balanced energy approach. The lateral confining pressure (f_l) is determined with the assumption that the transverse reinforcement has yielded. Moreover, Equations (13) and (14) define the lateral confining pressure, which is reduced by (K_e) to exclude the concrete cover from the lateral pressure. Fig 6-4 shows the effective confined core by rectangular hoop reinforcement.

$$f_l' = k_e \qquad (13)$$

$$k_e = A_e/(A_c * (1 - p_{cc}) \qquad (14)$$

Where f_l' is the effective lateral confining pressure, k_e is the confinement effectiveness ratio, A_e is the effective confined concrete area, A_c is the entire core area, and p_{cc} is the ratio of the area of longitudinal reinforcement to the area of the core section.

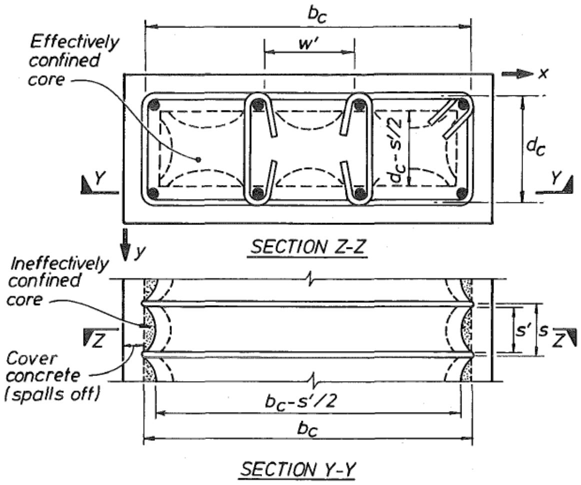


Fig 6-4: The Effective Confined Core for Rectangular Hoop Reinforcement (Mander et al., 1988).

The lateral confining pressure is calculated on both sides of the rectangular section using Equations (15) and (16), and these values are used to calculate the confined strength ratio f_{cc}'/f_{co}' based on the chart provided by (Mander et al., 1988), as shown in Fig 6-5.

$$f_{lx}' = \frac{A_{sx}}{s * d_c} * k_e * f_{yh} \qquad (15)$$

$$f_{ly}' = \frac{A_{sy}}{s * b_c} * k_e * f_{yh} \qquad (16)$$

Where $A_{sx}\ and\ A_{sy}$ are the total area of transverse bars running in each direction, s is the spacing between hoops, $d_c\ and\ b_c$ are the dimensions of the effective confined core, and f_{yh} is the yielding stress of transverse reinforcement.

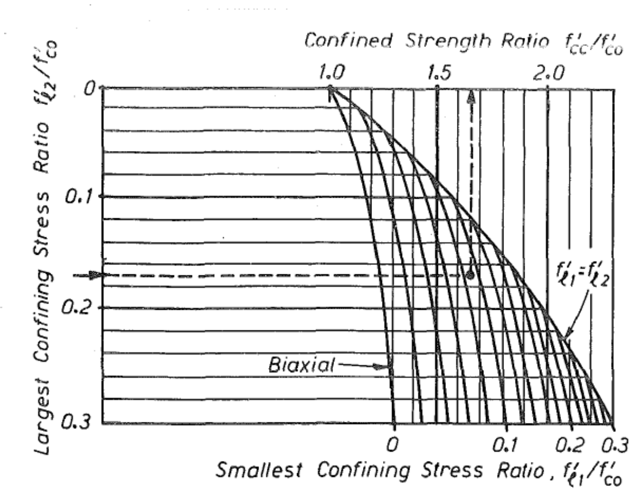


Fig 6-5: Confined Strength Determination from Lateral Confining Pressure for Rectangular Sections (Mander et al., 1988).

After the calculation of the confined strength ratio f'_{cc}/f'_{co}, the compressive strength of confined concrete f'_{cc} can be found. The strain corresponding to the maximum stress can be found using Equation (17). In order to calculate the ultimate strain, (Mander et al., 1988) assumed that failure state is when first hoop reaches fracture. Using the energy balance approach, the energy stored in the transverse reinforcement increase the available ductility of the confined concrete members. The stress-strain curves for unconfined and confined concrete in Fig 6-6 were used by (Mander et al., 1988) to evaluate the ultimate strain. Equation (18) can be used to calculate the ultimate strain for confined concrete. The equations mentioned above that determine the variables of the constitutive models for concrete are used in the current study.

$$\epsilon'_{\mathrm{cc}} = \epsilon'_{\mathrm{co}} * \left(1 + 5 * \left(\frac{f'_{cc}}{f'_{c}} - 1\right)\right) \qquad (17)$$

$$110\mathrm{ps} = \int_0^{\epsilon'_{\mathrm{cu}}} f_c d\epsilon_{\mathrm{c}} + \int_0^{\epsilon'_{\mathrm{cu}}} f_{sl} d\epsilon_{\mathrm{c}} - 0.017 * \left(f'_{co}\right)^{0.5} \; (18)$$

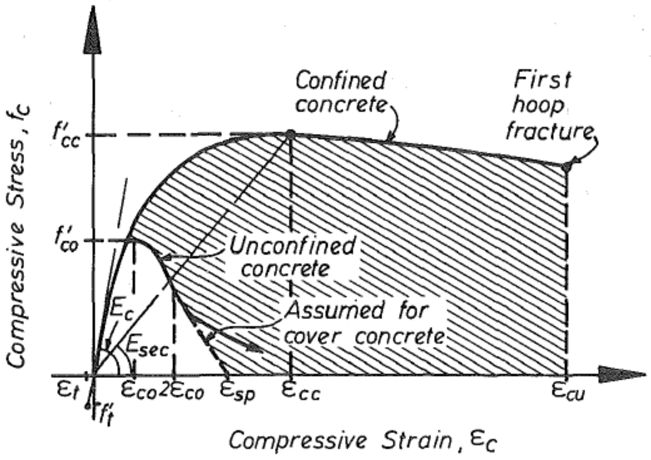


Fig 6-6: Stress-Strain Model Proposed for Monotonic Loading of Confined and Unconfined Concrete (Mander et al., 1988).

6.2.2 Steel:

Steel reinforcement is modeled using the Steel02 constitutive model. The input parameters required to define Steel02 model are yield strength (F_y), initial elastic tangent (E), strain hardening ratio (b), which is defined as ratio between post-yield tangent and initial elastic tangent, and three constants (R_0, cR_1, cR_2) in order to control transition from elastic to plastic range. Fig 6-2b shows the material law with the main input variables.

6.3 Beam-Column Joint Input Parameters:

The beam-column joint is modeled according to Lowes and Altoontash (Lowes and Altoontash, 2003). The bonding-slip response of beam-column longitudinal reinforcement is modeled as eight zero-length translational springs. Uniaxial Material Bar slip constitutive model is used to define the bonding-slip response, which was calibrated by (Lowes and Altoontash, 2003) as shown in Fig 6-7. The input parameters that define the Bar slip constitutive model are compressive strength of the concrete (f_c), (yield strength (f_y), modulus of elasticity (E_s), ultimate strength (f_u), hardening modulus (E_h), development length (I_d), and diameter (d_b)) of the reinforcing steel bars, number of anchored bars (n_b), dimension of the member (beam or column) perpendicular to the dimension of the plane (depth), dimension perpendicular to the direction in which the reinforcing steel is placed (height), the bond strength of anchored bars (bs_{flag}), (strong or weak), and position of reinforcement, (beam's top, beam's bottom, or column).

The four zero-length shear springs simulate the interface-shear deformations and are defined as uniaxial Material Elastic. Furthermore, the zero-length rotational spring represents the joint's shear deformations. Uniaxial material pinching is used to determine the pinched load-deformation response. The input parameter of pinching material is the envelope stresses and strains, which define the backbone curve for the shear panel, as shown in Fig 6-8. The envelope points of the backbone curve can be determined for joints with shear reinforcements based on the equations from (Kim and LaFave, 2009).

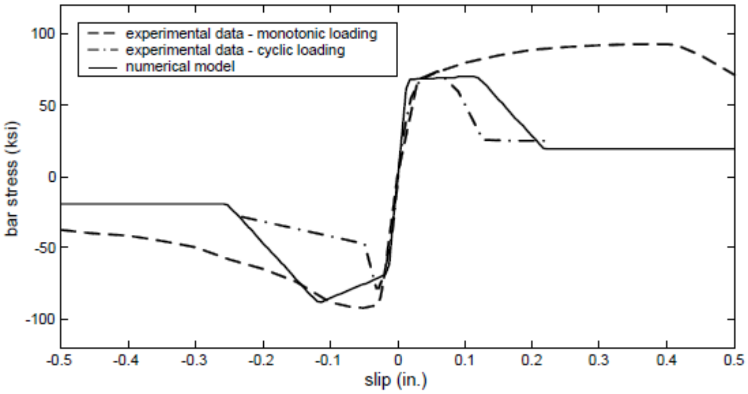


Fig 6-7: Envelope for Hysteretic Bar Slip Versus Slip Response (Lowes and Altoontash, 2003; Viwathanatepa et al. 1979).

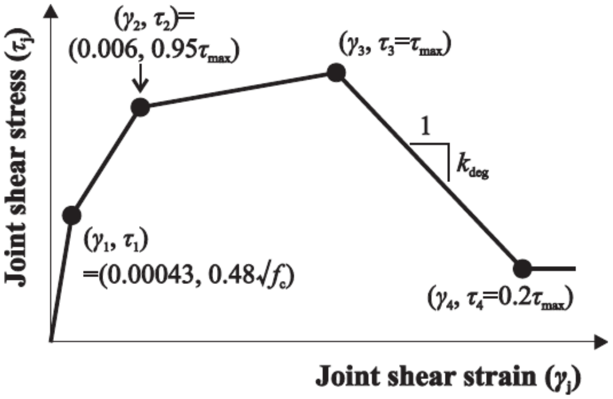


Fig 6-8: Envelope of Joint Shear Stress-Strain Relationship (Jeon et al., 2015).

The first two points of the backbone curve are taken from (Anderson et al. 2008), which was developed for beam-column joints without transverse reinforcement. The third point is determined based on Equations (19) and (20) from (Kim and LaFave, 2009), shown below. The fourth point represents the residual strength which is equal to 20% of the maximum shear strength. In the case of beam-column joints without transverse reinforcement, the first two points are the same. The maximum shear strength at the third point is calculated by extrapolating the second point and third point, knowing the strain at the maximum shear strength (third point) and the slope of the segment (e2-e3). Fig 6-9 shows the backbone curve for joints without transverse reinforcement.

$$v_i\ (MPa) = 1.07\alpha_1 * \beta_1 * \eta_1 * (JI)^{0.15} * (BI)^{0.3} * (f_c')^{0.75} \quad (19)$$

$$\gamma\ (rad) = \alpha_{\gamma 1} * \beta_{\gamma 1} * \eta_{\gamma 1} * (JI)^{0.10} * BI * \left(\frac{v_i'}{f_c}\right)^{1.75} \quad (20)$$

where α_1 is a parameter that describes the in-plane geometry 1.0 for interior connections, 0.7 for exterior connections, and 0.4 for knee connections; β_1 a parameter describing the out-of-plane geometry, 1.0 for subassemblies with zero or one transverse beam, and 1.18 if there are two transverse beams; η_1 describes the joint eccentricity; JI is the joint transverse reinforcement index, which equals the volumetric joint transverse reinforcement ratio multiplied by the ratio of the yield stress of joint transverse reinforcement to the compressive strength of concrete. BI is the beam reinforcement index that equals the beam reinforcement ratio multiplied by the ratio of the yield stress of beam reinforcement to the compressive strength of concrete.

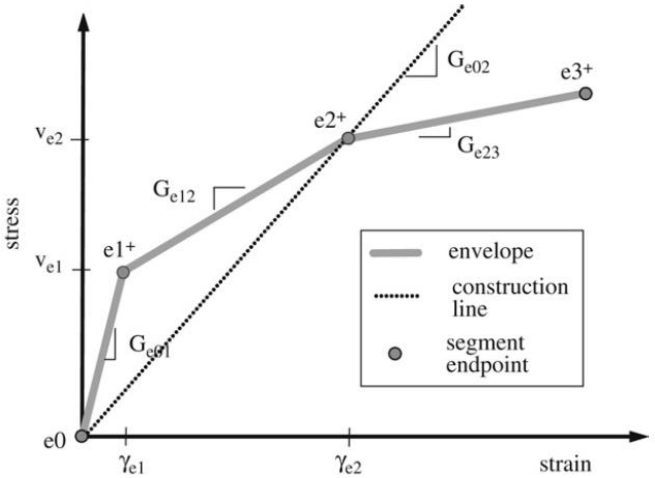


Fig 6-9: The Backbone Curve for Joints without Transverse Reinforcement (Anderson et al. 2008).

The joint model described previously and the relations required to define its constitutive models are used in the current study.

6.4 Infill Walls Modeling:

6.4.1 Infill Walls Modeling Methods:

In the literature, there are many different modeling techniques to simulate the infill walls' behavior, which can be divided into two main categories, namely the micro and macro models. In the micro model, the infill panel is discretized into numerous elements, which consider the local effect of each component in the infill walls, such as mortar, bricks, and the interface element. This type of model is suitable for local analysis and not for global analysis due to the large number of parameters that should be calibrated. In contrast, the macro model is the most common technique utilized to model the infill walls because of its simplicity.

Furthermore, many researchers proposed simplified macro-models for infill walls, such as (Polyakov, 1960), who suggested that infills can be modeled as equivalent one-diagonal struts. (Holmes, 1961) modeled the infills as equivalent pin-joint diagonal struts having the same material and thickness as the infill walls. (Mainstone, 1974) proposed different methods for evaluating the effective width of equivalent diagonal strut based on experimental tests. However, (Zarnic and Tomazevic, 1988) developed a macro-model that takes into account the strength and stiffness of the infill walls, and (Saneinejad and Hobbs, 1995) predicted the stiffness and strength degradation of (Zarnic and Tomazevic, 1988) macro-model using a numerical model. (Dolsek and Fajfar, 2002) proposed a single strut model with a tri-linear response with elastic hardening.

In general, presenting infill walls as one single strut model was found to be insufficient to model the whole behavior of infill walls subjected to seismic actions. (Schmidt, 1989) proposed a double diagonal strut that takes into account the strength, stiffness of the infill wall, and frame-infill interaction. (El-Dakhakhni et al., 2003) modeled infill walls as three non-parallel struts to include the interaction between infill and RC-frame, which also captures the corner crushing failure mechanism. (Rodrigues et al., 2005) proposed a macro-model that simulates the non-linear behavior of infill walls subjected to cyclic loads. Each infill wall is represented by an equivalent bi-diagonal compression strut model with four strut elements (rigid behavior) and one central strut (non-linear behavior).

6.4.2 Infill Walls Modeling Criteria:

To perform an accurate analysis of the RC frame structures with infilled walls, the non-linear behavior of these structural elements should be taken into account. In this study, infill walls are modeled according to Rodrigues (Rodrigues, 2010), which uses an equivalent bi-diagonal-strut model. This model accurately represents the global response and energy dissipation of structures with infill walls. The typical bi-diagonal strut model represents the infill panels as four support strut elements with rigid behavior. The central strut element considers the non-linear behavior of the infill wall, as shown in Fig 6-10. Using OpenSees program, the four diagonal elements are modeled as elastic beam-column and non-linear beam-column for the central elements. The idealization of the central element nonlinear monotonic behavior is modeled as pinching material. The input parameters for infill walls constitutive models are discussed in the following sub-sections.

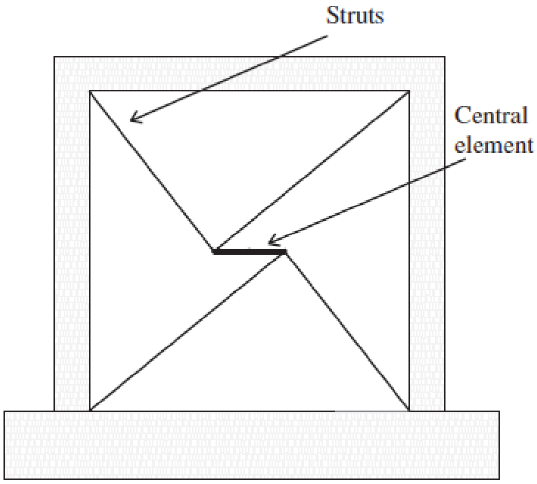


Fig 6-10: Macro-model for the simulation of an infill panel (Rodrigues. 2010).

6.4.2.1 Concrete Masonry Infill Walls Constitutive Model:

The parameters required to define the constitutive model can be obtained following different authors and international code recommendations. According to Mainstone (Mainstone, 1971), the thickness of the strut model (t_w) is the same as the infill wall, the width of the strut model can be evaluated using Equations (21) and (22), and the initial lateral stiffness K_{in} is based on Equation (23). Fig 6-11 shows the equivalent diagonal strut representation.

$$W_{eff} = 0.175(\lambda_h * H)^{-0.4} * (H^2 + L^2)^{0.5} \tag{21}$$

$$\lambda_h = \left(\frac{E_w * t_w * \sin(2 * \theta)}{4 * E_c * I_c * H_{in}}\right)^{0.25} \tag{22}$$

Where E_w and E_c are the modulus of elasticity of the infill wall and the concrete of the frame system, $\theta = \arctan\left(\frac{H}{L}\right)$ is the diagonal strut angle, H and L are the center-to-center height and width of the infill panel, H_{in} is the clear height of the infill panel, and I_c is the moment of inertia of the column of the frame system. The strut element carries only compression, and its constitutive model force-deformation envelope curve, as shown in Fig 6-12, is developed based on the equations and the recommended model values from (Mainstone, 1971; Matjaž and Peter, 2002). The maximum strength of infill walls is determined using the simplified Equation (24) (Žarnić and Gostič, 1997).

$$K_{in} = \left(E_w * W_{eff} * \frac{t_w}{(L^2 + H^2)^{0.5}}\right) * (\cos\theta)^2 \tag{23}$$

$$F_{max} = F_u = 0.818 * \frac{L_{in} * t_w * f_{tp}}{C_1} * (1 + (C_1^2 + 1)^{0.5}), C1 = 1.925 * \left(\frac{L_{in}}{H_{in}}\right) \tag{24}$$

Where $f_{tp}(F_c)$ is the cracking strength of the infill wall, and L_{in} is the clear width of the infill wall. The recommended values for these parameters are obtained from (Matjaž and Peter, 2002). The modulus of elasticity of the infill wall varied between 6.5 and 8.2 GPa, the modulus of elasticity of concrete is the same as designed value, the value of cracking strength of infill varied between 0.28 and 0.40 MPa, the displacement in the horizontal direction at maximum strength (u_u) ranges between 0.5 to 0.6% of the story height, and the softening segment has a stiffness equal to 5% of the initial stiffness. Moreover, according to FEMA-307. (FEMA-307, 1999), the diagonal cracking (u_c) tends to occur at 0.2 to 0.4% of the inter-story drift.

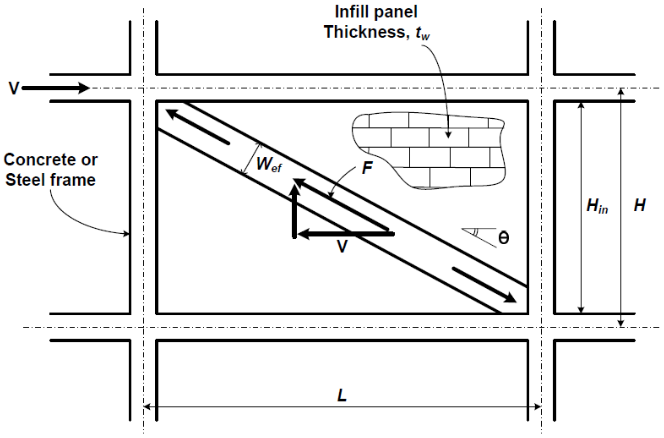


Fig 6-11: Equivalent Diagonal Strut Model (Ko et al. 2008).

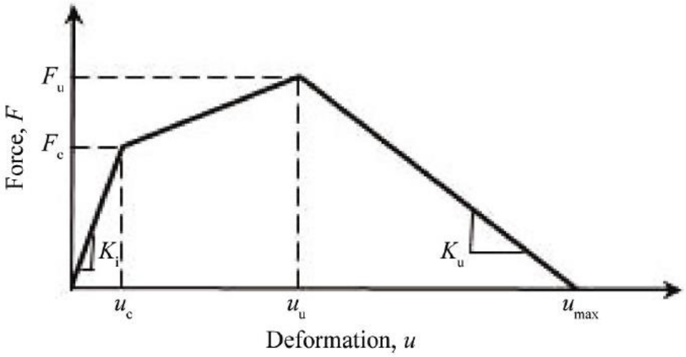


Fig 6-12: Force-Deformation Relationship for Masonry Infill Walls (Al-Nimry et al. 2015).

In this study, and based on the conducted interviews explained in section 4.1, the exterior concrete masonry walls are considered to have a 20cm thickness. And according to Al-Nimry et al. (Al-Nimry et al., 2015), the typical concrete masonry infills in Jordan have a unit compressive strength equal to 3.0MPa. The experimental data and the recommended input parameters for masonry published in (Al-Nimry et al., 2015) are used to represent the infill walls in Palestine. This is accepted as the construction practice in Jordan and Palestine is similar.

6.4.2.2 Stone-Concrete Infill Walls Constitutive Model:

(Al-Nimry, 2010, 2012, 2014) investigated the lateral response of RC frames with stone-concrete infill walls as shown in Fig 4-1 using quasi-static experimentations. The author concluded that the stone-concrete infill panels can be modeled using two non-linear link elements. Each element is assigned a multi-linear plastic property with only non-linear behavior in the axial direction. The constitutive model of the stone-concrete infill panel is shown in Fig 6-13, and the recommended envelope points for force-deformation definition are shown in Table 6-1. According to the author, the initial axial stiffness is estimated based on the equation from (Fajfar et al., 2001) as given in Equation (25), the modulus of elasticity of the infill panel (E_p) is assumed to be 14.0 GPa, and the effective width of the equivalent strut can be taken as one-tenth the length of the equivalent strut.

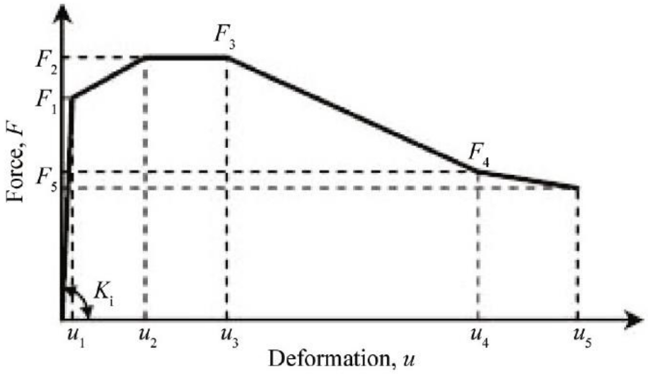


Fig 6-13: Force-Deformation Relationship for Stone-Concrete Infill Walls (Al-Nimry et al., 2015).

Table 6-1: Multi-linear force-displacement definition for the non-linear link element (Al-Nimry et al., 2015).

Point	Axial Deformation (mm)	Axial Force (kN)
(F1, U1)	1.290	540
(F2, U2)	10.50	639
(F3, U3)	20.79	639
(F4, U4)	52.50	360
(F5, U5)	65.00	350

$$K_{in} = \left(E_w * W_{eff} * \frac{t_w}{r}\right) \tag{25}$$

The stone-concrete infill panel model described previously and the relations required to define its constitutive models are used in this study as the typical cross-section of the stone-concrete infill panel presented by (Al-Nimry et al., 2015). The recommended values are checked by calculating the initial stiffness of the diagonal strut using Equation (25) and comparing it with the initial stiffness from Table 6-1. The initial stiffness calculated from Equation 25 using $H = 3m,\ L = 4m,\ and\ t_w = 0.3m$ equals 420000 kN/m, and the initial stiffness from the table using (F1, U1) equals 418604.7 kN/m.

6.5 Model Verification:

6.5.1 Data Collection:

In this research, a non-linear static pushover analysis is performed for RC frames. The numerical models for RC Frames are developed as explained in the previously explained sections. The numerical analysis results are validated using experimental data retrieved from the open literature. The data available on nonlinear analysis of RC-frame performed by (Vecchio and Emara, 1992) is utilized to verify the developed model. The followings are the structural and geometrical data of the RC frame (Vecchio and Emara, 1992).

1) The frame spans 3500mm c/c, the first story height is 2200mm c/c, and the second is 2000mm c/c.
2) The reinforcement detailing is shown in Fig 6-14. Beams and columns are 400mm in depth and 300mm wide.
3) Material properties:
   - The compressive strength of concrete is 30MPa
   - Steel properties: yield strength is 418MPa, ultimate tensile stress is 596MPa, modulus of elasticity is 192500MPa, and stain hardening modulus is 3100MPa.
   - The shear reinforcement has a yield strength of 454MPa and ultimate stress 640MPa.
4) 700kN axial load was applied to each column from the top end to activate the p-delta effect.

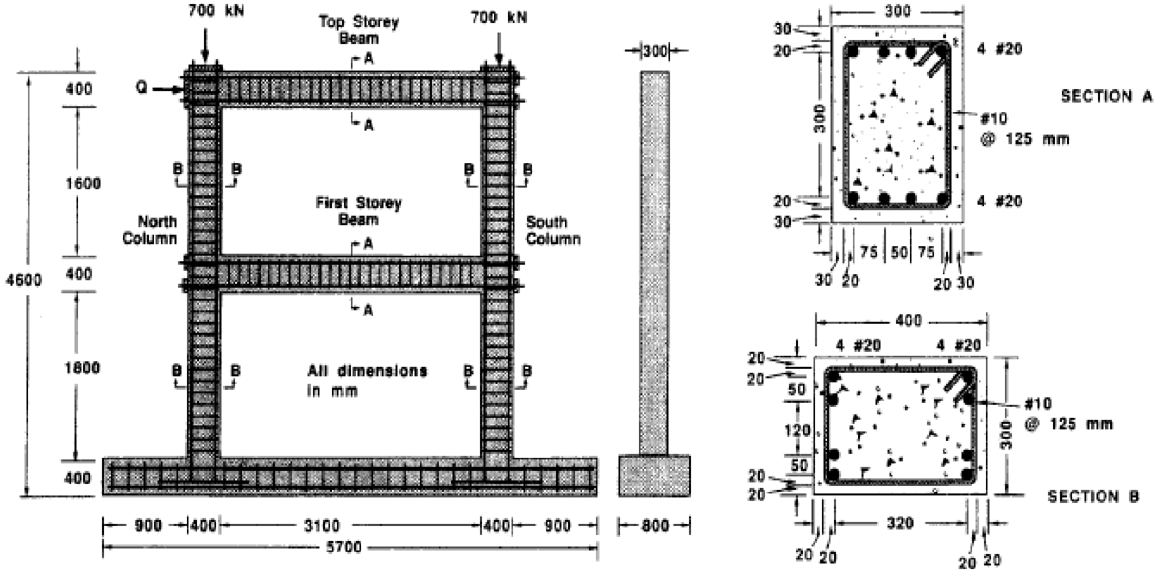


Fig 6-14: Details of the Frame (Vecchio and Emara, 1992).

6.5.2 Modelling Criteria and Results of Non-Linear Static Pushover Analysis:

The frame system was first modeled using beam and column elements without the beam-column joint model. Both beams and columns were modeled as force beam-column. The beam-column joint was added later to the model to investigate the effect of its modeling. The bar-slip spring stiffness was defined based on the existing material. The sheer panel constitutive model was built based on section 5.3. The results of both models were compared to the capacity curve provided by (Vecchio and Emara, 1992), as shown in Fig 6-15.

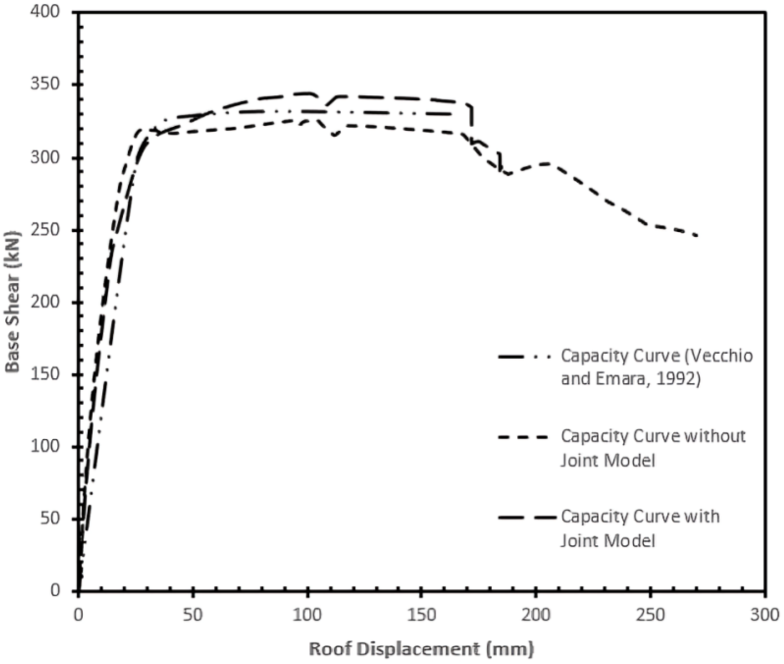


Fig 6-15: Verification of Pushover Analysis Model.

As seen in this figure, there is a good correlation between the pushover analysis results of tested frame and the analysis results of the developed numerical model. In the case of the capacity curve with the joint model, the base shear of the frame system is higher than the reference by 4.2%. However, it anticipates failure earlier than the numerical model without the joint model.

Furthermore, the model was used to examine the significance of the structural deficiencies in the construction practices on the capacity curve of the frame system. Inadequate stirrup spacing for both beams and columns, inadequate joint detailing, and short anchorage of longitudinal bars in the joint were considered. The analysis results are shown in Fig 6-16.

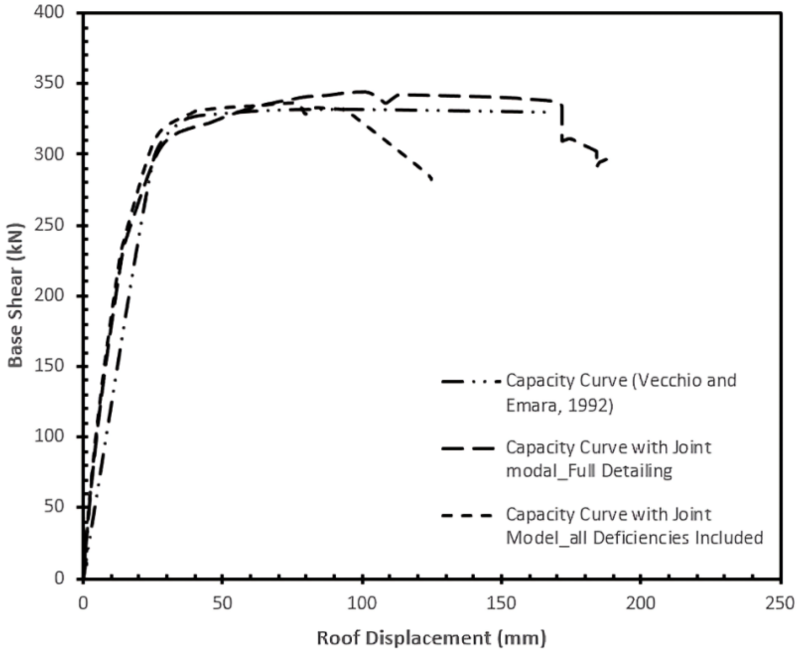


Fig 6-16: Capacity Curve Including the Deficiencies in the Model.

It can be seen that including the deficiencies keeps the base shear at almost the same level; however, it dramatically decreases the frame system's ductility. The response modification factor indicates the ductility of the frame system. Therefore, its value is affected by such deficiencies.

6.5.3 Response Modification Factor for Both Models:

Based on the capacity curves in Fig 6-15 and Fig 6-16 and the procedure explained in chapter three for evaluating the response modification factor R, the R factor was calculated for both models as the following:

- Model 1 (joint model with full detailing)

1) Ductility Factor (R_μ):

- $T = 0.1997s, u = \frac{\Delta_{max}}{\Delta_y} = \frac{172mm}{27mm} = 6.37$
- $R_u = (2 * 6.37 - 1)^{0.5} \ = 3.426$

2) Over Strength Factor (R0):

The maximum base shear (V_0) is found from the capacity curve. The designed base shear (V_d) is evaluated using Equivalent Lateral Force ELF in SEI/ASCE7-16 since the designed base shear (V_d) is not documented by the authors in the study (Vecchio and Emara, 1992).

- $R_s = \frac{V_0}{V_d} = \frac{344kN}{199.4kN} = 1.725$

3) Redundancy Factor (R_r):

- $R_r = 0.71, since\ it\ has\ one\ vertical\ seismic\ framing$

$$\boldsymbol{R = 3.426 * 1.725 * 0.71 = 4.196,}$$
$$\boldsymbol{which\ is\ less\ than\ the\ recommended\ by\ ASCE\ 7-16}$$

The value recommended by SEI/ASCE 7-16 for the ductile moment-resisting frame is 5, considered unconservative compared to the calculated value.

- Model 2 (joint model including all considered deficiencies such as insufficient beam and column shear reinforcement, short anchorage length in longitudinal beam bars, and inadequate beam-column joint shear reinforcement)

1) Ductility Factor (R_μ):

- $T = 0.1997s, u = \frac{\Delta_{max}}{\Delta_y} = \frac{96mm}{27mm} = 3.56$
- $R_u = (2 * 2.89 - 1)^{0.5} = 2.47$

2) Over Strength Factor (R_0):

- $R_s = \frac{V_0}{V_d} = \frac{334kN}{199.4kN} = 1.675$

3) Redundancy Factor (R_r):

- $R_r = 0.71, since\ it\ has\ one\ vertical\ seismic\ framing$

$$\boldsymbol{R = 2.47 * 1.675 * 0.71 =, 2.94}$$

It can be seen that there is a reduction in the R-factor, which indicates that including the structural deficiencies of construction practice in the models affect the R-factor.

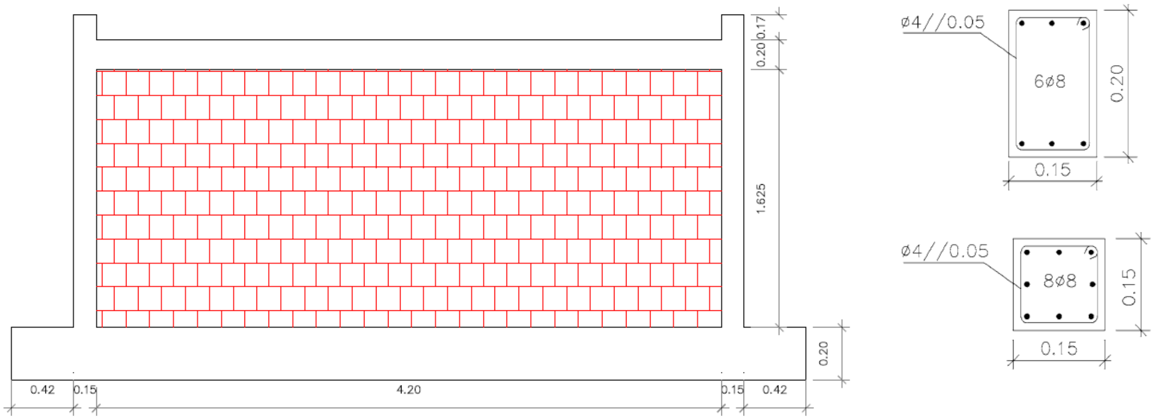
6.6 Infill Walls Modeling Verification:

6.6.1 Data Collection:

In the current study, infill walls are modeled according to Rodrigues (Rodrigues, 2010), which uses an equivalent bi-diagonal-strut model (see section 6.4). The numerical model with reinforced concrete frames is validated using experimental data retrieved from the open literature due to the significance of infill wall modeling and how that could affect the analysis results. The data available on the nonlinear analysis of the RC-frame performed by two authors are utilized to verify the developed model. The followings are the structural and geometrical data of the RC frame (Rodrigues et al., 2010) & (Cavaleri et al., 2004).

- **The data available from (Rodrigues et al., 2010)**

1) The frame spans 4350mm c/c, and the story height is 1825mm c/c.
2) The reinforcement detailing is shown in Fig 6-17. Beams are 200mm in depth and 150mm wide. Columns are 150mm in depth and 150mm wide
3) Material properties:
    - The compressive strength of concrete is 25.3MPa
    - Steel properties: yield strength is 434Mpa, and modulus of elasticity is 190000Mpa.
4) Material properties of infills:
    - The compressive strength of infill is 2.2MPa
    - Elasticity modulus E_m 3119MPa
    - Diagonal compression strength f_m 1.1MPa



(a) **(b)**

Fig 6-17: a) Single-story single-bay infilled masonry RC frame. b) Cross-section dimensions and detailing of RC beam and column (Vecchio and Emara, 1992).

- **The data available from (Cavaleri et al., 2004)**

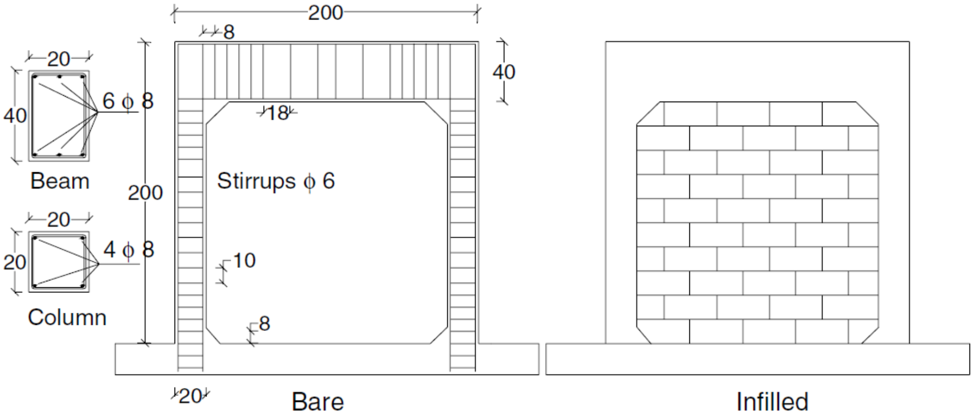
1) The frame spans 1800mm c/c, and the story height is 1800mm c/c.

2) The reinforcement detailing is shown in Fig 6-18. Beams are 400mm in depth and 200mm wide. Columns are 200mm in depth and 200mm wide

3) Material properties:
    - The compressive strength of concrete is 30Mpa
    - Steel properties: yield strength is 434Mpa, and modulus of elasticity is 190000Mpa.

3) 200kN axial load was applied to each column from the top end.

5) Material properties of infills:
    - The compressive strength of infill is 3MPa
    - Elasticity modulus E_m 7350MPa



a) b)

Fig 6-18: a) Single-story single-bay RC frame. Cross-section and detailing of RC elements. b) Single-story single-bay infilled masonry RC frame. (Cavaleri et al., 2004).

6.6.2 Modelling Criteria and Results:

The two frame systems were modeled using beam and column elements without the beam-column joint model. Both beams and columns were modeled as force beam-column elements. The infills were modeled according to Rodrigues. (2010) with the corresponding behavior of the masonry infill wall. The results of both models were compared to the capacity curves provided by both authors (Rodrigues et al., 2010; Cavaleri et al., 2004), as shown in Fig 6-19 and 6-20, respectively.

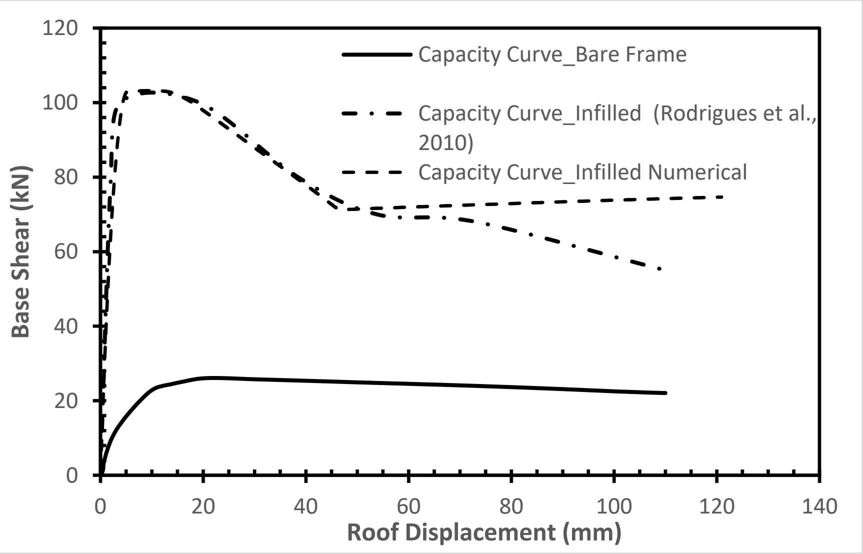


Fig 6-19: Verification of Pushover Analysis Model.

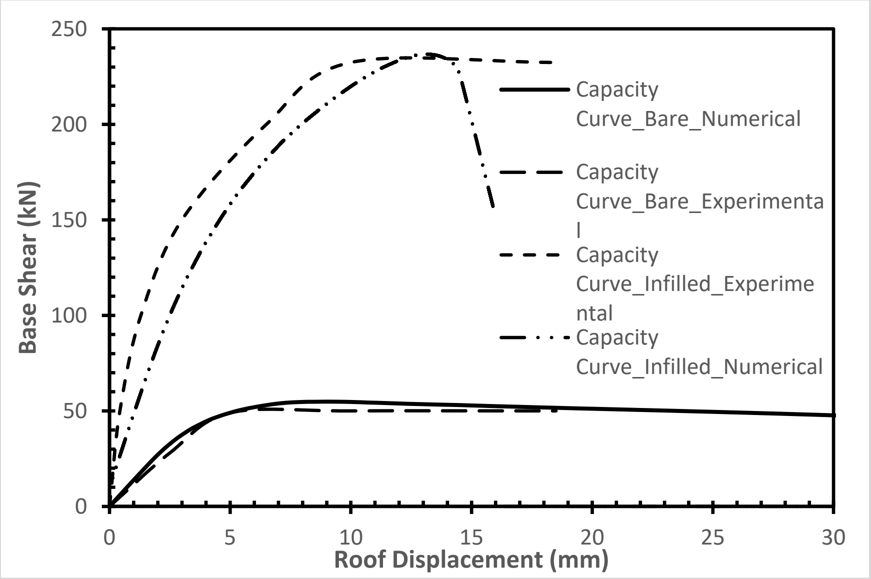


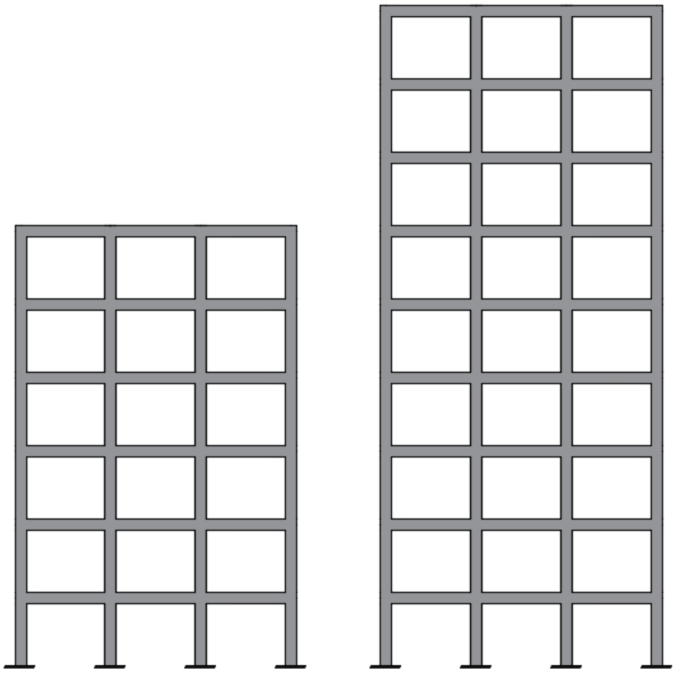
Fig 6-20: Verification of Pushover Analysis Model.

As seen in Fig 6-19 and Fig 6-20, there is a good correlation between the pushover analysis results of tested frames and the analysis results of the developed numerical models. Therefore, the model of infill walls and their corresponding material properties can be used in this research.

CHAPTER 7: EVALUATION OF R FACTORS OF PROTOTYPE FRAME SYSTEMS

7.1 RC MRF Prototypes and their Analysis:

This section discusses the non-linear analysis results of 2D reinforced concrete frames with and without infill walls. Twelve 2D RC-moment resisting frames (MRFs) were selected for the analysis to show the effect of structural deficiencies, infill walls (stone-concrete and masonry-concrete), and soft story mechanism (ground level without infills) on the value of response modification factor R, which is an indicator on the performance of the buildings, plastic energy capacity and ductility. The prototype models have the same height and length of bays which are 3 meters and 4 meters, respectively. The steel reinforcement bars have a yielding strength equal to 420 MPa, while the concrete compressive strength is considered as 28 MPa in this study. Fig 7-1 shows a typical prototype model for 3B6S and 3B9S.



a) b)

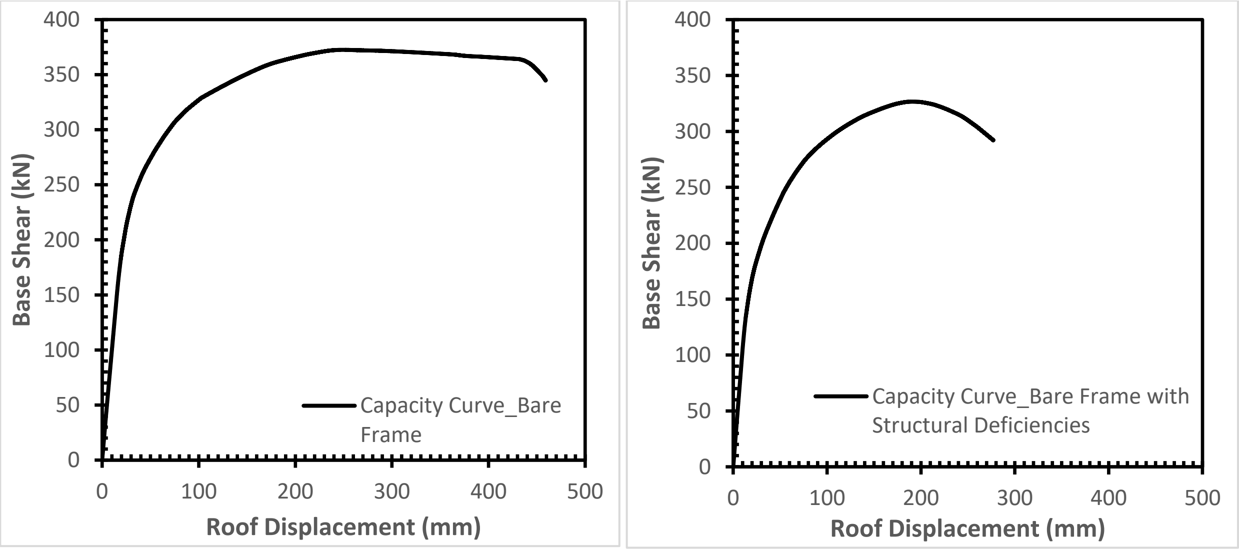
Fig 7-1: a) Elevation of three bay-six storey (3B6S) b) Elevation of three bay-nine storey (3B9S).

In order to perform non-linear analysis, modal analysis was conducted first on each prototype model to determine the fundamental elastic period of vibration. This parameter can be utilized to predict the accuracy of the analytical model before starting the non-linear analysis. Results from modal analysis for each prototype model are summarized in Table 7-1.

Table 7-1: Fundamental period for each prototype model.

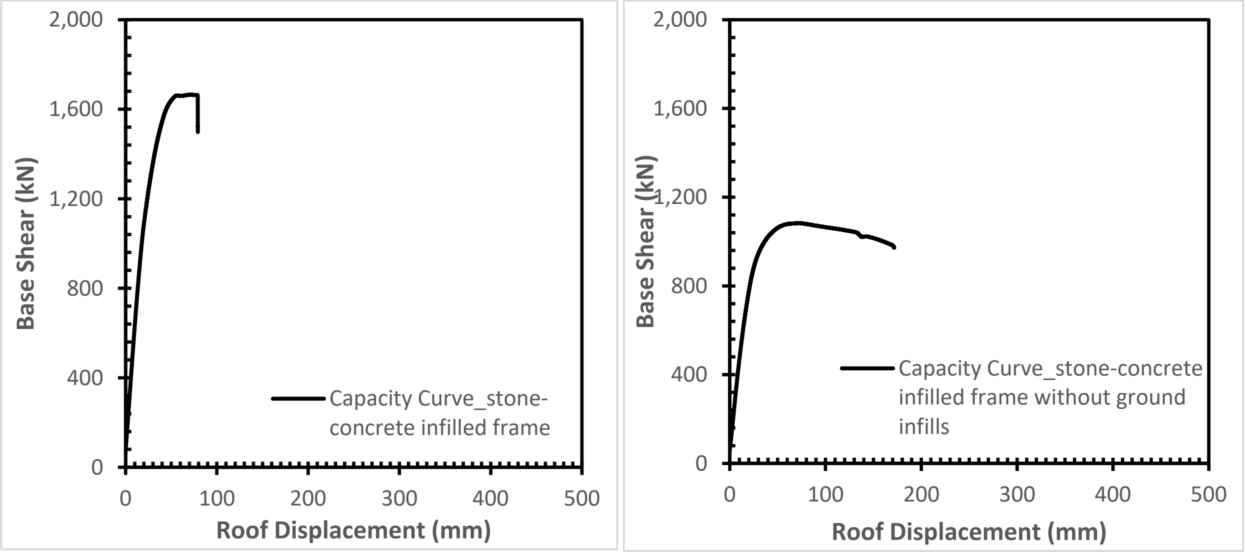
Building System		**Fundamental Period T_i (s)**
1)	3B6S MRFs-Bare Frame	**0.78**
2)	3B6S MRFs-Bare Frame with Structural Deficiencies	**0.75**
3)	3B6S MRFs-Stone-Concrete Infilled Frame	**0.49**
4)	3B6S MRFs-Stone-Concrete Infilled Frame without ground infills	**0.49**
5)	3B6S MRFs-Masonry-Concrete Infilled Frame	**0.45**
6)	3B6S MRFs-Masonry-Concrete Infilled Frame without ground infills	**0.45**
7)	3B9S MRFs-Bare Frame	**1.065**
8)	3B9S MRFs-Bare Frame with Structural Deficiencies	**1.01**
9)	3B9S MRFs-Stone-Concrete Infilled Frame	**0.70**
10)	3B9S MRFs-Stone-Concrete Infilled Frame without ground infills	**0.70**
11)	3B9S MRFs-Masonry-Concrete Infilled Frame	**0.67**
12)	3B9S MRFs-Masonry-Concrete Infilled Frame without ground infills	**0.67**

Furthermore, the non-linear static pushover analysis was performed for each prototype model using an inverted triangular load pattern. Each model was set to be pushed to a 10% drift to let the structure reach the maximum displacement in the pushover curve (capacity curve). The analysis procedure is utilized to assess the structural capacity and then evaluate the response modification factor R. The analysis results show a total of 12 pushover curves for the selected models. Fig 7-2a to Fig 7-7b present the pushover curves for each prototype model.



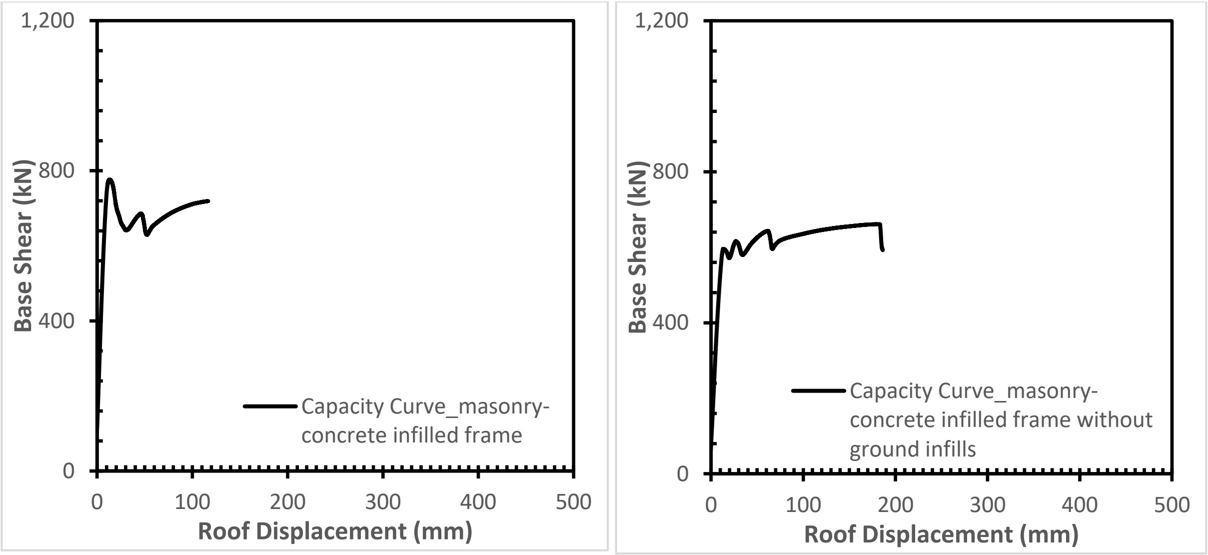
a) b)

Fig 7-2: a) Pushover curve of 3B6S MRFs-bare frame. b) Pushover curve of 3B6S MRFs-bare frame with structural deficiencies



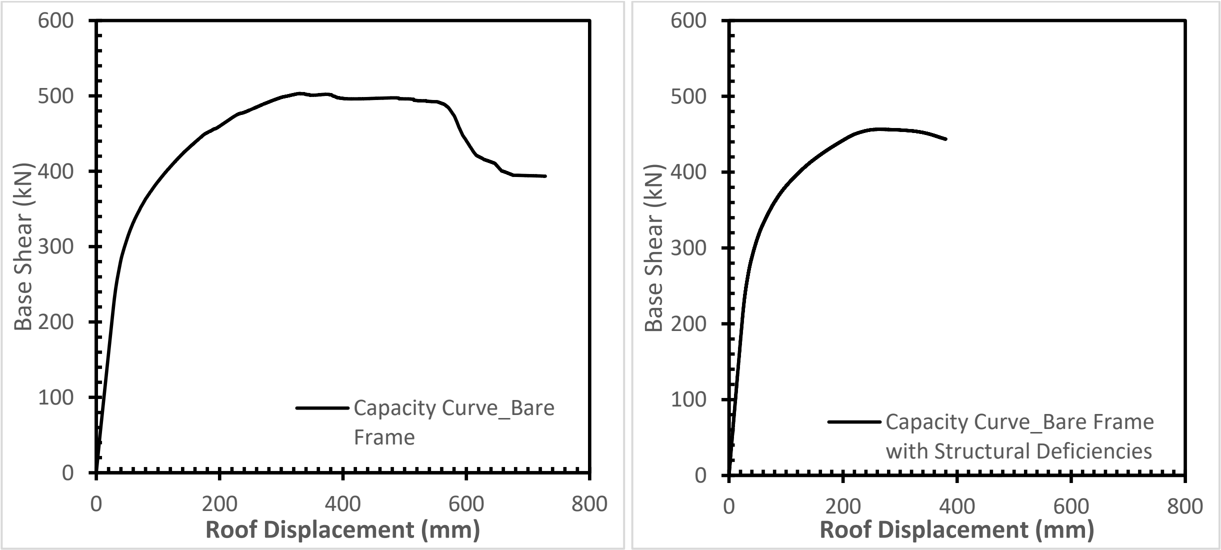
a) b)

Fig 7-3: a) Pushover curve of 3B6S MRFs- stone-concrete infilled frame. b) Pushover curve of 3B6S MRFs- stone-concrete infilled frame without ground infills



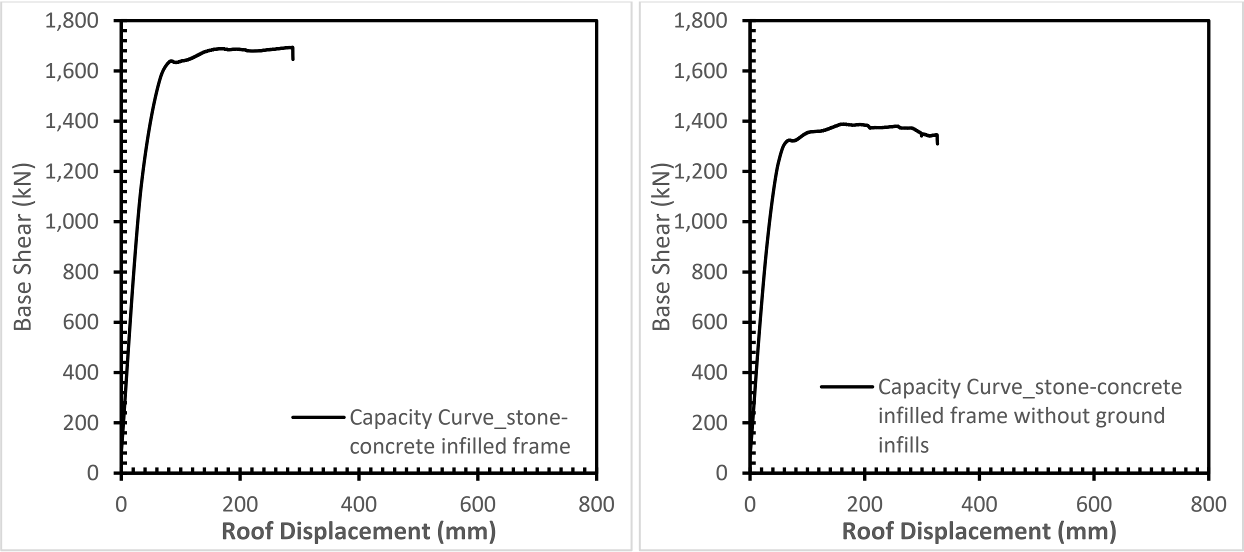
a) b)

Fig 7-4: a) Pushover curve of 3B6S MRFs- masonry-concrete infilled frame. b) Pushover curve of 3B6S MRFs- masonry-concrete infilled frame without ground infills



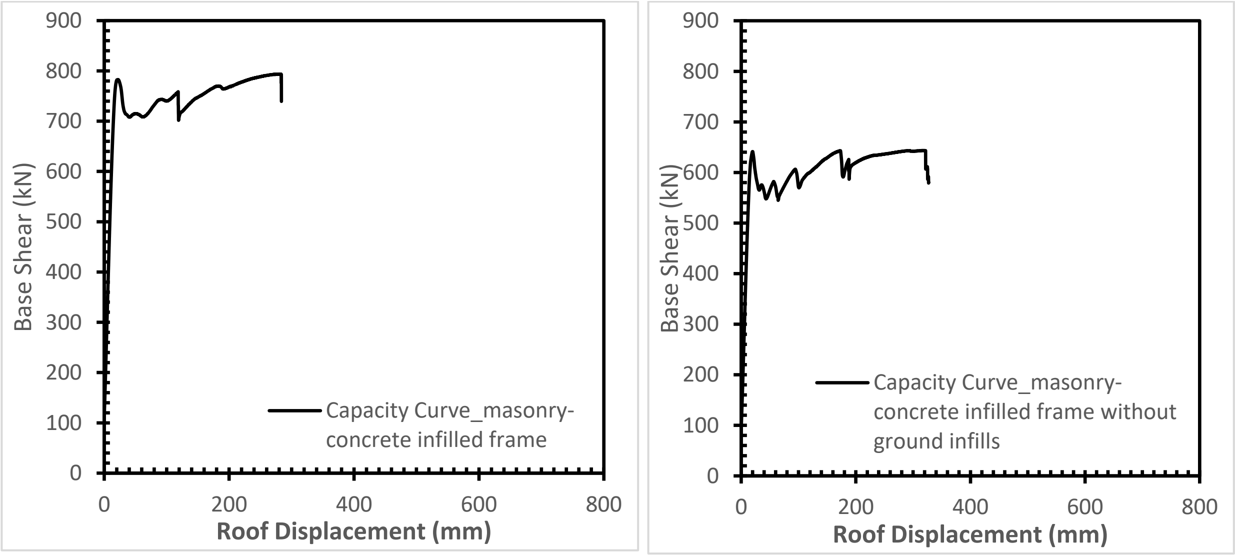
b) b)

Fig 7-5: a) Pushover curve of 3B9S MRFs-bare frame. b) Pushover curve of 3B9S MRFs-bare frame with structural deficiencies



b) b)

Fig 7-6: a) Pushover curve of 3B9S MRFs- stone-concrete infilled frame. b) Pushover curve of 3B9S MRFs- stone-concrete infilled frame without ground infills



b) b)

Fig 7-7: a) Pushover curve of 3B9S MRFs- masonry-concrete infilled frame. b) Pushover curve of 3B9S MRFs- masonry-concrete infilled frame without ground infills

7.2 Evaluation of R factor:

The R factor for each prototype model is computed based on the procedure explained previously in chapter three. The record of the calculation of R is presented in Table 7-2, while the full record of the calculation of R-factor can be found in Appendix 2. It was assumed that the redundancy factor (Rr) is 0.86 for all prototype models to present the critical case (three-line framing). Furthermore, the value of the calculated R factor for each frame model, the recommended R values from SEI/ASCE 7-16, Euro code (EC8), and the Egyptian code (ECP-201), and the used R-factor in the analysis are summarized in Table 7-3.

Table 7-2: The record of the calculation of the R-Factor.

Building System		Ductility Factor ($R\mu$)	Over Strength Factor (R0)	Redundancy Factor (Rr)
1)	3B6S MRFs-Bare Frame	**4.026**	**1.81**	**0.86**
2)	3B6S MRFs-Bare Frame with Structural Deficiencies	**2.47**	**1.6**	**0.86**
3)	3B6S MRFs-Stone-Concrete Infilled Frame	**1.87**	**………**	**………**
4)	3B6S MRFs-Stone-Concrete Infilled Frame without ground infills	**3.2**	**………**	**………**
5)	3B6S MRFs-Masonry-Concrete Infilled Frame	**3.26**	**………**	**………**
6)	3B6S MRFs-Masonry-Concrete Infilled Frame without ground infills	**4.17**	**………**	**………**
7)	3B9S MRFs-Bare Frame	**4.75**	**1.509**	**0.86**
8)	3B9S MRFs-Bare Frame with Structural Deficiencies	**2.98**	**1.37**	**0.86**
9)	3B9S MRFs-Stone-Concrete Infilled Frame	**2.93**	**………**	**………**
10)	3B9S MRFs-Stone-Concrete Infilled Frame without ground infills	**3.67**	**………**	**………**
11)	3B9S MRFs-Masonry-Concrete Infilled Frame	**5.2**	**………**	**………**
12)	3B9S MRFs-Masonry-Concrete Infilled Frame without ground infills	**5.44**	**………**	**………**

Table 7-3: Comparison between the calculated R and recommended R factors from the adopted seismic codes and standards, and the used R-factor in the analysis.

Building System		R Calculated	R (Used in the Analysis)	R (ASCE 7-16)	R (Europe, EC8)	R (Egyptian, ECP-201)
1)	3B6S MRFs-Bare Frame	**6.26**	**5**	**8**	**5**	**7**
2)	3B6S MRFs-Bare Frame with Structural Deficiencies	**3.38**				
3)	3B6S MRFs-Stone-Concrete Infilled Frame	**1.87**				
4)	3B6S MRFs-Stone-Concrete Infilled Frame without ground	**3.20**				
5)	3B6S MRFs-Masonry-Concrete Infilled Frame	**3.26**				
6)	3B6S MRFs-Masonry-Concrete Infilled Frame	**4.17**				
7)	3B9S MRFs-Bare Frame	**6.17**				
8)	3B9S MRFs-Bare Frame with Structural Deficiencies	**3.51**				
9)	3B9S MRFs-Stone-Concrete Infilled Frame	**2.93**				
10)	3B9S MRFs-Stone-Concrete Infilled Frame without ground	**3.67**				
11)	3B9S MRFs-Masonry-Concrete Infilled Frame	**5.20**				
12)	3B9S MRFs-Masonry-Concrete Infilled Frame	**5.44**				

Table 7-3 shows that the calculated R factors are highly affected by the type of structural frame system and the ductility of the building system. It was found that the R factors for the bare frames (high ductile frame system) 3B6S and 3B9S are around 6.2, which is less than the suggested R values from the SEI/ASCE 7-16 and the Egyptian code (ECP-201). Moreover, including structural deficiencies (non-ductile frame system) reduces the R factor from 6.2 to 3.45. This reduction is significant, and the value is far from the recommended values in the SEI/ASCE 7-16, Euro code (EC8), and the Egyptian code (ECP-201). Therefore, the R factors for the bare frame are unconservative by approximately 22.5%, 11.4% for the SEI/ASCE 7-16, and the Egyptian code (ECP-201), respectively. In the case of R factors for bare frames with structural deficiencies, the values are unsafe compared with the recommended values in the above-mentioned codes and standards used in the current practice in Palestine.

Furthermore, it was also noted that using the procedure for calculating the R factor based on the ATC-19 for a frame system with infill walls gives an unrepresentative and unrealistic R factor. As the R factor represents the design force reduction on the condition that the system meets the ductility demand ratio and dissipates the kinetic energy through its plastic-energy capacity without exceeding a life-safety performance objective. Therefore, defining the R factor is corrected with the ductility ratio. In the case of frame systems with infill walls, R values are relatively high. However, it does not indicate ductile behavior, and in most cases, a soft-storey mechanism is activated. To investigate this more, performance and damage assessment are discussed in the next chapter.

It is also worth noting that the R factor computed from the analysis may still need further reduction to take into account structural irregularity and discrepancies in construction. In order to be valid in Palestine.

CHAPTER 8: PERFORMANCE ASSESSMENT FOR STRUCTURAL BUILDINGS

8.1 Introduction to the Performance-Based Seismic Assessment:

The concept of performance-based seismic assessment has been initiated due to the major developments in structures' seismic analysis and design over the years. Traditionally, seismic design codes provide enforceable criteria that can achieve a minimum level of safety and acceptable performance of buildings during an earthquake by specifying a minimum requirement for strength, stiffness and ductility, determining the proper materials, and elements detailing and configuration. Based on this, a minimum level of performance is implied by the requirements of seismic design codes. However, the performance of building structures that are not designed or explicitly constructed according to the seismic design codes. Existing buildings with poor seismic behavior and design are often not identified by conventional standards and need to be assessed. Therefore, Performance-based seismic assessment was introduced for such assessment, and it is now extended for applications for existing and new buildings.

Performance-based seismic assessment is a consistent framework that has been developed particularly in the past 20 years. It accurately predicts building performance under earthquakes through quantitative tools that characterize seismic hazard, non-linear response of structures, elements behavior, damage, and expected losses. However, the assessment outcomes can be affected by the type of analysis, numerical modeling procedure, and how engineers understand seismic behavior (Kam et al., 2017). Moreover, the framework of seismic performance assessment of structures depends mainly on the available structural system capacity and seismic demand produced by earthquake load. Relating the structural capacity with the demand determines whether the performance meets the defined requirements.

Many methods were developed to evaluate the seismic performance of building structures. Some estimate the structural capacity using linear analysis procedures, while others utilize non-linear analysis procedures such as non-linear static pushover and non-linear dynamic analysis. In FEMA-356, and SEI/ASCE41-17, nonlinear analysis techniques are adopted, and definitions of multiple structural performance levels are presented, as shown in Fig 8-1. Furthermore, the acceptance criteria are defined for various structural elements. This chapter discusses the adopted performance assessment procedure and the acceptance criteria in the following sections.

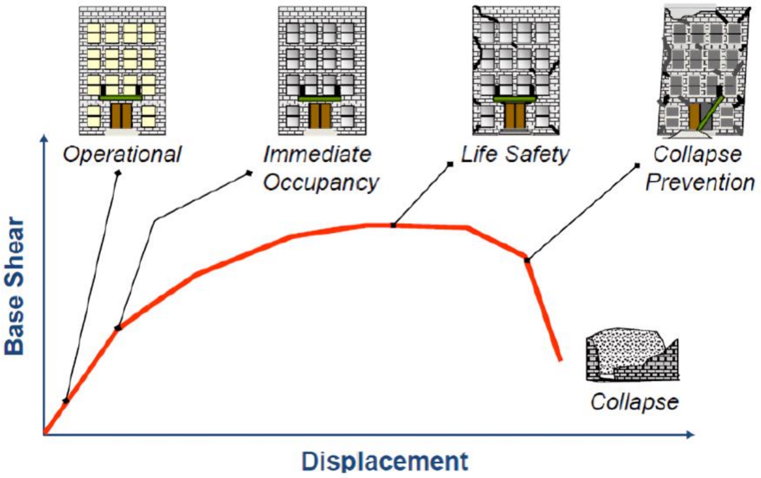


Fig 8-1: Various performance levels according to the SEI/ASCE41-17 (Kam et al., 2017).

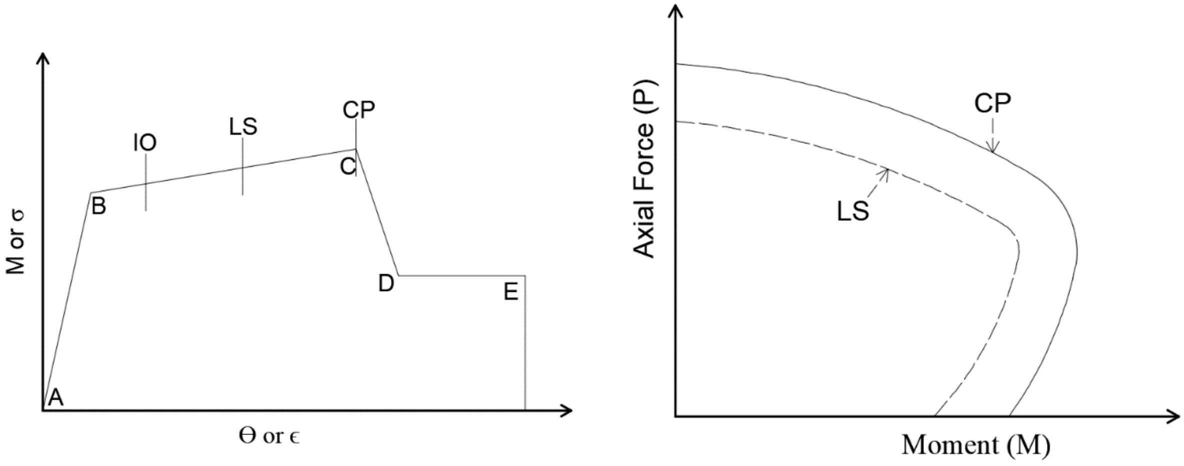
8.2 Performance Point and Element Performance Level:

As described in section 3.6, the performance point is evaluated for each prototype frame model under an earthquake event. According to the SEI/ASCE 41-17, in order to satisfy the life safety performance level, a seismic event with a probability of exceedance equal to 10%/50 years should be considered in the analysis. The target displacement is calculated based on a seismic event 10%/50 years probability of exceedance. The life safety objective limit is 0.75 times the maximum displacement in the pushover curve for each prototype model. Table 8-1 shows the target and limit displacement at the life safety performance level.

However, to determine the performance level for each structural element, force-deformation relationship is defined for each model element through the frame hinges assigned to reflect the nonlinear behavior of the model. In the case of the beam and beam-column joint elements, a moment-rotation relationship was chosen to identify the performance of beam and beam-column joint elements, as shown in Fig 8-2a. The x-axis is set to be the rotation (Ѳ), and the y-axis is the moment stress (M). The performance levels are located in segments B-C and D-E. In contrast, infill models are presented by axial stress-strain relationship, as shown in Fig 8-2a. The x-axis is set to be the strain (mm/mm), and the y-axis is the axial stress. The performance levels are located on segments B-C and D-E as well.

Furthermore, the design interaction diagram (P-M) was utilized to evaluate the seismic performance of columns at the hinges near the supports, as shown in Fig 8-2b. The x-axis is set to be the moment stress (M), and the y-axis is the axial force (P). It was also considered that the collapse prevention performance level is located on the P-M interaction curve, and the life safety performance level is set to be 0.75 times any value on the P-M interaction curve (SEI/ASCE41-17).

Looking at the element level, the element model's target performance (Θ, ϵ, or M) is depicted at the time step where the target displacement is achieved. The limits for different performance objectives for each force-deformation relationship are determined according to SEI/ASCE 41-17. The yielding point is at point B in Fig 8-2a, and intermediate occupancy (IO) occurs where the deformation equals 0.67 times the deformation limit for life safety (LS). Life safety (LS) is estimated where the deformation is at 0.75 times the deformation at point C. Lastly, collapse prevention (CP) is at 1.0 times the deformation at point C on the curve.



a)

b)

Fig 8-2: a) Force-deformation relationship for beam, beam-column joint, and infill model. b) Force-deformation relationship for column.

Table 8-1: Target displacement and life safety performance level limits.

	Building System	Target Displacement (mm)	Limit for LS performance level (mm)
1)	3B6S MRFs-Bare Frame	144.5	345.0
2)	3B6S MRFs-Bare Frame with Structural Deficiencies	144.8	207.8
3)	3B6S MRFs-Stone-Concrete Infilled Frame	90.5	59.2
4)	3B6S MRFs-Stone-Concrete Infilled Frame without ground infills	90.9	128.3
5)	3B6S MRFs-Masonry-Concrete Infilled Frame	91.4	87.3
6)	3B6S MRFs-Masonry-Concrete Infilled Frame without ground infills	91.8	137.5
7)	3B9S MRFs-Bare Frame	190.9	435.0
8)	3B9S MRFs-Bare Frame with Structural Deficiencies	190.9	285.0
9)	3B9S MRFs-Stone-Concrete Infilled Frame	126.6	216.6
10)	3B9S MRFs-Stone-Concrete Infilled Frame without ground infills	126.7	244.9
11)	3B9S MRFs-Masonry-Concrete Infilled Frame	127.2	212.5
12)	3B9S MRFs-Masonry-Concrete Infilled Frame without ground infills	127.5	240.6

It was noted from Table 8-1 that (3B6S, 3B9S) MRFs-bare frames and (3B6S, 3B9S) MRFs-bare frames with structural deficiencies pass the acceptance criteria of life safety performance level with a high margin between maximum displacement and target displacement. On the other hand, 3B6S MRFs-stone-concrete and 3B6S MRFs-masonry-concrete infilled frames did not satisfy the acceptance criteria of life safety performance level. This refers to the high rigidity and brittle behavior of infilled frames at ground level. The 3B6S and 3B9S infilled frames without ground infills fulfilled the acceptance criteria according to the SEI/ASCE 41-17. However, the observations during the analysis showed that the inter-story drift for the first floor, in general, was much larger than the inter-story drifts for the rest of the floors. This concludes that a soft-story mechanism might have occurred. Further investigation on this problem is discussed in section 8.3.

8.3 Performance and Damage-Based Seismic Assessment:

After completing the modeling stage, the analysis was performed using the non-linear static pushover method on each prototype frame model. Evaluation of performance points using the coefficient method (CM) was explained in section 3.6 and performed in section 8.2. In this section, each prototype model's performance and damage assessment was performed. The seismic event 10%/50-year probability of exceedance is considered since the majority of building structures are analyzed and designed to a 10% probability of exceedance seismic event. The damage was monitored at a time step where the target displacement was achieved.

1) 3B6S MRFs- Ductile and Non-Ductile Bare Frame

The analysis was performed on 3B6S MRFs- ductile and non-ductile (with structural deficiencies) bare frames, and the results are summarized in Fig 8-3 to Fig 8-4b. It can be concluded that the performance of both models at target displacement step is within the life safety objective; in other words, none of the hinges exceeded the life safety (LS) performance level, as shown in Fig 8-5a and Fig 8-5b. For the case of 3B6S MRFs- ductile bare frame, it can also be seen that the hinges in beams and beam-column joint hinges are distributed on all floors and reached intermediate occupancy IO performance level. Column hinges formed at lower levels and within the LS performance level. This indicates a ductile frame behavior.

In contrast, the 3B6S MRFs- non-ductile bare frame analysis showed that more hinges in beam and beam-column joint elements reached LS performance level. Column hinges formed at lower levels and within the LS performance level. Comparing it with the performance 3B6S MRFs- ductile bare frame, it can be noted that more damage occurred to the structural elements. The code was generated using OpenSees to perform a complete non-linear static pushover analysis for both 3B6S MRFs- ductile and non-ductile bare frame shown in Appendix 3 and 4.

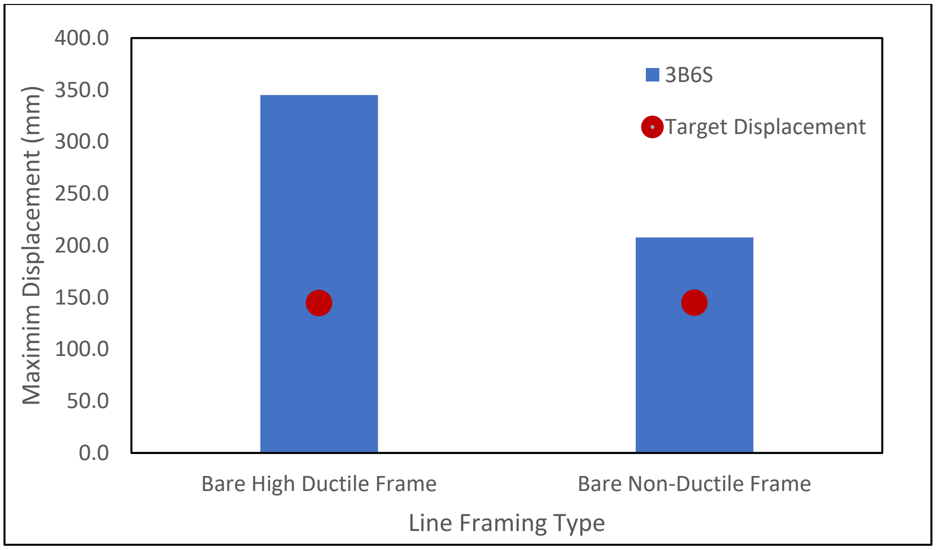
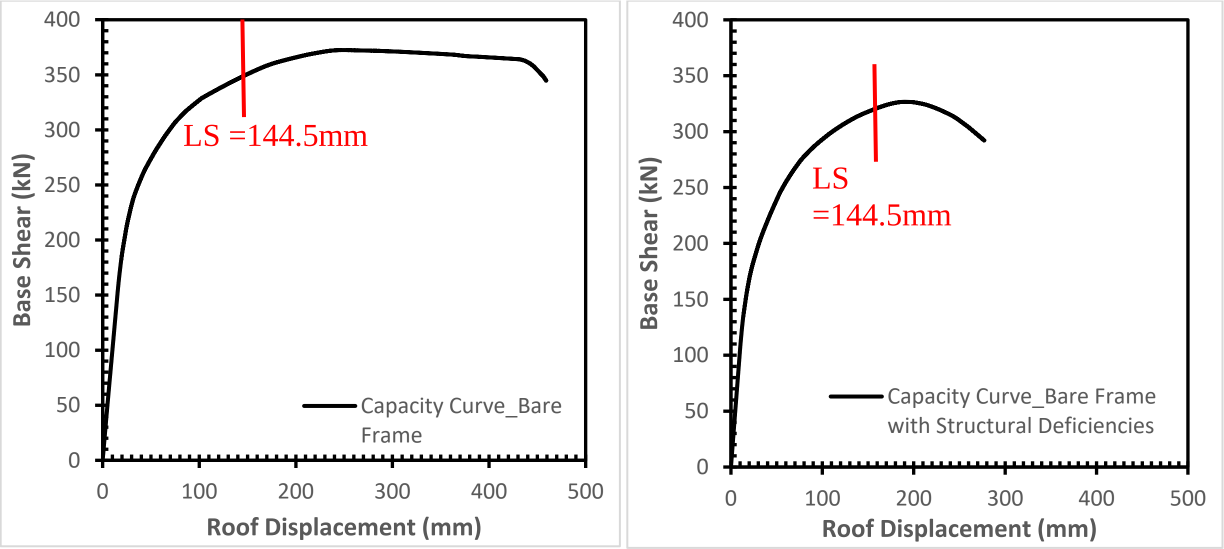
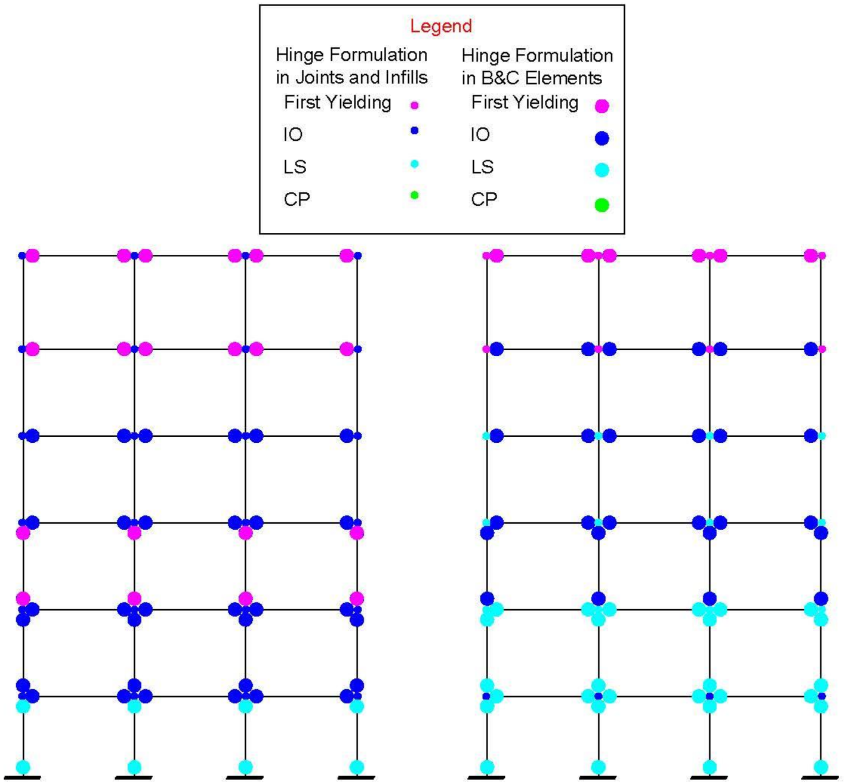


Fig 8-3: Determination of performance level for 3B6S MRFs- ductile and non-ductile bare frame under seismic event (10%/50 years)



a) b)

Fig 8-4: a) Performance point for 3B6S MRFs- ductile bare frame. b) Performance point for 3B6S MRFs- non-ductile bare frame.



a) b)

Fig 8-5: a) Hinge formation at the assigned elements in 3B6S MRFs-bare frame. b) Hinge formation at the assigned elements in 3B6S MRFs- non-ductile bare frame.

2) 3B6S MRFs-Ductile Bare Frame, Masonry-Concrete Infilled Frame, and Masonry-Concrete Infilled Frame without ground infills

In this sub-section, a comparison of performance and seismic behavior of 3B6S MRFs-ductile bare frame and masonry-concrete infilled frame with and without ground infills is made. The analysis results of performance level determination are summarized in Fig 8-6, and it can be seen that there is a high reduction in performance in both masonry-concrete infilled frames with and without ground infills compared with a ductile bare frame. In the case of the masonry-concrete infilled frame, the performance at the target displacement step is almost at the life safety objective, as shown in Fig 8-7a, while the masonry-concrete infilled frame without ground infills achieved the target displacement, as shown in Fig 8-7b.

The type of hinges created in each model element, beam, beam-column joint, and infill hinges in the masonry-concrete infilled frame are within the life safety (LS) performance level. Column hinges are concentrated on the first floor and reached intermediate occupancy (IO) performance level, as shown in Fig 8-8a. It also can be noted that all hinges are clustered on lower and middle floors, and the upper floors do not contribute to the frame structural system performance. The poor distribution of hinges indicates bad performance.

On the other hand, the analysis results of the masonry-concrete infilled frame without ground infills shown in Fig 8-8b show that beam and beam-column joint hinges are within the life safety (LS) performance level and concentrated on lower floors. Column hinges were developed on the first floor and failed before achieving the target displacement. The masonry-concrete infilled frame's overall behavior without ground infills satisfies the target displacement. However, looking closely at its performance, collapse prevention (CP) for first-floor columns occurred, which indicates a soft-story mechanism. The code was generated using OpenSees to perform full non-linear static pushover analysis for both 3B6S MRFs- masonry-concrete infilled frames with and without ground infills shown in Appendix 5 and 6, respectively.

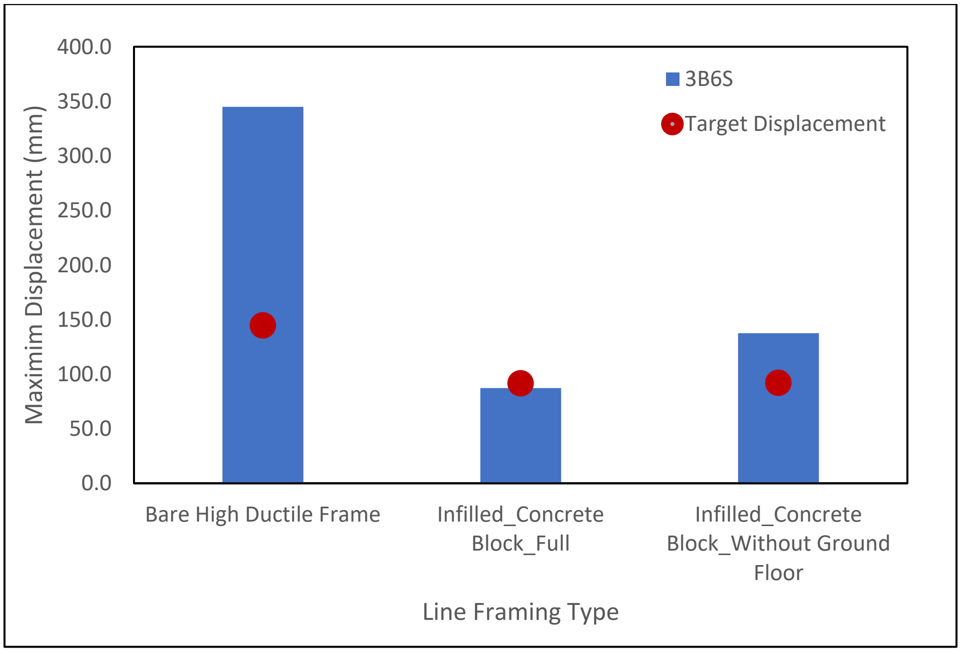
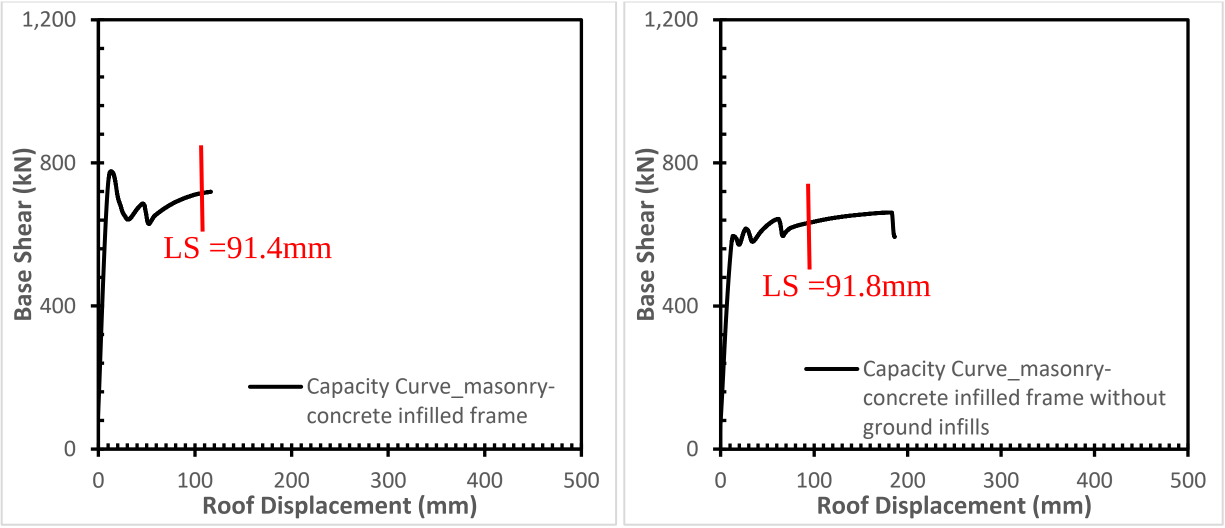
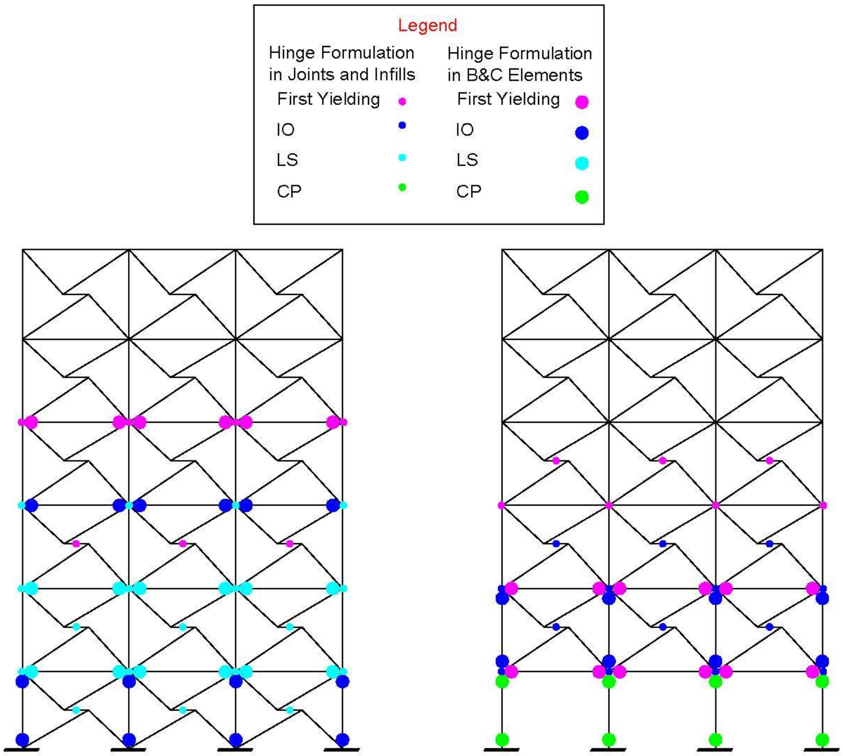


Fig 8-6: Determination of performance level for 3B6S MRFs-ductile bare frame and masonry-concrete infilled frame with and without ground infills under seismic event (10%/50 years).



a) b)

Fig 8-7: a) Performance point for 3B6S MRFs- masonry-concrete infilled frame. b) Performance point for 3B6S MRFs- masonry-concrete infilled frame without ground infills.



a) b)

Fig 8-8: a) Hinge formation at the assigned elements in 3B6S MRFs- masonry-concrete infilled frame. b) Hinge formation at the assigned elements in 3B6S MRFs- masonry-concrete infilled frame without ground infills.

3) 3B9S MRFs- Ductile and Non-Ductile Bare Frame

Similar behavior to 3B6S MRFs- ductile and non-ductile bare frame was observed during the analysis of 3B9S MRFs- ductile and non-ductile (with structural deficiencies) bare frame. Both models at the target displacement step are within the life safety objective, as summarized in Fig 8-9 to Fig 8-10b. Moreover, none of the hinges exceeded the life safety (LS) performance level, as shown in Fig 8-11a and Fig 8-11b. Beam and beam-column joint hinges in 3B9S MRFs- ductile bare frames are distributed on almost all floors and reached a maximum performance level of intermediate occupancy (IO). However, Column hinges are created at lower and middle levels and within the LS performance level. In comparison with the 3B9S MRFs- non-ductile bare frame, beams, columns, and beam-column joints formed hinges with LS performance levels. In addition, comparing both models with the performance of 3B6S MRFs- ductile and non-ductile bare frames, it can be concluded that frame systems with a higher elevation, columns tend to form plastic hinges at the upper levels, which gives an indication of lower performance. The code was generated using OpenSees to perform full non-linear static pushover analysis for both 3B9S MRFs- ductile and non-ductile bare frames shown in Appendix 7 and 8.

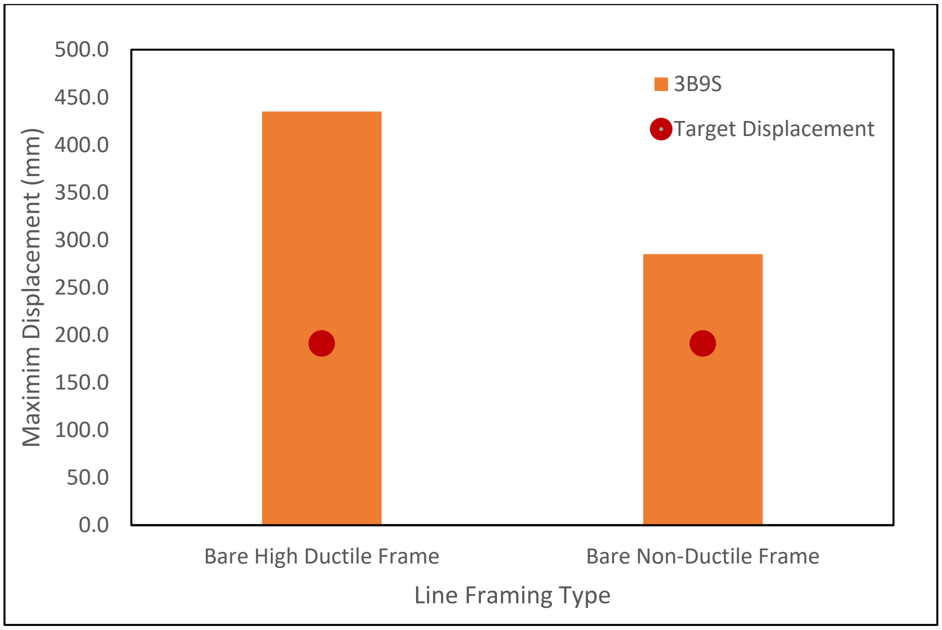
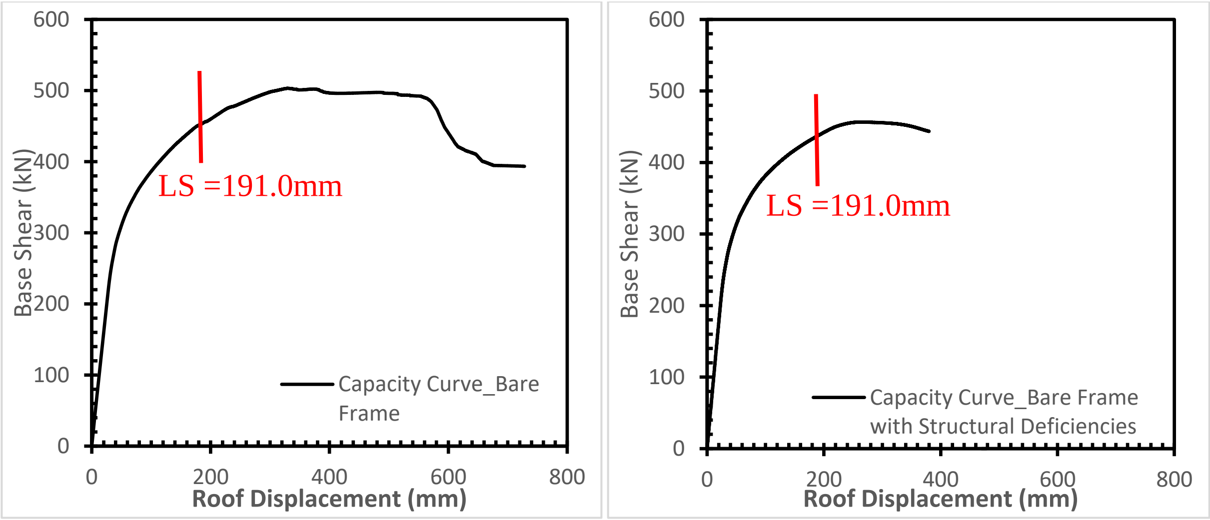
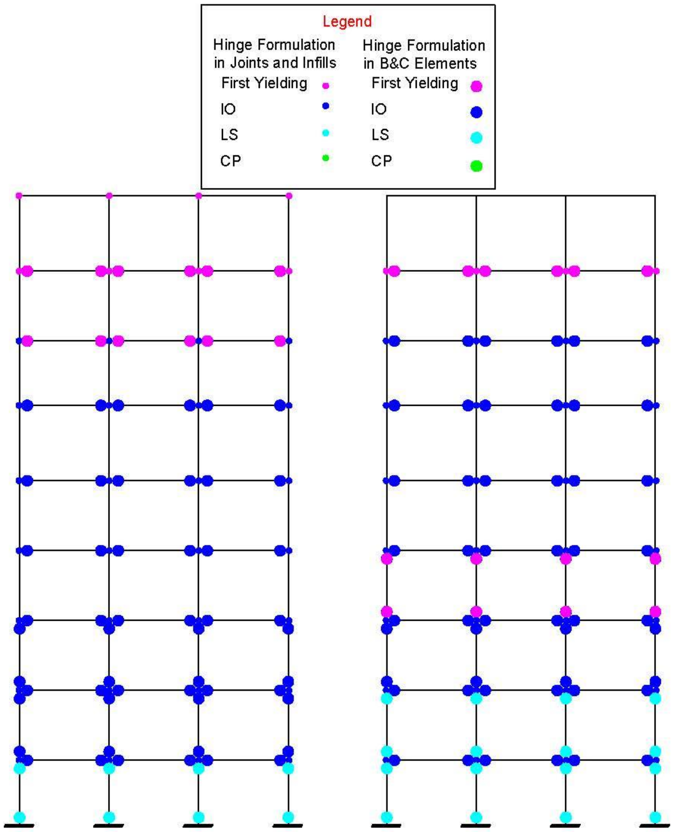


Fig 8-9: Determination of performance level for 3B9S MRFs- ductile and non-ductile bare frame under seismic event (10%/50 years)



b) b)

Fig 8-10: a) Performance point for 3B9S MRFs- ductile bare frame. b) Performance point for 3B9S MRFs- non-ductile bare frame.



b) b)

Fig 8-11: a) Hinge formation at the assigned elements in 3B9S MRFs-bare frame. b) Hinge formation at the assigned elements in 3B9S MRFs- non-ductile bare frame.

4) 3B9S MRFs-Ductile Bare Frame, Masonry-Concrete Infilled Frame, and Masonry-Concrete Infilled Frame without ground infills

Fig 8-12 summarizes the results of performance level analysis on 3B9S MRFs-masonry-concrete infilled frame with and without ground infills. It shows a considerable performance loss due to the existence of infills compared with a ductile bare frame. The performance of 3B9S MRFs-masonry-concrete infilled frame models at the target displacement step did not exceed the life safety objective, as shown in Fig 8-13a and Fig 8-13b. From Fig 8-14a, it can be noted that the 3B9S MRFs-masonry-concrete infilled frame formed plastic hinges in beams, columns, and infills with LS performance level. The beam-column joint hinges reached the yielding point. All model element hinges were concentrated at lower and middle floor levels. In contrast, 3B9S MRFs-masonry-concrete infilled frames without ground infills formed hinges with IO performance levels in beams and beam-column joint elements. The Column hinges at first story reached collapse prevention performance level exceeding the LS performance level. This indicates that the soft-story mechanism is activated. In the same way, a comparison with 3B6S MRFs-masonry-concrete infilled frame with and without ground infills show that as the elevation of the frame system increases, the effect of soft-story increases. The code was generated using OpenSees to perform full non-linear static pushover analysis for both 3B9S MRFs- masonry-concrete infilled frames with and without ground infills shown in Appendix 9 and 10, respectively.

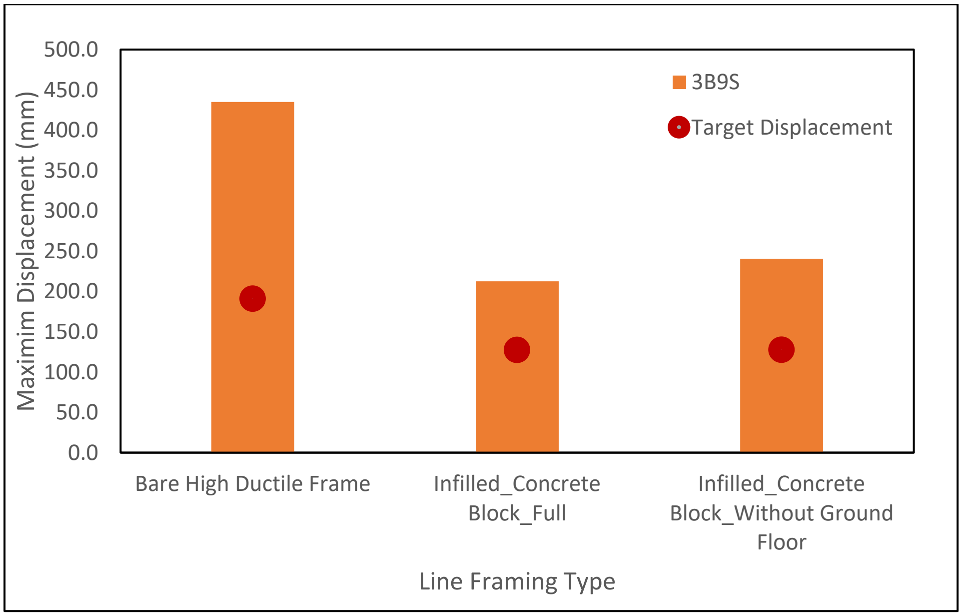
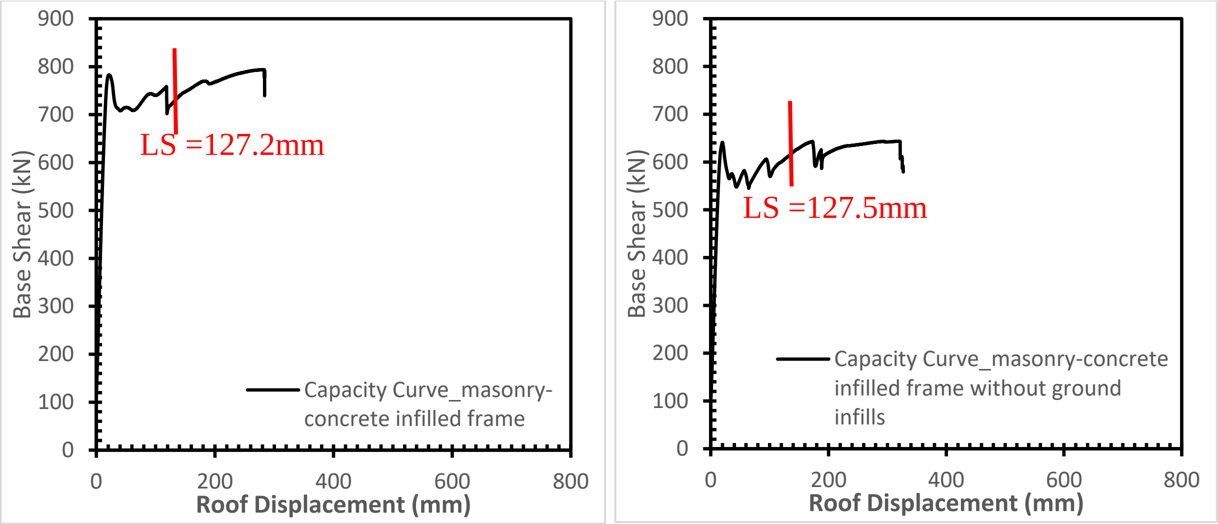
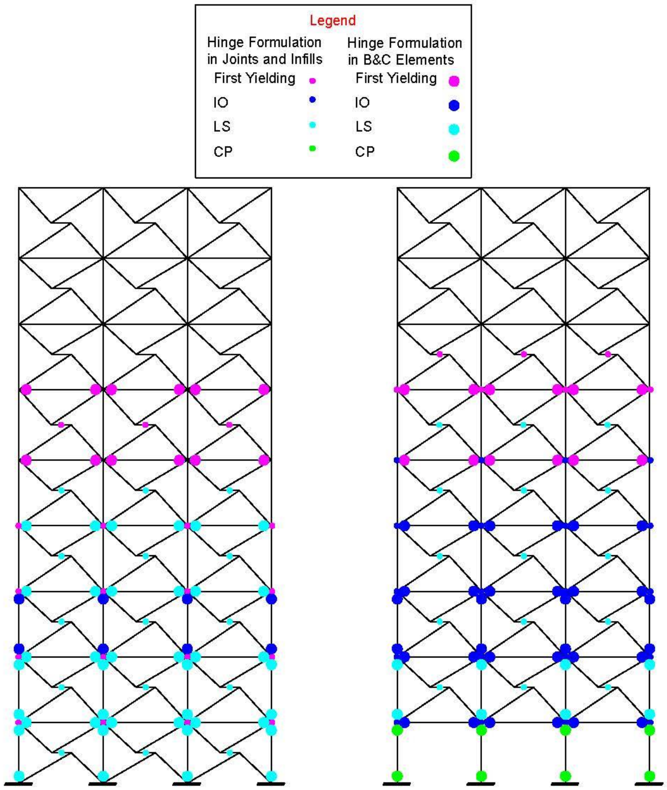


Fig 8-12: Determination of performance level for 3B9S MRFs-ductile bare frame and masonry-concrete infilled frame with and without ground infills under seismic event (10%/50 years).



b) b)

Fig 8-13: a) 3B9S MRFs- masonry-concrete infilled frame performance point. b) Performance point for 3B9S MRFs- masonry-concrete infilled frame without ground infills.



b) b)

Fig 8-14: a) Hinge formation at the assigned elements in 3B9S MRFs- masonry-concrete infilled frame. b) Hinge formation at the assigned elements in 3B9S MRFs- masonry-concrete infilled frame without ground infills.

5) 3B6S and 3B9S MRFs-Stone-Concrete Infilled Frame with/without ground infills

The performance and the behavior of 3B6S and 3B9S MRFs-stone-concrete infilled frames with/without ground infills have been observed to be the same as 3B6S, and 3B9S MRFs-masonry-concrete infilled frames with/without ground infills. Therefore, the data analysis regarding each prototype model is shown in Appendix 15. The code generated using the OpenSees program to perform complete non-linear static pushover analysis for 3B6S and 3B9S MRFs-stone-concrete infilled frames with/without ground infills are shown in Appendix 11,12,13 and 14, respectively.

Based on the data analysis performed in sections 8.2 and 8.3, it was found that the inclusion of structural deficiencies in the frame systems increased damage to the structural elements and negatively affected the distribution pattern of the hinges. Consequently, a change in the performance of buildings is anticipated. The existence of infills (stone-concrete or masonry-concrete) clusters the hinge formation on the lower and middle floors of the frame system, leaving the upper floors without any contribution to the performance of the structural frame system. Therefore, bad performance is expected due to the poor distribution of hinges in structural elements. It was also noted that the increase in the building's height increases the rate of the soft-story mechanism formation by the early development of plastic hinges in column elements.

The above analysis of performance assessment supports the need to review the R factors, which describe the seismic behavior of buildings in Palestine. This response modification factor should realistically describe the plastic hinge formation pattern, their ultimate capacity, and expected plastic deformation. Therefore, the use of R factors from the reviewed standards and even calculating them using conventional methods should be reviewed, and consequently, R factors values are reduced.

SUMMARY, CONCLUSIONS, LIMITATIONS AND RECOMMENDATIONS

1. Summary and Conclusions

This research aims to investigate analytically the response modification factor (R factor) of reinforced concrete moment resisting frames (MRFs) in Palestine and compare it with the R factor values recommended by the seismic design codes and standards used in the area. The starting point was to determine a representative model for the analysis, which was done through semi-structured interviews held with highly skilled/active engineers, contractors, and construction managers. Consequently, Numerical models of the examined structural systems were developed using the OpenSees to investigate the performance of the buildings in Palestine. These models were verified using the data available on the non-linear analysis of the RC-frame (Vecchio and Emara, 1992). It was found that the model built in Opensees can accurately show the non-linear behavior of a frame system. Therefore, twelve 2D RC-moment resisting frames (MRFs) are taken as typical RC frames in Palestine. Each frame system is analyzed using a non-linear static pushover analysis to evaluate the response modification factor. Response modification factor R was further examined by performing performance and damage assessment for specific performance objectives. Lastly, a comparison between the calculated response modification factor and the values from the seismic codes of practice is performed.

Based on the modeling and analysis results, the following are the major research findings:

1. It was found that the R factor in the seismic codes and standards adopted in Palestine gives an inaccurate representation of the building structure's response during a seismic event.
2. The R factor recommended in the SEI/ASCE 7-16, Euro code (EC8), and the Egyptian code (ECP-201) is unconservative to be used in Palestine under the current construction technology and thus should be decreased. The current status of calculating the base shear is assuming R-value for the building as a skeleton. However, when considering the infills, the performance and the behavior are affected, and the R factor as well.
3. It was also noted that the inclusion of structural deficiencies (insufficient beam and column shear reinforcement, short anchorage length in longitudinal beam bars, and inadequate beam-column joint shear reinforcement) causes the poor distribution of plastic hinges in

the structural elements. Consequently, a significant loss in ductility of the frame system and a reduction in the R factor was observed.

4. The building's height is an important parameter that affects the performance of structures and highlights the impact of the soft-story mechanism.
5. Using the stone-concrete or masonry-concrete infills increases the strength and stiffness of the frame system. However, it dramatically decreases ductility. Consequently, the R factor is reduced, and severe damage may appear in some elements because of the considerable loss in ductility.
6. Using stone-concrete or masonry-concrete infills in frame systems without ground infills create a soft story mechanism that may cause serious problems such as forming a weak story and the possibility of stone failing. All that could be life-threatening, especially in case of stone or block failing.

2. Limitations

Several restrictions have been faced in this research, and it is summarized in the following points:

a) The unavailability of experimental data to validate the modeling criteria of structural deficiencies and their effect on the seismic performance of frame structures may affect the analysis results.

b) The limitations of using pushover analysis are mainly related to the selection of the horizontal load pattern and target displacement at the roof mass center. The load pattern is assumed to be an invariant lateral load pattern, and the system has a constant distribution of inertial forces. In contrast, the top roof displacement represents a single degree of freedom system (SDOFs) target displacement for a muti-degree of freedom system (MDOFs), and that could be a not good indicator of the behavior and performance of the overall structure, especially if the structure is dominated by more than one mode shape.

c) The assumption of using the same value of redundancy factor for all prototype models (bare frame and infilled frame) may influence the values of the response modification factor.

d) During non-linear static pushover analysis, the first failure occurrence for the system was not identified, and the failure of the system mainly depended on the degradation of the pushover curve.

e) The in-plane analysis of infill walls was adopted in this research, and the out-of-plane was not considered, which may count as a limitation.

3. Recommendations

Based on the analysis results and the conclusions mentioned above, the following recommendations are given:

- Near future solution:

1. Solve structural deficiencies in the current construction practice by focusing on structural detailing along with seismic analysis and design. Supervision on construction sites is mandatory to ensure that defects such as inadequate joint shear reinforcement, insufficient development length of beam bars, and inadequate stirrup spacing in beams and columns are repealed.
2. The full separation between infills (brittle behavior) and the frame system should be done to let the structure deform without any stresses on the infills. This could be applied using anchored joints and providing a seismic gap between infills and the frame system.
3. In the case of building structures under the current construction technology, a value of response modification factor R in a range of [2-3] should be taken to ensure that minimum performance of structures occurs.

- Far future solution:

1. Increasing the relative ductility of infill walls by implementing ductile materials in the infill walls, such as reinforced steel. Further investigation should be done to ensure that the system achieves the acceptance criteria for life safety performance level.

2. Alternatives for the cladding of reinforced concrete buildings in Palestine should be developed. Stone or masonry infills should be replaced with less weight and lower rigidity, such as foam stone or glass panels.

Finally, it is crucial to note that the R factor computed from the analysis may still need further reduction to take into account structural irregularity and discrepancies in construction practice, such as the value of compressive strength of concrete, in order to have a more precise prediction for the R factor.

REFERENES:

Abdelwahed, B. "Beam-column joints reinforcement detailing adequacy in case of a corner column loss-numerical analysis." Latin American Journal of Solids and Structures 16 (2019).

Abdi, Heshmatollah, Farzad Hejazi, Mohd Saleh Jaafar, and Izian Binti Abd Karim. "Response modification factors for reinforced concrete structures equipped with viscous damper devices." Periodica Polytechnica Civil Engineering 62, no. 1 (2018): 11-25.

Abdi, Heshmatollah, Farzad Hejazi, and Mohd S. Jaafar. "Response modification factor-Review paper." In *IOP Conference Series: Earth and Environmental Science*, vol. 357, no. 1, p. 012003. IOP Publishing, 2019.

Abou Karaki. N. "Synthese et Carte Seismotectonique des pays de la Borrdure Orientate de la Mediterranee: Sismicite du System de Failles du Jourdain-MerMorte.". Ph.D. Thesis. in French. Institute de Physique du Globe (IPGS). Univ. Strasbourg. France. (1987).p.417.

ACI Committee 318. 2019. Building code requirements for structural concrete (ACI 318-19): an ACI standard; Commentary on building code requirements for structural concrete (ACI 318R-19).

Ahmad, Naveed, Muhammad Rizwan, Muhammad Ashraf, Akhtar Naeem Khan, and Qaisar Ali. "Seismic collapse safety of reinforced concrete moment resisting frames with/without beam-column joint detailing." Bulletin of the New Zealand Society for Earthquake Engineering 54, no. 1 (2021): 1-20.

Akbar, Junaid, Naveed Ahmad, Muhammad Rizwan, Sairash Javed, and Bashir Alam. "Response Modification Factor of RC Frames Strengthened with RC Haunches." Shock and Vibration 2020 (2020).

Al-Nimry, Hanan, Musa Resheidat, and Saddam Qeran. "Rapid assessment for seismic vulnerability of low and medium rise infilled RC frame buildings." Earthquake Engineering and Engineering Vibration 14, no. 2 (2015): 275-293.

Al-Nimry, H. "Evaluation of Seismic Performance of Stone-Concrete Residential Buildings in Jordan." Final Report (2010).

Al-Nimry, H. "Seismic Vulnerability of Residential Buildings in Jordan and its Locality." In Proceedings of the 15th World Conference on Earthquake Engineering. 2012.

Al-Nimry, Hanan S. "Quasi-static testing of RC infilled frames and confined stone-concrete bearing walls." Journal of Earthquake Engineering 18, no. 1 (2014): 1-23.

Al-Tarazi, Eid A. "Regional seismic hazard study for the eastern Mediterranean (Trans-Jordan, Levant and Antakia) and Sinai region." Journal of African Earth Sciences 28, no. 3 (1999): 743-750.

Ambraseys, Nicholas Nicholas, Charles Peter Melville, and Robin Dartrey Adams. The seismicity of Egypt, Arabia and the Red Sea: a historical review. Cambridge university press, 2005.

Anderson, Meredith, Dawn Lehman, and John Stanton. "A cyclic shear stress–strain model for joints without transverse reinforcement." Engineering Structures 30, no. 4 (2008): 941-954.

Applied Technology Council, and Structural Engineers Association of California. *Tentative Provisions for the Development of Seismic Regulations for Buildings: A Cooperative Effort with the Design Professions, Building Code Interests, and the Research Community*. Vol. 3, no. 6. Department of Commerce, National Bureau of Standards, 1978.

Applied Technology Council, and Structural Engineers Association of California*:* A critical review of current approaches to earthquake resistant design 1995.

Applied Technology Council, Partnership for Response, and United States. Federal Emergency Management Agency. Evaluation of Earthquake Damaged Concrete and Masonry Wall Buildings: Basic Procedures Manual. The Agency, 1999.
ATC. (1995), A critical review of current approaches to earthquakeresistant design. ATC-34, “Applied Technology Council”, Redwood City, California, 1995:31–6.

ATC. Structural response modification factors. ATC-19, Applied Technology Council, Redwood City, California, pp. 5–32 (1995).

ASCE SEI/ASCE 7-16. Minimum design loads for buildings and other structures. Reston (USA): American Society of Civil Engineers (2016).

Structural Engineering Institute. "ASCE Standard, ASCE/SEI, 41-17: Seismic Evaluation and Retrofit of Existing Buildings." American Society of Civil Engineers, 2017.

Badrashi. "Response Modification Factor for Reinforced Concrete Buildings in Pakistan." PhD. diss., University of Engineering and Technology, (2016).

Boore, David M., William B. Joyner, and Thomas E. Fumal. "Equations for estimating horizontal response spectra and peak acceleration from western North American earthquakes: A summary of recent work." Seismological research letters 68, no. 1 (1997): 128-153.

Borzi, B., and A. S. Elnashai. "Refined force reduction factors for seismic design." Engineering structures 22, no. 10 (2000): 1244-1260.

Brahmavrathan, Divya, and C. Arunkumar. "Evaluation of response reduction factor of irregular reinforced concrete framed structures." Indian J Sci Technol 9, no. 23 (2016): 1-8.

British Standards Institution. 1996. Eurocode 8: design provisions for earthquake resistance of structures. London: British Standards Institution.

Calvi, Gian Michele, Guido Magenes, and Stefano Pampanin. "Relevance of beam-column joint damage and collapse in RC frame assessment." Journal of Earthquake Engineering 6, no. spec01 (2002): 75-100.

Cavaleri, Liborio, Marinella Fossetti, and Maurizio Papia. "Effect of vertical loads on lateral response of infilled frames." In Proceedings 13th World Conference on Earthquake Engineering. 2004.

Chaulagain, Hemchandra, Hugo Rodrigues, Enrico Spacone, Ramesh Guragain, Radhakrishna Mallik, and Humberto Varum. "Response reduction factor of irregular RC buildings in Kathmandu valley." Earthquake Engineering and Engineering Vibration 13, no. 3 (2014): 455-470.

Code, Uniform Building. "Uniform building code." In *International Conference of Building Officials, Whittier, CA*. 1997.

Dabbeek, Jalal AL. "An assessment on disaster risk reduction in the occupied Palestinian territory." (2010).

D’Amato, Michele, Franco Braga, Rosario Gigliotti, Sashi Kunnath, and Michelangelo Laterza. "Validation of a modified steel bar model incorporating bond-slip for seismic assessment of concrete structures." Journal of Structural Engineering 138, no. 11 (2012): 1351-1360.

Daza, L. G. "Correlation between minimum building strength and the response modification factor." In *Challenges, Opportunities and Solutions in Structural Engineering and Construction*, pp. 319-324. CRC Press, 2009.
Mitchell, Denis, and Patrick Paultre. "Ductility and overstrength in seismic design of reinforced concrete structures." *Canadian Journal of Civil Engineering* 21, no. 6 (1994): 1049-1060.

Dolšek, Matjaž, and Peter Fajfar. "Mathematical modelling of an infilled RC frame structure based on the results of pseudo-dynamic tests." Earthquake engineering & structural dynamics 31, no. 6 (2002): 1215-1230.

El-Dakhakhni, Wael W., Mohamed Elgaaly, and Ahmad A. Hamid. "Three-strut model for concrete masonry-infilled steel frames." Journal of Structural Engineering 129, no. 2 (2003): 177-185.

Elnashai, A. S., and A. M. Mwafy. "Calibration of Force reduction factors for RC Building." Journal of Earthquake Engineering 6, no. 2 (2002): 239-273.

ECP-201, Egyptian Code for Calculating Loads and Forces in Structural Work and Masonry, National Research Center for Housing and Building, Giza, Egypt, 2011.

Eshghi, S., and V. Zanjanizadeh. "Cyclic behavior of slender R/C columns with insufficient lap splice length." In 14th World Conference on Earthquake Engineering, pp. 12-17. 2008.

Fajfar, P., M. Dolsek, R. Zarnic, and S. Gostic. "Development of numerical methodologies for infilled frames, towards European integration in seismic design and upgrading of building structures project." Euroquake-Project, Final Report (2001).

Federal Emergency Management Agency (FEMA). "NEHRP Guidelines for the Seismic Rehabilitation of Buildings" (FEMA 273), Washington, DC, October (1997).

FEMA 356, Prestandard and Commentary for the Seismic Rehabilitation of Buildings, FEMA Publication no. 356, The American Society of Civil Engineers for the Federal Emergency Management Agency,Washington, DC, USA, 2000.

Filippou, Filip C., Egor Paul Popov, and Vitelmo Victorio Bertero. "Effects of bond deterioration on hysteretic behavior of reinforced concrete joints." (1983): 137-147.

Freund, Raphael, Israel Zak, and Z. W. I. Garfunkel. "Age and rate of the sinistral movement along the Dead Sea Rift." Nature 220, no. 5164 (1968): 253-255.

Garfunkel, Zvi, Zvi Ben-Avraham, and Elisa Kagan, eds. Dead Sea transform fault system: reviews. Vol. 6. Springer, 2014.

Goel, Rakesh K. "Variability and accuracy of target displacement from nonlinear static procedures." International Scholarly Research Notices 2011 (2011).

Goksu, C., H. Yilmaz, S. R. Chowdhury, K. Orakcal, and A. Ilki. "The effect of lap splice length on the cyclic lateral load behavior of RC members with low-strength concrete and plain bars." Advances in Structural Engineering 17, no. 5 (2014): 639-658.

Grigoratos, Iason, Jamal Dabeek, Marta Faravelli, Antonella Di Meo, Vania Cerchiello, Barbara Borzi, Ricardo Monteiro, and Paola Ceresa. "Development of a fragility and exposure model for Palestine–application to the city of Nablus." Procedia engineering 161 (2016): 2023-2029.

Hanson, Robert D., Ian D. Aiken, Douglas K. Nims, Ph J. Richter, and R. E. Bachman. "State-of-the-art and state-of-the-practice in seismic energy dissipation." In *Proceedings of the Seminar on Seismic Isolation, Passive Energy Dissipation, and Active Control, Applied Technology Council*, vol. 2, pp. 449-471. 1993.

Holmes, Malcolm. "Steel frames with brickwork and concrete infilling." proceedings of the Institution of civil Engineers 19, no. 4 (1961): 473-478.

Hussein, Manar M., Manar Gamal, and Walid A. Attia. "Seismic response modification factor for RC-frames with non-uniform dimensions." Cogent Engineering 8, no. 1 (2021): 1923363.

IBC. "International Building Code", 2000 Edition, International Code Council, Falls Church,VA, (2000).

IL Standard SI 413. June 1995. ,Amendment No.5, September 2009

Izadinia, Mohssen, Mohammad Ali Rahgozar, and Omid Mohammadrezaei. "Response modification factor for steel moment-resisting frames by different pushover analysis methods." *Journal of Constructional Steel Research* 79 (2012): 83-90.

Jeon, Jong-Su, Laura N. Lowes, Reginald DesRoches, and Ioannis Brilakis. "Fragility curves for non-ductile reinforced concrete frames that exhibit different component response mechanisms." Engineering Structures 85 (2015): 127-143.

Kappos, A. J. "Seismic damage indices for RC buildings: evaluation of concepts and procedures." Progress in structural Engineering and Materials 1, no. 1 (1997): 78-87.

Kam, W. Y., and R. Jury. "Performance-Based Seismic Assessment: Simplified Methods and Collapse Indicators." In 16th World Conference on Earthquake Engineerinf (WCEE), Santiago, Chile. 2017.

Kim, Jaehong, James M. LaFave, and J. Song. "Joint shear behaviour of reinforced concrete beam–column connections." Magazine of concrete research 61, no. 2 (2009): 119-132.

Kim, Jinkoo, and Hyunhoon Choi. "Response modification factors of chevron-braced frames." Engineering structures 27, no. 2 (2005): 285-300.

Ko, Hyun, Yong-Koo Park, and Dong-Guen Lee. "Evaluation of Seismic Behavior for Low-Rise RC Moment Resisting Frame with Masonry Infill Walls." In 14th World Conference on Earthquake Engineering, Beijing, China. 2008.

Krawinkler, Helmut, and G. D. P. K. Seneviratna. "Pros and cons of a pushover analysis of seismic performance evaluation." Engineering structures 20, no. 4-6 (1998): 452-464.

Lin, Wesley Wei-Chih. "Modelling effects of insufficient lap splices on a deficient reinforced concrete frame." Master's thesis, Middle East Technical University, 2013.

Louzai, Amar, and Ahmed Abed. "Evaluation of the seismic behavior factor of reinforced concrete frame structures based on comparative analysis between non-linear static pushover and incremental dynamic analyses." Bulletin of Earthquake Engineering 13, no. 6 (2015): 1773-1793.

Lowes, Laura N., and Arash Altoontash. "Modeling reinforced-concrete beam-column joints subjected to cyclic loading." Journal of Structural Engineering 129, no. 12 (2003): 1686-1697.

Mainstone RJ. On the stiffness and strength of infilled frames. Proceeding of Institution of Civil Engineers (ICE), Supplement (IV), Paper No. 7360, 1971; 57–90.

Ma'moun. "Evaluation of Response Modification Factor of the Reinforced Concrete Structures with Shear Walls having different sizes of Openings against the Lateral Loading." M.Sc. diss., Near East University, (2020).

Mander, John B., Michael JN Priestley, and R. Park. "Theoretical stress-strain model for confined concrete." Journal of structural engineering 114, no. 8 (1988): 1804-1826.

Mazzoni, Silvia, Frank McKenna, Michael H. Scott, and Gregory L. Fenves. "OpenSees command language manual." *Pacific Earthquake Engineering Research (PEER) Center* 264 (2006): 137-158.

Mergos, Panagiotis E., and Andreas J. Kappos. "A gradual spread inelasticity model for R/C beam-columns, accounting for flexure, shear and anchorage slip." Engineering Structures 44 (2012): 94-106.

Mitra, Nilanjan, and Laura N. Lowes. "Evaluation, calibration, and verification of a reinforced concrete beam-column joint model." Journal of Structural Engineering 133, no. 1 (2007): 105-120.

Mondal, Apurba, Siddhartha Ghosh, and G. R. Reddy. "Performance-based evaluation of the response reduction factor for ductile RC frames." Engineering structures 56 (2013): 1808-1819.

Nasirzadeh, Farnad, Hamed Mazandaranizadeh, and Mehdi Rouhparvar. "Quantitative risk allocation in construction projects using cooperative-bargaining game theory." International Journal of Civil Engineering 14, no. 3 (2016): 161-170.

NEHRP. "Recommended Provisions for Seismic Regulations for New Buildings and OtherStructures". 2000 Edition, Part 2: Commentary, Building Seismic Safety Council, Washington, DC (2000).
Miranda, Eduardo, and Vitelmo V. Bertero. "Evaluation of strength reduction factors for earthquake-resistant design." *Earthquake spectra* 10, no. 2 (1994): 357-379.

Newmark, Nathan M., and William J. Hall. "Earthquake spectra and design." *Engineering monographs on earthquake criteria* (1982).

Ning, Chao-Lie, Bo Yu, and Bing Li. "Beam-column joint model for non-linear analysis of non-seismically detailed reinforced concrete frame." Journal of Earthquake Engineering 20, no. 3 (2016): 476-502.

Pahlavan, Hossein, Mohsenali Shaianfar, Gholamreza Ghodrati Amiri, and Milad Pahlavan. "Probabilistic seismic vulnerability assessment of the structural deficiencies in Iranian in-filled RC frame structures." *Journal of Vibroengineering* 17, no. 5 (2015): 2444-2454.

Polyakov, S. V. "On the interaction between masonry filler walls and enclosing frame when loaded in the plane of the wall." Translations in earthquake engineering 2, no. 3 (1960): 36-42.

Pampanin, Stefano, Gian Michele Calvi, and M. Moratti. "Seismic behavior of RC beam-column joints designed for gravity only." (2002).

Pandit, Tekkan, and Hemchandra Chaulagain. "Evaluation of Response Reduction Factor of Existing Masonry Infilled RC-Buildings in Pokhara." Himalayan Journal of Applied Science and Engineering 1, no. 1 (2020): 41-51.

Park, R. "A summary of results of simulated seismic load tests on reinforced concrete beam-column joints, beams and columns with substandard reinforcing details." Journal of Earthquake Engineering 6, no. 02 (2002): 147-174.

Park, Sangjoon, and Khalid M. Mosalam. "Simulation of reinforced concrete frames with non-ductile beam-column joints." Earthquake Spectra 29, no. 1 (2013): 233-257.

Patel, Nirav, and Prutha Vyas. "Evaluation Of Response Modification Factor For Moment Resisting Frames." Kalpa Publications in Civil Engineering 1 (2017): 118-123.

Patel, Nirav, and Sandip A. Vasanwala. "Evaluation of Response Reduction Factor for Un-reinforced Masonry-Infilled RC Buildings." In Innovations in Infrastructure, pp. 525-535. Springer, Singapore, 2019.

Rajeev, Anupoju, Sai Sharath Parsi, Sudharshan N. Raman, Tuan Ngo, and Amit Shelke. "Experimental and numerical investigation of an exterior reinforced concrete beam-column joint subjected to shock loading." International Journal of Impact Engineering 137 (2020): 103473.

Ramanjaneyulu, K., Balthasar Novák, Saptarshi Sasmal, Constanze Roehm, N. Lakshmanan, and Nagesh R. Iyer. "Seismic performance evaluation of exterior beam-column sub-assemblages designed according to different codal recommendations." Structure and Infrastructure Engineering 9, no. 8 (2013): 817-833.

Rodrigues, H. "Desenvolvimento e calibraçao de modelos numéricos para a análise sısmica de edifıcios." Universidade do Porto (2005).

Saneinejad, Abolghasem, and Brian Hobbs. "Inelastic design of infilled frames." Journal of Structural Engineering 121, no. 4 (1995): 634-650.

Schmidt, T. "An approach of modelling masonry infilled frames by the FE method and a modified equivalent strut method." Annual Journal on Concrete and Concrete Structures." Darmstadt, Germany: Darmstadt University (1989).

Seifi, M., J. Noorzaei, M. S. Jaafar, and E. Panah. "Nonlinear static pushover analysis in earthquake engineering: State of development." In Proceeding of International Conference on Construction Building Technology, Kuala Lumpur. 2008.

“Seismic Hazard Map.” An-Najah National University: Seismic Hazard Map. Accessed June 17, 2022. https://www.najah.edu/en/community/scientific-centers/urban-planning-and-disaster-risk-reduction-center/earth-sciences-and-seismic-engineering-unit/seismic-hazard-map/.

Shafaei, Jalil, Mohammad Sajjad Zareian, Abdollah Hosseini, and Mohammd Sadegh Marefat. "Effects of joint flexibility on lateral response of reinforced concrete frames." Engineering Structures 81 (2014): 412-431.

Shapira, Avi. "Magnitude scales for regional earthquakes monitored in Israel." Israel Journal of Earth-Sciences 37, no. 1 (1988): 17-22.

Shendkar, Mangesh, and Pradeep Kumar. "Response reduction factor of RC framed structures with semi-interlocked masonry and unreinforced masonry infill." ICI J (2018): 24-28.

Shin, Myoungsu, and James M. LaFave. "Modeling of cyclic joint shear deformation contributions in RC beam-column connections to overall frame behavior." Structural Engineering and Mechanics 18, no. 5 (2004): 645-670.

Vamvatsikos, Dimitrios, and C. Allin Cornell. "Incremental dynamic analysis." Earthquake engineering & structural dynamics 31, no. 3 (2002): 491-514.

Vecchio, Frank J., and Mohamed Basil Emara. "Shear deformations in reinforced concrete frames." *ACI Structural journal* 89, no. 1 (1992): 46-56.

Venture, SAC Joint, and Guidelines Development Committee. Recommended seismic design criteria for new steel moment-frame buildings. Vol. 350. Washington, DC, USA: Federal Emergency Management Agency, 2000.

Viwathanatepa, Suthipoul, Egor Paul Popov, and Vitelmo Victorio Bertero. Effects of generalized loadings on bond of reinforcing bars embedded in confined concrete blocks. Oakland, CA, USA: University of California, Earthquake Engineering Research Center, 1979.

Wang, Hui Ying. "Influence of high mode effects on ductility reduction factors for MDOF shear-type structures." In Applied Mechanics and Materials, vol. 578, pp. 412-416. Trans Tech Publications Ltd, 2014.

Wood, Harry O., and Frank Neumann. "Modified Mercalli intensity scale of 1931." Bulletin of the Seismological Society of America 21, no. 4 (1931): 277-283.
Wu, J. P. "Inelastic response spectra with high damping." *J. Structural Division, ASCE* 115, no. 6 (1989): 1412-1431.

Yassin, Mohd Hisham Mohd. "Nonlinear analysis of prestressed concrete structures under monotonic and cyclic loads." PhD diss., University of California, Berkeley, 1994.

Youssef, M., and A. Ghobarah. "Modelling of RC beam-column joints and structural walls." Journal of earthquake engineering 5, no. 01 (2001): 93-111.

Zafar, Adeel. "Response modification factor of reinforced concrete moment resisting frames in developing countries." M.Sc. diss., University of Illinois, (2010).

Zarnic, Roko, and Miha Tomazevic. "An experimentally obtained method for evaluation of the behavior of masonry infilled RC frames." In Proceedings of the 9th world conference on earthquake engineering, vol. 6, pp. 163-168. 1988.

Žarnić, Roko, and Samo Gostič. "Masonry infilled frames as an effective structural sub-assemblage." Seismic design methodologies for the next generation of codes (1997): 335-346.

عمر حلاحلة, and محمد. "Influence of Bearing Non-Reinforced Parameter Walls with Stone Cladding on Fundamental Period Computation." PhD diss., 2019 ,جامعة النجاح الوطنية.

Appendix 1 – A Copy of the Interview Form and the Questions that have been Discussed

- This Interview is for research purposes in accordance with student's research regulations by Birzeit University.
- The aim is to identify building prototypes and shortcomings in construction practice.
- The collected data is confidential and will be used in the study and analysis for a master's degree thesis.
- It should last about 30 minutes. With your permission, I will audiotape and take notes during the interview, The recording is to accurately record the information you provide, and will be used for transcription purposes only.

Respondent Information

Name of the interviewee…………………………………………………………………………..
Years of experience……………………………………………………………………………….

Contact Details
E-mail……………………………………………… Telephone……………………………..

Section One: Building Characteristics

- Type of buildings based on construction material (Rank 1-3)

  Reinforced Concrete ☐

  Steel ☐

  Others ☐

- How buildings are constructed? (how many projects done as?)

  Floor by floor (stone infills are built simultaneously with RC concretes? ☐

  Skeleton and then cladding stone? ☐

- Type of infill?
  - type of material?
  - thickness?

- What is the building structural system (Trend?)
  - Moment resisting Frames ☐
  - Walls ☐

    One direction?

    both directions (aside from the stairs shear wall?)

- What is the building occupancy (most buildings you designed or constructed? Can you offer a percentage?)

  Residential ☐

  Commercial ☐

  Others ☐

- Number of stories of Buildings (including basements) (Rank the of building's number of stories following your experience in construction industry)

(1-3) stories ☐ (4-6) stories ☐
(7-9) stories ☐ (10-12) stories ☐
More than 12 stories ☐

- Rank the typical floor areas following your experience in construction industry.

Less than 100 m^2 ☐
(100-500) m^2 ☐
More than 500 m^2 ☐

Can you offer percentages of buildings that have the following irregularities?

- Plan regularity (Refer to Figure 1)

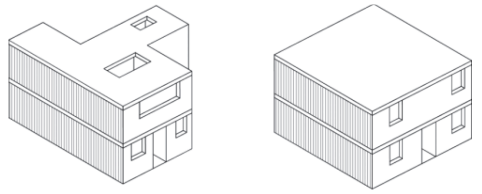
Regular plans ☐
Irregular plans ☐

- Plan symmetry (Refer to Figure 2)

Symmetric plan ☐
Anti-symmetric plan ☐

- Regularity of elevation (Refer to Figure 3)

Regular along elevation ☐
Irregular along elevation ☐



(A) (B)

Figure 1: A) Irregular plans and B) Regular plans

(A) (B)

Figure 2: A) Symmetric B) Anti-symmetric

(A) (B)

Figure 3: A) Regular elevation B) Irregular elevation

Section Two: Design Current Practice:

- What is the adapted design code in your designs?

- Which provisions you used to design buildings?

- Do you account for earthquake seismic loads in your analysis and design?

  Yes ☐ No ☐

  If your answer for the previous question is “Yes”, describe briefly how do you take these loads into consideration?

Section Three: Shortcomings and Deficiencies in Design and Construction Current Practice:

- Do the Buildings have soft storey in terms of infills, shear walls?

- Do the Buildings have short columns in buildings? (If they saw such columns in their work experience)
    - The columns stiffened partially along the height by infill or Mezzanine floor

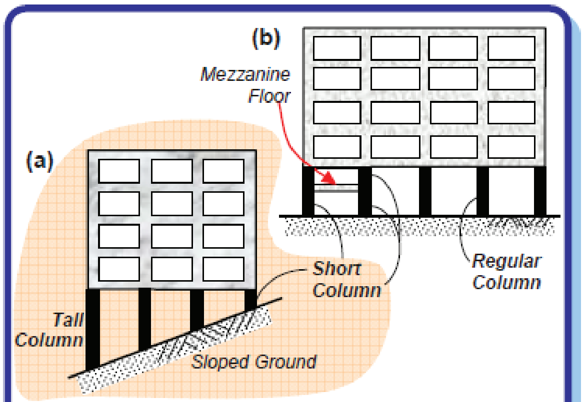


Figure 1: Buildings with short columns – *two explicit examples of common occurrences.*

Source: IITK-BMTPC Earthquake, C.V.R. Murty Indian Institute of Technology Kanpur Kanpur, India

- In beams and columns, is transverse reinforcement provided? what is the provided spacing? and how is it calculated?

- Can you describe the joint area in the current construction practice (transverse reinforcement, anchoring of steel bars)?

- What are the used provisions for development length?

- What are the typical concrete covers provided for RC elements?

- Other deficiencies you would like to elaborate or describe in details?

- In scale of 1-10 as shown below, how do you expect the existing structures to withstand severe earthquake load (1 indicates poor, 10 indicates perfect performance)?

Poor (1)	2	3	4	5	6	7	8	9	Perfect (10)

Given Rank………

- In scale of 1-10 as shown below, how do you expect the designed structures to withstand severe earthquake load (1 indicates poor, 10 indicates perfect performance)?

Poor (1)	2	3	4	5	6	7	8	9	Perfect (10)

Given Rank………

End of Interview

Appendix 2 – Full Record of the Calculation of R Factor

❖ Response Modification Factor Calculation for Each Prototype Model:

1) 3B6S MRFs-Bare Frame

Based on the capacity curve in Fig 7-2a, and the procedure explained in chapter three for evaluating the response modification factor R, the R factor was calculated for 3B6S MRFs-bare frame as the following:

1) Ductility Factor (R_μ):

- $T = 0.78, u = \frac{\Delta_{max}}{\Delta_y} = \frac{440mm}{90mm} = 4.88$
- $R_u = (2 * 4.88 - 1)^{0.5} + 2 * (T - 0.5) * (4.88 - (2 * 4.88 - 1)^{0.5} = 4.026$

2) Over Strength Factor ($\mathrm{R_0}$):

The maximum base shear (V_0) is found from the capacity curve. The designed base shear (V_d) is evaluated using Equivalent Lateral Force ELF in ASCE7-16.

- $V_0 = 372kN$
- V_d, the following equation has been utilized in the base shear calculations:

V = Cs W

Where:

- V is the base shear (KN)
- Cs is seismic response coefficient, ASCE7-16 section 12.8.1.1
- W is the effective seismic weight as defined in ASCE7-16, section 12.7.2

The given parameters for the frame system are as the following:

- Response modification factor R= 5
- Seismic coefficient C_s= 0.1075 (12.8.2)

$$V_d = Cs * W = 10.75\% * 1912.6 = 205.6kN$$

- $R_s = \frac{V_0}{V_d} = \frac{372kN}{205.6kN} = 1.81$

3) Redundancy Factor (R_r):

- $R_r = 0.86, since\ it\ is\ assuned\ to\ have\ three\ vertical\ seismic\ framing$

$$\boldsymbol{R = 4.026 * 1.81 * 0.86 = 6.26}$$

2) 3B6S MRFs-Bare Frame with Structural Deficiencies

Based on the capacity curve in Fig 7-2b, and the procedure explained in chapter three for evaluating the response modification factor R, the R factor was calculated for 3B6S MRFs-bare frame with structural deficiencies as the following:

1) Ductility Factor (R_μ):

- $T = 0.75, u = \frac{\Delta_{max}}{\Delta_y} = \frac{220mm}{80mm} = 2.75$
- $R_u = (2 * 2.75 - 1)^{0.5} + 2 * (T - 0.5) * (2.75 - (2 * 2.75 - 1)^{0.5} = 2.47$

2) Over Strength Factor (R_0):

The maximum base shear (V_0) is found from the capacity curve. The designed base shear (V_d) is evaluated using Equivalent Lateral Force ELF in ASCE7-16.

- $V_0 = 327kN$
- $V_d = 205.6kN$
- $R_s = \frac{V_0}{V_d} = \frac{327kN}{205.6kN} = 1.6$

3) Redundancy Factor (R_r):

- $R_r = 0.86, since\ it\ is\ assuned\ to\ have\ three\ vertical\ seismic\ framing$

$$\boldsymbol{R = 2.47 * 1.6 * 0.86 = 3.38}$$

3) 3B6S MRFs-Stone-Concrete Infilled Frame

Based on the capacity curve in Fig 7-3a, the ductility ratio for 3B6S MRFs-stone-concrete infilled frame is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{79mm}{35mm} = 2.26$
- $R_u = (2 * u - 1)^{0.5} = 1.87$

4) 3B6S MRFs-Stone-Concrete Infilled Frame without ground infills

Based on the capacity curve in Fig 7-3b, the ductility ratio for 3B6S MRFs-stone-concrete infilled frame without ground infills is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{170mm}{30mm} = 5.6$
- $R_u = (2 * u - 1)^{0.5} = 3.2$

5) 3B6S MRFs-Masonry-Concrete Infilled Frame

Based on the capacity curve in Fig 7-4a, the ductility ratio for 3B6S MRFs- masonry-concrete infilled frame is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{116mm}{20mm} = 5.8$
- $R_u = (2 * u - 1)^{0.5} = 3.26$

6) 3B6S MRFs-Masonry-Concrete Infilled Frame without ground infills

Based on the capacity curve in Fig 7-4b, the ductility ratio for 3B6S MRFs- masonry-concrete infilled frame without ground infills is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{184mm}{20mm} = 9.2$

- $R_u = (2 * u - 1)^{0.5} = 4.17$

7) 3B9S MRFs-Bare Frame

Based on the capacity curve in Fig 7-5a, and the procedure explained in chapter three for evaluating the response modification factor R, the R factor was calculated for 3B9S MRFs-bare frame as the following:

1) Ductility Factor (R_μ):

- $T = 1.06, u = \frac{\Delta_{max}}{\Delta_y} = \frac{550mm}{116mm} = 4.75$
- $R_u = 4.75$

2) Over Strength Factor (R_0):

The maximum base shear (V_0) is found from the capacity curve. The designed base shear (V_d) is evaluated using Equivalent Lateral Force ELF in ASCE7-16.

- $V_0 = 503kN$
- V_d, the following equation has been utilized in the base shear calculations:

V = Cs W

Where:

- V is the base shear (KN)
- Cs is seismic response coefficient, ASCE7-16 section 12.8.1.1
- W is the effective seismic weight as defined in ASCE7-16, section 12.7.2

The given parameters for the frame system are as the following:

- Response modification factor R= 5
- Seismic coefficient C_s= 0.1075 (12.8.2)

$$V_d = Cs * W = 10.75\% * 3100.7 = 333.32kN$$

- $R_s = \frac{V_0}{V_d} = \frac{503kN}{333.32kN} = 1.509$

3) Redundancy Factor (R_r):

- $R_r = 0.86, \mathit{since\ it\ is\ assuned\ to\ have\ three\ vertical\ seismic\ framing}$

$$\boldsymbol{R = 4.75 * 1.509 * 0.86 = 6.172}$$

8) 3B9S MRFs-Bare Frame with Structural Deficiencies

Based on the capacity curve in Fig 7-5b, and the procedure explained in chapter three for evaluating the response modification factor R, the R factor was calculated for 3B9S MRFs-bare frame with structural deficiencies as the following:

1) Ductility Factor (R_μ):

- $T = 1.01, u = \frac{\Delta_{max}}{\Delta_y} = \frac{305mm}{103mm} = 2.98$
- $R_u = 2.98$

2) Over Strength Factor (R_0):

The maximum base shear (V_0) is found from the capacity curve. The designed base shear (V_d) is evaluated using Equivalent Lateral Force ELF in ASCE7-16.

- $V_0 = 456kN$
- $V_d = 333.32kN$
- $R_s = \frac{V_0}{V_d} = \frac{456kN}{333.32kN} = 1.37$

3) Redundancy Factor (R_r):

- $R_r = 0.86, \mathit{since\ it\ is\ assuned\ to\ have\ three\ vertical\ seismic\ framing}$

$$\boldsymbol{R = 2.98 * 1.37 * 0.86 = 3.51}$$

9) 3B9S MRFs-Stone-Concrete Infilled Frame

Based on the capacity curve in Fig 7-6a, the ductility ratio for 3B9S MRFs-stone-concrete infilled frame is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{270mm}{75mm} = 3.6$
- $R_u = (2 * u - 1)^{0.5} + 2 * (T - 0.5) * (u - (2 * u - 1)^{0.5} = 2.93$

10) 3B9S MRFs-Stone-Concrete Infilled Frame without ground infills

Based on the capacity curve in Fig 7-6b, the ductility ratio for 3B9S MRFs- stone-concrete infilled frame without ground infills is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{310mm}{62mm} = 5.0$
- $R_u = (2 * u - 1)^{0.5} + 2 * (T - 0.5) * (u - (2 * u - 1)^{0.5} = 3.67$

11) 3B9S MRFs-Masonry-Concrete Infilled Frame

Based on the capacity curve in Fig 7-7a, the ductility ratio for 3B9S MRFs- masonry-concrete infilled frame is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{272mm}{33.17mm} = 8.2$
- $R_u = (2 * u - 1)^{0.5} + 2 * (T - 0.5) * (u - (2 * u - 1)^{0.5} = 5.2$

12) 3B9S MRFs-Masonry-Concrete Infilled Frame without ground infills

Based on the capacity curve in Fig 7-4b, the ductility ratio for 3B9S MRFs- masonry-concrete infilled frame without ground infills is calculated as the following:

Ductility Factor (R_{μ}):

- $u = \frac{\Delta_{max}}{\Delta_y} = \frac{305mm}{34mm} = 8.85$

- $R_u = (2 * u - 1)^{0.5} + 2 * (T - 0.5) * (u - (2 * u - 1)^{0.5} = 5.44$

Appendix 3 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B6S MRFs-Ductile Bare Frame

Appendix 3: 3B6S Bare Frame

1) Complementry files were defined to organize and make the procedure easier:

1. Library for Units

```
       # define UNITS-----------------------------------------------------------------
set m 1.;                             # define basic units--output units
set kn 1.;                            # define basic units--output units
set sec 1.;                           # define basic units--output units

set LunitTXT "meter";                 # define basic-unit text for output
set FunitTXT "kn";                    # define basic-unit text for output
set TunitTXT "sec";                   # define basic-unit text for output

set cm [expr $m/100.];                # define engineering units
set mm [expr $m/1000.];
set ton [expr $kn/10.];
set pa [expr $kn*0.001/pow($m,2) ];
set kpa [expr $pa*1000.];
set mpa [expr $kpa*1000.];
set gpa [expr $mpa*1000.];
set density [expr $kn/pow($m,3)];

set g [expr 9.81*$m/pow($sec,2)];     # gravitational acceleration
set Ubig 1.e10;         # a really large number
```

2. Building RC Cross-Section (Fiber Appraoch)

```
   proc BuildRCrectSection {id HSec BSec coverH coverB coreID coverID steelID numBarsTop barAreaTop numBarsBot barAreaBot numBarsIntTot barAreaInt nfCoreY nfCoreZ nfCoverY nfCoverZ} {
   ################################################
   # BuildRCrectSection $id $HSec $BSec $coverH $coverB $coreID $coverID $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
   ################################################
   # 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-z 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
   #    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 z 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
   #    nfCoreZ - number of fibers in the core patch in the z direction
   #    nfCoverY - number of fibers in the cover patches with long sides in the y direction
   #    nfCoverZ - number of fibers in the cover patches with long sides in the z direction
   #
   #                        y
   #                        ^
   #                        |
   #             ---------------------     --   --
   #             |   o     o     o    |     |    -- coverH
   #             |                      |     |
   #             |   o           o    |     |
   #    z <--- |          +          |     HSec
   #             |   o           o    |     |
   #             |                      |     |
   #             |   o o o o o o    |     |    -- coverH
   #             ---------------------     --   --
   #             |-------Bsec------|
   #             |---| coverB  |---|
   #
   #                       y
```

```
   #                       ^
   #                       |
   #             ---------------------
   #             |\      cover        /|
   #             | \------Top------/ |
   #             |c|                   |c|
   #             |o|                   |o|
   #  z <-----|v|       core      |v|  HSec
   #             |e|                   |e|
   #             |r|                    |r|
   #             | /-------Bot------\ |
   #             |/      cover        \|
   #             ---------------------
   #                       Bsec
   #
   #
   # 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
   #
   set coverY [expr $HSec/2.0];     # The distance from the section z-axis to the edge of the cover concrete -- outer edge of cover concrete
   set coverZ [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 z-axis to the edge of the core concrete --  edge of the core concrete/inner edge of cover concrete
   set coreZ [expr $coverZ-$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 $nfCoreZ $nfCoreY -$coreY $coreZ -$coreY -$coreZ $coreY -$coreZ $coreY $coreZ

      # Define the four cover patches
      patch quadr $coverID 2 $nfCoverY -$coverY $coverZ -$coreY $coreZ $coreY $coreZ $coverY $coverZ
      patch quadr $coverID 2 $nfCoverY -$coreY -$coreZ -$coverY -$coverZ $coverY -$coverZ $coreY -$coreZ
      patch quadr $coverID $nfCoverZ 2 -$coverY $coverZ -$coverY -$coverZ -$coreY -$coreZ -$coreY $coreZ
      patch quadr $coverID $nfCoverZ 2 $coreY $coreZ $coreY -$coreZ $coverY -$coverZ $coverY $coverZ

      # define reinforcing layers
      layer straight $steelID $numBarsInt $barAreaInt  -$coreY $coreZ $coreY $coreZ;   # intermediate skin reinf. +z
      layer straight $steelID $numBarsInt $barAreaInt  -$coreY -$coreZ $coreY -$coreZ; # intermediate skin reinf. -z
      layer straight $steelID $numBarsTop $barAreaTop $coreY $coreZ $coreY -$coreZ; # top layer reinfocement
      layer straight $steelID $numBarsBot $barAreaBot  -$coreY $coreZ  -$coreY -$coreZ;   # bottom layer reinforcement

   }; # end of fibersection definition
};    # end of procedure

   3. Display The Model in 2D

   proc DisplayModel2D { {ShapeType nill} {dAmp 5}  {xLoc 10} {yLoc 10} {xPixels 512} {yPixels 384} {nEigen 1} } {
   ######################################################################################
   ## DisplayModel2D $ShapeType $dAmp $xLoc $yLoc $xPixels $yPixels $nEigen
   ######################################################################################
   ## display Node Numbers, Deformed or Mode Shape in 2D problem
   ##       Silvia Mazzoni & Frank McKenna, 2006
   ##
   ##    ShapeType :    type of shape to display. # options: ModeShape , NodeNumbers , DeformedShape
   ##    dAmp :      relative amplification factor for deformations
   ##    xLoc,yLoc  :   horizontal & vertical location in pixels of graphical window (0,0=upper left-most corner)
   ##    xPixels,yPixels : width & height of graphical window in pixels
   ##    nEigen :       if nEigen not=0, show mode shape for nEigen eigenvalue
   ##
   #######################################################################################
   global TunitTXT;              # load time-unit text
   global ScreenResolutionX ScreenResolutionY;  # read global values for screen resolution

   if {  [info exists TunitTXT] != 1} {set TunitTXT ""};    # set blank if it has not been defined previously.

   if {  [info exists ScreenResolutionX] != 1} {set ScreenResolutionX 1024};     # set default if it has not been defined previously.
   if {  [info exists ScreenResolutionY] != 1} {set ScreenResolutionY 768};      # set default if it has not been defined previously.

   if {$xPixels == 0} {
      set xPixels [expr int($ScreenResolutionX/2)];
      set yPixels [expr int($ScreenResolutionY/2)]
      set xLoc 10
```

```
      set yLoc 10
   }
   if {$ShapeType == "nill"} {
      puts ""; puts ""; puts "------------------"
      puts "View the Model? (N)odes, (D)eformedShape, anyMode(1),(2),(#). Press enter for NO."
      gets stdin answer
      if {[llength $answer]>0 } {
         if {$answer != "N" & $answer != "n"} {
            puts "Modify View Scaling Factor=$dAmp? Type factor, or press enter for NO."
            gets stdin answerdAmp
            if {[llength $answerdAmp]>0 } {
               set dAmp $answerdAmp
            }
         }
         if {[string index $answer 0] == "N" || [string index $answer 0] == "n"} {
            set ShapeType NodeNumbers
         } elseif {[string index $answer 0] == "D" ||[string index $answer 0] == "d" } {
            set ShapeType DeformedShape
         } else {
            set ShapeType ModeShape
            set nEigen $answer
         }
      } else {
         return
      }
   }

   if {$ShapeType ==  "ModeShape" } {
      set lambdaN [eigen $nEigen];     # perform eigenvalue analysis for ModeShape
      set lambda [lindex $lambdaN [expr $nEigen-1]];
      set omega [expr pow($lambda,0.5)]
      set PI   [expr 2*asin(1.0)];     # define constant
      set Tperiod [expr 2*$PI/$omega];      # period (sec.)
      set fmt1 "Mode Shape, Mode=%.1i Period=%.3f %s  "
      set windowTitle [format $fmt1 $nEigen $Tperiod  $TunitTXT]
   } elseif  {$ShapeType ==  "NodeNumbers" } {
      set windowTitle "Node Numbers"
   } elseif  {$ShapeType ==  "DeformedShape" } {
      set windowTitle "Deformed Shape"
   }

   set viewPlane XY
   recorder display $windowTitle $xLoc $yLoc $xPixels $yPixels  -wipe ; # display recorder
   DisplayPlane $ShapeType $dAmp $viewPlane $nEigen 0
   after 3000; #pause for 2 seconds to display
}

   4. Display Plane Deformed Shape for 2D Model

   proc DisplayPlane {ShapeType dAmp viewPlane {nEigen 0}  {quadrant 0}} {
   #######################################################################################
   ## DisplayPlane $ShapeType $dAmp $viewPlane $nEigen $quadrant
   #######################################################################################
   ## setup display parameters for specified viewPlane and display
   ##          Silvia Mazzoni & Frank McKenna, 2006
   ##
   ##    ShapeType :    type of shape to display. # options: ModeShape , NodeNumbers , DeformedShape
   ##    dAmp :      relative amplification factor for deformations
   ##    viewPlane : set local xy axes in global coordinates (XY,YX,XZ,ZX,YZ,ZY)
   ##    nEigen :       if nEigen not=0, show mode shape for nEigen eigenvalue
   ## quadrant:      quadrant where to show this figure (0=full figure)
   ##
   #######################################################################################

   set Xmin [lindex [nodeBounds] 0];   # view bounds in global coords -  will add padding on the sides
   set Ymin [lindex [nodeBounds] 1];
   set Zmin [lindex [nodeBounds] 2];
   set Xmax [lindex [nodeBounds] 3];
   set Ymax [lindex [nodeBounds] 4];
   set Zmax [lindex [nodeBounds] 5];

   set Xo 0;   # center of local viewing system
   set Yo 0;
   set Zo 0;

   set uLocal [string index $viewPlane 0];   # viewPlane local-x axis in global coordinates
   set vLocal [string index $viewPlane 1];   # viewPlane local-y axis in global coordinates


   if  {$viewPlane =="3D" } {
```

```
      set uMin $Zmin+$Xmin
      set uMax $Zmax+$Xmax
      set vMin $Ymin
      set vMax $Ymax
      set wMin -10000
      set wMax 10000
      vup 0 1 0; # dirn defining up direction of view plane
   } else {
      set keyAxisMin "X $Xmin Y $Ymin Z $Zmin"
      set keyAxisMax "X $Xmax Y $Ymax Z $Zmax"
      set axisU [string index $viewPlane 0];
      set axisV [string index $viewPlane 1];
      set uMin [string map $keyAxisMin $axisU]
      set uMax [string map $keyAxisMax $axisU]
      set vMin [string map $keyAxisMin $axisV]
      set vMax [string map $keyAxisMax $axisV]
      if {$viewPlane =="YZ" || $viewPlane =="ZY" } {
         set wMin $Xmin
         set wMax $Xmax
      } elseif  {$viewPlane =="XY" || $viewPlane =="YX" } {
         set wMin $Zmin
         set wMax $Zmax
      } elseif  {$viewPlane =="XZ" || $viewPlane =="ZX" } {
         set wMin $Ymin
         set wMax $Ymax
      } else {
      return -1
      }
   }

   set epsilon 1e-3; # make windows width or height not zero when the Max and Min values of a coordinate are the same

   set uWide [expr $uMax - $uMin+$epsilon];
   set vWide [expr $vMax - $vMin+$epsilon];
   set uSide [expr 0.25*$uWide];
   set vSide [expr 0.25*$vWide];
   set uMin [expr $uMin - $uSide];
   set uMax [expr $uMax + $uSide];
   set vMin [expr $vMin - $vSide];
   set vMax [expr $vMax + 2*$vSide];   # pad a little more on top, because of window title
   set uWide [expr $uMax - $uMin+$epsilon];
   set vWide [expr $vMax - $vMin+$epsilon];
   set uMid [expr ($uMin+$uMax)/2];
   set vMid [expr ($vMin+$vMax)/2];

   # keep the following general, as change the X and Y and Z for each view plane
   # next three commmands define viewing system, all values in global coords
   vrp $Xo $Yo $Zo;    # point on the view plane in global coord, center of local viewing system
   if {$vLocal == "X"} {
      vup 1 0 0; # dirn defining up direction of view plane
   } elseif {$vLocal == "Y"} {
      vup 0 1 0; # dirn defining up direction of view plane
   } elseif {$vLocal == "Z"} {
      vup 0 0 1; # dirn defining up direction of view plane
   }
   if {$viewPlane =="YZ" } {
      vpn 1 0 0; # direction of outward normal to view plane
      prp 10000. $uMid $vMid ; # eye location in local coord sys defined by viewing system
      plane 10000 -10000; # distance to front and back clipping planes from eye
   } elseif  {$viewPlane =="ZY" } {
      vpn -1 0 0; # direction of outward normal to view plane
      prp -10000. $vMid $uMid ; # eye location in local coord sys defined by viewing system
      plane 10000 -10000; # distance to front and back clipping planes from eye
   } elseif  {$viewPlane =="XY"  } {
      vpn 0 0 1; # direction of outward normal to view plane
      prp $uMid $vMid 10000; # eye location in local coord sys defined by viewing system
      plane 10000 -10000; # distance to front and back clipping planes from eye
   } elseif  {$viewPlane =="YX" } {
      vpn 0 0 -1; # direction of outward normal to view plane
      prp $uMid $vMid -10000; # eye location in local coord sys defined by viewing system
      plane 10000 -10000; # distance to front and back clipping planes from eye
   } elseif  {$viewPlane =="XZ" } {
      vpn 0 -1 0; # direction of outward normal to view plane
      prp $uMid -10000 $vMid ; # eye location in local coord sys defined by viewing system
      plane 10000 -10000; # distance to front and back clipping planes from eye
   } elseif  {$viewPlane =="ZX" } {
      vpn 0 1 0; # direction of outward normal to view plane
      prp $uMid 10000 $vMid ; # eye location in local coord sys defined by viewing system
      plane 10000 -10000; # distance to front and back clipping planes from eye
   } elseif  {$viewPlane =="3D" } {
```

```
      vpn 1 0.25 1.25; # direction of outward normal to view plane
      prp -100 $vMid 10000; # eye location in local coord sys defined by viewing system
      plane 10000 -10000; # distance to front and back clipping planes from eye
   }  else {
      return -1
   }
   # next three commands define view, all values in local coord system
   if  {$viewPlane =="3D" } {
      viewWindow [expr $uMin-$uWide/4] [expr $uMax/2] [expr $vMin-0.25*$vWide] [expr $vMax]
   } else {
      viewWindow $uMin $uMax $vMin $vMax
   }
   projection 1;  # projection mode, 0:prespective, 1: parallel
   fill 1;     # fill mode; needed only for solid elements

   if {$quadrant == 0} {
      port -1 1 -1 1    # area of window that will be drawn into (uMin,uMax,vMin,vMax);
   } elseif {$quadrant == 1} {
      port 0 1 0 1   # area of window that will be drawn into (uMin,uMax,vMin,vMax);
   } elseif {$quadrant == 2} {
      port -1 0 0 1  # area of window that will be drawn into (uMin,uMax,vMin,vMax);
   } elseif {$quadrant == 3} {
      port -1 0 -1 0    # area of window that will be drawn into (uMin,uMax,vMin,vMax);
   } elseif {$quadrant == 4} {
      port 0 1 -1 0  # area of window that will be drawn into (uMin,uMax,vMin,vMax);
   }

   if {$ShapeType ==  "ModeShape" } {
      display -$nEigen 0  [expr 5.*$dAmp];   # display mode shape for mode $nEigen
   } elseif  {$ShapeType ==  "NodeNumbers" } {
      display 1 -1 0  ;       # display node numbers
   } elseif  {$ShapeType ==  "DeformedShape" }  {
      display 1 2 $dAmp;      # display deformed shape  the 2 makes the nodes small
   }
};
                                #
######################################################################################


   5. Procedure for Defining Uniaxial Pinching Material

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

# #

# procUniaxialPinching.tcl #

# procedure for activating the pinching material given its parameters in the form of list #

# created NM (nmitra@u.washington.edu) dated : Feb 2002 #

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

proc procUniaxialPinching { materialTag pEnvelopeStress nEnvelopeStress pEnvelopeStrain nEnvelopeStrain rDisp rForce uForce gammaK gammaD gammaF gammaE damage} {

}


2) 2D Model Definition for 3B6S Bare Frame:


#performing nonlinear static pushover analysis on 3B6S Bare Frame
##########################################################################################################################

   wipe all;
# define model builder
#    model basic builder  -ndm $ndm <-ndf $ndf>
     model basic builder  -ndm 2  -ndf 3

     set dataDir Results;           # set up name of data directory
     file mkdir $dataDir;           # create data directory
     source Libunits.tcl;               # define basic system units
     source DisplayModel2D.tcl;        # procedure for displaying a 2D View of model
     source DisplayPlane.tcl;       # procedure for displaying a plane in a model

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

# buiding geometry
###########################################################################################################################
```

```

# dimensions

    set span1 4000.0;
    set span2 4000.0;
    set span3 4000.0;
    set storey1 3000.0;
    set storey2 3000.0;
    set storey3 3000.0;
    set storey4 3000.0;
    set storey5 3000.0;
    set storey6 3000.0;

# main grid lines
    # vertical axis, x
    set x1 [expr 0];
    set x2 [expr $x1+$span1];
    set x3 [expr $x2+$span2];
    set x4 [expr $x3+$span3];

    # hoeizontal axis, y
    set z0 [expr 0];
    set z1 [expr $z0+$storey1];
    set z2 [expr $z1+$storey2];
    set z3 [expr $z2+$storey3];
    set z4 [expr $z3+$storey4];
    set z5 [expr $z4+$storey5];
    set z6 [expr $z5+$storey6];

# definition of nodes

      #assigning node tages                    # for axises A,B,C, and D.
      set N_A0     1;
      set N_B0     2;
      set N_C0     3;
      set N_D0     4;
      set N_A1     5;
      set N_B1     6;
      set N_C1     7;
      set N_D1     8;
      set N_A2     9;
      set N_B2     10;
      set N_C2     11;
      set N_D2     12;
      set N_A3     13;
      set N_B3     14;
      set N_C3     15;
      set N_D3     16;
      set N_A4     17;
      set N_B4     18;
      set N_C4     19;
      set N_D4     20;
      set N_A5     21;
      set N_B5     22;
      set N_C5     23;
      set N_D5     24;
      set N_A6     25;
      set N_B6     26;
      set N_C6     27;
      set N_D6     28;

      set N_A10_R   29;                          # N_Aij_R    i: story level.   j: axis number
      set N_A10_A   30;
      set N_A10_L   31;
      set N_A20_R   32;
      set N_A20_A   33;
      set N_A20_L   34;
      set N_A30_R   35;
      set N_A30_A   36;
      set N_A30_L   37;
      set N_A40_R   38;
      set N_A40_A   39;
      set N_A40_L   40;
      set N_A50_R   41;
      set N_A50_A   42;
      set N_A50_L   43;
      set N_A60_R   44;
      set N_A60_A   45;
      set N_A60_L   46;
```

```

set N_B11_R   47;
set N_B11_A   48;
set N_B11_L   49;
set N_B21_R   50;
set N_B21_A   51;
set N_B21_L   52;
set N_B31_R   53;
set N_B31_A   54;
set N_B31_L   55;
set N_B41_R   56;
set N_B41_A   57;
set N_B41_L   58;
set N_B51_R   59;
set N_B51_A   60;
set N_B51_L   61;
set N_B61_R   62;
set N_B61_A   63;
set N_B61_L   64;

set N_C12_R   65;
set N_C12_A   66;
set N_C12_L   67;
set N_C22_R   68;
set N_C22_A   69;
set N_C22_L   70;
set N_C32_R   71;
set N_C32_A   72;
set N_C32_L   73;
set N_C42_R   74;
set N_C42_A   75;
set N_C42_L   76;
set N_C52_R   77;
set N_C52_A   78;
set N_C52_L   79;
set N_C62_R   80;
set N_C62_A   81;
set N_C62_L   82;

set N_D13_R   83;
set N_D13_A   84;
set N_D13_L   85;
set N_D23_R   86;
set N_D23_A   87;
set N_D23_L   88;
set N_D33_R   89;
set N_D33_A   90;
set N_D33_L   91;
set N_D43_R   92;
set N_D43_A   93;
set N_D43_L   94;
set N_D53_R   95;
set N_D53_A   96;
set N_D53_L   97;
set N_D63_R   98;
set N_D63_A   99;
set N_D63_L   100;


#node $nodetag (ndm $coords) <-mass (ndf $massvalues)>

set col_halfdepA [expr 600/2];              # This is used to define the joint dimensions.
set col_halfdepB [expr 600/2];
set col_halfdepC [expr 600/2];
set col_halfdepD [expr 600/2];
set beam_halfdep1 [expr 300/2];
set beam_halfdep2 [expr 300/2];
set beam_halfdep3 [expr 300/2];
set beam_halfdep4 [expr 300/2];
set beam_halfdep5 [expr 300/2];
set beam_halfdep6 [expr 300/2];

node  $N_A0      $x1 $z0;
node  $N_B0      $x2 $z0;
node  $N_C0      $x3 $z0;
node  $N_D0      $x4 $z0;
node  $N_A1      $x1 [expr $z1-$beam_halfdep1];
node  $N_B1      $x2 [expr $z1-$beam_halfdep1];
node  $N_C1      $x3 [expr $z1-$beam_halfdep1];
node  $N_D1      $x4 [expr $z1-$beam_halfdep1];
```

```
      node  $N_A2      $x1 [expr $z2-$beam_halfdep2];
      node  $N_B2      $x2 [expr $z2-$beam_halfdep2];
      node  $N_C2      $x3 [expr $z2-$beam_halfdep2];
      node  $N_D2      $x4 [expr $z2-$beam_halfdep2];
      node  $N_A3      $x1 [expr $z3-$beam_halfdep3];
      node  $N_B3      $x2 [expr $z3-$beam_halfdep3];
      node  $N_C3      $x3 [expr $z3-$beam_halfdep3];
      node  $N_D3      $x4 [expr $z3-$beam_halfdep3];
      node  $N_A4      $x1 [expr $z4-$beam_halfdep4];
      node  $N_B4      $x2 [expr $z4-$beam_halfdep4];
      node  $N_C4      $x3 [expr $z4-$beam_halfdep4];
      node  $N_D4      $x4 [expr $z4-$beam_halfdep4];
      node  $N_A5      $x1 [expr $z5-$beam_halfdep5];
      node  $N_B5      $x2 [expr $z5-$beam_halfdep5];
      node  $N_C5      $x3 [expr $z5-$beam_halfdep5];
      node  $N_D5      $x4 [expr $z5-$beam_halfdep5];
      node  $N_A6      $x1 [expr $z6-$beam_halfdep6];
      node  $N_B6      $x2 [expr $z6-$beam_halfdep6];
      node  $N_C6      $x3 [expr $z6-$beam_halfdep6];
      node  $N_D6      $x4 [expr $z6-$beam_halfdep6];


############### add nodes - joints #######################################################

                                                   #  R: node at the right side of joint
                                                   #  A: node above the joint
                                                   #  L: node at the left side of the joint
      node $N_A10_R   [expr $x1+$col_halfdepA] $z1;
      node $N_A10_A   $x1 [expr $z1+$beam_halfdep1];
      node $N_A10_L   [expr $x1-$col_halfdepA] $z1;
      node $N_A20_R   [expr $x1+$col_halfdepA] $z2;
      node $N_A20_A   $x1 [expr $z2+$beam_halfdep2];
      node $N_A20_L   [expr $x1-$col_halfdepA] $z2;
      node $N_A30_R   [expr $x1+$col_halfdepA] $z3;
      node $N_A30_A   $x1 [expr $z3+$beam_halfdep3];
      node $N_A30_L   [expr $x1-$col_halfdepA] $z3;
      node $N_A40_R   [expr $x1+$col_halfdepA] $z4;
      node $N_A40_A   $x1 [expr $z4+$beam_halfdep4];
      node $N_A40_L   [expr $x1-$col_halfdepA] $z4;
      node $N_A50_R   [expr $x1+$col_halfdepA] $z5;
      node $N_A50_A   $x1 [expr $z5+$beam_halfdep5];
      node $N_A50_L   [expr $x1-$col_halfdepA] $z5;
      node $N_A60_R   [expr $x1+$col_halfdepA] $z6;
      node $N_A60_A   $x1 [expr $z6+$beam_halfdep6];
      node $N_A60_L   [expr $x1-$col_halfdepA] $z6;

      node $N_B11_R   [expr $x2+$col_halfdepB] $z1;
      node $N_B11_A   $x2 [expr $z1+$beam_halfdep1];
      node $N_B11_L   [expr $x2-$col_halfdepB] $z1;
      node $N_B21_R   [expr $x2+$col_halfdepB] $z2;
      node $N_B21_A   $x2 [expr $z2+$beam_halfdep2];
      node $N_B21_L   [expr $x2-$col_halfdepB] $z2;
      node $N_B31_R   [expr $x2+$col_halfdepB] $z3;
      node $N_B31_A   $x2 [expr $z3+$beam_halfdep3];
      node $N_B31_L   [expr $x2-$col_halfdepB] $z3;
      node $N_B41_R   [expr $x2+$col_halfdepB] $z4;
      node $N_B41_A   $x2 [expr $z4+$beam_halfdep4];
      node $N_B41_L   [expr $x2-$col_halfdepB] $z4;
      node $N_B51_R   [expr $x2+$col_halfdepB] $z5;
      node $N_B51_A   $x2 [expr $z5+$beam_halfdep5];
      node $N_B51_L   [expr $x2-$col_halfdepB] $z5;
      node $N_B61_R   [expr $x2+$col_halfdepB] $z6;
      node $N_B61_A   $x2 [expr $z6+$beam_halfdep6];
      node $N_B61_L   [expr $x2-$col_halfdepB] $z6;

      node $N_C12_R   [expr $x3+$col_halfdepC] $z1;
      node $N_C12_A   $x3 [expr $z1+$beam_halfdep1];
      node $N_C12_L   [expr $x3-$col_halfdepC] $z1;
      node $N_C22_R   [expr $x3+$col_halfdepC] $z2;
      node $N_C22_A   $x3 [expr $z2+$beam_halfdep2];
      node $N_C22_L   [expr $x3-$col_halfdepC] $z2;
      node $N_C32_R   [expr $x3+$col_halfdepC] $z3;
      node $N_C32_A   $x3 [expr $z3+$beam_halfdep3];
      node $N_C32_L   [expr $x3-$col_halfdepC] $z3;
      node $N_C42_R   [expr $x3+$col_halfdepC] $z4;
      node $N_C42_A   $x3 [expr $z4+$beam_halfdep4];
      node $N_C42_L   [expr $x3-$col_halfdepC] $z4;
      node $N_C52_R   [expr $x3+$col_halfdepC] $z5;
      node $N_C52_A   $x3 [expr $z5+$beam_halfdep5];
      node $N_C52_L   [expr $x3-$col_halfdepC] $z5;
```

```
      node $N_C62_R   [expr $x3+$col_halfdepC] $z6;
      node $N_C62_A   $x3 [expr $z6+$beam_halfdep6];
      node $N_C62_L   [expr $x3-$col_halfdepC] $z6;

      node $N_D13_R   [expr $x4+$col_halfdepD] $z1;
      node $N_D13_A   $x4 [expr $z1+$beam_halfdep1];
      node $N_D13_L   [expr $x4-$col_halfdepD] $z1;
      node $N_D23_R   [expr $x4+$col_halfdepD] $z2;
      node $N_D23_A   $x4 [expr $z2+$beam_halfdep2];
      node $N_D23_L   [expr $x4-$col_halfdepD] $z2;
      node $N_D33_R   [expr $x4+$col_halfdepD] $z3;
      node $N_D33_A   $x4 [expr $z3+$beam_halfdep3];
      node $N_D33_L   [expr $x4-$col_halfdepD] $z3;
      node $N_D43_R   [expr $x4+$col_halfdepD] $z4;
      node $N_D43_A   $x4 [expr $z4+$beam_halfdep4];
      node $N_D43_L   [expr $x4-$col_halfdepD] $z4;
      node $N_D53_R   [expr $x4+$col_halfdepD] $z5;
      node $N_D53_A   $x4 [expr $z5+$beam_halfdep5];
      node $N_D53_L   [expr $x4-$col_halfdepD] $z5;
      node $N_D63_R   [expr $x4+$col_halfdepD] $z6;
      node $N_D63_A   $x4 [expr $z6+$beam_halfdep6];
      node $N_D63_L   [expr $x4-$col_halfdepD] $z6;


# restraints

      #basefix  $nodetag  (ndf  $constraints)
      fix       $N_A0           1 1 1;
      fix       $N_B0           1 1 1;
      fix       $N_C0           1 1 1;
      fix       $N_D0           1 1 1;


############################################################################################################################
##
# material definitions
############################################################################################################################
##

#   Definition of materials IDs

       #set C_confinedB     1;
       set C_confined    1;
       set C_unconfined  2;
       set R_steel       3;



#   basic parameters for materials-con-concrete

# ConfinedConcrete01 Material

      #$tag           integer tag identifying material.
      #$secType         tag for the transverse reinforcement configuration.
      #$fpc           unconfined cylindrical strength of concrete specimen.
      #$Ec             initial elastic modulus of unconfined concrete.
      #<-epscu $epscu> OR <-gamma $gamma> confined concrete ultimate strain.
      #<-nu $nu> OR <-varub> OR <-varnoub>   Poisson's Ratio.
      #$L1            length/diameter of square/circular core section measured respect to the hoop center line.
      #($L2)            additional dimensions when multiple hoops are being used.
      #$phis         hoop diameter. If section arrangement has multiple hoops it refers to the external hoop.
      #$S             hoop spacing.
      #$fyh           yielding strength of the hoop steel.
      #$Es0             elastic modulus of the hoop steel.
      #$haRatio         hardening ratio of the hoop steel.
      #$mu            ductility factor of the hoop steel.
      #$phiLon       diameter of longitudinal bars.

#   basic parameters for materials-uncon-concrete

       set unconfc   -28.0;        # compression strength for concrete
       set unconepsc  -0.002;                 # strain at maximum stress in compression
       set unconfu   [expr $unconfc*0.18];          # ultamate stress for concrete
       set unconepsu  -0.01;                  # strain at ultimate stress in compression
       set unconlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set unconft   [expr $unconfc*-0.1];         # maximum stress in tension for concrete
       set unconEt   [expr $unconft/0.002];        # elastic modulus in tension
       set unconE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

#   basic parameters for material--steel        # ReinforcingSteel uniaxial material object. This object is intended to be u
```

```
sed in a reinforced concrete fiber section as the steel reinforcing material.

       set Fy 420.0;    # Yield stress in tension
       set Fu 596.0;    # Ultimate stress in tension
       set Es 200000.0;    # Initial elastic tangent
       set Esh 3100.0;     # Tangent at initial strain hardening
       set esh 0.01;       # Strain corresponding to initial strain hardening
       set eult 0.09;         # Strain at peak stress

   #uniaxialMaterial ReinforcingSteel $matTag $fy $fu $Es $Esh $esh $eult    Define ReinforcingSteel uniaxial material
   uniaxialMaterial ReinforcingSteel $R_steel $Fy $Fu $Es $Esh $esh $eult -DMBuck 6 0.8 -CMFatigue 0.2600 0.5000 0.3890 -IsoHard 4.3000 0.01

#   definition of ConfinedConcrete01 material

    #uniaxialMaterial ConfinedConcrete01 $tag        $secType $fpc  $Ec   -epscu $epscu   $nu    $L1   $L2    $phis  $S   $fyh  $Es0     $haRatio  $mu  $phiLon  -stRatio $stRatio
    #uniaxialMaterial ConfinedConcrete01  $C_confinedB     R      -28 24870.1 -epscu -0.04   -varUB 250.0 1450.0 10.0  125.0 420.0 200000.0 0.00      3100.0 12.0    -stRatio 0.85
    #uniaxialMaterial ConfinedConcrete01  $C_confinedC     R      -28 24870.1 -epscu -0.04   -varUB 550.0 200.0  10.0  125.0 420.0 200000.0 0.00      3100.0 18.0    -stRatio 0.85


#   basic parameters for materials-con-concrete

       set confc   -32.5;        # compression strength for concrete
       set conepsc  -0.003;                 # strain at maximum stress in compression
       set confu   [expr $confc*0.18];           # ultamate stress for concrete
       set conepsu  -0.04;                  # strain at ultimate stress in compression
       set conlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set conft   [expr $confc*-0.1];          # maximum stress in tension for concrete
       set conEt   [expr $conft/0.002];         # elastic modulus in tension
       set conE0   [expr 2*$confc/$conepsc];      #intial elastic tangent

    # uniaxialMaterial Concrete02  $matTag     $fpc $epsc0 $fpcu  $epsU  $lambda  $ft  $Ets
    uniaxialMaterial   Concrete02  $C_unconfined $unconfc  $unconepsc  $unconfu    $unconepsu  $unconlambda  $unconft  $unconEt;
    uniaxialMaterial   Concrete02  $C_confined $confc  $conepsc  $confu    $conepsu  $conlambda  $conft  $conEt;

################################################################################################################################################
# definition of the Sections
################################################################################################################################################

#   define sections IDs

    set col25x60  1;
    set beam150x30 2;

#   define section parameters

    set pi         3.141593;
    set rebar_12  [expr $pi*12.0*12.0/4];  # area rebar 12mm
    set rebar_18  [expr $pi*18.0*18.0/4];
    set w_col     250.0;   # column width
    set h_col     600.0;   # column hieght
    set c_col     20.0;  # column cover
    set w_beam    1500.0;  # beam width
    set h_beam    300.0;   # beam hieght
    set c_beam    30.0;  # beam cover

#   load procedure for fiber section

source BuildRCrectSection.tcl;

#   build sections

    #BuildRCrectSection $ColSecTag $HSec   $BSec   $coverH  $coverB  $coreID      $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
    BuildRCrectSection  $col25x60  $h_col  $w_col  $c_col   $c_col   $C_confined $C_unconfined $R_steel 4           $rebar_18   4           $rebar_18   2              $rebar_18   8        8        8         8
    BuildRCrectSection  $beam150x30 $h_beam $w_beam $c_beam  $c_beam $C_confined $C_unconfined $R_steel 12          $rebar_12   8          $rebar_12   0              $rebar_12   8        8        8         8


##############################################################################################################################
# beam column joint definition
############################################################################################################################
```

```
##

# dimensions of the joint respectively
set JointWidth [expr $h_col]; set JointHeight [expr $h_beam]; set JointDepth $w_col ;
set JointVolume [expr $JointWidth*$JointHeight*$JointDepth];

######################### details for the material models of bar slip of the beam ####################################

set bs_fc 28.0; set bs_fs 420.0; set bs_es 200000; set bs_fsu 596; set bs_dbar 12.0; set bs_esh 3100.0;
set bs_wid $w_col; set bs_dep $h_beam;
set bsT_nbars 12; set bsB_nbars 8;
set bs_ljoint $h_col;

########################## details for the material models of bar slip of the column ##################################

set cs_fc 28.0; set cs_fs 420.0; set cs_es 200000.0; set cs_fsu 596; set cs_dbar 18.0; set cs_esh 3100.0;
set cs_wid $w_col; set cs_dep $h_col;
set cs_nbars 5;
set cs_ljoint $h_beam;

########################################################################################################################
#bar slip definition

# for beam bottom

set bsid1 11
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid1 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsB_nbars $bs_wid $bs_dep  strong beambot

# for beam top

set bsid2 21
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid2 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsT_nbars $bs_wid $bs_dep  strong beamtop

# for columns
set bsid3 31

#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid3 $cs_fc $cs_fs $cs_es $cs_fsu $cs_esh $cs_dbar $cs_ljoint $cs_nbars $cs_wid $cs_dep  strong column


############################ material for shear panel ##############################################################

## Positive/Negative envelope Stress

set spid1 41;
set A 0.78;
set p1 [expr 2.539*$A]; set p2 [expr 3.005*$A]; set p3 [expr 3.163*$A]; set p4 [expr 0.6326*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1*$JointVolume] [expr $p2*$JointVolume] [expr $p3*$JointVolume] [expr $p4*$JointVolume]]
set nEnvStrsp [list [expr -$p1*$JointVolume] [expr -$p2*$JointVolume] [expr -$p3*$JointVolume] [expr -$p4*$JointVolume]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.0008 0.015  0.035 0.04]
set nEnvStnsp [list -0.0008 -0.015  -0.035 -0.04]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]
```

```


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $spid1 [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy

#################### beam column joint ###########################################

##element BeamColumnJoint tag? iNode? jNode? kNode? lNode? matTag1? matTag2? matTag3? matTag4?
## matTag5? matTag6? matTag7? matTag8? matTag9? matTag10? matTag11? matTag12? matTag13?
## <element Height factor?> <element Width factor?>
## please note: the four nodes are in anticlockwise direction around the element
## requires material tags for all 13 different components within the element.
## the first 12 being that of spring and the last of the shear panel

set jointA1 611
set jointA2 612
set jointA3 613
set jointA4 614
set jointA5 615
set jointA6 616

set jointB1 621
set jointB2 622
set jointB3 623
set jointB4 624
set jointB5 625
set jointB6 626

set jointC1 631
set jointC2 632
set jointC3 633
set jointC4 634
set jointC5 635
set jointC6 636

set jointD1 641
set jointD2 642
set jointD3 643
set jointD4 644
set jointD5 645
set jointD6 646

# add material Properties - command: uniaxialMaterial matType matTag ...
#command: uniaxialMaterial Elastic tag? E?

uniaxialMaterial Elastic 71 10000000000.0

element beamColumnJoint $jointA1 $N_A1 $N_A10_R $N_A10_A $N_A10_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1
```

```
$bsid2 71 $spid1
element beamColumnJoint $jointA2 $N_A2 $N_A20_R $N_A20_A $N_A20_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA3 $N_A3 $N_A30_R $N_A30_A $N_A30_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA4 $N_A4 $N_A40_R $N_A40_A $N_A40_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA5 $N_A5 $N_A50_R $N_A50_A $N_A50_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA6 $N_A6 $N_A60_R $N_A60_A $N_A60_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointB1 $N_B1 $N_B11_R $N_B11_A $N_B11_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB2 $N_B2 $N_B21_R $N_B21_A $N_B21_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB3 $N_B3 $N_B31_R $N_B31_A $N_B31_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB4 $N_B4 $N_B41_R $N_B41_A $N_B41_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB5 $N_B5 $N_B51_R $N_B51_A $N_B51_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB6 $N_B6 $N_B61_R $N_B61_A $N_B61_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointC1 $N_C1 $N_C12_R $N_C12_A $N_C12_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC2 $N_C2 $N_C22_R $N_C22_A $N_C22_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC3 $N_C3 $N_C32_R $N_C32_A $N_C32_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC4 $N_C4 $N_C42_R $N_C42_A $N_C42_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC5 $N_C5 $N_C52_R $N_C52_A $N_C52_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC6 $N_C6 $N_C62_R $N_C62_A $N_C62_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointD1 $N_D1 $N_D13_R $N_D13_A $N_D13_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD2 $N_D2 $N_D23_R $N_D23_A $N_D23_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD3 $N_D3 $N_D33_R $N_D33_A $N_D33_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD4 $N_D4 $N_D43_R $N_D43_A $N_D43_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD5 $N_D5 $N_D53_R $N_D53_A $N_D53_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD6 $N_D6 $N_D63_R $N_D63_A $N_D63_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1


########################################################################################################################
# Elements definitions
########################################################################################################################

# COLUMN definition

# ---------------------------------------------------------
# Define geometric transformation
# ---------------------------------------------------------
set ColTransfTag 1;        # associate a tag to column transformation
geomTransf PDelta $ColTransfTag ; #Columns

# ------------------------------------------------------------------
# ---- element connectivity "Columns Definition"----------
# ------------------------------------------------------------------
set numIntPoints 4;
set integrationC "Lobatto $col25x60 $numIntPoints"

element forceBeamColumn 710      $N_A0   $N_A1 $ColTransfTag $integrationC
element forceBeamColumn 720      $N_A10_A   $N_A2 $ColTransfTag $integrationC
element forceBeamColumn 730      $N_A20_A   $N_A3 $ColTransfTag $integrationC
element forceBeamColumn 740      $N_A30_A   $N_A4 $ColTransfTag $integrationC
element forceBeamColumn 750      $N_A40_A   $N_A5 $ColTransfTag $integrationC
element forceBeamColumn 760      $N_A50_A   $N_A6 $ColTransfTag $integrationC

element forceBeamColumn 711      $N_B0   $N_B1 $ColTransfTag $integrationC
element forceBeamColumn 721      $N_B11_A   $N_B2 $ColTransfTag $integrationC
element forceBeamColumn 731      $N_B21_A   $N_B3 $ColTransfTag $integrationC
```

```
element forceBeamColumn 741      $N_B31_A   $N_B4 $ColTransfTag $integrationC
element forceBeamColumn 751      $N_B41_A   $N_B5 $ColTransfTag $integrationC
element forceBeamColumn 761      $N_B51_A   $N_B6 $ColTransfTag $integrationC

element forceBeamColumn 712      $N_C0   $N_C1 $ColTransfTag $integrationC
element forceBeamColumn 722      $N_C12_A   $N_C2 $ColTransfTag $integrationC
element forceBeamColumn 732      $N_C22_A   $N_C3 $ColTransfTag $integrationC
element forceBeamColumn 742      $N_C32_A   $N_C4 $ColTransfTag $integrationC
element forceBeamColumn 752      $N_C42_A   $N_C5 $ColTransfTag $integrationC
element forceBeamColumn 762      $N_C52_A   $N_C6 $ColTransfTag $integrationC

element forceBeamColumn 713      $N_D0   $N_D1 $ColTransfTag $integrationC
element forceBeamColumn 723      $N_D13_A   $N_D2 $ColTransfTag $integrationC
element forceBeamColumn 733      $N_D23_A   $N_D3 $ColTransfTag $integrationC
element forceBeamColumn 743      $N_D33_A   $N_D4 $ColTransfTag $integrationC
element forceBeamColumn 753      $N_D43_A   $N_D5 $ColTransfTag $integrationC
element forceBeamColumn 763      $N_D53_A   $N_D6 $ColTransfTag $integrationC


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

# BEAMS definition

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
set BeamTransfTag 2;          # associate a tag to beam transformation
geomTransf PDelta $BeamTransfTag  ; #Beams

# -------------------------------------------------------------------
# ---- element connectivity "Beamss Definition"----------
# -------------------------------------------------------------------
set numIntPoints_beams 5;
set integrationB "Lobatto $beam150x30 $numIntPoints_beams"

element forceBeamColumn  810    $N_A10_R $N_B11_L  $BeamTransfTag $integrationB
element forceBeamColumn  820    $N_A20_R $N_B21_L  $BeamTransfTag $integrationB
element forceBeamColumn  830    $N_A30_R $N_B31_L  $BeamTransfTag $integrationB
element forceBeamColumn  840    $N_A40_R $N_B41_L  $BeamTransfTag $integrationB
element forceBeamColumn  850    $N_A50_R $N_B51_L  $BeamTransfTag $integrationB
element forceBeamColumn  860    $N_A60_R $N_B61_L  $BeamTransfTag $integrationB

element forceBeamColumn  811    $N_B11_R $N_C12_L  $BeamTransfTag $integrationB
element forceBeamColumn  821    $N_B21_R $N_C22_L  $BeamTransfTag $integrationB
element forceBeamColumn  831    $N_B31_R $N_C32_L  $BeamTransfTag $integrationB
element forceBeamColumn  841    $N_B41_R $N_C42_L  $BeamTransfTag $integrationB
element forceBeamColumn  851    $N_B51_R $N_C52_L  $BeamTransfTag $integrationB
element forceBeamColumn  861    $N_B61_R $N_C62_L  $BeamTransfTag $integrationB

element forceBeamColumn  812    $N_C12_R $N_D13_L  $BeamTransfTag $integrationB
element forceBeamColumn  822    $N_C22_R $N_D23_L  $BeamTransfTag $integrationB
element forceBeamColumn  832    $N_C32_R $N_D33_L  $BeamTransfTag $integrationB
element forceBeamColumn  842    $N_C42_R $N_D43_L  $BeamTransfTag $integrationB
element forceBeamColumn  852    $N_C52_R $N_D53_L  $BeamTransfTag $integrationB
element forceBeamColumn  862    $N_C62_R $N_D63_L  $BeamTransfTag $integrationB

##################################################################################################################################
# display the model with the node numbers
   DisplayModel2D NodeNumbers

##################################################################################################################################
# gravity and masses load
##################################################################################################################################

    # timeSeries  "LinearDefault":   tsTag   cFactor
    timeSeries          Linear        1     -factor   1;

    # distributed loads

   #set   DL   11000.0;             # self weight add as point load (N)
    set   TLE   64800.0;            # TLE: Total Load at the middle columns
    set   TLM   129600.0;            # TLM: Total Load at the middle columns

    # pattern  PatternType  $PatternID  TimeSeriesType
    pattern       Plain          1             1       {
    #load  $nodeTag  (ndf  $LoadValues)
    load   $N_A10_A 0    [expr -$TLE]    0;
```

```
    load   $N_A20_A 0    [expr -$TLE]    0;
    load   $N_A30_A 0    [expr -$TLE]    0;
    load   $N_A40_A 0    [expr -$TLE]    0;
    load   $N_A50_A 0    [expr -$TLE]    0;
    load   $N_A60_A 0    [expr -$TLE]    0;

    load   $N_B11_A 0    [expr -$TLM]    0;
    load   $N_B21_A 0    [expr -$TLM]    0;
    load   $N_B31_A 0    [expr -$TLM]    0;
    load   $N_B41_A 0    [expr -$TLM]    0;
    load   $N_B51_A 0    [expr -$TLM]    0;
    load   $N_B61_A 0    [expr -$TLM]    0;

    load   $N_C12_A 0    [expr -$TLM]    0;
    load   $N_C22_A 0    [expr -$TLM]    0;
    load   $N_C32_A 0    [expr -$TLM]    0;
    load   $N_C42_A 0    [expr -$TLM]    0;
    load   $N_C52_A 0    [expr -$TLM]    0;
    load   $N_C62_A 0    [expr -$TLM]    0;

    load   $N_D13_A 0    [expr -$TLE]    0;
    load   $N_D23_A 0    [expr -$TLE]    0;
    load   $N_D33_A 0    [expr -$TLE]    0;
    load   $N_D43_A 0    [expr -$TLE]    0;
    load   $N_D53_A 0    [expr -$TLE]    0;
    load   $N_D63_A 0    [expr -$TLE]    0;

    #eleLoad  -ele  $eleTag1  <$eleTag2>  -type  -beamuniformload $wy
    #eleLoad   -ele     5           6       -type  -beamUniform     [expr -$DL];

    }


# masses

      set mass1  19440;
      set mass2  19440;
      set mass3  19440;
      set mass4  19440;
      set mass5  19440;
      set mass6  19440;



     # assign mass to nodes

     #mass  $nodetag  (ndf  $massvalues)
      mass   $N_A10_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A20_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A30_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A40_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A50_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A60_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_B11_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B21_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B31_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B41_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B51_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B61_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_C12_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C22_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C32_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C42_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C52_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C62_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_D13_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D23_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D33_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D43_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D53_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D63_L    [expr $mass1/2]  0.1  0.1;



puts "Model Built"
```

3) Gravity Analysis Procedure:

```
################################################################################

# start analysis

initialize

puts "ooo Analysis: Gravity ooo"

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

# set recorders

#   Node Recorder "Reactions":  fileName                <nodeTag>             dof   resptype
recorder Node -file $dataDir/RBase.out -time -node  $N_A0  $N_B0 $N_C0  $N_D0  -dof    2   reaction;

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

# analysis options


#  Constraint Handler
constraints  Plain

# DOF numberer
numberer  RCM

# System of Equations
system  ProfileSPD

# Convergence Test
test  NormDispIncr    1.00000E-006    100    0   2;

# Solution Algorithm
algorithm Newton

# Integrator
#integrator LoadControl $Lambda <$numIter $minLambda $maxLambda>
integrator  LoadControl 0.01

# Analysis Type
analysis Static

# Record initial state of model
record

#Analysis model
analyze   100

# Reset for next analysis case
setTime 0.0
loadConst
remove recorders
wipeAnalysis

puts "Gravity analysis completed"

```

4) Modal Analysis Procedure:

```
################################################################################

# start analysis

initialize

puts "ooo Analysis: ModalAnalysis ooo"

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

# set recorders

#   Node Recorder "EigenVectors":  fileName                <nodeTag>             dof   resptype
recorder Node -file $dataDir/ModalAnalysis_Node_EigenVectors_EigenVec1.out       -node  $N_A0  $N_A10_A $N_A20_A $N_A30_A $N_A40_A $N_A50_A $N_A60_A  -dof  1  eigen1
recorder Node -file $dataDir/ModalAnalysis_Node_EigenVectors_EigenVec2.out       -node  $N_A0  $N_A10_A $N_A20_A $N_A30_A $N_A40_A $N_A50_A $N_A60_A  -dof  1  eigen2

################################################################################
```

```

# analysis options


#  Constraint Handler
constraints  Transformation

# DOF numberer
numberer  Plain

# System of Equations
system  BandGeneral

# Convergence Test
test  NormDispIncr    1.00000E-5    50    0   2;

# Solution Algorithm
algorithm Newton

# Integrator
integrator  Newmark   5.0000000E-01    2.5000000E-01

# Analysis Type
analysis Transient

# Analysis model (and record responce)
      set pi [expr 2.0*asin(1.0)];                       # Definition of pi
   set nEigenI 1;                            # mode i = 1
   set nEigenJ 2;                            # mode j = 2
   set lambdaN [eigen [expr $nEigenJ]];            # eigenvalue analysis for nEigenJ modes
   set lambdaI [lindex $lambdaN [expr 0]];         # eigenvalue mode i = 1
   set lambdaJ [lindex $lambdaN [expr $nEigenJ-1]];   # eigenvalue mode j = 2
   set w1 [expr pow(($lambdaI*1000),0.5)];                # w1 (1st mode circular frequency)
   set w2 [expr pow(($lambdaJ*1000),0.5)];                # w2 (2nd mode circular frequency)
   set T1 [expr 2.0*$pi/$w1];                   # 1st mode period of the structure
   set T2 [expr 2.0*$pi/$w2];                   # 2nd mode period of the structure

puts "T1 is $T1"
puts "T2 is $T2"
# Record eigenvectors
record


# Reset for next analysis case
setTime 0.0
loadConst
remove recorders
wipeAnalysis

puts "Modal analysis completed"

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>             dof   resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A50_L  $N_A60_L   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A10_L  $N_A20_L -dof 1 2 disp;               # displacements of free nodes
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
recorder Element -file $dataDir/ele1sec1StressStrain.out -time -ele 710 section 1 fiber 0 0 $C_confined stressStrain
#recorder  Element   -file $dataDir/force10c.out    -time    -ele  710    section 1  fiber $R_steel 0. $C_unconfined stressStrain;
#recorder  Element   -file $dataDir/force60Bc.out    -time    -ele  860    section 1  fiber y z $C_unconfined stressStrain;

recorder  Element   -file $dataDir/force10B.out    -time    -ele 810 811 812 820 821 822 830 831 832 840 841 842 850 851 852
```

```
 860 861 862 localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 621 622 623 624 625 626 shearpanel stressStrain



# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
   set ViewScale 5;
   DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1800;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.01;      # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 1;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;         # define load pattern -- generalized
      load $N_A60_L 6 0 0
        load $N_A50_L 5 0 0
        load $N_A40_L 4 0 0
        load $N_A30_L 3 0 0
        load $N_A20_L 2 0 0
        load $N_A10_L 1 0 0


   }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;     # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;    # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
```

```
#          RelativeEnergyIncr --
variable TolStatic 0.01;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;   # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A60_L   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";   # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
   # if analysis fails, we try some other stuff, performance is slower inside this loop
   set Dstep 0.0;
   set ok 0
   while {$Dstep <= 1.0 && $ok == 0} {
      set controlDisp [nodeDisp $N_A60_L 1 ]
      set Dstep [expr $controlDisp/$Dmax]
      set ok [analyze 1 ]
      # if analysis fails, we try some other stuff
      # performance is slower inside this loop  global maxNumIterStatic;       # max no. of iterations performed before "failure to converge" is ret'd
      if {$ok != 0} {
         puts "Trying Newton with Initial Tangent .."
         test NormDispIncr   $Tol 3000 0
         algorithm Newton -initial
         set ok [analyze 1]
         test $testTypeStatic $TolStatic      $maxNumIterStatic    0
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying Broyden .."
         algorithm Broyden 8
         set ok [analyze 1 ]
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying NewtonWithLineSearch .."
         algorithm NewtonLineSearch 0.8
         set ok [analyze 1]
```

```
         algorithm $algorithmTypeStatic
      }

   }; # end while loop
};      # end if ok !0

# -----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
   puts [format $fmt1 "PROBLEM" $N_A60_L 1 [nodeDisp $N_A60_L 1] "mm"]
} else {
   puts [format $fmt1 "DONE"  $N_A60_L 1 [nodeDisp $N_A60_L 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 4 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B6S MRFs- Non-Ductile Bare Frame

Appendix 4: 3B6S Bare Frame with Structural Deficiencies

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B6S Bare Frame with structural deficiencies:

The code generated is the same as Appendix 3. However, some changes were applied to represent structural deficiencies

  1. Consider the effect of stirrups spacing

#   basic parameters for materials-con-concrete

# ConfinedConcrete01 Material

      #$tag           integer tag identifying material.
      #$secType       tag for the transverse reinforcement configuration.
      #$fpc           unconfined cylindrical strength of concrete specimen.
      #$Ec            initial elastic modulus of unconfined concrete.
      #<-epscu $epscu> OR <-gamma $gamma> confined concrete ultimate strain.
      #<-nu $nu> OR <-varub> OR <-varnoub>  Poisson's Ratio.
      #$L1            length/diameter of square/circular core section measured respect to the hoop center line.
      #($L2)          additional dimensions when multiple hoops are being used.
      #$phis        hoop diameter. If section arrangement has multiple hoops it refers to the external hoop.
      #$S           hoop spacing.
      #$fyh           yielding strength of the hoop steel.
      #$Es0           elastic modulus of the hoop steel.
      #$haRatio       hardening ratio of the hoop steel.
      #$mu            ductility factor of the hoop steel.
      #$phiLon        diameter of longitudinal bars.

#   basic parameters for materials-uncon-concrete

       set unconfc   -28.0;        # compression strength for concrete
       set unconepsc  -0.002;                 # strain at maximum stress in compression
       set unconfu   [expr $unconfc*0.18];          # ultamate stress for concrete
       set unconepsu  -0.01;                  # strain at ultimate stress in compression
       set unconlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set unconft   [expr $unconfc*-0.1];          # maximum stress in tension for concrete
       set unconEt   [expr $unconft/0.002];         # elastic modulus in tension
       set unconE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

#   basic parameters for material--steel        # ReinforcingSteel uniaxial material object. This object is intended to be used in a reinforced concrete fiber section as the steel reinforcing material.

       set Fy 420.0;     # Yield stress in tension
       set Fu 596.0;     # Ultimate stress in tension
       set Es 200000.0;   # Initial elastic tangent
       set Esh 3100.0;    # Tangent at initial strain hardening
       set esh 0.01;      # Strain corresponding to initial strain hardening
       set eult 0.09;     # Strain at peak stress

   #uniaxialMaterial ReinforcingSteel $matTag $fy $fu $Es $Esh $esh $eult    Define ReinforcingSteel uniaxial material
   uniaxialMaterial ReinforcingSteel $R_steel $Fy $Fu $Es $Esh $esh $eult -DMBuck 6 0.8 -CMFatigue 0.2600 0.5000 0.3890 -IsoHard 4.3000 0.01

#   definition of ConfinedConcrete01 material

```

```
    #uniaxialMaterial ConfinedConcrete01 $tag        $secType $fpc  $Ec   -epscu $epscu   $nu    $L1   $L2    $phis  $S   $fyh  $Es0     $haRatio  $mu  $phiLon  -stRatio $stRatio
    #uniaxialMaterial ConfinedConcrete01  $C_confinedB    R     -28 24870.1 -epscu -0.04   -varUB 250.0 1450.0 10.0  125.0 420.0 200000.0 0.00      3100.0 12.0    -stRatio 0.85
    #uniaxialMaterial ConfinedConcrete01  $C_confinedC    R     -28 24870.1 -epscu -0.04   -varUB 550.0 200.0  10.0  125.0 420.0 200000.0 0.00      3100.0 18.0    -stRatio 0.85


#   basic parameters for materials-con-concrete

       set confc   -28;        # compression strength for concrete
       set conepsc  -0.003;                 # strain at maximum stress in compression
       set confu   [expr $unconfc*0.18];           # ultamate stress for concrete
       set conepsu  -0.04;                  # strain at ultimate stress in compression
       set conlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set conft   [expr $unconfc*-0.1];          # maximum stress in tension for concrete
       set conEt   [expr $unconft/0.002];         # elastic modulus in tension
       set conE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

    # uniaxialMaterial Concrete02  $matTag     $fpc $epsc0 $fpcu  $epsU  $lambda  $ft  $Ets
    uniaxialMaterial   Concrete02  $C_unconfined $unconfc  $unconepsc  $unconfu    $unconepsu  $unconlambda  $unconft  $unconEt;
    uniaxialMaterial   Concrete02  $C_confined $confc  $conepsc  $confu    $conepsu  $conlambda  $conft  $conEt;

  2. Consider Development Length

######################### details for the material models of bar slip of the beam ####################################

set bs_fc 28.0; set bs_fs 420.0; set bs_es 200000; set bs_fsu 596; set bs_dbar 12.0; set bs_esh 3100.0;
set bs_wid $w_col; set bs_dep $h_beam;
set bsT_nbars 12; set bsB_nbars 8;
set bs_ljoint 50;

  3. Beam Column Joint Properties

  ############################# material for shear panel #############################################################

## Positive/Negative envelope Stress

set spid1 41;
set A 0.78;
set p1 [expr 1.899*$A]; set p2 [expr 2.466*$A]; set p3 [expr 2.596*$A]; set p4 [expr 0.5192*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1*$JointVolume] [expr $p2*$JointVolume] [expr $p3*$JointVolume] [expr $p4*$JointVolume]]
set nEnvStrsp [list [expr -$p1*$JointVolume] [expr -$p2*$JointVolume] [expr -$p3*$JointVolume] [expr -$p4*$JointVolume]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.00043 0.006  0.015 0.04]
set nEnvStnsp [list -0.00043 -0.006  -0.015 -0.04]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

```

```
#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $spid1 [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy


3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 3

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>             dof   resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A50_L  $N_A60_L   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A10_L  $N_A20_L -dof 1 2 disp;         # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
#recorder  Element   -file $dataDir/force10.out    -time    -ele  710    section 1  fiber y z $R_steel  stressStrain;
#recorder  Element   -file $dataDir/force60B.out    -time    -ele  860    section 1  fiber y z $R_steel stressStrain;
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860  localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 621 622 623 624 625 626 shearpanel stressStrain
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
  set ViewScale 5;
  DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1800;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.01;   # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 1;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;     # define load pattern -- generalized
```

```
    load $N_A60_L 6 0 0
        load $N_A50_L 5 0 0
        load $N_A40_L 4 0 0
        load $N_A30_L 3 0 0
        load $N_A20_L 2 0 0
        load $N_A10_L 1 0 0


  }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;    # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.

#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;   # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 0.01;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;  # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
```

```
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A60_L   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];               # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";  # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
  # if analysis fails, we try some other stuff, performance is slower inside this loop
  set Dstep 0.0;
  set ok 0
  while {$Dstep <= 1.0 && $ok == 0} {
    set controlDisp [nodeDisp $N_A60_L 1 ]
    set Dstep [expr $controlDisp/$Dmax]
    set ok [analyze 1 ]
    # if analysis fails, we try some other stuff
    # performance is slower inside this loop  global maxNumIterStatic;      # max no. of iterations performed before "failure to converge" is ret'd
    if {$ok != 0} {
      puts "Trying Newton with Initial Tangent .."
      test NormDispIncr   $Tol 3000 0
      algorithm Newton -initial
      set ok [analyze 1]
      test $testTypeStatic $TolStatic      $maxNumIterStatic    0
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying Broyden .."
      algorithm Broyden 8
      set ok [analyze 1 ]
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying NewtonWithLineSearch .."
      algorithm NewtonLineSearch 0.8
      set ok [analyze 1]
      algorithm $algorithmTypeStatic
    }

  };  # end while loop
};      # end if ok !0

# ----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
  puts [format $fmt1 "PROBLEM" $N_A60_L 1 [nodeDisp $N_A60_L 1] "mm"]
} else {
  puts [format $fmt1 "DONE"  $N_A60_L 1 [nodeDisp $N_A60_L 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 5 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B6S MRFs- Masonry-Concrete Infilled Frames

Appendix 5: 3B6S Masonry infilled Frame

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3

   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.

2) 2D Model Definition for 3B6S MRFs-Masonry-Concrete Infilled Frame :

```
#performing nonlinear static pushover analysis on 3B6S MRFs-Masonry-Concrete Infilled Frame
##############################################################################################################################

   wipe all;
# define model builder
#    model basic builder  -ndm $ndm <-ndf $ndf>
     model basic builder  -ndm 2  -ndf 3

     set dataDir Results;           # set up name of data directory
     file mkdir $dataDir;           # create data directory
     source Libunits.tcl;               # define basic system units
     source DisplayModel2D.tcl;        # procedure for displaying a 2D View of model
     source DisplayPlane.tcl;       # procedure for displaying a plane in a model

###############################################################################################################################
# buiding geometry
###############################################################################################################################


# dimensions

    set span1 4000.0;
    set span2 4000.0;
    set span3 4000.0;
    set storey1 3000.0;
    set storey2 3000.0;
    set storey3 3000.0;
    set storey4 3000.0;
    set storey5 3000.0;
    set storey6 3000.0;

# main grid lines
    # vertical axis, x
    set x1 [expr 0];
    set x2 [expr $x1+$span1];
    set x3 [expr $x2+$span2];
    set x4 [expr $x3+$span3];

    # hoeizontal axis, y
    set z0 [expr 0];
    set z1 [expr $z0+$storey1];
    set z2 [expr $z1+$storey2];
    set z3 [expr $z2+$storey3];
    set z4 [expr $z3+$storey4];
    set z5 [expr $z4+$storey5];
    set z6 [expr $z5+$storey6];

# definition of nodes

```

```
#assigning node tages                    # for axises A,B,C, and D.
set N_A0     1;
set N_B0     2;
set N_C0     3;
set N_D0     4;
set N_A1     5;
set N_B1     6;
set N_C1     7;
set N_D1     8;
set N_A2     9;
set N_B2     10;
set N_C2     11;
set N_D2     12;
set N_A3     13;
set N_B3     14;
set N_C3     15;
set N_D3     16;
set N_A4     17;
set N_B4     18;
set N_C4     19;
set N_D4     20;
set N_A5     21;
set N_B5     22;
set N_C5     23;
set N_D5     24;
set N_A6     25;
set N_B6     26;
set N_C6     27;
set N_D6     28;

set N_A10_R   29;                          # N_Aij_R    i: story level.   j: axis number
set N_A10_A   30;
set N_A10_L   31;
set N_A20_R   32;
set N_A20_A   33;
set N_A20_L   34;
set N_A30_R   35;
set N_A30_A   36;
set N_A30_L   37;
set N_A40_R   38;
set N_A40_A   39;
set N_A40_L   40;
set N_A50_R   41;
set N_A50_A   42;
set N_A50_L   43;
set N_A60_R   44;
set N_A60_A   45;
set N_A60_L   46;

set N_B11_R   47;
set N_B11_A   48;
set N_B11_L   49;
set N_B21_R   50;
set N_B21_A   51;
set N_B21_L   52;
set N_B31_R   53;
set N_B31_A   54;
set N_B31_L   55;
set N_B41_R   56;
set N_B41_A   57;
set N_B41_L   58;
set N_B51_R   59;
set N_B51_A   60;
set N_B51_L   61;
set N_B61_R   62;
set N_B61_A   63;
set N_B61_L   64;

set N_C12_R   65;
set N_C12_A   66;
set N_C12_L   67;
set N_C22_R   68;
set N_C22_A   69;
set N_C22_L   70;
set N_C32_R   71;
set N_C32_A   72;
set N_C32_L   73;
set N_C42_R   74;
set N_C42_A   75;
set N_C42_L   76;
```

```
set N_C52_R   77;
set N_C52_A   78;
set N_C52_L   79;
set N_C62_R   80;
set N_C62_A   81;
set N_C62_L   82;

set N_D13_R   83;
set N_D13_A   84;
set N_D13_L   85;
set N_D23_R   86;
set N_D23_A   87;
set N_D23_L   88;
set N_D33_R   89;
set N_D33_A   90;
set N_D33_L   91;
set N_D43_R   92;
set N_D43_A   93;
set N_D43_L   94;
set N_D53_R   95;
set N_D53_A   96;
set N_D53_L   97;
set N_D63_R   98;
set N_D63_A   99;
set N_D63_L   100;

#infill wall nodes
set N_W1A_L   3001;                          # l: means the node at the left side of the panel
set N_W1A_R   3002;                          # R: means the node at the right side of the panel
set N_W2A_L   3003;
set N_W2A_R   3004;
set N_W3A_L   3005;
set N_W3A_R   3006;
set N_W4A_L   3007;
set N_W4A_R   3008;
set N_W5A_L   3009;
set N_W5A_R   3010;
set N_W6A_L   3011;
set N_W6A_R   3012;

set N_W1B_L   3013;
set N_W1B_R   3014;
set N_W2B_L   3015;
set N_W2B_R   3016;
set N_W3B_L   3017;
set N_W3B_R   3018;
set N_W4B_L   3019;
set N_W4B_R   3020;
set N_W5B_L   3021;
set N_W5B_R   3022;
set N_W6B_L   3023;
set N_W6B_R   3024;

set N_W1C_L   3025;
set N_W1C_R   3026;
set N_W2C_L   3027;
set N_W2C_R   3028;
set N_W3C_L   3029;
set N_W3C_R   3030;
set N_W4C_L   3031;
set N_W4C_R   3032;
set N_W5C_L   3033;
set N_W5C_R   3034;
set N_W6C_L   3035;
set N_W6C_R   3036;


#node $nodetag (ndm $coords) <-mass (ndf $massvalues)>

set col_halfdepA [expr 600/2];              # This is used to define the joint dimensions.
set col_halfdepB [expr 600/2];
set col_halfdepC [expr 600/2];
set col_halfdepD [expr 600/2];
set beam_halfdep1 [expr 300/2];
set beam_halfdep2 [expr 300/2];
set beam_halfdep3 [expr 300/2];
set beam_halfdep4 [expr 300/2];
set beam_halfdep5 [expr 300/2];
set beam_halfdep6 [expr 300/2];

```

```
      node  $N_A0      $x1 $z0;
      node  $N_B0      $x2 $z0;
      node  $N_C0      $x3 $z0;
      node  $N_D0      $x4 $z0;
      node  $N_A1      $x1 [expr $z1-$beam_halfdep1];
      node  $N_B1      $x2 [expr $z1-$beam_halfdep1];
      node  $N_C1      $x3 [expr $z1-$beam_halfdep1];
      node  $N_D1      $x4 [expr $z1-$beam_halfdep1];
      node  $N_A2      $x1 [expr $z2-$beam_halfdep2];
      node  $N_B2      $x2 [expr $z2-$beam_halfdep2];
      node  $N_C2      $x3 [expr $z2-$beam_halfdep2];
      node  $N_D2      $x4 [expr $z2-$beam_halfdep2];
      node  $N_A3      $x1 [expr $z3-$beam_halfdep3];
      node  $N_B3      $x2 [expr $z3-$beam_halfdep3];
      node  $N_C3      $x3 [expr $z3-$beam_halfdep3];
      node  $N_D3      $x4 [expr $z3-$beam_halfdep3];
      node  $N_A4      $x1 [expr $z4-$beam_halfdep4];
      node  $N_B4      $x2 [expr $z4-$beam_halfdep4];
      node  $N_C4      $x3 [expr $z4-$beam_halfdep4];
      node  $N_D4      $x4 [expr $z4-$beam_halfdep4];
      node  $N_A5      $x1 [expr $z5-$beam_halfdep5];
      node  $N_B5      $x2 [expr $z5-$beam_halfdep5];
      node  $N_C5      $x3 [expr $z5-$beam_halfdep5];
      node  $N_D5      $x4 [expr $z5-$beam_halfdep5];
      node  $N_A6      $x1 [expr $z6-$beam_halfdep6];
      node  $N_B6      $x2 [expr $z6-$beam_halfdep6];
      node  $N_C6      $x3 [expr $z6-$beam_halfdep6];
      node  $N_D6      $x4 [expr $z6-$beam_halfdep6];


############### add nodes - joints #######################################################

                                                   #  R: node at the right side of joint
                                                   #  A: node above the joint
                                                   #  L: node at the left side of the joint
      node $N_A10_R   [expr $x1+$col_halfdepA] $z1;
      node $N_A10_A   $x1 [expr $z1+$beam_halfdep1];
      node $N_A10_L   [expr $x1-$col_halfdepA] $z1;
      node $N_A20_R   [expr $x1+$col_halfdepA] $z2;
      node $N_A20_A   $x1 [expr $z2+$beam_halfdep2];
      node $N_A20_L   [expr $x1-$col_halfdepA] $z2;
      node $N_A30_R   [expr $x1+$col_halfdepA] $z3;
      node $N_A30_A   $x1 [expr $z3+$beam_halfdep3];
      node $N_A30_L   [expr $x1-$col_halfdepA] $z3;
      node $N_A40_R   [expr $x1+$col_halfdepA] $z4;
      node $N_A40_A   $x1 [expr $z4+$beam_halfdep4];
      node $N_A40_L   [expr $x1-$col_halfdepA] $z4;
      node $N_A50_R   [expr $x1+$col_halfdepA] $z5;
      node $N_A50_A   $x1 [expr $z5+$beam_halfdep5];
      node $N_A50_L   [expr $x1-$col_halfdepA] $z5;
      node $N_A60_R   [expr $x1+$col_halfdepA] $z6;
      node $N_A60_A   $x1 [expr $z6+$beam_halfdep6];
      node $N_A60_L   [expr $x1-$col_halfdepA] $z6;

      node $N_B11_R   [expr $x2+$col_halfdepB] $z1;
      node $N_B11_A   $x2 [expr $z1+$beam_halfdep1];
      node $N_B11_L   [expr $x2-$col_halfdepB] $z1;
      node $N_B21_R   [expr $x2+$col_halfdepB] $z2;
      node $N_B21_A   $x2 [expr $z2+$beam_halfdep2];
      node $N_B21_L   [expr $x2-$col_halfdepB] $z2;
      node $N_B31_R   [expr $x2+$col_halfdepB] $z3;
      node $N_B31_A   $x2 [expr $z3+$beam_halfdep3];
      node $N_B31_L   [expr $x2-$col_halfdepB] $z3;
      node $N_B41_R   [expr $x2+$col_halfdepB] $z4;
      node $N_B41_A   $x2 [expr $z4+$beam_halfdep4];
      node $N_B41_L   [expr $x2-$col_halfdepB] $z4;
      node $N_B51_R   [expr $x2+$col_halfdepB] $z5;
      node $N_B51_A   $x2 [expr $z5+$beam_halfdep5];
      node $N_B51_L   [expr $x2-$col_halfdepB] $z5;
      node $N_B61_R   [expr $x2+$col_halfdepB] $z6;
      node $N_B61_A   $x2 [expr $z6+$beam_halfdep6];
      node $N_B61_L   [expr $x2-$col_halfdepB] $z6;

      node $N_C12_R   [expr $x3+$col_halfdepC] $z1;
      node $N_C12_A   $x3 [expr $z1+$beam_halfdep1];
      node $N_C12_L   [expr $x3-$col_halfdepC] $z1;
      node $N_C22_R   [expr $x3+$col_halfdepC] $z2;
      node $N_C22_A   $x3 [expr $z2+$beam_halfdep2];
      node $N_C22_L   [expr $x3-$col_halfdepC] $z2;
      node $N_C32_R   [expr $x3+$col_halfdepC] $z3;
```

```
      node $N_C32_A   $x3 [expr $z3+$beam_halfdep3];
      node $N_C32_L   [expr $x3-$col_halfdepC] $z3;
      node $N_C42_R   [expr $x3+$col_halfdepC] $z4;
      node $N_C42_A   $x3 [expr $z4+$beam_halfdep4];
      node $N_C42_L   [expr $x3-$col_halfdepC] $z4;
      node $N_C52_R   [expr $x3+$col_halfdepC] $z5;
      node $N_C52_A   $x3 [expr $z5+$beam_halfdep5];
      node $N_C52_L   [expr $x3-$col_halfdepC] $z5;
      node $N_C62_R   [expr $x3+$col_halfdepC] $z6;
      node $N_C62_A   $x3 [expr $z6+$beam_halfdep6];
      node $N_C62_L   [expr $x3-$col_halfdepC] $z6;

      node $N_D13_R   [expr $x4+$col_halfdepD] $z1;
      node $N_D13_A   $x4 [expr $z1+$beam_halfdep1];
      node $N_D13_L   [expr $x4-$col_halfdepD] $z1;
      node $N_D23_R   [expr $x4+$col_halfdepD] $z2;
      node $N_D23_A   $x4 [expr $z2+$beam_halfdep2];
      node $N_D23_L   [expr $x4-$col_halfdepD] $z2;
      node $N_D33_R   [expr $x4+$col_halfdepD] $z3;
      node $N_D33_A   $x4 [expr $z3+$beam_halfdep3];
      node $N_D33_L   [expr $x4-$col_halfdepD] $z3;
      node $N_D43_R   [expr $x4+$col_halfdepD] $z4;
      node $N_D43_A   $x4 [expr $z4+$beam_halfdep4];
      node $N_D43_L   [expr $x4-$col_halfdepD] $z4;
      node $N_D53_R   [expr $x4+$col_halfdepD] $z5;
      node $N_D53_A   $x4 [expr $z5+$beam_halfdep5];
      node $N_D53_L   [expr $x4-$col_halfdepD] $z5;
      node $N_D63_R   [expr $x4+$col_halfdepD] $z6;
      node $N_D63_A   $x4 [expr $z6+$beam_halfdep6];
      node $N_D63_L   [expr $x4-$col_halfdepD] $z6;


############### add nodes - infill walls ########################################################

       node $N_W1A_R [expr $x2*0.54] [expr $z0+($storey1*0.5)];
       node $N_W1A_L [expr $x2*0.46] [expr $z0+($storey1*0.5)];
       node $N_W2A_R [expr $x2*0.54] [expr $z1+($storey1*0.5)];
       node $N_W2A_L [expr $x2*0.46] [expr $z1+($storey1*0.5)];
       node $N_W3A_R [expr $x2*0.54] [expr $z2+($storey1*0.5)];
       node $N_W3A_L [expr $x2*0.46] [expr $z2+($storey1*0.5)];
       node $N_W4A_R [expr $x2*0.54] [expr $z3+($storey1*0.5)];
       node $N_W4A_L [expr $x2*0.46] [expr $z3+($storey1*0.5)];
       node $N_W5A_R [expr $x2*0.54] [expr $z4+($storey1*0.5)];
       node $N_W5A_L [expr $x2*0.46] [expr $z4+($storey1*0.5)];
       node $N_W6A_R [expr $x2*0.54] [expr $z5+($storey1*0.5)];
       node $N_W6A_L [expr $x2*0.46] [expr $z5+($storey1*0.5)];

       node $N_W1B_R [expr $x2+($span2*0.54)] [expr $z0+($storey1*0.5)];
       node $N_W1B_L [expr $x2+($span2*0.46)] [expr $z0+($storey1*0.5)];
       node $N_W2B_R [expr $x2+($span2*0.54)] [expr $z1+($storey1*0.5)];
       node $N_W2B_L [expr $x2+($span2*0.46)] [expr $z1+($storey1*0.5)];
       node $N_W3B_R [expr $x2+($span2*0.54)] [expr $z2+($storey1*0.5)];
       node $N_W3B_L [expr $x2+($span2*0.46)] [expr $z2+($storey1*0.5)];
       node $N_W4B_R [expr $x2+($span2*0.54)] [expr $z3+($storey1*0.5)];
       node $N_W4B_L [expr $x2+($span2*0.46)] [expr $z3+($storey1*0.5)];
       node $N_W5B_R [expr $x2+($span2*0.54)] [expr $z4+($storey1*0.5)];
       node $N_W5B_L [expr $x2+($span2*0.46)] [expr $z4+($storey1*0.5)];
       node $N_W6B_R [expr $x2+($span2*0.54)] [expr $z5+($storey1*0.5)];
       node $N_W6B_L [expr $x2+($span2*0.46)] [expr $z5+($storey1*0.5)];

       node $N_W1C_R [expr $x3+($span2*0.54)] [expr $z0+($storey1*0.5)];
       node $N_W1C_L [expr $x3+($span2*0.46)] [expr $z0+($storey1*0.5)];
       node $N_W2C_R [expr $x3+($span2*0.54)] [expr $z1+($storey1*0.5)];
       node $N_W2C_L [expr $x3+($span2*0.46)] [expr $z1+($storey1*0.5)];
       node $N_W3C_R [expr $x3+($span2*0.54)] [expr $z2+($storey1*0.5)];
       node $N_W3C_L [expr $x3+($span2*0.46)] [expr $z2+($storey1*0.5)];
       node $N_W4C_R [expr $x3+($span2*0.54)] [expr $z3+($storey1*0.5)];
       node $N_W4C_L [expr $x3+($span2*0.46)] [expr $z3+($storey1*0.5)];
       node $N_W5C_R [expr $x3+($span2*0.54)] [expr $z4+($storey1*0.5)];
       node $N_W5C_L [expr $x3+($span2*0.46)] [expr $z4+($storey1*0.5)];
       node $N_W6C_R [expr $x3+($span2*0.54)] [expr $z5+($storey1*0.5)];
       node $N_W6C_L [expr $x3+($span2*0.46)] [expr $z5+($storey1*0.5)];



# restraints

      #basefix  $nodetag  (ndf  $constraints)
      fix       $N_A0           1 1 1;
      fix       $N_B0           1 1 1;
```

```
      fix       $N_C0           1 1 1;
      fix       $N_D0           1 1 1;


##############################################################################################################################
# material definitions
##############################################################################################################################

#   Definition of materials IDs

       #set C_confinedB    1;
       set C_confined    1;
       set C_unconfined  2;
       set R_steel       3;
       set C_unconfinedw  4;



#   basic parameters for materials-con-concrete

# ConfinedConcrete01 Material

      #$tag           integer tag identifying material.
      #$secType         tag for the transverse reinforcement configuration.
      #$fpc           unconfined cylindrical strength of concrete specimen.
      #$Ec             initial elastic modulus of unconfined concrete.
      #<-epscu $epscu> OR <-gamma $gamma> confined concrete ultimate strain.
      #<-nu $nu> OR <-varub> OR <-varnoub>   Poisson's Ratio.
      #$L1            length/diameter of square/circular core section measured respect to the hoop center line.
      #($L2)            additional dimensions when multiple hoops are being used.
      #$phis         hoop diameter. If section arrangement has multiple hoops it refers to the external hoop.
      #$S             hoop spacing.
      #$fyh           yielding strength of the hoop steel.
      #$Es0             elastic modulus of the hoop steel.
      #$haRatio         hardening ratio of the hoop steel.
      #$mu            ductility factor of the hoop steel.
      #$phiLon       diameter of longitudinal bars.

#   basic parameters for materials-uncon-concrete

       set unconfc   -28.0;        # compression strength for concrete
       set unconepsc  -0.002;                 # strain at maximum stress in compression
       set unconfu   [expr $unconfc*0.18];           # ultamate stress for concrete
       set unconepsu  -0.01;                  # strain at ultimate stress in compression
       set unconlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set unconft   [expr $unconfc*-0.1];         # maximum stress in tension for concrete
       set unconEt   [expr $unconft/0.002];        # elastic modulus in tension
       set unconE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

#   basic parameters for material--steel        # ReinforcingSteel uniaxial material object. This object is intended to be used in a reinforced concrete fiber section as the steel reinforcing material.

       set Fy 420.0;    # Yield stress in tension
       set Fu 596.0;    # Ultimate stress in tension
       set Es 200000.0;    # Initial elastic tangent
       set Esh 3100.0;     # Tangent at initial strain hardening
       set esh 0.01;       # Strain corresponding to initial strain hardening
       set eult 0.09;         # Strain at peak stress

   #uniaxialMaterial ReinforcingSteel $matTag $fy $fu $Es $Esh $esh $eult    Define ReinforcingSteel uniaxial material
   uniaxialMaterial ReinforcingSteel $R_steel $Fy $Fu $Es $Esh $esh $eult -DMBuck 6 0.8 -CMFatigue 0.2600 0.5000 0.3890 -IsoHard 4.3000 0.01

#   definition of ConfinedConcrete01 material

    #uniaxialMaterial ConfinedConcrete01 $tag        $secType $fpc  $Ec   -epscu $epscu   $nu    $L1   $L2    $phis  $S   $fyh  $Es0     $haRatio  $mu  $phiLon  -stRatio $stRatio
    #uniaxialMaterial ConfinedConcrete01  $C_confinedB     R      -28 24870.1 -epscu -0.04   -varUB 250.0 1450.0 10.0  125.0 420.0 200000.0 0.00      3100.0 12.0    -stRatio 0.85
    #uniaxialMaterial ConfinedConcrete01  $C_confinedC     R      -28 24870.1 -epscu -0.04   -varUB 550.0 200.0  10.0  125.0 420.0 200000.0 0.00      3100.0 18.0    -stRatio 0.85

#   basic parameters for materials-con-concrete

       set confc   -32.5;         # compression strength for concrete
       set conepsc  -0.003;                 # strain at maximum stress in compression
       set confu   [expr $unconfc*0.18];           # ultamate stress for concrete
       set conepsu  -0.04;                  # strain at ultimate stress in compression
```

```
       set conlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set conft    [expr $unconfc*-0.1];         # maximum stress in tension for concrete
       set conEt    [expr $unconft/0.002];        # elastic modulus in tension
       set conE0    [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

    # uniaxialMaterial Concrete02  $matTag     $fpc $epsc0 $fpcu  $epsU  $lambda  $ft  $Ets
    uniaxialMaterial   Concrete02  $C_unconfined $unconfc  $unconepsc  $unconfu    $unconepsu  $unconlambda  $unconft  $unconEt;
    uniaxialMaterial   Concrete02  $C_confined $confc  $conepsc  $confu    $conepsu  $conlambda  $conft  $conEt;

############################################################################################################################################
# definition of the Sections
############################################################################################################################################

#   define sections IDs

    set col25x60  1;
    set beam150x30 2;

#   define section parameters

    set pi        3.141593;
    set rebar_12  [expr $pi*12.0*12.0/4];  # area rebar 12mm
    set rebar_18  [expr $pi*18.0*18.0/4];
    set w_col     250.0;   # column width
    set h_col     600.0;   # column hieght
    set c_col     20.0;  # column cover
    set w_beam    1500.0;  # beam width
    set h_beam    300.0;   # beam hieght
    set c_beam    30.0;  # beam cover

#   load procedure for fiber section

source BuildRCrectSection.tcl;

#   build sections

    #BuildRCrectSection $ColSecTag $HSec   $BSec   $coverH  $coverB  $coreID      $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
    BuildRCrectSection  $col25x60  $h_col  $w_col  $c_col   $c_col   $C_confined $C_unconfined $R_steel 4           $rebar_18   4           $rebar_18   2              $rebar_18   8        8        8         8
    BuildRCrectSection  $beam150x30 $h_beam $w_beam $c_beam  $c_beam $C_confined $C_unconfined $R_steel 12          $rebar_12   8          $rebar_12   0              $rebar_12   8        8        8         8


##############################################################################################################################
# beam column joint definition
##############################################################################################################################

# dimensions of the joint respectively
set JointWidth [expr $h_col]; set JointHeight [expr $h_beam]; set JointDepth $w_col ;
set JointVolume [expr $JointWidth*$JointHeight*$JointDepth];

######################### details for the material models of bar slip of the beam ####################################

set bs_fc 28.0; set bs_fs 420.0; set bs_es 200000; set bs_fsu 596; set bs_dbar 12.0; set bs_esh 3100.0;
set bs_wid $w_col; set bs_dep $h_beam;
set bsT_nbars 12; set bsB_nbars 8;
set bs_ljoint $h_col;

########################## details for the material models of bar slip of the column ##################################

set cs_fc 28.0; set cs_fs 420.0; set cs_es 200000.0; set cs_fsu 596; set cs_dbar 18.0; set cs_esh 3100.0;
set cs_wid $w_col; set cs_dep $h_col;
set cs_nbars 5;
set cs_ljoint $h_beam;

#######################################################################################################################
#bar slip definition

# for beam bottom

set bsid1 11
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid1 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsB_nbars $bs_wid $bs_dep  strong
```

```
beambot

# for beam top

set bsid2 21
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid2 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsT_nbars $bs_wid $bs_dep  strong beamtop

# for columns
set bsid3 31

#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid3 $cs_fc $cs_fs $cs_es $cs_fsu $cs_esh $cs_dbar $cs_ljoint $cs_nbars $cs_wid $cs_dep  strong column


############################ material for shear panel ##############################################################

## Positive/Negative envelope Stress

set spid1 41;
set A 0.78;
set p1 [expr 2.539*$A]; set p2 [expr 3.005*$A]; set p3 [expr 3.163*$A]; set p4 [expr 0.6326*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1*$JointVolume] [expr $p2*$JointVolume] [expr $p3*$JointVolume] [expr $p4*$JointVolume]]
set nEnvStrsp [list [expr -$p1*$JointVolume] [expr -$p2*$JointVolume] [expr -$p3*$JointVolume] [expr -$p4*$JointVolume]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.0008 0.015  0.035 0.04]
set nEnvStnsp [list -0.0008 -0.015  -0.035 -0.04]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $spid1 [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
```

```
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy

#################### beam column joint ###########################################

##element BeamColumnJoint tag? iNode? jNode? kNode? lNode? matTag1? matTag2? matTag3? matTag4?
## matTag5? matTag6? matTag7? matTag8? matTag9? matTag10? matTag11? matTag12? matTag13?
## <element Height factor?> <element Width factor?>
## please note: the four nodes are in anticlockwise direction around the element
## requires material tags for all 13 different components within the element.
## the first 12 being that of spring and the last of the shear panel

set jointA1 611
set jointA2 612
set jointA3 613
set jointA4 614
set jointA5 615
set jointA6 616

set jointB1 621
set jointB2 622
set jointB3 623
set jointB4 624
set jointB5 625
set jointB6 626

set jointC1 631
set jointC2 632
set jointC3 633
set jointC4 634
set jointC5 635
set jointC6 636

set jointD1 641
set jointD2 642
set jointD3 643
set jointD4 644
set jointD5 645
set jointD6 646

# add material Properties - command: uniaxialMaterial matType matTag ...
#command: uniaxialMaterial Elastic tag? E?

uniaxialMaterial Elastic 71 10000000000.0

element beamColumnJoint $jointA1 $N_A1 $N_A10_R $N_A10_A $N_A10_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA2 $N_A2 $N_A20_R $N_A20_A $N_A20_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3  71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA3 $N_A3 $N_A30_R $N_A30_A $N_A30_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA4 $N_A4 $N_A40_R $N_A40_A $N_A40_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA5 $N_A5 $N_A50_R $N_A50_A $N_A50_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA6 $N_A6 $N_A60_R $N_A60_A $N_A60_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointB1 $N_B1 $N_B11_R $N_B11_A $N_B11_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB2 $N_B2 $N_B21_R $N_B21_A $N_B21_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB3 $N_B3 $N_B31_R $N_B31_A $N_B31_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB4 $N_B4 $N_B41_R $N_B41_A $N_B41_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB5 $N_B5 $N_B51_R $N_B51_A $N_B51_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB6 $N_B6 $N_B61_R $N_B61_A $N_B61_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointC1 $N_C1 $N_C12_R $N_C12_A $N_C12_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC2 $N_C2 $N_C22_R $N_C22_A $N_C22_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
```

```
element beamColumnJoint $jointC3 $N_C3 $N_C32_R $N_C32_A $N_C32_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC4 $N_C4 $N_C42_R $N_C42_A $N_C42_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC5 $N_C5 $N_C52_R $N_C52_A $N_C52_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC6 $N_C6 $N_C62_R $N_C62_A $N_C62_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointD1 $N_D1 $N_D13_R $N_D13_A $N_D13_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD2 $N_D2 $N_D23_R $N_D23_A $N_D23_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD3 $N_D3 $N_D33_R $N_D33_A $N_D33_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD4 $N_D4 $N_D43_R $N_D43_A $N_D43_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD5 $N_D5 $N_D53_R $N_D53_A $N_D53_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD6 $N_D6 $N_D63_R $N_D63_A $N_D63_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1


###################################################################################################################
# Elements definitions
###################################################################################################################

# COLUMN definition

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
set ColTransfTag 1;        # associate a tag to column transformation
geomTransf PDelta $ColTransfTag ; #Columns

# --------------------------------------------------------------------
# ---- element connectivity "Columns Definition"----------
# --------------------------------------------------------------------
set numIntPoints 4;
set integrationC "Lobatto $col25x60 $numIntPoints"

element forceBeamColumn 710      $N_A0   $N_A1 $ColTransfTag $integrationC
element forceBeamColumn 720      $N_A10_A   $N_A2 $ColTransfTag $integrationC
element forceBeamColumn 730      $N_A20_A   $N_A3 $ColTransfTag $integrationC
element forceBeamColumn 740      $N_A30_A   $N_A4 $ColTransfTag $integrationC
element forceBeamColumn 750      $N_A40_A   $N_A5 $ColTransfTag $integrationC
element forceBeamColumn 760      $N_A50_A   $N_A6 $ColTransfTag $integrationC

element forceBeamColumn 711      $N_B0   $N_B1 $ColTransfTag $integrationC
element forceBeamColumn 721      $N_B11_A   $N_B2 $ColTransfTag $integrationC
element forceBeamColumn 731      $N_B21_A   $N_B3 $ColTransfTag $integrationC
element forceBeamColumn 741      $N_B31_A   $N_B4 $ColTransfTag $integrationC
element forceBeamColumn 751      $N_B41_A   $N_B5 $ColTransfTag $integrationC
element forceBeamColumn 761      $N_B51_A   $N_B6 $ColTransfTag $integrationC

element forceBeamColumn 712      $N_C0   $N_C1 $ColTransfTag $integrationC
element forceBeamColumn 722      $N_C12_A   $N_C2 $ColTransfTag $integrationC
element forceBeamColumn 732      $N_C22_A   $N_C3 $ColTransfTag $integrationC
element forceBeamColumn 742      $N_C32_A   $N_C4 $ColTransfTag $integrationC
element forceBeamColumn 752      $N_C42_A   $N_C5 $ColTransfTag $integrationC
element forceBeamColumn 762      $N_C52_A   $N_C6 $ColTransfTag $integrationC

element forceBeamColumn 713      $N_D0   $N_D1 $ColTransfTag $integrationC
element forceBeamColumn 723      $N_D13_A   $N_D2 $ColTransfTag $integrationC
element forceBeamColumn 733      $N_D23_A   $N_D3 $ColTransfTag $integrationC
element forceBeamColumn 743      $N_D33_A   $N_D4 $ColTransfTag $integrationC
element forceBeamColumn 753      $N_D43_A   $N_D5 $ColTransfTag $integrationC
element forceBeamColumn 763      $N_D53_A   $N_D6 $ColTransfTag $integrationC


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

# BEAMS definition

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
set BeamTransfTag 2;          # associate a tag to beam transformation
geomTransf PDelta $BeamTransfTag  ; #Beams
```

```
# --------------------------------------------------------------------
# ---- element connectivity "Beamss Definition"----------
# --------------------------------------------------------------------
set numIntPoints_beams 5;
set integrationB "Lobatto $beam150x30 $numIntPoints_beams"

element forceBeamColumn  810    $N_A10_R $N_B11_L  $BeamTransfTag $integrationB
element forceBeamColumn  820    $N_A20_R $N_B21_L  $BeamTransfTag $integrationB
element forceBeamColumn  830    $N_A30_R $N_B31_L  $BeamTransfTag $integrationB
element forceBeamColumn  840    $N_A40_R $N_B41_L  $BeamTransfTag $integrationB
element forceBeamColumn  850    $N_A50_R $N_B51_L  $BeamTransfTag $integrationB
element forceBeamColumn  860    $N_A60_R $N_B61_L  $BeamTransfTag $integrationB

element forceBeamColumn  811    $N_B11_R $N_C12_L  $BeamTransfTag $integrationB
element forceBeamColumn  821    $N_B21_R $N_C22_L  $BeamTransfTag $integrationB
element forceBeamColumn  831    $N_B31_R $N_C32_L  $BeamTransfTag $integrationB
element forceBeamColumn  841    $N_B41_R $N_C42_L  $BeamTransfTag $integrationB
element forceBeamColumn  851    $N_B51_R $N_C52_L  $BeamTransfTag $integrationB
element forceBeamColumn  861    $N_B61_R $N_C62_L  $BeamTransfTag $integrationB

element forceBeamColumn  812    $N_C12_R $N_D13_L  $BeamTransfTag $integrationB
element forceBeamColumn  822    $N_C22_R $N_D23_L  $BeamTransfTag $integrationB
element forceBeamColumn  832    $N_C32_R $N_D33_L  $BeamTransfTag $integrationB
element forceBeamColumn  842    $N_C42_R $N_D43_L  $BeamTransfTag $integrationB
element forceBeamColumn  852    $N_C52_R $N_D53_L  $BeamTransfTag $integrationB
element forceBeamColumn  862    $N_C62_R $N_D63_L  $BeamTransfTag $integrationB


##########################################################################################################################
# infill walls definitions
##########################################################################################################################

############ METHOD A ##########################################################

# diagonal members

set dia1A    101;
set dia2A    102;
set dia3A    103;
set dia4A    104;
set dia5A    105;
set dia6A    106;
set dia7A    107;
set dia8A    108;
set dia9A    109;
set dia10A  1010;
set dia11A  1011;
set dia12A  1012;
set dia13A  1013;
set dia14A  1014;
set dia15A  1015;
set dia16A  1016;
set dia17A  1017;
set dia18A  1018;
set dia19A  1019;
set dia20A  1020;
set dia21A  1021;
set dia22A  1022;
set dia23A  1023;
set dia24A  1024;

set dia1B   1025;
set dia2B   1026;
set dia3B   1027;
set dia4B   1028;
set dia5B   1029;
set dia6B   1030;
set dia7B   1031;
set dia8B   1032;
set dia9B   1033;
set dia10B  1034;
set dia11B  1035;
set dia12B  1036;
set dia13B  1037;
set dia14B  1038;
set dia15B  1039;
set dia16B  1040;
set dia17B  1041;
set dia18B  1042;
set dia19B  1043;
```

```
set dia20B  1044;
set dia21B  1045;
set dia22B  1046;
set dia23B  1047;
set dia24B  1048;

set dia1C    1049;
set dia2C    1050;
set dia3C    1051;
set dia4C    1052;
set dia5C    1053;
set dia6C    1054;
set dia7C    1055;
set dia8C    1056;
set dia9C    1057;
set dia10C  1058;
set dia11C  1059;
set dia12C  1060;
set dia13C  1061;
set dia14C  1062;
set dia15C  1063;
set dia16C  1064;
set dia17C  1065;
set dia18C  1066;
set dia19C  1067;
set dia20C  1068;
set dia21C  1069;
set dia22C  1070;
set dia23C  1071;
set dia24C  1072;

set width_wall 665;      #using equation 16 in the report, the width of the strut based on H=3m,L=4m, and t=0.3m
set t_wall 200;
set  Aw [expr $width_wall*$t_wall];   #cross-sectional
set Ew   10000000000.0;   #Young's Modulus
set Izw [expr $t_wall*(pow($width_wall,3))/12]; #second moment of area about the local z-axis

set WtransfTag 81;
geomTransf Linear $WtransfTag;

# --------------------------------------------------------------------
# ---- element connectivity "diagonal infill walls Definition"----------
# --------------------------------------------------------------------

element elasticBeamColumn $dia1A $N_A1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2A $N_B1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3A $N_A0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4A $N_B0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia5A $N_A2 $N_W2A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia6A $N_B2 $N_W2A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia7A $N_A10_A $N_W2A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia8A $N_B11_A $N_W2A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia9A $N_A3 $N_W3A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia10A $N_B3 $N_W3A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia11A $N_A20_A $N_W3A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia12A $N_B21_A $N_W3A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia13A $N_A4 $N_W4A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia14A $N_B4 $N_W4A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia15A $N_A30_A $N_W4A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia16A $N_B31_A $N_W4A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia17A $N_A5 $N_W5A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia18A $N_B5 $N_W5A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia19A $N_A40_A $N_W5A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia20A $N_B41_A $N_W5A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia21A $N_A6 $N_W6A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia22A $N_B6 $N_W6A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia23A $N_A50_A $N_W6A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia24A $N_B51_A $N_W6A_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1B $N_B1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2B $N_C1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3B $N_B0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4B $N_C0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia5B $N_B2 $N_W2B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia6B $N_C2 $N_W2B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia7B $N_B11_A $N_W2B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia8B $N_C12_A $N_W2B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia9B $N_B3 $N_W3B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia10B $N_C3 $N_W3B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia11B $N_B21_A $N_W3B_R $Aw $Ew $Izw $WtransfTag
```

```
element elasticBeamColumn $dia12B $N_C22_A $N_W3B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia13B $N_B4 $N_W4B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia14B $N_C4 $N_W4B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia15B $N_B31_A $N_W4B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia16B $N_C32_A $N_W4B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia17B $N_B5 $N_W5B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia18B $N_C5 $N_W5B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia19B $N_B41_A $N_W5B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia20B $N_C42_A $N_W5B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia21B $N_B6 $N_W6B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia22B $N_C6 $N_W6B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia23B $N_B51_A $N_W6B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia24B $N_C52_A $N_W6B_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1C $N_C1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2C $N_D1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3C $N_C0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4C $N_D0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia5C $N_C2 $N_W2C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia6C $N_D2 $N_W2C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia7C $N_C12_A $N_W2C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia8C $N_D13_A $N_W2C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia9C $N_C3 $N_W3C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia10C $N_D3 $N_W3C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia11C $N_C22_A $N_W3C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia12C $N_D23_A $N_W3C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia13C $N_C4 $N_W4C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia14C $N_D4 $N_W4C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia15C $N_C32_A $N_W4C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia16C $N_D33_A $N_W4C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia17C $N_C5 $N_W5C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia18C $N_D5 $N_W5C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia19C $N_C42_A $N_W5C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia20C $N_D43_A $N_W5C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia21C $N_C6 $N_W6C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia22C $N_D6 $N_W6C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia23C $N_C52_A $N_W6C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia24C $N_D53_A $N_W6C_R $Aw $Ew $Izw $WtransfTag

#  Central member

set cen1A 2001;
set cen2A 2002;
set cen3A 2003;
set cen4A 2004;
set cen5A 2005;
set cen6A 2006;
set cen1B 2007;
set cen2B 2008;
set cen3B 2009;
set cen4B 20010;
set cen5B 20011;
set cen6B 20012;
set cen1C 20013;
set cen2C 20014;
set cen3C 20015;
set cen4C 20016;
set cen5C 20017;
set cen6C 20018;

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
#set wallTransfTag 82;        # associate a tag to wall transformation
#geomTransf Linear $wallTransfTag  ; #walls

# --------------------------------------------------------------------
# ---- element connectivity "wall Definition"----------
# --------------------------------------------------------------------
#set numIntPoints_wall 2;

set wall_sec 91;
set wall_mat 92;





########################### material for infill walls #############################################################

```

```
## Positive/Negative envelope Stress

set A 1;
set p1 [expr 0.4*$A]; set p2 [expr 1.025*$A]; set p3 [expr 2.05*$A]; set p4 [expr 0.41*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1] [expr $p2] [expr $p3] [expr $p4]]
set nEnvStrsp [list [expr -$p1] [expr -$p2] [expr -$p3] [expr -$p4]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.000065 0.00385  0.00771 0.0120]
set nEnvStnsp [list -0.000065 -0.00385  -0.00771 -0.0120]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $wall_mat [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy

###################################### wall section ##############################################################################

#BuildRCrectSection $ColSecTag $HSec        $BSec   $coverH  $coverB  $coreID     $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
BuildRCrectSection  $wall_sec  $width_wall  $t_wall  20        20        $wall_mat   $wall_mat      $wall_mat 0            $rebar_18    0            $rebar_18    0               $rebar_18    8         8         8          8

set wallTransfTag 3;          # associate a tag to column transformation
geomTransf PDelta $wallTransfTag ; #Columns

# ---------------------------------------------------------------------
# ---- element connectivity "Columns Definition"----------
# ---------------------------------------------------------------------
```

```
set numIntPoints_wall 4;
set integrationw "Lobatto $wall_sec $numIntPoints_wall"

element forceBeamColumn $cen1A     $N_W1A_L $N_W1A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen2A     $N_W2A_L $N_W2A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen3A     $N_W3A_L $N_W3A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen4A     $N_W4A_L $N_W4A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen5A     $N_W5A_L $N_W5A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen6A     $N_W6A_L $N_W6A_R    $wallTransfTag $integrationw

element forceBeamColumn $cen1B     $N_W1B_L $N_W1B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen2B     $N_W2B_L $N_W2B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen3B     $N_W3B_L $N_W3B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen4B     $N_W4B_L $N_W4B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen5B     $N_W5B_L $N_W5B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen6B     $N_W6B_L $N_W6B_R    $wallTransfTag $integrationw

element forceBeamColumn $cen1C     $N_W1C_L $N_W1C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen2C     $N_W2C_L $N_W2C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen3C     $N_W3C_L $N_W3C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen4C     $N_W4C_L $N_W4C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen5C     $N_W5C_L $N_W5C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen6C     $N_W6C_L $N_W6C_R    $wallTransfTag $integrationw



############ METHOD B ##########################################################  Can be used in case of convergence problems (Validated Method Experimentally)

#set width_wall 665;      #using equation 16 in the report, the width of the strut based on H=3m,L=4m, and t=0.3m
#set t_wall 200;
#set  Aw [expr $width_wall*$t_wall];   #cross-sectional
#set Ew  10000000000.0;   #Young's Modulus
#set Izw [expr $t_wall*(pow($width_wall,3))/12];   #second moment of area about the local z-axis


#  Central member

#set cenA1 111;                       #A: means that the diagonal strut start from axis A to B, and the rest in that analogy.
#set cenA2 112;
#set cenA3 113;
#set cenA4 114;
#set cenA5 115;
#set cenA6 116;

#set cenB1 121;
#set cenB2 122;
#set cenB3 123;
#set cenB4 124;
#set cenB5 125;
#set cenB6 126;

#set cenC1 131;
#set cenC2 132;
#set cenC3 133;
#set cenC4 134;
#set cenC5 135;
#set cenC6 136;

# --------------------------------------------------------------------
# ---- element connectivity "wall Definition"----------
# --------------------------------------------------------------------

#set wall_sec 91;
#set wall_mat 92;





########################### material for infill walls #############################################################

## Positive/Negative envelope Stress

#       set unconfcw   -2.05;        # compression strength for concrete
#       set unconepscw  -0.000352;                 # strain at maximum stress in compression
#       set unconfuw   [expr $unconfcw*0.4];          # ultamate stress for concrete
#       set unconepsuw  -0.012;                  # strain at ultimate stress in compression
#       set unconlambdaw     0.1;               # ratio between reloading stiffness and itial stiffness in compression
```

```
#       set unconftw   [expr $unconfcw*-0.0001];         # maximum stress in tension for concrete
#       set unconEtw   [expr $unconftw/0.2];        # elastic modulus in tension
#       set unconE0w   8000;      #intial elastic tangent

#    uniaxialMaterial   Concrete02  $C_unconfinedw $unconfcw  $unconepscw  $unconfuw    $unconepsuw  $unconlambdaw  $unconftw  $unconEtw;



###################################### wall section ##########################################################################

#BuildRCrectSection $ColSecTag $HSec        $BSec   $coverH  $coverB  $coreID     $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
#BuildRCrectSection  $wall_sec  $width_wall  $t_wall  20       20       $C_unconfinedw   $C_unconfinedw     $C_unconfinedw 0           $rebar_18   0           $rebar_18   0               $rebar_18   8        8        8          8

#set wallTransfTag 3;         # associate a tag to column transformation
#geomTransf PDelta $wallTransfTag ; #Columns


# --------------------------------------------------------------------
# ---- element connectivity "Wall Definition"----------
# --------------------------------------------------------------------
#set numIntPoints_wall 4;
#set integrationw "Lobatto $wall_sec $numIntPoints_wall"

#element forceBeamColumn $cenA1    $N_A1 $N_B0       $wallTransfTag $integrationw
#element forceBeamColumn $cenA2    $N_A2 $N_B11_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA3    $N_A3 $N_B21_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA4    $N_A4 $N_B31_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA5    $N_A5 $N_B41_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA6    $N_A6 $N_B51_A    $wallTransfTag $integrationw

#element forceBeamColumn $cenB1    $N_B1 $N_C0       $wallTransfTag $integrationw
#element forceBeamColumn $cenB2    $N_B2 $N_C12_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB3    $N_B3 $N_C22_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB4    $N_B4 $N_C32_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB5    $N_B5 $N_C42_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB6    $N_B6 $N_C52_A    $wallTransfTag $integrationw

#element forceBeamColumn $cenC1    $N_C1 $N_D0       $wallTransfTag $integrationw
#element forceBeamColumn $cenC2    $N_C2 $N_D13_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC3    $N_C3 $N_D23_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC4    $N_C4 $N_D33_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC5    $N_C5 $N_D43_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC6    $N_C6 $N_D53_A    $wallTransfTag $integrationw






#############################################################################################################################
# display the model with the node numbers
   DisplayModel2D NodeNumbers

#############################################################################################################################
# gravity and masses load
#############################################################################################################################

    # timeSeries  "LinearDefault":   tsTag   cFactor
    timeSeries          Linear        1     -factor   1;

    # distributed loads

   #set   DL   11000.0;             # self weight add as point load (N)
    set   TLE   64800.0;            # TLE: Total Load at the middle columns
    set   TLM   129600.0;            # TLM: Total Load at the middle columns

    # pattern  PatternType  $PatternID  TimeSeriesType
    pattern       Plain          1             1       {
    #load  $nodeTag  (ndf  $LoadValues)
    load   $N_A10_A 0    [expr -$TLE]    0;
    load   $N_A20_A 0    [expr -$TLE]    0;
    load   $N_A30_A 0    [expr -$TLE]    0;
    load   $N_A40_A 0    [expr -$TLE]    0;
```

```
    load   $N_A50_A 0    [expr -$TLE]    0;
    load   $N_A60_A 0    [expr -$TLE]    0;

    load   $N_B11_A 0    [expr -$TLM]    0;
    load   $N_B21_A 0    [expr -$TLM]    0;
    load   $N_B31_A 0    [expr -$TLM]    0;
    load   $N_B41_A 0    [expr -$TLM]    0;
    load   $N_B51_A 0    [expr -$TLM]    0;
    load   $N_B61_A 0    [expr -$TLM]    0;

    load   $N_C12_A 0    [expr -$TLM]    0;
    load   $N_C22_A 0    [expr -$TLM]    0;
    load   $N_C32_A 0    [expr -$TLM]    0;
    load   $N_C42_A 0    [expr -$TLM]    0;
    load   $N_C52_A 0    [expr -$TLM]    0;
    load   $N_C62_A 0    [expr -$TLM]    0;

    load   $N_D13_A 0    [expr -$TLE]    0;
    load   $N_D23_A 0    [expr -$TLE]    0;
    load   $N_D33_A 0    [expr -$TLE]    0;
    load   $N_D43_A 0    [expr -$TLE]    0;
    load   $N_D53_A 0    [expr -$TLE]    0;
    load   $N_D63_A 0    [expr -$TLE]    0;

    #eleLoad  -ele  $eleTag1  <$eleTag2>  -type  -beamuniformload $wy
    #eleLoad   -ele     5           6      -type   -beamUniform     [expr -$DL];

    }


# masses

      set mass1  19440;
      set mass2  19440;
      set mass3  19440;
      set mass4  19440;
      set mass5  19440;
      set mass6  19440;



     # assign mass to nodes

     #mass  $nodetag  (ndf  $massvalues)
      mass   $N_A10_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A20_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A30_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A40_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A50_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A60_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_B11_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B21_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B31_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B41_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B51_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B61_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_C12_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C22_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C32_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C42_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C52_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C62_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_D13_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D23_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D33_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D43_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D53_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D63_L    [expr $mass1/2]  0.1  0.1;



puts "Model Built"
```

3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 3

5) Pushover Analysis Procedure:

```
################################################################################

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>            dof   resptype
#recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
#recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A5  $N_A6   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;              # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele   5 6    section fiber y z  $R_steel stressStrain
#recorder  Element   -file $dataDir/force10.out    -time    -ele  710    section 1  fiber y z $R_steel  stressStrain;
#recorder  Element   -file $dataDir/force60B.out    -time    -ele  860    section 1  fiber y z $R_steel stressStrain;
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860  localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 621 622 623 624 625 626 shearpanel stressStrain
recorder  Element   -file $dataDir/Strut1.out    -time    -ele 111 112 113 114 115 116 121 122 123 124 125 126  section 1  fiber y z  stressStrain;
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
   set ViewScale 5;
   DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1800;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.01;      # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 1;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;         # define load pattern -- generalized
      load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


   }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;     # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
```

```
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;    # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 0.01;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)    # 1: print information on each step;
variable testTypeStatic EnergyIncr ;   # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A6   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";   # format for screen/file output of DONE/PROBLEM a
```

```
nalysis
if {$ok != 0} {
   # if analysis fails, we try some other stuff, performance is slower inside this loop
   set Dstep 0.0;
   set ok 0
   while {$Dstep <= 1.0 && $ok == 0} {
      set controlDisp [nodeDisp $N_A6 1 ]
      set Dstep [expr $controlDisp/$Dmax]
      set ok [analyze 1 ]
      # if analysis fails, we try some other stuff
      # performance is slower inside this loop  global maxNumIterStatic;       # max no. of iterations performed before "failure to converge" is ret'd
      if {$ok != 0} {
         puts "Trying Newton with Initial Tangent .."
         test NormDispIncr   $Tol 3000 0
         algorithm Newton -initial
         set ok [analyze 1]
         test $testTypeStatic $TolStatic      $maxNumIterStatic    0
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying Broyden .."
         algorithm Broyden 8
         set ok [analyze 1 ]
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying NewtonWithLineSearch .."
         algorithm NewtonLineSearch 0.8
         set ok [analyze 1]
         algorithm $algorithmTypeStatic
      }

   }; # end while loop
};      # end if ok !0

# ----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
   puts [format $fmt1 "PROBLEM" $N_A6 1 [nodeDisp $N_A6 1] "mm"]
} else {
   puts [format $fmt1 "DONE"  $N_A6 1 [nodeDisp $N_A6 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 6 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B6S MRFs- Masonry-Concrete Infilled Frames without Ground Infills

Appendix 6: 3B6S masonry infilled Frame without ground infills

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B6S MRFs-Masonry-Concrete Infilled Frame without ground infills:

The code generated is the same as Appendix 5. However, infill walls at ground level shall be removed as the following:

-These elements must be removed:

  1. Diagonal members

element elasticBeamColumn $dia1A $N_A1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2A $N_B1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3A $N_A0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4A $N_B0 $N_W1A_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1B $N_B1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2B $N_C1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3B $N_B0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4B $N_C0 $N_W1B_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1C $N_C1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2C $N_D1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3C $N_C0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4C $N_D0 $N_W1C_R $Aw $Ew $Izw $WtransfTag

  2. Central members

element forceBeamColumn $cen1A    $N_W1A_L $N_W1A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1B    $N_W1B_L $N_W1B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1C    $N_W1C_L $N_W1C_R    $wallTransfTag $integrationw


3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 3

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

```

```
#   Node Recorder "Displacements":  fileName                <nodeTag>           dof   resptype
#recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
#recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A5  $N_A6   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;         # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
#recorder  Element   -file $dataDir/force10.out    -time    -ele  710    section 1  fiber y z $R_steel  stressStrain;
#recorder  Element   -file $dataDir/force60B.out    -time    -ele  860    section 1  fiber y z $R_steel stressStrain;
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860  localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 621 622 623 624 625 626 shearpanel stressStrain

recorder  Element   -file $dataDir/Strut1.out    -time    -ele  112 113 114 115 116  122 123 124 125 126  section 1  fiber y z  stressStrain;
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce ;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
  set ViewScale 5;
  DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1800;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.01;   # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 1;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;     # define load pattern -- generalized
    load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


  }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;    # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.

#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;   # try UmfPack for large model
system $systemTypeStatic

```

```
# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 1;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;  # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A6   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";  # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
  # if analysis fails, we try some other stuff, performance is slower inside this loop
  set Dstep 0.0;
  set ok 0
  while {$Dstep <= 1.0 && $ok == 0} {
    set controlDisp [nodeDisp $N_A6 1 ]
    set Dstep [expr $controlDisp/$Dmax]
    set ok [analyze 1 ]
    # if analysis fails, we try some other stuff
    # performance is slower inside this loop  global maxNumIterStatic;      # max no. of iterations performed before "failure to converge" is ret'd
    if {$ok != 0} {
      puts "Trying Newton with Initial Tangent .."
      test NormDispIncr   $Tol 3000 0
      algorithm Newton -initial
      set ok [analyze 1]
      test $testTypeStatic $TolStatic      $maxNumIterStatic    0
      algorithm $algorithmTypeStatic
    }
```

```
    if {$ok != 0} {
      puts "Trying Broyden .."
      algorithm Broyden 8
      set ok [analyze 1 ]
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying NewtonWithLineSearch .."
      algorithm NewtonLineSearch 0.8
      set ok [analyze 1]
      algorithm $algorithmTypeStatic
    }

  };  # end while loop
};      # end if ok !0

# ----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
  puts [format $fmt1 "PROBLEM" $N_A6 1 [nodeDisp $N_A6 1] "mm"]
} else {
  puts [format $fmt1 "DONE"  $N_A6 1 [nodeDisp $N_A6 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 7 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B9S MRFs-Ductile Bare Frame

Appendix 7: 3B9S Bare Frame

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3

   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.

2) 2D Model Definition for 3B9S Bare Frame:

```
#performing nonlinear static pushover analysis on 3B9S Bare Frame
##########################################################################################################################

   wipe all;
# define model builder
#    model basic builder  -ndm $ndm <-ndf $ndf>
     model basic builder  -ndm 2  -ndf 3

     set dataDir Results;           # set up name of data directory
     file mkdir $dataDir;           # create data directory
     source Libunits.tcl;               # define basic system units
     source DisplayModel2D.tcl;        # procedure for displaying a 2D View of model
     source DisplayPlane.tcl;       # procedure for displaying a plane in a model

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

# buiding geometry
###########################################################################################################################


# dimensions

    set span1 4000.0;
    set span2 4000.0;
    set span3 4000.0;
    set storey1 3000.0;
    set storey2 3000.0;
    set storey3 3000.0;
    set storey4 3000.0;
    set storey5 3000.0;
    set storey6 3000.0;
    set storey7 3000.0;
    set storey8 3000.0;
    set storey9 3000.0;

# main grid lines
    # vertical axis, x
    set x1 [expr 0];
    set x2 [expr $x1+$span1];
    set x3 [expr $x2+$span2];
    set x4 [expr $x3+$span3];

    # hoeizontal axis, y
    set z0 [expr 0];
    set z1 [expr $z0+$storey1];
    set z2 [expr $z1+$storey2];
    set z3 [expr $z2+$storey3];
    set z4 [expr $z3+$storey4];
    set z5 [expr $z4+$storey5];
    set z6 [expr $z5+$storey6];
```

```
    set z7 [expr $z6+$storey7];
    set z8 [expr $z7+$storey8];
    set z9 [expr $z8+$storey9];

# definition of nodes

      #assigning node tages                    # for axises A,B,C, and D.
      set N_A0     1;
      set N_B0     2;
      set N_C0     3;
      set N_D0     4;
      set N_A1     5;
      set N_B1     6;
      set N_C1     7;
      set N_D1     8;
      set N_A2     9;
      set N_B2     10;
      set N_C2     11;
      set N_D2     12;
      set N_A3     13;
      set N_B3     14;
      set N_C3     15;
      set N_D3     16;
      set N_A4     17;
      set N_B4     18;
      set N_C4     19;
      set N_D4     20;
      set N_A5     21;
      set N_B5     22;
      set N_C5     23;
      set N_D5     24;
      set N_A6     25;
      set N_B6     26;
      set N_C6     27;
      set N_D6     28;
      set N_A7     29;
      set N_B7     30;
      set N_C7     31;
      set N_D7     32;
      set N_A8     33;
      set N_B8     34;
      set N_C8     35;
      set N_D8     36;
      set N_A9     37;
      set N_B9     38;
      set N_C9     39;
      set N_D9     40;

      set N_A10_R   41;                          # N_Aij_R    i: story level.   j: axis number
      set N_A10_A   42;
      set N_A10_L   43;
      set N_A20_R   44;
      set N_A20_A   45;
      set N_A20_L   46;
      set N_A30_R   47;
      set N_A30_A   48;
      set N_A30_L   49;
      set N_A40_R   50;
      set N_A40_A   51;
      set N_A40_L   52;
      set N_A50_R   53;
      set N_A50_A   54;
      set N_A50_L   55;
      set N_A60_R   56;
      set N_A60_A   57;
      set N_A60_L   58;
      set N_A70_R   59;
      set N_A70_A   60;
      set N_A70_L   61;
      set N_A80_R   62;
      set N_A80_A   63;
      set N_A80_L   64;
      set N_A90_R   65;
      set N_A90_A   66;
      set N_A90_L   67;

      set N_B11_R   68;
      set N_B11_A   69;
      set N_B11_L   70;
      set N_B21_R   71;
```

```
    set N_B21_A   72;
    set N_B21_L   73;
    set N_B31_R   74;
    set N_B31_A   75;
    set N_B31_L   76;
    set N_B41_R   77;
    set N_B41_A   78;
    set N_B41_L   79;
    set N_B51_R   80;
    set N_B51_A   81;
    set N_B51_L   82;
    set N_B61_R   83;
    set N_B61_A   84;
    set N_B61_L   85;
    set N_B71_R   86;
    set N_B71_A   87;
    set N_B71_L   88;
    set N_B81_R   89;
    set N_B81_A   90;
    set N_B81_L   91;
    set N_B91_R   92;
    set N_B91_A   93;
    set N_B91_L   94;

    set N_C12_R   95;
    set N_C12_A   96;
    set N_C12_L   97;
    set N_C22_R   98;
    set N_C22_A   99;
    set N_C22_L   100;
    set N_C32_R   101;
    set N_C32_A   102;
    set N_C32_L   103;
    set N_C42_R   104;
    set N_C42_A   105;
    set N_C42_L   106;
    set N_C52_R   107;
    set N_C52_A   108;
    set N_C52_L   109;
    set N_C62_R   110;
    set N_C62_A   111;
    set N_C62_L   112;
    set N_C72_R   113;
    set N_C72_A   114;
    set N_C72_L   115;
    set N_C82_R   116;
    set N_C82_A   117;
    set N_C82_L   118;
    set N_C92_R   119;
    set N_C92_A   120;
    set N_C92_L   121;

    set N_D13_R   122;
    set N_D13_A   123;
    set N_D13_L   124;
    set N_D23_R   125;
    set N_D23_A   126;
    set N_D23_L   127;
    set N_D33_R   128;
    set N_D33_A   129;
    set N_D33_L   130;
    set N_D43_R   131;
    set N_D43_A   132;
    set N_D43_L   133;
    set N_D53_R   134;
    set N_D53_A   135;
    set N_D53_L   136;
    set N_D63_R   137;
    set N_D63_A   138;
    set N_D63_L   139;
    set N_D73_R   140;
    set N_D73_A   141;
    set N_D73_L   142;
    set N_D83_R   143;
    set N_D83_A   144;
    set N_D83_L   145;
    set N_D93_R   146;
    set N_D93_A   147;
    set N_D93_L   148;

```

```

      #node $nodetag (ndm $coords) <-mass (ndf $massvalues)>

      set col_halfdepA [expr 600/2];              # This is used to define the joint dimensions.
      set col_halfdepB [expr 600/2];
      set col_halfdepC [expr 600/2];
      set col_halfdepD [expr 600/2];
      set beam_halfdep1 [expr 300/2];
      set beam_halfdep2 [expr 300/2];
      set beam_halfdep3 [expr 300/2];
      set beam_halfdep4 [expr 300/2];
      set beam_halfdep5 [expr 300/2];
      set beam_halfdep6 [expr 300/2];
      set beam_halfdep7 [expr 300/2];
      set beam_halfdep8 [expr 300/2];
      set beam_halfdep9 [expr 300/2];

      node  $N_A0      $x1 $z0;
      node  $N_B0      $x2 $z0;
      node  $N_C0      $x3 $z0;
      node  $N_D0      $x4 $z0;
      node  $N_A1      $x1 [expr $z1-$beam_halfdep1];
      node  $N_B1      $x2 [expr $z1-$beam_halfdep1];
      node  $N_C1      $x3 [expr $z1-$beam_halfdep1];
      node  $N_D1      $x4 [expr $z1-$beam_halfdep1];
      node  $N_A2      $x1 [expr $z2-$beam_halfdep2];
      node  $N_B2      $x2 [expr $z2-$beam_halfdep2];
      node  $N_C2      $x3 [expr $z2-$beam_halfdep2];
      node  $N_D2      $x4 [expr $z2-$beam_halfdep2];
      node  $N_A3      $x1 [expr $z3-$beam_halfdep3];
      node  $N_B3      $x2 [expr $z3-$beam_halfdep3];
      node  $N_C3      $x3 [expr $z3-$beam_halfdep3];
      node  $N_D3      $x4 [expr $z3-$beam_halfdep3];
      node  $N_A4      $x1 [expr $z4-$beam_halfdep4];
      node  $N_B4      $x2 [expr $z4-$beam_halfdep4];
      node  $N_C4      $x3 [expr $z4-$beam_halfdep4];
      node  $N_D4      $x4 [expr $z4-$beam_halfdep4];
      node  $N_A5      $x1 [expr $z5-$beam_halfdep5];
      node  $N_B5      $x2 [expr $z5-$beam_halfdep5];
      node  $N_C5      $x3 [expr $z5-$beam_halfdep5];
      node  $N_D5      $x4 [expr $z5-$beam_halfdep5];
      node  $N_A6      $x1 [expr $z6-$beam_halfdep6];
      node  $N_B6      $x2 [expr $z6-$beam_halfdep6];
      node  $N_C6      $x3 [expr $z6-$beam_halfdep6];
      node  $N_D6      $x4 [expr $z6-$beam_halfdep6];
      node  $N_A7      $x1 [expr $z7-$beam_halfdep7];
      node  $N_B7      $x2 [expr $z7-$beam_halfdep7];
      node  $N_C7      $x3 [expr $z7-$beam_halfdep7];
      node  $N_D7      $x4 [expr $z7-$beam_halfdep7];
      node  $N_A8      $x1 [expr $z8-$beam_halfdep8];
      node  $N_B8      $x2 [expr $z8-$beam_halfdep8];
      node  $N_C8      $x3 [expr $z8-$beam_halfdep8];
      node  $N_D8      $x4 [expr $z8-$beam_halfdep8];
      node  $N_A9      $x1 [expr $z9-$beam_halfdep9];
      node  $N_B9      $x2 [expr $z9-$beam_halfdep9];
      node  $N_C9      $x3 [expr $z9-$beam_halfdep9];
      node  $N_D9      $x4 [expr $z9-$beam_halfdep9];



############### add nodes - joints #########################################################

                                                   #  R: node at the right side of joint
                                                   #  A: node above the joint
                                                   #  L: node at the left side of the joint
      node $N_A10_R   [expr $x1+$col_halfdepA] $z1;
      node $N_A10_A   $x1 [expr $z1+$beam_halfdep1];
      node $N_A10_L   [expr $x1-$col_halfdepA] $z1;
      node $N_A20_R   [expr $x1+$col_halfdepA] $z2;
      node $N_A20_A   $x1 [expr $z2+$beam_halfdep2];
      node $N_A20_L   [expr $x1-$col_halfdepA] $z2;
      node $N_A30_R   [expr $x1+$col_halfdepA] $z3;
      node $N_A30_A   $x1 [expr $z3+$beam_halfdep3];
      node $N_A30_L   [expr $x1-$col_halfdepA] $z3;
      node $N_A40_R   [expr $x1+$col_halfdepA] $z4;
      node $N_A40_A   $x1 [expr $z4+$beam_halfdep4];
      node $N_A40_L   [expr $x1-$col_halfdepA] $z4;
      node $N_A50_R   [expr $x1+$col_halfdepA] $z5;
      node $N_A50_A   $x1 [expr $z5+$beam_halfdep5];
      node $N_A50_L   [expr $x1-$col_halfdepA] $z5;
```

```
        node $N_A60_R   [expr $x1+$col_halfdepA] $z6;
        node $N_A60_A   $x1 [expr $z6+$beam_halfdep6];
        node $N_A60_L   [expr $x1-$col_halfdepA] $z6;
        node $N_A70_R   [expr $x1+$col_halfdepA] $z7;
        node $N_A70_A   $x1 [expr $z7+$beam_halfdep7];
        node $N_A70_L   [expr $x1-$col_halfdepA] $z7;
        node $N_A80_R   [expr $x1+$col_halfdepA] $z8;
        node $N_A80_A   $x1 [expr $z8+$beam_halfdep8];
        node $N_A80_L   [expr $x1-$col_halfdepA] $z8;
        node $N_A90_R   [expr $x1+$col_halfdepA] $z9;
        node $N_A90_A   $x1 [expr $z9+$beam_halfdep9];
        node $N_A90_L   [expr $x1-$col_halfdepA] $z9;

        node $N_B11_R   [expr $x2+$col_halfdepB] $z1;
        node $N_B11_A   $x2 [expr $z1+$beam_halfdep1];
        node $N_B11_L   [expr $x2-$col_halfdepB] $z1;
        node $N_B21_R   [expr $x2+$col_halfdepB] $z2;
        node $N_B21_A   $x2 [expr $z2+$beam_halfdep2];
        node $N_B21_L   [expr $x2-$col_halfdepB] $z2;
        node $N_B31_R   [expr $x2+$col_halfdepB] $z3;
        node $N_B31_A   $x2 [expr $z3+$beam_halfdep3];
        node $N_B31_L   [expr $x2-$col_halfdepB] $z3;
        node $N_B41_R   [expr $x2+$col_halfdepB] $z4;
        node $N_B41_A   $x2 [expr $z4+$beam_halfdep4];
        node $N_B41_L   [expr $x2-$col_halfdepB] $z4;
        node $N_B51_R   [expr $x2+$col_halfdepB] $z5;
        node $N_B51_A   $x2 [expr $z5+$beam_halfdep5];
        node $N_B51_L   [expr $x2-$col_halfdepB] $z5;
        node $N_B61_R   [expr $x2+$col_halfdepB] $z6;
        node $N_B61_A   $x2 [expr $z6+$beam_halfdep6];
        node $N_B61_L   [expr $x2-$col_halfdepB] $z6;
        node $N_B71_R   [expr $x2+$col_halfdepB] $z7;
        node $N_B71_A   $x2 [expr $z7+$beam_halfdep7];
        node $N_B71_L   [expr $x2-$col_halfdepB] $z7;
        node $N_B81_R   [expr $x2+$col_halfdepB] $z8;
        node $N_B81_A   $x2 [expr $z8+$beam_halfdep8];
        node $N_B81_L   [expr $x2-$col_halfdepB] $z8;
        node $N_B91_R   [expr $x2+$col_halfdepB] $z9;
        node $N_B91_A   $x2 [expr $z9+$beam_halfdep9];
        node $N_B91_L   [expr $x2-$col_halfdepB] $z9;

        node $N_C12_R   [expr $x3+$col_halfdepC] $z1;
        node $N_C12_A   $x3 [expr $z1+$beam_halfdep1];
        node $N_C12_L   [expr $x3-$col_halfdepC] $z1;
        node $N_C22_R   [expr $x3+$col_halfdepC] $z2;
        node $N_C22_A   $x3 [expr $z2+$beam_halfdep2];
        node $N_C22_L   [expr $x3-$col_halfdepC] $z2;
        node $N_C32_R   [expr $x3+$col_halfdepC] $z3;
        node $N_C32_A   $x3 [expr $z3+$beam_halfdep3];
        node $N_C32_L   [expr $x3-$col_halfdepC] $z3;
        node $N_C42_R   [expr $x3+$col_halfdepC] $z4;
        node $N_C42_A   $x3 [expr $z4+$beam_halfdep4];
        node $N_C42_L   [expr $x3-$col_halfdepC] $z4;
        node $N_C52_R   [expr $x3+$col_halfdepC] $z5;
        node $N_C52_A   $x3 [expr $z5+$beam_halfdep5];
        node $N_C52_L   [expr $x3-$col_halfdepC] $z5;
        node $N_C62_R   [expr $x3+$col_halfdepC] $z6;
        node $N_C62_A   $x3 [expr $z6+$beam_halfdep6];
        node $N_C62_L   [expr $x3-$col_halfdepC] $z6;
        node $N_C72_R   [expr $x3+$col_halfdepC] $z7;
        node $N_C72_A   $x3 [expr $z7+$beam_halfdep7];
        node $N_C72_L   [expr $x3-$col_halfdepC] $z7;
        node $N_C82_R   [expr $x3+$col_halfdepC] $z8;
        node $N_C82_A   $x3 [expr $z8+$beam_halfdep8];
        node $N_C82_L   [expr $x3-$col_halfdepC] $z8;
        node $N_C92_R   [expr $x3+$col_halfdepC] $z9;
        node $N_C92_A   $x3 [expr $z9+$beam_halfdep9];
        node $N_C92_L   [expr $x3-$col_halfdepC] $z9;

        node $N_D13_R   [expr $x4+$col_halfdepD] $z1;
        node $N_D13_A   $x4 [expr $z1+$beam_halfdep1];
        node $N_D13_L   [expr $x4-$col_halfdepD] $z1;
        node $N_D23_R   [expr $x4+$col_halfdepD] $z2;
        node $N_D23_A   $x4 [expr $z2+$beam_halfdep2];
        node $N_D23_L   [expr $x4-$col_halfdepD] $z2;
        node $N_D33_R   [expr $x4+$col_halfdepD] $z3;
        node $N_D33_A   $x4 [expr $z3+$beam_halfdep3];
        node $N_D33_L   [expr $x4-$col_halfdepD] $z3;
        node $N_D43_R   [expr $x4+$col_halfdepD] $z4;
        node $N_D43_A   $x4 [expr $z4+$beam_halfdep4];
```

```
      node $N_D43_L   [expr $x4-$col_halfdepD] $z4;
      node $N_D53_R   [expr $x4+$col_halfdepD] $z5;
      node $N_D53_A   $x4 [expr $z5+$beam_halfdep5];
      node $N_D53_L   [expr $x4-$col_halfdepD] $z5;
      node $N_D63_R   [expr $x4+$col_halfdepD] $z6;
      node $N_D63_A   $x4 [expr $z6+$beam_halfdep6];
      node $N_D63_L   [expr $x4-$col_halfdepD] $z6;
      node $N_D73_R   [expr $x4+$col_halfdepD] $z7;
      node $N_D73_A   $x4 [expr $z7+$beam_halfdep7];
      node $N_D73_L   [expr $x4-$col_halfdepD] $z7;
      node $N_D83_R   [expr $x4+$col_halfdepD] $z8;
      node $N_D83_A   $x4 [expr $z8+$beam_halfdep8];
      node $N_D83_L   [expr $x4-$col_halfdepD] $z8;
      node $N_D93_R   [expr $x4+$col_halfdepD] $z9;
      node $N_D93_A   $x4 [expr $z9+$beam_halfdep9];
      node $N_D93_L   [expr $x4-$col_halfdepD] $z9;


# restraints

      #basefix  $nodetag  (ndf  $constraints)
      fix       $N_A0           1 1 1;
      fix       $N_B0           1 1 1;
      fix       $N_C0           1 1 1;
      fix       $N_D0           1 1 1;


##############################################################################################################################
# material definitions
##############################################################################################################################

#   Definition of materials IDs

       #set C_confinedB    1;
       set C_confined    1;
       set C_unconfined  2;
       set R_steel       3;



#   basic parameters for materials-con-concrete

# ConfinedConcrete01 Material

      #$tag           integer tag identifying material.
      #$secType         tag for the transverse reinforcement configuration.
      #$fpc           unconfined cylindrical strength of concrete specimen.
      #$Ec             initial elastic modulus of unconfined concrete.
      #<-epscu $epscu> OR <-gamma $gamma> confined concrete ultimate strain.
      #<-nu $nu> OR <-varub> OR <-varnoub>   Poisson's Ratio.
      #$L1            length/diameter of square/circular core section measured respect to the hoop center line.
      #($L2)            additional dimensions when multiple hoops are being used.
      #$phis         hoop diameter. If section arrangement has multiple hoops it refers to the external hoop.
      #$S             hoop spacing.
      #$fyh           yielding strength of the hoop steel.
      #$Es0             elastic modulus of the hoop steel.
      #$haRatio         hardening ratio of the hoop steel.
      #$mu            ductility factor of the hoop steel.
      #$phiLon       diameter of longitudinal bars.

#   basic parameters for materials-uncon-concrete

       set unconfc   -28.0;        # compression strength for concrete
       set unconepsc  -0.002;                 # strain at maximum stress in compression
       set unconfu   [expr $unconfc*0.18];          # ultamate stress for concrete
       set unconepsu  -0.01;                  # strain at ultimate stress in compression
       set unconlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set unconft   [expr $unconfc*-0.1];         # maximum stress in tension for concrete
       set unconEt   [expr $unconft/0.002];        # elastic modulus in tension
       set unconE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

#   basic parameters for material--steel        # ReinforcingSteel uniaxial material object. This object is intended to be used in a reinforced concrete fiber section as the steel reinforcing material.

       set Fy 420.0;    # Yield stress in tension
       set Fu 596.0;    # Ultimate stress in tension
       set Es 200000.0;    # Initial elastic tangent
       set Esh 3100.0;     # Tangent at initial strain hardening
```

```
       set esh 0.01;       # Strain corresponding to initial strain hardening
       set eult 0.09;         # Strain at peak stress

   #uniaxialMaterial ReinforcingSteel $matTag $fy $fu $Es $Esh $esh $eult    Define ReinforcingSteel uniaxial material
   uniaxialMaterial ReinforcingSteel $R_steel $Fy $Fu $Es $Esh $esh $eult -DMBuck 6 0.8 -CMFatigue 0.2600 0.5000 0.3890 -IsoHard 4.3000 0.01

#   definition of ConfinedConcrete01 material

    #uniaxialMaterial ConfinedConcrete01 $tag        $secType $fpc  $Ec   -epscu $epscu   $nu    $L1   $L2    $phis  $S   $fyh  $Es0     $haRatio  $mu  $phiLon  -stRatio $stRatio
    #uniaxialMaterial ConfinedConcrete01  $C_confinedB    R     -28 24870.1 -epscu -0.04   -varUB 250.0 1450.0 10.0  125.0 420.0 200000.0 0.00      3100.0 12.0    -stRatio 0.85
    #uniaxialMaterial ConfinedConcrete01  $C_confinedC    R     -28 24870.1 -epscu -0.04   -varUB 550.0 200.0  10.0  125.0 420.0 200000.0 0.00      3100.0 18.0    -stRatio 0.85


#   basic parameters for materials-con-concrete

       set confc   -32.5;        # compression strength for concrete
       set conepsc  -0.003;                  # strain at maximum stress in compression
       set confu   [expr $unconfc*0.18];           # ultamate stress for concrete
       set conepsu  -0.04;                   # strain at ultimate stress in compression
       set conlambda     0.1;                # ratio between reloading stiffness and itial stiffness in compression
       set conft   [expr $unconfc*-0.1];          # maximum stress in tension for concrete
       set conEt   [expr $unconft/0.002];         # elastic modulus in tension
       set conE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

    # uniaxialMaterial Concrete02  $matTag     $fpc $epsc0 $fpcu  $epsU  $lambda  $ft  $Ets
    uniaxialMaterial   Concrete02  $C_unconfined $unconfc  $unconepsc  $unconfu    $unconepsu  $unconlambda  $unconft  $unconEt;
    uniaxialMaterial   Concrete02  $C_confined $confc  $conepsc  $confu    $conepsu  $conlambda  $conft  $conEt;

################################################################################################################################################
# definition of the Sections
################################################################################################################################################

#   define sections IDs

    set col40x60  1;
    set beam150x30 2;

#   define section parameters

    set pi        3.141593;
    set rebar_12  [expr $pi*12.0*12.0/4];  # area rebar 12mm
    set rebar_16  [expr $pi*16.0*16.0/4];
    set w_col     400.0;   # column width
    set h_col     600.0;   # column hieght
    set c_col     20.0;  # column cover
    set w_beam    1500.0;  # beam width
    set h_beam    300.0;   # beam hieght
    set c_beam    30.0;  # beam cover

#   load procedure for fiber section

source BuildRCrectSection.tcl;

#   build sections

    #BuildRCrectSection $ColSecTag $HSec   $BSec   $coverH  $coverB  $coreID      $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
    BuildRCrectSection  $col40x60  $h_col  $w_col  $c_col   $c_col   $C_confined $C_unconfined $R_steel 9           $rebar_16   9           $rebar_16   2              $rebar_16   8        8        8         8
    BuildRCrectSection  $beam150x30 $h_beam $w_beam $c_beam  $c_beam $C_confined $C_unconfined $R_steel 12          $rebar_12   8          $rebar_12   0              $rebar_12   8        8        8         8


##############################################################################################################################
# beam column joint definition
##############################################################################################################################

# dimensions of the joint respectively
set JointWidth [expr $h_col]; set JointHeight [expr $h_beam]; set JointDepth $w_col ;
set JointVolume [expr $JointWidth*$JointHeight*$JointDepth];

```

```
######################### details for the material models of bar slip of the beam #####################################

set bs_fc 28.0; set bs_fs 420.0; set bs_es 200000; set bs_fsu 596; set bs_dbar 12.0; set bs_esh 3100.0;
set bs_wid $w_col; set bs_dep $h_beam;
set bsT_nbars 12; set bsB_nbars 8;
set bs_ljoint $h_col;

########################## details for the material models of bar slip of the column ##################################

set cs_fc 28.0; set cs_fs 420.0; set cs_es 200000.0; set cs_fsu 596; set cs_dbar 16.0; set cs_esh 3100.0;
set cs_wid $w_col; set cs_dep $h_col;
set cs_nbars 9;
set cs_ljoint $h_beam;

########################################################################################################################
#bar slip definition

# for beam bottom

set bsid1 11
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid1 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsB_nbars $bs_wid $bs_dep  strong beambot

# for beam top

set bsid2 21
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid2 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsT_nbars $bs_wid $bs_dep  strong beamtop

# for columns
set bsid3 31

#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid3 $cs_fc $cs_fs $cs_es $cs_fsu $cs_esh $cs_dbar $cs_ljoint $cs_nbars $cs_wid $cs_dep  strong column


############################ material for shear panel #############################################################

## Positive/Negative envelope Stress

set spid1 41;
set A 0.78;
set p1 [expr 2.539*$A]; set p2 [expr 3.393*$A]; set p3 [expr 3.57*$A]; set p4 [expr 0.7143*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1*$JointVolume] [expr $p2*$JointVolume] [expr $p3*$JointVolume] [expr $p4*$JointVolume]]
set nEnvStrsp [list [expr -$p1*$JointVolume] [expr -$p2*$JointVolume] [expr -$p3*$JointVolume] [expr -$p4*$JointVolume]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.0008 0.015  0.035 0.04]
set nEnvStnsp [list -0.0008 -0.015  -0.035 -0.04]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

```

```
#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $spid1 [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy

#################### beam column joint ###########################################

##element BeamColumnJoint tag? iNode? jNode? kNode? lNode? matTag1? matTag2? matTag3? matTag4?
## matTag5? matTag6? matTag7? matTag8? matTag9? matTag10? matTag11? matTag12? matTag13?
## <element Height factor?> <element Width factor?>
## please note: the four nodes are in anticlockwise direction around the element
## requires material tags for all 13 different components within the element.
## the first 12 being that of spring and the last of the shear panel

set jointA1 611
set jointA2 612
set jointA3 613
set jointA4 614
set jointA5 615
set jointA6 616
set jointA7 617
set jointA8 618
set jointA9 619

set jointB1 621
set jointB2 622
set jointB3 623
set jointB4 624
set jointB5 625
set jointB6 626
set jointB7 627
set jointB8 628
set jointB9 629

set jointC1 631
set jointC2 632
set jointC3 633
set jointC4 634
set jointC5 635
set jointC6 636
set jointC7 637
set jointC8 638
set jointC9 639

set jointD1 641
set jointD2 642
set jointD3 643
set jointD4 644
set jointD5 645
set jointD6 646
set jointD7 647
set jointD8 648
set jointD9 649

```

```
# add material Properties - command: uniaxialMaterial matType matTag ...
#command: uniaxialMaterial Elastic tag? E?

uniaxialMaterial Elastic 71 10000000000.0

element beamColumnJoint $jointA1 $N_A1 $N_A10_R $N_A10_A $N_A10_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA2 $N_A2 $N_A20_R $N_A20_A $N_A20_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA3 $N_A3 $N_A30_R $N_A30_A $N_A30_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA4 $N_A4 $N_A40_R $N_A40_A $N_A40_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA5 $N_A5 $N_A50_R $N_A50_A $N_A50_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA6 $N_A6 $N_A60_R $N_A60_A $N_A60_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA7 $N_A7 $N_A70_R $N_A70_A $N_A70_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA8 $N_A8 $N_A80_R $N_A80_A $N_A80_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA9 $N_A9 $N_A90_R $N_A90_A $N_A90_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointB1 $N_B1 $N_B11_R $N_B11_A $N_B11_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB2 $N_B2 $N_B21_R $N_B21_A $N_B21_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB3 $N_B3 $N_B31_R $N_B31_A $N_B31_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB4 $N_B4 $N_B41_R $N_B41_A $N_B41_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB5 $N_B5 $N_B51_R $N_B51_A $N_B51_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB6 $N_B6 $N_B61_R $N_B61_A $N_B61_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB7 $N_B7 $N_B71_R $N_B71_A $N_B71_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB8 $N_B8 $N_B81_R $N_B81_A $N_B81_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB9 $N_B9 $N_B91_R $N_B91_A $N_B91_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointC1 $N_C1 $N_C12_R $N_C12_A $N_C12_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC2 $N_C2 $N_C22_R $N_C22_A $N_C22_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC3 $N_C3 $N_C32_R $N_C32_A $N_C32_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC4 $N_C4 $N_C42_R $N_C42_A $N_C42_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC5 $N_C5 $N_C52_R $N_C52_A $N_C52_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC6 $N_C6 $N_C62_R $N_C62_A $N_C62_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC7 $N_C7 $N_C72_R $N_C72_A $N_C72_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC8 $N_C8 $N_C82_R $N_C82_A $N_C82_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC9 $N_C9 $N_C92_R $N_C92_A $N_C92_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointD1 $N_D1 $N_D13_R $N_D13_A $N_D13_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD2 $N_D2 $N_D23_R $N_D23_A $N_D23_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD3 $N_D3 $N_D33_R $N_D33_A $N_D33_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD4 $N_D4 $N_D43_R $N_D43_A $N_D43_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD5 $N_D5 $N_D53_R $N_D53_A $N_D53_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD6 $N_D6 $N_D63_R $N_D63_A $N_D63_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD7 $N_D7 $N_D73_R $N_D73_A $N_D73_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD8 $N_D8 $N_D83_R $N_D83_A $N_D83_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD9 $N_D9 $N_D93_R $N_D93_A $N_D93_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
```

```


##########################################################################################################################
# Elements definitions
##########################################################################################################################

# COLUMN definition

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
set ColTransfTag 1;        # associate a tag to column transformation
geomTransf PDelta $ColTransfTag ; #Columns

# --------------------------------------------------------------------
# ---- element connectivity "Columns Definition"----------
# --------------------------------------------------------------------
set numIntPoints 4;
set integrationC "Lobatto $col40x60 $numIntPoints"

element forceBeamColumn 710      $N_A0   $N_A1 $ColTransfTag $integrationC
element forceBeamColumn 720      $N_A10_A   $N_A2 $ColTransfTag $integrationC
element forceBeamColumn 730      $N_A20_A   $N_A3 $ColTransfTag $integrationC
element forceBeamColumn 740      $N_A30_A   $N_A4 $ColTransfTag $integrationC
element forceBeamColumn 750      $N_A40_A   $N_A5 $ColTransfTag $integrationC
element forceBeamColumn 760      $N_A50_A   $N_A6 $ColTransfTag $integrationC
element forceBeamColumn 770      $N_A60_A   $N_A7 $ColTransfTag $integrationC
element forceBeamColumn 780      $N_A70_A   $N_A8 $ColTransfTag $integrationC
element forceBeamColumn 790      $N_A80_A   $N_A9 $ColTransfTag $integrationC

element forceBeamColumn 711      $N_B0   $N_B1 $ColTransfTag $integrationC
element forceBeamColumn 721      $N_B11_A   $N_B2 $ColTransfTag $integrationC
element forceBeamColumn 731      $N_B21_A   $N_B3 $ColTransfTag $integrationC
element forceBeamColumn 741      $N_B31_A   $N_B4 $ColTransfTag $integrationC
element forceBeamColumn 751      $N_B41_A   $N_B5 $ColTransfTag $integrationC
element forceBeamColumn 761      $N_B51_A   $N_B6 $ColTransfTag $integrationC
element forceBeamColumn 771      $N_B61_A   $N_B7 $ColTransfTag $integrationC
element forceBeamColumn 781      $N_B71_A   $N_B8 $ColTransfTag $integrationC
element forceBeamColumn 791      $N_B81_A   $N_B9 $ColTransfTag $integrationC

element forceBeamColumn 712      $N_C0   $N_C1 $ColTransfTag $integrationC
element forceBeamColumn 722      $N_C12_A   $N_C2 $ColTransfTag $integrationC
element forceBeamColumn 732      $N_C22_A   $N_C3 $ColTransfTag $integrationC
element forceBeamColumn 742      $N_C32_A   $N_C4 $ColTransfTag $integrationC
element forceBeamColumn 752      $N_C42_A   $N_C5 $ColTransfTag $integrationC
element forceBeamColumn 762      $N_C52_A   $N_C6 $ColTransfTag $integrationC
element forceBeamColumn 772      $N_C62_A   $N_C7 $ColTransfTag $integrationC
element forceBeamColumn 782      $N_C72_A   $N_C8 $ColTransfTag $integrationC
element forceBeamColumn 792      $N_C82_A   $N_C9 $ColTransfTag $integrationC

element forceBeamColumn 713      $N_D0   $N_D1 $ColTransfTag $integrationC
element forceBeamColumn 723      $N_D13_A   $N_D2 $ColTransfTag $integrationC
element forceBeamColumn 733      $N_D23_A   $N_D3 $ColTransfTag $integrationC
element forceBeamColumn 743      $N_D33_A   $N_D4 $ColTransfTag $integrationC
element forceBeamColumn 753      $N_D43_A   $N_D5 $ColTransfTag $integrationC
element forceBeamColumn 763      $N_D53_A   $N_D6 $ColTransfTag $integrationC
element forceBeamColumn 773      $N_D63_A   $N_D7 $ColTransfTag $integrationC
element forceBeamColumn 783      $N_D73_A   $N_D8 $ColTransfTag $integrationC
element forceBeamColumn 793      $N_D83_A   $N_D9 $ColTransfTag $integrationC


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

# BEAMS definition

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
set BeamTransfTag 2;          # associate a tag to beam transformation
geomTransf PDelta $BeamTransfTag  ; #Beams

# --------------------------------------------------------------------
# ---- element connectivity "Beamss Definition"----------
# --------------------------------------------------------------------
set numIntPoints_beams 5;
set integrationB "Lobatto $beam150x30 $numIntPoints_beams"

element forceBeamColumn  810    $N_A10_R $N_B11_L  $BeamTransfTag $integrationB
element forceBeamColumn  820    $N_A20_R $N_B21_L  $BeamTransfTag $integrationB
element forceBeamColumn  830    $N_A30_R $N_B31_L  $BeamTransfTag $integrationB
```

```
element forceBeamColumn  840    $N_A40_R $N_B41_L  $BeamTransfTag $integrationB
element forceBeamColumn  850    $N_A50_R $N_B51_L  $BeamTransfTag $integrationB
element forceBeamColumn  860    $N_A60_R $N_B61_L  $BeamTransfTag $integrationB
element forceBeamColumn  870    $N_A70_R $N_B71_L  $BeamTransfTag $integrationB
element forceBeamColumn  880    $N_A80_R $N_B81_L  $BeamTransfTag $integrationB
element forceBeamColumn  890    $N_A90_R $N_B91_L  $BeamTransfTag $integrationB

element forceBeamColumn  811    $N_B11_R $N_C12_L  $BeamTransfTag $integrationB
element forceBeamColumn  821    $N_B21_R $N_C22_L  $BeamTransfTag $integrationB
element forceBeamColumn  831    $N_B31_R $N_C32_L  $BeamTransfTag $integrationB
element forceBeamColumn  841    $N_B41_R $N_C42_L  $BeamTransfTag $integrationB
element forceBeamColumn  851    $N_B51_R $N_C52_L  $BeamTransfTag $integrationB
element forceBeamColumn  861    $N_B61_R $N_C62_L  $BeamTransfTag $integrationB
element forceBeamColumn  871    $N_B71_R $N_C72_L  $BeamTransfTag $integrationB
element forceBeamColumn  881    $N_B81_R $N_C82_L  $BeamTransfTag $integrationB
element forceBeamColumn  891    $N_B91_R $N_C92_L  $BeamTransfTag $integrationB

element forceBeamColumn  812    $N_C12_R $N_D13_L  $BeamTransfTag $integrationB
element forceBeamColumn  822    $N_C22_R $N_D23_L  $BeamTransfTag $integrationB
element forceBeamColumn  832    $N_C32_R $N_D33_L  $BeamTransfTag $integrationB
element forceBeamColumn  842    $N_C42_R $N_D43_L  $BeamTransfTag $integrationB
element forceBeamColumn  852    $N_C52_R $N_D53_L  $BeamTransfTag $integrationB
element forceBeamColumn  862    $N_C62_R $N_D63_L  $BeamTransfTag $integrationB
element forceBeamColumn  872    $N_C72_R $N_D73_L  $BeamTransfTag $integrationB
element forceBeamColumn  882    $N_C82_R $N_D83_L  $BeamTransfTag $integrationB
element forceBeamColumn  892    $N_C92_R $N_D93_L  $BeamTransfTag $integrationB

###################################################################################################################################
# display the model with the node numbers
   DisplayModel2D NodeNumbers

###################################################################################################################################
# gravity and masses load
###################################################################################################################################

    # timeSeries  "LinearDefault":   tsTag   cFactor
    timeSeries          Linear        1     -factor   1;

    # distributed loads

   #set   DL   11000.0;             # self weight add as point load (N)
    set   TLE   68100.0;            # TLE: Total Load at the middle columns
    set   TLM   136100.0;            # TLM: Total Load at the middle columns

    # pattern  PatternType  $PatternID  TimeSeriesType
    pattern       Plain          1             1       {
    #load  $nodeTag  (ndf  $LoadValues)
    load   $N_A10_A 0    [expr -$TLE]    0;
    load   $N_A20_A 0    [expr -$TLE]    0;
    load   $N_A30_A 0    [expr -$TLE]    0;
    load   $N_A40_A 0    [expr -$TLE]    0;
    load   $N_A50_A 0    [expr -$TLE]    0;
    load   $N_A60_A 0    [expr -$TLE]    0;
    load   $N_A70_A 0    [expr -$TLE]    0;
    load   $N_A80_A 0    [expr -$TLE]    0;
    load   $N_A90_A 0    [expr -$TLE]    0;

    load   $N_B11_A 0    [expr -$TLM]    0;
    load   $N_B21_A 0    [expr -$TLM]    0;
    load   $N_B31_A 0    [expr -$TLM]    0;
    load   $N_B41_A 0    [expr -$TLM]    0;
    load   $N_B51_A 0    [expr -$TLM]    0;
    load   $N_B61_A 0    [expr -$TLM]    0;
    load   $N_B71_A 0    [expr -$TLM]    0;
    load   $N_B81_A 0    [expr -$TLM]    0;
    load   $N_B91_A 0    [expr -$TLM]    0;

    load   $N_C12_A 0    [expr -$TLM]    0;
    load   $N_C22_A 0    [expr -$TLM]    0;
    load   $N_C32_A 0    [expr -$TLM]    0;
    load   $N_C42_A 0    [expr -$TLM]    0;
    load   $N_C52_A 0    [expr -$TLM]    0;
    load   $N_C62_A 0    [expr -$TLM]    0;
    load   $N_C72_A 0    [expr -$TLM]    0;
    load   $N_C82_A 0    [expr -$TLM]    0;
    load   $N_C92_A 0    [expr -$TLM]    0;

```

```
    load   $N_D13_A 0    [expr -$TLE]    0;
    load   $N_D23_A 0    [expr -$TLE]    0;
    load   $N_D33_A 0    [expr -$TLE]    0;
    load   $N_D43_A 0    [expr -$TLE]    0;
    load   $N_D53_A 0    [expr -$TLE]    0;
    load   $N_D63_A 0    [expr -$TLE]    0;
    load   $N_D73_A 0    [expr -$TLE]    0;
    load   $N_D83_A 0    [expr -$TLE]    0;
    load   $N_D93_A 0    [expr -$TLE]    0;

    #eleLoad  -ele  $eleTag1  <$eleTag2>  -type  -beamuniformload $wy
    #eleLoad   -ele     5           6      -type  -beamUniform     [expr -$DL];

    }


# masses

      set mass1  20420;
      set mass2  20420;
      set mass3  20420;
      set mass4  20420;
      set mass5  20420;
      set mass6  20420;
      set mass7  20420;
      set mass8  20420;
      set mass9  20420;



     # assign mass to nodes

     #mass  $nodetag  (ndf  $massvalues)
      mass   $N_A10_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A20_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A30_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A40_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A50_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A60_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A70_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A80_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_A90_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_B11_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B21_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B31_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B41_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B51_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B61_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B71_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B81_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_B91_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_C12_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C22_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C32_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C42_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C52_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C62_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C72_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C82_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_C92_L    [expr $mass1/2]  0.1  0.1;

      mass   $N_D13_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D23_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D33_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D43_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D53_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D63_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D73_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D83_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D93_L    [expr $mass1/2]  0.1  0.1;



puts "Model Built"


3) Gravity Analysis Procedure:

```

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

```
#################################################################################

# start analysis

initialize

puts "ooo Analysis: ModalAnalysis ooo"

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

# set recorders

#   Node Recorder "EigenVectors":  fileName                <nodeTag>             dof   resptype
recorder Node -file $dataDir/ModalAnalysis_Node_EigenVectors_EigenVec1.out       -node  $N_A0  $N_A10_A $N_A20_A $N_A30_A $N_A40_A $N_A50_A $N_A60_A $N_A70_A $N_A80_A $N_A90_A -dof  1  eigen1
recorder Node -file $dataDir/ModalAnalysis_Node_EigenVectors_EigenVec2.out       -node  $N_A0  $N_A10_A $N_A20_A $N_A30_A $N_A40_A $N_A50_A $N_A60_A $N_A70_A $N_A80_A $N_A90_A -dof  1  eigen2

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

# analysis options


#  Constraint Handler
constraints  Transformation

# DOF numberer
numberer  Plain

# System of Equations
system  BandGeneral

# Convergence Test
test  NormDispIncr    1.00000E-5    50    0   2;

# Solution Algorithm
algorithm Newton

# Integrator
integrator  Newmark   5.0000000E-01    2.5000000E-01

# Analysis Type
analysis Transient

# Analysis model (and record responce)
      set pi [expr 2.0*asin(1.0)];                       # Definition of pi
   set nEigenI 1;                              # mode i = 1
   set nEigenJ 2;                              # mode j = 2
   set lambdaN [eigen [expr $nEigenJ]];              # eigenvalue analysis for nEigenJ modes
   set lambdaI [lindex $lambdaN [expr 0]];          # eigenvalue mode i = 1
   set lambdaJ [lindex $lambdaN [expr $nEigenJ-1]];   # eigenvalue mode j = 2
   set w1 [expr pow(($lambdaI*1000),0.5)];                # w1 (1st mode circular frequency)
   set w2 [expr pow(($lambdaJ*1000),0.5)];                # w2 (2nd mode circular frequency)
   set T1 [expr 2.0*$pi/$w1];                    # 1st mode period of the structure
   set T2 [expr 2.0*$pi/$w2];                    # 2nd mode period of the structure

puts "T1 is $T1"
puts "T2 is $T2"
# Record eigenvectors
record


# Reset for next analysis case
setTime 0.0
loadConst
remove recorders
wipeAnalysis

puts "Modal analysis completed"
```

5) Pushover Analysis Procedure:

```
#################################################################################

# start analysis

```

```
puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>             dof   resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A80_L  $N_A90_L   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A10_L  $N_A20_L -dof 1 2 disp;              # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
#recorder  Element   -file $dataDir/force10.out    -time    -ele  710    section 1  fiber y z $R_steel  stressStrain;
#recorder  Element   -file $dataDir/force90B.out    -time    -ele  890    section 1  fiber y z $R_steel stressStrain;
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810 811 812 820 821 822 830 831 832 840 841 842 850 851 852 860 861 862 870 871 872 880 881 882 890 891 892 localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 617 618 619 621 622 623 624 625 626 627 628 629 shearpanel stressStrain
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce ;

# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
   set ViewScale 5;
   DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1000;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.1;    # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 10;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;         # define load pattern -- generalized
      load $N_A90_L 9 0 0
        load $N_A80_L 8 0 0
        load $N_A70_L 7 0 0
        load $N_A60_L 6 0 0
        load $N_A50_L 5 0 0
        load $N_A40_L 4 0 0
        load $N_A30_L 3 0 0
        load $N_A20_L 2 0 0
        load $N_A10_L 1 0 0


   }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;     # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:
```

```

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;    # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 3;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;   # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A90_L   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";   # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
   # if analysis fails, we try some other stuff, performance is slower inside this loop
   set Dstep 0.0;
   set ok 0
   while {$Dstep <= 1.0 && $ok == 0} {
      set controlDisp [nodeDisp $N_A90_L 1 ]
```

```
      set Dstep [expr $controlDisp/$Dmax]
      set ok [analyze 1 ]
      # if analysis fails, we try some other stuff
      # performance is slower inside this loop  global maxNumIterStatic;       # max no. of iterations performed before "failure to converge" is ret'd
      if {$ok != 0} {
         puts "Trying Newton with Initial Tangent .."
         test NormDispIncr   $Tol 3000 0
         algorithm Newton -initial
         set ok [analyze 1]
         test $testTypeStatic $TolStatic      $maxNumIterStatic    0
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying Broyden .."
         algorithm Broyden 8
         set ok [analyze 1 ]
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying NewtonWithLineSearch .."
         algorithm NewtonLineSearch 0.8
         set ok [analyze 1]
         algorithm $algorithmTypeStatic
      }

   }; # end while loop
};      # end if ok !0

# ----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
   puts [format $fmt1 "PROBLEM" $N_A90_L 1 [nodeDisp $N_A90_L 1] "mm"]
} else {
   puts [format $fmt1 "DONE"  $N_A90_L 1 [nodeDisp $N_A90_L 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 8 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B9S MRFs-Non-Ductile Bare Frame

Appendix 8: 3B9S Bare Frame with Structural Deficiencies

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B9S Bare Frame with structural deficiencies :

The code generated is the same as Appendix 7. However, some changes were applied to represent structural deficiencies

  1. Consider the effect of stirrups spacing

The code generated is the same as Appendix 4.

  2. Consider Development Length

The code generated is the same as Appendix 4.

  3. Beam Column Joint Properties

The code generated is the same as Appendix 4.


3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 7

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>             dof   resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A80_L  $N_A90_L   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A10_L  $N_A20_L -dof 1 2 disp;         # displacements of free nodes

#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
#recorder  Element   -file $dataDir/force10.out    -time    -ele  710    section 1  fiber y z $R_steel  stressStrain;
#recorder  Element   -file $dataDir/force90B.out    -time    -ele  890    section 1  fiber y z $R_steel stressStrain;
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860 870 880 890 localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 617 618 619 621 622 623 624 625 626 627 628 629 shearpanel stressStrain
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce
```

```
;


# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
  set ViewScale 5;
  DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1000;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.009;    # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 10;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;     # define load pattern -- generalized
    load $N_A90_L 9 0 0
        load $N_A80_L 8 0 0
        load $N_A70_L 7 0 0
        load $N_A60_L 6 0 0
        load $N_A50_L 5 0 0
        load $N_A40_L 4 0 0
        load $N_A30_L 3 0 0
        load $N_A20_L 2 0 0
        load $N_A10_L 1 0 0


  }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;    # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;   # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
```

```
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 3;                         # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)    # 1: print information on each step;
variable testTypeStatic EnergyIncr ;  # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A90_L   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";  # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
  # if analysis fails, we try some other stuff, performance is slower inside this loop
  set Dstep 0.0;
  set ok 0
  while {$Dstep <= 1.0 && $ok == 0} {
    set controlDisp [nodeDisp $N_A90_L 1 ]
    set Dstep [expr $controlDisp/$Dmax]
    set ok [analyze 1 ]
    # if analysis fails, we try some other stuff
    # performance is slower inside this loop  global maxNumIterStatic;      # max no. of iterations performed before "failure to converge" is ret'd
    if {$ok != 0} {
      puts "Trying Newton with Initial Tangent .."
      test NormDispIncr   $Tol 3000 0
      algorithm Newton -initial
      set ok [analyze 1]
      test $testTypeStatic $TolStatic      $maxNumIterStatic    0
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying Broyden .."
      algorithm Broyden 8
      set ok [analyze 1 ]
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying NewtonWithLineSearch .."
      algorithm NewtonLineSearch 0.8
```

```
      set ok [analyze 1]
      algorithm $algorithmTypeStatic
    }

  };  # end while loop
};      # end if ok !0

# -----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
  puts [format $fmt1 "PROBLEM" $N_A90_L 1 [nodeDisp $N_A90_L 1] "mm"]
} else {
  puts [format $fmt1 "DONE"  $N_A90_L 1 [nodeDisp $N_A90_L 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 9 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B9S MRFs- Masonry-Concrete Infilled Frames

Appendix 9: 3B9S Masonry infilled Frame

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B9S MRFs-Masonry-Concrete Infilled Frame :


#performing nonlinear static pushover analysis on 3B9S MRFs-Masonry-Concrete Infilled Frame
##########################################################################################################################

   wipe all;
# define model builder
#    model basic builder  -ndm $ndm <-ndf $ndf>
     model basic builder  -ndm 2  -ndf 3

     set dataDir Results;           # set up name of data directory
     file mkdir $dataDir;           # create data directory
     source Libunits.tcl;               # define basic system units
     source DisplayModel2D.tcl;        # procedure for displaying a 2D View of model
     source DisplayPlane.tcl;       # procedure for displaying a plane in a model

###########################################################################################################################
# buiding geometry
###########################################################################################################################


# dimensions

    set span1 4000.0;
    set span2 4000.0;
    set span3 4000.0;
    set storey1 3000.0;
    set storey2 3000.0;
    set storey3 3000.0;
    set storey4 3000.0;
    set storey5 3000.0;
    set storey6 3000.0;
    set storey7 3000.0;
    set storey8 3000.0;
    set storey9 3000.0;

# main grid lines
    # vertical axis, x
    set x1 [expr 0];
    set x2 [expr $x1+$span1];
    set x3 [expr $x2+$span2];
    set x4 [expr $x3+$span3];

    # hoeizontal axis, y
    set z0 [expr 0];
    set z1 [expr $z0+$storey1];
    set z2 [expr $z1+$storey2];
    set z3 [expr $z2+$storey3];
    set z4 [expr $z3+$storey4];
    set z5 [expr $z4+$storey5];
    set z6 [expr $z5+$storey6];
```

```
    set z7 [expr $z6+$storey7];
    set z8 [expr $z7+$storey8];
    set z9 [expr $z8+$storey9];

# definition of nodes

      #assigning node tages                    # for axises A,B,C, and D.
      set N_A0     1;
      set N_B0     2;
      set N_C0     3;
      set N_D0     4;
      set N_A1     5;
      set N_B1     6;
      set N_C1     7;
      set N_D1     8;
      set N_A2     9;
      set N_B2     10;
      set N_C2     11;
      set N_D2     12;
      set N_A3     13;
      set N_B3     14;
      set N_C3     15;
      set N_D3     16;
      set N_A4     17;
      set N_B4     18;
      set N_C4     19;
      set N_D4     20;
      set N_A5     21;
      set N_B5     22;
      set N_C5     23;
      set N_D5     24;
      set N_A6     25;
      set N_B6     26;
      set N_C6     27;
      set N_D6     28;
      set N_A7     29;
      set N_B7     30;
      set N_C7     31;
      set N_D7     32;
      set N_A8     33;
      set N_B8     34;
      set N_C8     35;
      set N_D8     36;
      set N_A9     37;
      set N_B9     38;
      set N_C9     39;
      set N_D9     40;

      set N_A10_R   41;                          # N_Aij_R    i: story level.   j: axis number
      set N_A10_A   42;
      set N_A10_L   43;
      set N_A20_R   44;
      set N_A20_A   45;
      set N_A20_L   46;
      set N_A30_R   47;
      set N_A30_A   48;
      set N_A30_L   49;
      set N_A40_R   50;
      set N_A40_A   51;
      set N_A40_L   52;
      set N_A50_R   53;
      set N_A50_A   54;
      set N_A50_L   55;
      set N_A60_R   56;
      set N_A60_A   57;
      set N_A60_L   58;
      set N_A70_R   59;
      set N_A70_A   60;
      set N_A70_L   61;
      set N_A80_R   62;
      set N_A80_A   63;
      set N_A80_L   64;
      set N_A90_R   65;
      set N_A90_A   66;
      set N_A90_L   67;

      set N_B11_R   68;
      set N_B11_A   69;
      set N_B11_L   70;
      set N_B21_R   71;
```

```
    set N_B21_A   72;
    set N_B21_L   73;
    set N_B31_R   74;
    set N_B31_A   75;
    set N_B31_L   76;
    set N_B41_R   77;
    set N_B41_A   78;
    set N_B41_L   79;
    set N_B51_R   80;
    set N_B51_A   81;
    set N_B51_L   82;
    set N_B61_R   83;
    set N_B61_A   84;
    set N_B61_L   85;
    set N_B71_R   86;
    set N_B71_A   87;
    set N_B71_L   88;
    set N_B81_R   89;
    set N_B81_A   90;
    set N_B81_L   91;
    set N_B91_R   92;
    set N_B91_A   93;
    set N_B91_L   94;

    set N_C12_R   95;
    set N_C12_A   96;
    set N_C12_L   97;
    set N_C22_R   98;
    set N_C22_A   99;
    set N_C22_L   100;
    set N_C32_R   101;
    set N_C32_A   102;
    set N_C32_L   103;
    set N_C42_R   104;
    set N_C42_A   105;
    set N_C42_L   106;
    set N_C52_R   107;
    set N_C52_A   108;
    set N_C52_L   109;
    set N_C62_R   110;
    set N_C62_A   111;
    set N_C62_L   112;
    set N_C72_R   113;
    set N_C72_A   114;
    set N_C72_L   115;
    set N_C82_R   116;
    set N_C82_A   117;
    set N_C82_L   118;
    set N_C92_R   119;
    set N_C92_A   120;
    set N_C92_L   121;

    set N_D13_R   122;
    set N_D13_A   123;
    set N_D13_L   124;
    set N_D23_R   125;
    set N_D23_A   126;
    set N_D23_L   127;
    set N_D33_R   128;
    set N_D33_A   129;
    set N_D33_L   130;
    set N_D43_R   131;
    set N_D43_A   132;
    set N_D43_L   133;
    set N_D53_R   134;
    set N_D53_A   135;
    set N_D53_L   136;
    set N_D63_R   137;
    set N_D63_A   138;
    set N_D63_L   139;
    set N_D73_R   140;
    set N_D73_A   141;
    set N_D73_L   142;
    set N_D83_R   143;
    set N_D83_A   144;
    set N_D83_L   145;
    set N_D93_R   146;
    set N_D93_A   147;
    set N_D93_L   148;

```

```
#infill wall nodes
set N_W1A_L   3001;                         # l: means the node at the left side of the panel
set N_W1A_R   3002;                         # R: means the node at the right side of the panel
set N_W2A_L   3003;
set N_W2A_R   3004;
set N_W3A_L   3005;
set N_W3A_R   3006;
set N_W4A_L   3007;
set N_W4A_R   3008;
set N_W5A_L   3009;
set N_W5A_R   3010;
set N_W6A_L   3011;
set N_W6A_R   3012;
set N_W7A_L   3013;
set N_W7A_R   3014;
set N_W8A_L   3015;
set N_W8A_R   3016;
set N_W9A_L   3017;
set N_W9A_R   3018;

set N_W1B_L   3019;
set N_W1B_R   3020;
set N_W2B_L   3021;
set N_W2B_R   3022;
set N_W3B_L   3023;
set N_W3B_R   3024;
set N_W4B_L   3025;
set N_W4B_R   3026;
set N_W5B_L   3027;
set N_W5B_R   3028;
set N_W6B_L   3029;
set N_W6B_R   3030;
set N_W7B_L   3031;
set N_W7B_R   3032;
set N_W8B_L   3033;
set N_W8B_R   3034;
set N_W9B_L   3035;
set N_W9B_R   3036;

set N_W1C_L   3037;
set N_W1C_R   3038;
set N_W2C_L   3039;
set N_W2C_R   3040;
set N_W3C_L   3041;
set N_W3C_R   3042;
set N_W4C_L   3043;
set N_W4C_R   3044;
set N_W5C_L   3045;
set N_W5C_R   3046;
set N_W6C_L   3047;
set N_W6C_R   3048;
set N_W7C_L   3049;
set N_W7C_R   3050;
set N_W8C_L   3051;
set N_W8C_R   3052;
set N_W9C_L   3053;
set N_W9C_R   3054;


#node $nodetag (ndm $coords) <-mass (ndf $massvalues)>

set col_halfdepA [expr 600/2];              # This is used to define the joint dimensions.
set col_halfdepB [expr 600/2];
set col_halfdepC [expr 600/2];
set col_halfdepD [expr 600/2];
set beam_halfdep1 [expr 300/2];
set beam_halfdep2 [expr 300/2];
set beam_halfdep3 [expr 300/2];
set beam_halfdep4 [expr 300/2];
set beam_halfdep5 [expr 300/2];
set beam_halfdep6 [expr 300/2];
set beam_halfdep7 [expr 300/2];
set beam_halfdep8 [expr 300/2];
set beam_halfdep9 [expr 300/2];

node  $N_A0      $x1 $z0;
node  $N_B0      $x2 $z0;
node  $N_C0      $x3 $z0;
node  $N_D0      $x4 $z0;
node  $N_A1      $x1 [expr $z1-$beam_halfdep1];
```

```
      node  $N_B1      $x2 [expr $z1-$beam_halfdep1];
      node  $N_C1      $x3 [expr $z1-$beam_halfdep1];
      node  $N_D1      $x4 [expr $z1-$beam_halfdep1];
      node  $N_A2      $x1 [expr $z2-$beam_halfdep2];
      node  $N_B2      $x2 [expr $z2-$beam_halfdep2];
      node  $N_C2      $x3 [expr $z2-$beam_halfdep2];
      node  $N_D2      $x4 [expr $z2-$beam_halfdep2];
      node  $N_A3      $x1 [expr $z3-$beam_halfdep3];
      node  $N_B3      $x2 [expr $z3-$beam_halfdep3];
      node  $N_C3      $x3 [expr $z3-$beam_halfdep3];
      node  $N_D3      $x4 [expr $z3-$beam_halfdep3];
      node  $N_A4      $x1 [expr $z4-$beam_halfdep4];
      node  $N_B4      $x2 [expr $z4-$beam_halfdep4];
      node  $N_C4      $x3 [expr $z4-$beam_halfdep4];
      node  $N_D4      $x4 [expr $z4-$beam_halfdep4];
      node  $N_A5      $x1 [expr $z5-$beam_halfdep5];
      node  $N_B5      $x2 [expr $z5-$beam_halfdep5];
      node  $N_C5      $x3 [expr $z5-$beam_halfdep5];
      node  $N_D5      $x4 [expr $z5-$beam_halfdep5];
      node  $N_A6      $x1 [expr $z6-$beam_halfdep6];
      node  $N_B6      $x2 [expr $z6-$beam_halfdep6];
      node  $N_C6      $x3 [expr $z6-$beam_halfdep6];
      node  $N_D6      $x4 [expr $z6-$beam_halfdep6];
      node  $N_A7      $x1 [expr $z7-$beam_halfdep7];
      node  $N_B7      $x2 [expr $z7-$beam_halfdep7];
      node  $N_C7      $x3 [expr $z7-$beam_halfdep7];
      node  $N_D7      $x4 [expr $z7-$beam_halfdep7];
      node  $N_A8      $x1 [expr $z8-$beam_halfdep8];
      node  $N_B8      $x2 [expr $z8-$beam_halfdep8];
      node  $N_C8      $x3 [expr $z8-$beam_halfdep8];
      node  $N_D8      $x4 [expr $z8-$beam_halfdep8];
      node  $N_A9      $x1 [expr $z9-$beam_halfdep9];
      node  $N_B9      $x2 [expr $z9-$beam_halfdep9];
      node  $N_C9      $x3 [expr $z9-$beam_halfdep9];
      node  $N_D9      $x4 [expr $z9-$beam_halfdep9];



############### add nodes - joints #######################################################

                                                   #  R: node at the right side of joint
                                                   #  A: node above the joint
                                                   #  L: node at the left side of the joint
      node $N_A10_R   [expr $x1+$col_halfdepA] $z1;
      node $N_A10_A   $x1 [expr $z1+$beam_halfdep1];
      node $N_A10_L   [expr $x1-$col_halfdepA] $z1;
      node $N_A20_R   [expr $x1+$col_halfdepA] $z2;
      node $N_A20_A   $x1 [expr $z2+$beam_halfdep2];
      node $N_A20_L   [expr $x1-$col_halfdepA] $z2;
      node $N_A30_R   [expr $x1+$col_halfdepA] $z3;
      node $N_A30_A   $x1 [expr $z3+$beam_halfdep3];
      node $N_A30_L   [expr $x1-$col_halfdepA] $z3;
      node $N_A40_R   [expr $x1+$col_halfdepA] $z4;
      node $N_A40_A   $x1 [expr $z4+$beam_halfdep4];
      node $N_A40_L   [expr $x1-$col_halfdepA] $z4;
      node $N_A50_R   [expr $x1+$col_halfdepA] $z5;
      node $N_A50_A   $x1 [expr $z5+$beam_halfdep5];
      node $N_A50_L   [expr $x1-$col_halfdepA] $z5;
      node $N_A60_R   [expr $x1+$col_halfdepA] $z6;
      node $N_A60_A   $x1 [expr $z6+$beam_halfdep6];
      node $N_A60_L   [expr $x1-$col_halfdepA] $z6;
      node $N_A70_R   [expr $x1+$col_halfdepA] $z7;
      node $N_A70_A   $x1 [expr $z7+$beam_halfdep7];
      node $N_A70_L   [expr $x1-$col_halfdepA] $z7;
      node $N_A80_R   [expr $x1+$col_halfdepA] $z8;
      node $N_A80_A   $x1 [expr $z8+$beam_halfdep8];
      node $N_A80_L   [expr $x1-$col_halfdepA] $z8;
      node $N_A90_R   [expr $x1+$col_halfdepA] $z9;
      node $N_A90_A   $x1 [expr $z9+$beam_halfdep9];
      node $N_A90_L   [expr $x1-$col_halfdepA] $z9;

      node $N_B11_R   [expr $x2+$col_halfdepB] $z1;
      node $N_B11_A   $x2 [expr $z1+$beam_halfdep1];
      node $N_B11_L   [expr $x2-$col_halfdepB] $z1;
      node $N_B21_R   [expr $x2+$col_halfdepB] $z2;
      node $N_B21_A   $x2 [expr $z2+$beam_halfdep2];
      node $N_B21_L   [expr $x2-$col_halfdepB] $z2;
      node $N_B31_R   [expr $x2+$col_halfdepB] $z3;
      node $N_B31_A   $x2 [expr $z3+$beam_halfdep3];
      node $N_B31_L   [expr $x2-$col_halfdepB] $z3;
```

```
      node $N_B41_R   [expr $x2+$col_halfdepB] $z4;
      node $N_B41_A   $x2 [expr $z4+$beam_halfdep4];
      node $N_B41_L   [expr $x2-$col_halfdepB] $z4;
      node $N_B51_R   [expr $x2+$col_halfdepB] $z5;
      node $N_B51_A   $x2 [expr $z5+$beam_halfdep5];
      node $N_B51_L   [expr $x2-$col_halfdepB] $z5;
      node $N_B61_R   [expr $x2+$col_halfdepB] $z6;
      node $N_B61_A   $x2 [expr $z6+$beam_halfdep6];
      node $N_B61_L   [expr $x2-$col_halfdepB] $z6;
      node $N_B71_R   [expr $x2+$col_halfdepB] $z7;
      node $N_B71_A   $x2 [expr $z7+$beam_halfdep7];
      node $N_B71_L   [expr $x2-$col_halfdepB] $z7;
      node $N_B81_R   [expr $x2+$col_halfdepB] $z8;
      node $N_B81_A   $x2 [expr $z8+$beam_halfdep8];
      node $N_B81_L   [expr $x2-$col_halfdepB] $z8;
      node $N_B91_R   [expr $x2+$col_halfdepB] $z9;
      node $N_B91_A   $x2 [expr $z9+$beam_halfdep9];
      node $N_B91_L   [expr $x2-$col_halfdepB] $z9;

      node $N_C12_R   [expr $x3+$col_halfdepC] $z1;
      node $N_C12_A   $x3 [expr $z1+$beam_halfdep1];
      node $N_C12_L   [expr $x3-$col_halfdepC] $z1;
      node $N_C22_R   [expr $x3+$col_halfdepC] $z2;
      node $N_C22_A   $x3 [expr $z2+$beam_halfdep2];
      node $N_C22_L   [expr $x3-$col_halfdepC] $z2;
      node $N_C32_R   [expr $x3+$col_halfdepC] $z3;
      node $N_C32_A   $x3 [expr $z3+$beam_halfdep3];
      node $N_C32_L   [expr $x3-$col_halfdepC] $z3;
      node $N_C42_R   [expr $x3+$col_halfdepC] $z4;
      node $N_C42_A   $x3 [expr $z4+$beam_halfdep4];
      node $N_C42_L   [expr $x3-$col_halfdepC] $z4;
      node $N_C52_R   [expr $x3+$col_halfdepC] $z5;
      node $N_C52_A   $x3 [expr $z5+$beam_halfdep5];
      node $N_C52_L   [expr $x3-$col_halfdepC] $z5;
      node $N_C62_R   [expr $x3+$col_halfdepC] $z6;
      node $N_C62_A   $x3 [expr $z6+$beam_halfdep6];
      node $N_C62_L   [expr $x3-$col_halfdepC] $z6;
      node $N_C72_R   [expr $x3+$col_halfdepC] $z7;
      node $N_C72_A   $x3 [expr $z7+$beam_halfdep7];
      node $N_C72_L   [expr $x3-$col_halfdepC] $z7;
      node $N_C82_R   [expr $x3+$col_halfdepC] $z8;
      node $N_C82_A   $x3 [expr $z8+$beam_halfdep8];
      node $N_C82_L   [expr $x3-$col_halfdepC] $z8;
      node $N_C92_R   [expr $x3+$col_halfdepC] $z9;
      node $N_C92_A   $x3 [expr $z9+$beam_halfdep9];
      node $N_C92_L   [expr $x3-$col_halfdepC] $z9;

      node $N_D13_R   [expr $x4+$col_halfdepD] $z1;
      node $N_D13_A   $x4 [expr $z1+$beam_halfdep1];
      node $N_D13_L   [expr $x4-$col_halfdepD] $z1;
      node $N_D23_R   [expr $x4+$col_halfdepD] $z2;
      node $N_D23_A   $x4 [expr $z2+$beam_halfdep2];
      node $N_D23_L   [expr $x4-$col_halfdepD] $z2;
      node $N_D33_R   [expr $x4+$col_halfdepD] $z3;
      node $N_D33_A   $x4 [expr $z3+$beam_halfdep3];
      node $N_D33_L   [expr $x4-$col_halfdepD] $z3;
      node $N_D43_R   [expr $x4+$col_halfdepD] $z4;
      node $N_D43_A   $x4 [expr $z4+$beam_halfdep4];
      node $N_D43_L   [expr $x4-$col_halfdepD] $z4;
      node $N_D53_R   [expr $x4+$col_halfdepD] $z5;
      node $N_D53_A   $x4 [expr $z5+$beam_halfdep5];
      node $N_D53_L   [expr $x4-$col_halfdepD] $z5;
      node $N_D63_R   [expr $x4+$col_halfdepD] $z6;
      node $N_D63_A   $x4 [expr $z6+$beam_halfdep6];
      node $N_D63_L   [expr $x4-$col_halfdepD] $z6;
      node $N_D73_R   [expr $x4+$col_halfdepD] $z7;
      node $N_D73_A   $x4 [expr $z7+$beam_halfdep7];
      node $N_D73_L   [expr $x4-$col_halfdepD] $z7;
      node $N_D83_R   [expr $x4+$col_halfdepD] $z8;
      node $N_D83_A   $x4 [expr $z8+$beam_halfdep8];
      node $N_D83_L   [expr $x4-$col_halfdepD] $z8;
      node $N_D93_R   [expr $x4+$col_halfdepD] $z9;
      node $N_D93_A   $x4 [expr $z9+$beam_halfdep9];
      node $N_D93_L   [expr $x4-$col_halfdepD] $z9;

############### add nodes - infill walls #######################################################

       node $N_W1A_R [expr $x2*0.54] [expr $z0+($storey1*0.5)];
       node $N_W1A_L [expr $x2*0.46] [expr $z0+($storey1*0.5)];
       node $N_W2A_R [expr $x2*0.54] [expr $z1+($storey1*0.5)];
```

```
       node $N_W2A_L [expr $x2*0.46] [expr $z1+($storey1*0.5)];
       node $N_W3A_R [expr $x2*0.54] [expr $z2+($storey1*0.5)];
       node $N_W3A_L [expr $x2*0.46] [expr $z2+($storey1*0.5)];
       node $N_W4A_R [expr $x2*0.54] [expr $z3+($storey1*0.5)];
       node $N_W4A_L [expr $x2*0.46] [expr $z3+($storey1*0.5)];
       node $N_W5A_R [expr $x2*0.54] [expr $z4+($storey1*0.5)];
       node $N_W5A_L [expr $x2*0.46] [expr $z4+($storey1*0.5)];
       node $N_W6A_R [expr $x2*0.54] [expr $z5+($storey1*0.5)];
       node $N_W6A_L [expr $x2*0.46] [expr $z5+($storey1*0.5)];
       node $N_W7A_R [expr $x2*0.54] [expr $z6+($storey1*0.5)];
       node $N_W7A_L [expr $x2*0.46] [expr $z6+($storey1*0.5)];
       node $N_W8A_R [expr $x2*0.54] [expr $z7+($storey1*0.5)];
       node $N_W8A_L [expr $x2*0.46] [expr $z7+($storey1*0.5)];
       node $N_W9A_R [expr $x2*0.54] [expr $z8+($storey1*0.5)];
       node $N_W9A_L [expr $x2*0.46] [expr $z8+($storey1*0.5)];

       node $N_W1B_R [expr $x2+($span2*0.54)] [expr $z0+($storey1*0.5)];
       node $N_W1B_L [expr $x2+($span2*0.46)] [expr $z0+($storey1*0.5)];
       node $N_W2B_R [expr $x2+($span2*0.54)] [expr $z1+($storey1*0.5)];
       node $N_W2B_L [expr $x2+($span2*0.46)] [expr $z1+($storey1*0.5)];
       node $N_W3B_R [expr $x2+($span2*0.54)] [expr $z2+($storey1*0.5)];
       node $N_W3B_L [expr $x2+($span2*0.46)] [expr $z2+($storey1*0.5)];
       node $N_W4B_R [expr $x2+($span2*0.54)] [expr $z3+($storey1*0.5)];
       node $N_W4B_L [expr $x2+($span2*0.46)] [expr $z3+($storey1*0.5)];
       node $N_W5B_R [expr $x2+($span2*0.54)] [expr $z4+($storey1*0.5)];
       node $N_W5B_L [expr $x2+($span2*0.46)] [expr $z4+($storey1*0.5)];
       node $N_W6B_R [expr $x2+($span2*0.54)] [expr $z5+($storey1*0.5)];
       node $N_W6B_L [expr $x2+($span2*0.46)] [expr $z5+($storey1*0.5)];
       node $N_W7B_R [expr $x2+($span2*0.54)] [expr $z6+($storey1*0.5)];
       node $N_W7B_L [expr $x2+($span2*0.46)] [expr $z6+($storey1*0.5)];
       node $N_W8B_R [expr $x2+($span2*0.54)] [expr $z7+($storey1*0.5)];
       node $N_W8B_L [expr $x2+($span2*0.46)] [expr $z7+($storey1*0.5)];
       node $N_W9B_R [expr $x2+($span2*0.54)] [expr $z8+($storey1*0.5)];
       node $N_W9B_L [expr $x2+($span2*0.46)] [expr $z8+($storey1*0.5)];

       node $N_W1C_R [expr $x3+($span2*0.54)] [expr $z0+($storey1*0.5)];
       node $N_W1C_L [expr $x3+($span2*0.46)] [expr $z0+($storey1*0.5)];
       node $N_W2C_R [expr $x3+($span2*0.54)] [expr $z1+($storey1*0.5)];
       node $N_W2C_L [expr $x3+($span2*0.46)] [expr $z1+($storey1*0.5)];
       node $N_W3C_R [expr $x3+($span2*0.54)] [expr $z2+($storey1*0.5)];
       node $N_W3C_L [expr $x3+($span2*0.46)] [expr $z2+($storey1*0.5)];
       node $N_W4C_R [expr $x3+($span2*0.54)] [expr $z3+($storey1*0.5)];
       node $N_W4C_L [expr $x3+($span2*0.46)] [expr $z3+($storey1*0.5)];
       node $N_W5C_R [expr $x3+($span2*0.54)] [expr $z4+($storey1*0.5)];
       node $N_W5C_L [expr $x3+($span2*0.46)] [expr $z4+($storey1*0.5)];
       node $N_W6C_R [expr $x3+($span2*0.54)] [expr $z5+($storey1*0.5)];
       node $N_W6C_L [expr $x3+($span2*0.46)] [expr $z5+($storey1*0.5)];
       node $N_W7C_R [expr $x3+($span2*0.54)] [expr $z6+($storey1*0.5)];
       node $N_W7C_L [expr $x3+($span2*0.46)] [expr $z6+($storey1*0.5)];
       node $N_W8C_R [expr $x3+($span2*0.54)] [expr $z7+($storey1*0.5)];
       node $N_W8C_L [expr $x3+($span2*0.46)] [expr $z7+($storey1*0.5)];
       node $N_W9C_R [expr $x3+($span2*0.54)] [expr $z8+($storey1*0.5)];
       node $N_W9C_L [expr $x3+($span2*0.46)] [expr $z8+($storey1*0.5)];


# restraints

      #basefix  $nodetag  (ndf  $constraints)
      fix       $N_A0           1 1 1;
      fix       $N_B0           1 1 1;
      fix       $N_C0           1 1 1;
      fix       $N_D0           1 1 1;


################################################################################################################################
# material definitions
################################################################################################################################

#   Definition of materials IDs

       #set C_confinedB    1;
       set C_confined    1;
       set C_unconfined  2;
       set R_steel       3;
       set C_unconfinedw  4;



```

```
#   basic parameters for materials-con-concrete

# ConfinedConcrete01 Material

      #$tag           integer tag identifying material.
      #$secType         tag for the transverse reinforcement configuration.
      #$fpc           unconfined cylindrical strength of concrete specimen.
      #$Ec             initial elastic modulus of unconfined concrete.
      #<-epscu $epscu> OR <-gamma $gamma> confined concrete ultimate strain.
      #<-nu $nu> OR <-varub> OR <-varnoub>   Poisson's Ratio.
      #$L1            length/diameter of square/circular core section measured respect to the hoop center line.
      #($L2)            additional dimensions when multiple hoops are being used.
      #$phis         hoop diameter. If section arrangement has multiple hoops it refers to the external hoop.
      #$S             hoop spacing.
      #$fyh           yielding strength of the hoop steel.
      #$Es0             elastic modulus of the hoop steel.
      #$haRatio         hardening ratio of the hoop steel.
      #$mu            ductility factor of the hoop steel.
      #$phiLon       diameter of longitudinal bars.

#   basic parameters for materials-uncon-concrete

       set unconfc   -28.0;        # compression strength for concrete
       set unconepsc  -0.002;                 # strain at maximum stress in compression
       set unconfu   [expr $unconfc*0.18];          # ultamate stress for concrete
       set unconepsu  -0.01;                  # strain at ultimate stress in compression
       set unconlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set unconft   [expr $unconfc*-0.1];          # maximum stress in tension for concrete
       set unconEt   [expr $unconft/0.002];        # elastic modulus in tension
       set unconE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

#   basic parameters for material--steel        # ReinforcingSteel uniaxial material object. This object is intended to be used in a reinforced concrete fiber section as the steel reinforcing material.

       set Fy 420.0;    # Yield stress in tension
       set Fu 596.0;    # Ultimate stress in tension
       set Es 200000.0;    # Initial elastic tangent
       set Esh 3100.0;     # Tangent at initial strain hardening
       set esh 0.01;       # Strain corresponding to initial strain hardening
       set eult 0.09;         # Strain at peak stress

   #uniaxialMaterial ReinforcingSteel $matTag $fy $fu $Es $Esh $esh $eult    Define ReinforcingSteel uniaxial material
   uniaxialMaterial ReinforcingSteel $R_steel $Fy $Fu $Es $Esh $esh $eult -DMBuck 6 0.8 -CMFatigue 0.2600 0.5000 0.3890 -IsoHard 4.3000 0.01

#   definition of ConfinedConcrete01 material

    #uniaxialMaterial ConfinedConcrete01 $tag        $secType $fpc  $Ec   -epscu $epscu   $nu    $L1   $L2    $phis  $S   $fyh  $Es0     $haRatio  $mu  $phiLon  -stRatio $stRatio
    #uniaxialMaterial ConfinedConcrete01  $C_confinedB     R      -28 24870.1 -epscu -0.04   -varUB 250.0 1450.0 10.0  125.0 420.0 200000.0 0.00      3100.0 12.0    -stRatio 0.85
    #uniaxialMaterial ConfinedConcrete01  $C_confinedC     R      -28 24870.1 -epscu -0.04   -varUB 550.0 200.0  10.0  125.0 420.0 200000.0 0.00      3100.0 18.0    -stRatio 0.85


#   basic parameters for materials-con-concrete

       set confc   -32.5;        # compression strength for concrete
       set conepsc  -0.003;                 # strain at maximum stress in compression
       set confu   [expr $unconfc*0.18];          # ultamate stress for concrete
       set conepsu  -0.04;                  # strain at ultimate stress in compression
       set conlambda     0.1;               # ratio between reloading stiffness and itial stiffness in compression
       set conft   [expr $unconfc*-0.1];          # maximum stress in tension for concrete
       set conEt   [expr $unconft/0.002];        # elastic modulus in tension
       set conE0   [expr 2*$unconfc/$unconepsc];      #intial elastic tangent

    # uniaxialMaterial Concrete02  $matTag     $fpc $epsc0 $fpcu  $epsU  $lambda  $ft  $Ets
    uniaxialMaterial   Concrete02  $C_unconfined $unconfc  $unconepsc  $unconfu    $unconepsu  $unconlambda  $unconft  $unconEt;
    uniaxialMaterial   Concrete02  $C_confined $confc  $conepsc  $confu    $conepsu  $conlambda  $conft  $conEt;

################################################################################################################################################
# definition of the Sections
################################################################################################################################################

#   define sections IDs

    set col40x60  1;
```

```
    set beam150x30 2;

#   define section parameters

    set pi        3.141593;
    set rebar_12  [expr $pi*12.0*12.0/4];  # area rebar 12mm
    set rebar_16  [expr $pi*16.0*16.0/4];
    set w_col     400.0;   # column width
    set h_col     600.0;   # column hieght
    set c_col     20.0;  # column cover
    set w_beam    1500.0;  # beam width
    set h_beam    300.0;   # beam hieght
    set c_beam    30.0;  # beam cover

#   load procedure for fiber section

source BuildRCrectSection.tcl;

#   build sections

    #BuildRCrectSection $ColSecTag $HSec   $BSec   $coverH  $coverB  $coreID      $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
    BuildRCrectSection  $col40x60  $h_col  $w_col  $c_col   $c_col   $C_confined $C_unconfined $R_steel 9           $rebar_16   9           $rebar_16   2               $rebar_16    8         8         8          8
    BuildRCrectSection  $beam150x30 $h_beam $w_beam $c_beam  $c_beam $C_confined $C_unconfined $R_steel 12          $rebar_12   8          $rebar_12   0               $rebar_12    8         8         8          8


##############################################################################################################################
# beam column joint definition
##############################################################################################################################

# dimensions of the joint respectively
set JointWidth [expr $h_col]; set JointHeight [expr $h_beam]; set JointDepth $w_col ;
set JointVolume [expr $JointWidth*$JointHeight*$JointDepth];

########################## details for the material models of bar slip of the beam ####################################

set bs_fc 28.0; set bs_fs 420.0; set bs_es 200000; set bs_fsu 596; set bs_dbar 12.0; set bs_esh 3100.0;
set bs_wid $w_col; set bs_dep $h_beam;
set bsT_nbars 12; set bsB_nbars 8;
set bs_ljoint $h_col;

########################### details for the material models of bar slip of the column ###################################

set cs_fc 28.0; set cs_fs 420.0; set cs_es 200000.0; set cs_fsu 596; set cs_dbar 16.0; set cs_esh 3100.0;
set cs_wid $w_col; set cs_dep $h_col;
set cs_nbars 9;
set cs_ljoint $h_beam;

##########################################################################################################################
#bar slip definition

# for beam bottom

set bsid1 11
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid1 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsB_nbars $bs_wid $bs_dep  strong beambot

# for beam top

set bsid2 21
#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid2 $bs_fc $bs_fs $bs_es $bs_fsu $bs_esh $bs_dbar $bs_ljoint $bsT_nbars $bs_wid $bs_dep  strong beamtop

# for columns
set bsid3 31

#uniaxialMaterial BarSlip $matTag $fc $fy $Es $fu $Eh $db $ld $nb $depth $height <$ancLratio> $bsFlag $type <$damage $unit>

uniaxialMaterial BarSlip $bsid3 $cs_fc $cs_fs $cs_es $cs_fsu $cs_esh $cs_dbar $cs_ljoint $cs_nbars $cs_wid $cs_dep  strong column
```

```
############################ material for shear panel ##############################################################

## Positive/Negative envelope Stress

set spid1 41;
set A 0.78;
set p1 [expr 2.539*$A]; set p2 [expr 3.393*$A]; set p3 [expr 3.57*$A]; set p4 [expr 0.7143*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1*$JointVolume] [expr $p2*$JointVolume] [expr $p3*$JointVolume] [expr $p4*$JointVolume]]
set nEnvStrsp [list [expr -$p1*$JointVolume] [expr -$p2*$JointVolume] [expr -$p3*$JointVolume] [expr -$p4*$JointVolume]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.0008 0.015  0.035 0.04]
set nEnvStnsp [list -0.0008 -0.015  -0.035 -0.04]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $spid1 [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy

################### beam column joint ###########################################

##element BeamColumnJoint tag? iNode? jNode? kNode? lNode? matTag1? matTag2? matTag3? matTag4?
## matTag5? matTag6? matTag7? matTag8? matTag9? matTag10? matTag11? matTag12? matTag13?
## <element Height factor?> <element Width factor?>
## please note: the four nodes are in anticlockwise direction around the element
## requires material tags for all 13 different components within the element.
## the first 12 being that of spring and the last of the shear panel

set jointA1 611
set jointA2 612
```

```
set jointA3 613
set jointA4 614
set jointA5 615
set jointA6 616
set jointA7 617
set jointA8 618
set jointA9 619

set jointB1 621
set jointB2 622
set jointB3 623
set jointB4 624
set jointB5 625
set jointB6 626
set jointB7 627
set jointB8 628
set jointB9 629

set jointC1 631
set jointC2 632
set jointC3 633
set jointC4 634
set jointC5 635
set jointC6 636
set jointC7 637
set jointC8 638
set jointC9 639

set jointD1 641
set jointD2 642
set jointD3 643
set jointD4 644
set jointD5 645
set jointD6 646
set jointD7 647
set jointD8 648
set jointD9 649

# add material Properties - command: uniaxialMaterial matType matTag ...
#command: uniaxialMaterial Elastic tag? E?

uniaxialMaterial Elastic 71 10000000000.0

element beamColumnJoint $jointA1 $N_A1 $N_A10_R $N_A10_A $N_A10_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA2 $N_A2 $N_A20_R $N_A20_A $N_A20_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA3 $N_A3 $N_A30_R $N_A30_A $N_A30_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA4 $N_A4 $N_A40_R $N_A40_A $N_A40_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA5 $N_A5 $N_A50_R $N_A50_A $N_A50_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA6 $N_A6 $N_A60_R $N_A60_A $N_A60_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA7 $N_A7 $N_A70_R $N_A70_A $N_A70_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA8 $N_A8 $N_A80_R $N_A80_A $N_A80_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointA9 $N_A9 $N_A90_R $N_A90_A $N_A90_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointB1 $N_B1 $N_B11_R $N_B11_A $N_B11_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB2 $N_B2 $N_B21_R $N_B21_A $N_B21_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB3 $N_B3 $N_B31_R $N_B31_A $N_B31_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB4 $N_B4 $N_B41_R $N_B41_A $N_B41_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB5 $N_B5 $N_B51_R $N_B51_A $N_B51_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB6 $N_B6 $N_B61_R $N_B61_A $N_B61_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB7 $N_B7 $N_B71_R $N_B71_A $N_B71_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB8 $N_B8 $N_B81_R $N_B81_A $N_B81_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointB9 $N_B9 $N_B91_R $N_B91_A $N_B91_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
```

```

element beamColumnJoint $jointC1 $N_C1 $N_C12_R $N_C12_A $N_C12_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC2 $N_C2 $N_C22_R $N_C22_A $N_C22_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC3 $N_C3 $N_C32_R $N_C32_A $N_C32_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC4 $N_C4 $N_C42_R $N_C42_A $N_C42_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC5 $N_C5 $N_C52_R $N_C52_A $N_C52_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC6 $N_C6 $N_C62_R $N_C62_A $N_C62_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC7 $N_C7 $N_C72_R $N_C72_A $N_C72_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC8 $N_C8 $N_C82_R $N_C82_A $N_C82_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointC9 $N_C9 $N_C92_R $N_C92_A $N_C92_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1

element beamColumnJoint $jointD1 $N_D1 $N_D13_R $N_D13_A $N_D13_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD2 $N_D2 $N_D23_R $N_D23_A $N_D23_L $bsid3 $bsid3 71  $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD3 $N_D3 $N_D33_R $N_D33_A $N_D33_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD4 $N_D4 $N_D43_R $N_D43_A $N_D43_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD5 $N_D5 $N_D53_R $N_D53_A $N_D53_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD6 $N_D6 $N_D63_R $N_D63_A $N_D63_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD7 $N_D7 $N_D73_R $N_D73_A $N_D73_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD8 $N_D8 $N_D83_R $N_D83_A $N_D83_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1
element beamColumnJoint $jointD9 $N_D9 $N_D93_R $N_D93_A $N_D93_L $bsid3 $bsid3 71 $bsid1 $bsid2 71 $bsid3 $bsid3 71 $bsid1 $bsid2 71 $spid1


#########################################################################################################################
# Elements definitions
#########################################################################################################################

# COLUMN definition

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
set ColTransfTag 1;        # associate a tag to column transformation
geomTransf PDelta $ColTransfTag ; #Columns

# --------------------------------------------------------------------
# ---- element connectivity "Columns Definition"----------
# --------------------------------------------------------------------
set numIntPoints 4;
set integrationC "Lobatto $col40x60 $numIntPoints"

element forceBeamColumn 710      $N_A0   $N_A1 $ColTransfTag $integrationC
element forceBeamColumn 720      $N_A10_A   $N_A2 $ColTransfTag $integrationC
element forceBeamColumn 730      $N_A20_A   $N_A3 $ColTransfTag $integrationC
element forceBeamColumn 740      $N_A30_A   $N_A4 $ColTransfTag $integrationC
element forceBeamColumn 750      $N_A40_A   $N_A5 $ColTransfTag $integrationC
element forceBeamColumn 760      $N_A50_A   $N_A6 $ColTransfTag $integrationC
element forceBeamColumn 770      $N_A60_A   $N_A7 $ColTransfTag $integrationC
element forceBeamColumn 780      $N_A70_A   $N_A8 $ColTransfTag $integrationC
element forceBeamColumn 790      $N_A80_A   $N_A9 $ColTransfTag $integrationC

element forceBeamColumn 711      $N_B0   $N_B1 $ColTransfTag $integrationC
element forceBeamColumn 721      $N_B11_A   $N_B2 $ColTransfTag $integrationC
element forceBeamColumn 731      $N_B21_A   $N_B3 $ColTransfTag $integrationC
element forceBeamColumn 741      $N_B31_A   $N_B4 $ColTransfTag $integrationC
element forceBeamColumn 751      $N_B41_A   $N_B5 $ColTransfTag $integrationC
element forceBeamColumn 761      $N_B51_A   $N_B6 $ColTransfTag $integrationC
element forceBeamColumn 771      $N_B61_A   $N_B7 $ColTransfTag $integrationC
element forceBeamColumn 781      $N_B71_A   $N_B8 $ColTransfTag $integrationC
element forceBeamColumn 791      $N_B81_A   $N_B9 $ColTransfTag $integrationC

element forceBeamColumn 712      $N_C0   $N_C1 $ColTransfTag $integrationC
element forceBeamColumn 722      $N_C12_A   $N_C2 $ColTransfTag $integrationC
```

```
element forceBeamColumn 732      $N_C22_A   $N_C3 $ColTransfTag $integrationC
element forceBeamColumn 742      $N_C32_A   $N_C4 $ColTransfTag $integrationC
element forceBeamColumn 752      $N_C42_A   $N_C5 $ColTransfTag $integrationC
element forceBeamColumn 762      $N_C52_A   $N_C6 $ColTransfTag $integrationC
element forceBeamColumn 772      $N_C62_A   $N_C7 $ColTransfTag $integrationC
element forceBeamColumn 782      $N_C72_A   $N_C8 $ColTransfTag $integrationC
element forceBeamColumn 792      $N_C82_A   $N_C9 $ColTransfTag $integrationC

element forceBeamColumn 713      $N_D0   $N_D1 $ColTransfTag $integrationC
element forceBeamColumn 723      $N_D13_A   $N_D2 $ColTransfTag $integrationC
element forceBeamColumn 733      $N_D23_A   $N_D3 $ColTransfTag $integrationC
element forceBeamColumn 743      $N_D33_A   $N_D4 $ColTransfTag $integrationC
element forceBeamColumn 753      $N_D43_A   $N_D5 $ColTransfTag $integrationC
element forceBeamColumn 763      $N_D53_A   $N_D6 $ColTransfTag $integrationC
element forceBeamColumn 773      $N_D63_A   $N_D7 $ColTransfTag $integrationC
element forceBeamColumn 783      $N_D73_A   $N_D8 $ColTransfTag $integrationC
element forceBeamColumn 793      $N_D83_A   $N_D9 $ColTransfTag $integrationC


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

# BEAMS definition

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
set BeamTransfTag 2;          # associate a tag to beam transformation
geomTransf PDelta $BeamTransfTag  ; #Beams

# --------------------------------------------------------------------
# ---- element connectivity "Beamss Definition"----------
# --------------------------------------------------------------------
set numIntPoints_beams 5;
set integrationB "Lobatto $beam150x30 $numIntPoints_beams"

element forceBeamColumn  810    $N_A10_R $N_B11_L  $BeamTransfTag $integrationB
element forceBeamColumn  820    $N_A20_R $N_B21_L  $BeamTransfTag $integrationB
element forceBeamColumn  830    $N_A30_R $N_B31_L  $BeamTransfTag $integrationB
element forceBeamColumn  840    $N_A40_R $N_B41_L  $BeamTransfTag $integrationB
element forceBeamColumn  850    $N_A50_R $N_B51_L  $BeamTransfTag $integrationB
element forceBeamColumn  860    $N_A60_R $N_B61_L  $BeamTransfTag $integrationB
element forceBeamColumn  870    $N_A70_R $N_B71_L  $BeamTransfTag $integrationB
element forceBeamColumn  880    $N_A80_R $N_B81_L  $BeamTransfTag $integrationB
element forceBeamColumn  890    $N_A90_R $N_B91_L  $BeamTransfTag $integrationB

element forceBeamColumn  811    $N_B11_R $N_C12_L  $BeamTransfTag $integrationB
element forceBeamColumn  821    $N_B21_R $N_C22_L  $BeamTransfTag $integrationB
element forceBeamColumn  831    $N_B31_R $N_C32_L  $BeamTransfTag $integrationB
element forceBeamColumn  841    $N_B41_R $N_C42_L  $BeamTransfTag $integrationB
element forceBeamColumn  851    $N_B51_R $N_C52_L  $BeamTransfTag $integrationB
element forceBeamColumn  861    $N_B61_R $N_C62_L  $BeamTransfTag $integrationB
element forceBeamColumn  871    $N_B71_R $N_C72_L  $BeamTransfTag $integrationB
element forceBeamColumn  881    $N_B81_R $N_C82_L  $BeamTransfTag $integrationB
element forceBeamColumn  891    $N_B91_R $N_C92_L  $BeamTransfTag $integrationB

element forceBeamColumn  812    $N_C12_R $N_D13_L  $BeamTransfTag $integrationB
element forceBeamColumn  822    $N_C22_R $N_D23_L  $BeamTransfTag $integrationB
element forceBeamColumn  832    $N_C32_R $N_D33_L  $BeamTransfTag $integrationB
element forceBeamColumn  842    $N_C42_R $N_D43_L  $BeamTransfTag $integrationB
element forceBeamColumn  852    $N_C52_R $N_D53_L  $BeamTransfTag $integrationB
element forceBeamColumn  862    $N_C62_R $N_D63_L  $BeamTransfTag $integrationB
element forceBeamColumn  872    $N_C72_R $N_D73_L  $BeamTransfTag $integrationB
element forceBeamColumn  882    $N_C82_R $N_D83_L  $BeamTransfTag $integrationB
element forceBeamColumn  892    $N_C92_R $N_D93_L  $BeamTransfTag $integrationB


##########################################################################################################################
# infill walls definitions
##########################################################################################################################

############ METHOD A ##########################################################

# diagonal members

set dia1A    101;
set dia2A    102;
set dia3A    103;
set dia4A    104;
set dia5A    105;
set dia6A    106;
```

```
set dia7A    107;
set dia8A    108;
set dia9A    109;
set dia10A  1010;
set dia11A  1011;
set dia12A  1012;
set dia13A  1013;
set dia14A  1014;
set dia15A  1015;
set dia16A  1016;
set dia17A  1017;
set dia18A  1018;
set dia19A  1019;
set dia20A  1020;
set dia21A  1021;
set dia22A  1022;
set dia23A  1023;
set dia24A  1024;
set dia25A  1025;
set dia26A  1026;
set dia27A  1027;
set dia28A  1028;
set dia29A  1029;
set dia30A  1030;
set dia31A  1031;
set dia32A  1032;
set dia33A  1033;
set dia34A  1034;
set dia35A  1035;
set dia36A  1036;


set dia1B   1037;
set dia2B   1038;
set dia3B   1039;
set dia4B   1040;
set dia5B   1041;
set dia6B   1042;
set dia7B   1043;
set dia8B   1044;
set dia9B   1045;
set dia10B  1046;
set dia11B  1047;
set dia12B  1048;
set dia13B  1049;
set dia14B  1050;
set dia15B  1051;
set dia16B  1052;
set dia17B  1053;
set dia18B  1054;
set dia19B  1055;
set dia20B  1056;
set dia21B  1057;
set dia22B  1058;
set dia23B  1059;
set dia24B  1060;
set dia25B  1061;
set dia26B  1062;
set dia27B  1063;
set dia28B  1064;
set dia29B  1065;
set dia30B  1066;
set dia31B  1067;
set dia32B  1068;
set dia33B  1069;
set dia34B  1070;
set dia35B  1071;
set dia36B  1072;

set dia1C    1073;
set dia2C    1074;
set dia3C    1075;
set dia4C    1076;
set dia5C    1077;
set dia6C    1078;
set dia7C    1079;
set dia8C    1080;
set dia9C    1081;
set dia10C  1082;
set dia11C  1083;
```

```
set dia12C  1084;
set dia13C  1085;
set dia14C  1086;
set dia15C  1087;
set dia16C  1088;
set dia17C  1089;
set dia18C  1090;
set dia19C  1091;
set dia20C  1092;
set dia21C  1093;
set dia22C  1094;
set dia23C  1095;
set dia24C  1096;
set dia25C  1097;
set dia26C  1098;
set dia27C  1099;
set dia28C  1100;
set dia29C  1101;
set dia30C  1102;
set dia31C  1103;
set dia32C  1104;
set dia33C  1105;
set dia34C  1106;
set dia35C  1107;
set dia36C  1108;

set width_wall 665;      #using equation 16 in the report, the width of the strut based on H=3m,L=4m, and t=0.3m
set t_wall 200;
set  Aw [expr $width_wall*$t_wall];   #cross-sectional
set Ew   10000000000.0;   #Young's Modulus
set Izw [expr $t_wall*(pow($width_wall,3))/12]; #second moment of area about the local z-axis

set WtransfTag 81;
geomTransf Linear $WtransfTag;

# -------------------------------------------------------------------
# ---- element connectivity "diagonal infill walls Definition"----------
# -------------------------------------------------------------------

element elasticBeamColumn $dia1A $N_A1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2A $N_B1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3A $N_A0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4A $N_B0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia5A $N_A2 $N_W2A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia6A $N_B2 $N_W2A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia7A $N_A10_A $N_W2A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia8A $N_B11_A $N_W2A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia9A $N_A3 $N_W3A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia10A $N_B3 $N_W3A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia11A $N_A20_A $N_W3A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia12A $N_B21_A $N_W3A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia13A $N_A4 $N_W4A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia14A $N_B4 $N_W4A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia15A $N_A30_A $N_W4A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia16A $N_B31_A $N_W4A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia17A $N_A5 $N_W5A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia18A $N_B5 $N_W5A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia19A $N_A40_A $N_W5A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia20A $N_B41_A $N_W5A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia21A $N_A6 $N_W6A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia22A $N_B6 $N_W6A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia23A $N_A50_A $N_W6A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia24A $N_B51_A $N_W6A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia25A $N_A7 $N_W7A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia26A $N_B7 $N_W7A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia27A $N_A60_A $N_W7A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia28A $N_B61_A $N_W7A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia29A $N_A8 $N_W8A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia30A $N_B8 $N_W8A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia31A $N_A70_A $N_W8A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia32A $N_B71_A $N_W8A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia33A $N_A9 $N_W9A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia34A $N_B9 $N_W9A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia35A $N_A80_A $N_W9A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia36A $N_B81_A $N_W9A_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1B $N_B1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2B $N_C1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3B $N_B0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4B $N_C0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
```

```
element elasticBeamColumn $dia5B $N_B2 $N_W2B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia6B $N_C2 $N_W2B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia7B $N_B11_A $N_W2B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia8B $N_C12_A $N_W2B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia9B $N_B3 $N_W3B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia10B $N_C3 $N_W3B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia11B $N_B21_A $N_W3B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia12B $N_C22_A $N_W3B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia13B $N_B4 $N_W4B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia14B $N_C4 $N_W4B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia15B $N_B31_A $N_W4B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia16B $N_C32_A $N_W4B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia17B $N_B5 $N_W5B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia18B $N_C5 $N_W5B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia19B $N_B41_A $N_W5B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia20B $N_C42_A $N_W5B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia21B $N_B6 $N_W6B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia22B $N_C6 $N_W6B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia23B $N_B51_A $N_W6B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia24B $N_C52_A $N_W6B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia25B $N_B7 $N_W7B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia26B $N_C7 $N_W7B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia27B $N_B61_A $N_W7B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia28B $N_C62_A $N_W7B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia29B $N_B8 $N_W8B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia30B $N_C8 $N_W8B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia31B $N_B71_A $N_W8B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia32B $N_C72_A $N_W8B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia33B $N_B9 $N_W9B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia34B $N_C9 $N_W9B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia35B $N_B81_A $N_W9B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia36B $N_C82_A $N_W9B_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1C $N_C1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2C $N_D1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3C $N_C0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4C $N_D0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia5C $N_C2 $N_W2C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia6C $N_D2 $N_W2C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia7C $N_C12_A $N_W2C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia8C $N_D13_A $N_W2C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia9C $N_C3 $N_W3C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia10C $N_D3 $N_W3C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia11C $N_C22_A $N_W3C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia12C $N_D23_A $N_W3C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia13C $N_C4 $N_W4C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia14C $N_D4 $N_W4C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia15C $N_C32_A $N_W4C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia16C $N_D33_A $N_W4C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia17C $N_C5 $N_W5C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia18C $N_D5 $N_W5C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia19C $N_C42_A $N_W5C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia20C $N_D43_A $N_W5C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia21C $N_C6 $N_W6C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia22C $N_D6 $N_W6C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia23C $N_C52_A $N_W6C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia24C $N_D53_A $N_W6C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia25C $N_C7 $N_W7C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia26C $N_D7 $N_W7C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia27C $N_C62_A $N_W7C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia28C $N_D63_A $N_W7C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia29C $N_C8 $N_W8C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia30C $N_D8 $N_W8C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia31C $N_C72_A $N_W8C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia32C $N_D73_A $N_W8C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia33C $N_C9 $N_W9C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia34C $N_D9 $N_W9C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia35C $N_C82_A $N_W9C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia36C $N_D83_A $N_W9C_R $Aw $Ew $Izw $WtransfTag

#  Central member

set cen1A 2001;
set cen2A 2002;
set cen3A 2003;
set cen4A 2004;
set cen5A 2005;
set cen6A 2006;
set cen7A 2007;
set cen8A 2008;
```

```
set cen9A 2009;
set cen1B 20010;
set cen2B 20011;
set cen3B 20012;
set cen4B 20013;
set cen5B 20014;
set cen6B 20015;
set cen7B 20016;
set cen8B 20017;
set cen9B 20018;
set cen1C 20019;
set cen2C 20020;
set cen3C 20021;
set cen4C 20022;
set cen5C 20023;
set cen6C 20024;
set cen7C 20025;
set cen8C 20026;
set cen9C 20027;

# ----------------------------------------------------------
# Define geometric transformation
# ----------------------------------------------------------
#set wallTransfTag 82;        # associate a tag to wall transformation
#geomTransf Linear $wallTransfTag  ; #walls

# --------------------------------------------------------------------
# ---- element connectivity "wall Definition"----------
# --------------------------------------------------------------------
#set numIntPoints_wall 2;

set wall_sec 91;
set wall_mat 92;





########################### material for infill walls ##############################################################

## Positive/Negative envelope Stress

set A 1;
set p1 [expr 0.4*$A]; set p2 [expr 1.025*$A]; set p3 [expr 2.05*$A]; set p4 [expr 0.41*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1] [expr $p2] [expr $p3] [expr $p4]]
set nEnvStrsp [list [expr -$p1] [expr -$p2] [expr -$p3] [expr -$p4]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.000065 0.00385  0.00771 0.0120]
set nEnvStnsp [list -0.000065 -0.00385  -0.00771 -0.0120]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
```

```
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $wall_mat [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy

###################################### wall section ############################################################################

#BuildRCrectSection $ColSecTag $HSec        $BSec   $coverH  $coverB  $coreID     $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
BuildRCrectSection  $wall_sec  $width_wall  $t_wall  20       20       $wall_mat   $wall_mat      $wall_mat 0           $rebar_16   0           $rebar_16   0              $rebar_16   8        8        8         8

set wallTransfTag 3;          # associate a tag to column transformation
geomTransf PDelta $wallTransfTag ; #Columns

# --------------------------------------------------------------------
# ---- element connectivity "Columns Definition"----------
# --------------------------------------------------------------------
set numIntPoints_wall 4;
set integrationw "Lobatto $wall_sec $numIntPoints_wall"

element forceBeamColumn $cen1A     $N_W1A_L $N_W1A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen2A     $N_W2A_L $N_W2A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen3A     $N_W3A_L $N_W3A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen4A     $N_W4A_L $N_W4A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen5A     $N_W5A_L $N_W5A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen6A     $N_W6A_L $N_W6A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen7A     $N_W7A_L $N_W7A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen8A     $N_W8A_L $N_W8A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen9A     $N_W9A_L $N_W9A_R    $wallTransfTag $integrationw

element forceBeamColumn $cen1B     $N_W1B_L $N_W1B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen2B     $N_W2B_L $N_W2B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen3B     $N_W3B_L $N_W3B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen4B     $N_W4B_L $N_W4B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen5B     $N_W5B_L $N_W5B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen6B     $N_W6B_L $N_W6B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen7B     $N_W7B_L $N_W7B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen8B     $N_W8B_L $N_W8B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen9B     $N_W9B_L $N_W9B_R    $wallTransfTag $integrationw

element forceBeamColumn $cen1C     $N_W1C_L $N_W1C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen2C     $N_W2C_L $N_W2C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen3C     $N_W3C_L $N_W3C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen4C     $N_W4C_L $N_W4C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen5C     $N_W5C_L $N_W5C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen6C     $N_W6C_L $N_W6C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen7C     $N_W7C_L $N_W7C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen8C     $N_W8C_L $N_W8C_R    $wallTransfTag $integrationw
element forceBeamColumn $cen9C     $N_W9C_L $N_W9C_R    $wallTransfTag $integrationw



############ METHOD B ##########################################################  Can be used in case of convergence problems (Validated Method Experimentally)


#set width_wall 665;      #using equation 16 in the report, the width of the strut based on H=3m,L=4m, and t=0.3m
```

```
#set t_wall 200;
#set  Aw [expr $width_wall*$t_wall];   #cross-sectional
#set Ew  10000000000.0;   #Young's Modulus
#set Izw [expr $t_wall*(pow($width_wall,3))/12];   #second moment of area about the local z-axis


#  Central member

#set cenA1 111;                       #A: means that the diagonal strut start from axis A to B, and the rest in that analogy.
#set cenA2 112;
#set cenA3 113;
#set cenA4 114;
#set cenA5 115;
#set cenA6 116;
#set cenA7 117;
#set cenA8 118;
#set cenA9 119;

#set cenB1 121;
#set cenB2 122;
#set cenB3 123;
#set cenB4 124;
#set cenB5 125;
#set cenB6 126;
#set cenB7 127;
#set cenB8 128;
#set cenB9 129;

#set cenC1 131;
#set cenC2 132;
#set cenC3 133;
#set cenC4 134;
#set cenC5 135;
#set cenC6 136;
#set cenC7 137;
#set cenC8 138;
#set cenC9 139;

# --------------------------------------------------------------------
# ---- element connectivity "wall Definition"----------
# --------------------------------------------------------------------

#set wall_sec 91;
#set wall_mat 92;





########################## material for infill walls #############################################################

## Positive/Negative envelope Stress

#       set unconfcw   -2.05;        # compression strength for concrete
#       set unconepscw  -0.000352;                 # strain at maximum stress in compression
#       set unconfuw   [expr $unconfcw*0.4];          # ultamate stress for concrete
#       set unconepsuw  -0.012;                  # strain at ultimate stress in compression
#       set unconlambdaw     0.1;                # ratio between reloading stiffness and itial stiffness in compression
#       set unconftw   [expr $unconfcw*-0.0001];         # maximum stress in tension for concrete
#       set unconEtw   [expr $unconftw/0.2];        # elastic modulus in tension
#       set unconE0w   8000;      #intial elastic tangent

#    uniaxialMaterial   Concrete02  $C_unconfinedw $unconfcw  $unconepscw  $unconfuw    $unconepsuw  $unconlambdaw  $unconftw  $unconEtw;



##################################### wall section ########################################################################

#BuildRCrectSection $ColSecTag $HSec        $BSec   $coverH  $coverB  $coreID     $coverID      $steelID $numBarsTop $barAreaTop $numBarsBot $barAreaBot $numBarsIntTot $barAreaInt $nfCoreY $nfCoreZ $nfCoverY $nfCoverZ
#BuildRCrectSection  $wall_sec  $width_wall  $t_wall  20       20       $C_unconfinedw   $C_unconfinedw     $C_unconfinedw 0           $rebar_16   0           $rebar_16   0              $rebar_16   8        8        8         8

#set wallTransfTag 3;         # associate a tag to column transformation
#geomTransf PDelta $wallTransfTag ; #Columns


```

```
# --------------------------------------------------------------------
# ---- element connectivity "Wall Definition"----------
# --------------------------------------------------------------------
#set numIntPoints_wall 4;
#set integrationw "Lobatto $wall_sec $numIntPoints_wall"

#element forceBeamColumn $cenA1    $N_A1 $N_B0       $wallTransfTag $integrationw
#element forceBeamColumn $cenA2    $N_A2 $N_B11_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA3    $N_A3 $N_B21_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA4    $N_A4 $N_B31_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA5    $N_A5 $N_B41_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA6    $N_A6 $N_B51_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA7    $N_A7 $N_B61_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA8    $N_A8 $N_B71_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenA9    $N_A9 $N_B81_A    $wallTransfTag $integrationw

#element forceBeamColumn $cenB1    $N_B1 $N_C0       $wallTransfTag $integrationw
#element forceBeamColumn $cenB2    $N_B2 $N_C12_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB3    $N_B3 $N_C22_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB4    $N_B4 $N_C32_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB5    $N_B5 $N_C42_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB6    $N_B6 $N_C52_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB7    $N_B7 $N_C62_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB8    $N_B8 $N_C72_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenB9    $N_B9 $N_C82_A    $wallTransfTag $integrationw

#element forceBeamColumn $cenC1    $N_C1 $N_D0       $wallTransfTag $integrationw
#element forceBeamColumn $cenC2    $N_C2 $N_D13_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC3    $N_C3 $N_D23_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC4    $N_C4 $N_D33_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC5    $N_C5 $N_D43_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC6    $N_C6 $N_D53_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC7    $N_C7 $N_D63_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC8    $N_C8 $N_D73_A    $wallTransfTag $integrationw
#element forceBeamColumn $cenC9    $N_C9 $N_D83_A    $wallTransfTag $integrationw




###################################################################################################################################
# display the model with the node numbers
   DisplayModel2D NodeNumbers

###################################################################################################################################
# gravity and masses load
###################################################################################################################################

    # timeSeries  "LinearDefault":   tsTag   cFactor
    timeSeries          Linear        1     -factor   1;

    # distributed loads

   #set   DL   11000.0;             # self weight add as point load (N)
    set   TLE   68100.0;            # TLE: Total Load at the middle columns
    set   TLM   136100.0;            # TLM: Total Load at the middle columns

    # pattern  PatternType  $PatternID  TimeSeriesType
    pattern       Plain          1             1       {
    #load  $nodeTag  (ndf  $LoadValues)
    load   $N_A10_A 0    [expr -$TLE]    0;
    load   $N_A20_A 0    [expr -$TLE]    0;
    load   $N_A30_A 0    [expr -$TLE]    0;
    load   $N_A40_A 0    [expr -$TLE]    0;
    load   $N_A50_A 0    [expr -$TLE]    0;
    load   $N_A60_A 0    [expr -$TLE]    0;
    load   $N_A70_A 0    [expr -$TLE]    0;
    load   $N_A80_A 0    [expr -$TLE]    0;
    load   $N_A90_A 0    [expr -$TLE]    0;

    load   $N_B11_A 0    [expr -$TLM]    0;
    load   $N_B21_A 0    [expr -$TLM]    0;
    load   $N_B31_A 0    [expr -$TLM]    0;
    load   $N_B41_A 0    [expr -$TLM]    0;
    load   $N_B51_A 0    [expr -$TLM]    0;
    load   $N_B61_A 0    [expr -$TLM]    0;
    load   $N_B71_A 0    [expr -$TLM]    0;
    load   $N_B81_A 0    [expr -$TLM]    0;
```

```
        load   $N_B91_A 0    [expr -$TLM]    0;

        load   $N_C12_A 0    [expr -$TLM]    0;
        load   $N_C22_A 0    [expr -$TLM]    0;
        load   $N_C32_A 0    [expr -$TLM]    0;
        load   $N_C42_A 0    [expr -$TLM]    0;
        load   $N_C52_A 0    [expr -$TLM]    0;
        load   $N_C62_A 0    [expr -$TLM]    0;
        load   $N_C72_A 0    [expr -$TLM]    0;
        load   $N_C82_A 0    [expr -$TLM]    0;
        load   $N_C92_A 0    [expr -$TLM]    0;

        load   $N_D13_A 0    [expr -$TLE]    0;
        load   $N_D23_A 0    [expr -$TLE]    0;
        load   $N_D33_A 0    [expr -$TLE]    0;
        load   $N_D43_A 0    [expr -$TLE]    0;
        load   $N_D53_A 0    [expr -$TLE]    0;
        load   $N_D63_A 0    [expr -$TLE]    0;
        load   $N_D73_A 0    [expr -$TLE]    0;
        load   $N_D83_A 0    [expr -$TLE]    0;
        load   $N_D93_A 0    [expr -$TLE]    0;

        #eleLoad  -ele  $eleTag1  <$eleTag2>  -type  -beamuniformload $wy
        #eleLoad   -ele     5           6      -type  -beamUniform     [expr -$DL];

        }


    # masses

          set mass1  20420;
          set mass2  20420;
          set mass3  20420;
          set mass4  20420;
          set mass5  20420;
          set mass6  20420;
          set mass7  20420;
          set mass8  20420;
          set mass9  20420;



         # assign mass to nodes

         #mass  $nodetag  (ndf  $massvalues)
          mass   $N_A10_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A20_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A30_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A40_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A50_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A60_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A70_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A80_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_A90_L    [expr $mass1/2]  0.1  0.1;

          mass   $N_B11_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B21_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B31_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B41_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B51_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B61_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B71_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B81_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_B91_L    [expr $mass1/2]  0.1  0.1;

          mass   $N_C12_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C22_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C32_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C42_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C52_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C62_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C72_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C82_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_C92_L    [expr $mass1/2]  0.1  0.1;

          mass   $N_D13_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_D23_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_D33_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_D43_L    [expr $mass1/2]  0.1  0.1;
          mass   $N_D53_L    [expr $mass1/2]  0.1  0.1;
```

```
      mass   $N_D63_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D73_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D83_L    [expr $mass1/2]  0.1  0.1;
      mass   $N_D93_L    [expr $mass1/2]  0.1  0.1;



puts "Model Built"


3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 7

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>            dof   resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A8  $N_A9   -dof  1  disp
recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;               # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860 870 880 890 localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 617 618 619 621 622 623 624 625 626 627 628 629 shearpanel stressStrain
recorder  Element   -file $dataDir/Strut1.out    -time    -ele 111 112 113 114 115 116 117 118 119 121 122 123 124 125 126 127 128 129 section 1  fiber y z  stressStrain;
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
   set ViewScale 5;
   DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1000;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.1;    # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 10;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;         # define load pattern -- generalized
      load $N_A9 9 0 0
        load $N_A8 8 0 0
        load $N_A7 7 0 0
        load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


   }


# ----------- set up analysis parameters

```

```
# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;     # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;    # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 10;                           # Convergence Test: tolerance
variable maxNumIterStatic 10000;                 # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                 # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;   # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A9   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
```

```
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";   # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
   # if analysis fails, we try some other stuff, performance is slower inside this loop
   set Dstep 0.0;
   set ok 0
   while {$Dstep <= 1.0 && $ok == 0} {
      set controlDisp [nodeDisp $N_A9 1 ]
      set Dstep [expr $controlDisp/$Dmax]
      set ok [analyze 1 ]
      # if analysis fails, we try some other stuff
      # performance is slower inside this loop  global maxNumIterStatic;       # max no. of iterations performed before "failure to converge" is ret'd
      if {$ok != 0} {
         puts "Trying Newton with Initial Tangent .."
         test NormDispIncr   $Tol 3000 0
         algorithm Newton -initial
         set ok [analyze 1]
         test $testTypeStatic $TolStatic      $maxNumIterStatic    0
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying Broyden .."
         algorithm Broyden 8
         set ok [analyze 1 ]
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying NewtonWithLineSearch .."
         algorithm NewtonLineSearch 0.8
         set ok [analyze 1]
         algorithm $algorithmTypeStatic
      }

   }; # end while loop
};      # end if ok !0

# -----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
   puts [format $fmt1 "PROBLEM" $N_A9 1 [nodeDisp $N_A9 1] "mm"]
} else {
   puts [format $fmt1 "DONE"  $N_A9 1 [nodeDisp $N_A9 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 10 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B9S MRFs- Masonry-Concrete Infilled Frames without Ground Infills

Appendix 10: 3B9S masonry infilled Frame without ground infills

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B9S MRFs-Masonry-Concrete Infilled Frame without ground infills:

The code generated is the same as Appendix 9. However, infill walls at ground level shall be removed as the following:

-These elements must be removed:

  1. Diagonal members

element elasticBeamColumn $dia1A $N_A1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2A $N_B1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3A $N_A0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4A $N_B0 $N_W1A_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1B $N_B1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2B $N_C1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3B $N_B0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4B $N_C0 $N_W1B_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1C $N_C1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2C $N_D1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3C $N_C0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4C $N_D0 $N_W1C_R $Aw $Ew $Izw $WtransfTag

  2. Central members

element forceBeamColumn $cen1A    $N_W1A_L $N_W1A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1B    $N_W1B_L $N_W1B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1C    $N_W1C_L $N_W1C_R    $wallTransfTag $integrationw


3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 7

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

```

```
#   Node Recorder "Displacements":  fileName                <nodeTag>             dof   resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A8  $N_A9   -dof  1  disp
recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;          # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time     -ele  5 6    section fiber y z  $R_steel stressStrain
recorder  Element   -file $dataDir/force10B.out    -time     -ele 810  820  830  840  850  860 870 880 890 localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 617 618 619 621 622 623 624 625 626 627 628 629 shearpanel stressStrain
recorder  Element   -file $dataDir/Strut1.out    -time    -ele  112 113 114 115 116 117 118 119  122 123 124 125 126 127 128 129 section 1  fiber y z  stressStrain;
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
  set ViewScale 5;
  DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model


# characteristics of pushover analysis
set Dmax 1000;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.1;    # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 0.0001;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;     # define load pattern -- generalized
    load $N_A9 9 0 0
        load $N_A8 8 0 0
        load $N_A7 7 0 0
        load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


  }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;    # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.

#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;   # try UmfPack for large model
system $systemTypeStatic
```

```
# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 20;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;  # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A9   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";  # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
  # if analysis fails, we try some other stuff, performance is slower inside this loop
  set Dstep 0.0;
  set ok 0
  while {$Dstep <= 1.0 && $ok == 0} {
    set controlDisp [nodeDisp $N_A9 1 ]
    set Dstep [expr $controlDisp/$Dmax]
    set ok [analyze 1 ]
    # if analysis fails, we try some other stuff
    # performance is slower inside this loop  global maxNumIterStatic;      # max no. of iterations performed before "failure to converge" is ret'd
    if {$ok != 0} {
      puts "Trying Newton with Initial Tangent .."
      test NormDispIncr   $Tol 3000 0
      algorithm Newton -initial
      set ok [analyze 1]
      test $testTypeStatic $TolStatic      $maxNumIterStatic    0
      algorithm $algorithmTypeStatic
```

```
    }
    if {$ok != 0} {
      puts "Trying Broyden .."
      algorithm Broyden 8
      set ok [analyze 1 ]
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying NewtonWithLineSearch .."
      algorithm NewtonLineSearch 0.8
      set ok [analyze 1]
      algorithm $algorithmTypeStatic
    }

  };  # end while loop
};      # end if ok !0

# -----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
  puts [format $fmt1 "PROBLEM" $N_A9 1 [nodeDisp $N_A9 1] "mm"]
} else {
  puts [format $fmt1 "DONE"  $N_A9 1 [nodeDisp $N_A9 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 11 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B6S MRFs- Stone-Concrete Infilled Frames

Appendix 11: 3B6S stone infilled Frame

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B6S MRFs-Stone-Concrete Infilled Frame:

The code generated is the same as Appendix 5. However, constitutive model of stone-concrete infilled frame and dimensions of infill elements shall be used as the following

########################### material for infill walls ##############################################################

## Positive/Negative envelope Stress

set A 1;
set p1 [expr 4.36*$A]; set p2 [expr 5.16*$A]; set p3 [expr 5.16*$A]; set p4 [expr 1.03*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1] [expr $p2] [expr $p3] [expr $p4]]
set nEnvStrsp [list [expr -$p1] [expr -$p2] [expr -$p3] [expr -$p4]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.000258 0.0021 0.004158   0.013]
set nEnvStnsp [list -0.000258 -0.0021 -0.004158  -0.013]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
```

```
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $wall_mat [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy

and the dimensions of each element should be replaced as the following:

set width_wall 615;      #using equation 16 in the report, the width of the strut based on H=3m,L=4m, and t=0.3m
set t_wall 300;

3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 3

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>            dof   resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A5  $N_A6   -dof  1  disp
recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;               # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
   set ViewScale 5;
   DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1800;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.01;      # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 1;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;         # define load pattern -- generalized
      load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


```

```
   }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;     # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.

#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;    # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 0.01;                         # Convergence Test: tolerance
variable maxNumIterStatic 10000;                 # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                 # Convergence Test: flag used to print information on convergence (optional)    # 1: print information on each step;
variable testTypeStatic EnergyIncr ;    # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
```

```
integrator DisplacementControl  $N_A6   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";   # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
   # if analysis fails, we try some other stuff, performance is slower inside this loop
   set Dstep 0.0;
   set ok 0
   while {$Dstep <= 1.0 && $ok == 0} {
      set controlDisp [nodeDisp $N_A6 1 ]
      set Dstep [expr $controlDisp/$Dmax]
      set ok [analyze 1 ]
      # if analysis fails, we try some other stuff
      # performance is slower inside this loop  global maxNumIterStatic;       # max no. of iterations performed before "failure to converge" is ret'd
      if {$ok != 0} {
         puts "Trying Newton with Initial Tangent .."
         test NormDispIncr   $Tol 3000 0
         algorithm Newton -initial
         set ok [analyze 1]
         test $testTypeStatic $TolStatic      $maxNumIterStatic    0
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying Broyden .."
         algorithm Broyden 8
         set ok [analyze 1 ]
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying NewtonWithLineSearch .."
         algorithm NewtonLineSearch 0.8
         set ok [analyze 1]
         algorithm $algorithmTypeStatic
      }

   }; # end while loop
};      # end if ok !0

# -----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
   puts [format $fmt1 "PROBLEM" $N_A6 1 [nodeDisp $N_A6 1] "mm"]
} else {
   puts [format $fmt1 "DONE"  $N_A6 1 [nodeDisp $N_A6 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 12 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B6S MRFs- Stone-Concrete Infilled Frames without Ground Infills

Appendix 12: 3B6S stone infilled Frame without ground infills

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B6S MRFs-Stone-Concrete Infilled Frame without ground infills:

The code generated is the same as Appendix 11. However, infill walls at ground level shall be removed as the following:

-These elements must be removed:

  1. Diagonal members

element elasticBeamColumn $dia1A $N_A1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2A $N_B1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3A $N_A0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4A $N_B0 $N_W1A_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1B $N_B1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2B $N_C1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3B $N_B0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4B $N_C0 $N_W1B_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1C $N_C1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2C $N_D1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3C $N_C0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4C $N_D0 $N_W1C_R $Aw $Ew $Izw $WtransfTag

  2. Central members

element forceBeamColumn $cen1A    $N_W1A_L $N_W1A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1B    $N_W1B_L $N_W1B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1C    $N_W1C_L $N_W1C_R    $wallTransfTag $integrationw


3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 3

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

```

```
#   Node Recorder "Displacements":  fileName                <nodeTag>            dof   resptype
#recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
#recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A5  $N_A6   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;         # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6   section fiber y z  $R_steel stressStrain
#recorder  Element   -file $dataDir/force10.out    -time    -ele  710    section 1  fiber y z $R_steel  stressStrain;
#recorder  Element   -file $dataDir/force60B.out    -time    -ele  860    section 1  fiber y z $R_steel stressStrain;
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860  localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 621 622 623 624 625 626 shearpanel stressStrain
recorder  Element   -file $dataDir/Strut1.out    -time    -ele  112 113 114 115 116  122 123 124 125 126  section 1  fiber y z  stressStrain;
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
  set ViewScale 5;
  DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1800;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.01;   # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 1;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;     # define load pattern -- generalized
    load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


  }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;    # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;   # try UmfPack for large model
system $systemTypeStatic

```

```
# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 1;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                 # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;  # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A6   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";  # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
  # if analysis fails, we try some other stuff, performance is slower inside this loop
  set Dstep 0.0;
  set ok 0
  while {$Dstep <= 1.0 && $ok == 0} {
    set controlDisp [nodeDisp $N_A6 1 ]
    set Dstep [expr $controlDisp/$Dmax]
    set ok [analyze 1 ]
    # if analysis fails, we try some other stuff
    # performance is slower inside this loop  global maxNumIterStatic;      # max no. of iterations performed before "failure to converge" is ret'd
    if {$ok != 0} {
      puts "Trying Newton with Initial Tangent .."
      test NormDispIncr   $Tol 3000 0
      algorithm Newton -initial
      set ok [analyze 1]
      test $testTypeStatic $TolStatic      $maxNumIterStatic    0
      algorithm $algorithmTypeStatic
    }
```

```
    if {$ok != 0} {
      puts "Trying Broyden .."
      algorithm Broyden 8
      set ok [analyze 1 ]
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying NewtonWithLineSearch .."
      algorithm NewtonLineSearch 0.8
      set ok [analyze 1]
      algorithm $algorithmTypeStatic
    }

  };  # end while loop
};      # end if ok !0

# -----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
  puts [format $fmt1 "PROBLEM" $N_A6 1 [nodeDisp $N_A6 1] "mm"]
} else {
  puts [format $fmt1 "DONE"  $N_A6 1 [nodeDisp $N_A6 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

Appendix 13 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B9S MRFs- Stone-Concrete Infilled Frames

Appendix 13: 3B9S Stone infilled Frame

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B9S MRFs-Stone-Concrete Infilled Frame:

The code generated is the same as Appendix 9. However, constitutive model of stone-concrete infilled frame and dimensions of infill elements shall be used as the following

########################### material for infill walls #############################################################

## Positive/Negative envelope Stress

set A 1;
set p1 [expr 4.36*$A]; set p2 [expr 5.16*$A]; set p3 [expr 5.16*$A]; set p4 [expr 1.03*$A];

## stress1 stress2 stress3 stress4
set pEnvStrsp [list [expr $p1] [expr $p2] [expr $p3] [expr $p4]]
set nEnvStrsp [list [expr -$p1] [expr -$p2] [expr -$p3] [expr -$p4]]

## Positive/Negative envelope Strain
## strain1 strain2 strain3 strain4

set pEnvStnsp [list 0.000258 0.0021 0.004158   0.013]
set nEnvStnsp [list -0.000258 -0.0021 -0.004158  -0.013]

## Ratio of maximum deformation at which reloading begins
## Pos_env. Neg_env.
set rDispsp [list 0.2 0.2]

## Ratio of envelope force (corresponding to maximum deformation) at which reloading begins

### Pos_env. Neg_env.
set rForcesp [list 0.2 0.2]


## Ratio of monotonic strength developed upon unloading
### Pos_env. Neg_env.

set uForcesp [list 0.0 0.0]


## Coefficients for Unloading Stiffness degradation

## gammaK1 gammaK2 gammaK3 gammaK4 gammaKLimit

#set gammaKsp [list 1.13364492409642 0.0 0.10111033064469 0.0 0.91652498468618]

set gammaKsp [list 0.0 0.0 0.0 0.0 0.0]

#### Coefficients for Reloading Stiffness degradation
### gammaD1 gammaD2 gammaD3 gammaD4 gammaDLimit

#set gammaDsp [list 0.12 0.0 0.23 0.0 0.95]

set gammaDsp [list 0.0 0.0 0.0 0.0 0.0]
#### Coefficients for Strength degradation
```

```
### gammaF1 gammaF2 gammaF3 gammaF4 gammaFLimit

#set gammaFsp [list 1.11 0.0 0.319 0.0 0.125]
set gammaFsp [list 0.0 0.0 0.0 0.0 0.0]

set gammaEsp 10.0

uniaxialMaterial Pinching4 $wall_mat [lindex $pEnvStrsp 0] [lindex $pEnvStnsp 0] \
[lindex $pEnvStrsp 1] [lindex $pEnvStnsp 1] [lindex $pEnvStrsp 2] \
[lindex $pEnvStnsp 2] [lindex $pEnvStrsp 3] [lindex $pEnvStnsp 3] \
[lindex $nEnvStrsp 0] [lindex $nEnvStnsp 0] \
[lindex $nEnvStrsp 1] [lindex $nEnvStnsp 1] [lindex $nEnvStrsp 2] \
[lindex $nEnvStnsp 2] [lindex $nEnvStrsp 3] [lindex $nEnvStnsp 3] \
[lindex $rDispsp 0] [lindex $rForcesp 0] [lindex $uForcesp 0] \
[lindex $rDispsp 1] [lindex $rForcesp 1] [lindex $uForcesp 1] \
[lindex $gammaKsp 0] [lindex $gammaKsp 1] [lindex $gammaKsp 2] [lindex $gammaKsp 3] [lindex $gammaKsp 4] \
[lindex $gammaDsp 0] [lindex $gammaDsp 1] [lindex $gammaDsp 2] [lindex $gammaDsp 3] [lindex $gammaDsp 4] \
[lindex $gammaFsp 0] [lindex $gammaFsp 1] [lindex $gammaFsp 2] [lindex $gammaFsp 3] [lindex $gammaFsp 4] \
$gammaEsp energy
```

and the dimensions of each element should be replaced as the following:

```
set width_wall 615;      #using equation 16 in the report, the width of the strut based on H=3m,L=4m, and t=0.3m
set t_wall 300;
```

3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 7

5) Pushover Analysis Procedure:

```
#################################################################################

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

#   Node Recorder "Displacements":  fileName                <nodeTag>             dof   resptype
#recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
#recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A8  $N_A9   -dof  1  disp
#recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;               # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860 870 880 890 localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 617 618 619 621 622 623 624 625 626 627 628 629 shearpanel stressStrain
recorder  Element   -file $dataDir/Strut1.out    -time    -ele 111 112 113 114 115 116 117 118 119 121 122 123 124 125 126 127 128 129 section 1  fiber y z  stressStrain;
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
   set ViewScale 5;
   DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1000;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.1;    # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 0.01;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;         # define load pattern -- generalized
      load $N_A9 9 0 0
```

```
        load $N_A8 8 0 0
        load $N_A7 7 0 0
        load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


   }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;     # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.
#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;    # try UmfPack for large model
system $systemTypeStatic

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 10;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;   # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis
```

```
#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A9   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";   # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
   # if analysis fails, we try some other stuff, performance is slower inside this loop
   set Dstep 0.0;
   set ok 0
   while {$Dstep <= 1.0 && $ok == 0} {
      set controlDisp [nodeDisp $N_A9 1 ]
      set Dstep [expr $controlDisp/$Dmax]
      set ok [analyze 1 ]
      # if analysis fails, we try some other stuff
      # performance is slower inside this loop  global maxNumIterStatic;       # max no. of iterations performed before "failure to converge" is ret'd
      if {$ok != 0} {
         puts "Trying Newton with Initial Tangent .."
         test NormDispIncr   $Tol 3000 0
         algorithm Newton -initial
         set ok [analyze 1]
         test $testTypeStatic $TolStatic      $maxNumIterStatic    0
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying Broyden .."
         algorithm Broyden 8
         set ok [analyze 1 ]
         algorithm $algorithmTypeStatic
      }
      if {$ok != 0} {
         puts "Trying NewtonWithLineSearch .."
         algorithm NewtonLineSearch 0.8
         set ok [analyze 1]
         algorithm $algorithmTypeStatic
      }

   }; # end while loop
};      # end if ok !0

# ---------------------------------------------------------------------------------------------------------
if {$ok != 0 } {
   puts [format $fmt1 "PROBLEM" $N_A9 1 [nodeDisp $N_A9 1] "mm"]
} else {
   puts [format $fmt1 "DONE"  $N_A9 1 [nodeDisp $N_A9 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
```

```
wipe all;
```

Appendix 14 – The Code Generated Using the OpenSees Program to Perform Pushover Analysis for 3B9S MRFs- Stone-Concrete Infilled Frames without Ground Infills

Appendix 14: 3B9S stone infilled Frame without ground infills

```

1) Complementry files were defined to organize and make the procedure easier:

   1. Library for Units

The code generated is the same as Appendix 3

   2. Building RC Cross-Section (Fiber Appraoch)

The code generated is the same as Appendix 3

   3. Display The Model in 2D

The code generated is the same as Appendix 3

   4. Display Plane Deformed Shape for 2D Model

The code generated is the same as Appendix 3


   5. Procedure for Defining Uniaxial Pinching Material

The code generated is the same as Appendix 3.


2) 2D Model Definition for 3B9S MRFs-Stone-Concrete Infilled Frame without ground infills:

The code generated is the same as Appendix 13. However, infill walls at ground level shall be removed as the following:

-These elements must be removed:

  1. Diagonal members

element elasticBeamColumn $dia1A $N_A1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2A $N_B1 $N_W1A_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3A $N_A0 $N_W1A_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4A $N_B0 $N_W1A_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1B $N_B1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2B $N_C1 $N_W1B_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3B $N_B0 $N_W1B_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4B $N_C0 $N_W1B_R $Aw $Ew $Izw $WtransfTag

element elasticBeamColumn $dia1C $N_C1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia2C $N_D1 $N_W1C_L $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia3C $N_C0 $N_W1C_R $Aw $Ew $Izw $WtransfTag
element elasticBeamColumn $dia4C $N_D0 $N_W1C_R $Aw $Ew $Izw $WtransfTag

  2. Central members

element forceBeamColumn $cen1A    $N_W1A_L $N_W1A_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1B    $N_W1B_L $N_W1B_R    $wallTransfTag $integrationw
element forceBeamColumn $cen1C    $N_W1C_L $N_W1C_R    $wallTransfTag $integrationw


3) Gravity Analysis Procedure:

The code generated is the same as Appendix 3

4) Modal Analysis Procedure:

The code generated is the same as Appendix 7

5) Pushover Analysis Procedure:

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

# start analysis


puts "ooo Analysis: Pushover ooo"

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

# set recorders

#   Global behaviour

```

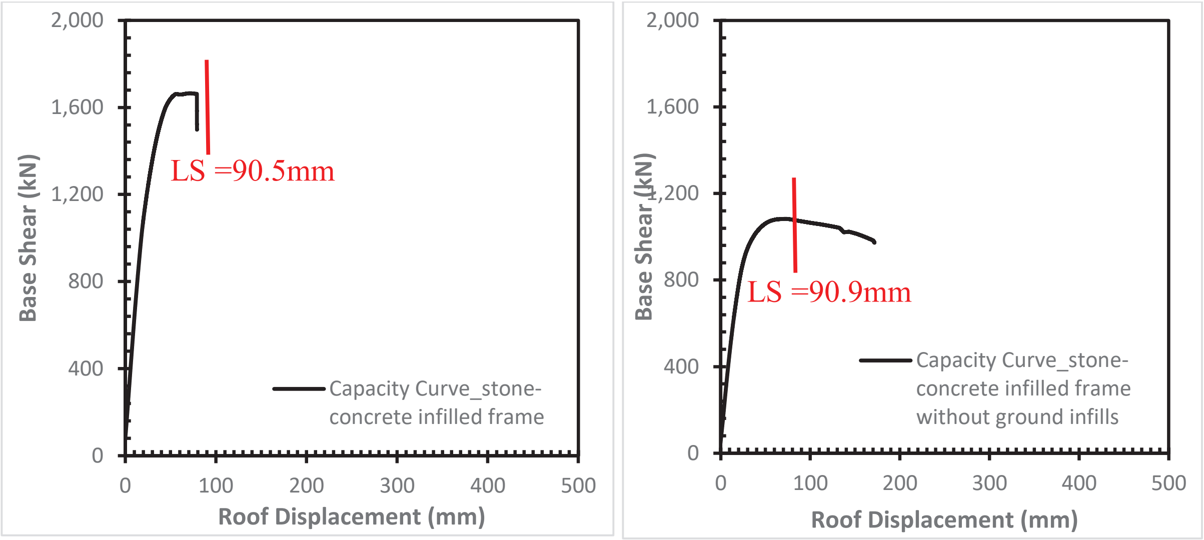
```
#   Node Recorder "Displacements":  fileName                <nodeTag>             dof    resptype
recorder  Node   -file $dataDir/Pushover_Horizontal_Reactions.out   -time    -node  $N_A0  $N_B0 $N_C0  $N_D0   -dof  1  reaction
recorder  Node   -file $dataDir/Pushover_Storey_Displacement.out    -time    -node  $N_A8  $N_A9   -dof  1  disp
recorder Node -file $dataDir/DFree.out -time -node $N_A1  $N_A2 -dof 1 2 disp;          # displacements of free nodes
#recorder  Element   -file $dataDir/stressStrain.out    -time    -ele  5 6    section fiber y z  $R_steel stressStrain
recorder  Element   -file $dataDir/force10B.out    -time    -ele 810  820  830  840  850  860 870 880 890 localForce;
recorder Element  -file $dataDir/SP1.out -time -ele 611 612 613 614 615 616 617 618 619 621 622 623 624 625 626 627 628 629 shearpanel stressStrain
recorder  Element   -file $dataDir/Strut1.out    -time    -ele  112 113 114 115 116 117 118 119  122 123 124 125 126 127 128 129 section 1  fiber y z  stressStrain;
recorder  Element   -file $dataDir/force10.out    -time    -ele  710 720 730 740 750 760 711 721 731 741 751 761  localForce;
# analysis options


set tStart [clock clicks  -milliseconds]


# display deformed shape:
  set ViewScale 5;
  DisplayModel2D DeformedShape $ViewScale ; # display deformed shape, the scaling factor needs to be adjusted for each model

# characteristics of pushover analysis
set Dmax 1000;    # maximum displacement of pushover. push to 10% drift.
set Dincr 0.1;    # displacement increment for pushover. you want this to be very small, but not too small to slow down the analysis
set Tol 1;
# create load pattern for lateral pushover load
pattern Plain 200 Linear {;     # define load pattern -- generalized
    load $N_A9 9 0 0
        load $N_A8 8 0 0
        load $N_A7 7 0 0
        load $N_A6 6 0 0
        load $N_A5 5 0 0
        load $N_A4 4 0 0
        load $N_A3 3 0 0
        load $N_A2 2 0 0
        load $N_A1 1 0 0


  }


# ----------- set up analysis parameters

# ## CONSTRAINTS handler >> Determines how the constraint equations are enforced in the analysis

#          >> Plain Constraints -- Removes constrained degrees of freedom from the system of equations (only for homogeneous equations)
#          >> Lagrange Multipliers -- Uses the method of Lagrange multipliers to enforce constraints
#          >> Penalty Method -- Uses penalty numbers to enforce constraints --good for static analysis with non-homogeneous eqns (rigidDiaphragm)
#          >> Transformation Method -- Performs a condensation of constrained degrees of freedom
variable constraintsTypeStatic Transformation;    # default;
constraints $constraintsTypeStatic

# DOF NUMBERER (number the degrees of freedom in the domain):

#        Determines the mapping between equation numbers and degrees-of-freedom
#          Plain -- Uses the numbering provided by the user
#          RCM -- Renumbers the DOF to minimize the matrix band-width using the Reverse Cuthill-McKee algorithm
set numbererTypeStatic RCM
numberer $numbererTypeStatic


# SYSTEM:

#   Linear Equation Solvers (how to store and solve the system of equations in the analysis)
#   -- provide the solution of the linear system of equations Ku = P. Each solver is tailored to a specific matrix topology.

#          ProfileSPD -- Direct profile solver for symmetric positive definite matrices
#          BandGeneral -- Direct solver for banded unsymmetric matrices
#          BandSPD -- Direct solver for banded symmetric positive definite matrices
#          SparseGeneral -- Direct solver for unsymmetric sparse matrices
#          SparseSPD -- Direct solver for symmetric sparse matrices
#          UmfPack -- Direct UmfPack solver for unsymmetric matrices
set systemTypeStatic UmfPack;   # try UmfPack for large model
system $systemTypeStatic
```

```

# TEST: # convergence test to

#   -- Accept the current state of the domain as being on the converged solution path
#   -- determine if convergence has been achieved at the end of an iteration step
#          NormUnbalance -- Specifies a tolerance on the norm of the unbalanced load at the current iteration
#          NormDispIncr -- Specifies a tolerance on the norm of the displacement increments at the current iteration
#          EnergyIncr-- Specifies a tolerance on the inner product of the unbalanced load and displacement increments at the current iteration
#          RelativeNormUnbalance --
#          RelativeNormDispIncr --
#          RelativeEnergyIncr --
variable TolStatic 10;                        # Convergence Test: tolerance
variable maxNumIterStatic 10000;                # Convergence Test: maximum number of iterations that will be performed before "failure to converge" is returned
variable printFlagStatic 0;                # Convergence Test: flag used to print information on convergence (optional)   # 1: print information on each step;
variable testTypeStatic EnergyIncr ;  # Convergence-test type
test $testTypeStatic $TolStatic $maxNumIterStatic $printFlagStatic;

# Solution ALGORITHM: -- Iterate from the last time step to the current
#          Linear -- Uses the solution at the first iteration and continues
#          Newton -- Uses the tangent at the current iteration to iterate to convergence
#          ModifiedNewton -- Uses the tangent at the first iteration to iterate to convergence
#          NewtonLineSearch --
#          KrylovNewton --
#          BFGS --
#          Broyden --
variable algorithmTypeStatic Newton
algorithm $algorithmTypeStatic;

# Static INTEGRATOR: -- determine the next time step for an analysis

#          LoadControl -- Specifies the incremental load factor to be applied to the loads in the domain
#          DisplacementControl -- Specifies the incremental displacement at a specified DOF in the domain
#          Minimum Unbalanced Displacement Norm -- Specifies the incremental load factor such that the residual displacement norm in minimized
#          Arc Length -- Specifies the incremental arc-length of the load-displacement path
# Transient INTEGRATOR: -- determine the next time step for an analysis including inertial effects
#          Newmark -- The two parameter time-stepping method developed by Newmark
#          HHT -- The three parameter Hilbert-Hughes-Taylor time-stepping method
#          Central Difference -- Approximates velocity and acceleration by centered finite differences of displacement
integrator DisplacementControl  $N_A9   1 $Dincr

# ANALYSIS  -- defines what type of analysis is to be performed

#          Static Analysis -- solves the KU=R problem, without the mass or damping matrices.
#          Transient Analysis -- solves the time-dependent analysis. The time step in this type of analysis is constant. The time step in the output is also constant.
#          variableTransient Analysis -- performs the same analysis type as the Transient Analysis object. The time step, however, is variable. This method is used when
#                 there are convergence problems with the Transient Analysis object at a peak or when the time step is too small. The time step in the output is also variable.
set analysisTypeStatic Static
analysis $analysisTypeStatic


#  ---------------------------------    perform Static Pushover Analysis
set Nsteps [expr int($Dmax/$Dincr)];        # number of pushover analysis steps
set ok [analyze $Nsteps];                # this will return zero if no convergence problems were encountered
set fmt1 "%s Pushover analysis: CtrlNode %.24i, dof %.1i, Disp=%.4f %s";  # format for screen/file output of DONE/PROBLEM analysis
if {$ok != 0} {
  # if analysis fails, we try some other stuff, performance is slower inside this loop
  set Dstep 0.0;
  set ok 0
  while {$Dstep <= 1.0 && $ok == 0} {
    set controlDisp [nodeDisp $N_A9 1 ]
    set Dstep [expr $controlDisp/$Dmax]
    set ok [analyze 1 ]
    # if analysis fails, we try some other stuff
    # performance is slower inside this loop  global maxNumIterStatic;      # max no. of iterations performed before "failure to converge" is ret'd
    if {$ok != 0} {
      puts "Trying Newton with Initial Tangent .."
      test NormDispIncr   $Tol 3000 0
      algorithm Newton -initial
      set ok [analyze 1]
      test $testTypeStatic $TolStatic      $maxNumIterStatic    0
      algorithm $algorithmTypeStatic
```

```
    }
    if {$ok != 0} {
      puts "Trying Broyden .."
      algorithm Broyden 8
      set ok [analyze 1 ]
      algorithm $algorithmTypeStatic
    }
    if {$ok != 0} {
      puts "Trying NewtonWithLineSearch .."
      algorithm NewtonLineSearch 0.8
      set ok [analyze 1]
      algorithm $algorithmTypeStatic
    }

  };  # end while loop
};      # end if ok !0

# -----------------------------------------------------------------------------------------------------
if {$ok != 0 } {
  puts [format $fmt1 "PROBLEM" $N_A9 1 [nodeDisp $N_A9 1] "mm"]
} else {
  puts [format $fmt1 "DONE"  $N_A9 1 [nodeDisp $N_A9 1] "mm"]
}


# Stop timing of this analysis sequence
set tStop [clock clicks  -milliseconds]
puts "o Time taken: [expr ($tStop-$tStart)/1000.0] sec"

puts "pushover analysis completed"

# Reset for next analysis sequence
wipe all;
```

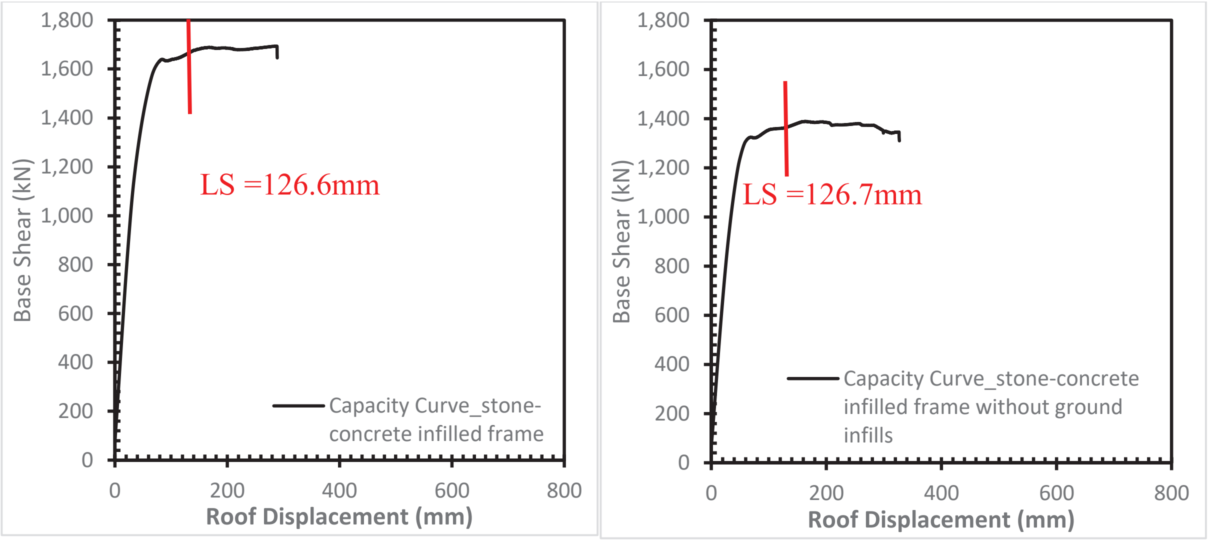
Appendix 15 – The Data Analysis of 3B6S and 3B9S MRFs-Stone-Concrete Infilled Frames with/without Ground Infills

3B6S and 3B9S MRFs-Stone-Concrete Infilled Frame with/without ground infills



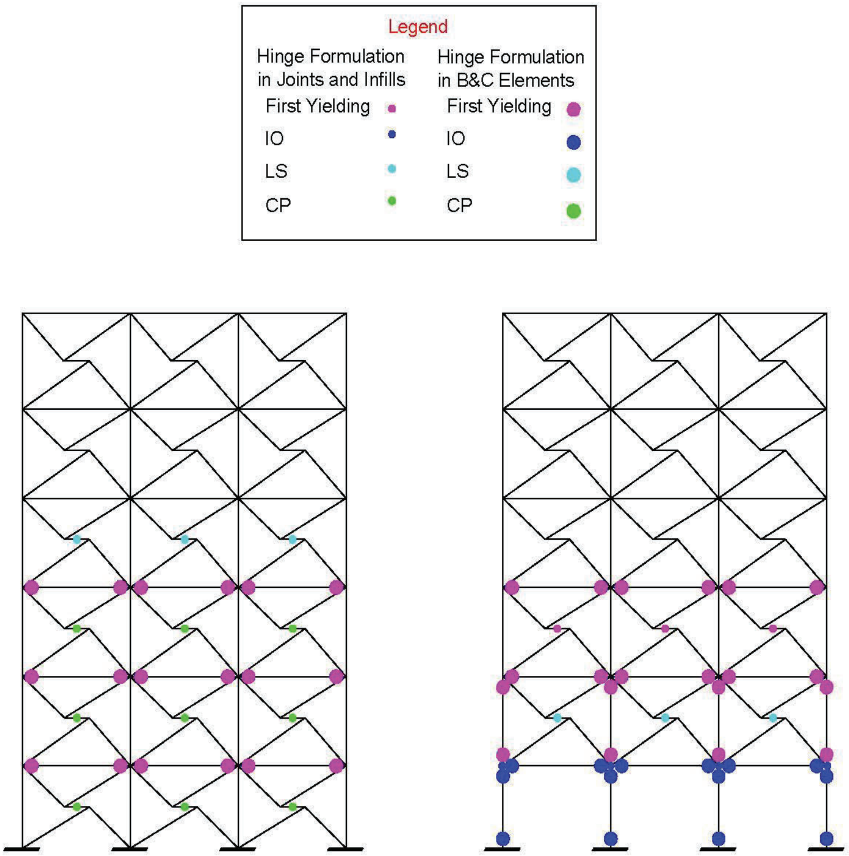
a) b)

Fig 1: a) Performance point for 3B6S MRFs- stone-concrete infilled frame. b) Performance point for 3B6S MRFs- stone-concrete infilled frame without ground infills.



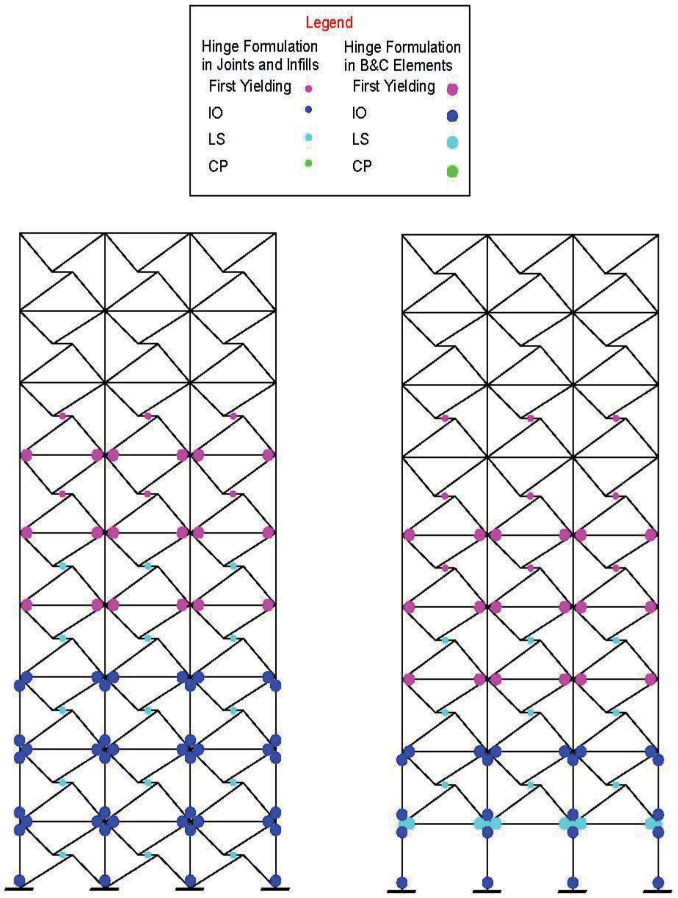
a) b)

Fig 2: a) Performance point for 3B9S MRFs- stone-concrete infilled frame. b) Performance point for 3B9S MRFs- stone-concrete infilled frame without ground infills.



a) b)

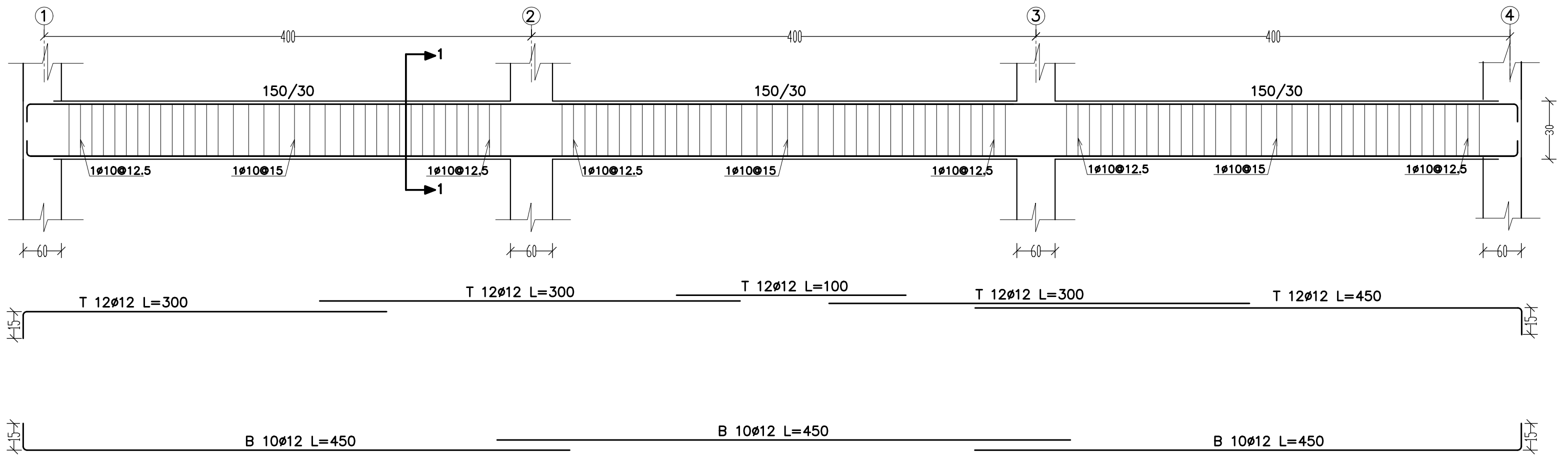
Fig 3: a) Hinge formation at the assigned elements in 3B6S MRFs- stone-concrete infilled frame. b) Hinge formation at the assigned elements in 3B6S MRFs- stone-concrete infilled frame without ground infills.



a) b)

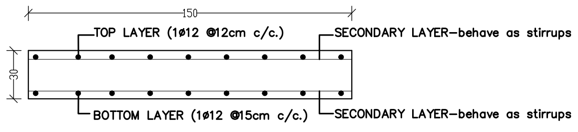
Fig 4: a) Hinge formation at the assigned elements in 3B9S MRFs- stone-concrete infilled frame. b) Hinge formation at the assigned elements in 3B9S MRFs- stone-concrete infilled frame without ground infills.

Appendix 16 – Full Structural Detailing for 3B6S and 3B9S Ductile and Non-Ductile Frame Systems



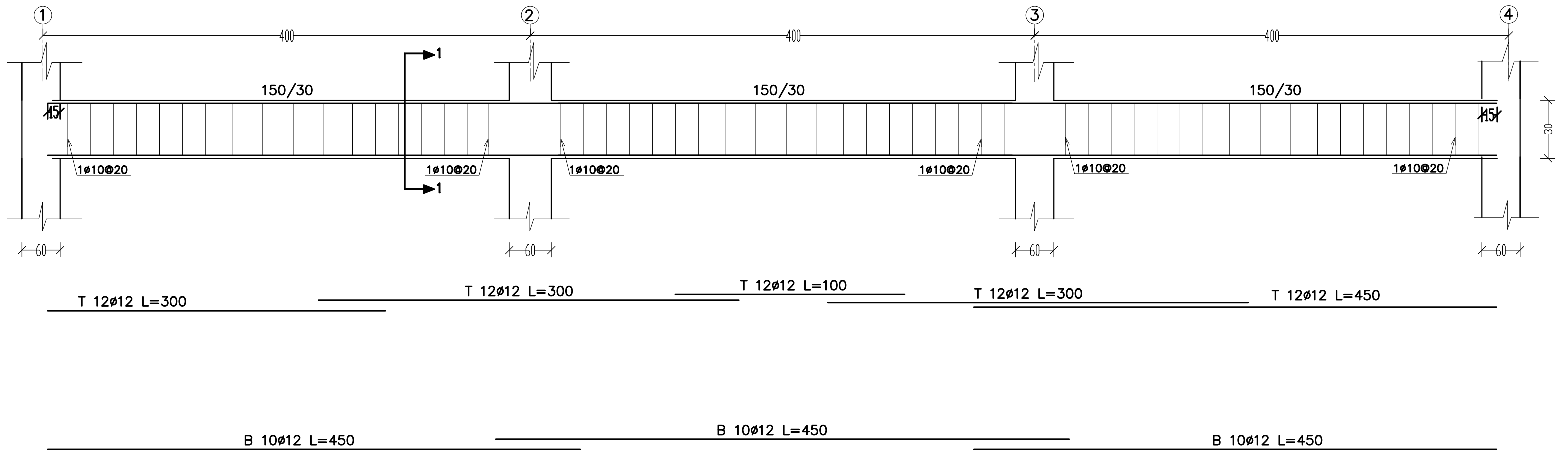
Typical Beam Detailing of Ductile 3B6S and 3B9S Frame System

1:50–H
1:20–V



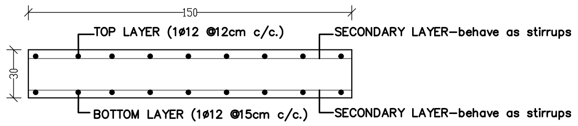
Section (1–1) in Beam B 150/30

1:25



Typical Beam Detailing of Non–Ductile 3B6S and 3B9S Frame System

1:50–H
1:20–V



Section (1–1) in Beam B 150/30

1:25

Faculty of Engineering
Civil Engineering Department
Master's Thesis
(ENCE 860)

Project Name :
3B6S and 3B9S Typical
Prototype Model

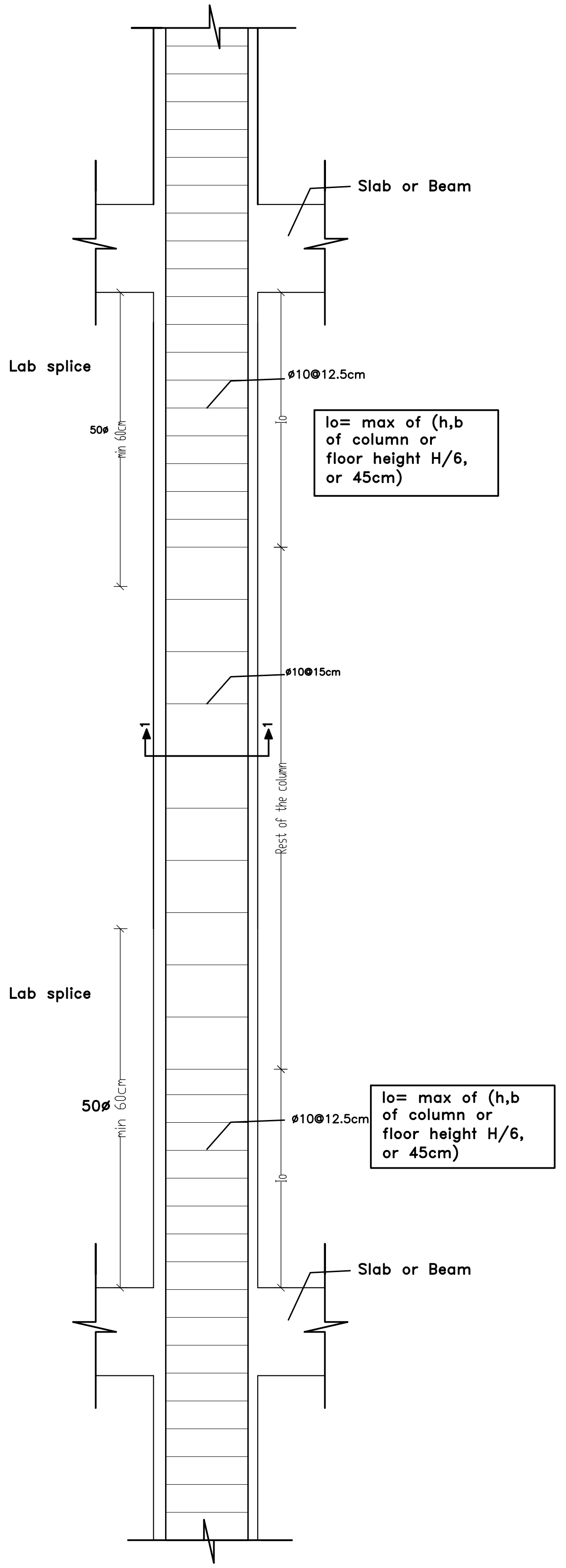
Supervised by :
Dr.Ghada Karaki

Drawing Title :
Appendix 16 :
Beam Detailing for 3B6S and
3B9S Ductile and Non-Ductile
Frame System

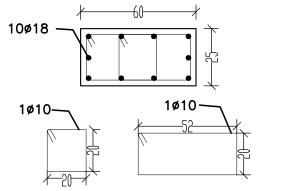
Done by:
Mohammad Safi 1195113

Notes :
1) All Dimensions Are In Cm.
2) All Levels Are In meter

Scale:	Date:	Drawing No:
Variable	16/07/2022	1

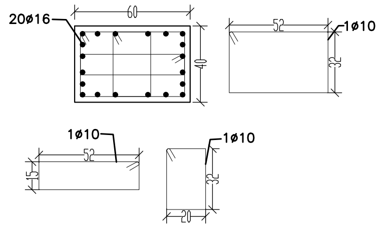


General Columns Detailing for 3B6S and 3B9S Ductile Frame System



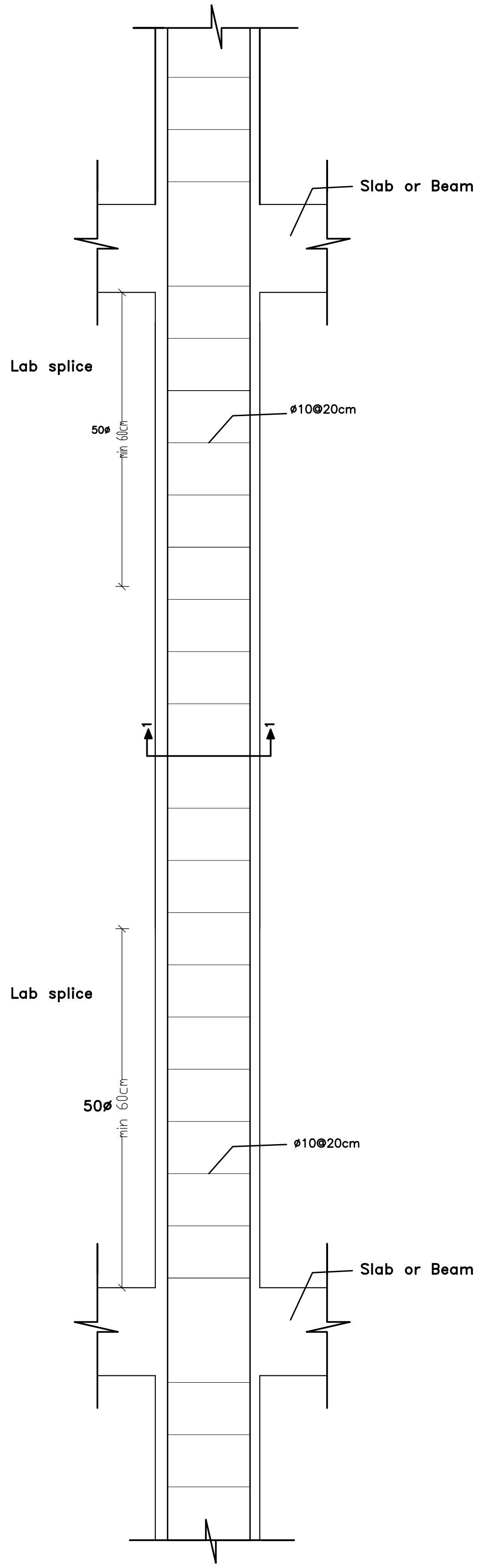
Stirrups spacing as per general column elevation

Section (1-1) in Typical 3B6S Columns

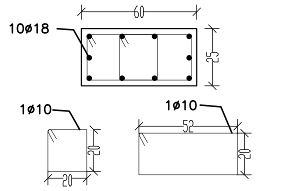


Stirrups spacing as per general column elevation

Section (1-1) in Typical 3B9S Columns

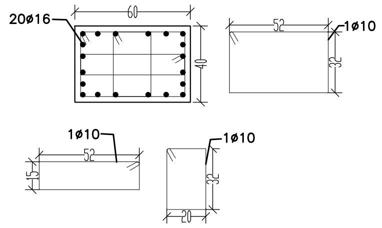


General Columns Detailing for 3B6S and 3B9S Non-Ductile Frame System



Stirrups spacing as per general column elevation

Section (1-1) in Typical 3B6S Columns



Stirrups spacing as per general column elevation

Section (1-1) in Typical 3B9S Columns

BIRZEIT UNIVERSITY

Faculty of Engineering
Civil Engineering Department
Master's Thesis
(ENCE 860)

Project Name :
3B6S and 3B9S Typical
Prototype Model

Supervised by :
Dr.Ghada Karaki

Drawing Title :
Appendix 16 :
Columns Detailing for 3B6S and 3B9S Ductile and Non-Ductile Frame System

Done by:
Mohammad Safi 1195113

Notes :
1) All Dimensions Are In Cm.
2) All Levels Are In meter

Scale:	Date:	Drawing No:
Variable	16/07/2022	2