T2R2 東京科学大学 リサーチリポジトリ
Science Tokyo Research Repository

論文 / 著書情報
Article / Book Information

題目(和文)	
Title(English)	Seismic Performance Evaluation and Design of Damped-outrigger System Incorporating Buckling-restrained Braces
著者(和文)	LINPao-Chun
Author(English)	Pao-Chun Lin
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第11299号, 授与年月日:2019年9月20日, 学位の種別:課程博士, 審査員:竹内 徹,坂田 弘安,五十嵐 規矩夫,田村 修次,佐藤 大樹
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第11299号, Conferred date:2019/9/20, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Powered by T2R2 (Science Tokyo Research Repository)

Seismic Performance Evaluation and Design of Damped-outrigger System Incorporating Buckling-restrained Braces

Pao-Chun Lin

Advisor: Professor Toru Takeuchi

Department of Architecture and Building Engineering
Tokyo Institute of Technology

A thesis submitted for the degree of
Doctor of Engineering
June 2019

Acknowledgement

Foremost, I would like to express my sincere gratitude to my advisor Professor Toru Takeuchi for the continuous support of my doctoral study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my doctoral study.

My sincere thanks also go to Associate Professor Matsui and Dr. Terazawa for who gave me invaluable comments and warm encouragements.

I thank my fellow lab mates in Takeuchi Lab.: Panumas Saingam, Deepshikha Nair, Kurtulus Atasever, Xingchen Chen, Ben Sitler, Xiaotian Yang, Can Liu, Yanpeng Li, Jun Ji, Leming Gu, and Kang Liu for the stimulating discussions, for the life we were working together, and for all the fun we have had in the last three years. Also, I thank my friends in National Taiwan University and National Center for Research on Earthquake Engineering: Ming-Chieh Chuang, An-Chien Wu, and Ching-Yi Tsai. In particular, I thank Professor Keh-Chyuan Tsai for enlightening me with the attitude as being a successful researcher.

I would also like to express my gratitude to my family for their moral support and warm encouragements. Finally, I gratefully appreciate the financial support of Japan-Taiwan Exchange Association (公益財団法人日本台湾交流協会) and the Research Fellowship of the Japan Society for the Promotion of Science (日本学術振興会) that made it possible to complete my thesis.

Abstract

The outrigger system has been an effective solution in mitigating seismic response in the core-tube type tall buildings. It increases the overall building lateral stiffness by mobilizing the perimeter columns' axial stiffness. However, the elastic design concept of the conventional outrigger usually results in large force demands on the outrigger members, increasing both complexity and costs in engineering practices. The concept of damped-outrigger was proposed to increase the damping, instead of increasing the stiffness, by incorporating energy dissipation devices such as viscous dampers. This research studies the seismic behavior of structures with damped-outrigger incorporating buckling-restrained brace (BRB) as an energy dissipation device (BRB-outrigger). When viscous dampers are adopted in a damped-outrigger system in order to control responses induced from wind and seismic loads, the design requirements and velocity ranges corresponding to these two demands are usually different. The wide axial force capacity range and feasible stiffness of BRB allow the BRB-outrigger system to be an alternative in resisting seismic loads. A properly designed BRB-outrigger can function as a conventional outrigger through the BRB's high elastic stiffness during frequent small earthquakes, and it can dissipate energy during moderate to maximum considered earthquake through the BRB's stable hysteretic behavior. The outrigger elevations and the relationships between the axial stiffness of the BRB, the perimeter column, and the flexural stiffness of the outrigger truss are in close relation with the seismic performance. The main objectives of this research are to investigate the seismic performance of structure incorporating single or double layers of BRB-outrigger and to propose optimal design method in order to minimize seismic response by performing a series of parametric study.

The analytical model used in this research is simplified from a real core-tube type building and contains only the core structure, the BRB, the perimeter columns, and the outrigger truss. The analytical models have heights of 64 m, 128 m, 256 m, and 384 m. Two types of dimensionless parameters are developed. The first type of parameter is the outrigger effect factor, which indicates the magnitude of the outrigger effect. The shorter outrigger truss span and the stiffer perimeter column axial stiffness enhance the outrigger effect. The second type is the BRB stiffness parameter, which indicates the relationships between the axial stiffness of the BRB, perimeter column, and the flexural stiffness of the outrigger truss. By using the proposed

dimensionless parameters, the analytical model can be constructed. In addition, the BRB yield deformation is crucial as it determines when the BRB starts dissipating energy. Too large or too small BRB yield deformation could cause low energy dissipation efficiency or early fracture of the BRB during the earthquakes. A method to determine a proper BRB yield deformation is proposed.

The spectral analysis (SA) is used to evaluate seismic response in the parametric study. The equivalent damping ratio to incorporate the effect of BRB's yielding and energy dissipation mechanism is included. The base shear and roof displacement relationship of each mode are obtained by applying modal pushover analysis, as the BRBs in different outrigger levels may not yield simultaneously. The responses of the first four modes are calculated separately and then combined together by using the square root of the sum of the squares method. The smoothened design spectrum is used as seismic input in SA. The SA results are validated by performing nonlinear response history analysis (NLRHA). The NLRHA is performed by using eight ground motions, which are scaled to fit the design spectrum. Both the SA and NLRHA show similar responses and trends.

The maximum roof drift, the maximum inter-story drift, the maximum overturning moment at core structure base, and the maximum perimeter column axial force are utilized as indicators to indicate the seismic performance. Based on the analysis results, the optimal outrigger elevation in order to minimize seismic response in a single BRB-outrigger system is 70% to 80% of the building height. In a double BRB-outrigger system, the optimal outrigger elevations are 70% to 80% and 30% to 60% of the building height for the upper and lower BRB-outriggers, respectively. The BRB-outrigger in the structure with greater outrigger effect factor is more efficient to reduce seismic response. In addition, the larger axial stiffness ratio of the BRB to the perimeter column improves the seismic performance in reducing roof and inter-story drift responses. However, the efficiency in seismic reduction becomes lower with the increasing BRB stiffness, and the acceleration responses can be increased. According to the analysis results, a step-by-step design recommendation and design charts are proposed for engineers to design the structure with either single or double layers of BRB-outrigger in the preliminary design stage without the needs of time-consuming iteration tasks. Furthermore, this research introduced three different types of BRB-outrigger configurations in order to meet different architecture requirements and economical solutions.

Table of Contents

Chapter 1 Introduction

1.1 Preface 1-3
1.2 Background 1-5
  1.2.1 Introduction of outrigger system 1-5
  1.2.2 Outrigger system examples 1-7
1.3 Objectives 1-12
1.4 Thesis outline 1-14
1.5 References 1-15

Chapter 2 Literature review

2.1 Introduction 2-3
2.2 Research on outrigger system 2-3
2.3 Mechanical behavior of BRB 2-8
2.4 Summary 2-11
2.5 References 2-12

Chapter 3 Analytical models

3.1 Introduction 3-3
3.2 Simplified models 3-3
  3.2.1 Uniform mass model 3-7
  3.2.2 Discrete mass model 3-11
3.3 Parameter definitions 3-13
  3.3.1 Parameters for single BRB-outrigger system 3-13
  3.3.2 Parameters for dual BRB-outrigger system 3-14
3.4 Analysis procedure 3-16
3.5 Member-by-member models 3-33
  3.5.1 32-story MBM model 3-34
  3.5.2 96-story MBM model 3-36
3.6 Summary 3-39
3.7 References 3-40

Chapter 4 Analysis methods

4.1 Introduction 4-3
4.2 Modal analysis 4-3
4.3 Spectral analysis 4-5
4.4 Nonlinear response history analysis 4-9
4.5 Analysis examples 4-10
  4.5.1 Analysis results of the 32-story model 4-13
  4.5.2 Analysis results of the 96-story model 4-30
4.6 Summary 4-49
4.7 References 4-51

Chapter 5 Programming for parametric study

5.1 Introduction 5-3
5.2 Computer program procedure 5-3
5.3 Main program 5-5
5.4 Input file 5-6

Chapter 6 Preliminary analysis

6.1 Introduction 6-3
6.2 Modal analysis 6-3
6.3 BRB yield deformation 6-10
6.3.1 Calculation of BRB yield deformation 6-10
6.3.2 Effect of yielding roof drift ratio 6-12
6.3.3 Effect of outrigger truss flexural stiffness 6-16
6.3.4 Effect of BRB yield deformation ratio 6-18
6.4 BRB energy dissipation efficiency based on SA 6-28
6.4.1 Effect of BRB yield deformation ratio and roof yielding drift ratio 6-28
6.4.2 Effect of BRB stiffness parameters 6-35
6.5 Summary 6-40
6.6 References 6-41

Chapter 7 Analysis results for optimal design

7.1 Introduction 7-3
7.2 Single BRB-outrigger system 7-3
7.2.1 Maximum roof drift 7-3
7.2.2 Maximum inter-story drift 7-6
7.2.3 Maximum overturning moment 7-8
7.2.4 Maximum perimeter column axial force 7-11
7.2.5 BRB energy dissipation efficiency 7-13
7.2.6 Summary of optimal design for single BRB-outrigger system 7-15
7.3 Dual BRB-outrigger system 7-18
7.3.1 Maximum roof drift 7-18
7.3.2 Maximum inter-story drift 7-29
7.3.3 Maximum overturning moment 7-39
7.3.4 Maximum perimeter column axial force 7-45
7.3.5 BRB energy dissipation efficiency 7-50
7.3.6 Summary of optimal design for dual BRB-outrigger system 7-60
7.4 Summary 7-60
7.5 References 7-62

Chapter 8 Design recommendation and design examples

8.1 Introduction 8-3
8.2 Optimal design of single BRB-outrigger system 8-3
8.2.1 Introduction of the design example 8-6
8.2.2 Analysis result of the design example 8-9
8.3 Optimal design of dual BRB-outrigger system 8-13
8.3.1 Introduction of the design example 8-16

8.3.2 Design charts 8-16
8.3.3 Analysis result of the design example 8-19
8.4 Summary 8-28

Chapter 9 BRB-outrigger configurations
9.1 Introduction 9-3
9.2 BRB-outrigger configurations 9-3
9.2.1 Ordinary BRB-outrigger system 9-3
9.2.2 BRB-truss outrigger system 9-4
9.2.3 Giant-BRB outrigger system 9-4
9.3 Parameter definitions and analytical models 9-6
9.4 Analysis results 9-8
9.4.1 Optimal outrigger elevations 9-8
9.4.2 Effects of outrigger effect factor and outrigger stiffness ratio 9-14
9.5 Design Examples 9-18
9.5.1 Introduction of the example models 9-18
9.5.2 Seismic response of the example models 9-22
9.5.3 Comparison between OB, BT, and GB outrigger configurations 9-35
9.6 Summary 9-36

Chapter 10 Application of BRB-outrigger
10.1 Introduction 10-3
10.2 Inconsistent perimeter column axial stiffness 10-3
10.2.1 Design examples 10-5
10.3 Linearly changed core rigidity and perimeter column axial stiffness 10-12
10.3.1 Single BRB-outrigger system 10-13
10.3.2 Dual BRB-outrigger system 10-18
10.4 Summary 10-30

Chapter 11 Conclusions 11-3

Appendix A C++ script of computer program for parametric study A-1
Appendix B OpenSees tcl script of 32-story Single DM model B-1
Appendix C OpenSees tcl script of 32-story Single MBM model C-1

Notations

A_c	cross-sectional area of infill concrete of SRC perimeter column
A_{DM}	cross-sectional area of BRB in the DM model
A_e	cross-sectional area of elastic segment of a BRB
A_g	cross-sectional area of steel box perimeter column
$a_{\max}$	maximum roof acceleration
A_p	cross-sectional area of plastic segment of a BRB
A_{pc}	effective cross-sectional area of perimeter column at elevation at building base
A_s	cross-sectional area of steel box of SRC perimeter column
A_t	cross-sectional area of transition segment of a BRB
$\mathbf{B}$	system matrix of structure with BRB-outrigger system
$\mathbf{B}_{dual}$	system matrix of structure with dual BRB-outrigger system
BRB_1	the BRB in lower outrigger of dual BRB-outrigger system
BRB_2	the BRB in upper outrigger of dual BRB-outrigger system
$\mathbf{B}_{single}$	system matrix of structure with single BRB-outrigger system
$C_{1,\max}$	maximum perimeter column axial force in the 1^{st} story
$D_{h,n}$	reduction factor for response spectrum of the n^{th} mode response
D_{Mc}	drop of the $M_{c,\max}$ if compared with the Core model without outrigger
D_{T1}	drop of the first mode vibration period of the elastic system if compared with the Core model without outrigger
D_{θ}	drop of the $\theta_{\max}$ if compared with the Core model without outrigger
E	modulus of elasticity of steel
e	eccentric distance between the BRB and perimeter column central lines
E_{BRB}	ratio of energy dissipated by BRB to the total input energy
E_{BRB1}	ratio of energy dissipated by the BRB in lower outrigger of dual BRB-outrigger system to the total input energy
E_{BRB2}	ratio of energy dissipated by the BRB in upper outrigger of dual BRB-outrigger system to the total input energy
E_c	modulus of elasticity of concrete
E_d	energy dissipated by the BRB-outrigger per loop
E_{d1}	energy dissipated by the BRB_1
E_{d2}	energy dissipated by the BRB_2
EI	core structure flexural rigidity
E_s	strain energy of the system
$E_{s,c}$	modified modulus of elasticity of perimeter column subjected to compression

$E_{s,t}$ modified modulus of elasticity of perimeter column subjected to tension

F force matrix of structure with BRB-outriggers

f'_c compressive strength of concrete

F_{cr} critical buckling stress

F_e elastic buckling stress

$\mathbf{F}_j$ force matrix of j^{th} core structure segment

G_s surface soil layer amplification factor

h building height

h_0 inherent damping ratio

h_d damping ratio

$h_{eq,n}$ equivalent damping ratio of the n^{th} mode

h_j elevation of the j^{th} BRB-outrigger level

h_t vertical span of the BRB in the GB outrigger configuration

${}_jM_j$ bending moment at top of j^{th} core structure segment

${}_jM_{j\text{-}1}$ bending moment at bottom of j^{th} core structure segment

${}_jP_j$ shear at top of j^{th} core structure segment

${}_jP_{j\text{-}1}$ shear at bottom of j^{th} core structure segment

K'_n post-yield modal stiffness of the n^{th} mode

k_b combined effective stiffness of outrigger truss and perimeter column below the outrigger truss for single BRB-outrigger system

k_{bt} flexural stiffness of the BT outrigger

k_c axial stiffness of perimeter column with length of h

$k_{c,c}$ compressive axial stiffness of perimeter column with height of h

$k_{c,cn}$ compressive axial stiffness of perimeter column within the n^{th} segment

$k_{c,t}$ tensile axial stiffness of perimeter column with height of h

$k_{c,tn}$ tensile axial stiffness of perimeter column within the n^{th} segment

k_{c12} effective axial stiffness of perimeter column within the range from elevation x_1 to elevation x_2

k_{cj} axial stiffness of the perimeter column within the j^{th} segment

k_d axial stiffness of the BRB in the single BRB-outrigger

$k_{d,gb}$ axial stiffness of the BRB in the GB outrigger configuration

k_{d1} axial stiffness of the BRB in the lower BRB-outrigger of dual BRB-outrigger

k_{d2} axial stiffness of the BRB in the upper BRB-outrigger of dual BRB-outrigger

k_{dj} axial stiffness of the BRB in the j^{th} level of BRB-outrigger

k_{eff} BRB effective axial stiffness

$K_{eq,i}$ equivalent stiffness when the roof displacement reaches its maximum of

$y_{\max,i}$

$K_{eq,n}$	equivalent stiffness when roof displacement reaches $y_{\max,n}$ in the n^{th} mode shape
$\mathbf{k_g}$	matrix of rotational spring stiffness resulting from BRB-outriggers
k_g	rotational stiffness provided by the single BRB-outrigger
K_i	elastic modal stiffness of the i^{th} mode
K_n	modal stiffness of the n^{th} mode
$k_{og,BT}$	flexural stiffness combined with outrigger truss and BRB in the BT outrigger configuration
$k_{og,GB}$	flexural stiffness provided by the GB outrigger
$k_{og,OB}$	flexural stiffness combined with outrigger truss and BRB in the OB outrigger configuration
$k_{rg,BT}$	rotational stiffness provided by the BT outrigger system
$k_{rg,GB}$	rotational stiffness provided by the GB outrigger system
$k_{rg,OB}$	rotational stiffness provided by the OB outrigger system
k_t	flexural stiffness of outrigger truss in the single BRB-outrigger
k_{t1}	flexural stiffness of lower outrigger truss in the dual BRB-outrigger
k_{t2}	flexural stiffness of upper outrigger truss in the dual BRB-outrigger
k_{tj}	flexural stiffness of outrigger truss in the j^{th} level of BRB-outrigger
L_0	working point-to-working point length of BRB
L_c	effective length of the perimeter column in the 1^{st} story
L_e	length of elastic zone of BRB
L_j	length of the j^{th} core structure segment
L_p	length of plastic zone of BRB
L_t	length of transition zone of BRB
l_t	outrigger span
m	mass per unit height along the core structure of UM model
$M_{c,\max}$	maximum overturning moment at core structure base
M_n	modal mass of the n^{th} mode
$\mathbf{M_o}$	matrix of moments applied by BRB-outriggers on core structure
$M_{o,BT}$	moment applied by the BT outrigger until the core structure rotation at outrigger elevation reaches θ_1
$M_{o,GB}$	moment applied by the GB outrigger until the core structure rotation at outrigger elevation reaches θ_1
$M_{o,OB}$	moment applied by the OB outrigger until the core structure rotation at outrigger elevation reaches θ_1
N	BRB axial force
N_{cu}	maximum axial force capacity of BRB

$N_{cu,1}$	maximum axial force capacity of BRB_1
$N_{cu,2}$	maximum axial force capacity of BRB_2
N_y	BRB axial yield force
p	post-yield stiffness ratio of BRB in analytical model
P_{gb}	axial force of the BRB in the GB outrigger configuration
p_n	post-yield stiffness ratio of the n^{th} mode
P_u	maximum axial force demand in the 1^{st} story perimeter column
r	radius of gyration of perimeter column in the 1^{st} story
R_a	roof acceleration reduction ratio
R_{CPD}	cumulative plastic deformation ratio of BRB
R_{CPD1}	cumulative plastic deformation ratio of the BRB in lower outrigger of dual BRB-outrigger system
R_{CPD2}	cumulative plastic deformation ratio of the BRB in upper outrigger of dual BRB-outrigger system
R_d	roof drift reduction ratio
R_{d2c}	BRB stiffness parameter of dual BRB-outrigger system (ratio of k_{d2} to k_c)
R_{db}	BRB stiffness parameter of single BRB-outrigger system (ratio of k_d to k_b)
R_{dc}	BRB stiffness parameter of single BRB-outrigger system (ratio of k_d to k_c)
R_{dt}	BRB stiffness parameter of single BRB-outrigger system (ratio of k_d to k_t)
R_{dt1}	BRB stiffness parameter of dual BRB-outrigger system (ratio of k_{d1} to k_{t1})
R_{dt2}	BRB stiffness parameter of dual BRB-outrigger system (ratio of k_{d2} to k_{t2})
R_{kd}	BRB stiffness parameter of dual BRB-outrigger system (ratio of k_{d1} to k_{d2})
R_{udy}	BRB yield deformation ratio of dual BRB-outrigger system (ratio of $u_{d,y1}/u_{d,y2}$)
R_y	ratio of the expected yield stress to the specified minimum yield stress
S_{A0}	design acceleration response spectrum at ground surface
S_{bc}	outrigger stiffness parameter of single BRB-outrigger system
S_{bc07}	the S_{bc} value when α=0.7
S_{bc1}	outrigger stiffness parameter of lower BRB-outrigger in dual BRB-outrigger system
S_{bc2}	outrigger stiffness parameter of upper BRB-outrigger in dual BRB-outrigger system
$S_{bc2,07}$	the S_{bc2} value when α_2=0.7
S_{cc}	outrigger effect when α varies from 0 to 1
S_{cc07}	outrigger effect factor (when α=0.7)
S_d	spectral displacement
t	time
T	vibration period

T'_n	the vibration period after the BRBs have yield
T_1	the first mode vibration period
T_2	the second mode vibration period
$T_{eq,i}$	equivalent vibration period of the i^{th} mode
T_i	the i^{th} mode vibration period
T_n	the elastic vibration period of the n^{th} mode
$\mathbf{u}$	displacement matrix of structure with BRB-outriggers
u_{bt}	flexural deformation of the BT outrigger
$u_{bt,y}$	flexural deformation of the BT outrigger when BRB yields
u_c	axial deformation of perimeter column below outrigger elevation in the OB outrigger configuration
$u_{c,BT}$	axial deformation of perimeter column below outrigger elevation in the BT outrigger configuration
$u_{c,GB}$	axial deformation of perimeter column below outrigger elevation in the GB outrigger configuration
u_d	axial deformation of the BRB in the OB outrigger configuration
$u_{d,gb,v}$	vertical component of the axial deformation of the BRB in the GB outrigger configuration
$u_{d,y}$	axial yield deformation of BRB in OB outrigger configuration
$u_{d,ygb}$	axial yield deformation of BRB in GB outrigger system
u_{dj}	axial deformation of the BRB at the j^{th} level of BRB-outrigger
$\mathbf{u_j}$	displacement matrix of the j^{th} core structure segment
u_t	flexural deformation of the outrigger truss in the OB outrigger configuration
$V_{c,\max}$	maximum core structure base shear
$y(x,t)$	lateral displacement of core structure at a point aparts from core structure base with a distance of x and at time of t
y_j	lateral displacement of core structure at top of j^{th} core structure segment
$y_j(x_j,t)$	lateral displacement of core structure within the j^{th} segment, at a point aparts from the j^{th} segment's bottom with a distance of x_j and at time of t
y_{j-1}	lateral displacement of core structure at bottom of j^{th} core structure segment
$y_{\max,n}$	maximum roof displacement when model deforms in the n^{th} mode shape
$y_{top,n}$	roof lateral displacement when BRB yields in the n^{th} mode shape
Γ_n	modal participation factor of the n^{th} mode
T_1	the first mode vibration period
α	ratio of outrigger elevation to building height in single BRB-outrigger system
$\alpha_{opt,heq1}$	optimal outrigger elevation for maximizing the first mode equivalent damping ratio

$\alpha_{opt,Mc}$	optimal outrigger elevation for minimizing $M_{c,\max}$
$\alpha_{opt,T1}$	optimal outrigger elevation for maximizing outrigger effect
$\alpha_{opt,\theta}$	optimal outrigger elevation for minimizing $\theta_{\max}$
α_1	ratio of lower outrigger elevation to upper outrigger elevation in dual BRB-outrigger system
α_2	ratio of upper outrigger elevation to building height in dual BRB-outrigger system
β	BRB compression adjustment factor
ε_y	yield strain of BRB steel core material
ϕ_i	mode shape of the i^{th} mode
$\gamma_{\max}$	maximum inter-story drift
η	inclined angle of the BRB in the GB outrigger configuration
μ	ratio of core structure flexural rigidity at core structure top to the core structure flexural rigidity at core structure base
μ_n	ductility ratio of the n^{th} mode response
$\boldsymbol{\theta}$	matrix of rotations of core structure at BRB-outrigger elevations
θ	roof drift
θ_1	core structure rotation at the height of lower BRB-outrigger elevation in dual BRB-outrigger system
θ_2	core structure rotation at the height of upper BRB-outrigger elevation in dual BRB-outrigger system
θ_j	core structure rotation at top of j^{th} core structure segment
$\theta_{j\text{-}1}$	core structure rotation at bottom of j^{th} core structure segment
$\theta_{\max}$	maximum roof drift ratio
θ_r	roof drift ratio of core structure when BRB yields
θ_y	core structure rotation at the height of outrigger elevation when BRB yields
σ_y	material yield stress of BRB steel core
$\sigma_{y,DM}$	material yield stress of BRB in the DM model
τ	ratio of perimeter column cross-sectional area at building top to perimeter column cross-sectional area at building base
ω	angular frequency
ω_h	BRB material strain hardening factor
ψ	SRSS combined deformed shape
$k_{t,e}$	outrigger flexural stiffness of conventional single outrigger system
$k_{t2,e}$	upper outrigger flexural stiffness of conventional dual outrigger system
$k_{t1,e}$	lower outrigger flexural stiffness of conventional dual outrigger system

BRB	buckling-restrained brace

BRBF	buckling-restrained braced frame
DM	discrete mass with 1-m mass spacing
DM4	discrete mass with 4-m mass spacing
MBM	member by member
MRF	moment resisting frame
MPA	modal pushover analysis
NLRHA	nonlinear response history analysis
SA	spectral analysis
UM	uniform mass
SRSS	square root of the sum of the squares

1

INTRODUCTION

CHAPTER CONTENTS

1.1 Preface 1-3
1.2 Background 1-5
1.2.1 Introduction of outrigger system 1-5
1.2.2 Outrigger system examples 1-7
1.3 Objectives 1-12
1.4 Thesis outline 1-14
1.5 References 1-15

1.1 PREFACE

High-rise buildings are usually symbols of economic development. As the growth of global economic, the numbers and heights of high-rise building have been continually increasing since the 1970s. While the buildings grow higher, the design becomes more critical due to the increased lateral wind and seismic load demands. As a result, numerous of structure types, lateral force resisting system, and energy dissipation devices have been developed (Figure 1.1.1) (Ali and Moon, 2007). The rigid frame (or moment resisting frame, MRF) system is efficient in resisting seismic lateral loads for low-rise buildings. As the building height increases, the braced frame systems are introduced to increase the building lateral stiffness and strength. While the building height keeps increasing, the flexural deformation mode becomes more significant, and the use of MRF or braced frame systems become less efficient. Therefore, the frame-tube systems are introduced and currently widely used in tall buildings around the world. (Sun *et al.*, 2012)

By using the frame-tube system layout, the outrigger system can be easily arranged to connect the perimeter column with the core structure, so that the core structure moment demands can be reduced by mobilizing the perimeter column axial stiffness. Therefore, the outrigger system can be implemented in frame-tube type high-rise buildings easily. The outrigger system has not been classified as one of lateral force resisting system in any code specification, however, the outrigger system has been implemented in many high-rise building projects around the world. The concept of damped-outrigger system has been introduced in the last ten years (Smith and Willford, 2007). The energy dissipation devices such as buckling-restrained braces and viscous dampers are incorporated in the outrigger system so that the damped-outrigger system not only increases the building lateral stiffness but also dissipates energy (Viise, Ragan and Swanson, 2014). The damped-outrigger system is still a new concept for high-rise building design, more researches are needed, and a practical design guide is desired.

On the other hand, for the past 30 years, the buckling-restrained brace (BRB) has been a popular energy dissipation device as it provides high stiffness and strength, meanwhile exhibits stable hysteretic behavior. As shown in Figure 1.1.2, a conventional BRB consists of a steel core and a restrainer made by steel tube with infill mortar. The simple composition and easy fabrication process make the BRB a very economical and high-efficiency solution for buildings in resisting seismic demands. As the BRB is a displacement-depend damper, it is usually adopted as a brace in steel or reinforced concrete frame (buckling-restrained braced frame, BRBF). Because of the buckling-restrained mechanism, the BRB develops similar tension and compression force capacities. Therefore, the BRB can efficiently provide supplemental stiffness to the frame, and also provides satisfactory energy dissipation efficiency after the steel core has yielded. In view of the maturity of fabrication technique and application, the utilization of BRB in a various structural system should be possible and practical.

This research aims to study the behavior of the damped-outrigger system when BRB is adopted as the energy dissipation device. It is anticipated that the proposed structural system can perform satisfactory seismic performance by making full use of advantages of both the damped-outrigger and BRB.

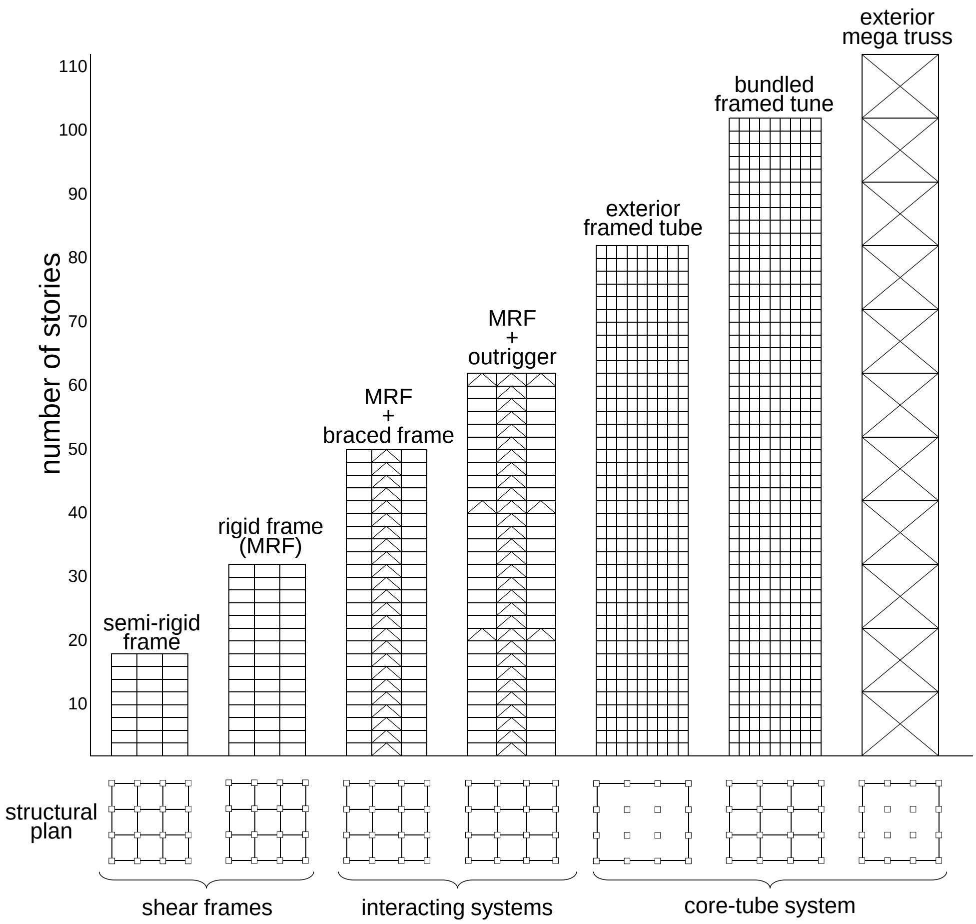


Figure 1.1.1 Structural system comparison

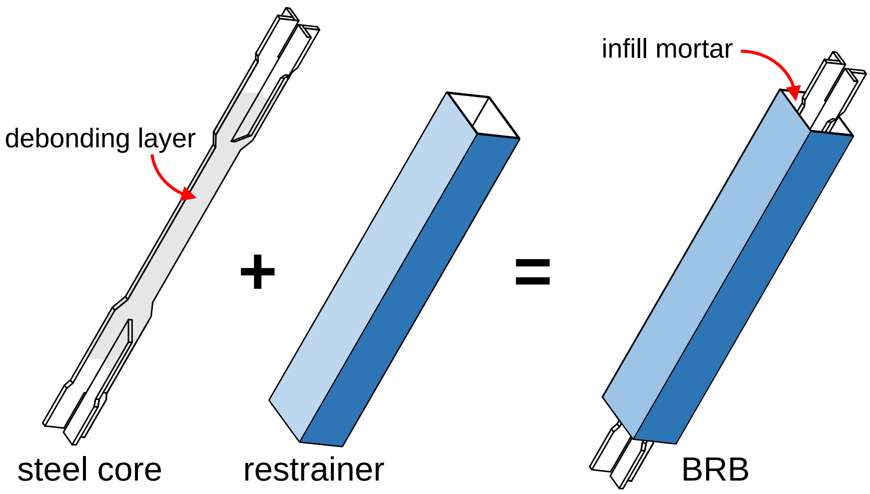
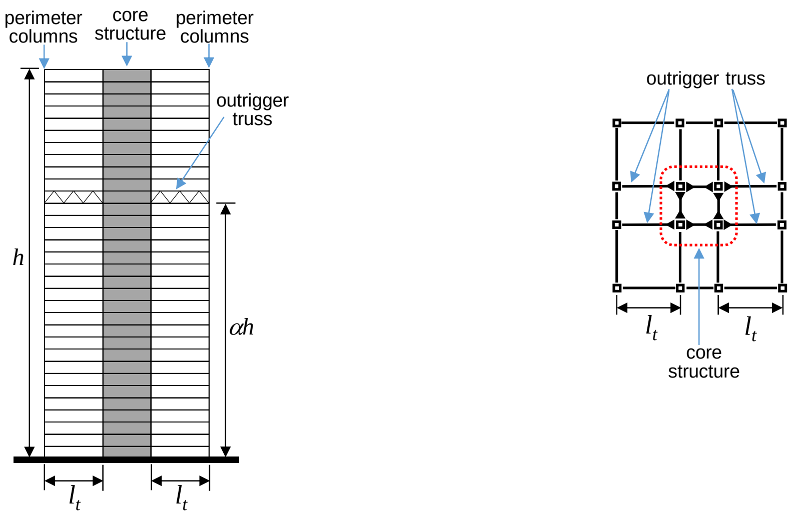


Figure 1.1.2 Composition of a conventional BRB

1.2 BACKGROUND

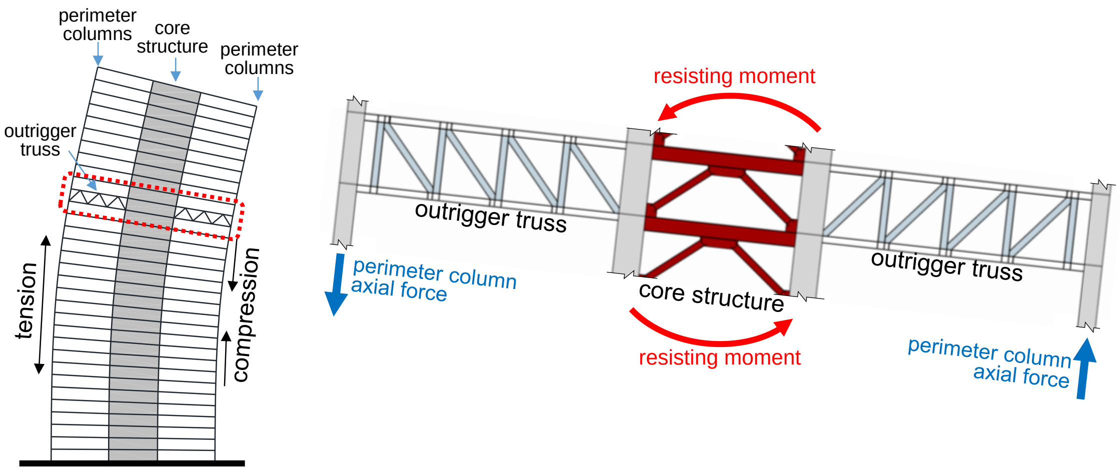
1.2.1 Introduction of outrigger system

The concept of the outrigger system has been introduced and widely implemented in core-tube type buildings around the world since the 1970s to 1980s (Sun *et al.*, 2012). The conventional outrigger mitigates building seismic responses by increasing the system stiffness. Figure 1.2.1a and Figure 1.2.1b show a typical building elevation with a single outrigger and the floor framing plan on the outrigger floor. The core structure provides the majority of the lateral force resistance capacity, and the perimeter columns are responsible for supporting the gravity loads. When the building deforms horizontally as shown in Figure 1.2.2, the core structure’s flexural deformation triggers the relative stiff outrigger truss to rotate. The outrigger truss then triggers additional extension or compression on the perimeter columns below the outrigger. The outrigger mobilizes the axial stiffness of the perimeter columns to apply a resisting moment on the core structure, thus, the flexural demand on the core structure can be reduced. As the number of outrigger increases, the effect in reducing the core structure moment demand is more significant. Figure 1.2.3a shows a building with two layers of outrigger, and Figure 1.2.3b illustrates the distributions of moment demand for the core structure with and without outrigger. However, when the number of outrigger increases, the seismic demand could be increased due to the increased stiffness. In addition, the elastic design concept of outrigger usually results in large force demands in the outrigger members, increasing both complexity and costs in engineering practices (Viise, Ragan and Swanson, 2014).



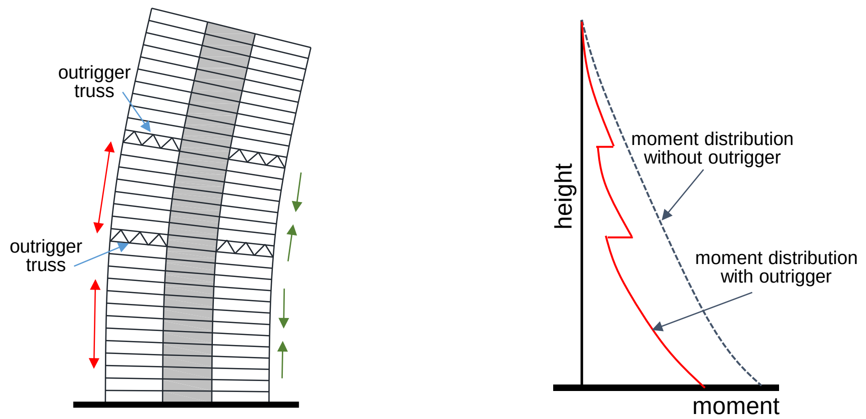
(a) (b)

Figure 1.2.1 The (a) elevation and (b) floor framing plan on outrigger floor of structure with a single layer of outrigger



(a) (b)

Figure 1.2.2 The (a) deformed shape of a structure with outrigger and (b) the enlargement of the outrigger layer



(a) (b)

Figure 1.2.3 The (a) elevation of building with two layers of outrigger, and (b) illustration of core structure moment distribution with and without outrigger

The concept of a damped-outrigger has been proposed to increase the damping ratio, instead of increasing the stiffness, by inserting energy dissipation devices at the outrigger truss ends as shown in Figure 1.2.4. The dampers dissipate energy through the relative movements between outrigger truss ends and the perimeter columns. Since the outrigger truss, the damper, and the perimeter column are acting in series, the force demand for the outrigger truss is limited by the maximum force capacity of the damper. Most of the studies have focused on the damped-outrigger using viscous damper, which is usually designed to resist wind loads. The past studies also indicated that the elevation of damped-outrigger significantly affects the damping ratio developed by the viscous damper. Most of them indicated that the optimal elevation of a damped-outrigger incorporating viscous dampers ranges from 50% to 80% of the building height (Tan, Fang, and Zhou, 2014; Huang and Takeuchi, 2017). In addition, the damped-outrigger with viscous damper has been implemented in real construction projects (Willford and Smith, 2008).

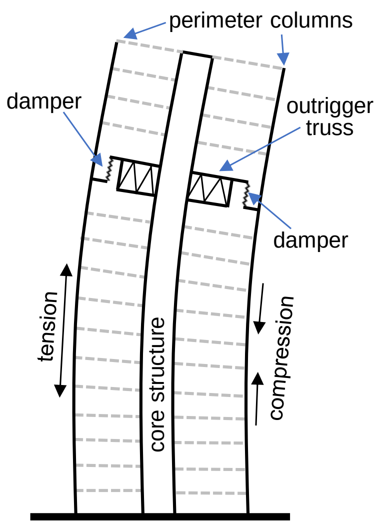


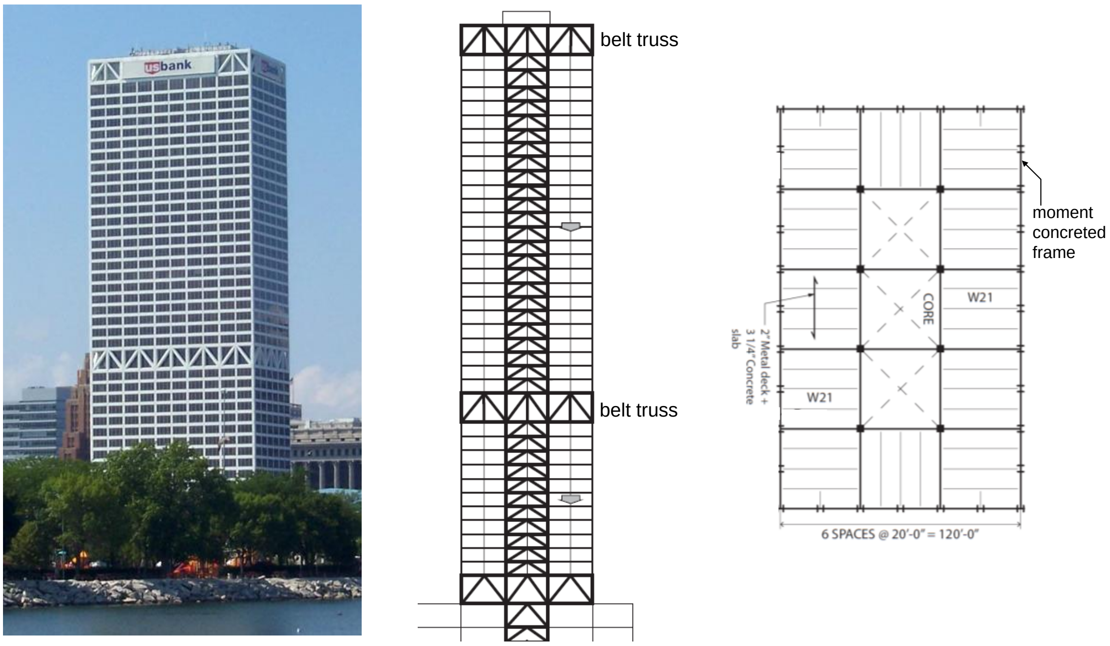
Figure 1.2.4 Illustration of structure with damped-outrigger

1.2.2 Outrigger system examples

This section introduces several examples of a structure incorporating a conventional or damped-outrigger system. The earliest of a constructional project incorporating outrigger system can be traced back to 1970s, and the numbers of the new construction project that uses conventional and damped-outriggers are increasing.

(1) U.S. Bank Center (Milwaukee, U.S.A)

The U.S. Bank Center (Figure 1.2.5a) is a 42-story (183 m tall) all-steel core-and-outrigger system structure and was completed in 1973. As shown in Figure 1.2.5b and Figure 1.2.5c, the structural system contains a core structure continues from the base to the roof and two layers of 2-story deep outrigger placing at the mechanical levels. The belt truss is placed at exterior in order to engage the all the perimeter columns to develop sufficient resisting loads. During the time of designing the building, it was anticipated that the core-and-outrigger system was only applicable to mid-rise buildings. However, the lateral stiffness is increased by 30% due to the utilization of outrigger and belt truss (Sun *et al.*, 2012).



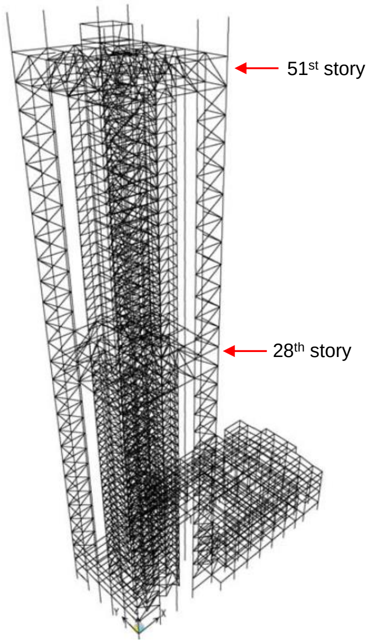
(a) (b) (c)

Figure 1.2.5 The (a) photo, (b) elevation, and (c) structural framing plan of U.S. Bank Center (source, (a): https://en.wikipedia.org/wiki/U.S._Bank_Center_(Milwaukee)#/media/File:US_Bank_Center.jpg, (b) and (c): (Sun *et al.*, 2012))

(2) New York Times Tower (New York, U.S.A)

The New York Times Tower is a 52-story (220 m tall) building and was completed in 2007 (Figure 1.2.6). The core structure with a plan dimension of 20 m by 27 m is connected by two outriggers at the 28^{th} and 51^{st} stories, respectively. Two special features are pointed out in this design example. As the relative stiff outrigger could redistribute the gravity load demands between core structure and perimeter columns, the construction sequence becomes essential in order to make sure the gravity loads are shared accurately. In addition, as the perimeter columns are explored to the weather, the outrigger may be subjected to additional force demands and the floor slope may change under extreme temperature conditions. These special features of using outrigger system were considered in this design example (Sun *et al.*, 2012).

(a)

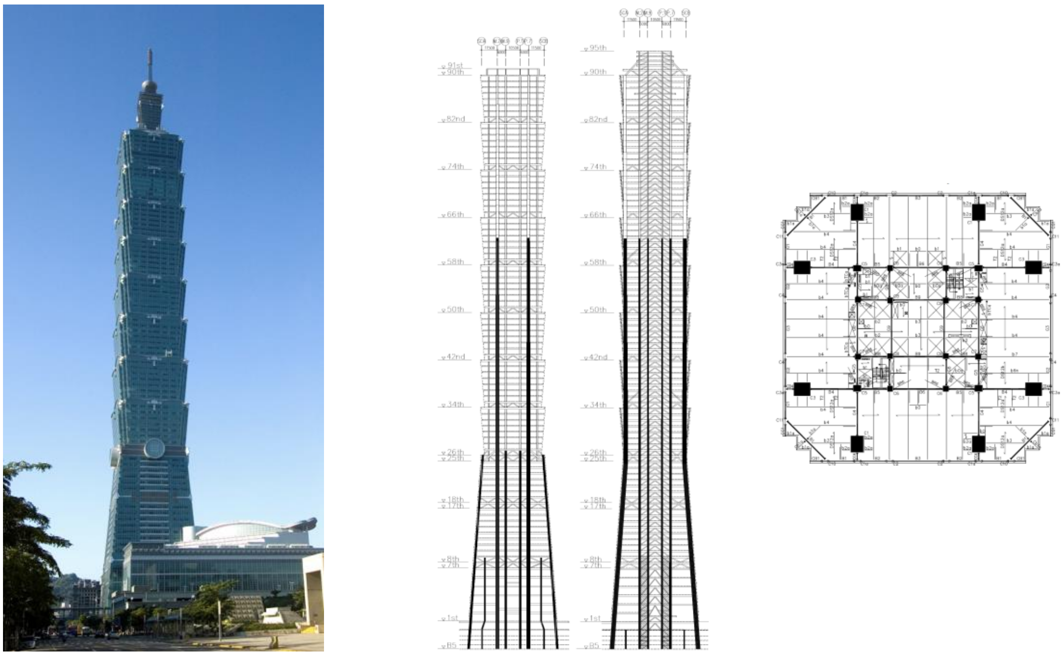


(b)

Figure 1.2.6 The (a) photo and (b) structural system of New York Times Tower (source, (a): https://en.wikipedia.org/wiki/The_New_York_Times_Building#/media/File:New_York_Times_building.png, (b): (Sun *et al.*, 2012))

(3) Taipei 101 (Taipei, Taiwan)

Taipei 101 (Figure 1.2.7a) is a 101-story (508 m tall) building and had been the world's tallest building from 2004 to 2010. The Taipei 101 adopts the core-tube system and multiple outriggers as the main lateral load resisting system in sustaining seismic force. The tuned mass damper is adopted to resist lateral wind load demand. The shape of Taipei 101 is composed of eight 8-story moduli standing above a tapering base, and each of the outriggers is placed at each setback (Figure 1.2.7b to d) between two adjacent 8-story modulus. The braced core structure is connected with 8 concrete-filled 3 m by 2.4 m perimeter columns by outrigger truss together with the belt truss arranged at building perimeter. As the building locates in a high seismic zone, the Taipei 101 was designed to remain elastic under a 285-year-return-period earthquake (Poon *et al.*, 2004; 甘錫瀅, 張敬昌 and 謝紹松, 2017).

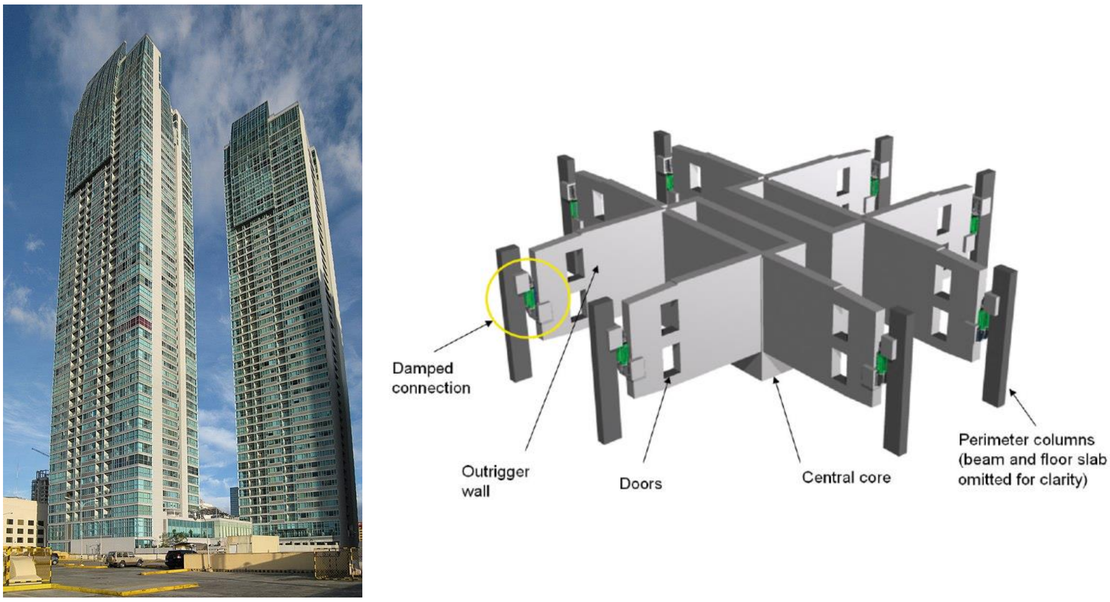


(a) (b) (c) (d)

Figure 1.2.7 The (a) photo, (b) structural elevation at perimeter moment frame, and (c) core structure, and the (d) structural framing plan of Taipei 101
(source, (a): https://commons.wikimedia.org/wiki/File:Taipei101.portrait.altonthompson.jpg, (b)~(d):(甘錫瀅, 張敬昌 and 謝紹松, 2017))

(4) The St. Francis Shangri-La Place (Manila, Philippines)

The St. Francis Shangri-La Place is a 60-story (212 m tall) building and was completed in 2009 (Figure 1.2.8a). The building adopts concrete core structure and perimeter moment frame. A layer of damped-outrigger is arranged at the half-height of the building. As shown in Figure 1.2.8b, the damped-outrigger is composed of eight outrigger walls. Two vertically arranged viscous damper is placed between the outrigger wall end and perimeter column. This damped-outrigger is designed to supply supplemental damping ratio in order to mitigate response due to wind loads. Under the design wind speed, the damping ratio can be increased by 5% to 11% (Infanti, Robinson and Smith, 2008; Willford and Smith, 2008).



(a) (b)

Figure 1.2.8 The (a) photo and (b) concrete outrigger walls connect to perimeter column with viscous dampers

(source, (a): https://en.wikipedia.org/wiki/The_St._Francis_Shangri-La_Place#/media/File:Shangri-La_Towers_Panarama,_Manila,_Philippines_-_panoramio.jpg by Andrew Martin, (b): (Smith and Willford, 2007))

(5) Wilshire Grand Center (Los Angeles, United States)

The Wilshire Grand Center is a 73-story (282 m tall) building and was completed in 2017 (Figure 1.2.9a). This building adopts three groups of outriggers that distribute along with the building height as shown in Figure 1.2.9b. The three outriggers extend from the 28th to the 31st floor, from the 53rd to the 59th floor, and from the 70th to the 73rd floor, respectively. The top and bottom outriggers adopt diagonal BRBs which connect the core structure to perimeter columns directly, while the middle outrigger adopts "X-braced Vierendeel" configuration. Figure 1.2.9c and Figure 1.2.9d show the detail of lower and upper outrigger. Each of the BRBs in lower and upper outriggers has axial force capacity of 9777 kN. The lower outrigger uses two BRBs that are arranged in parallel in order to meet required stiffness and strength as shown in Figure 1.2.9c. The BRBs in the outrigger not only assist in reducing the seismic response but also function as fuses to prevent excessive force demands in the connections between the outrigger and core structure (Nieblas and Tran, 2015; Joseph, Gulec and Schwaiger, 2016). As the BRB dissipates energy through its inelastic hysteretic responses, the outrigger system which adopts BRBs is also classified as damped-outrigger.

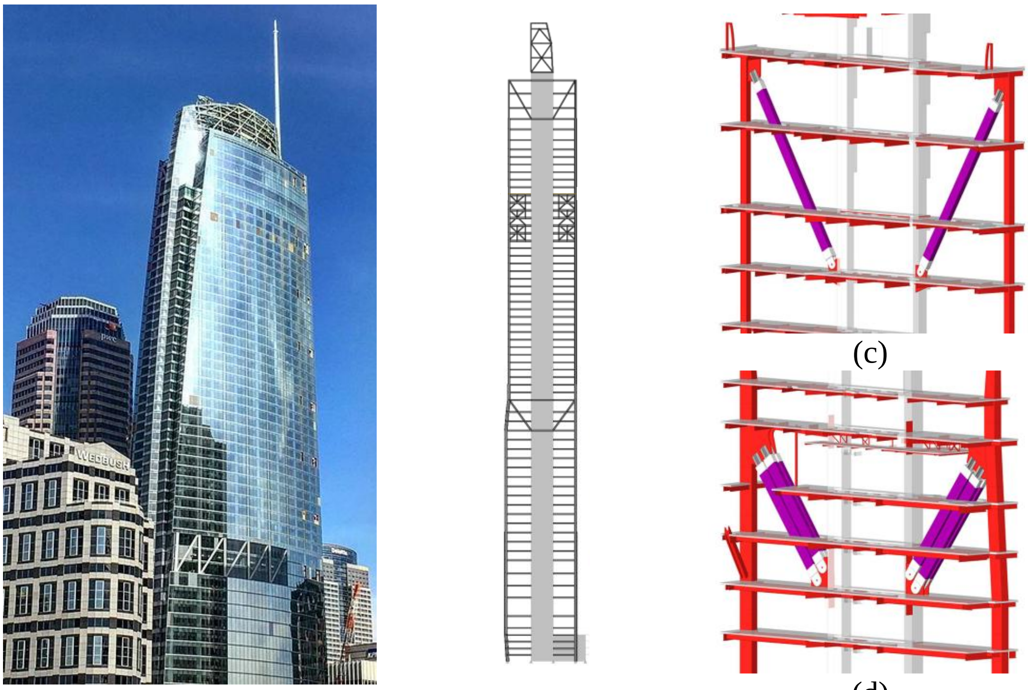


Figure 1.2.9 The (a) photo, (b) elevation, and (c) detail of the upper and (d) bottom outriggers

(source, (a): https://upload.wikimedia.org/wikipedia/commons/thumb/5/56/Wilshire_Grand.jpg/512px-Wilshire_Grand.jpg?uselang=zh-tw by Fredchang931124, (b) to (d): (Nieblas and Tran, 2015))

1.3 OBJECTIVES

Yet the outrigger system has not been classified as one of lateral force resisting system in any code specification, both the conventional and damped-outrigger have been applied in many tall building constructions around the world. This research intends to combine the benefits of using outrigger system and BRB together and proposes the damped-outrigger system incorporating BRB as energy dissipation device (BRB-outrigger) as shown in Figure 1.2.4 and the dampers are replaced with BRBs. Figure 1.3.1 illustrates the arrangement of BRB-outrigger system in a 3D schematic view. In the BRB-outrigger system, the BRBs are arranged vertically between the outrigger truss ends and the perimeter columns. The outrigger truss, BRB, and the perimeter column below the outrigger elevation act in series. Therefore, the maximum force demands in the outrigger truss members and perimeter columns are limited by BRB's axial force capacity. This provides engineers with clear force demands when designing the perimeter columns and outrigger truss members. As shown in Figure 1.3.2, when the building deforms towards the right, the right BRB is in compression and the left in tension. As the BRB's axial deformation exceeds its yield deformation as expected during large earthquakes, the BRB dissipates energy through its inelastic deformation, thereby reducing building seismic response. The stable BRB hysteresis response provides the system with a stable energy dissipation mechanism. During minor earthquakes, a properly designed BRB-outrigger system can behave like a traditional elastic outrigger through BRB's elastic responses. In addition, the feasible BRB strength and stiffness are suitable for various structural configurations.

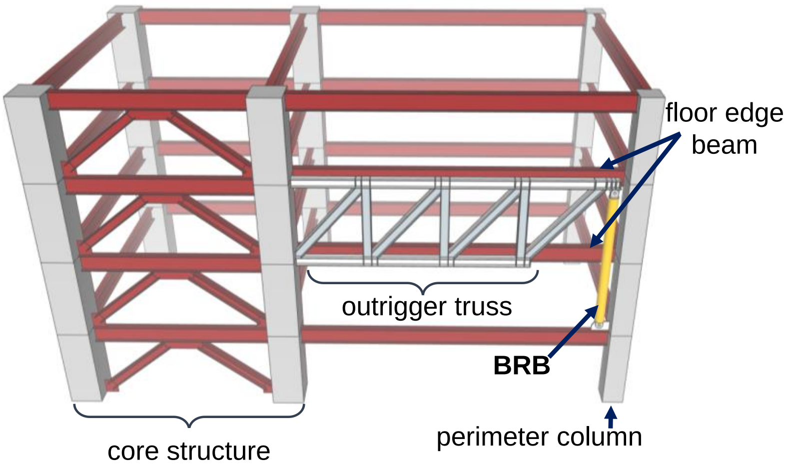


Figure 1.3.1 The 3D schematic view of the BRB-outrigger system

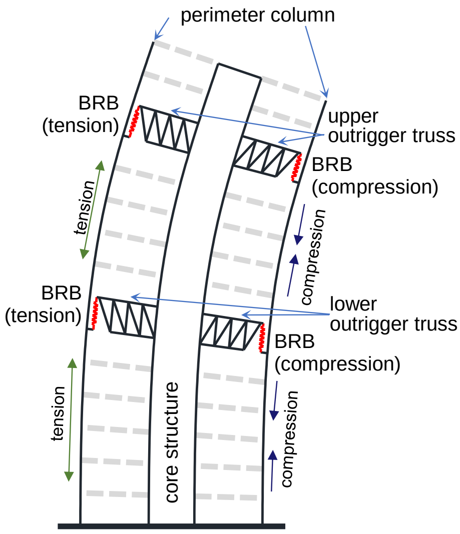


Figure 1.3.2 The laterally deformed structure with two layers of BRB-outrigger

The aims of this study are as follows:

(1) Provide the methods and simplified structure in estimating seismic response for building with one or more layers of BRB-outrigger.

(2) Propose dimensionless parameters to indicate the suitability of using BRB-outrigger system for buildings, and the stiffness relationships between outrigger truss, the BRB, and the perimeter column.

(3) Through the parametric study, investigate the optimal outrigger elevations and the optimal stiffness relationships between the core structure, outrigger truss, BRB, and perimeter columns in order to best reduce seismic response by using BRB-outrigger system.

(4) Provide design recommendations and design examples for structure with single and double layers of BRB-outriggers.

(5) Configure different BRB-outrigger arrangements in order to meet various structural plans and architectural requirements.

1.4 THESIS OUTLINE

This thesis contains eleven chapters, the outlines of each chapter are as follows:

Chapter 1 Introduction, to introduce the background and objective of this research.

Chapter 2 Literature review, to introduce the past research results on both the conventional and damped-outrigger systems.

Chapter 3 Analysis models, to introduce the simplified models used in the parametric study. The effectiveness of using the simplified model to represent the real building is verified by comparing analysis results with a member-by-member model.

Chapter 4 Analysis methods, to introduce the analysis methods used in the parametric study. The analysis methods include spectral analysis and nonlinear response history analysis. The detail process of each analysis method is introduced.

Chapter 5 Programming for parametric study, to introduce the programming and procedure to deal with a large number of analysis in the parametric study.

Chapter 6 Preliminary analysis, to introduce the analysis which is required before performing the parametric study. The method in determining BRB yield deformation (strength) and the equivalent damping ratio developed by the BRB-outrigger are presented.

Chapter 7 Analysis results for optimal design, the analysis results of parameter study are shown, and the optimal design parameters in order to minimize seismic response are discussed in this chapter.

Chapter 8 Design recommendation and design examples, the design procedure and recommendation for the BRB-outrigger system are introduced. The design examples of structure with BRB-outrigger systems are introduced.

Chapter 9 BRB-outrigger configurations, three different types of BRB-outrigger configurations are introduced in order to meet various architectural requirements. The design examples of BRB-outrigger system with different configurations are demonstrated and compared.

Chapter 10 Application of BRB-outrigger, the effect when core structure flexural rigidity varies along with the building height, and when the perimeter column develops different axial stiffness in tension and compression, for example, concrete-filled steel tubes (CFT) column, are discussed in this chapter.

Chapter 11 Conclusions, the conclusions of this research.

1.5 REFERENCES

Ali, M. M. and Moon, K. S. (2007) "Structural Developments in Tall Buildings: Current Trends and Future Prospects," *Architectural Science Review*, 50(3), pp. 205–223. doi: 10.3763/asre.2007.5027.

Huang, B. and Takeuchi, T. (2017) "Dynamic response evaluation of damped-outrigger systems with various heights," *Earthquake Spectra*. doi: 10.1193/051816EQS082M.

Infanti, S., Robinson, J. and Smith, R. (2008) "Viscous Dampers for High-Rise Buildings," in *The 14th World Conference on Earthquake Engineering*.

Joseph, L. M., Gulec, C. K. and Schwaiger, J. M. (2016) "Wilshire Grand: Outrigger Designs and Details for a Highly Seismic Site," *International Journal of High-Rise Buildings*. doi: 10.21022/ijhrb.2016.5.1.1.

Nieblas, G. and Tran, P. (2015) "Wilshire Grand," *Structure magazine*, pp. 34–36.

Poon, D. C. K. *et al.* (2004) "Structural design of Taipei 101, the world's tallest building," *Proceedings of the CTBUH 2004 Seoul Conference, Seoul, Korea*, pp. 271–278.

Smith, R. J. and Willford, M. R. (2007) "The damped outrigger concept for tall buildings," *Structural Design of Tall and Special Buildings*. doi: 10.1002/tal.413.

Sun, C. H. *et al.* (2012) *Outrigger Design for High-Rise Buildings, Outrigger Design for High-Rise Buildings*. London: Routledge. doi: 10.1201/9781315661971.

Takeuchi, T. and Wada, A. (2017) *Buckling- Restrained Braces and Application*. Tokyo: The Japan Society of Seismic Isolation.

Tan, P., Fang, C. and Zhou, F. (2014) “Dynamic characteristics of a novel damped outrigger system,” *Earthquake Engineering and Engineering Vibration*. doi: 10.1007/s11803-014-0231-3.

Viise, J., Ragan, P. and Swanson, J. (2014) “BRB and FVD alternatives to conventional steel brace outriggers,” in *CTBUH 2014 Shanghai Conference*, pp. 691–699.

Watanabe, A. *et al.* (1988) “Properties of brace encased in buckling-restraining concrete and steel tube,” *Proceedings of the 9th world conference on earthquake engineering*, 4, p. 1.

Willford, M. R. and Smith, R. J. (2008) “Performance based seismic and wind engineering for 60 story twin towers in Manila,” in *Proceedings of the 14th World Conference on Earthquake Engineering*, p. 1. Available at: http://www.dl.edi-info.ir/Performance based seismic and wind engineering for 60 story twin towers in manila.pdf.

甘錫瀅, 張敬昌 and 謝紹松 (2017) “台北101大樓的結構工程設計,” 中工高雄會刊, 25(1), pp. 12–21.

2

LITERATURE REVIEW

CHAPTER CONTENTS

2.1 Introduction 2-3
2.2 Research on outrigger system 2-3
2.3 Mechanical behavior of BRB 2-8
2.4 Summary 2-11
2.5 References 2-12

2.1 INTRODUCTION

The research on the application of the outrigger system in tall buildings has begun since 30 years ago. During the past decades, the studies and applications of the damped-outrigger system have continued increasing. On the other hand, BRB has been a reliable and economical solution for seismic issues for more than two decades. In this chapter, the past key studies on conventional and damped-outrigger systems, and the key mechanical properties of BRB are introduced.

2.2 RESEARCH ON OUTRIGGER SYSTEM

In the conventional outrigger systems, the outrigger truss connects the perimeter column to a relatively stiff core structure. When lateral loads, such as seismic or wind loads, are applied to the building, the outrigger system applies a resisting moment on the core structure by mobilizing the axial stiffness of the perimeter columns. The outrigger system is found to effectively reduce the roof drift, inter-story drift, and bending moment of the core structure by increasing the stiffness of the system (Smith and Salim, 1981). However, the elastic design concept of the conventional outrigger could result in excessive force demands on outrigger member and perimeter columns, which increases difficulties and costs in engineering practices (Viise, Ragan and Swanson, 2014).

In order to avoid the excessive force demands in the conventional outrigger members and to implement energy dissipation mechanisms into the outrigger system, the concept of the damped-outrigger was proposed by inserting dampers between the outrigger truss end and the perimeter column (Smith and Willford, 2007). The dampers dissipate energy through the relative movement between the outrigger truss end and the perimeter column. The optimal damped-outrigger elevations in order to maximize the system damping ratio when viscous dampers are adopted were investigated using complex eigenvalue analysis (Chen *et al.*, 2010; Huang and Takeuchi, 2017). In addition, the study (Tan, Fang, and Zhou, 2014) reported the damped-outrigger system using the dynamic stiffness method, which is feasible for buildings with more than two outriggers. The study (Morales-Beltran *et al.*, 2018) reported the optimal single damped-outrigger elevation to be approximately 70% to 80% of the building height. In addition, placing a conventional and damped-outrigger at elevations of 70% and 50% of the building height, respectively, is effective in reducing structural damages (Morales-Beltran *et al.*, 2018). The study (Wu and Li, 2003) reported that the overturning moment at the core structure base can be reduced when one of the multiple outriggers is close to the building foundation. The study (Xing, Zhou and Aguaguiña, 2019) investigated the optimal damped-outrigger elevations when viscous damper and BRB are incorporated through both numerical analysis and experiments. It indicated that the optimal positions of the damped-outrigger with BRB and viscous damper are at the 7th and 5th zone, respectively, based on the 9-zone model structure. The damper-outrigger systems incorporating viscous dampers have been utilized in actual construction projects to reduce the wind load effect (Willford and Smith, 2008). In addition, the BRBs (Watanabe *et al.*, 1988; Takeuchi and Wada, 2017) have been used as outrigger truss members in order to prevent excessive force demands on adjacent members in a real construction project (Joseph, Gulec and Schwaiger, 2016).

(1) Smith and Salim, 1981

Smith and Salim investigated the performance of a structure with multiple outriggers while subjecting to lateral wind loads as illustrated in Figure 2.2.1. The following assumptions are used: (1) the system is linear elastically, (2) only axial forces are included in the columns, (3) the outriggers are rigidly attached to the core, and the core is represented by a cantilever column, (4) the sectional properties of core, column, and outriggers are uniform throughout the building height, and (5) the distribution of wind loading is uniform throughout the building height. This study proposed a dimensionless parameter, ω, to indicate the stiffness relationships between core, column, and outrigger. It was found that the drift and core moment can be effectively reduced when outriggers are adopted. The cases with one to four layers of outriggers were considered. When $\omega = 0.5$, the optimal outrigger elevations in order to minimizing drift response are 70% of the building height for the case with one outrigger layer, 80% and 50% of the building height for the case with two outrigger layers, 80%, 60%, and 40% of the building height for the case with three outrigger layers, and 80%, 70%, 50%, and 40% of the building height for the case with four outrigger layers. The larger number of outrigger is better in reducing drift response. However, each additional outrigger has a lesser effect. This study indicates that the case with four outrigger layers appears to be the maximum justifiable number for controlling drift response. In addition, if the outriggers are placed at elevations lower than the optimal elevations, the core moment can be better reduced.

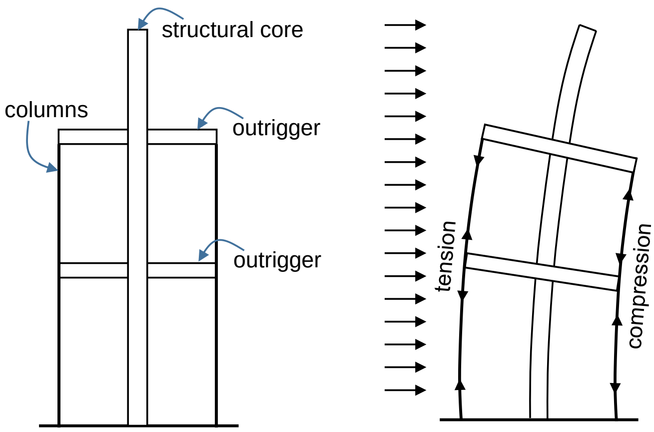


Figure 2.2.1 (a) Outrigger-braced structure and (b) response to lateral loading (Smith and Salim, 1981)

(2) Wu and Li, 2003

Wu and Li investigated the seismic behavior of multi-outrigger-braced tall buildings. The trapezoid form of lateral wind load distribution was considered. The effects of outrigger elevations and structural stiffness on the fundamental vibration period, top drift, and core structure base moment were studied. The assumptions used in this study are similar to the previous one (Smith and Salim, 1981). It was found that when the lateral load patterns change from uniform to triangular, the optimal outrigger elevations are 4% to 5% higher than those of structure subjected to uniform lateral load. In addition, when two layers of outriggers are adopted, and when one of the two outriggers is placed at the building top, the optimal lower outrigger is lower than the optimal outrigger when only one outrigger is adopted. On the other hand, this study also indicated that the overturning moment at core structure base can be reduced when one of the multiple outriggers is close to the building foundation.

(3) Chen *et al.*, 2010

Chen et al. investigated the seismic behavior of structure with a layer of damped-outrigger employing viscous dampers numerically. Figure 2.2.2 shows the simplified structure model. The core structure is represented by a cantilever column with uniform cross-sectional property and a fixed end at its bottom. The outrigger truss is assumed to be rigid. The lateral deformation of the core structure is calculated based on D'Alembert's principle and Bernoulli-Euler beam theory. A Heaviside step function is used to indicate two different vibrations of the core structure above and below the damped-outrigger. By applying appropriate boundary conditions, the system's complex natural frequencies, mode shapes, and damping ratios can be obtained. The optimal outrigger elevations that result in the maximum damping ratio value for the first five modes are studied. The approximate equations to decide the optimal outrigger elevations and its corresponding damping ratios were provided for engineering design purpose.

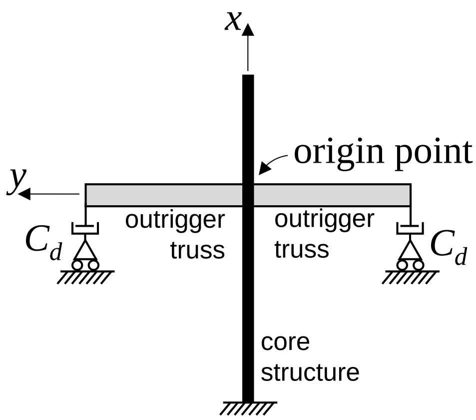


Figure 2.2.2 Cantilever column with damped outriggers (Chen *et al.*, 2010)

(4) Huang and Takeuchi, 2017

Huang and Takeuchi also investigated the dynamic response of damped-outrigger system employing viscous dampers by using the complex eigenvalue analysis model as shown in Figure 2.2.3. The core structure's lateral deformation is solved based on D'Alembert's principle and Bernoulli-Euler beam theory. A Heaviside step function is used to separate the core structure's dynamic behavior above and below the outrigger. The flexural stiffness of the outrigger truss (k_t) and the axial stiffness of the perimeter column (k_c) were considered. Similar to the previous study (Chen *et al.*, 2010), this study utilized complex eigenvalue analysis on a continuous cantilever column model. The accuracy was verified by performing analysis on the member-by-member model. A response spectrum analysis procedure, that considers reductions of response spectrum due to the increased damping ratio, was used to evaluate the seismic peak response of a structure. Dimensionless parameters that describe the required damper sizes and relationships between outrigger truss flexural stiffness and core structure rigidity were used to in the analysis. This study indicated that for structure with a single layer of damped-outrigger, in order to maximize the modal damping ratio, the optimal outrigger elevation is between 0.5 to 0.8 of building height.

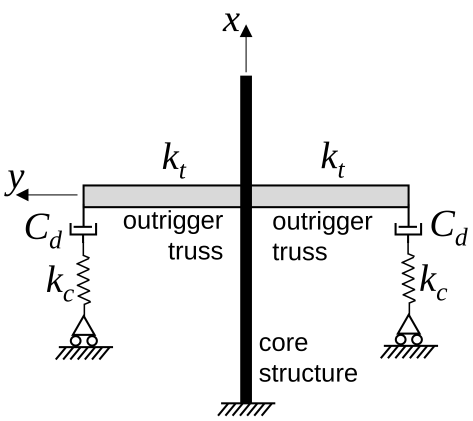


Figure 2.2.3 Complex eigenvalue analysis method (Huang and Takeuchi, 2017)

(5) Tan, Fang and Zhou, 2014

Tan et al. investigated the seismic behavior of a structure with single layer of damped-outrigger that employs viscous damper by using dynamic stiffness method (DSM). The effect of viscous damper and perimeter column is simplified as a rotational spring attached on the core structure. The lateral deformation of the core structure is also solved based on D'Alembert's principle and Bernoulli-Euler beam theory and is expressed in matrices format. Instead of using a Heaviside step function to distinguish the core structure above and below outrigger, the lateral deformation of each core structure segment is expressed in matrix form. The dynamic stiffness of the overall system can be combined from individual core structure segments and the rotational springs. The DSM is more flexible when the number of outriggers is greater than 1. This research indicated that the stiffness ratio of the core structure to the column should be less than 4 in order to achieve a 5% additional damping ratio.

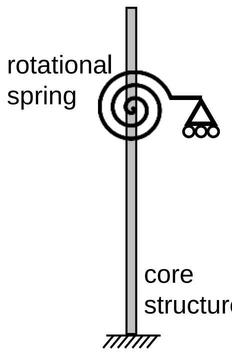


Figure 2.2.4 Analytical model used in previous study (Tan, Fang and Zhou, 2014)

(6) Morales-Beltran *et al.*, 2018

Morales-Beltran et al. studied the seismic behavior of structure with single and double outriggers by performing nonlinear response history analysis. Both the damped- and conventional outriggers were considered in this research. It was reported that for single damped-outrigger, the optimal outrigger elevation is approximately 0.7 to 0.8 times building height. The cases of double damped-outrigger and combined damped and conventional outriggers display a larger increase of damping ratio than single damped-outrigger configuration. In addition, placing a damped-outrigger and a conventional outrigger at the elevations of 0.5 and 0.7 times the building height, respectively, is effective in reducing structural damages. This study also pointed out that when both seismic performances and cost are considered, the single outrigger structure may be a more economical solution than the multiple outrigger structure.

2.3 MECHANICAL BEHAVIOR OF BRB

The concept of BRB has been proposed in Japan since the 1970s through a series of experiments (Wakabayashi *et al.*, 1973; Watanabe *et al.*, 1988). By continuing this concept, the researches in Japan, India, US, and Taiwan have developed various types of BRB and conducted experiments (Uang, Nakashima and Tsai, 2004; Takeuchi and Wada, 2017). As indicated in Figure 2.3.1, the BRB is capable of developing full yield axial force capacities in both tension and compression. Therefore, the buckling-restrained braced frame (BRBF) can develop similar lateral force resistance when the brace is in tension and in compression. If compared with the conventional braced frame, the good hysteretic behavior of BRB leads the BRBF to exhibit more satisfactory energy dissipation performance than the conventional braced frame.

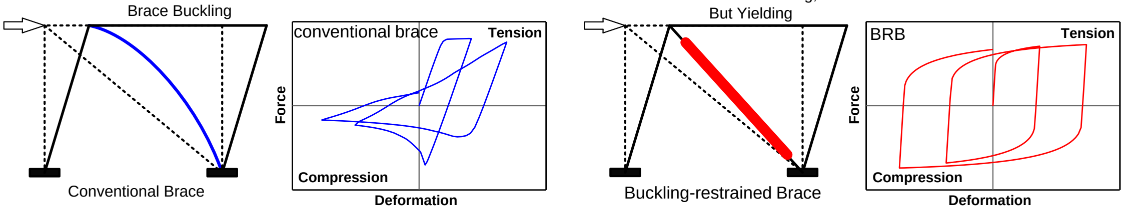


Figure 2.3.1 Comparison between conventional brace and BRB

As shown in Figure 1.1.2 and Figure 2.2.3, the composition of a conventional BRB contains the steel core, the restraining member, and the debonding mechanism:

(1) Steel core: The steel core is usually composed of steel plates, and is also the axial force resisting member. As shown in Figure 2.3.2, the steel core can be divided into three segments (plastic, transition, and elastic segments). The cross-sectional area of the plastic segment (A_p) is the smallest among the three segments. The plastic segment deforms inelastically during moderate to maximum considered earthquakes and dissipates energy. The gradually changing cross-sectional area in the transition segment avoids stress concentration between the two segments that deform elastically and inelastically. The cross-sectional area in the elastic segment (A_e) is the greatest among the three segments. The elastic segment ensures the connection between BRB and frame to deform elastically and provides sufficient rigidity to avoid buckling outside the restraining member end.

(2) Restraining member: When the BRB is subjected to compression, the restraining member prevents the steel core from global buckling and local bulging failure (Lin *et al.*, 2016) due to steel core high mode buckling. The restraining member is usually composed by a steel tube with infill mortar.

(3) Debonding mechanism: Due to the Poisson effect and high mode buckling of the steel core, the steel core contacts with the restraining member and develops fiction forces when it is subjected to compression. In order to avoid developing excessive friction force, a layer of debonding material is used to cover the steel core before it is encased in a restraining member. The debonding layer is usually made by silicone or rubber panels. In addition, the debonding mechanism can also be achieved by applying an air gap between the steel core and restraining member.

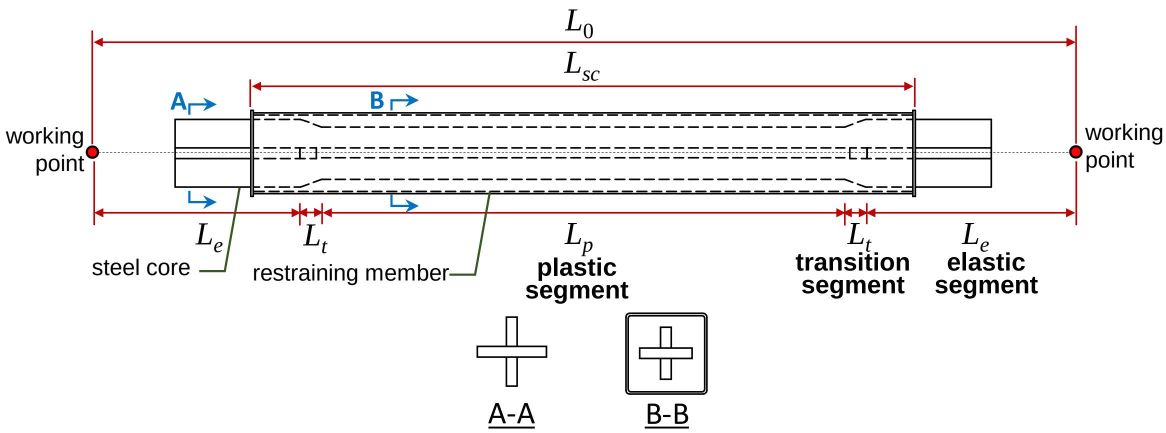


Figure 2.3.2 Composition of a conventional BRB with cruciform steel core

The BRB axial yield force capacity (N_y) can be calculated as follows:

$$N_y = A_p \sigma_y \tag{2.3.1}$$

Where σ_y is material nominal yield stress of the BRB steel core. The maximum axial force capacity of the BRB (N_{cu}) is calculated as follows (ANSI/AISC 341-16, 2016):

$$N_{cu} = \beta \omega_h R_y N_y \tag{2.3.2}$$

Where β, ω_h, and R_y are the BRB compression adjustment factor, the steel core material strain hardening factor, and the ratio of the expected yield stress to the specified minimum yield stress of steel core material, respectively. The β = 1.15 is used in this research. The values of ω_h and R_y for different materials used in this research are shown in Table 2.3.1. The effective axial stiffness of the BRB (k_{eff}) is calculated by connecting the plastic, transition, and elastic segments in series connection as follows:

$$k_{eff} = \frac{EA_pA_tA_e}{L_pA_tA_e + 2L_tA_pA_e + 2L_eA_pA_t} \tag{2.3.3}$$

Where A_t (=(A_p+A_e)/2) is the equivalent cross-sectional area of the transition segment, and E is the modulus of elasticity of steel.

Table 2.3.1 The ground motions used in NLRHA

steel material	σ_y (MPa)	R_y	ω_h
SN490	325	1.2	1.3
SN400	235	1.3	1.5
SM570	420	1.1	1.3

In this study, the cumulative plastic deformation ratio (R_{CPD}) (ANSI/AISC 341-16, 2016) is used to indicate the magnitude of inelastic deformation of a BRB. Figure 2.2.3 shows the relationship between steel core stress and strain. The R_{CPD} is the ratio of summation of plastic strain to the yield strain, and can be calculated as follows:

$$R_{CPD} = \frac{\sum_i \Delta\varepsilon_p^i}{\varepsilon_y} \tag{2.3.4}$$

Where ε_y is the yield strain of steel core material. The greater R_{CPD} value indicates the larger amount of plastic deformation is gained. Too large R_{CPD} value indicates the possibility of steel core fracture is high.

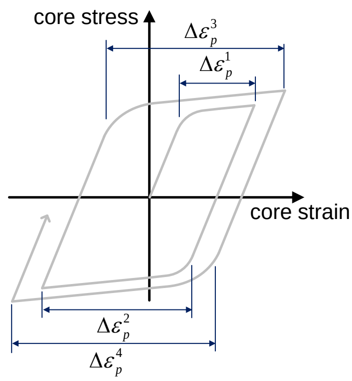


Figure 2.3.3 Illustration of cumulative plastic deformation of BRB

2.4 SUMMARY

This study investigates the seismic behavior of a multiple-outrigger system incorporating BRB as an energy dissipation device (BRB-outrigger). The concept of damped-outrigger is similar to the previous studies. However, the BRB-outrigger provides large stiffness to the system and dissipates energy through the BRB's hysteretic behavior. In addition, the maximum force demands for the outrigger truss members and perimeter column are effectively limited by the maximum axial force capacity of the BRBs. When the viscous dampers are employed in the damped-outrigger system, the velocity-dependent viscous dampers are efficient in limiting the maximum acceleration and also applicable to control wind load vibration. However, the design requirements and velocity ranges corresponding to the wind and seismic demands are usually different. The high strength and stiffness of BRB make the BRB-outrigger system to be more suitable in controlling the maximum inter-story drift response and mitigating damages for non-structural elements of the building. In addition, the BRB-outrigger system can function as a conventional outrigger system during frequent small earthquakes through its high elastic stiffness of the BRBs, and it can dissipate energy during moderate to maximum considered earthquake through the BRBs' stable hysteretic behavior.

When BRB is adopted as the energy dissipation device in the damped-outrigger system, the dynamic characteristics of the overall structure and the associated seismic demand are affected by both the outrigger elevation and the stiffness of the BRB. Therefore, the aims of this study are to propose a method to evaluate the seismic response of the multiple BRB-outrigger system with various outrigger elevations, investigate the optimal outrigger elevations in order to minimize the seismic response, and study the relationships between the flexural rigidity of the core structure and the axial stiffness of the perimeter columns and the BRBs. The dynamic characteristics are studied and the seismic response is evaluated using the spectral analysis (SA), incorporating the concept of equivalent damping ratio to include the inelastic responses of the BRBs. The SA results are then validated by performing a nonlinear response history analysis (NLRHA). The continuous seismic response distributions with various outrigger elevations are demonstrated. The maximum roof drift ratio, inter-story drift, overturning moment at the core structure base, and the additional axial force demand for the perimeter column are adopted as indicators for judging the optimal outrigger elevations. This study aims to propose a design recommendation and presents design charts for preliminary design purposes.

2.5 REFERENCES

ANSI/AISC 341-16 (2016) *Seismic Provisions for Structural Steel Buildings, American Institute of Steel Construction*. doi: 111.

Chen, Y. *et al.* (2010) "Analysis of Tall Buildings with Damped Outriggers," *Journal of Structural Engineering*. doi: 10.1061/(asce)st.1943-541x.0000247.

Huang, B. and Takeuchi, T. (2017) "Dynamic response evaluation of damped-outrigger systems with various heights," *Earthquake Spectra*. doi: 10.1193/051816EQS082M.

Joseph, L. M., Gulec, C. K. and Schwaiger, J. M. (2016) "Wilshire Grand: Outrigger Designs and Details for a Highly Seismic Site," *International Journal of High-Rise Buildings*. doi: 10.21022/ijhrb.2016.5.1.1.

Lin, P. C. *et al.* (2016) "Seismic design and testing of buckling-restrained braces with a thin profile," *Earthquake Engineering and Structural Dynamics*. doi: 10.1002/eqe.2660.

Morales-Beltran, M. *et al.* (2018) "Energy dissipation and performance assessment of double damped outriggers in tall buildings under strong earthquakes," *Structural Design of Tall and Special Buildings*. doi: 10.1002/tal.1554.

Smith, R. J. and Willford, M. R. (2007) "The damped outrigger concept for tall buildings," *Structural Design of Tall and Special Buildings*. doi: 10.1002/tal.413.

Smith, S. and Salim, I. (1981) "Parameter Study of Outrigger-Braced Tall Building Structures," *Journal of the Structural Division*.

Takeuchi, T. and Wada, A. (2017) *Buckling-Restrained Braces and Application*. Tokyo: The Japan Society of Seismic Isolation.

Tan, P., Fang, C. and Zhou, F. (2014) "Dynamic characteristics of a novel damped outrigger system," *Earthquake Engineering and Engineering Vibration*. doi: 10.1007/s11803-014-0231-3.

Uang, C. M., Nakashima, M. and Tsai, K. C. (2004) "Researc and Aplication of Buckling-Restrained Braced Frames," *Steel Structures*.

Viise, J., Ragan, P. and Swanson, J. (2014) "BRB and FVD alternatives to conventional steel brace outriggers," in *CTBUH 2014 Shanghai Conference*, pp. 691–

699.

Wakabayashi, M. *et al.* (1973) "Experimental study of elasto-plastic properties of precast concrete wall panels with built-in insulating braces," *Summaries of technical papers of annual meeting, Architectural Institute of Japan*, pp. 1041–1044.

Watanabe, A. *et al.* (1988) "Properties of brace encased in buckling-restraining concrete and steel tube," *Proceedings of the 9th world conference on earthquake engineering*, 4, p. 1.

Willford, M. R. and Smith, R. J. (2008) "Performance based seismic and wind engineering for 60 story twin towers in Manila," in *Proceedings of the 14th World Conference on Earthquake Engineering*, p. 1. Available at: http://www.dl.edi-info.ir/Performance based seismic and wind engineering for 60 story twin towers in manila.pdf.

Wu, J. R. and Li, Q. S. (2003) "Structural performance of multi-outrigger-braced tall buildings," *Structural Design of Tall and Special Buildings*, 12(2), pp. 155–176. doi: 10.1002/tal.219.

Xing, L., Zhou, Y. and Aguaguiña, M. (2019) "Optimal vertical configuration of combined energy dissipation outriggers," *Structural Design of Tall and Special Buildings*. doi: 10.1002/tal.1579.

3

ANALYTICAL MODELS

CHAPTER CONTENTS

3.1 Introduction 3-3
3.2 Simplified models 3-3
3.2.1 Uniform mass model 3-7
3.2.2 Discrete mass model 3-11
3.3 Parameter definitions 3-13
3.3.1 Parameters for single BRB-outrigger system 3-13
3.3.2 Parameters for dual BRB-outrigger system 3-14
3.4 Analysis procedure 3-16
3.5 Member-by-member models 3-33
3.5.1 32-story MBM model 3-34
3.5.2 96-story MBM model 3-36
3.6 Summary 3-39
3.7 References 3-40

3.1 INTRODUCTION

This chapter presents the analytical models used in this study. In order to investigate the seismic performance of structures with BRB-outrigger, the BRB-outrigger structure is simplified (simplified structure) for the purpose of the parametric study. Two types of simplified structures were introduced, the first type of simplified structure is constructed from numerical calculation procedure, and the second type of simplified model is constructed by using OpenSees (McKenna, 1997) directly. The effectiveness of using the simplified structures to represent the real building with BRB-outriggers is verified by comparing with the analysis results calculated from using member-by-member (MBM) models constructed by using OpenSees, and all the details such as floors and real dimension of core structure are included. In addition, the dimensionless parameters for the purpose of parametric study, and the methods in constructing analytical models are introduced.

3.2 SIMPLIFIED MODELS

A core-tube type structure with n levels of BRB-outrigger and a height of h is simplified as shown in Figure 3.2.1, where l_t is the outrigger span. The core structure is represented as a cantilever column, and the width of the core structure is neglected. The lateral flexural rigidity and mass of the building are assumed to be concentrated at the core structure. The bases of the perimeter columns are free to rotate about the out-of-plane axis. The two ends of each BRB are pin-connected to the perimeter column and outrigger truss ends, respectively. The ends of outrigger truss close to the core structure have full moment transfer capacity. In the j^{th} level of the BRB-outrigger, k_{tj} and k_{dj} are the flexural stiffness of outrigger truss with a span of l_t and the BRB axial stiffness, respectively. The elevation of the j^{th} level of BRB-outrigger (h_j) is defined as the distance measuring from the perimeter column base to the connection between BRB to perimeter column. The h_j is calculated as follows:

$$h_j = \prod_{i=j}^{n} \alpha_i h \tag{3.2.1}$$

As the BRB-outriggers apply resisting moments on the core structure, the structure can be further simplified to a continuous cantilever column with rotational springs attached as shown in Figure 3.2.2. However, as all the BRB-outriggers share the same perimeter column, the rotations of the rotational springs are interdependent to each other. If $\mathbf{M_o}$ is the matrix of moments applied on the core structure by the BRB-outriggers and $\boldsymbol{\theta}$ is the matrix of core structure rotations at each outrigger elevation, then the relationship between $\mathbf{M_o}$ and $\boldsymbol{\theta}$ can be expressed as follows:

$$\mathbf{M_o} = \begin{bmatrix} M_{o1} \\ M_{o2} \\ \vdots \\ M_{o(n-1)} \\ M_{o(n)} \end{bmatrix} = 2l_t \begin{bmatrix} k_{d1}u_{d1} \\ k_{d2}u_{d2} \\ \vdots \\ k_{dn-1}u_{dn-1} \\ k_{dn}u_{dn} \end{bmatrix} = 2l_t\mathbf{C}\mathbf{D}^{-1} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{n-1} \\ \theta_n \end{bmatrix} = 2l_t\mathbf{C}\mathbf{D}^{-1}\boldsymbol{\theta} = \mathbf{k_g}\boldsymbol{\theta} \tag{3.2.2}$$

Where $\mathbf{k_g}$ is the rotational spring stiffness matrix resulting from the BRB-outrigger effect, and u_{dj} is the axial deformation of the BRB in the j^{th} level of BRB-outrigger. The **C** and **D** matrices are as follows:

$$\mathbf{C} = \begin{bmatrix} k_{c1}+k_{c2} & -k_{c2} & \cdots & 0 & 0 \\ -k_{c2} & k_{c2}+k_{c3} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & k_{cn-1}+k_{cn} & -k_{cn} \\ 0 & 0 & \cdots & -k_{cn} & k_{cn} \end{bmatrix} \tag{3.2.3}$$

$$\mathbf{D} = \begin{bmatrix} \frac{1}{l_t}+\frac{k_{c1}+k_{c2}}{l_t}\left(\frac{1}{k_{d1}}+\frac{1}{k_{t1}}\right) & -\frac{k_{c2}}{l_t}\left(\frac{1}{k_{d1}}+\frac{1}{k_{t1}}\right) & \cdots & 0 & 0 \\ -\frac{k_{c2}}{l_t}\left(\frac{1}{k_{d2}}+\frac{1}{k_{t2}}\right) & \frac{1}{l_t}+\frac{k_{c2}+k_{c3}}{l_t}\left(\frac{1}{k_{d2}}+\frac{1}{k_{t2}}\right) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{l_t}+\frac{k_{c(n-1)}+k_{cn}}{l_t}\left(\frac{1}{k_{d(n-1)}}+\frac{1}{k_{t(n-1)}}\right) & -\frac{k_{cn}}{l_t}\left(\frac{1}{k_{d(n-1)}}+\frac{1}{k_{t(n-1)}}\right) \\ 0 & 0 & \cdots & -\frac{k_{cn}}{l_t}\left(\frac{1}{k_{dn}}+\frac{1}{k_{tn}}\right) & \frac{1}{l_t}+\frac{k_{cn}}{l_t}\left(\frac{1}{k_{dn}}+\frac{1}{k_{tn}}\right) \end{bmatrix} \tag{3.2.4}$$

Where k_{cj} is the axial stiffness of the perimeter column of segment j ($1 \leq j \leq n$) as shown in Figure 3.2.1. It should be noted that the Equation (3.2.2) is valid when all the BRB deform elastically. In addition, for simplicity, the cross-sectional properties of the perimeter columns are assumed to be identical along the building height. The effect when perimeter column cross-sectional area reduces with increasing elevation on seismic response is discussed in Chapter 10. In this research, the seismic response and optimal design for structure with single and dual BRB-outrigger are studied.

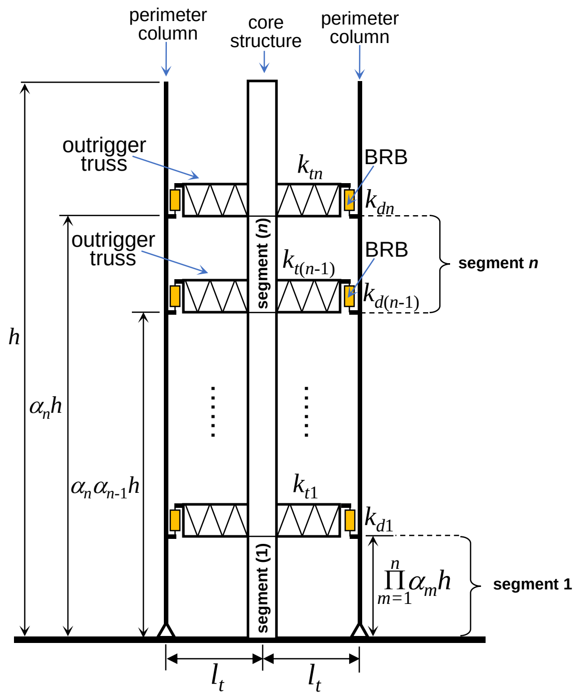


Figure 3.2.1 The simplified structure with *n* levels of BRB-outrigger

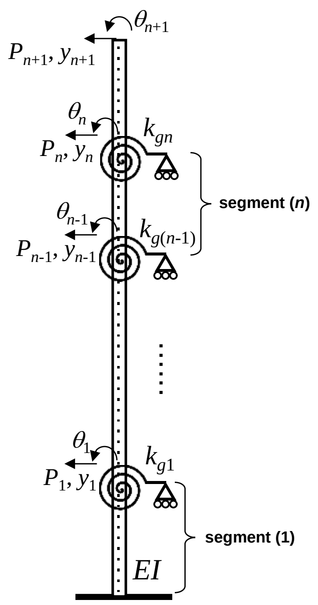


Figure 3.2.2 The simplified structure with *n* level of rotational springs

Figure 3.2.3 and Figure 3.2.4 illustrate the simplified structure with single BRB-outrigger. The outrigger elevation is expressed by α, and the BRB axial stiffness, outrigger truss flexural stiffness, and the perimeter column stiffness with a height of h are expressed as k_d, k_t, and k_c, respectively. The **C** and **D** matrices and the Equation (3.2.2) can be expressed as follows:

$$\mathbf{C} = \left[k_{c1}\right] = \left[\frac{k_c}{\alpha}\right] \tag{3.2.5}$$

$$\mathbf{D} = \left[\frac{1}{l_t} + \frac{k_{c1}}{l_t}\left(\frac{1}{k_d} + \frac{1}{k_t}\right)\right] \tag{3.2.6}$$

$$\mathbf{M_o} = \left[M_{o1}\right] = \frac{2l_t^2}{\dfrac{\alpha}{k_c} + \dfrac{1}{k_d} + \dfrac{1}{k_t}}\theta = \frac{2l_t^2}{\dfrac{1}{k_b} + \dfrac{1}{k_d}}\theta = k_g\theta,\ k_g = \frac{2l_t^2}{\dfrac{\alpha}{k_c} + \dfrac{1}{k_d} + \dfrac{1}{k_t}} = \frac{2l_t^2}{\dfrac{1}{k_b} + \dfrac{1}{k_d}} \tag{3.2.7}$$

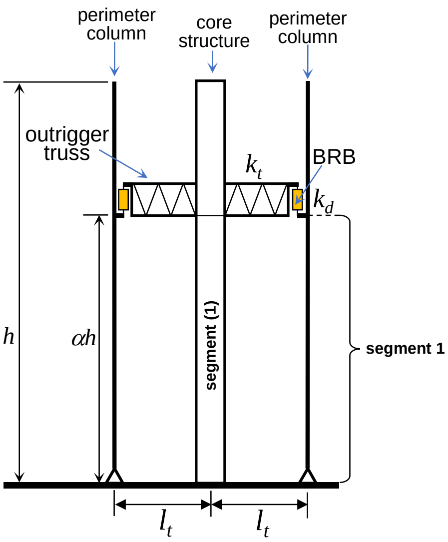


Figure 3.2.3 The simplified structure with single BRB-outrigger

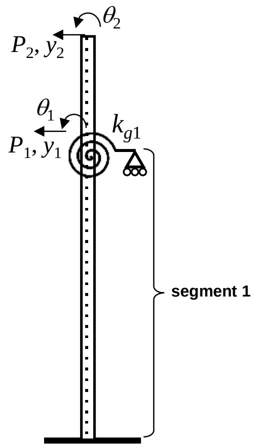


Figure 3.2.4 The simplified structure with one level of rotational spring

Figure 3.2.5 and Figure 3.2.6 illustrate the simplified structure with dual BRB-outrigger. The lower and upper outrigger elevations are $\alpha_1\alpha_2 h$ and $\alpha_2 h$, respectively. The lower and upper outrigger truss flexural stiffnesses are k_{t1} and k_{t2}, respectively. The BRB in the lower and upper BRB-outriggers are known as BRB_1 and BRB_2, respectively. The axial stiffness of BRB_1 and BRB_2 are k_{d1} and k_{d2}, respectively. The **C** and **D** matrices can be expressed as follows:

$$\mathbf{C} = \begin{bmatrix} k_{c1} + k_{c2} & -k_{c2} \\ -k_{c2} & k_{c2} \end{bmatrix} \tag{3.2.8}$$

$$\mathbf{D} = \begin{bmatrix} \dfrac{1}{l_t} + \dfrac{k_{c1} + k_{c2}}{l_t}\left(\dfrac{1}{k_{d1}} + \dfrac{1}{k_{t1}}\right) & -\dfrac{k_{c2}}{l_t}\left(\dfrac{1}{k_{d1}} + \dfrac{1}{k_{t1}}\right) \\ -\dfrac{k_{c2}}{l_t}\left(\dfrac{1}{k_{d2}} + \dfrac{1}{k_{t2}}\right) & \dfrac{1}{l_t} + \dfrac{k_{c2}}{l_t}\left(\dfrac{1}{k_{d2}} + \dfrac{1}{k_{t2}}\right) \end{bmatrix} \tag{3.2.9}$$

$$\mathbf{k_g} = 2l_t\mathbf{C}\mathbf{D}^{-1} \tag{3.2.10}$$

Based on the simplified structure introduced above, two analytical models are developed. The uniform mass (UM) mass model assumes that the mass is uniformly distributed along with the core structure height. The UM model is used for performing modal analysis and spectral analysis (SA). In addition to the UM model, the discrete mass (DM) model assumes the masses are lumped at the nodes that uniformly distribute along with the core structure height. The DM model is used to perform modal pushover analysis (MPA) and nonlinear response history analysis (NLRHA) by using OpenSees.

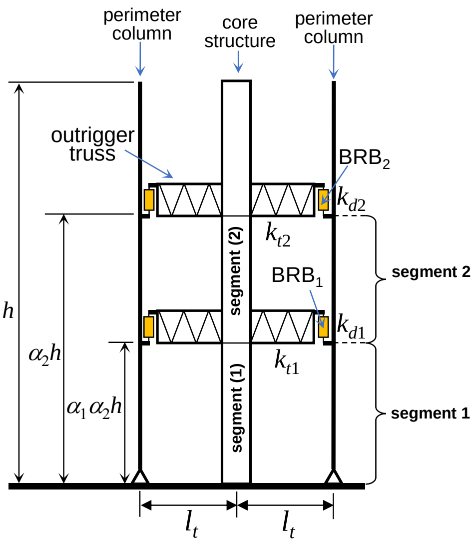


Figure 3.2.5 The simplified structure with dual BRB-outrigger

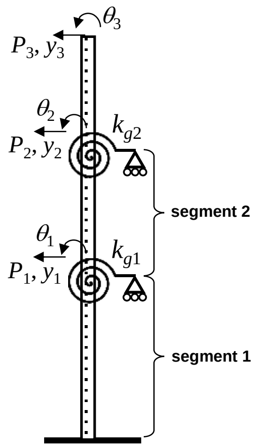


Figure 3.2.6 The simplified structure with two levels of rotational springs

3.2.1 Uniform mass model

The UM model assumes that the mass is uniformly distributed along with the core structure height, and the flexural rigidity of the core structure (*EI*) is kept constant throughout the core structure height. The process of constructing the UM model with *n* levels of BRB-outrigger is first to divide the core structure into *n*+1 segments by the outrigger elevations as shown in Figure 3.2.1. For the j^{th} segment core structure, the lateral displacement, $y_j(x_j, t)$, at a point apart from the j^{th} segment's bottom and with a distance of x_j and at time of t, can be solved by applying the D'Alembert's principle as follows:

$$EI\frac{\partial^4 y_j(x_j,t)}{\partial x_j^4}+m\frac{\partial^2 y_j(x_j,t)}{\partial t^2}=0 \tag{3.2.11}$$

Where m is the mass per unit height. It is assumed that the lateral displacement, y_j, are in the form as follows:

$$y_j(x_j,t)=Y_j(x_j)Q_j(t) \tag{3.2.12}$$

Substitute Equation (3.2.12) into Equation (3.2.11), the following equation is obtained:

$$\frac{EIY_j''''(x_j)}{mY_j(x_j)}=\frac{-\ddot{Q}_j(t)}{Q_j(t)}=\omega^2,\ \text{where}\ Y_j''''(x_j)=\frac{d^4Y_j(x_j)}{dx_j^4},\ \ddot{Q}_j(t)=\frac{d^2Q_j(t)}{dt^2} \tag{3.2.13}$$

The left and right terms in Equation (3.2.13) must be valid for any x_j or t. Therefore, the Equation (3.2.13) should be a constant (ω^2) and the following equation can be derived:

$$Y_j''''\left(x_j\right)-\frac{m\omega^2}{EI}Y_j\left(x_j\right)=0 \tag{3.2.14}$$

The solutions of Equation (3.2.14) can be expressed as follows:

$$\begin{aligned}&Y_j\left(x_j\right)=A_{j1}\cosh\left(\frac{\lambda}{h}x_j\right)+A_{j2}\sinh\left(\frac{\lambda}{h}x_j\right)+A_{j3}\cos\left(\frac{\lambda}{h}x_j\right)+A_{j4}\sin\left(\frac{\lambda}{h}x_j\right)\\&\text{where } \lambda^4=\frac{m\omega^2h^4}{EI}\end{aligned} \tag{3.2.15}$$

Where ω is the angular frequency. By applying the boundary conditions at the ends of the j^{th} segment, the Equation (3.2.15) can be expressed in the matrix as follows:

$$\mathbf{u}_j=\begin{bmatrix}y_{j-1}\\ \theta_{j-1}h\\ y_j\\ \theta_j h\end{bmatrix}=\begin{bmatrix}1&0&1&0\\0&\lambda&0&\lambda\\C_j&S_j&c_j&s_j\\ \lambda S_j&\lambda C_j&-\lambda s_j&\lambda c_j\end{bmatrix}\begin{bmatrix}A_{j1}\\A_{j2}\\A_{j3}\\A_{j4}\end{bmatrix}=\mathbf{D}_j\mathbf{A}_j \tag{3.2.16}$$

Where $\mathbf{u}_j$ is the displacement matrix for the j^{th} segment. As shown in Figure 3.2.2, the y_j and $y_{j\text{-}1}$ are lateral displacements at top and bottom of j^{th} core structure segment, respectively, the θ_j and $\theta_{j\text{-}1}$ are core structure rotations at top and bottom of j^{th} core structure segment, respectively. The parameters C_j, S_j, c_j, and s_j are as follows:

$$C_j=\cosh\left(\frac{\lambda}{h}L_j\right),\ S_j=\sinh\left(\frac{\lambda}{h}L_j\right),\ c_j=\cos\left(\frac{\lambda}{h}L_j\right),\ s_j=\sin\left(\frac{\lambda}{h}L_j\right) \tag{3.2.17}$$

Where L_j is the length of the j^{th} segment and can be calculated as follows:

$$L_j=h\left(1-\alpha_{j-1}\right)\prod_{i=j}^{n}\alpha_i \tag{3.2.18}$$

The coefficients in Equation (3.2.15) are expressed in the matrix, $\mathbf{A}_j$, and can be solved as follows:

$$\mathbf{A}_j=\mathbf{D}_j^{-1}\mathbf{u}_j=\frac{1}{2-2C_jc_j}\begin{bmatrix}1-c_jC_j-s_jS_j&\dfrac{c_jS_j-C_js_j}{\lambda}&C_j-c_j&\dfrac{s_j-S_j}{\lambda}\\ c_jS_j+C_js_j&\dfrac{s_jS_j-c_jC_j+1}{\lambda}&-S_j-s_j&\dfrac{C_j-c_j}{\lambda}\\ 1-C_jc_j+S_js_j&\dfrac{C_js_j-c_jS_j}{\lambda}&c_j-C_j&\dfrac{S_j-s_j}{\lambda}\\ -c_jS_j-C_js_j&\dfrac{1-c_jC_j-s_jS_j}{\lambda}&S_j+s_j&\dfrac{c_j-C_j}{\lambda}\end{bmatrix}\begin{bmatrix}y_{j-1}\\ \theta_{j-1}h\\ y_j\\ \theta_j h\end{bmatrix} \tag{3.2.19}$$

As shown in Figure 3.2.2, the shears (${}_jP_{j-1}$ and ${}_jP_j$) and bending moments (${}_jM_{j-1}$ and ${}_jM_j$) at the bottom and top of the j^{th} core structure segment can be calculated as follows:

$$
\begin{aligned}
{}_jP_{j-1} &= EIY_j'''(0) = \frac{EI}{h^3}\left(A_{j2}\lambda^3 - A_{j4}\lambda^3\right) \\
{}_jP_j &= -EIY_j'''(L_j) = \frac{EI}{h^3}\left[-A_{j1}\lambda^3 S_j - A_{j2}\lambda^3 C_j - A_{j3}\lambda^3 s_j + A_{j4}\lambda^3 c_j\right] \\
\frac{{}_jM_{j-1}}{h} &= -\frac{EIY_j''(0)}{h} = \frac{EI}{h^3}\left(-A_{j1}\lambda^2 + A_{j3}\lambda^2\right) \\
\frac{{}_jM_j}{h} &= \frac{EIY_j''(L_j)}{h} = \frac{EI}{h^3}\left(A_{j1}\lambda^2 C_j + A_{j2}\lambda^2 S_j - A_{j3}\lambda^2 c_j - A_{j4}\lambda^2 s_j\right)
\end{aligned}
\tag{3.2.20}
$$

Substitute the solution of $\mathbf{A}_j$ in Equation (3.2.19) into Equation (3.2.20), the force matrix $\mathbf{F}_j$ for the j^{th} segment can be expressed as follows:

$$
\mathbf{F}_j = \begin{bmatrix} {}_jP_{j-1} \\ {}_jM_{j-1}/h \\ {}_jP_j \\ {}_jM_j/h \end{bmatrix} = \frac{EI}{h^3}\begin{bmatrix} 0 & \lambda^3 & 0 & -\lambda^3 \\ -\lambda^2 & 0 & \lambda^2 & 0 \\ -\lambda^3 S_j & -\lambda^3 C_j & -\lambda^3 s_j & \lambda^3 c_j \\ \lambda^2 C_j & \lambda^2 S_j & -\lambda^2 c_j & -\lambda^2 s_j \end{bmatrix}\begin{bmatrix} A_{j1} \\ A_{j2} \\ A_{j3} \\ A_{j4} \end{bmatrix} =
$$

$$
\begin{bmatrix}
\frac{EI\left(S_j c_j + C_j s_j\right)\lambda^3}{h^3\left(1-C_j c_j\right)} & \frac{EI\lambda^2 S_j s_j}{h^3\left(1-C_j c_j\right)} & \frac{-EI\left(S_j + s_j\right)\lambda^3}{h^3\left(1-C_j c_j\right)} & \frac{EI\left(C_j - c_j\right)\lambda^2}{h^3\left(1-C_j c_j\right)} \\
\frac{EI\lambda^2 S_j s_j}{h^3\left(1-C_j c_j\right)} & \frac{EI\lambda\left(C_j s_j - S_j c_j\right)}{h^3\left(1-C_j c_j\right)} & \frac{EI\left(c_j - C_j\right)\lambda^2}{h^3\left(1-C_j c_j\right)} & \frac{EI\lambda\left(S_j - s_j\right)}{h^3\left(1-C_j c_j\right)} \\
\frac{-EI\left(S_j + s_j\right)\lambda^3}{h^3\left(1-C_j c_j\right)} & \frac{EI\lambda^2\left(c_j - C_j\right)}{h^3\left(1-C_j c_j\right)} & \frac{EI\left(S_j c_j + C_j s_j\right)\lambda^3}{h^3\left(1-C_j c_j\right)} & \frac{-EI\lambda^2 S_j s_j}{h^3\left(1-C_j c_j\right)} \\
\frac{EI\left(C_j - c_j\right)\lambda^2}{h^3\left(1-C_j c_j\right)} & \frac{EI\lambda\left(S_j - s_j\right)}{h^3\left(1-C_j c_j\right)} & \frac{-EI\lambda^2 S_j s_j}{h^3\left(1-C_j c_j\right)} & \frac{EI\lambda\left(C_j s_j - S_j c_j\right)}{h^3\left(1-C_j c_j\right)}
\end{bmatrix}
\begin{bmatrix} y_{j-1} \\ \theta_{j-1}h \\ y_j \\ \theta_j h \end{bmatrix} = \mathbf{B}_j\mathbf{u}_j
\tag{3.2.21}
$$

Considering all n+1 segments and the rotational spring stiffness ($\mathbf{k_g}$) resulting from the n levels of BRB-outrigger, the system matrix $\mathbf{B}$ (2n+2 by 2n+2) that express the relationship between force ($\mathbf{F}$, 2n+2 by 1) and deformation ($\mathbf{u}$, 2n+2 by 1) matrices can be constructed. For a structure with a single BRB-outrigger system, the system matrix $\mathbf{B}_{single}$ with a 4 by 4 dimension and can be expressed as follows:

$$
\mathbf{F} = \begin{bmatrix} P_1 \\ M_1/h \\ P_2 \\ M_2/h \end{bmatrix} = \begin{bmatrix} B_{11} & B_{12} & B_{13} & B_{14} \\ B_{21} & B_{22}+k_{g11} & B_{23} & B_{24} \\ B_{31} & B_{32} & B_{33} & B_{34} \\ B_{41} & B_{42} & B_{43} & B_{44} \end{bmatrix}\begin{bmatrix} y_1 \\ \theta_1 h \\ y_2 \\ \theta_2 h \end{bmatrix} = \mathbf{B}_{single}\mathbf{u}
\tag{3.2.22}
$$

Where

$$B_{11}=\frac{EI\left(S_1c_1+C_1s_1\right)\lambda^3}{h^3\left(1-C_1c_1\right)}+\frac{EI\left(S_2c_2+C_2s_2\right)\lambda^3}{h^3\left(1-C_2c_2\right)} \quad B_{12}=B_{21}=\frac{-EI\lambda^2S_1s_1}{h^3\left(1-C_1c_1\right)}+\frac{EI\lambda^2S_2s_2}{h^3\left(1-C_2c_2\right)}$$

$$B_{13}=B_{31}=\frac{-EI\left(S_2+s_2\right)\lambda^3}{h^3\left(1-C_2c_2\right)} \quad B_{14}=B_{41}=\frac{EI\left(C_2-c_2\right)\lambda^2}{h^3\left(1-C_2c_2\right)}$$

$$B_{22}=\frac{EI\left(C_1s_1-S_1c_1\right)\lambda}{h^3\left(1-C_1c_1\right)}+\frac{EI\left(C_2s_2-S_2c_2\right)\lambda}{h^3\left(1-C_2c_2\right)} \quad B_{23}=B_{32}=\frac{EI\left(c_2-C_2\right)\lambda^2}{h^3\left(1-C_2c_2\right)}$$

$$B_{24}=B_{42}=\frac{EI\left(S_2-s_2\right)\lambda}{h^3\left(1-C_2c_2\right)} \quad B_{33}=\frac{EI\left(S_2c_2+C_2s_2\right)\lambda^3}{h^3\left(1-C_2c_2\right)}$$

$$B_{34}=B_{43}=\frac{-EI\lambda^2S_2s_2}{h^3\left(1-C_2c_2\right)} \quad B_{44}=\frac{EI\left(C_2s_2-S_2c_2\right)\lambda}{h^3\left(1-C_2c_2\right)} \quad k_{g11}=\frac{2l_t^2}{\frac{\alpha}{k_c}+\frac{1}{k_d}+\frac{1}{k_t}}=\frac{2l_t^2}{\frac{1}{k_b}+\frac{1}{k_d}}$$

For a structure with dual BRB-outrigger system, the system matrix $\mathbf{B}_{dual}$ with a 6 by 6 dimension and can be expressed as follows:

$$\mathbf{F}=\begin{bmatrix}P_1\\M_1/h\\P_2\\M_2/h\\P_3\\M_3/h\end{bmatrix}=\begin{bmatrix}B_{11}&B_{12}&B_{13}&B_{14}&0&0\\B_{21}&B_{22}+k_{g11}&B_{23}&B_{24}+k_{g12}&0&k_{g13}\\B_{31}&B_{32}&B_{33}&B_{34}&B_{35}&B_{36}\\B_{41}&B_{42}+k_{g21}&B_{43}&B_{44}+k_{g22}&B_{45}&B_{46}+k_{g23}\\0&0&B_{53}&B_{54}&B_{55}&B_{56}\\0&k_{g31}&B_{63}&B_{64}+k_{g32}&B_{65}&B_{66}+k_{g33}\end{bmatrix}\begin{bmatrix}y_1\\\theta_1h\\y_2\\\theta_2h\\y_3\\\theta_3h\end{bmatrix}=\mathbf{B}_{dual}\mathbf{u} \qquad (3.2.23)$$

Where

$$B_{11}=\frac{EI\left(S_1c_1+C_1s_1\right)\lambda^3}{h^3\left(1-C_1c_1\right)}+\frac{EI\left(S_2c_2+C_2s_2\right)\lambda^3}{h^3\left(1-C_2c_2\right)} \quad B_{12}=B_{21}=\frac{-EI\lambda^2S_1s_1}{h^3\left(1-C_1c_1\right)}+\frac{EI\lambda^2S_2s_2}{h^3\left(1-C_2c_2\right)}$$

$$B_{13}=B_{31}=\frac{-EI\left(S_2+s_2\right)\lambda^3}{h^3\left(1-C_2c_2\right)} \quad B_{14}=B_{41}=\frac{EI\left(C_2-c_2\right)\lambda^2}{h^3\left(1-C_2c_2\right)}$$

$$B_{22}=\frac{EI\lambda\left(C_1s_1-S_1c_1\right)}{h^3\left(1-C_1c_1\right)}+\frac{EI\lambda\left(C_2s_2-S_2c_2\right)}{h^3\left(1-C_2c_2\right)} \quad B_{23}=B_{32}=\frac{EI\left(c_2-C_2\right)\lambda^2}{h^3\left(1-C_2c_2\right)}$$

$$B_{24}=B_{42}=\frac{EI\lambda\left(S_2-s_2\right)}{h^3\left(1-C_2c_2\right)} \quad B_{33}=\frac{EI\left(S_2c_2+C_2s_2\right)\lambda^3}{h^3\left(1-C_2c_2\right)}+\frac{EI\left(S_3c_3+C_3s_3\right)\lambda^3}{h^3\left(1-C_3c_3\right)}$$

$$B_{34}=B_{43}=\frac{-EI\lambda^2S_2s_2}{h^3\left(1-C_2c_2\right)}+\frac{EI\lambda^2S_3s_3}{h^3\left(1-C_3c_3\right)} \quad B_{35}=B_{53}=\frac{-EI\left(S_3+s_3\right)\lambda^3}{h^3\left(1-C_3c_3\right)}$$

$$B_{36}=B_{63}=\frac{EI\left(C_3-c_3\right)\lambda^2}{h^3\left(1-C_3c_3\right)} \quad B_{44}=\frac{EI\lambda\left(C_2s_2-S_2c_2\right)}{h^3\left(1-C_2c_2\right)}+\frac{EI\lambda\left(C_3s_3-S_3c_3\right)}{h^3\left(1-C_3c_3\right)}$$

$$B_{45}=B_{54}=\frac{EI\left(c_3-C_3\right)\lambda^2}{h^3\left(1-C_3c_3\right)} \quad B_{46}=B_{64}=\frac{EI\lambda\left(S_3-s_3\right)}{h^3\left(1-C_3c_3\right)}$$

$$B_{55}=\frac{EI\left(S_3c_3+C_3s_3\right)\lambda^3}{h^3\left(1-C_3c_3\right)} \quad B_{56}=B_{65}=\frac{-EI\lambda^2S_3s_3}{h^3\left(1-C_3c_3\right)}$$

$$B_{66}=\frac{EI\lambda\left(C_3s_3-S_3c_3\right)}{h^3\left(1-C_3c_3\right)}$$

Where k_{gpq} refers to the element in the p^{th} row and q^{th} column in the $\mathbf{k_g}$ calculated from Equation (3.2.10). The λ_r, ω_r, and the associated vibration periods of the r^{th}

mode can be solved by calculating det **B** = 0. The r^{th} mode shape can be obtained by substituting the λ_r into Equation (3.2.19) to obtain $\mathbf{A}_j$ for the j^{th} segment first, then substituting $\mathbf{A}_j$ into Equation (3.2.15) to calculate the deformed shape of the j^{th} segment. After the deformed shapes of 1^{st} to n^{th} segments are computed, the mode shape of the whole structure can be combined together. When the dynamic properties of the system such as the vibration periods and mode shapes are obtained, the modal analysis and the SA can be performed. As the mass is assumed to concentrate on the core structure and distribute throughout the building height uniformly, the abovementioned analytical model is known as the uniform mass (UM) model. The features of using UM model are as follows:

(1) The matrix calculations can be easily implemented in software such as Matlab or Microsoft Excel. The calculation of dynamic properties does not require using structure analysis software.

(2) The calculation efficiency is high.

(3) The calculation is valid for linearly elastic force and deformation relationship. When the system deforms nonlinearly, the system matrix **B** is required to be modified. This process significantly increases the difficulty and complicacy of using UM model in performing the NLRHA.

(4) Furthermore, when *EI*, mass, and perimeter column cross-sectional property are not constant throughout the building height. The complicacy of UM model would be significantly increased.

3.2.2 Discrete mass model

In order to verify the effectiveness of using UM model, to perform the NLRHA using OpenSees, and to analyze the structure when *EI* and perimeter column's cross-sectional property vary with elevation (Chapter 10), the discrete mass (DM) model is proposed. The concept of DM model is to construct the simplified structure shown in Figure 3.2.1 by using OpenSees. Figure 3.2.7 shows the DM model of a dual BRB-outrigger system. The core structure is modeled by a cantilever column (beam column element), and each of the outrigger trusses is modeled by a cantilever beam (beam column element) with flexural stiffness of k_t (single BRB-outrigger system) or k_{t1} and k_{t2} (dual BRB-outrigger system). The perimeter column bases are free to rotate about the out-of-plane direction. As shown in Figure 3.2.7, the perimeter column segment 1 range from the base (Point O) to Point A, and the perimeter column segment 2 ranges from Point A to Point C. Point B connects the lower BRB with outrigger truss end and do not interact with perimeter column. The mass is distributed evenly along with the core structure height with a fixed spacing of 1m. The BRBs are modeled by using truss elements, while the others are modeled using beam column elements. The material models are bilinear and have a post-yield stiffness ratio of 0.01 for the BRB elements, and are linearly elastic for the other members. In order to perform the parametric study, the length of each BRB is fixed at 1 m. If k_d and $u_{d,y}$ are the axial stiffness and yield deformation of the BRB, the cross-sectional area A_{DM} and the material yield stress $\sigma_{y,DM}$ assigned for the BRB element material in the DM model can be calculated as follows:

$$A_{DM} = \frac{k_d \times 1m}{E} \tag{3.2.24}$$

$$\sigma_{y,DM} = \frac{u_{d,y} k_d}{A_{DM}} \tag{3.2.25}$$

Where E is the modulus of elasticity (200 GPa) of steel material. If compare with the UM model, the DM model can be used for performing modal analysis, modal pushover analysis (MPA), SA, and NLRHA. However, the analysis of SA would be less efficient because of the large amount of degree of freedom in the DM model.

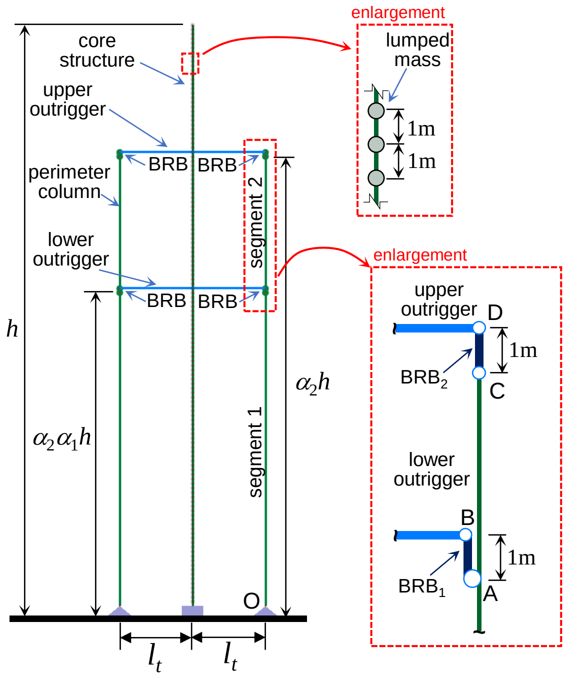


Figure 3.2.7 The DM model

3.3 PARAMETER DEFINITIONS

In order to investigate the optimal design of the BRB-outrigger system, dimensionless parameters are used for the parameter study. Two sets of parameters are used. The first sets are the outrigger stiffness parameters, which are used to indicate the magnitudes of the outrigger effect on the structure. The second sets are BRB stiffness parameters which are used to describe the stiffness relationships between the BRB, perimeter column, and the outrigger truss.

3.3.1 Parameters for single BRB-outrigger system

The outrigger stiffness parameter (S_{bc}) is defined as the ratio of rotational stiffness provided by the single BRB-outrigger, when the BRB axial stiffness (k_d) is infinity to the rotational stiffness of the core structure and is expressed as follows:

$$S_{bc} = \frac{k_b l_t^2}{EI/h} = \frac{l_t^2 h}{EI\left(1/k_t + \alpha/k_c\right)} \tag{3.3.1}$$

Where k_t and k_c are the flexural stiffness of the outrigger truss and the axial stiffness of the perimeter column with a length of h, respectively. k_b is the combined effective stiffness of the outrigger truss and the perimeter column below the outrigger truss. A larger S_{bc} value indicates a more significant outrigger effect. A longer outrigger truss span (l_t) while k_b remains constant, or a stiffer outrigger truss (greater k_t), or a stiffer perimeter column (greater k_c) can enhance the outrigger effect. However, when EI, h, l_t, k_t, and k_c are kept constant while increasing outrigger elevation (increasing α), the outrigger effect would be smaller. When the building height (h) increases, the value of EI/h would be larger as the increase of EI would be greater than h due to higher seismic lateral force demands. Therefore, the S_{bc} value would be smaller for a taller building with a single BRB-outrigger.

The BRB stiffness parameters (R_{dt}, R_{dc}, and R_{db}) are defined as follows:

$$R_{dt} = \frac{k_d}{k_t} \tag{3.3.2}$$

$$R_{dc} = \frac{k_d}{k_c} \tag{3.3.3}$$

$$R_{db} = \frac{k_d}{k_b} = k_d\left(\frac{1}{k_t} + \frac{\alpha}{k_c}\right) = R_{dt} + \alpha R_{dc} \tag{3.3.4}$$

The Equation (3.3.1) can be express in terms of BRB stiffness parameters as follows:

$$S_{bc} = \frac{l_t^2 h}{EIR_{db}} k_d = \frac{l_t^2 h}{EI\left(R_{dt} + \alpha R_{dc}\right)} k_d \tag{3.3.5}$$

In the design practices, the perimeter column sizes are usually determined according to gravity load demands. Therefore, it is possible that the BRB has to be designed after the perimeter column sizes are determined. The R_{dc} describes the stiffness relationship between BRB and the perimeter column, and R_{db} describes the relationship between BRB and the combination of perimeter column and outrigger truss. The larger R_{db} or R_{dc} value indicates the BRB is stiffer. A stiffer BRB may result in higher efficiency in mitigating seismic response, however, it could also amplify seismic demand due to shorter vibration period, and the cost of BRB would increase. Therefore, the value of optimal R_{db} or R_{dc} in order to minimize seismic response could provide an easy and straightforward method for structural engineers to roughly design the BRB in the preliminary design stage. In addition, a stiffer outrigger truss could provide a greater BRB axial deformation demand, so that the BRB could develop a good hysteretic response. Therefore, a smaller R_{dt} value should be preferred as it results in a larger BRB axial deformation demand. However, when l_t is very long, to design a very stiff outrigger truss becomes uneconomical as a very large member size of the outrigger truss would be required in order to provide the wanted k_t. Alternative solutions with different BRB-outrigger configurations are introduced in Chapter 9. For the parametric study, the model shown in Figure 3.2.1 is used.

3.3.2 Parameters for dual BRB-outrigger system

For the structure with dual BRB-outrigger system, the two sets of parameters similar to the single BRB-outrigger system are used. As the outrigger number increases, the parameters are added with a subscript of 1 or 2 to represent the lower or upper outrigger, respectively. The outrigger stiffness parameters S_{bc1} and S_{bc2} are defined as the ratio of rotational stiffness provided by the lower and upper BRB-outrigger, respectively, when the BRB axial stiffness is infinity, to the rotational stiffness of the core structure, and can be calculated as follows:

$$S_{bc1} = \left(\frac{l_t^2}{1/k_{t1} + \alpha_1 \alpha_2 / k_c}\right) \Big/ \left(\frac{EI}{h}\right) \tag{3.3.6}$$

$$S_{bc2} = \left(\frac{l_t^2}{1/k_{t2} + \alpha_2/k_c} \right) \Bigg/ \left(\frac{EI}{h} \right) \tag{3.3.7}$$

Where k_{t1} and k_{t2} are the outrigger truss flexural stiffness of the lower and upper outriggers, respectively. k_c is the perimeter column axial stiffness with a length of h. The values of S_{bc1} and S_{bc2} are used to indicate the magnitude of the effects of the lower and upper outriggers on structure, respectively. The greater the values of S_{bc1} and S_{bc2} suggest the outrigger effect is greater. The outrigger effect can be enhanced by a longer outrigger truss span (l_t), stiffer outrigger trusses (greater k_{t1} or k_{t2}), and stiffer perimeter columns (greater k_c). For very tall buildings, the value of EI/h could significantly increase because of the higher seismic demand. Therefore, the S_{bc1} and S_{bc2} values are set to be smaller when a building is taller. In addition to S_{bc1} and S_{bc2} parameters, the BRB stiffness parameters (R_{d2c} and R_{kd}) are defined as follows:

$$R_{d2c} = \frac{k_{d2}}{k_c} \tag{3.3.8}$$

$$R_{kd} = \frac{k_{d1}}{k_{d2}} \tag{3.3.9}$$

Where k_{d1} and k_{d2} are the axial stiffness of BRB in the lower (BRB_1) and upper (BRB_2) outriggers, respectively. In the design practices, the perimeter column sizes are primarily designed according to the gravity load requirements. Therefore, the parameters of R_{d2c} and R_{kd} can provide engineers with a simple estimation of the required BRB sizes. The parameter R_{kd} is defined as the ratio of the axial stiffness of BRB_1 (k_{d1}) to the axial stiffness of BRB_2 (k_{d2}). When R_{kd} is greater than 1.0, the BRB_1 is stiffer than the BRB_2, and vice versa. If R_{kd}=0, it is a single BRB-outrigger system. The parameters R_{dt1} and R_{dt2} are used to describe the ratios of k_{d1} to k_{t1} and k_{d2} to k_{t2}, respectively, as follows:

$$R_{dt1} = \frac{k_{d1}}{k_{t1}} \tag{3.3.10}$$

$$R_{dt2} = \frac{k_{d2}}{k_{t2}} \tag{3.3.11}$$

In order to generate sufficient deformation demands on the BRB_1 and BRB_2, both R_{dt1} and R_{dt2} should be as small as possible. The outrigger stiffness parameters can be expressed by the BRB stiffness parameters as follows:

$$S_{bc1} = \frac{l_t^2 R_{kd} h}{EI\left(R_{dt1} + \alpha_1 \alpha_2 R_{kd} R_{d2c}\right)} k_{d2} \tag{3.3.12}$$

$$S_{bc2} = \frac{l_t^2 h}{EI\left(R_{dt2} + \alpha_2 R_{d2c}\right)} k_{d2} \tag{3.3.13}$$

3.4 ANALYSIS PROCEDURE

The models used in this study have heights of 64 m (16-story), 128 m (32-story), 256 m (64-story), and 384 m (96-story). Figure 3.4.1 shows the structural framing plan of the BRB-outrigger level for the analytical models. The two elevations of BRB-outriggers are adopted as the lateral force resisting system in the EW direction. For simplicity, only one outrigger elevation and half of the core structure are considered in the analytical model. Each story is 4 m high, and the dead load (which is also the mass source) and live load are 0.8 tonf/m^2 and 0.3 tonf/m^2, respectively. Therefore, the mass is approximate 900 ton for each story. In the UM model, the mass is 225 ton/m that uniformly distributes along with the core structure. In the DM model, the mass is 225 ton assigned on each of the nodes that evenly distribute throughout the core structure height with a spacing of 1 m (Figure 3.2.7). Table 3.4.1 and Figure 3.4.2 shows the details of the analytical models. The magnitude of outrigger effect is indicated by the S_{bc} value when α is 0.7 (S_{bc07}) if single BRB-outrigger is used, and by S_{bc2} value when α_2 is 0.7 ($S_{bc2,07}$) if dual BRB-outrigger is used. The S_{bc07} or $S_{bc2,07}$ are set to be constant for the analytical models with the same building height (h), and are set smaller for taller building (Huang and Takeuchi, 2017). The values of EI are selected so that the fundamental vibration period of the core structure is within a realistic range (for example 0.03h). The 96-story core structure's fundamental period is set to be smaller than 10 sec because the structure with a fundamentalperiod longer than 10 sec would be significantly affected by wind load.

Table 3.4.1 Parameters of analytical models for BRB-outrigger system

model	h (m)	l_t (m)	EI (kN-m^2)	fundamental period of core structure (sec)	S_{bc07} or $S_{bc2,07}$
16-story	64	16	4.1×10^9	1.74	3.03
32-story	128	16	1.6×10^{10}	3.50	1.38
64-story	256	16	6.5×10^{10}	6.92	0.66
96-story	384	16	2.2×10^{11}	9.76	0.30

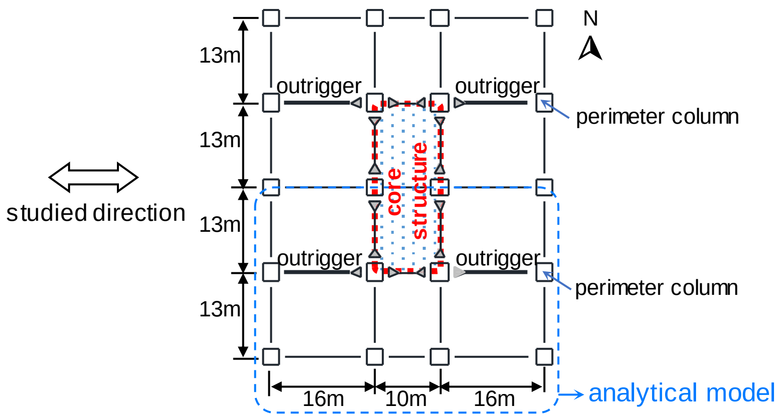


Figure 3.4.1 Structural framing plan of BRB-outrigger layer for numerical models

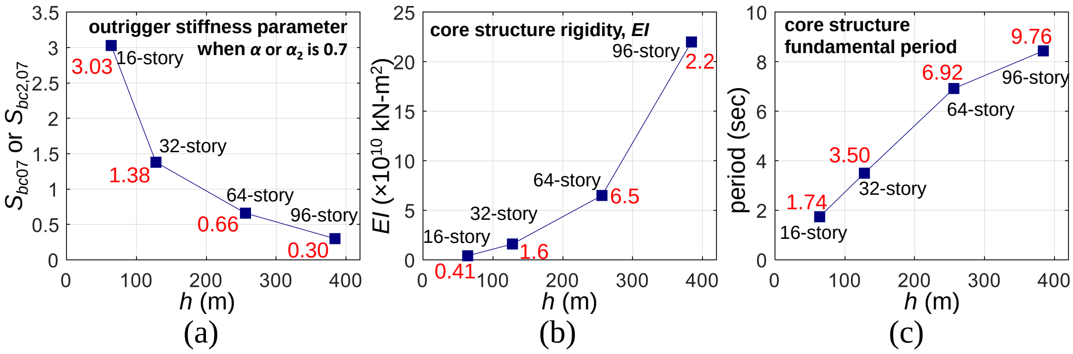


Figure 3.4.2 The relationships between (a) S_{bc07} or $S_{bc2,07}$, (b) EI, (c) fundamental period of core structure and building height of the analytical models

Figure 3.4.3 shows the flow chart for constructing analytical models for the purpose of the parametric study. The detail is illustrated as follows:

(1) Select the BRB-outrigger configuration (single or dual BRB-outrigger system) and the building height h (64 m, 128 m, 256 m, or 384 m), so that the outrigger stiffness parameter (S_{bc07} for the single BRB-outrigger system, or $S_{bc2,07}$ for dual BRB-outrigger system) and the core structure flexural rigidity (EI) are determined from Table 3.4.1.

(2a) For the structure with single BRB-outrigger system, two analysis methods are introduced. In the first method (Method I or Met. I), for each analysis set, the models are constructed by a given R_{db} value and then vary α from 0 to 1. The k_d can be calculated from the given R_{db} by using the Equation (3.3.5) with the S_{bc} value when α = 0.7 (S_{bc07}) given in Table 3.4.1. As k_b and k_d are fixed in each analysis set, varying α from 0 to 1 would not affect the rotational stiffness provided the BRB-outrigger (k_g) as indicated in Equation (3.2.7). The k_b (=$1/(1/k_t+\alpha/k_c)$) is kept constant by fixing both k_t and k_c/α. Thus, k_c is proportional to α when α varies from 0 to 1. This suggests that k_c is larger when α is larger in Met. I. This method provides a straightforward analysis procedure as the k_g is independent of α in each analysis set. However, Met. I is less practical as the perimeter column would not change with α in the design practices.

(2b) For the single BRB-outrigger system, in the second method (Method II or Met. II), the models are constructed by given R_{dc} and R_{dt} values and then varying α from 0 to 1. The k_d can be calculated from the given R_{dc} and R_{dt} by using the Equation (3.3.5) with the S_{bc} value when α = 0.7 (S_{bc07}) given in Table 3.4.1. As k_c, k_d, and k_t are fixed, the rotational stiffness provided by the BRB-outrigger (k_g) is smaller when α is higher as indicated in Equation (3.2.7). The Met. II is more realistic for practical design purpose since k_c and α are independent of each other.

(2c) For the dual BRB-outrigger system, the parameters R_{dc2}, R_{kd}, R_{dt1}, and R_{dt2} are given. The k_{d2} can be calculated by using Equation (3.3.13) with the S_{bc2} value when α_2 = 0.7 ($S_{bc2,07}$) shown in Table 3.4.1. The parameters such as k_{d1}, k_{t1}, k_{t2}, and k_c can be calculated from Equation (3.3.8) to Equation (3.3.13).

(3) For the single BRB-outrigger system, in each analysis set, the outrigger elevation α varies from 0 to 1. For the dual BRB-outrigger system, in each analysis set, both the lower (α_1) and upper (α_2) outrigger elevations vary from 0 to 1.

(4) For each outrigger elevation (α, α_1, α_2), construct the corresponding UM and DM models.

(5) Perform the analyses on each analytical model with the selected outrigger elevations (α, α_1, α_2). The detail of each analysis is explained in Chapter 4. The UM model is used to perform modal analysis and SA, and the DM model is used to perform modal analysis, MPA, SA, and NLRHA.

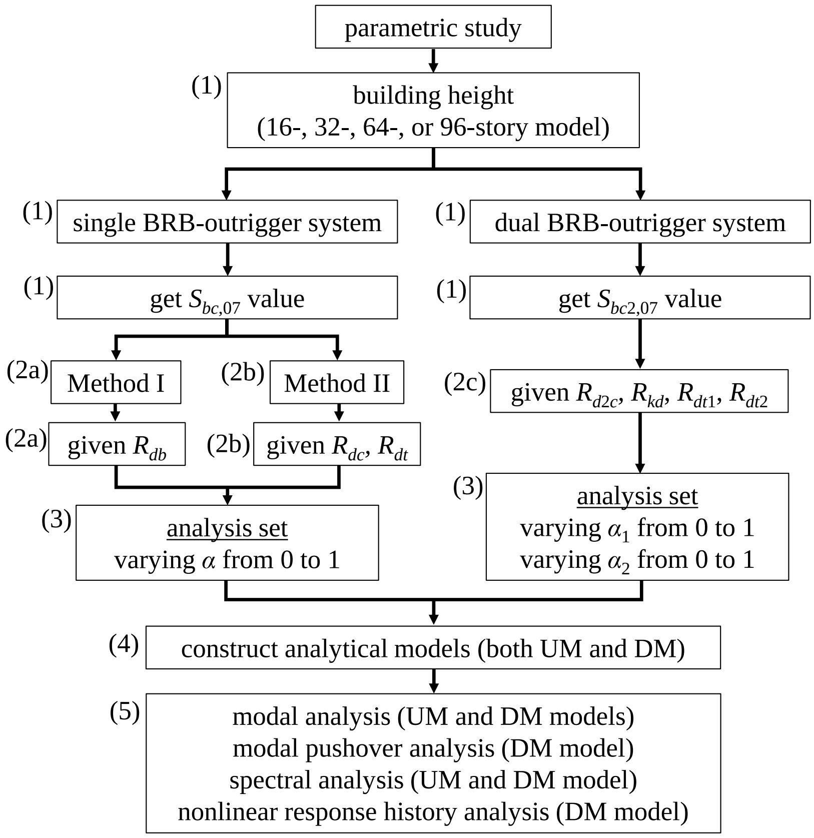


Figure 3.4.3 Flow chart of constructing analytical model

For the single BRB-outrigger models, two different methods (Met. I and Met. II) are used for constructing the analytical models. The Met. I fixes the R_{db} as constant (fixes both k_d and k_b as constants) while varying α from 0 to 1. Therefore, the outrigger stiffness parameter (S_{bc}) and rotational stiffness provided by BRB-outrigger (k_g) are also constants as indicated by Equation (3.3.5) and Equation (3.2.7), respectively. The Met. II fixed R_{dc} and R_{dt} as constants (fixes k_c, k_d, and k_t) while varying α from 0 to 1. Therefore, the outrigger stiffness parameter (S_{bc}) and the rotational stiffness provided by BRB-outrigger (k_g) are smaller when outrigger elevation (α) is higher as indicated by Equation (3.3.5) and Equation (3.2.7), respectively. The parameter variations while α increases from 0 to 1 in each analysis set in Met. I and Met. II are shown in Table 3.4.2. Figure 3.4.4 shows the relationships between S_{bc} and α for the 16-, 32-, 64-, and 96-story models with Met. I and Met. II. Figure 3.4.5 to Figure 3.4.8 show the relationships between k_g and α for the 16-, 32-, 64-, and 96-story models, respectively. When α is approximately smaller than 0.5, both S_{bc} and k_g excessively increase when R_{dc} is approximately greater than 1. However, when the α is higher than 0.5, both Met. I and Met. II provide similar ranges of S_{bc} and k_g.

Table 3.4.2 Parameter variations while α increases from 0 to 1 in each analysis set in Met. I and Met. II

Method	S_{bc}	k_g	k_b	k_t	k_c	k_d
I	fixed	fixed	fixed	fixed	increased	fixed
II	decreased	decreased	decreased	fixed	fixed	fixed

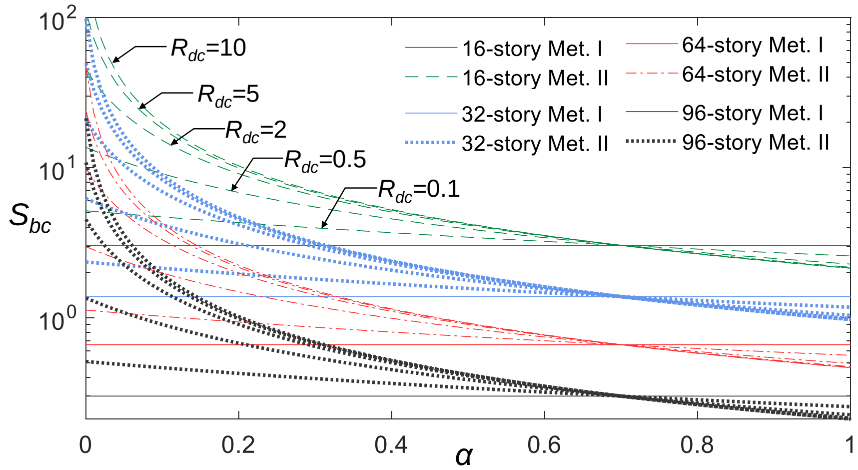


Figure 3.4.4 Relationships between S_{bc} and α for models using Met. I and Met. II

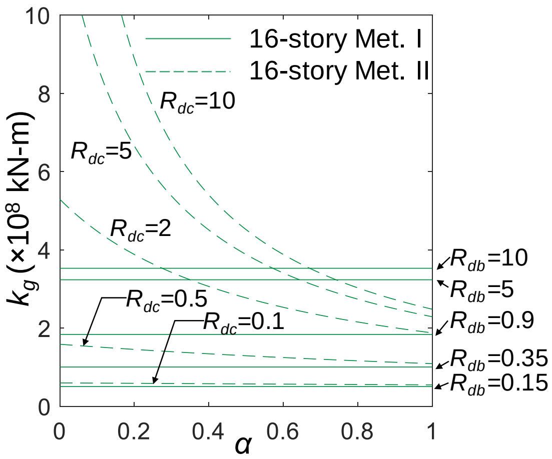


Figure 3.4.5 Relationships between k_g and α of 16-story model

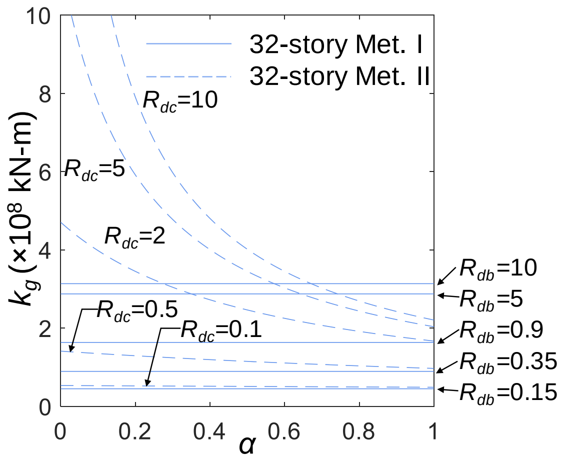


Figure 3.4.6 Relationships between k_g and α of 32-story model

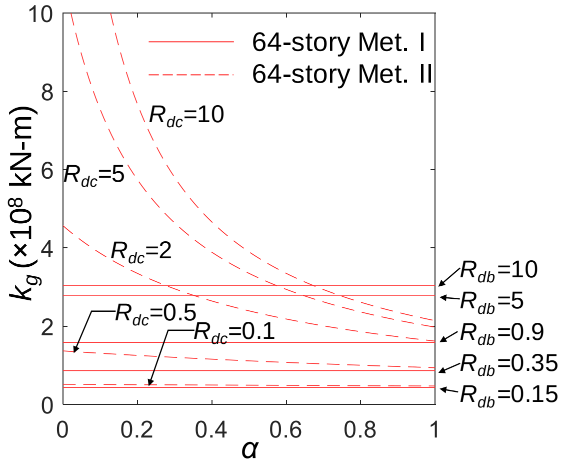


Figure 3.4.7 Relationships between k_g and α of 64-story model

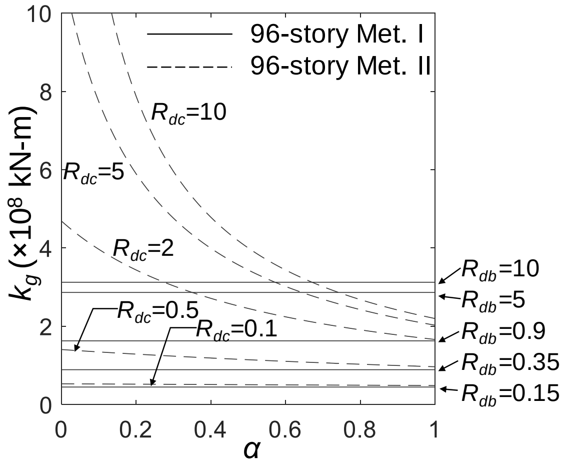


Figure 3.4.8 Relationships between k_g and α of 96-story model

For the dual BRB-outrigger system, the method in constructing analytical models is similar to the Met. II in single BRB-outrigger system. In each analysis set, when α_1 and α_2 vary from 0 to 1, the k_c, k_{d1}, k_{d2}, k_{t1}, and k_{t2} are computed from the given R_{d2c}, R_{kd}, R_{dt1}, and R_{dt2} values. Therefore, the S_{bc2}, S_{bc1}, k_{g2}, and k_{g1} vary with α_1 and α_2. Figure 3.4.9 to Figure 3.4.12 show the relationships between S_{bc2}, S_{bc1} and outrigger elevations (α_1 and α_2) when R_{d2c} varies between 0.1, 1, and 3 and R_{kd} varies between 0.5, 1, and 3 for the 16-, 32-, 64-, and 96-story models, respectively. The S_{bc2} changes with α_2 only. When α_2 is close to the core structure base, the S_{bc2} excessively increase because of the segment of the perimeter column below outrigger becomes shorter and thus provides greater stiffness. The value of S_{bc2} when α_2=0.7 ($S_{bc2,07}$) is fixed as shown in Table 3.4.1 and becomes smaller when α_2 is greater than 0.7, and becomes larger when α_2 is smaller than 0.7. The larger R_{d2c} value increases the variation of S_{bc2} distribution. In addition, the larger R_{kd} value introduces a relative stiffer BRB in the lower outrigger (BRB_1), and thus increase the value of S_{bc1}. When both α_1 and α_2 are small, the value of S_{bc1} excessively increases because of the segment of the perimeter column below outrigger elevation is shorter and provides greater stiffness. The rotational stiffness provided by BRB-outriggers $\mathbf{k_g}$ is in a dimension of 2 × 2 as indicated by Equation (3.2.2). The $\mathbf{k_g}$ is a symmetric matrix, and its component of k_{g22}, k_{g11}, and k_{g12} of the analytical models are shown from Figure 3.4.13 to Figure 3.4.16, from Figure 3.4.17 to Figure 3.4.20, and from Figure 3.4.21 to Figure 3.4.24, respectively. The k_{g22} mainly varies with α_2, and is larger when α_2 is smaller. This is because the segment of the perimeter column below the outrigger is shorter when α_2

is smaller and thus provides greater stiffness. k_{g22} increases when the lower outrigger is close to the upper outrigger (when α_1 is close to 1). When α_1=1, it is a single BRB-outrigger with a very stiff BRB-outrigger. The larger R_{d2c} results in stiffer BRB_2 and creates greater rotational stiffness, k_{g22}. As the BRB axial stiffness of BRB_1 is calculated based on k_{d2} and R_{kd}, the difference of R_{kd} would not affect the value of k_{g22}. The k_{g11} varies with α_1 and α_2. The smaller α_1 and α_2 result in greater k_{g11} because of the shorter perimeter column below outrigger provides greater axial stiffness. The larger R_{kd} value increases the BRB_1 axial stiffness, thus also increase the rotational stiffness provided by the lower BRB-outrigger. The larger R_{d2c} and R_{kd} values increase the differences between k_{d1} and k_{d2}, and therefore increase the ranges of k_{g11} distributions with respect to α_1 and α_2. The k_{g12} describes the relationships between the two outriggers. The larger R_{kd} value results in a stiffer BRB_1 and thus increases the amount of k_{g12}. If compare the dual BRB-outrigger system to the single BRB-outrigger system (keep the upper BRB-outrigger only), the negative value of k_{g12} suggests that when the core structure rotations at α_2 (θ_2 and θ) are the same and the core structure rotation at α_1 (θ_1) and α_2 (θ_2) elevations deform in the same direction in the dual BRB-outrigger system, the single BRB-outrigger applies larger resisting moment than the upper BRB-outrigger in dual BRB-outrigger system. However, if θ_1 and θ_2 deform in opposite directions, the moment applied by the upper BRB-outrigger in dual BRB-outrigger system is larger than the moment applied by the single BRB-outrigger only. This implies that the multiple BRB-outrigger systems would be a benefit in reducing seismic response if the building deforms in the mode shapes higher than the 2nd mode. However, if the 1st mode deformation dominates the overall responses, the single BRB-outrigger system would be more efficient than dual BRB-outrigger system as the resisting moments applied by individual BRB-outrigger could be reduced by each other in the multiple BRB-outrigger systems. As shown in Figure 3.4.21 to Figure 3.4.24, when α_1 is close to 1, the amount of k_{g12} value is larger. The effect from the lower BRB-outrigger on the upper BRB-outrigger becomes greater. Nevertheless, instead of the concentration resisting moment applied by single BRB-outrigger, the multiple BRB-outrigger systems could apply more uniform resisting moment distributions on the core structure, which would be more efficient in reducing inter-story drift response.

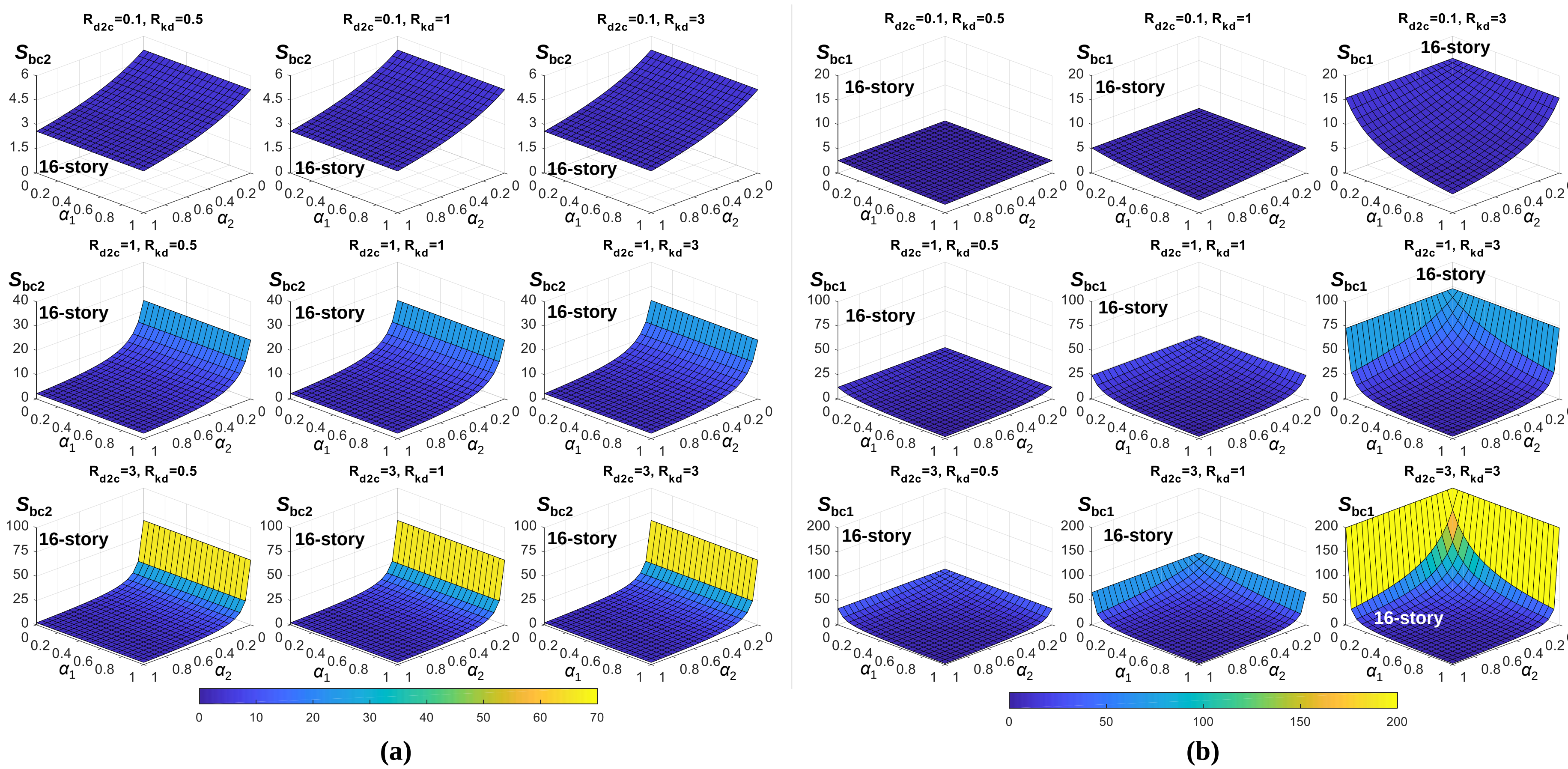


Figure 3.4.9 Relationships between (a) S_{bc2}, (b) S_{bc1} and outrigger elevations for the 16-story dual BRB-outrigger model

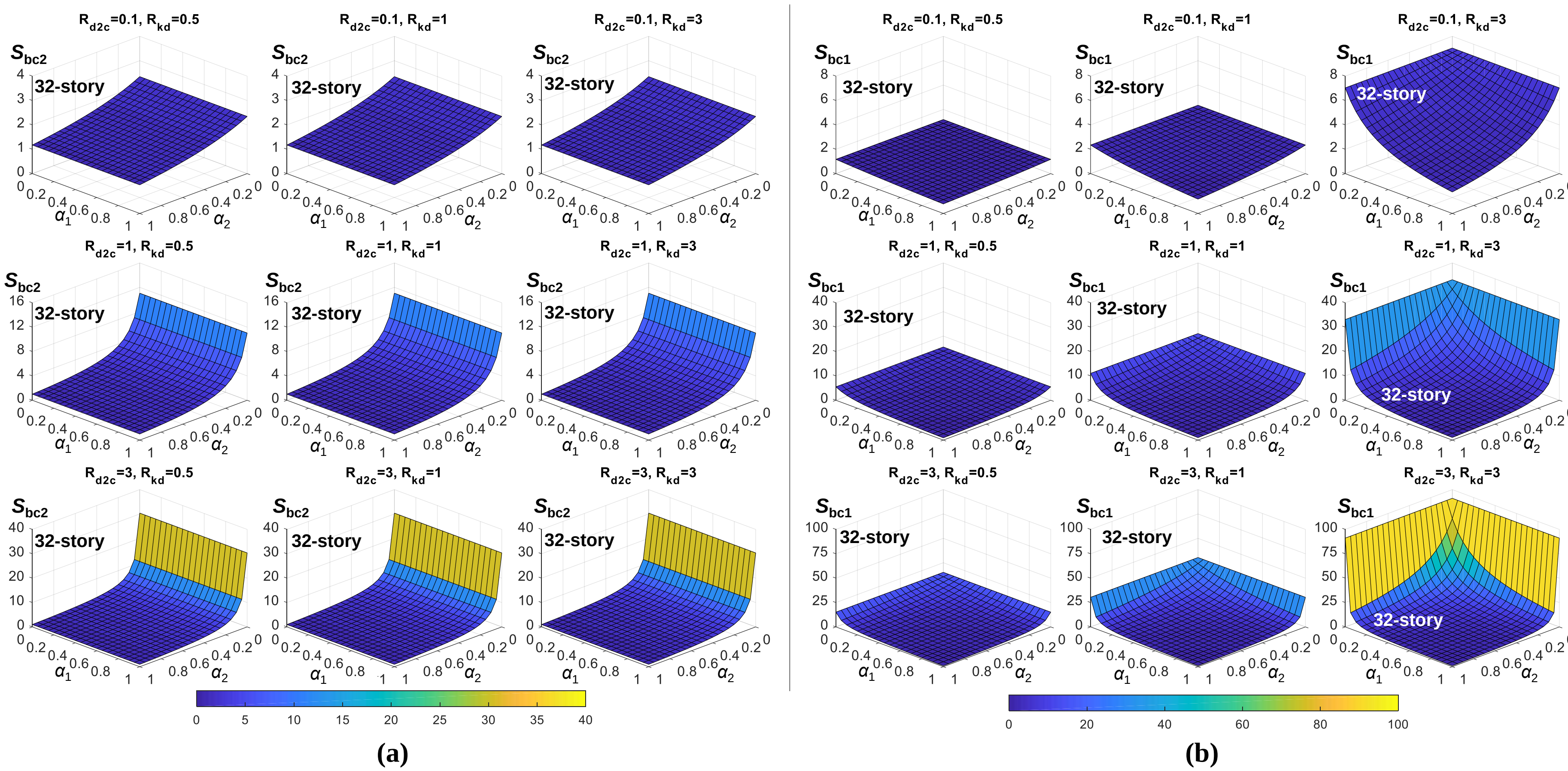


Figure 3.4.10 Relationships between (a) S_{bc2}, (b) S_{bc1} and outrigger elevations for the 32-story dual BRB-outrigger model

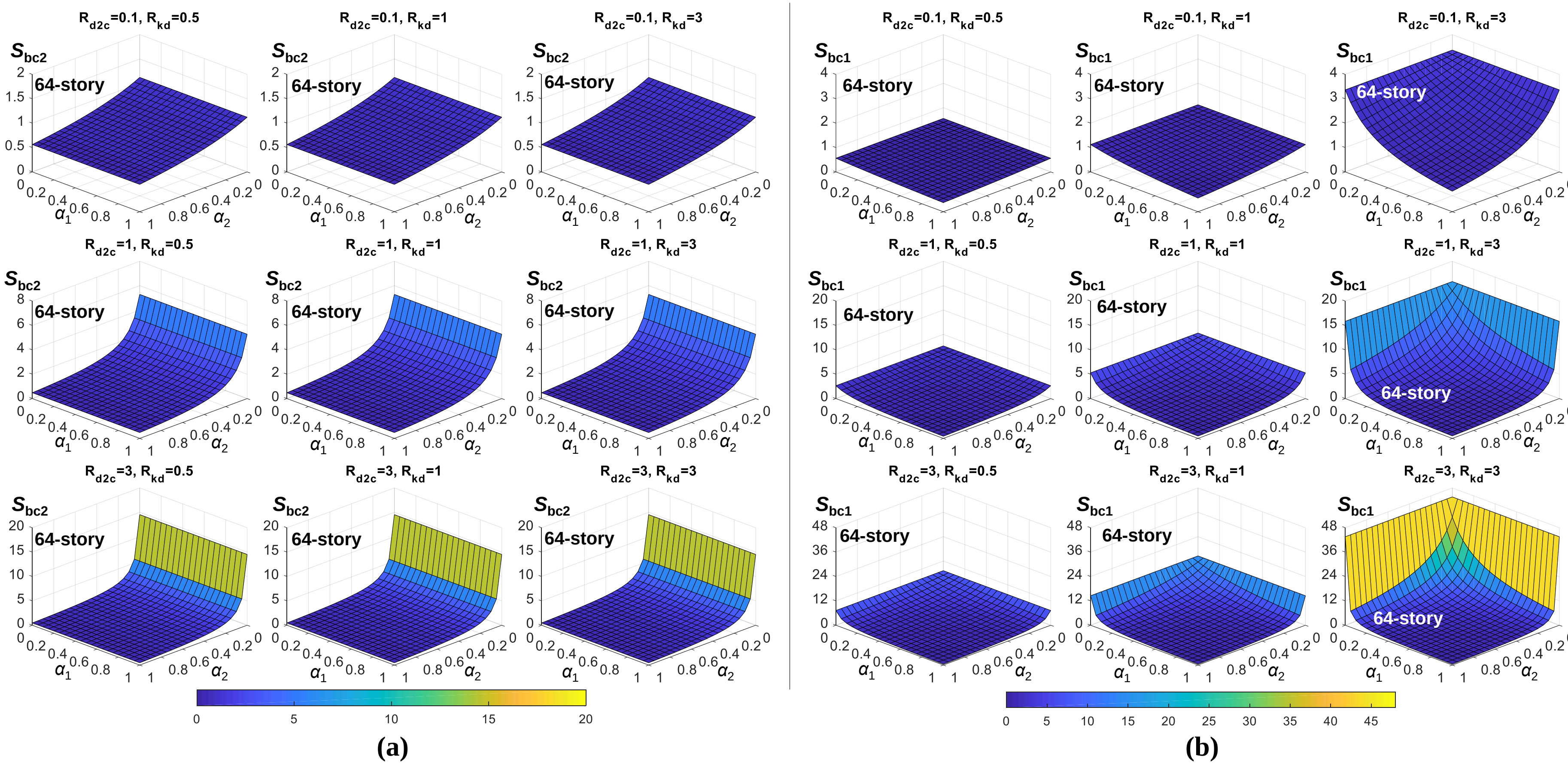


Figure 3.4.11 Relationships between (a) S_{bc2}, (b) S_{bc1} and outrigger elevations for the 64-story dual BRB-outrigger model

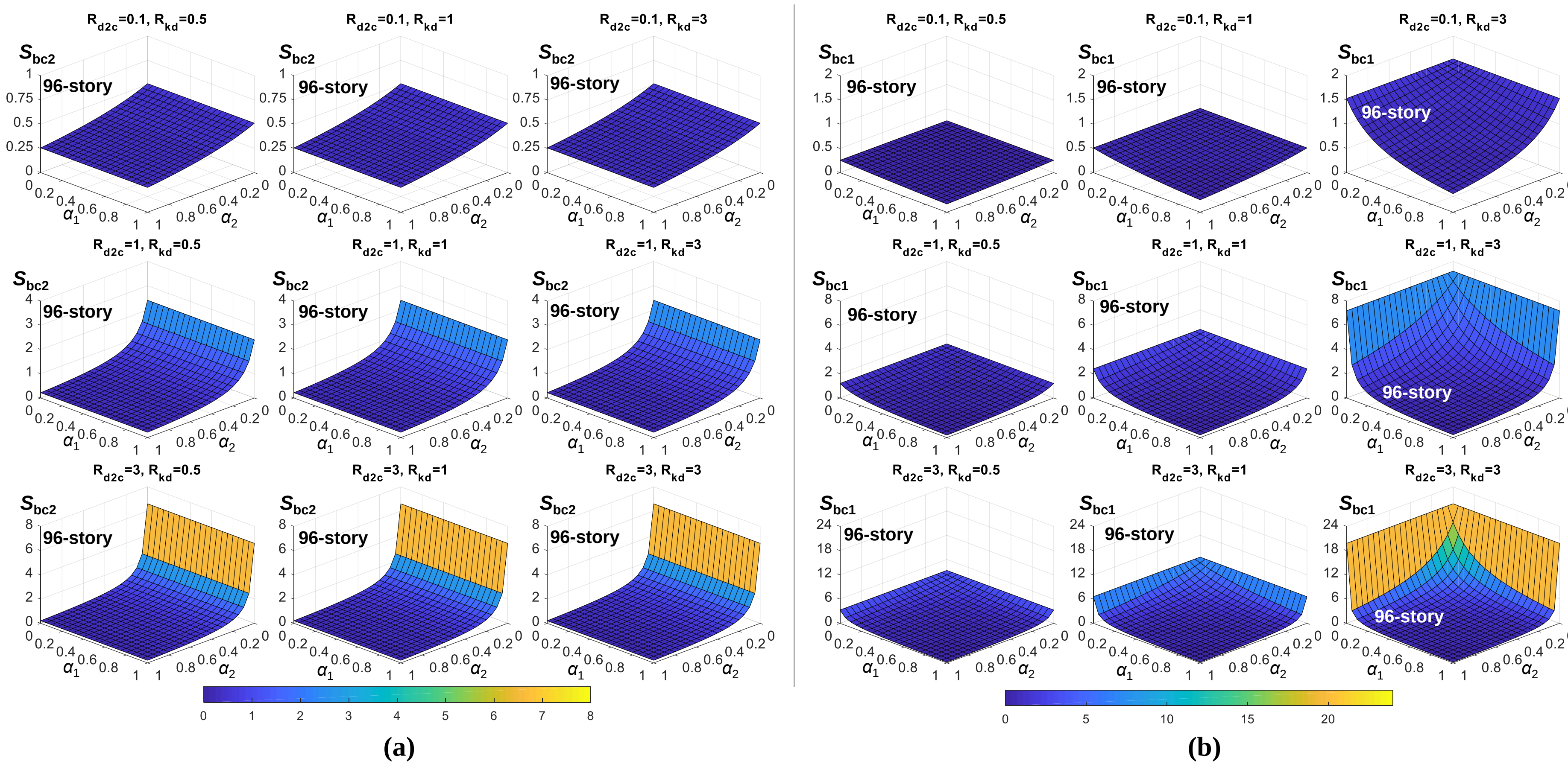


Figure 3.4.12 Relationships between (a) S_{bc2}, (b) S_{bc1} and outrigger elevations for the 96-story dual BRB-outrigger model

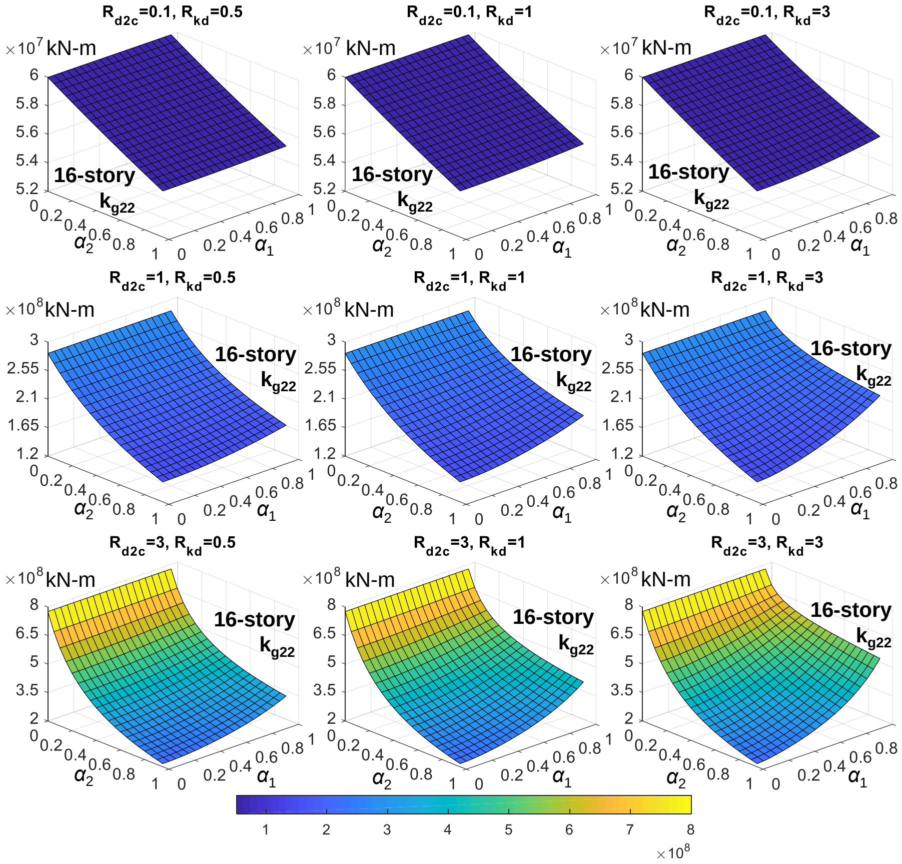


Figure 3.4.13 Relationships between k_{g22} and outrigger elevations for the 16-story dual BRB-outrigger model

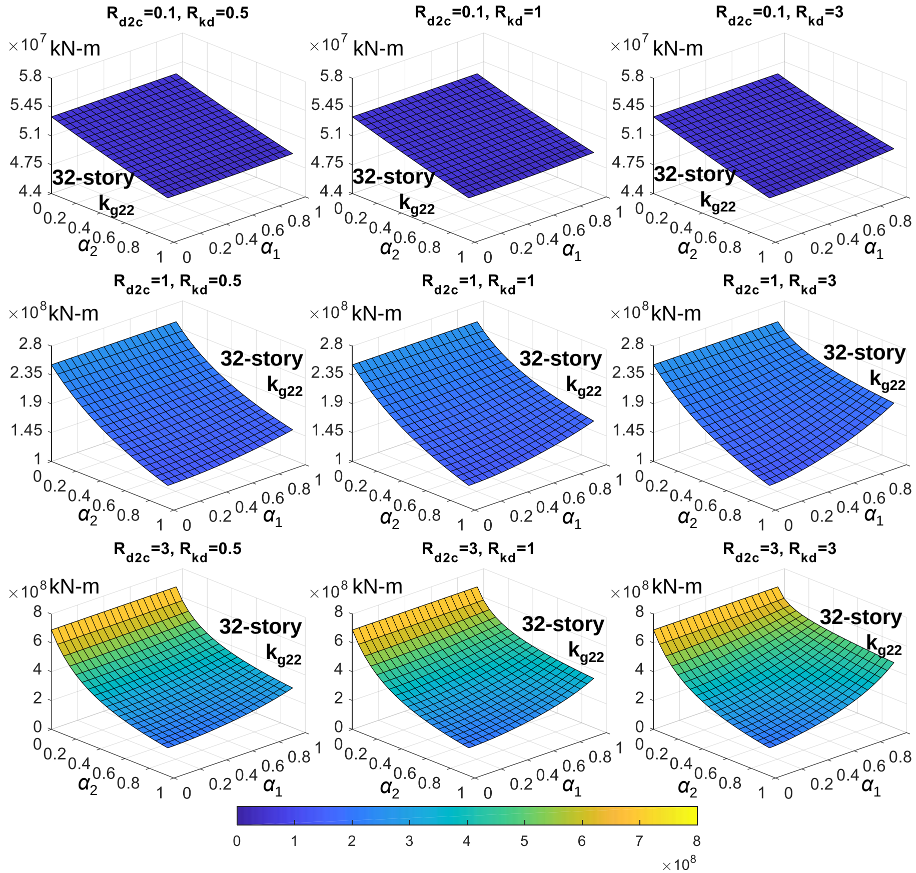


Figure 3.4.14 Relationships between k_{g22} and outrigger elevations for the 32-story dual BRB-outrigger model

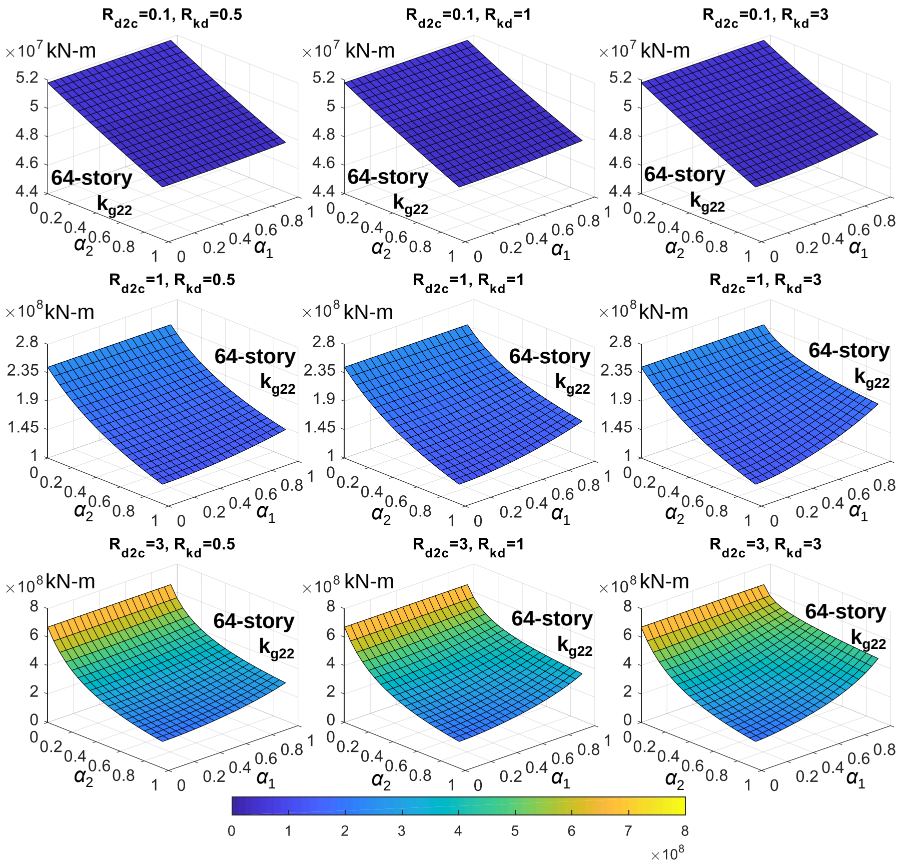


Figure 3.4.15 Relationships between k_{g22} and outrigger elevations for the 64-story dual BRB-outrigger model

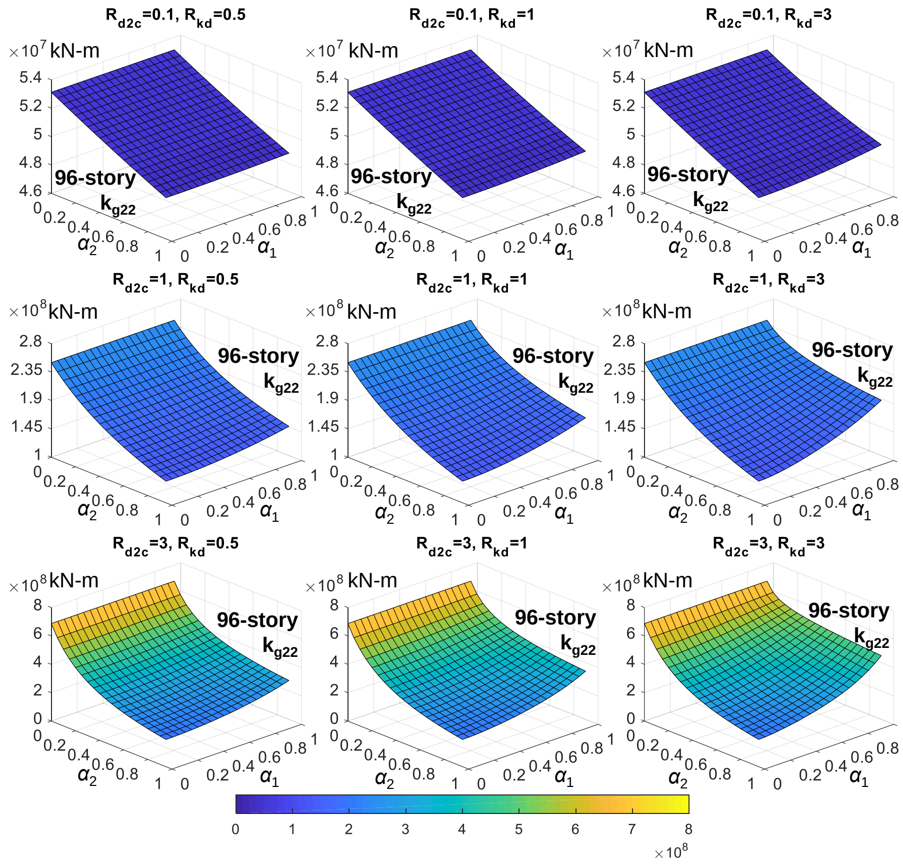


Figure 3.4.16 Relationships between k_{g22} and outrigger elevations for the 96-story dual BRB-outrigger model

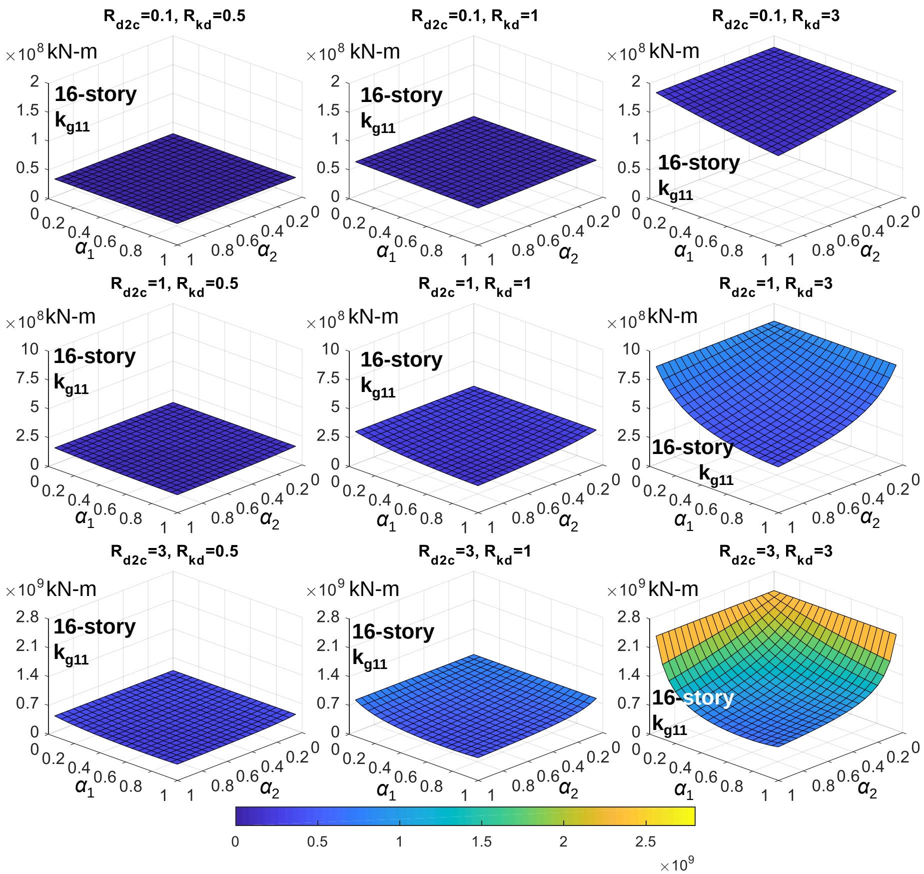


Figure 3.4.17 Relationships between k_{g11} and outrigger elevations for the 16-story dual BRB-outrigger model

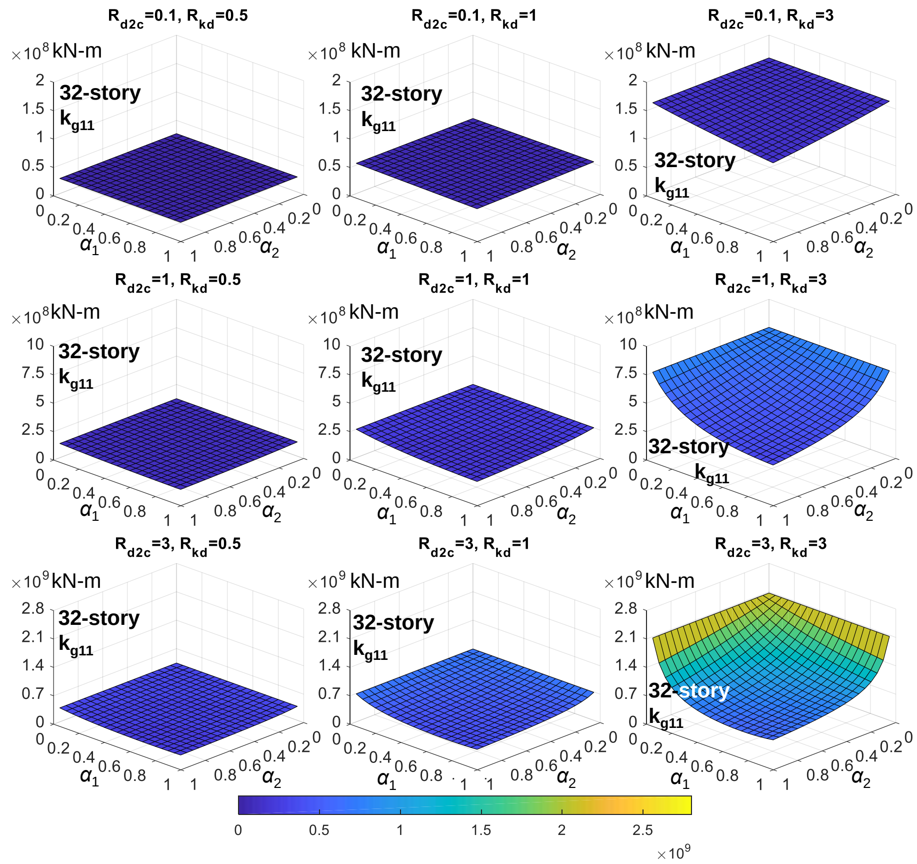


Figure 3.4.18 Relationships between k_{g11} and outrigger elevations for the 32-story dual BRB-outrigger model

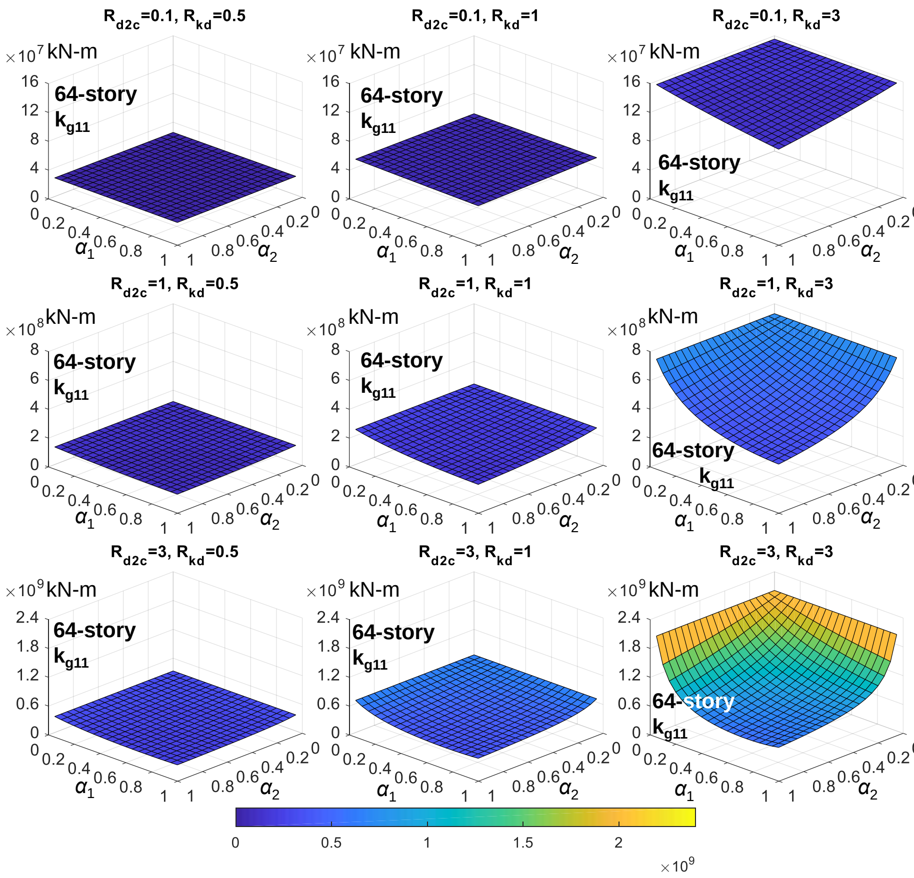


Figure 3.4.19 Relationships between k_{g11} and outrigger elevations for the 64-story dual BRB-outrigger model

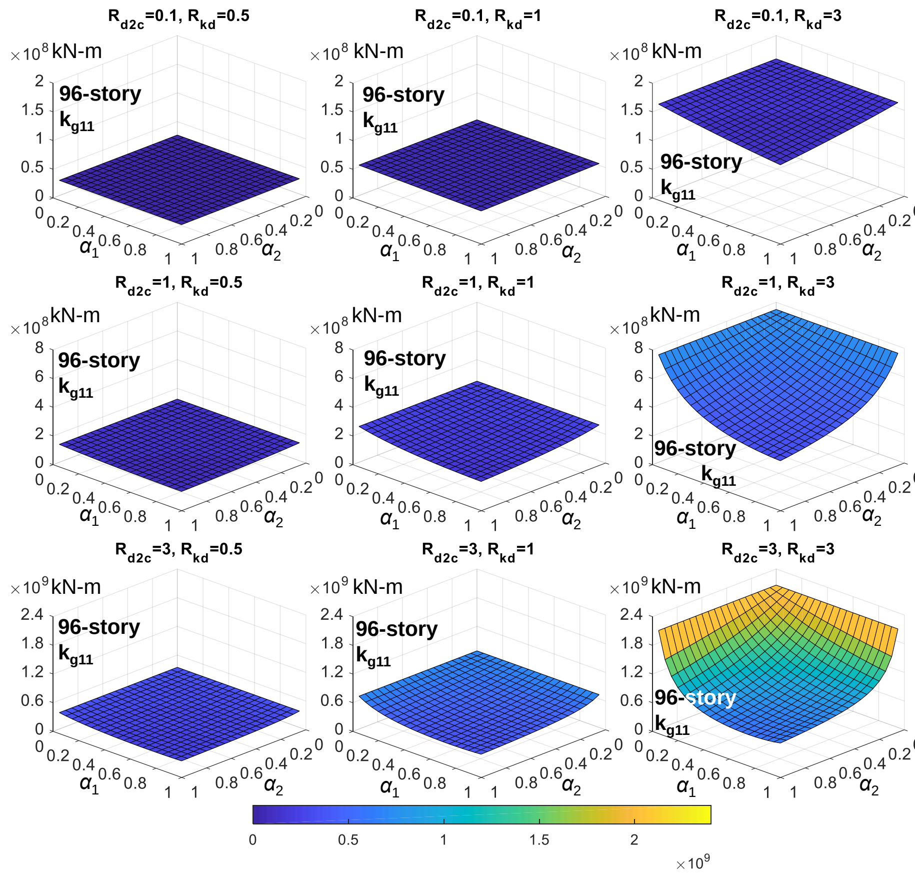


Figure 3.4.20 Relationships between k_{g11} and outrigger elevations for the 96-story dual BRB-outrigger model

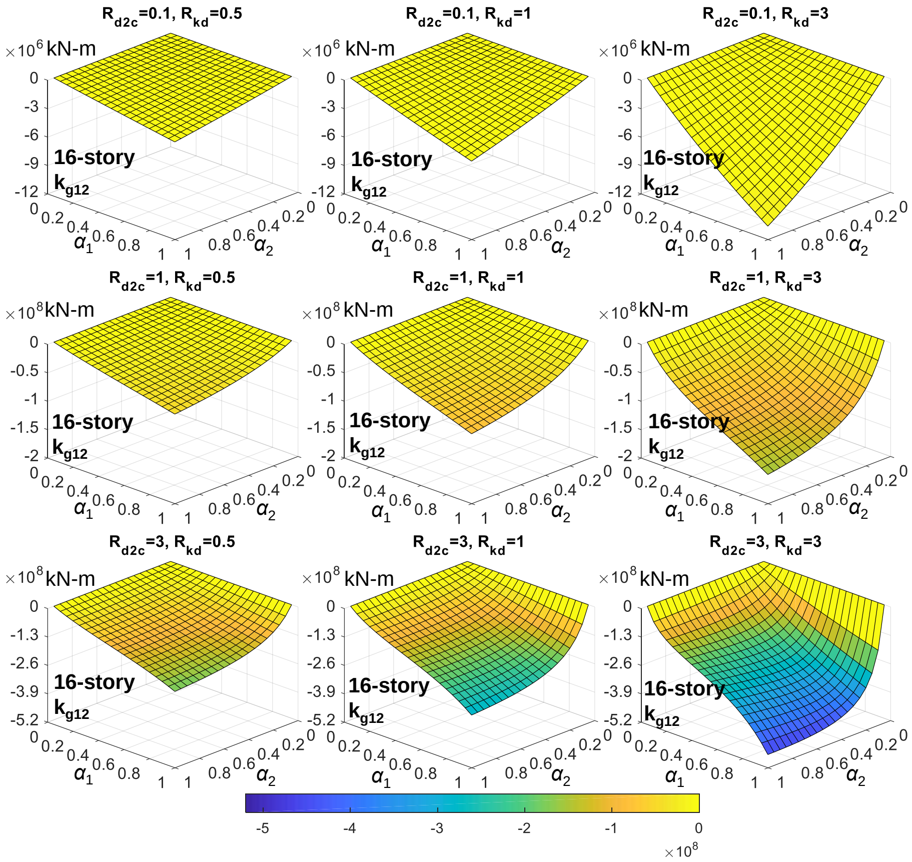


Figure 3.4.21 Relationships between k_{g12} and outrigger elevations for the 16-story dual BRB-outrigger model

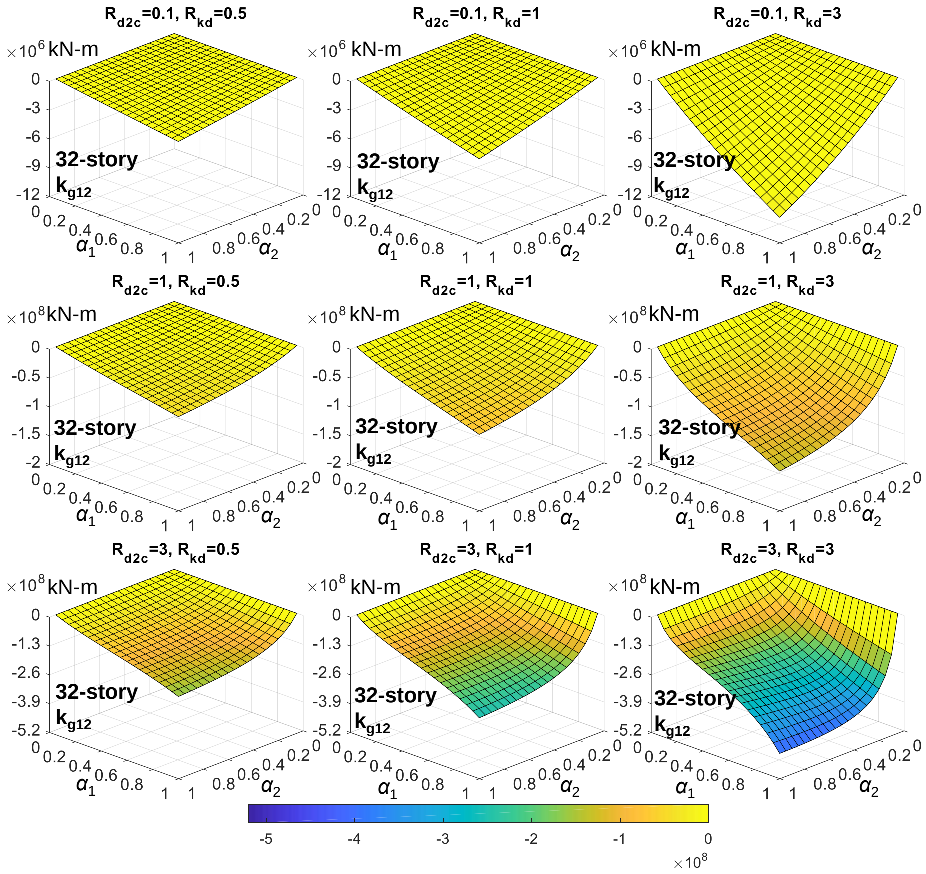


Figure 3.4.22 Relationships between k_{g12} and outrigger elevations for the 32-story dual BRB-outrigger model

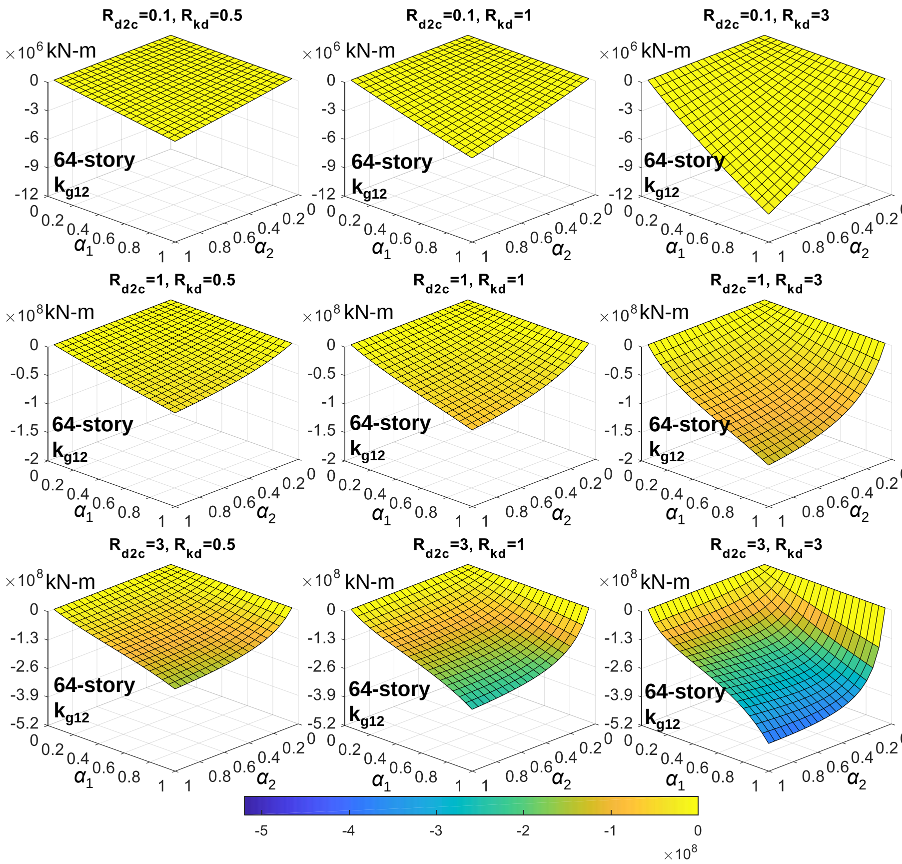


Figure 3.4.23 Relationships between k_{g12} and outrigger elevations for the 64-story dual BRB-outrigger model

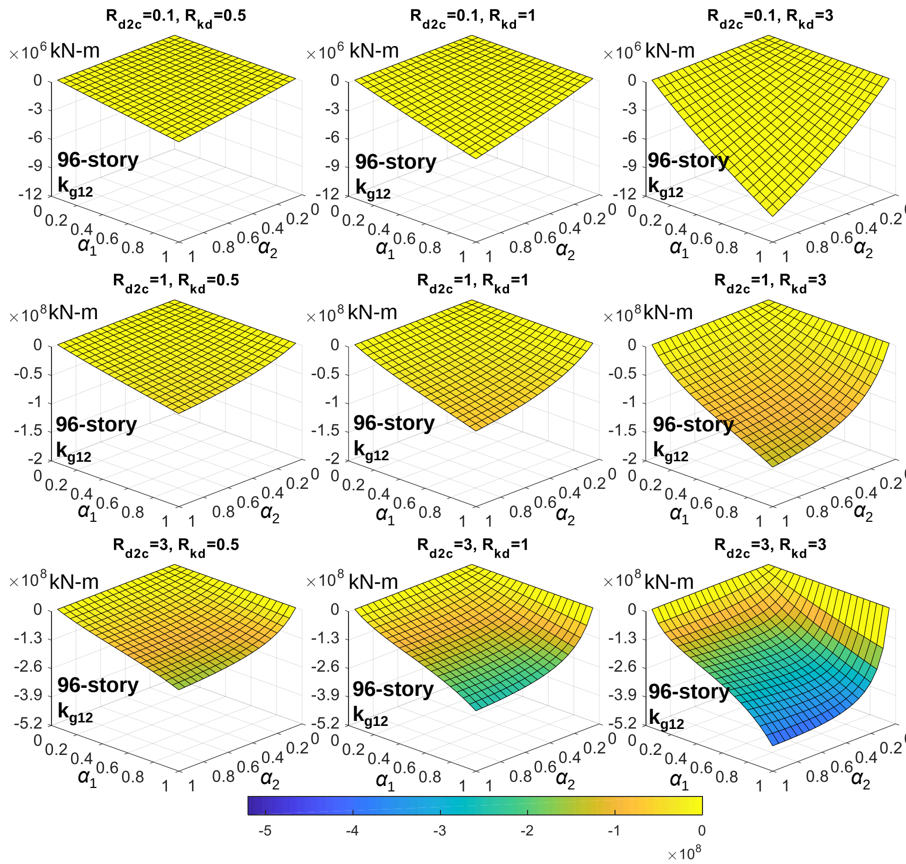


Figure 3.4.24 Relationships between k_{g12} and outrigger elevations for the 96-story dual BRB-outrigger model

3.5 MEMBER-BY-MEMBER MODELS

In order to verify the effectiveness of using the UM and MD models to represent structures with BRB-outriggers, the member-by-member (MBM) models, which include all details such as story level, core structure, outrigger truss, and floor beams, are analyzed by using OpenSees. The analysis results obtained from using MBM models are used to compare with the results obtained from using UM and DM models. Figure 3.5.1 shows the elevation and the enlargement of BRB-outrigger detail of the MBM model with single BRB-outrigger configuration. Each story is 4 m high, the mass at each floor level is separated at the mass nodes (Figure 3.5.1) on the core structure according to the tributary area. The core structure is represented by a braced frame to provide an equivalent flexural stiffness of *EI*. The perimeter columns are modeled by beam column elements, and their bases are free to rotate. Figure 3.5.1 also shows enlarged details of the outrigger truss. The outrigger truss's top and bottom chords locate at the n^{th}+1 and n^{th} floors, respectively. The BRB is modeled by using truss element, and its upper and lower ends connect to the outrigger's top chord end (Point E) and to the perimeter column at the n^{th} floor (Point D), respectively. The perimeter column between n^{th} and $(n+2)^{th}$ floors is represented by one continuous beam column element (ranges from Point D to Point H). For simplicity, no rigid diaphragm was assigned in the MBM model, and the secondary effects due to gravity loads are excluded. The material model for the BRB element is bilinear with a post-yield stiffness ratio of p (p = 0.01) and linearly elastic for the other elements. A 32-story and a 96-story model are used to construct the MBM models. The 32-story MBM model is used to verify the UM and DM models with single BRB-outrigger system. In addition, the 32-story MBM model is also used to compare the analysis results when the mass distributions along the building height are different in each analytical model. (uniform distribution, 1-m spacing, and 4-m spacing in the UM, DM, and MBM models, respectively). The 96-story model is used for structures with either single or dual BRB-outrigger system. The details including the BRB designs of the 32-story and 96-story MBM models are introduced in the following sections, and the analysis results are introduced in Chapter 4.

3.5.1 32-story MBM model

The 32-story single BRB-outrigger model with α = 0.7, R_{dt} = 0.1, and R_{dc} = 5.0 is chosen as the 32-story MBM example model. The required BRB axial stiffness (k_d = 2.43×10^6 kN/m) can be calculated from Equation (3.3.5) by replacing α with 0.7 and S_{bc} with 1.38. Figure 3.5.1 and Figure 3.5.2 show the elevation of the MBM model and the enlargement of BRB-outrigger detail, respectively. The BRB-outrigger occupies the space between the 23rd and the 24th floors. The outrigger truss is designed to have a depth equals to story height (4 m), and is modeled by truss element in the MBM model. The BRB is arranged vertically across one story height (4 m). The upper end of the BRB connects to the top chord of the outrigger truss, and the lower end of the BRB connects to a short beam which is fixed on the perimeter column at the 23rd floor. The two ends of the BRB are pinned-connection detail so that the BRB sustains only axial load. It should be noted that the as shown in Figure 3.5.2, the eccentric distance (e) between the BRB and perimeter column central lines could lead to a large bending moment demand on the perimeter column, which should be considered in real design applications. Figure 3.5.3 and Table 3.5.1 shows the detail of the BRB design. The BRB is made by SN490 grad steel with a yield stress of 325 MPa. The BRB effective axial stiffness (k_{eff}) is close to 2.43×10^6 kN/m (k_d). The steel casing size is Box 700 × 700 × 22 mm with a length of 3300 mm. The calculation of BRB axial deformation ($u_{d,y}$=5.9 mm) is introduced in Chapter 6. Therefore, the yield strength (N_y) and the maximum axial force capacity of the BRB ($N_{cu} = \beta \omega R_y N_y$, where β = 1.15 is the compression adjustment factor, ω = 1.3 is the material strain hardening factor, and R_y = 1.2 is the ratio of the expected yield stress to the specified minimum yield stress (ANSI/AISC 341-16, 2016)) are 14430 kN and 25887 kN, respectively. The perimeter columns are designed based on the axial force demand in the 1st story and considering the BRB develops its maximum compression, which is calculated as follows:

$$P_u = \left[1.2 \times \underset{\text{(dead load)}}{0.8}\left(\text{tonf/m}^2\right) + 1.6 \times \underset{\text{(live load)}}{0.3}\left(\text{tonf/m}^2\right)\right] \times \underset{\text{(tributary area)}}{104}\left(\text{m}^2\right) \times \underset{\text{(story)}}{32} + N_{cu} = 7431\ \text{tonf} \qquad (3.5.1)$$

Where 1.2 and 1.6 are load factors for the dead and live loads (ASCE, 2016), respectively. For simplicity, the live reduction factor is not considered. The size of the perimeter column is Box 1000 × 1000 × 85 mm made by SN490 grade steel (yield stress = 325 MPa). The capacity of the perimeter column is calculated based on the code specification (AISC (American Institute of Steel Construction), 2016) with a buckling length of 4 m (1st story height) and an effective length factor of 1. Therefore, the compressive capacity of the perimeter column is 100319 kN. If considering a strength reduction factor of 0.9, the axial force demand-to-capacity ratio (DCR) of the perimeter column is 0.81. Based on the design result, the parameters of the 32-story MBM example model are shown in Table 3.5.2.

Table 3.5.1 BRB design of the 32-story MBM example model

cross-sectional area (mm^2)			segment length (mm)			k_d (kN/m)	$u_{d,y}$ (mm)
core	transition	joint	core	transition	joint		
44400	56400	68400	2800	100	500	2462173	5.9

Table 3.5.2 Parameters of the 32-story MBM example model

k_d (kN/m)	k_c (kN/m)	k_t (kN/m)	R_{dc}	R_{dt}	S_{bc07}
2462173	486094	24621730	5.07	0.1	1.38

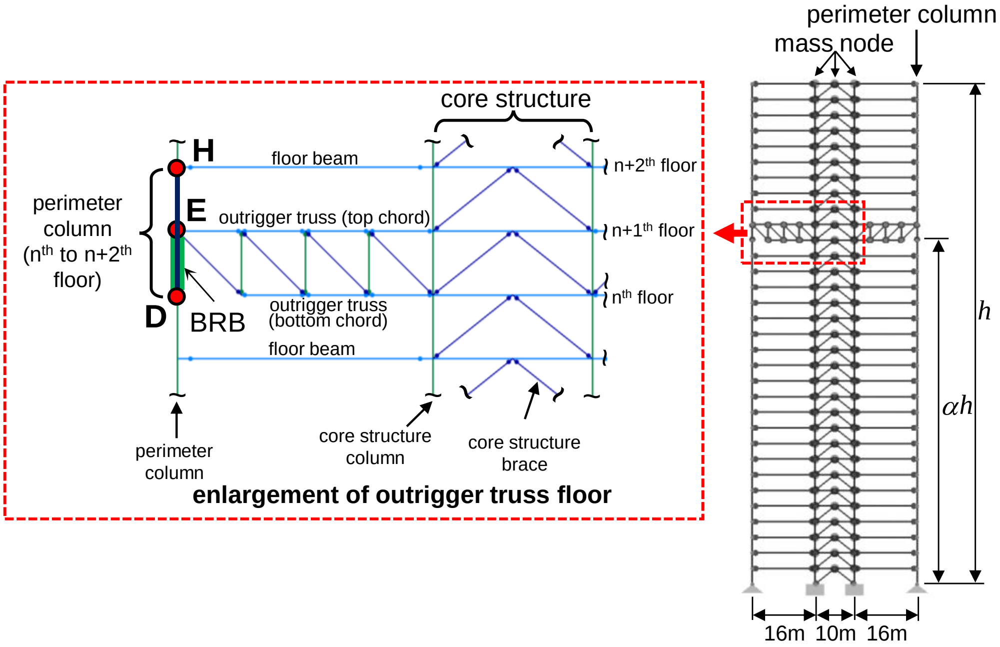


Figure 3.5.1 Elevation and enlarged BRB-outrigger detail of MBM model

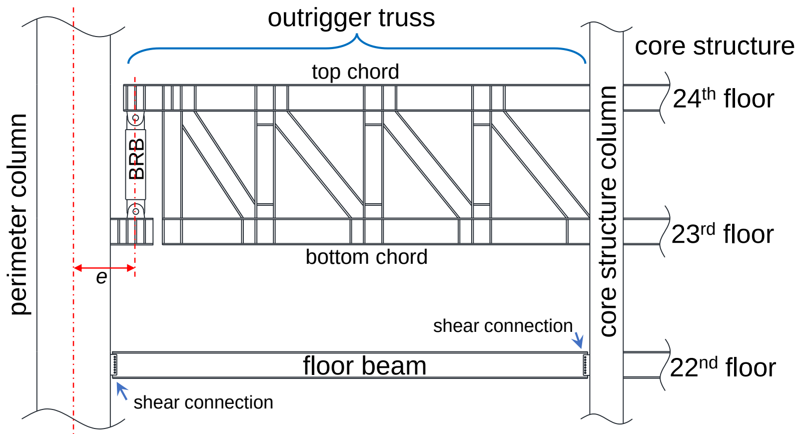


Figure 3.5.2 Enlargement of the BRB-outrigger detail in the 32-story MBM model

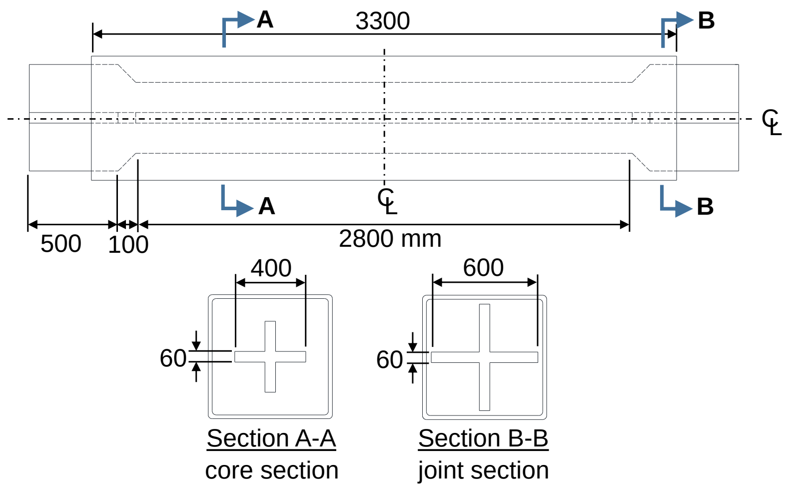


Figure 3.5.3 Detail of the BRB design in the 32-story MBM example model

3.5.2 96-story MBM model

The 96-story dual BRB-outrigger model with α_2 = 0.7 and α_1 = 0.5 is chosen as the 96-story MBM example model. Table 3.5.3, Figure 3.5.4, and Figure 3.5.5 show the design details of the BRBs in the upper (BRB_2) and lower (BRB_1) outriggers. Both the BRB_2 and BRB_1 are made by SN490 grade steel. The steel casing sizes are Box 600 × 600 × 22 mm and Box 700 × 700 × 22 mm for the BRB_2 and BRB_1, respectively. As shown in Figure 3.5.6, the lower and upper outrigger trusses occupy the space between the 34th and the 35th and between the 68th and the 69th floors, respectively. As the BRB's yield deformations are greater than 10 mm, both BRB_1 and BRB_2 are arranged vertically spanning two stories. The BRB's upper end connects to the outrigger truss top chord end, and the lower end connects to a bracket which is fixed to the perimeter column. It should be noted that as shown in Figure 3.5.6, the lower end of the BRB stands on the floor beam, which has shear connection detail at its both ends. This design configuration prevents an additional moment demand on the perimeter column if compared with the case as shown in Figure 3.5.2. However, the strength of the floor beam below the BRB should be strong enough to sustain the maximum force developed from the BRB. The perimeter column is designed based on the force demand in the 1st story and considering both BRB_2 and BRB_1 develop their maximum force capacities, which can be calculated as follows:

$$P_u = \left[1.2\times \underset{\text{(dead load)}}{0.8}\left(\text{tonf/m}^2\right)+1.6\times \underset{\text{(live load)}}{0.3}\left(\text{tonf/m}^2\right)\right]\times \underset{\text{(tributary area)}}{104}\left(\text{m}^2\right)\times \underset{\text{(story)}}{96} + N_{cu,1} + N_{cu,2} \\ = 18086 \text{ tonf} \tag{3.5.2}$$

Where $N_{cu,1}$ (16661 kN) and $N_{cu,2}$ (19725 kN) are the maximum force capacities for the BRB_1 and BRB_2, respectively. The size of the perimeter column is Box 2200 × 2200 ×100 mm made by SN490 grade steel with infill concrete of compressive strength of 10000 psi. As the infill concrete is not able to develop tensile strength, the calculation of k_c (437500 kN/m) considers the contribution from the steel only. The DCR of the perimeter column in sustaining axial loads is 0.51. The effect when considering the contribution from infill concrete on compressive axial stiffness is discussed in Chapter 10. Based on the design result, the parameters of the 96-story MBM example model are shown in Table 3.5.4.

Table 3.5.3 BRB design of the 96-story MBM example model

member	cross-sectional area (mm^2)			segment length (mm)			k_d (kN/m)	$u_{d,y}$ (mm)
	core	transition	joint	core	transition	joint		
BRB_2	33831	43281	52731	7000	75	425	883017	12.5
BRB_1	28575	40815	53055	5000	136	1364	858146	10.8

Table 3.5.4 Parameters of the 96-story MBM example model

k_{d2} (kN/m)	k_{d1} (kN/m)	k_c (kN/m)	k_{t2} (kN/m)	k_{t1} (kN/m)	R_{d2c}	R_{kd}	R_{dt2}	R_{dt1}	$S_{bc2,07}$
883017	858146	437500	8830170	8581460	2.02	0.97	0.1	0.1	0.26

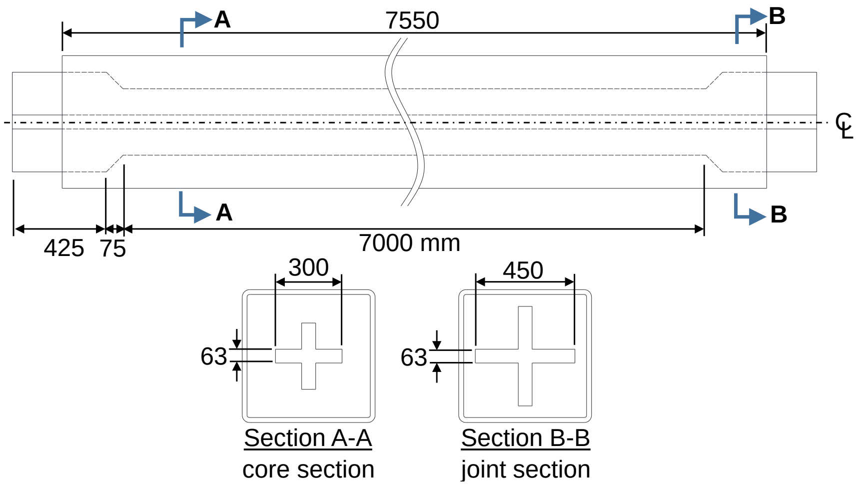


Figure 3.5.4 Detail of the BRB_2 design in the 96-story MBM example model

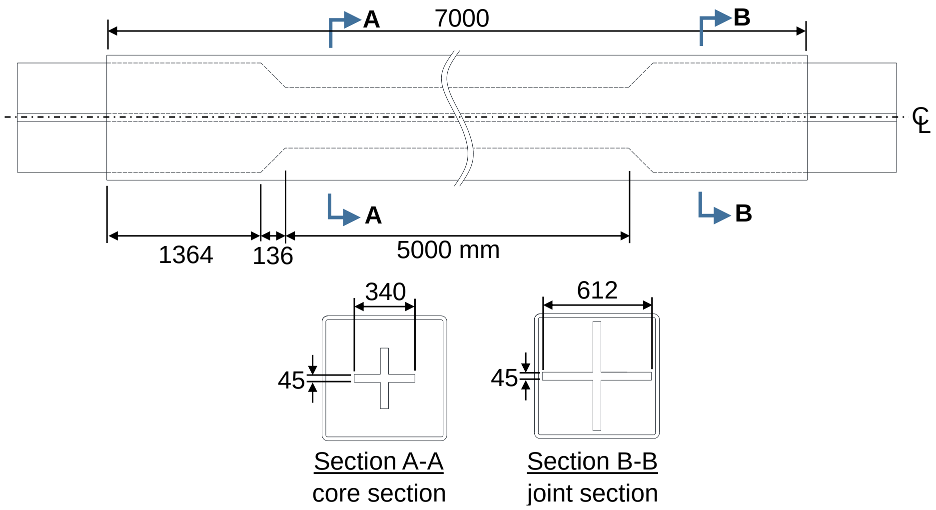


Figure 3.5.5 Detail of the BRB_1 design in the 96-story MBM example model

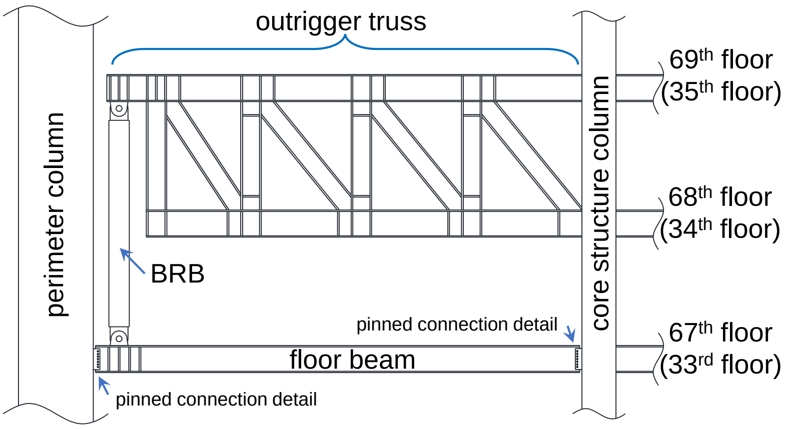


Figure 3.5.6 Enlargement of the BRB-outrigger detail in the 96-story MBM model

3.6 SUMMARY

The detail of three different types of analytical models used in this study is discussed in this chapter. In addition, the methods in constructing analytical are introduced. The summaries of this chapter are as follows:

(1) The structure with BRB-outriggers can be simplified as a cantilever column with attached rotational springs. The moments applied by each BRB-outrigger and the core structure rotations can be expressed by using matrices. With the D'Alembert's principle, the dynamic properties such as mode periods and mode shapes can be calculated. This calculation procedure is known as UM model.

(2) As the UM model is limited to linear elastic response. The DM model is developed by using OpenSees based on the simplified structure. The DM model is used to verify the UM model, and to perform SA and NLRHA.

(3) The MBM model is developed by using OpenSees in order to verify the effectiveness of using UM and DM model to represent the structure with BRB-outriggers. The details such as story level and core structure are. Table 3.6.1 summaries the analytical model used in different analyses.

(4) Two groups of dimensionless parameters are introduced for the purpose of the parametric study. The outrigger stiffness parameter is defined as the ratio of rotational stiffness provided by outrigger, when k_d is infinity, to the rotational stiffness of core structure. The outrigger stiffness parameter indicates the magnitude of the outrigger effect. The BRB stiffness parameters describe the relationships between the perimeter column axial stiffness, BRB axial stiffness, and the outrigger truss flexural stiffness. Table 3.6.2 summaries the parameters used in single and dual BRB-outrigger systems.

(5) The outrigger stiffness parameter when α=0.7 (for single BRB-outrigger system) or α_2=0.7 (for dual BRB-outrigger system) are used to indicate the magnitude of outrigger effect. The larger outrigger stiffness parameter suggests the outrigger effect is more significant.

(6) Two different methods are used for the parametric study on single BRB-outrigger system. The Met. I keeps the rotational stiffness resulted from BRB-outrigger as constant, and allow the perimeter column axial stiffness vary with α while varying α from 0 to 1. The Met. II keeps the perimeter column axial stiffness as constant,

the rotational stiffness resulted from BRB-outrigger is smaller when BRB-outrigger elevation is higher. Met. I is simpler, and Met. II is more realistic.

Table 3.6.1 The analytical models used in different analyses

Model	modal analysis	SA	NLRHA
UM	○	○	×
DM	○	○	○
MBM	○	○	○

Table 3.6.2 Summary of the outrigger and BRB stiffness parameters

BRB-outrigger configuration	outrigger stiffness parameter	BRB stiffness parameter
single	S_{bc}, S_{bc07}	R_{dt}, R_{bc} (Met. I), R_{dc} (Met. II)
dual	S_{bc1}, S_{bc2}, $S_{bc2,07}$	R_{d2c}, R_{kd}, R_{dt1}, R_{dt2}

3.7 REFERENCES

AISC (American Institute of Steel Construction) (2016) "Specifications for Structural Steel Buildings, ANSI-AISC 360-16," in *AISC (American Institute of Steel Construction)*. doi: 111.

ANSI/AISC 341-16 (2016) *Seismic Provisions for Structural Steel Buildings, American Institute of Steel Construction*. doi: 111.

ASCE (2016) *Minimum design loads for buildings and other structures ASCE7-16, ASCE standard*. doi: 10.1061/9780784412916.

Huang, B. and Takeuchi, T. (2017) "Dynamic response evaluation of damped-outrigger systems with various heights," *Earthquake Spectra*. doi: 10.1193/051816EQS082M.

McKenna, F. (1997) *Object oriented finite element programming frameworks for analysis, algorithms and parallel computing*. University of California, Berkeley. Available at: http://opensees.berkeley.edu/OpenSees/doc/fmkdiss.pdf.

4

ANALYSIS METHODS

CHAPTER CONTENTS

4.1 Introduction 4-3
4.2 Modal analysis 4-3
4.3 Spectral analysis 4-5
4.4 Nonlinear response history analysis 4-9
4.5 Analysis examples 4-10
4.5.1 Analysis results of the 32-story model 4-13
4.5.2 Analysis results of the 96-story model 4-30
4.6 Summary 4-49
4.7 References 4-51

4.1 INTRODUCTION

This chapter introduces the analysis methods for studying seismic performance of BRB-outrigger system. The spectral analysis (SA) results are used to demonstrate the seismic performance of BRB-outrigger system. The modal analysis, which is necessary for performing SA, can be performed by using the UM and DM models (introduced in Chapter 3) is introduced first. The SA procedure including modal pushover analysis (MPA) in order to consider the BRBs yield at a different time for the dual BRB-outrigger system is then introduced. The SA results are confirmed by performing the nonlinear response history analysis (NLRHA). In addition, the comparisons between the analysis results calculated from using UM, DM, and MBM models are demonstrated.

4.2 MODAL ANALYSIS

The modal analysis is performed in order to acquire the dynamic characteristics of the BRB-outrigger system such as vibration periods, mode shapes, modal participation factors, and so on, which are necessary for performing SA. As described in Chapter 3, the vibration period and mode shapes of the UM model can be calculated by solving the det **B** = 0, where **B** is the system matrix that describes the relationships between the core structure deformation and external loads. If $y(x, t)$ is the lateral displacement of the core structure at a point apart from the core structure base with a distance of x and at time of t, the following formula can be expressed by applying the D'Alembert's principle:

$$EI\frac{\partial^4 y(x,t)}{\partial x^4}+m\frac{\partial^2 y(x,t)}{\partial t^2}=0 \tag{4.2.1}$$

Where EI and m are the core structure flexural rigidity and the mass per unit height along with the core structure height, respectively. If $\phi_r(x)$ is the r^{th} mode shape, the $y(x, t)$ can be expressed by modal superposition as follows:

$$y(x,t)=\sum_{r=1}^{\infty}\phi_r(x)Q_r(t) \tag{4.2.2}$$

Substitute Equation (4.2.2) into Equation (4.2.1), the following equation can be obtained:

$$m\sum_{r=1}^{\infty}\phi_r(x)\ddot{Q}_r(t)+EI\sum_{r=1}^{\infty}\phi_r''''(x)Q_r(t)=0 \tag{4.2.3}$$

Where

$$\phi_r''''(x)=\frac{d^4\phi_r(x)}{dx^4},\quad \ddot{Q}_r(t)=\frac{d^2\phi_r(t)}{dt^2} \tag{4.2.4}$$

By multiplying Equation (4.2.3) by $\phi_n(x)$, which is the n^{th} mode shape, applying the modal orthogonality, and then integrating with respect to x from 0 to h, the Equation (4.2.3) can be expressed as follows:

$$\ddot{Q}_n(t)\int_0^h m\left[\phi_n(x)\right]^2 dx+Q_n(t)\int_0^h EI\phi_n(x)\phi_n''''(x)dx=0 \tag{4.2.5}$$

The modal mass (M_n), modal stiffness (K_n), elastic vibration period (T_n) of the n^{th} mode can be calculated as follows:

$$M_n=\int_0^h m\left[\phi_n(x)\right]^2 dx \tag{4.2.6}$$

$$K_n=\int_0^h EI\phi_n(x)\phi_n''''(x)dx,\ \text{where}\ \phi_n''''(x)=\frac{d^4\phi_n(x)}{dx^4} \tag{4.2.7}$$

$$T_n=2\pi\sqrt{\frac{M_n}{K_n}} \tag{4.2.8}$$

The modal participation factor of the n^{th} mode Γ_n is calculated as follows:

$$\Gamma_n=\frac{\int_0^h m\phi_n(x)dx}{\int_0^h m\left[\phi_n(x)\right]^2 dx} \tag{4.2.9}$$

For the UM model, the n^{th} mode shape, $\phi_n(x)$, can be calculated by combining all individual deformed shape $Y_j(x_j)$ indicated in Equation (3.2.15). For the DM model, the $\phi_n(x)$ is in a format of discrete data and can be obtained by performing modal analysis by using OpenSees directly. The Equation (4.2.6), Equation (4.2.7) and Equation (4.2.9) can be expressed as follows:

$$M_n=\sum_{i=1}^{q}m_i\left[\phi_n(x_i)\right]^2 \tag{4.2.10}$$

$$K_n=EI\sum_{i=1}^{q}\phi_n(x_i)\phi_n''''(x_i)s \tag{4.2.11}$$

$$\Gamma_n=\frac{\sum_{i=1}^{q}m_i\phi_n(x_i)}{\sum_{i=1}^{q}m_i\left[\phi_n(x_i)\right]^2} \tag{4.2.12}$$

Where q is the number of nodes with the lumped mass that distribute along with the core structure, x_i is the distance of the i^{th} node measured from the core structure base, and s (= 1 m) is the spacing of nodes with the lumped mass in the DM model (Figure 3.2.7).

4.3 SPECTRAL ANALYSIS

The spectral analysis (SA) procedure, which incorporates the equivalent damping ratio (Nathan M. Newmark and Rosenblueth, 1971) in order to consider the effect of BRB's inelastic deformation, is used to evaluate the seismic performance of a structure with BRB-outrigger system. The level 2 design spectrum of the Building Standard Law of Japan is adopted as the seismic demand for the SA. The design spectrum is defined by $S_{A0}\times G_s$, where the S_{A0} and G_s are the design acceleration response spectrum at the ground surface and surface soil layer amplification factor, respectively, and are defined as follows:

$$S_{A0}\left(\mathrm{m/sec^2}\right)=\begin{cases}3.2+30T & T<0.16\\ 8.0 & 0.16\le T<0.64\\ 5.12/T & 0.64\le T\end{cases}$$
$$G_s=\begin{cases}1.5 & T<0.64\\ 1.5\left(T/0.64\right) & 0.64\le T<0.864\\ 2.025 & 0.864\le T\end{cases} \tag{4.3.1}$$

Wher T is the vibration period.

The response of each mode is computed based on the design spectrum separately, then the overall response is obtained by combining the responses from all the considered modes using the square root of the sum of the squares (SRSS) rule. In the analytical model, as only the BRBs can yield, it is anticipated that the drops in system stiffness and changes in mode shapes are small after the BRBs yield. Therefore, it is assumed that the modal superposition principle based on elastic mode shapes is still applicable when the BRBs deform inelastically (Chopra and Goel, 2002). The maximum inter-story drift ($\gamma_{\max}$), core structure base shear ($V_{c,\max}$), and overturning moment ($M_{c,\max}$) are computed based on the SRSS combined deformed shape. The responses of the first four modes are considered. The SA procedure is illustrated as step-by-step as follows:

(1) Calculate the elastic vibration periods and mode shapes for the first four modes as described in Section 4.2 by using both the UM and DM models.

(2) Obtain the relationship between base shear and roof displacement when the model deforms in the n^{th} mode shape. For the single BRB-outrigger system, as the BRBs in the outrigger are the only elements that could yield, the relationship between base shear and roof displacement when the model deforms in the n^{th} mode is a bilinear relation as shown in Figure 4.3.1(a). Where $y_{top,n}$ is the roof displacement when the BRB yields, $y_{\max,n}$ is maximum roof displacement, K'_n is the post-yield modal stiffness, and $K_{eq,n}$ is the equivalent stiffness when the roof displacement raches $y_{\max,n}$ in the n^{th} mode shape. The K'_n and $K_{eq,n}$ can be calculated as follows:

$$K'_n = \left(\frac{T_n}{T'_n} \right)^2 K_n = p_n K_n \tag{4.3.2}$$

$$K_{eq,n} = K'_n + \frac{K_n - K'_n}{\mu_n} \tag{4.3.3}$$

Where T'_n is the vibration period after BRBs have yielded in the n^{th} mode shape, and p_n is the post-yield stiffness ratio of the n^{th} mode. The T'_n can be calculated by replacing the k_d with post-yield stiffness ($pk_d = 0.01k_d$) in the UM and DM models from the modal analysis. For the dual and multiple BRB-outrigger systems, as the BRBs in different layers of outrigger would not yield simultaneously, the relation between base shear and roof displacement would not be bilinear. Thus, it would be difficult and impractical to use the UM model to estimate the system stiffness after the first BRB-outrigger yields. Instead, the DM model is used to perform modal pushover analysis (MPA) (Chopra and Goel, 2002) by using OpenSees to obtain the base shear and roof displacement relation for the dual BRB-outrigger system as shown in Figure 4.3.1(b). It should be noted that, as it is anticipated that the yieldings of BRBs only slightly change the elastic mode shapes, while performing MPA in OpenSees, the lateral load pattern in MPA for each mode is kept the same as the elastic mode shape even after the BRBs yield.

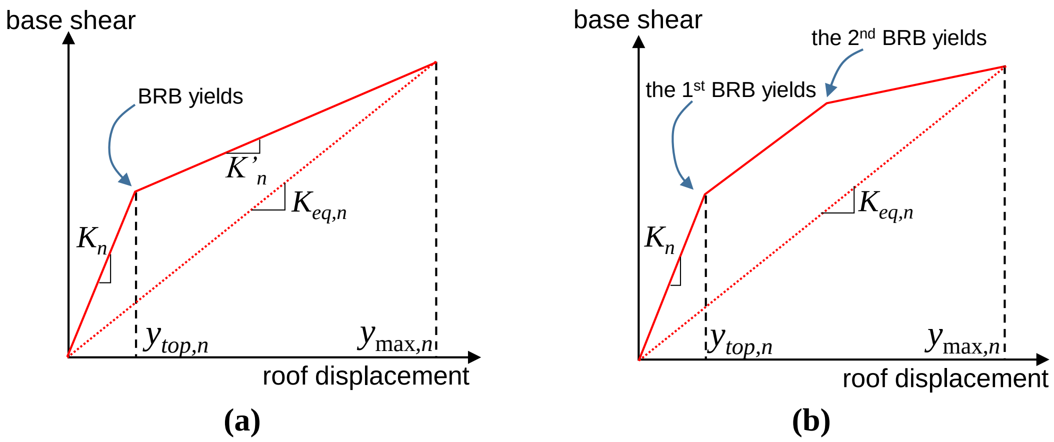


Figure 4.3.1 The base shear and roof displacement relationships for the (a) single and (b) dual BRB-outrigger systems

(3) Calculate the equivalent damping ratio ($h_{eq,n}$) of the n^{th} mode with ductility of μ_n. The $h_{eq,n}$ and μ_n can be calculated as follows:

$$h_{eq,n} = h_0 + \frac{1}{y_{\max,n}} \int_{y_{top,n}}^{y_{\max,n}} \frac{E_d(y)}{4\pi E_s(y)} dy, \quad \mu_n = \frac{y_{\max,n}}{y_{top,n}} \tag{4.3.4}$$

Where h_0 (= 0.02) is the inherent damping ratio, $E_d(y)$ and $E_s(y)$ are energy dissipated by BRB-outrigger system per loop, and strain energy when roof displacement reaches y, respectively, as shown in Figure 4.3.2. For single BRB-outrigger system, the Equation (4.3.4) can be expressed by using p_n as follows:

$$h_{eq,n} = h_0 + \frac{2}{\pi p_n \mu_n} \ln\left(\frac{1 - p_n + p_n \mu_n}{\mu_n^{p_n}}\right), \quad \mu_n = \frac{y_{\max,n}}{y_{top,n}} \geq 1 \tag{4.3.5}$$

The Equation (4.3.5) is used for the UM model with single BRB-outrigger system. For the dual BRB-outrigger system and DM model, the $h_{eq,n}$ is calculated from the Equation (4.3.4) by using the roof displacement and base shear relationship obtained from MPA. This calculation is completed by programming, which is introduced in Chapter 5.

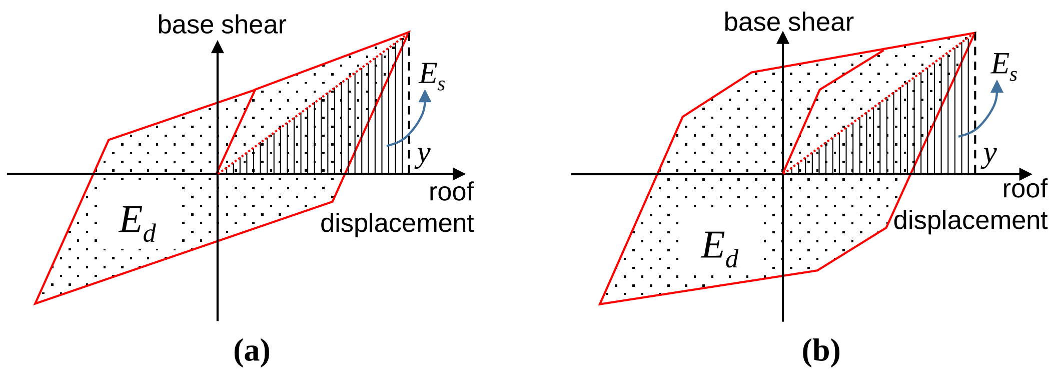


Figure 4.3.2 The relationships between E_d and E_s for the (a) single and (b) dual BRB-outrigger systems

(4) As the damping ratio is increased because of BRBs' inelastic responses, the response spectrum is then reduced. The reduction factor $D_{h,n}$ of the n^{th} mode response is expressed as follows (Kasai, Fu, and Watanabe, 1998):

$$D_{h,n} = \sqrt{\frac{1+\kappa h_0}{1+\kappa h_{eq,n}}}, \quad \begin{array}{l} \kappa = 25 \text{ for observed ground motions} \\ \kappa = 75 \text{ for artifical ground motions} \end{array} \tag{4.3.6}$$

If $S_d(T, h_d)$ is the spectral displacement at period T and damping ratio h_d, the maximum roof displacement ($y'_{\max,n}$) can be calculated as follows:

$$y'_{\max,n} = D_{h,n} S_d\left(T_{eq,n}, h_0\right) \Gamma_n \phi_n\left(h\right) \tag{4.3.7}$$

Where Γ_n is the modal participation factor of the n^{th} mode, $\phi_n(h)$ is the roof displacement in the n^{th} mode shape, and $T_{eq,n}$ is the equivalent vibration period of the n^{th} mode, which is calculated as follows:

$$T_{eq,n} = T_n \sqrt{\frac{K_n}{K_{eq,n}}} \tag{4.3.8}$$

(5) If the difference between the $y'_{\max,n}$ calculated in Equation (4.3.7) and $y_{\max,n}$ used in Equation (4.3.4) or Equation (4.3.5) is greater than allowable limit (1% for example), replace the $y_{\max,n}$ with $y'_{\max,n}$ in Equation (4.3.4) or Equation (4.3.5) and repeat from Step (2). This iteration procedure should be continued until the $y_{\max,n}$ is close enough to $y'_{\max,n}$ (when the difference is smaller than 1%).

(6) Repeat the above steps for the first four mode responses, then combine by using SRSS rule.

If $\psi(x)$ is the SRSS combined deformed shape, the maximum roof drift ($\theta_{\max}$), inter-story drift ($\gamma_{\max}$), core structure base shear ($V_{c,\max}$), and core structure base overturning moment ($M_{c,\max}$) can be calculated as follows:

$$\theta_{\max} = \frac{\left|\psi\left(h\right)\right|}{h} \tag{4.3.9}$$

$$\gamma_{\max} = \max\left[\left|\frac{d}{dx}\psi\left(x\right)\right|, 0 \le x \le h\right] \tag{4.3.10}$$

$$V_{c,\max} = EI\left|\frac{d^3}{dx^3}\psi\left(x\right)\right|_{x=0} \tag{4.3.11}$$

$$M_{c,\max} = EI\left|\frac{d^2}{dx^2}\psi\left(x\right)\right|_{x=0} \tag{4.3.12}$$

The $\theta_{\max}$, $\gamma_{\max}$, $V_{c,\max}$, and $M_{c,\max}$ are used as indicators to indicate the seismic performance of the structure with BRB-outrigger system.

4.4 NONLINEAR RESPONSE HISTORY ANALYSIS

The result of nonlinear response history analysis (NLRHA) performed by using the DM model developed in OpenSees is used to verify the SA results. The input ground motions include 7 observed and 1 artificial ground motions. Table 4.4.1 and Figure 4.4.1 show the details and response spectra of the originally observed ground motions, respectively. Figure 4.4.1 also shows the design spectrum as indicated in Equation (4.3.1). For each analysis, the spectral accelerations of the ground motions are scaled so that the mean of spectral accelerations fit the design spectral acceleration within the range of $0.2T_1$ and $1.5T_1$, where T_1 is the 1st mode period (ASCE, 2016). The Rayleigh damping ratio of 0.02 for the 1st and 2nd modes was applied in all NLRHA. The means of NLRHA results obtained from using 8 ground motions are used to verify the SA results.

Table 4.4.1 The ground motions used in NLRHA

ground motion	earthquake event	date	magnitude	depth (km)	PGA (gal)
Tohoku	Miyagi	Jun. 12th, 1978	M7.7	44	258
El Centro	El Centro	May 18th, 1940	M6.9	16	342
Taft	Kern Country	Jul. 21st, 1952	M7.3	16	176
Kumamoto	Kumamoto	Apr. 16th, 2016	M7.0	10	627
KobeJMA	Great Hanshin	Jan. 17th, 1995	M6.9	18	821
Sendai	Tohoku	Mar. 11th, 2011	M9.0	29	1517
ChiChi	ChiChi	Sep. 21st, 1999	M7.3	33	439
BCJ-L2	artificial	-	-	-	356

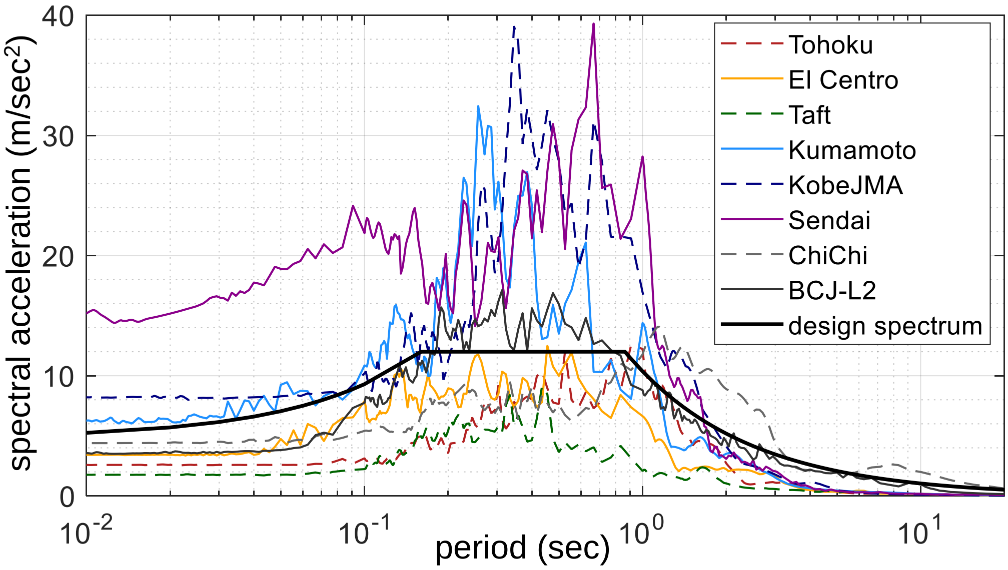


Figure 4.4.1 The response spectra of ground motions used in NLRHA

4.5 ANALYSIS EXAMPLES

The modal analysis, SA, and NLRHA are performed on the 32-story and 96-story analytical models with BRB-outriggers using UM, DM, and MBM models, which are introduced in Chapter 3. The analysis results obtained from MBM models are used to verify the effectiveness of using UM and DM model. In addition, the seismic response between structures with one (single, $\alpha = 0.7$) or two layers (dual, $\alpha_2 = 0.7$, $\alpha_1 = 0.5$) of BRB-outrigger, and between the conventional and BRB-outrigger systems are compared. Table 4.5.1 and Table 4.5.2 show the identifications of 32- and 96-story models, respectively. The conventional outrigger systems with single and dual outrigger configurations are known as SingleElastic and DualElastic, respectively. The BRB-outrigger systems with single and dual outrigger configurations are known as Single and Dual, respectively. Figure 4.5.1 and Figure 4.5.2 illustrate the simplified structures and DM models, respectively, for the Single, SingleElastic, Dual, and DualElastic models. In order to compare the seismic response between the BRB-outrigger and conventional outrigger systems, their fundamental vibration periods are set to be the same. Therefore, the outrigger truss flexural stiffness of the SingleElastic model, $k_{t,e}$, equals to the combined stiffness of k_t and k_d of the Single model. For the same reason, the upper and lower outrigger flexural stiffness of the DualElastic model, $k_{t2,e}$ and $k_{t1,e}$, equal to the combined stiffness of k_{t2} with k_{d2} and k_{t1} with k_{d1}, respectively. In the DM models as shown in Figure 4.5.2, the SingleElatic and DualElastic models are constructed by using the Single and Dual models but replacing bilinear material of the BRB element with a linear elastic material model. Thus, the rotational stiffness provided by BRB-outrigger and conventional outrigger are identical to each other. In addition, as indicated in Chapter 3, the DM model assumes the mass is lumped at the nodes that evenly distribute along with the core structure height with a spacing of 1 m. However, in the MBM model, the mass is concentrated in each story level (story height of 4 m). Therefore, analyses on the 32-story DM model with 1-m and 4-m mass spacing are performed in order to confirm the effect of different mass spacing on seismic response. The DM model with 4-m mass spacing is known as DM4 model. Table 4.5.3 summaries the analytical models and analyses performed. The details of the analytical models are introduced in Chapter 3, and analysis results are introduced in the following sections.

Figure 4.5.3 to Figure 4.5.5 illustrate the conventional outrigger system with three different connection details at outrigger end to perimeter column. In order to prevent additional bending moment applied on perimeter column, the outrigger end can be designed with a pin connection and attached to perimeter column as shown in Figure 4.5.3, or with two stoppers below and above the outrigger truss's top chord and with inserted deformable material such as rubber between the gaps (Figure 4.5.4). In addition, the top chord of the outrigger truss can be connected to perimeter column with moment connection detail and a reduced beam section (RBS) as shown in Figure 4.5.5 in order to limit the maximum bending moment applied on the perimeter column.

Table 4.5.1 Identifications of the 32-story models

32-story model	without BRB	with BRB
without outrigger	Core	
1 layer outrigger	SingleElastic	Single

Table 4.5.2 Identifications of the 96-story models

96-story model	without BRB	with BRB
without outrigger	Core	
1 layer outrigger	SingleElastic	Single
2 layers outriggers	DualElastic	Dual

Table 4.5.3 The analytical models and the corresponding analyses performed

model	modal analysis		SA		NLRHA	
	32-story	96-story	32-story	96-story	32-story	96-story
UM	○	○	○	×	×	×
DM	○	○	○	○	○	○
DM4	○	×	○	×	○	×
MBM	○	○	○	○	○	○

○: performed, ×: not performed

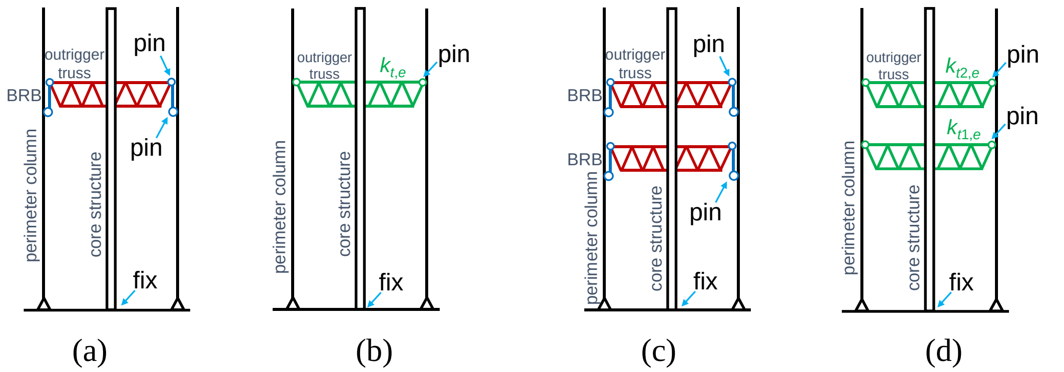


Figure 4.5.1 Illustrations of the simplified model of the (a) Single, (b) SingleElastic, (c) Dual, and (d) DualElastic models

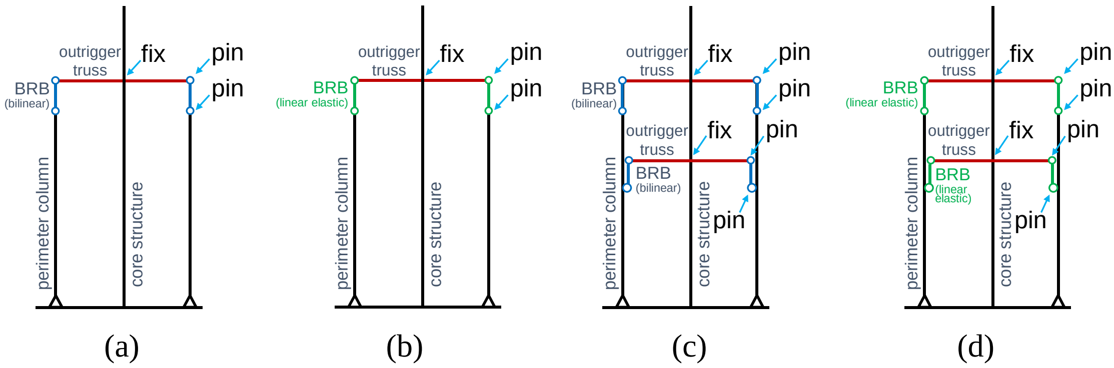


Figure 4.5.2 Illustrations of the (a) Single, (b) SingleElastic, (c) Dual, and (d) DualElastic DM models

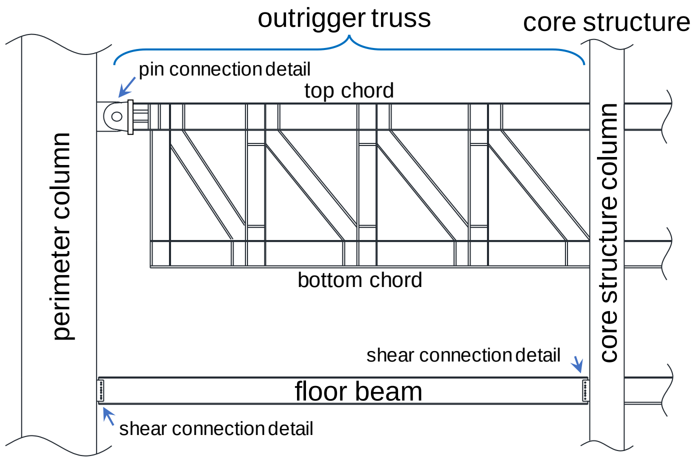


Figure 4.5.3 Illustration of conventional outrigger with pin connection detail at outrigger end

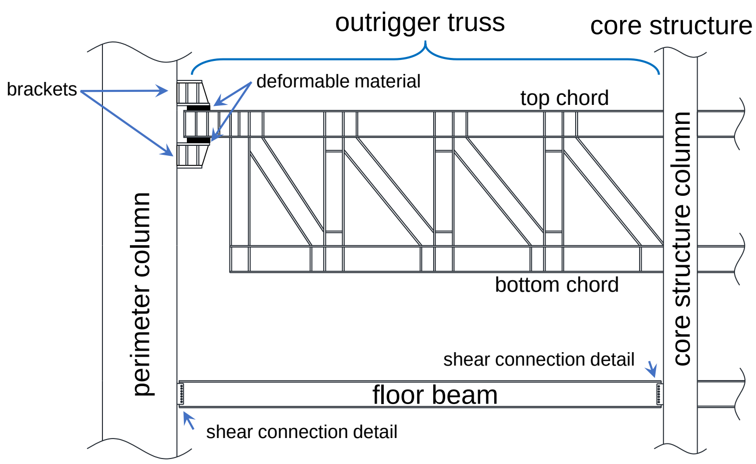


Figure 4.5.4 Illustration of conventional outrigger with stoppers detail at outrigger end

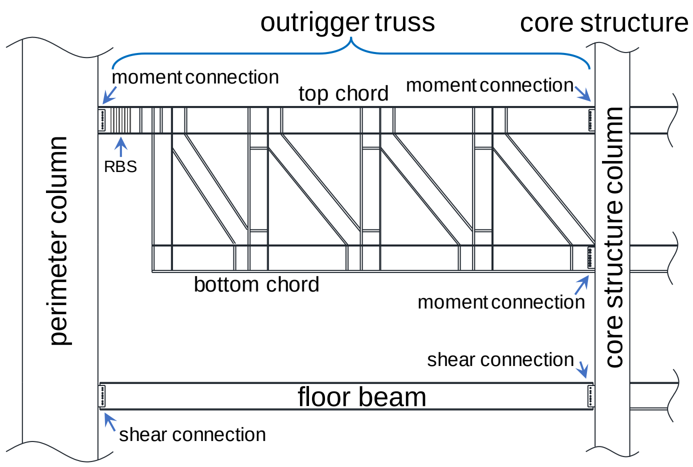


Figure 4.5.5 Illustration of conventional with RBS detail at outrigger end

4.5.1 Analysis results of the 32-story model

Table 4.5.4 and Figure 4.5.6 show the modal analysis results and mode shapes of the first four modes. The vibration periods calculated from using UM and DM models are close to each other. This suggests that the DM model developed in OpenSees with the mass spacing of 1 m is a good representation of the UM model. As the masses are concentrated at nodes with 4 m spacing in MBM and DM4 model, the vibration periods are slightly longer if compared with UM and DM models. In addition, the vibration period differences between DM4 and MBM models could be due to that the core structure span of 10 m is not included in the DM4 model, and the behavior of the braced frame in the MBM model may not accurately resemble a cantilever column in the DM and DM4 models. Table 4.5.4 also shows the vibration periods of the models with (Single) and without (Core) BRB-outrigger. The much shorter 1st to the 4th mode vibration periods found in the Single models as compared to Core models suggest the outrigger effectively in increasing system stiffness. In addition, if compare the Single the Core models, the 1st mode mass participation ratios become greater and the 2nd mode mass participation ratios become smaller. It appears that when the outrigger system is used, the contribution from the 1st mode response becomes more significant. The mode shapes shown in Figure 4.5.6 suggest that the mode shapes calculated from UM, DM, DM4, and MBM models are similar. The 1st and 2nd mode shapes show that, if compared with the Core model, the Single models could better reduce inter-story drift responses near the outrigger elevation.

Table 4.5.4 The modal analysis results of the 32-story models

model \ mode		vibration period (sec)				mass participation ratio (%)			
		1st	2nd	3rd	4th	1st	2nd	3rd	4th
UM	Core	3.472	0.554	0.198	0.101	68.2	21	7.2	3.6
	Single	2.476	0.511	0.198	0.100	72.6	16.5	7.2	3.7
DM	Core	3.499	0.558	0.199	0.102	68.2	20.9	7.2	3.7
	Single	2.489	0.515	0.199	0.100	72.6	16.5	7.2	3.7
DM4	Core	3.581	0.571	0.204	0.104	65.7	23.6	6.1	4.6
	Single	2.540	0.529	0.204	0.102	70.1	19.1	6.2	4.6
MBM	Core	3.758	0.632	0.244	0.137	67.4	21.4	7.5	3.7
	Single	2.526	0.561	0.244	0.135	73.5	15.3	7.4	3.8

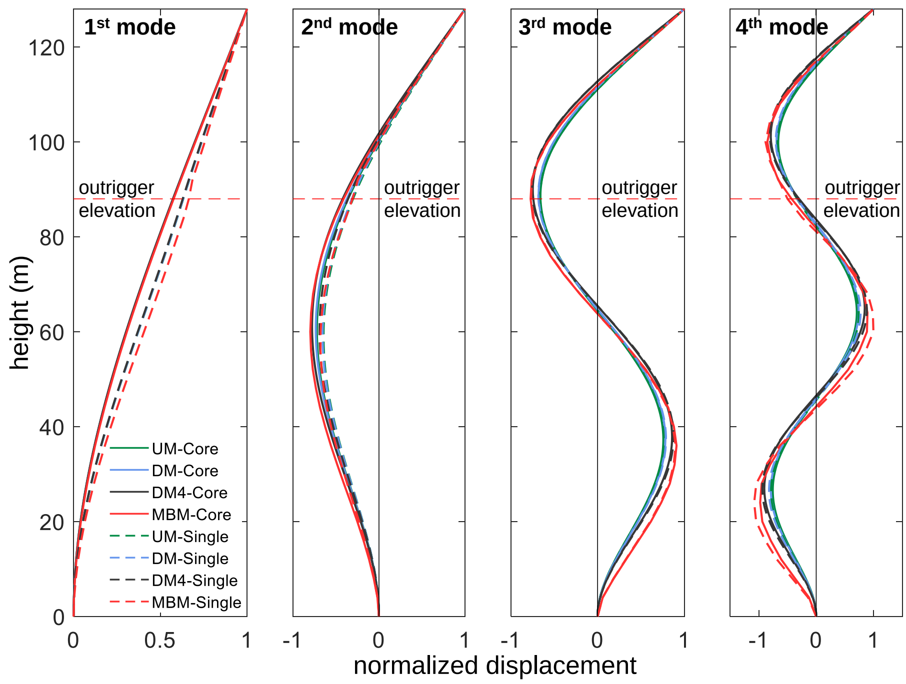


Figure 4.5.6 The first four mode shapes of the 32-story example model

Table 4.5.5 and Figure 4.5.7 show the scale factors and the scaled spectra of the ground motions used in the NLRHA. As the 1st mode vibration periods (T_1=2.489 sec) of the Single and SingleElastic models are the same, the scale factors of ground motions for both the Single and SingleElastic models are the same. The same scaling factors are also used for NLHRA on Core models in order to compare their seismic performance. Figure 4.5.8, Figure 4.5.9, and Figure 4.5.10 show the roof drift (θ) histories calculated from each NLRHA from using DM, DM4, and MBM models, respectively. The responses and trends calculated from using DM and DM4 models are very close to each other, and the differences between DM, DM4, and MBM models are marginal. Although the peak responses between DM1, DM4, and MBM models are slightly different, this may not affect the purpose of parametric analysis in investigating the optimal design parameters. Therefore, the DM model is used for performing both SA and NLRHA in the parametric study sections. Figure 4.5.11 shows the θ histories calculated from each NLRHA by using MBM models considering the secondary effect due to the dead and live loads. The load factors of 1.2 and 1.6 are applied for the dead and live loads, respectively. The results of θ response calculated from using the MBM models considering secondary effect are only slightly larger than the results without considering secondary effect. Figure 4.5.12 shows the BRB normalized axial force (N/N_y) and core strain responses of each NLRHA calculated from using DM, DM4, and MBM models. The yield deformation, elastic stiffness, and post-yield stiffness can be well modeled by using the 1-m long BRB element in the DM and DM4 models. The differences of the BRB core strain ranges between DM and MBM model could be due to that the 10-m span of the core structure is not included in the DM and DM4 model. Figure 4.5.13, Figure 4.5.14 and Figure 4.5.15 show the maximum lateral displacement, maximum inter-story drift, and maximum lateral acceleration distributions throughout the building height, respectively. The responses with and without considering secondary effect are also compared (in MBM models only). The analysis results with and without considering secondary effect are only slightly different. The trends of lateral deformation and inter-story drift distributions throughout the building height are not significantly affected because of the secondary effect. As the secondary effect only results in marginal differences, and would not significantly affect the purpose of the parametric study, for simplicity, the secondary effect is neglected in the parametric study. Figure 4.5.16 shows the maximum seismic responses calculated from each NLRHA. Where

a_{max} and $C_{1,max}$ are the maximum roof acceleration and the maximum axial force of the perimeter column in the 1st story, respectively. The analysis results obtained from using DM model generally well agree with the results obtained from MBM model.

As shown from the roof drift histories (Figure 4.5.8, Figure 4.5.9, and Figure 4.5.10) and the maximum lateral deformation throughout the building height (Figure 4.5.13), the Core models exhibit larger θ_{max} responses than the others in most of the NLRHA. The SingleElastic model also exhibits the largest θ_{max} responses in the NLRHA with El Centro, Kumamoto, KobeJMA, Sendai, and ChiChi ground motions. This could be due to that the SingleElastic model, which has a larger system stiffness than the Core model but without energy dissipation mechanism, may amplify the base shear demand and thus exhibit larger θ_{max}. In addition, if compare the SingleElastic with the Single models, the Single models with BRBs' yielding and energy dissipation mechanisms are more effective in reducing the roof drift responses after the θ_{max} occurs. In addition, as indicated from the maximum inter-story drift distributions (Figure 4.5.14), the inter-story drift at the elevation close to outrigger can be effectively reduced in both the Single and SingleElastic models. However, the Single models generally achieve smaller maximum inter-story drift responses than the SingleElastic models. As shown in Figure 4.5.15, the reduction in acceleration response because of the BRB-outrigger can be less significant if compared with the reductions in θ_{max} and γ_{max}. This is because that the BRB-outrigger system amplifies the acceleration response due to greater stiffness of the system. However, the yielding of the BRB could result in slightly smaller acceleration response if compared with the SingleElastic model. Figure 4.5.16 summaries the maximum seismic responses. The numbers shown in Figure 4.5.16 indicate the ratio of peak response if compared with the Core model (θ_{max}, a_{max}, and $M_{c,max}$) or compared with the SingleElastic model ($C_{1,max}$) under each ground motion. The maximum roof acceleration (a_{max}) responses as shown in Figure 4.5.16 suggest that when outrigger is applied, the increase of system's stiffness could amplify the acceleration and base overturning moment responses if compared with the SingleElastic models. However, when the energy dissipation mechanism is implemented in the outrigger system (Single models), the amplified seismic responses shown in the SingleElastic cases could be avoided. Figure 4.5.16 also shows the maximum perimeter column axial force ($C_{1,max}$) calculated from NLRHA. As the outrigger provides resisting moment on the core structure by mobilizing the perimeter column's axial stiffness, the axial force demand for the perimeter column must increase. The responses of $C_{1,max}$ shown in Figure 4.5.16 indicate that the yield of BRB can effectively limit $C_{1,max}$. In summary, based on the peak responses shown in Figure 4.5.16, the BRB-outrigger system is efficient in reducing θ_{max}, $M_{c,max}$, and $C_{1,max}$ responses. The greater seismic intensity generally results in greater reductions in θ_{max} and $M_{c,max}$ because of greater BRB energy dissipation. The acceleration response can be amplified in the SingleElastic models, but can be reduced when BRB is incorporated.

Since the BRB is modeled using bilinear truss element, the relationship between the moment applied by the BRB-outrigger on core structure (M_o) and the core structure rotation at outrigger elevation (θ) is a bilinear hysteretic response. Figure 4.5.17 shows the relationship between M_o and θ of the 32-story example model obtained using DM model. This bilinear hysteretic response between M_o and θ can cause different phases during an earthquake. For example, when the core structure rotation is in positive direction, the moment applied by the BRB-outrigger can be in either clockwise or counterclockwise. This effect leads to a severer vibration response of the overturning moment in the Single model (Figure 4.5.18). Although the overturning moment in the Single model behaves a high frequency-like response, the peak overturning moments are still smaller than the Core model.

Table 4.5.5 Scale factors of ground motions used for 32-story model in NLRHA

Tohoku	El Centro	Taft	Kumamoto	KobeJMA	Sendai	ChiChi	BCJ-L2
2.03	2.54	4.90	1.84	1.19	1.33	0.65	1.14

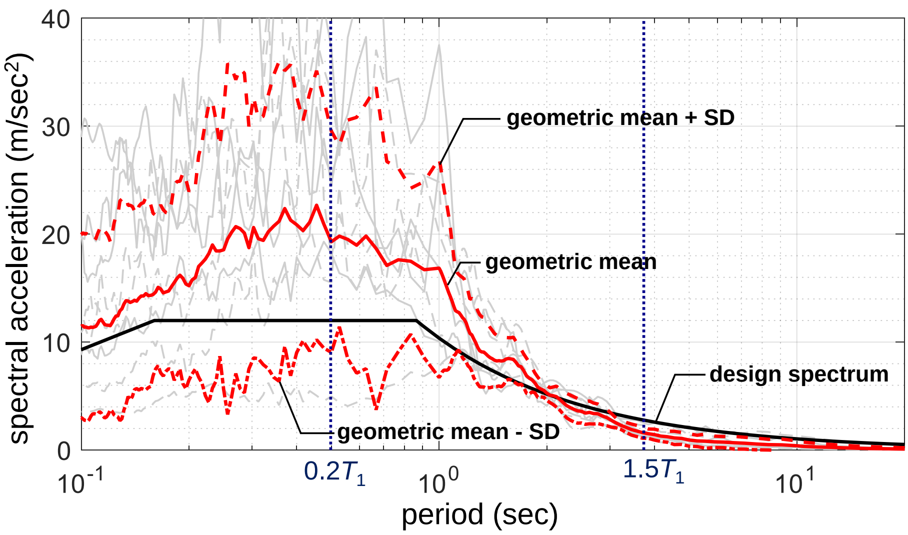


Figure 4.5.7 Spectra of the scaled ground motions used for 32-story Single and SingleElastic models in NLRHA (SD = standard deviation)

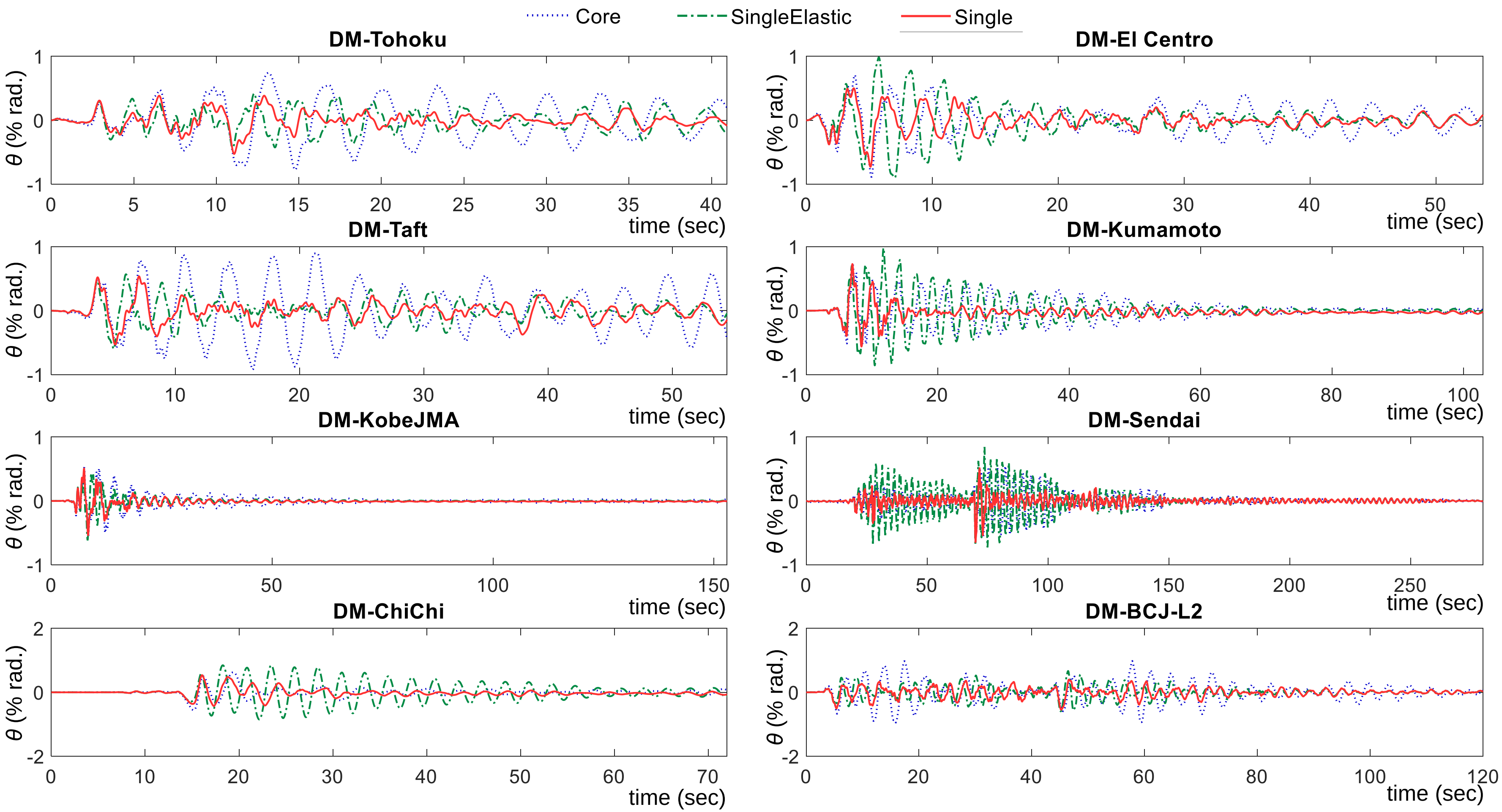


Figure 4.5.8 The roof drift history of the 32-story example model under different ground motions from NLRHA using DM models

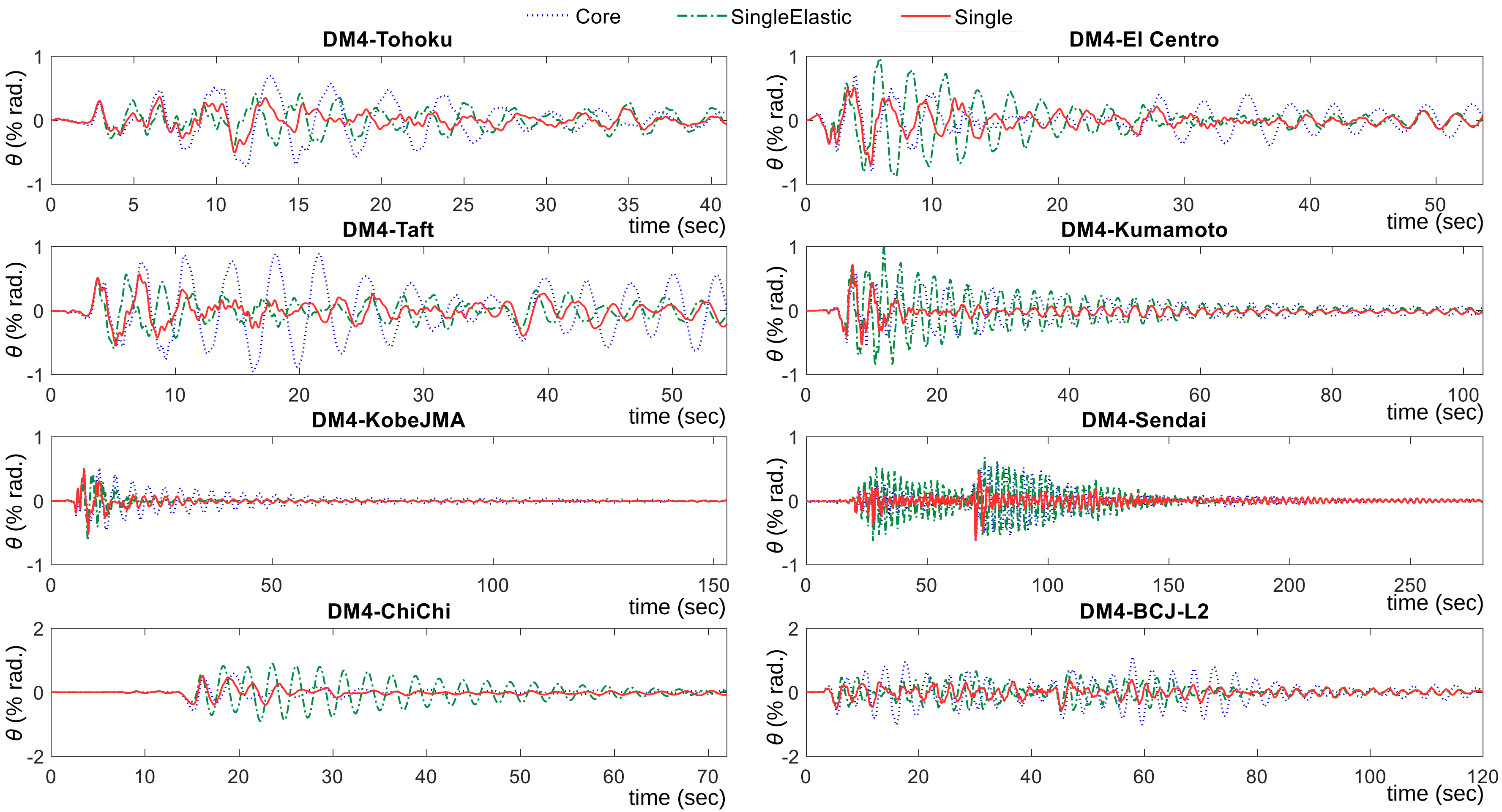


Figure 4.5.9 The roof drift history of the 32-story example model under different ground motions from NLRHA using DM4 models

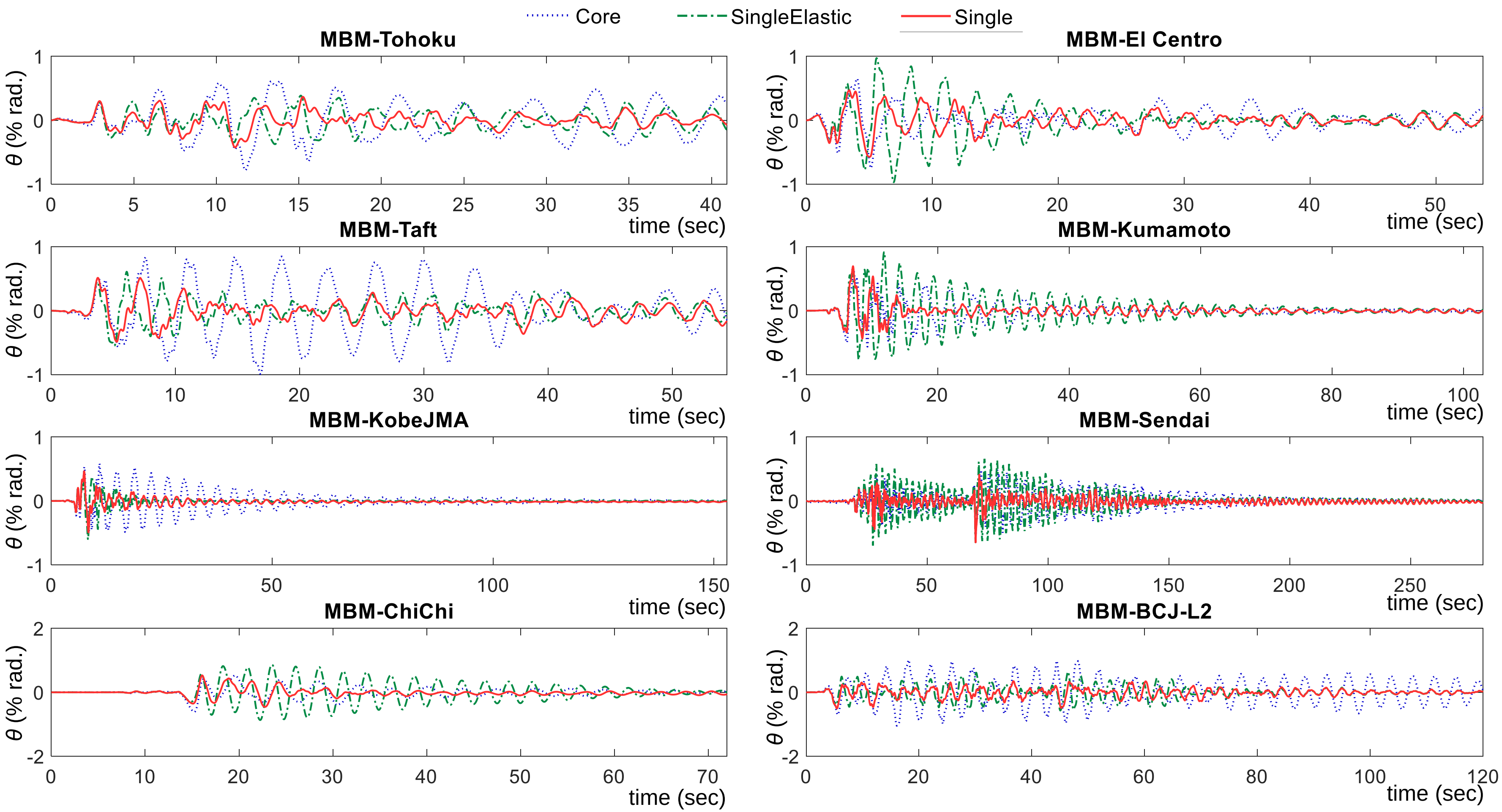


Figure 4.5.10 The roof drift history of the 32-story example model under different ground motions from NLRHA using MBM models

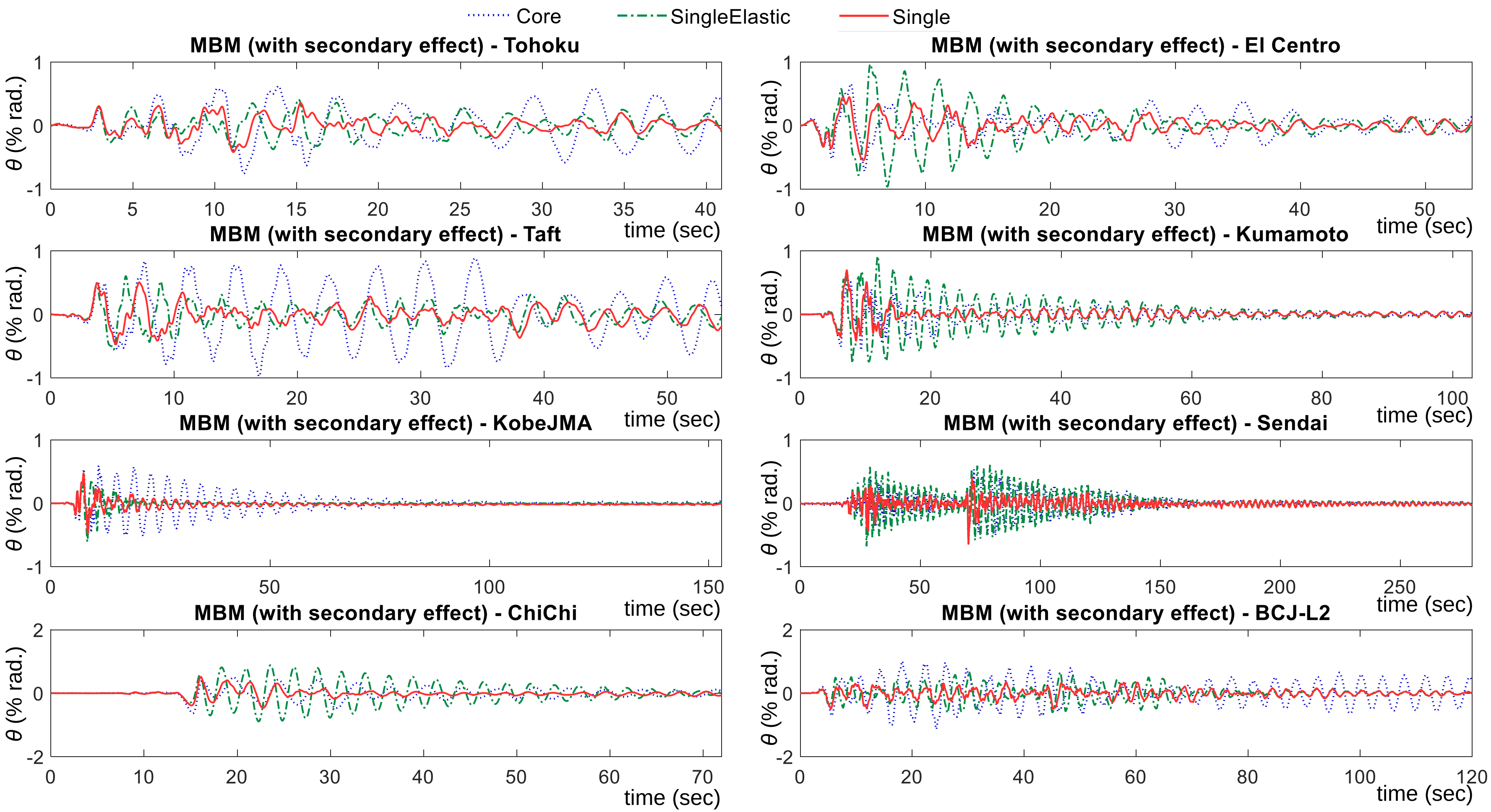


Figure 4.5.11 The roof drift history of the 32-story example model under different ground motions from NLRHA using MBM models considering the secondary effect

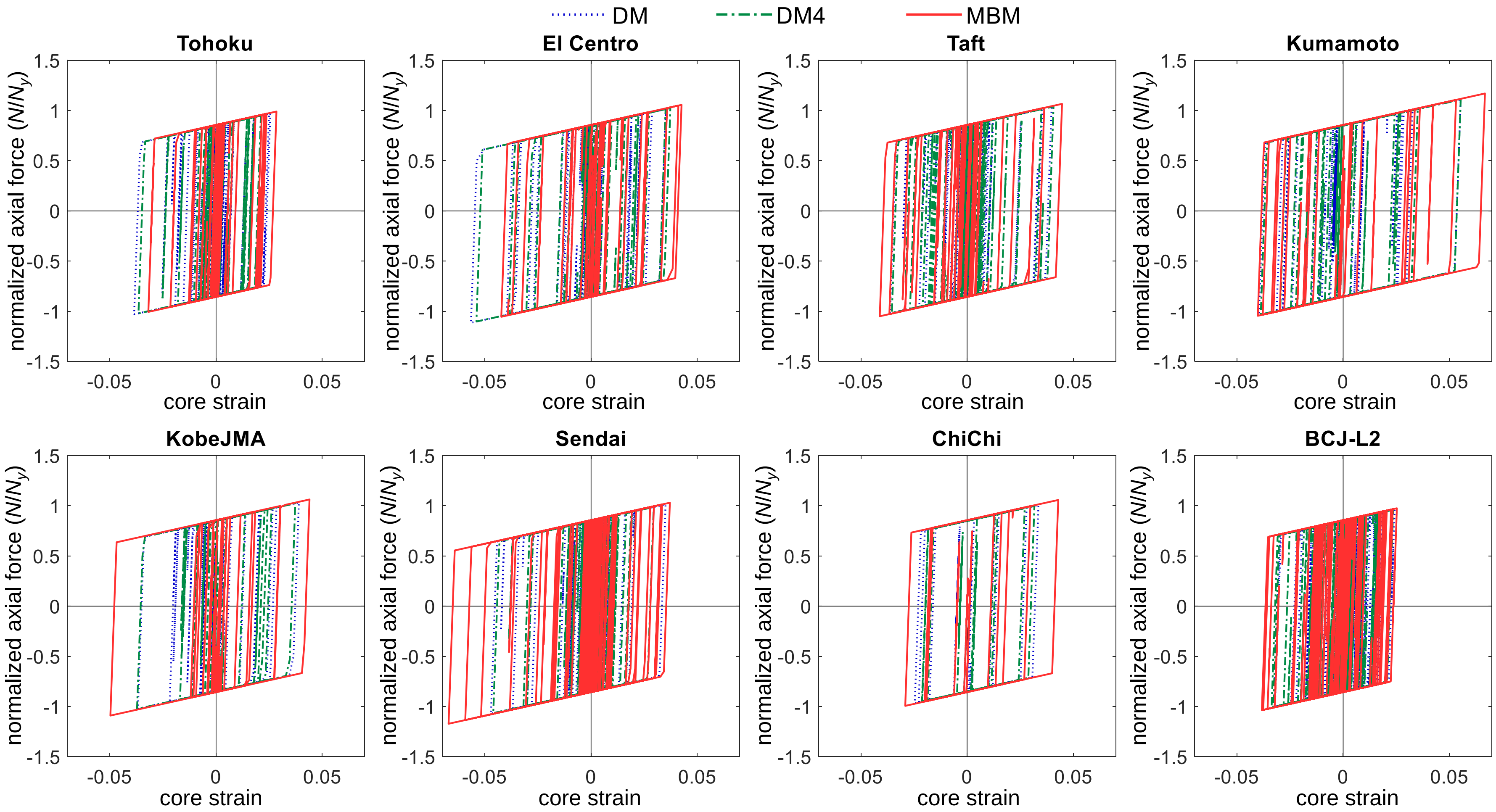


Figure 4.5.12 The relationships between BRB normalized axial force and core strain of the 32-story example model from NLRHA using DM, DM4, and MBM models

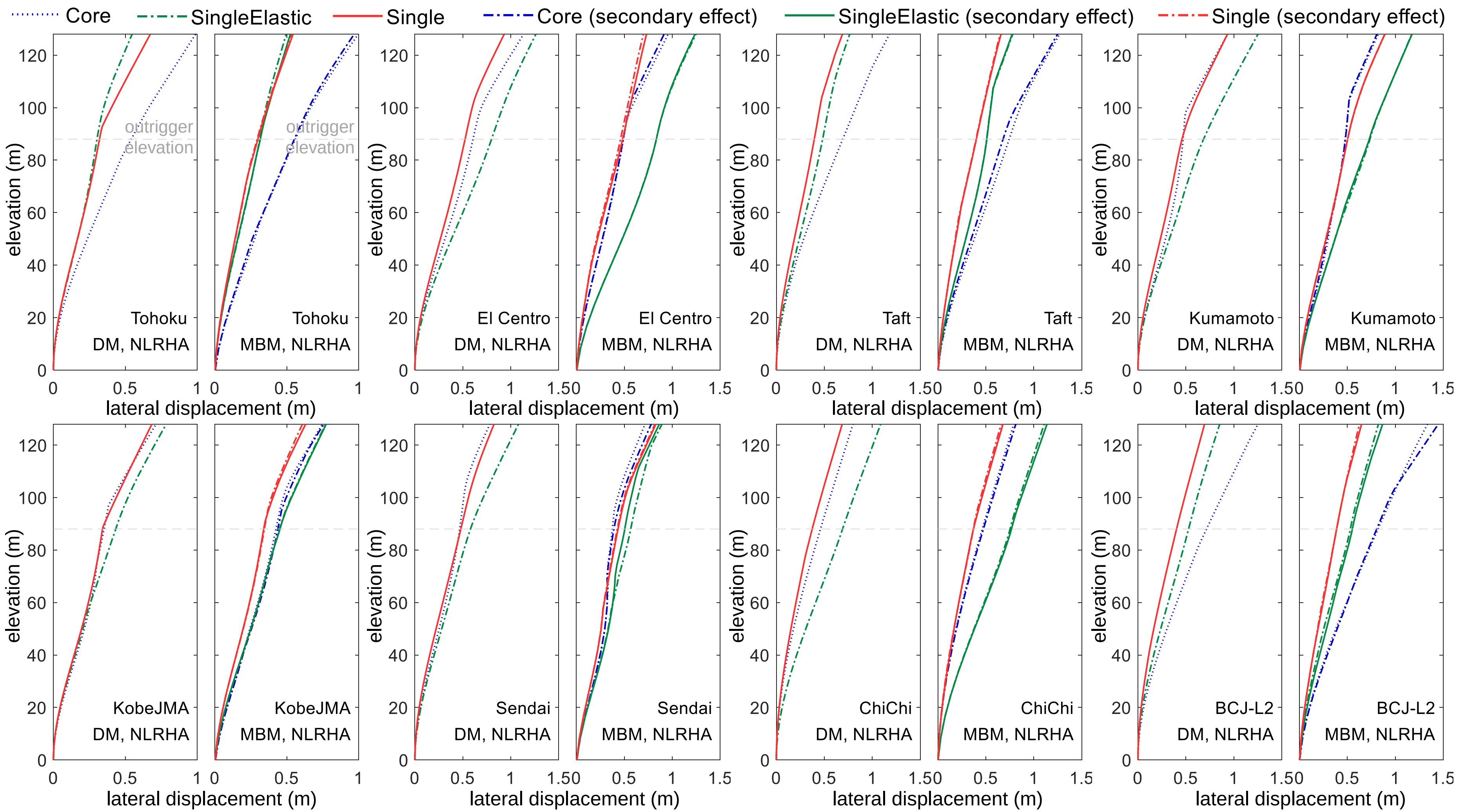


Figure 4.5.13 The maximum lateral displacement distribution of the 32-story example model from NLRHA

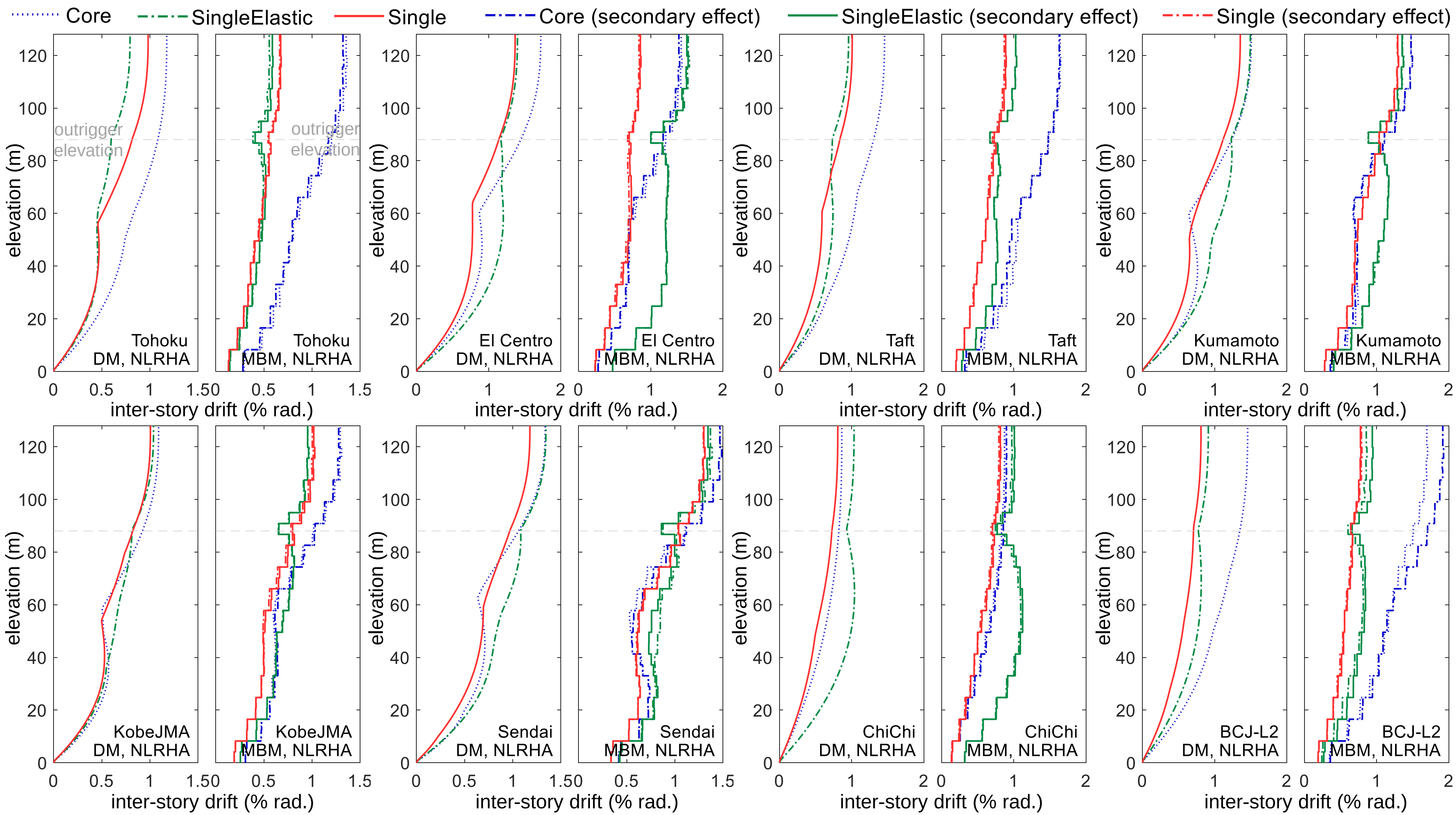


Figure 4.5.14 The maximum inter-story drift distribution of the 32-story example model from NLRHA

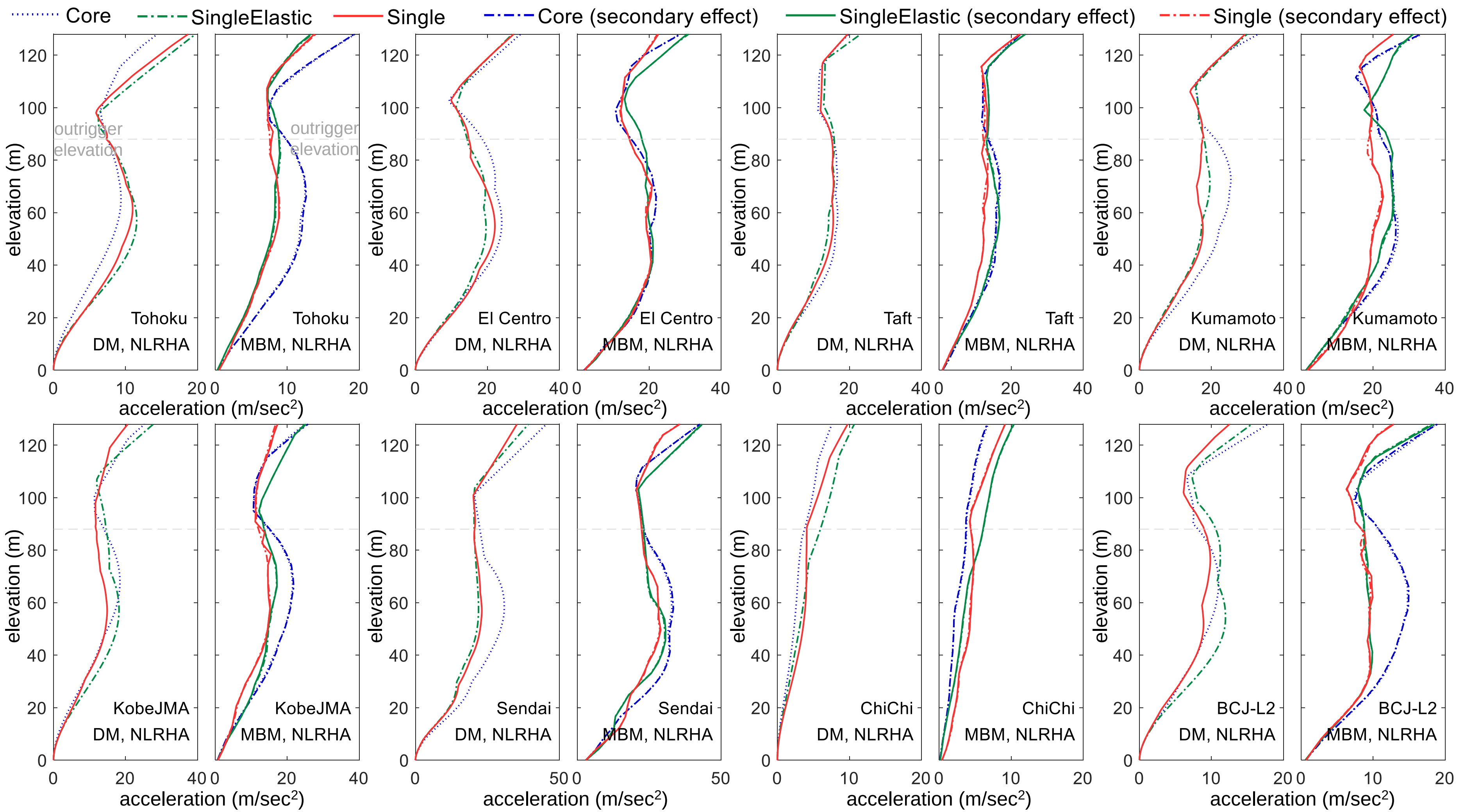
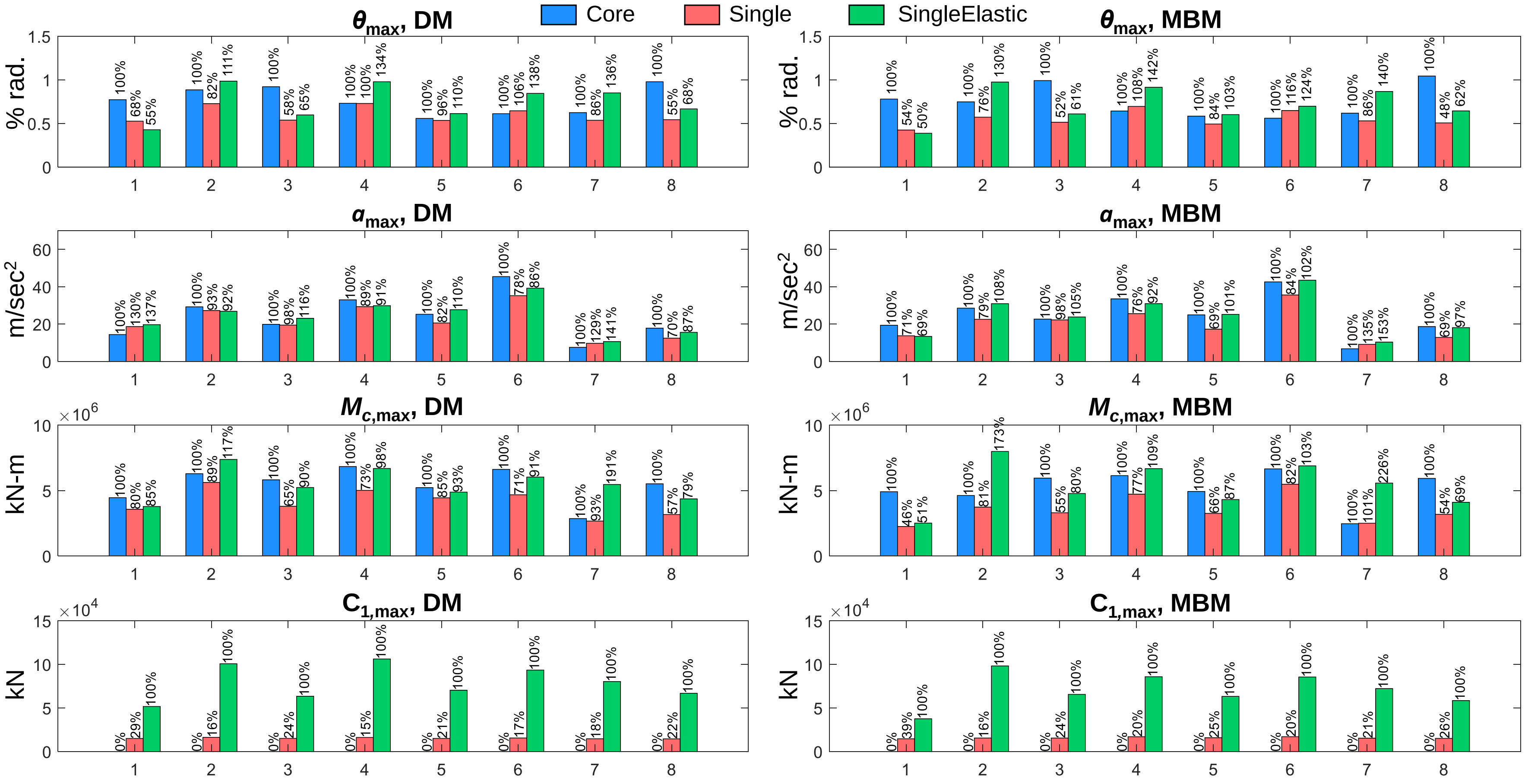


Figure 4.5.15 The maximum lateral acceleration distribution of the 32-story example model from NLRHA



Ground motion: 1=Tohoku, 2=El Centro, 3=Taft, 4=Kumamoto, 5=KobeJMA, 6=Sendai, 7=ChiChi, 8=BCJ-L2

Figure 4.5.16 The maximum seismic responses of the 32-story example model calculated from NLRHA

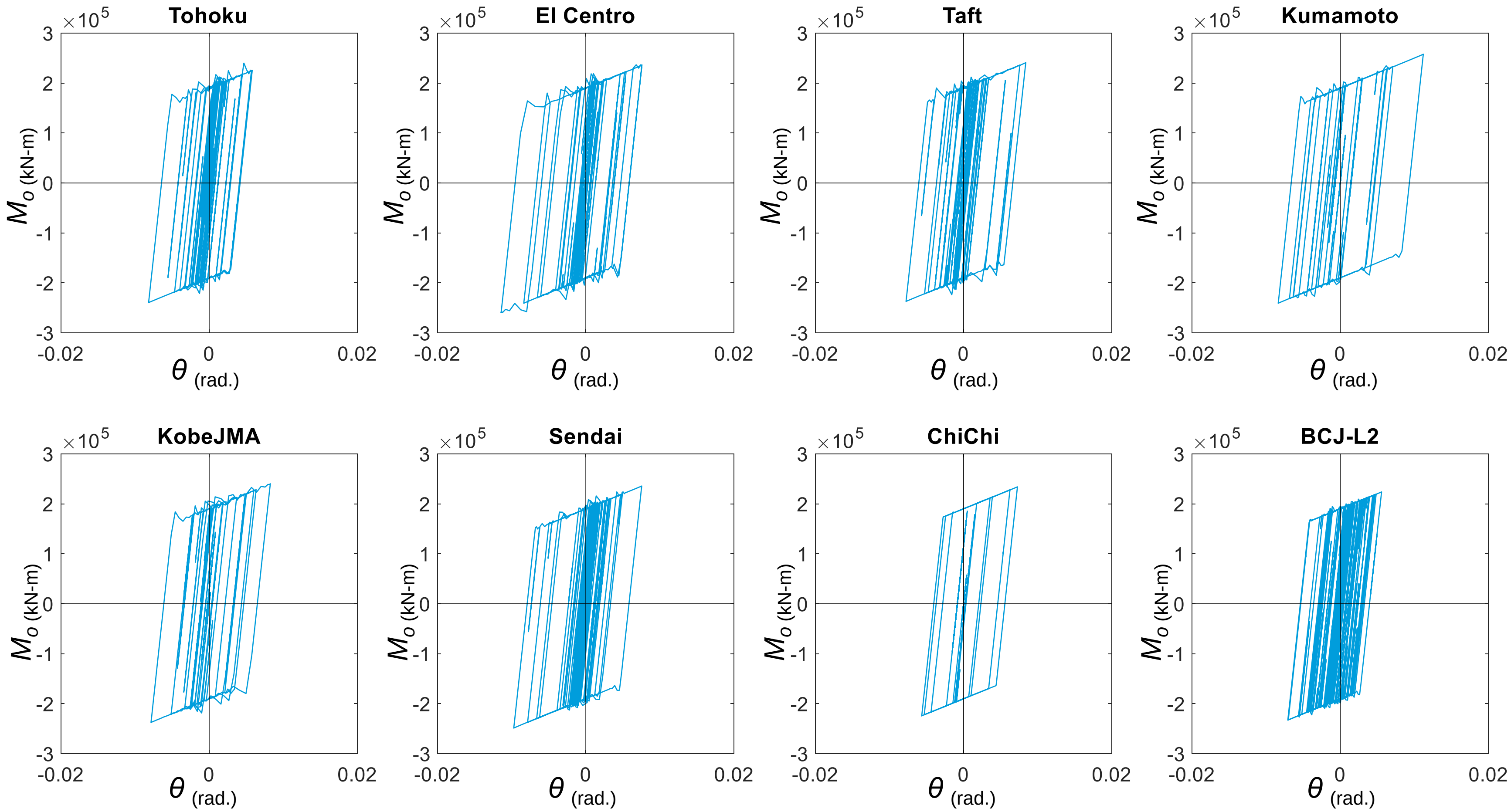


Figure 4.5.17 The relationship between M_o and θ for the 32-story Single model

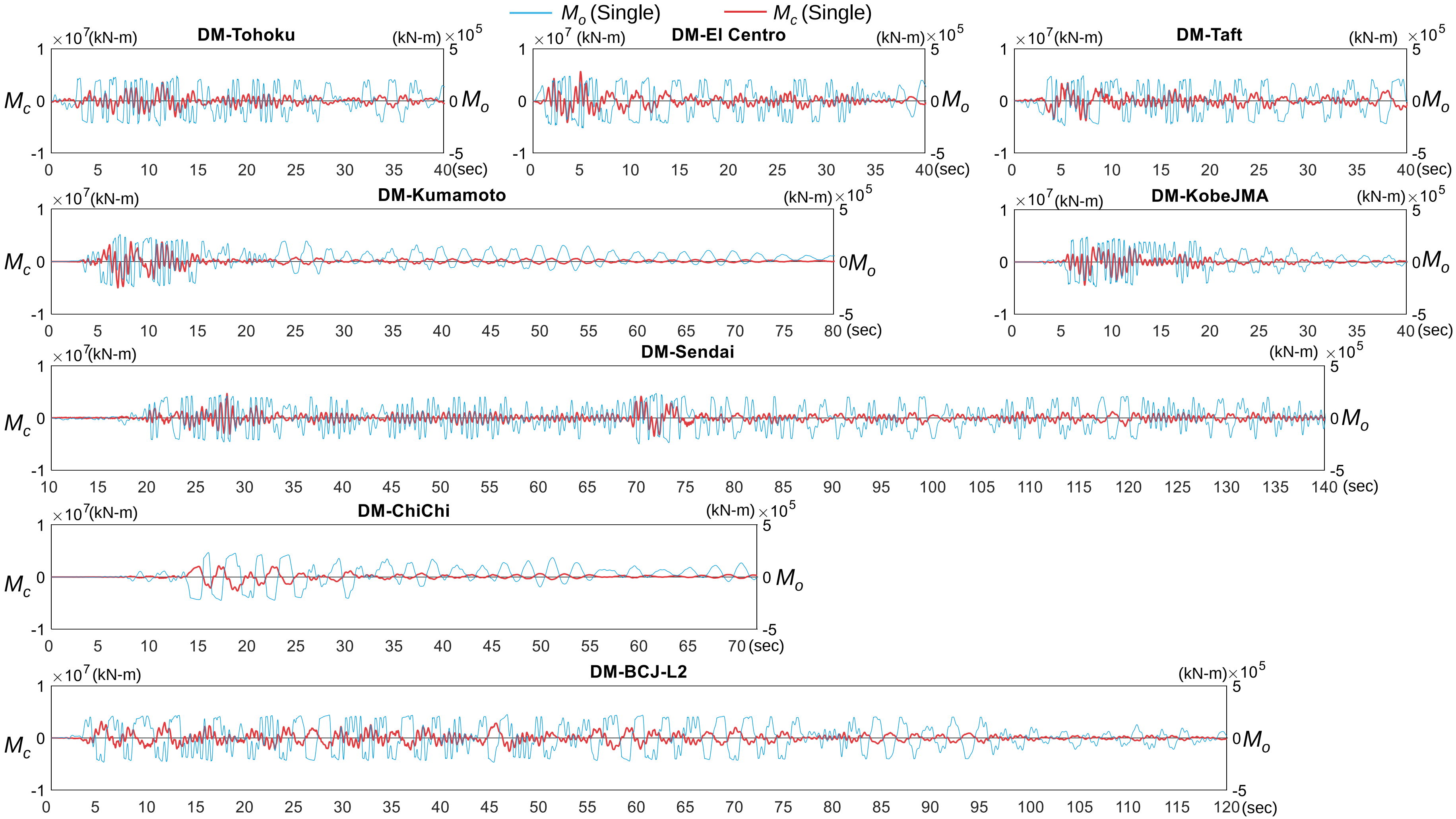
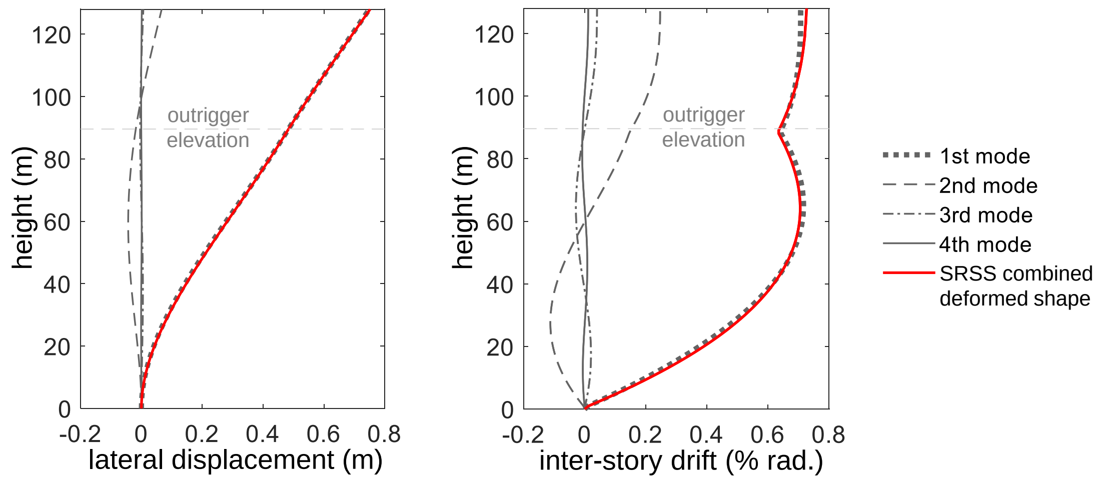


Figure 4.5.18 Overturning moment (M_c) and moment applied by the BRB-outrigger (M_o) of the 32-story Single model

Table 4.5.6 shows the SA result of the 32-story Single model. Figure 4.5.19 shows lateral displacement and inter-story drift distributions along the building height of each mode and SRSS combined deformed shape calculated from SA. The first mode dominates the overall response since the first mode has the maximum contribution in θ_{max}. The inter-story drift is significantly reduced at the outrigger elevation. The E_d/E_s value, which is the ratio of energy dissipated by BRB to the system's strain energy, shows that the energy dissipation mechanism is also governed by the first mode response. The 2nd mode response only results in slight inelastic deformation and increases a small amount of equivalent damping ratio. From the $T_{eq,n}$ (Equation (4.3.8)), the 1st and 2nd modal stiffness are decreased by approximately 41% and 0.8%, respectively. The contributions from the 3rd and 4th modes in θ_{max} are relatively small if compared with the 1st and 2nd mode responses. Therefore, the SA procedure considering the first four modes' responses should be sufficient.

Table 4.5.6 SA results of the 32-story Single model

mode	$T_{eq,n}$ (sec)	yield roof drift ratio ($y_{top,n}/h$, % rad.)	ductility, μ_n	E_d/E_s (%)	$h_{eq,n}$	θ_{max} (% rad.)
1st	3.242	0.128	4.56	52	0.086	0.585
2nd	0.517	0.023	2.29	0.8	0.021	0.052
3rd	0.199	0.429	0.01	0	0.020	0.005
4th	0.100	0.045	0.02	0	0.020	0.001



(a) **(b)**

Figure 4.5.19 (a) Lateral displacement and (b) inter-story drift distributions of each mode and SRSS combined deformed shape calculated from SA

Table 4.5.7 summarizes the maximum responses obtained from SA and the NLRHA result by using BCJ-L2 ground motion. In order to compare with the SA result, the BCJ-L2 ground motion is scaled so that its spectral acceleration at the fundamental periods of Single and SingleElastic models (T_1=2.489 sec) match with the spectral acceleration of the design spectrum (T_1 scale, (Shome and Cornell, 1998)). The scale factor is 1.35. The SA generally well agree with the NLRHA results. However, the calculations of $V_{c,\max}$ and $M_{c,\max}$ of SA are based on the elastic mode shape and linearly elastic material model (Equation (4.3.11) and Equation (4.3.12)). Therefore, the $V_{c,\max}$ and $M_{c,\max}$ obtained from SA procedure would be less accurate if compare with the NLRHA results. In addition, the close values of E_d/E_s and h_{eq} calculated from SA and NLRHA suggests that the calculation of equivalent damping ratio (Equation (4.3.5)) and the use of reduction factor (Equation (4.3.6)) are applicable.

Table 4.5.7 Maximum responses calculated from SA and NLRHA (BCJ-L2 ground motion) results of the 32-story example model

		DM			MBM		
		Core	Single Elastic	Single	Core	Single Elastic	Single
$\theta_{\max}$ (% rad.)	SA	1.12	0.78	0.59	-	-	-
	NLRHA	1.16	0.79	0.64	1.24	0.76	0.63
$\gamma_{\max}$ (% rad.)	SA	1.56	0.95	0.73	-	-	-
	NLRHA	1.73	1.08	0.97	2.02	1.02	1.06
$V_{c,\max}$ ($\times 10^4$kN)	SA	7.4	9.1	7.6	-	-	-
	NLRHA	12.4	11.2	9.6	16.0	14.6	10.4
$M_{c,\max}$ ($\times 10^6$kN-m)	SA	5.3	5.1	4.0	-	-	-
	NLRHA	6.5	5.2	3.9	7.0	4.9	4.1
E_d/E_s (%)	NLRHA	0	0	72.7	0	0	56.6
h_{eq}	SA	0.02	0.02	0.054	-	-	-
	NLRHA	0.02	0.02	0.078	0.02	0.02	0.065
$C_{1,\max}$ ($\times 10^4$kN)	NLRHA	-	7.9	1.5	-	6.9	1.6

4.5.2 Analysis results of the 96-story model

Table 4.5.8 and Figure 4.5.20 show the modal analysis results and the mode shapes of the first four modes. The vibration periods obtained from using UM, DM, and MBM models are close to each other. The 4-m mass spacing in the MBM model results in slightly longer periods than the DM and UM models. However, the marginal differences between the vibration periods and mode shapes of the 96-story model suggest that the DM model is still a good representation of the UM model. Table 4.5.8

also shows that the vibration periods of Dual model are smaller than Single models, and the vibration periods of Single models are smaller than Core models. The more decreases of vibration periods, the outrigger effect is more significant. However, the decreases of vibration periods between Core and Single models are larger than that between Dual and Single models. This indicates that the outrigger effect may not be proportional to the number of BRB-outrigger layer.

Table 4.5.8 The modal analysis results of the 96-story models

model \ mode		vibration period (sec)				mass participation ratio (%)			
		1st	2nd	3rd	4th	1st	2nd	3rd	4th
UM	Core	8.427	1.345	0.480	0.245	65.7	20.2	6.9	3.5
	Single	7.786	1.325	0.480	0.245	66.5	19.4	6.9	3.5
	Dual	7.614	1.310	0.480	0.243	66.2	19.9	6.8	3.5
DM	Core	8.449	1.348	0.481	0.246	65.7	20.2	6.9	3.5
	Single	7.804	1.329	0.481	0.245	66.5	19.4	6.9	3.6
	Dual	7.631	1.313	0.481	0.244	66.2	19.9	6.8	3.5
MBM	Core	8.485	1.361	0.486	0.248	65.8	20.2	6.9	3.5
	Single	7.835	1.341	0.486	0.248	66.6	19.4	6.9	3.6
	Dual	7.658	1.325	0.485	0.246	66.2	19.9	6.8	3.5

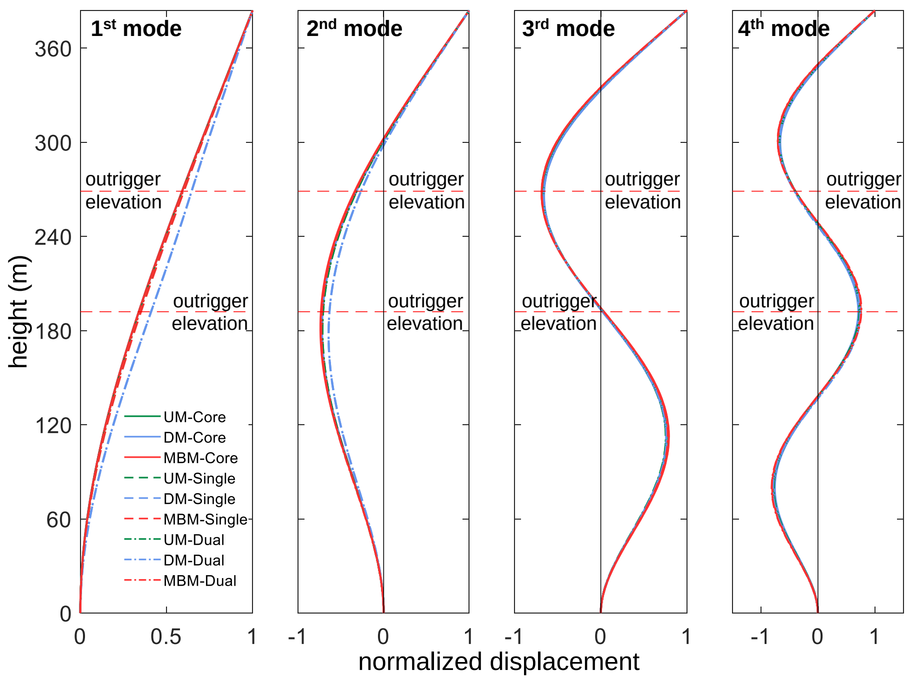


Figure 4.5.20 The first four mode shapes of the 96-story example mode

Table 4.5.9 and Figure 4.5.21 show the scale factors and the scaled spectra of the ground motions used in the NLRHA. Figure 4.5.22 and Figure 4.5.23 show the roof drift (θ) histories under each NLRHA calculated from using DM and MBM models, respectively. Figure 4.5.24 shows the normalized axial force (N/N_y) and core strain responses of the BRB in the Single model calculated from each NLRHA by using DM and MBM models. Figure 4.5.25 and Figure 4.5.26 show the normalized axial force and core strain responses of BRB_2 and BRB_1, respectively, in the Dual model under each NLRHA by using DM and MBM models. In the Dual model, as the core structure rotation demand in the 1st mode deformed shape is smaller at the lower outrigger elevation, the deformations of BRB_1 are smaller than the BRB_2. The very similar responses calculated from using DM and MBM models suggest that the DM model with 1-m long BRB elements well resemble the BRBs in Dual model even when the BRB lengths are longer than one story height. Figure 4.5.27, Figure 4.5.28, and Figure 4.5.29 show the maximum lateral displacement, maximum inter-story drift, and maximum lateral acceleration distributions throughout the building height, respectively. As shown in Figure 4.5.29, the reduction in acceleration response of the 96-storu Dual model is only slightly smaller than the 32-story Single model. This can be due to the 96-story has smaller $S_{bc2,07}$ value (0.26) than the 32-story Single model ($S_{bc,07}$=1.38). Figure 4.5.30 shows the maximum seismic responses calculated from NLRHA. The numbers shown in Figure 4.5.30 indicate the ratio of peak response if compared with the Core model ($\theta_{\max}$, $a_{\max}$, and $M_{c,\max}$) or compared with the DualElastic model ($C_{1,\max}$ and $C_{2,\max}$) under each ground motion. The maximum responses obtained from using DM model well agree with the results calculated from using MBM model. In addition, the marginal differences between analysis results from DM and MBM models suggest the DM model can be used to represent the MBM model for the structure with dual BRB-outrigger system. In summary, based on the peak responses shown in Figure 4.5.30, the BRB-outrigger system is efficient in reducing $\theta_{\max}$, $M_{c,\max}$, and $C_{1,\max}$ responses. The greater seismic intensity generally results in greater reductions in $\theta_{\max}$ and $M_{c,\max}$ because of greater BRB energy dissipation. The acceleration response can be amplified in the SingleElastic and DualElastic models, but can be reduced slightly or remain similar to the Core model when BRB is incorporated.

Since the BRB is modeled using bilinear truss element, the relationship between the moments applied by the upper (M_{o2}) and lower BRB-outrigger (M_{o1}) on core structure and the core structure rotations at the upper (θ_2) and lower (θ_1) outrigger elevations are bilinear hysteretic responses. Figure 4.5.31 and Figure 4.5.32 show the relationship between M_{o2} and θ_2, M_{o1} and θ_1 of the 96-story example Dual model obtained using DM model. Similar to the 32-story Single model example, the bilinear hysteretic response shown in Figure 4.5.31 and Figure 4.5.32 cause different phases during an earthquake (Figure 4.5.33). As the number of BRB-outrigger increases, the high frequency-like response of overturning moment (M_c) in the 96-story Dual model is severer than the 32-story Single model. However, the peak overturning moment in the 96-story Dual model is still smaller than the peak value in the Core model.

Table 4.5.9 The scale factors of the ground motions used for NLRHA

Tohoku	El Centro	Taft	Kumamoto	KobeJMA	Sendai	ChiChi	BCJ-L2
3.73	3.58	5.54	2.75	2.37	2.51	0.67	1.19

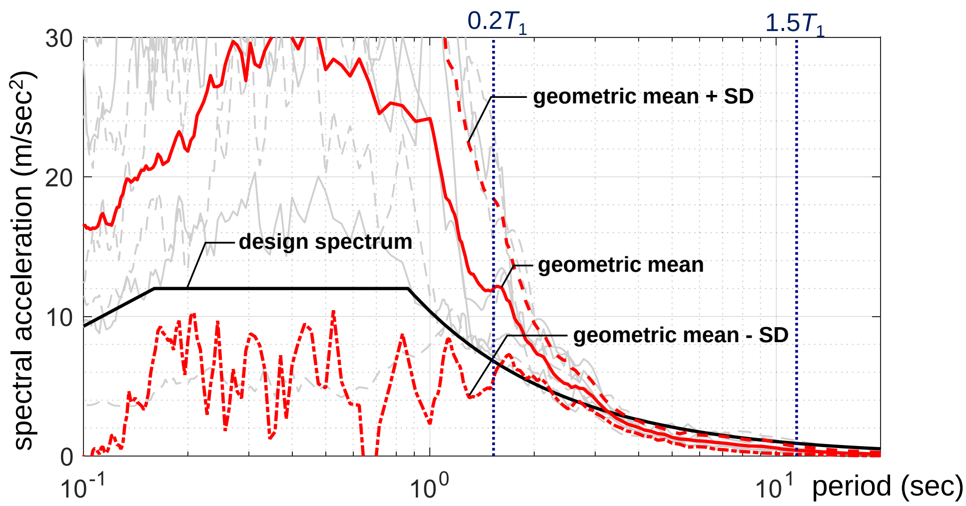


Figure 4.5.21 Spectra of the scaled ground motions used for 96-story example NLRHA (SD = standard deviation)

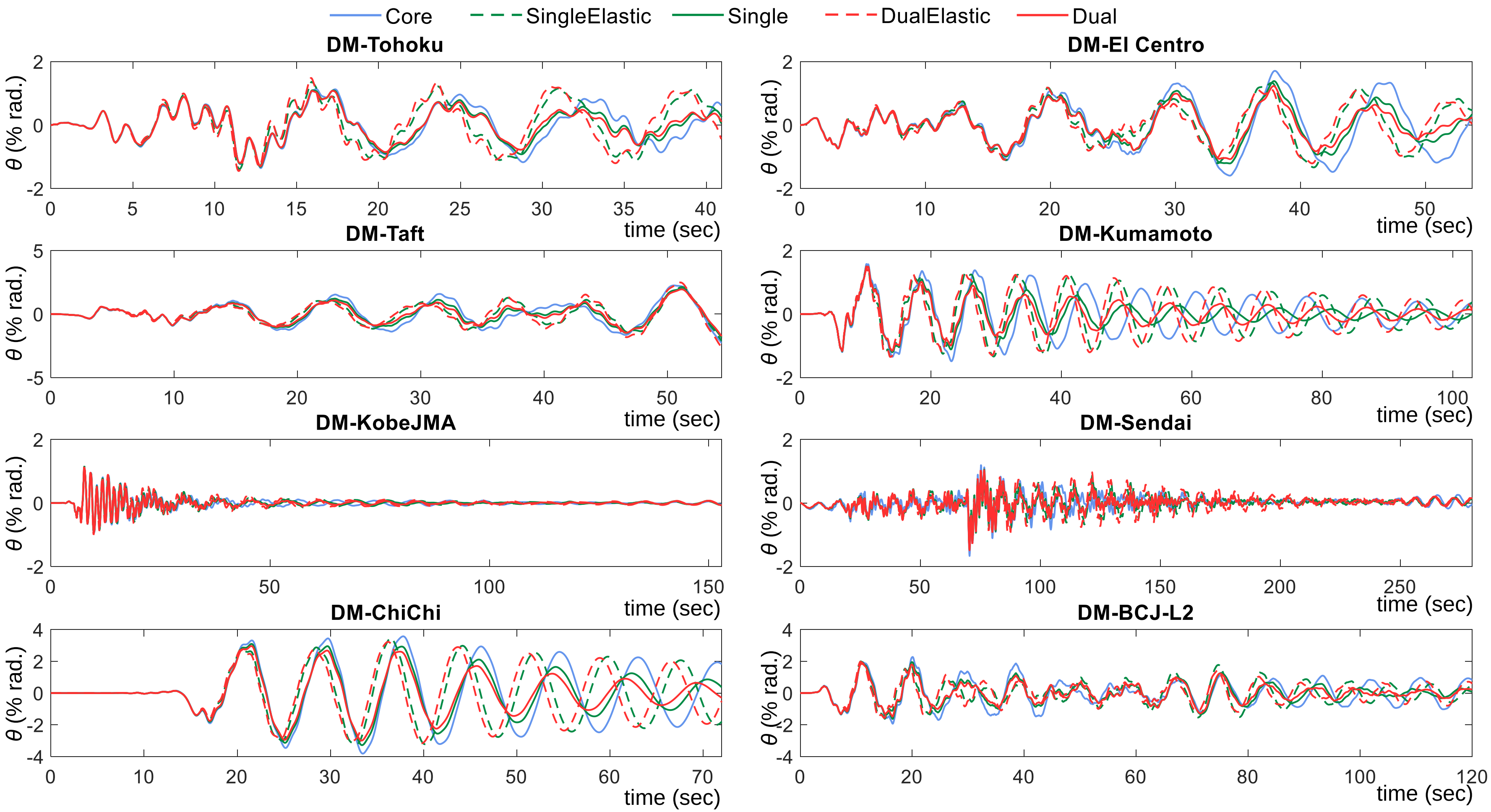


Figure 4.5.22 The roof drift history of the 96-story example model under different ground motions calculated from NLRHA using DM models

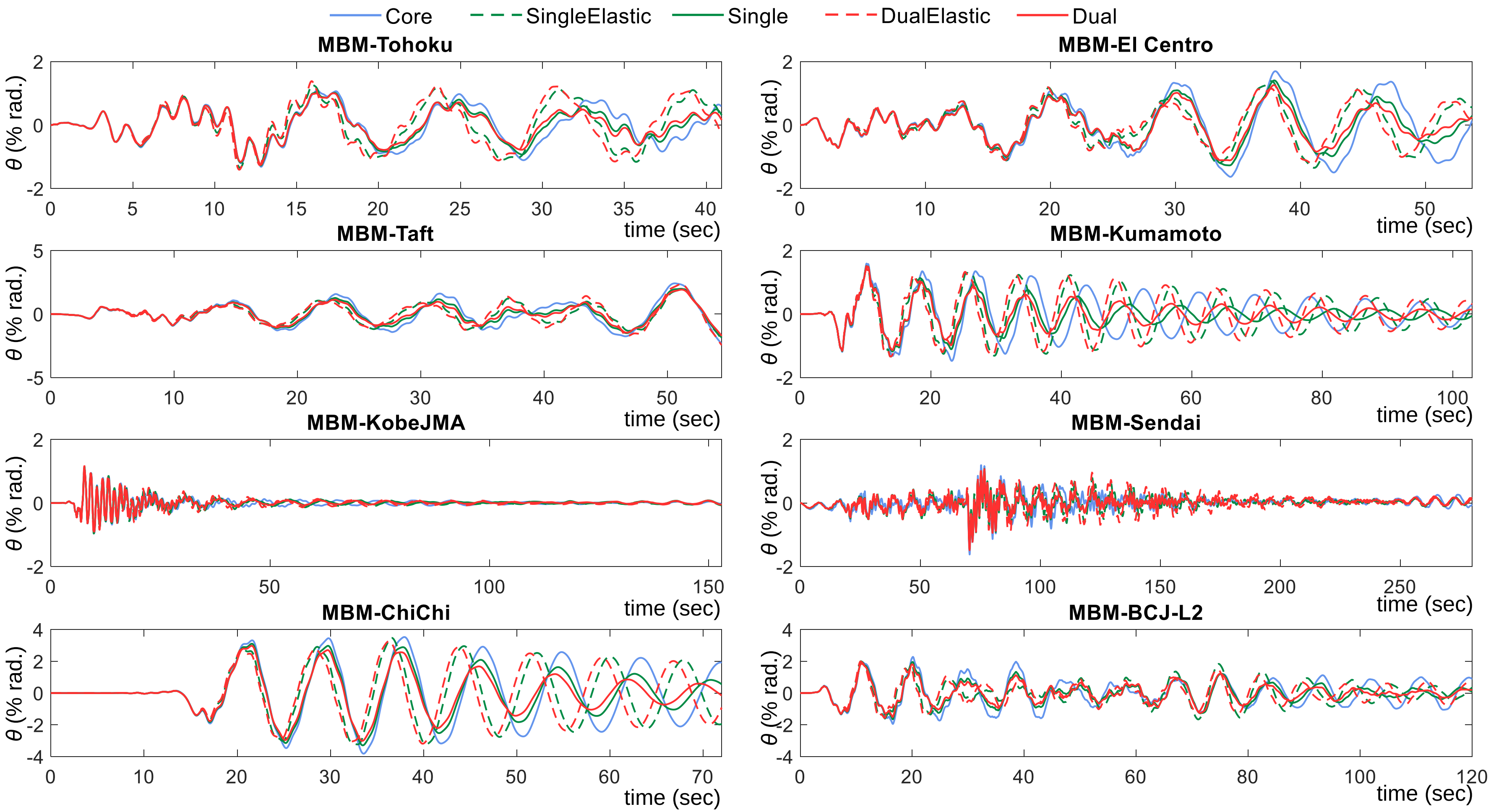


Figure 4.5.23 The roof drift history of the 96-story example model under different ground motions calculated from NLRHA using MBM models

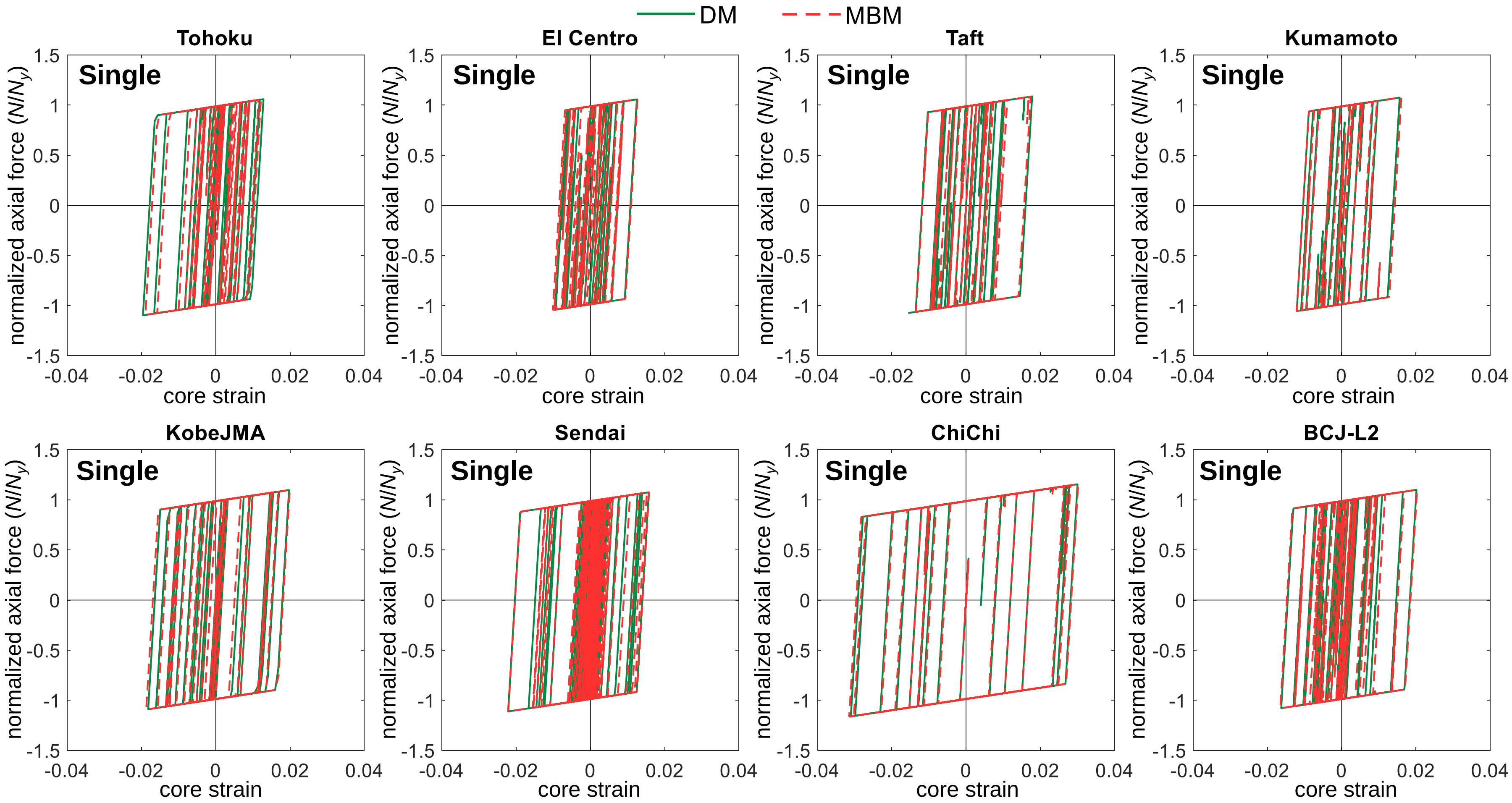


Figure 4.5.24 The relationships between BRB normalized axial force and core strain of the 96-story Single example model calculated from NLRHA using DM and MBM models

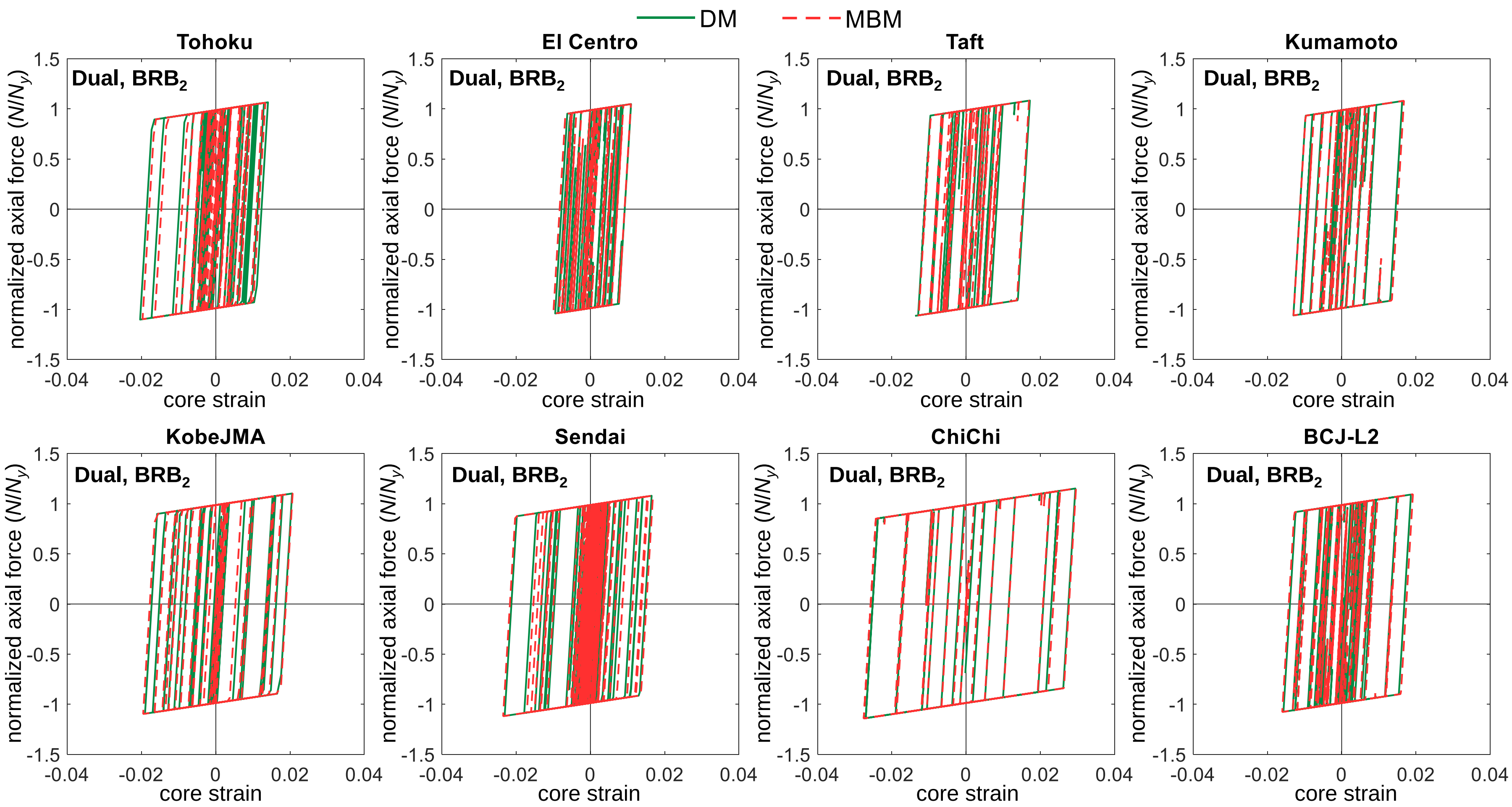


Figure 4.5.25 The relationships between BRB_2 normalized axial force and core strain of the 96-story Dual example model calculated from NLRHA using DM and MBM models

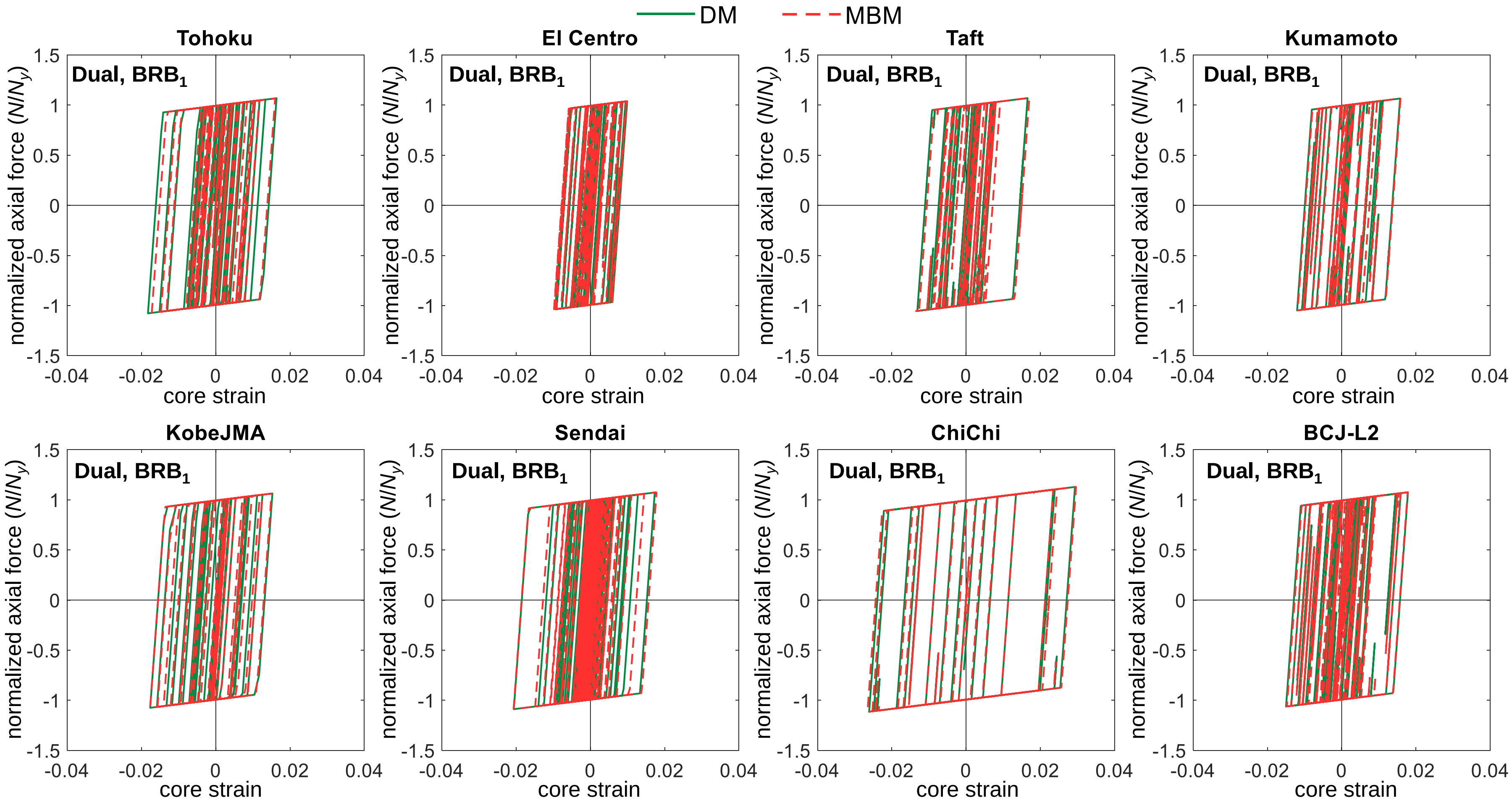


Figure 4.5.26 The relationships between BRB_1 normalized axial force and core strain of the 96-story Dual example model calculated from NLRHA using DM and MBM models

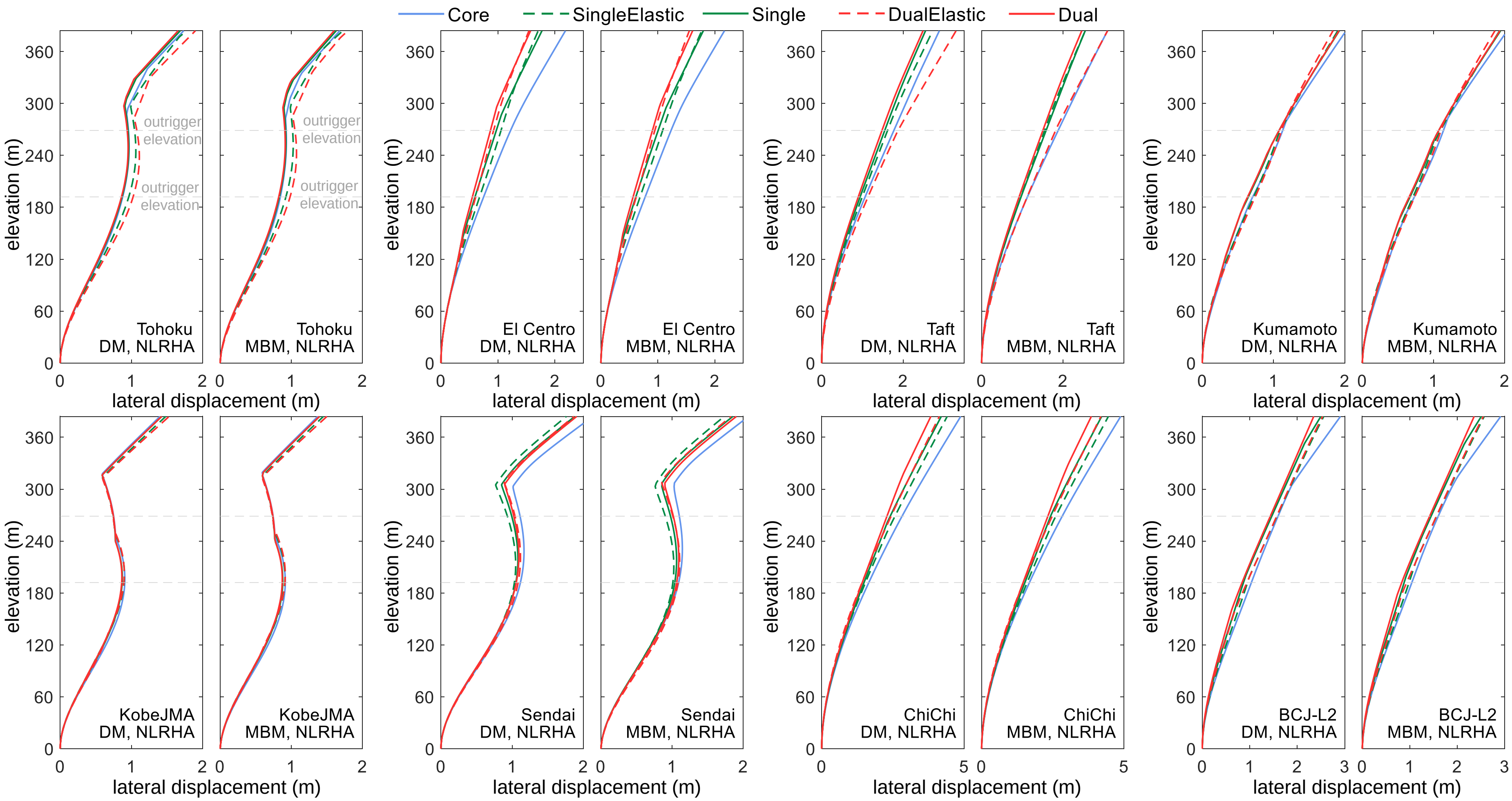


Figure 4.5.27 The maximum lateral displacement distribution of the 96-story example model from NLRHA

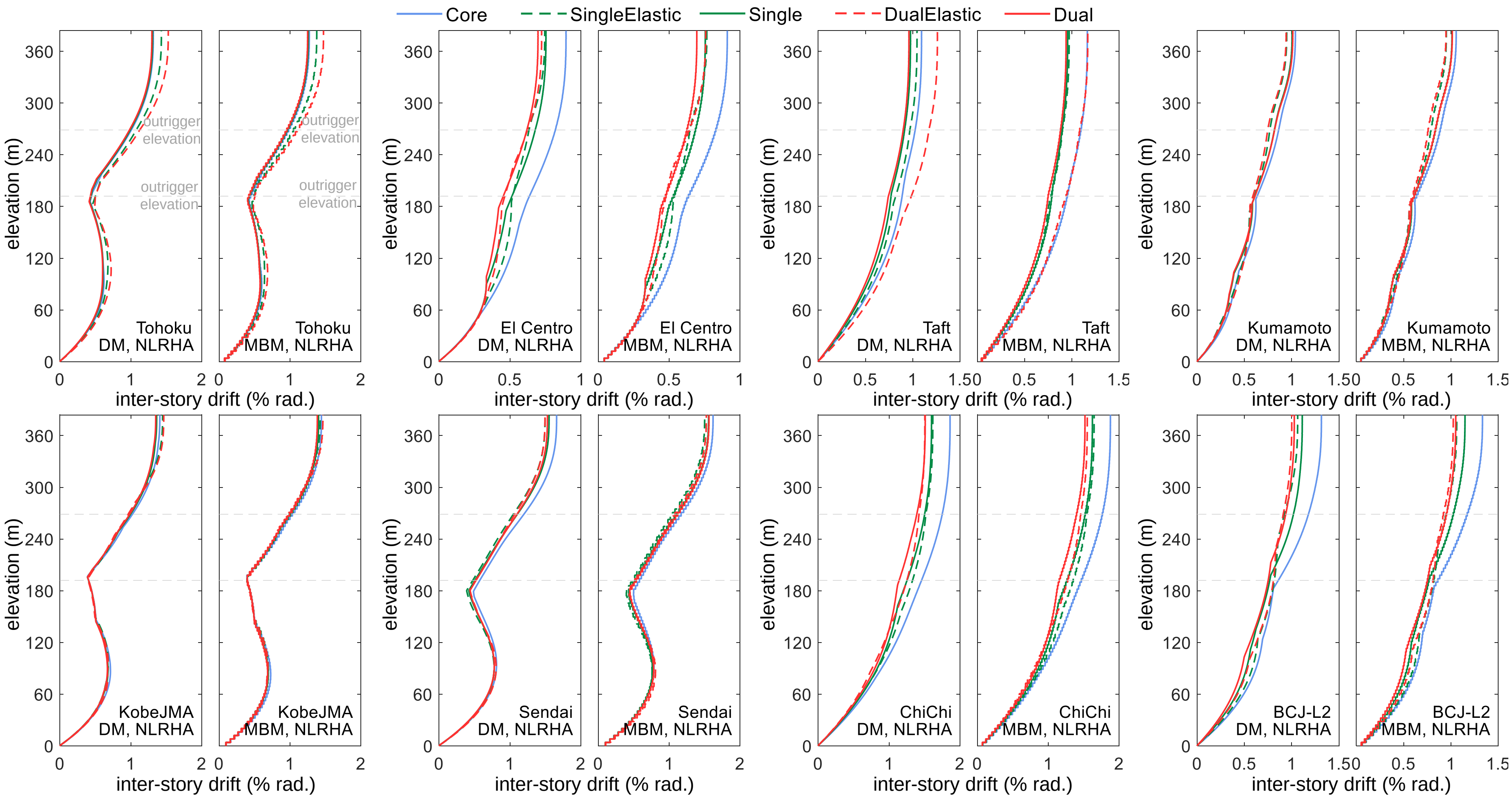


Figure 4.5.28 The maximum inter-story drift distribution of the 96-story example model from NLRHA

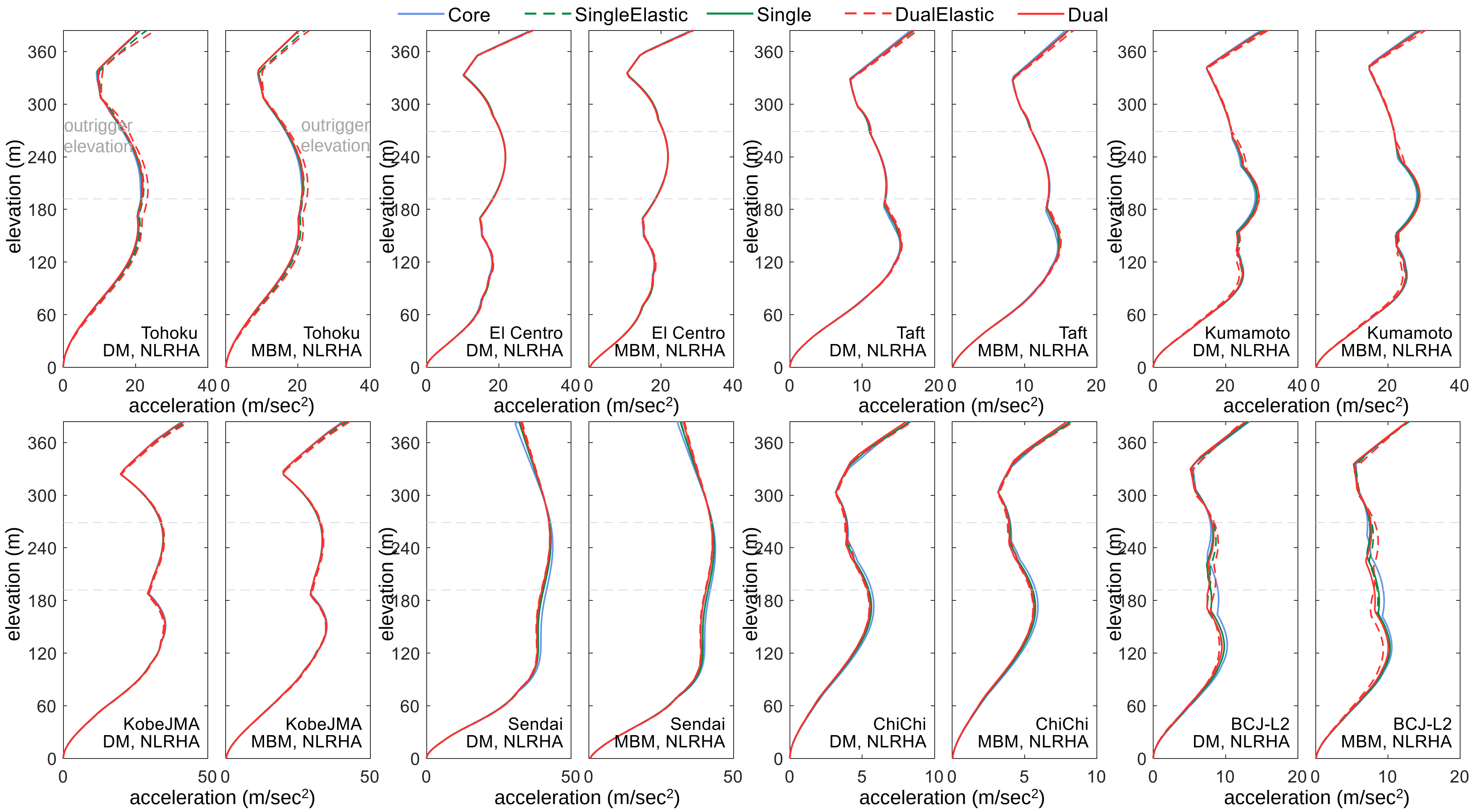
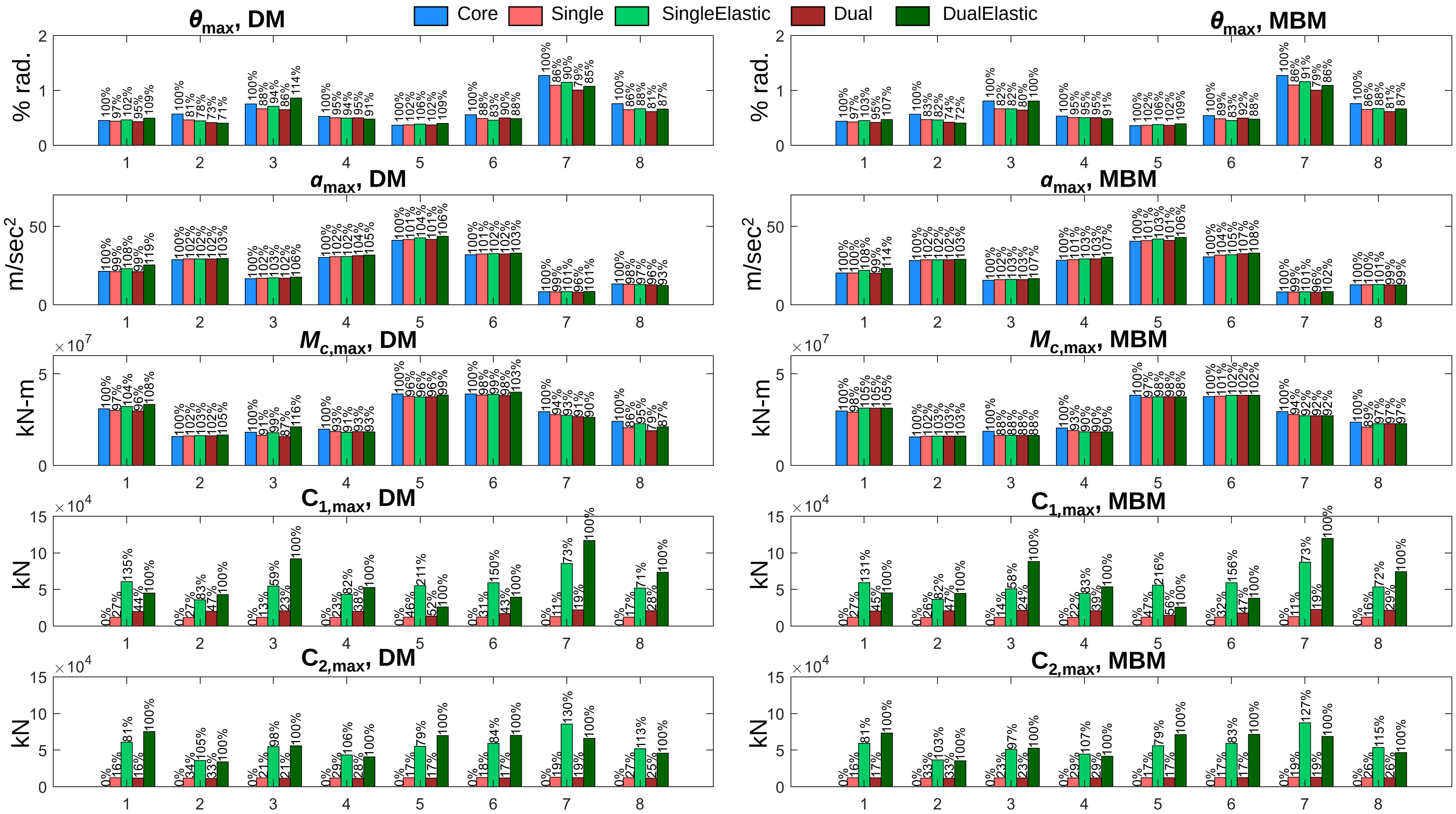


Figure 4.5.29 The maximum lateral acceleration distribution of the 96-story example model from NLRHA



Ground motion: 1=Tohoku, 2=El Centro, 3=Taft, 4=Kumamoto, 5=KobeJMA, 6=Sendai, 7=ChiChi, 8=BCJ-L2

Figure 4.5.30 The maximum seismic responses of the 96-story example model from NLRHA

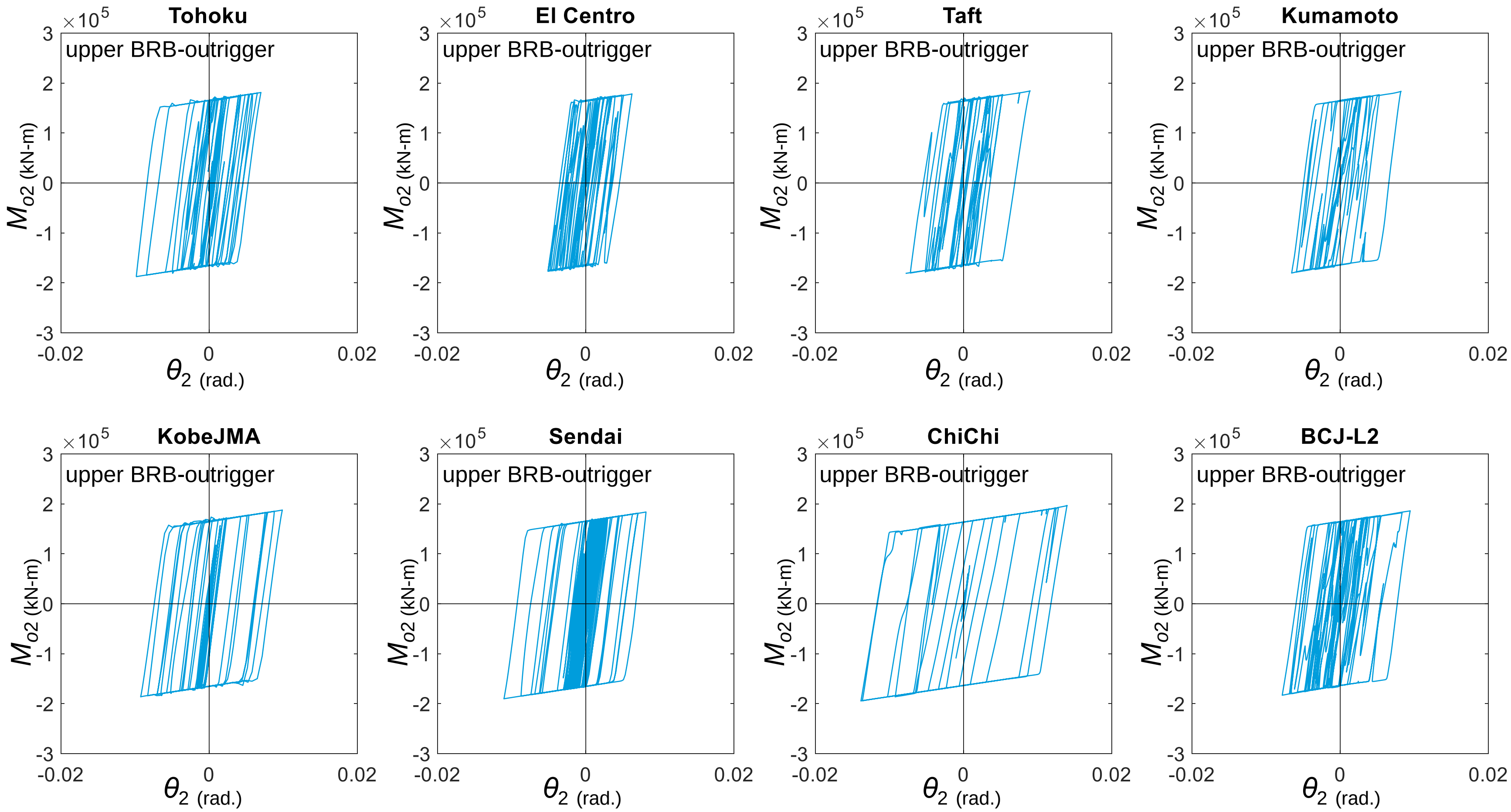


Figure 4.5.31 The relationship between M_{o2} and θ_2 for the 96-story Dual model

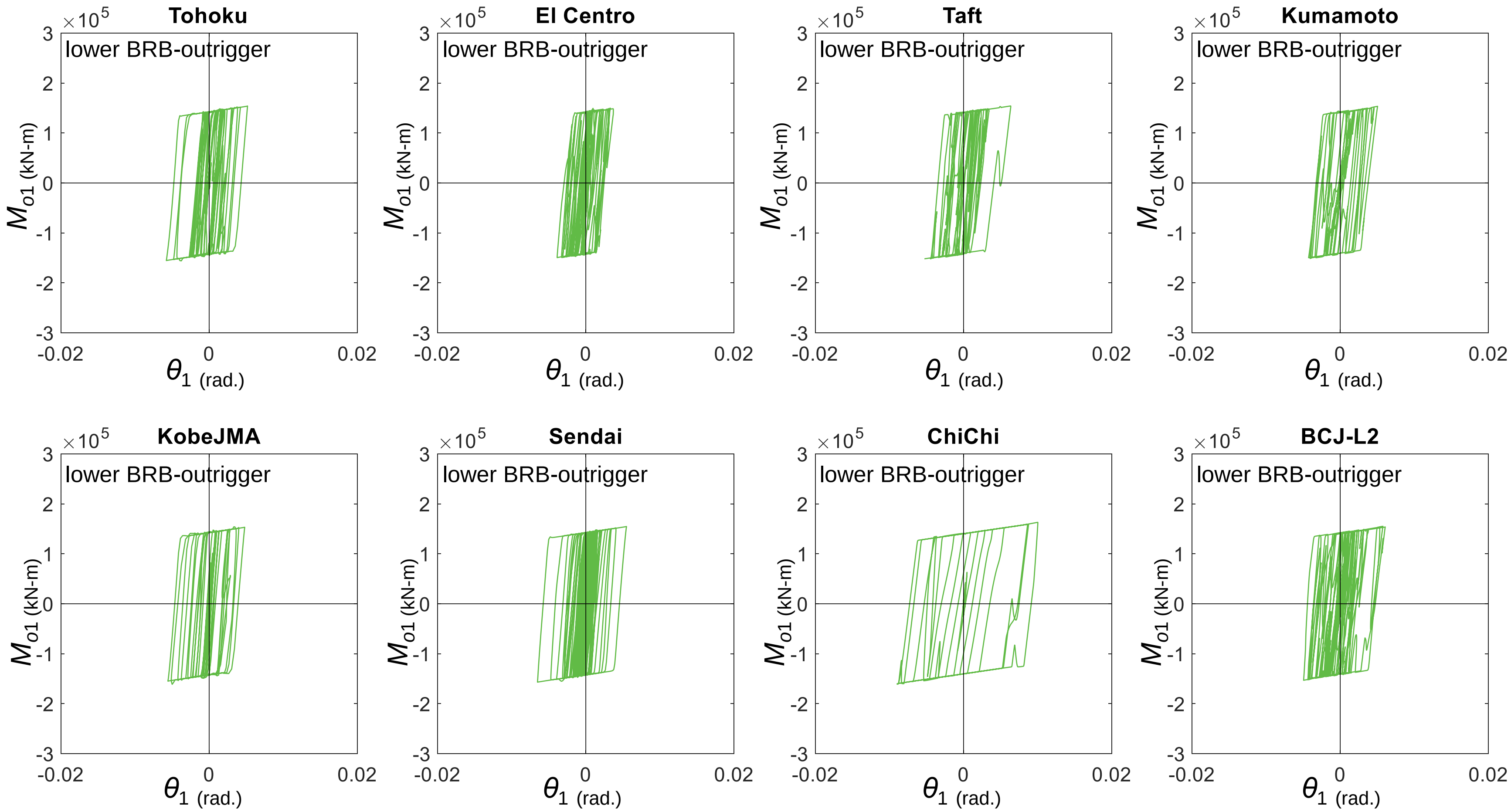


Figure 4.5.32 The relationship between M_{o1} and θ_1 for the 96-story Dual model

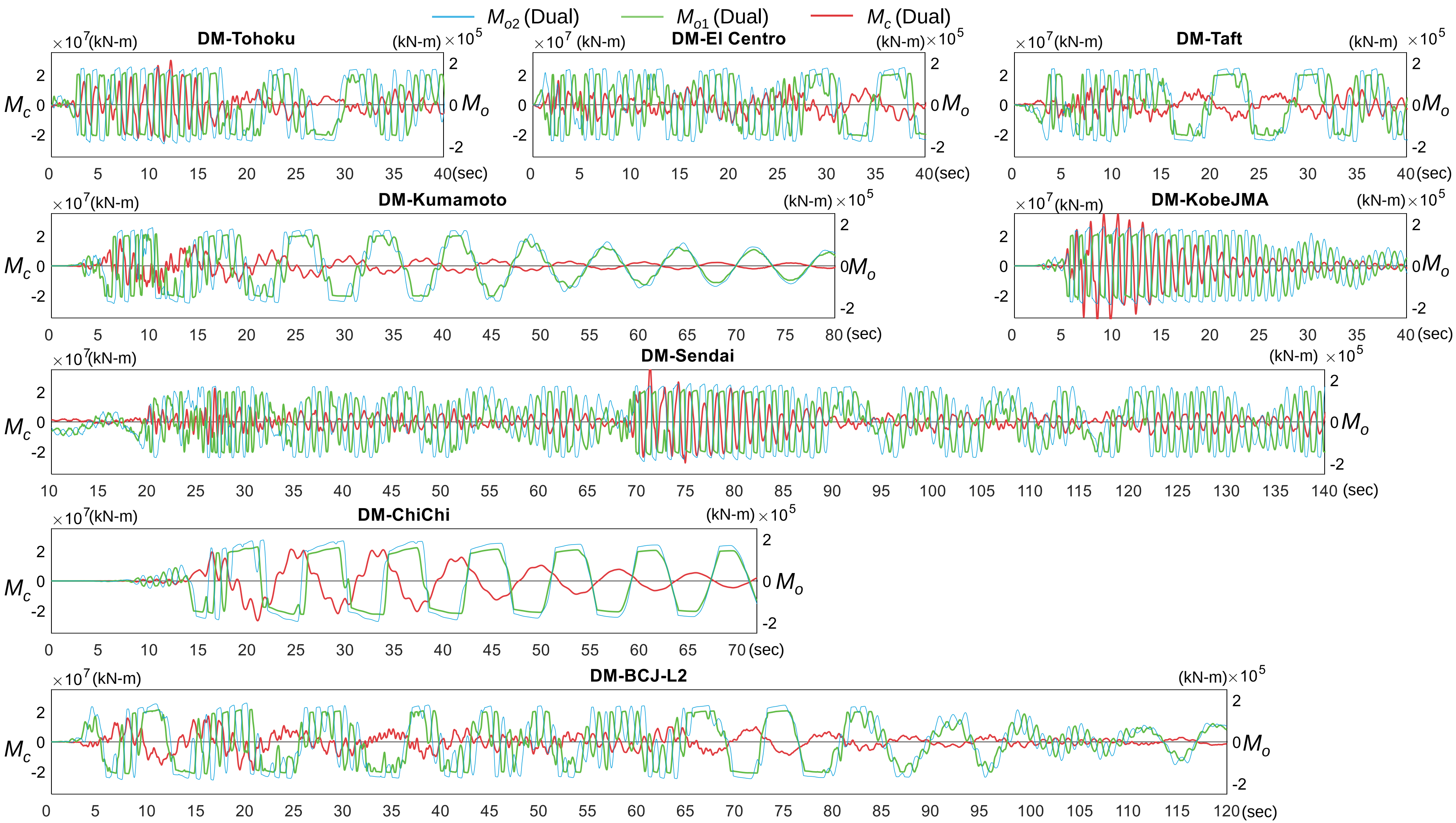


Figure 4.5.33 The maximum seismic responses of the 96-story example model from NLRHA

Table 4.5.10 and Figure 4.5.34 show the SA results and the MPA curves of the 96-story Dual model when the roof displacement reaches $y'_{\max,n}$ (Equation (4.3.7)). The MPA curves indicate that the BRB_1 and BRB_2 yield almost simultaneously in the 1st and 2nd mode responses. The 1st and 2nd modal stiffness decrease by approximately 18% and 21%, respectively. The 1st mode dominates the overall response, as it has the maximum contribution in $\theta_{\max}$. The E_d/E_s value also suggests that the 1st mode response almost govern the energy dissipation mechanism. The 2nd mode response only results in slight inelastic deformation and slightly increases the equivalent damping ratio. If compare with the 32-story model, the inelastic response contributed from the 2nd mode increases. This indicates that inelastic responses contributed from the 2nd mode responses would be more significant in the taller structure that has greater 2nd mode mass participation ratio. In addition, the contributions from the 3rd and 4th modes in $\theta_{\max}$ are relatively small if compared to the 1st and 2nd modes. The SA procedure considering the first four modes' responses should be sufficient for the 96-story models. Table 4.5.11 shows the maximum responses calculated from SA and NLRHA by using BCJ-L2 ground motion. The scale factor (T_1 scale, (Shome and Cornell, 1998)) for the 96-story model is 1.44. As shown from the roof drift histories (Figure 4.5.22 and Figure 4.5.23) and the maximum lateral deformation throughout the building height (Figure 4.5.27), the Core models exhibit greater $\theta_{\max}$ responses than the others in most of the NLRHA. The SingleElastic and DualElastic models, which result in a stiffer system, also exhibit larger $\theta_{\max}$ than Core in some NLRHA. The Single and Dual models with the BRBs' yielding and energy dissipation mechanisms are more efficient in reducing the roof drift responses after the $\theta_{\max}$ occurs. The Dual models generally perform better than the Single models. In addition, as indicated from the maximum inter-story drift distributions (Figure 4.5.28), the inter-story drift at the elevations close to outrigger can be effectively reduced. The Dual and DualElastic models generally perform better than the Single and SingleElastic models, and the Dual models with BRBs' energy dissipation mechanism perform better than the DualElastic models in mitigating $\theta_{\max}$ and $\gamma_{\max}$. As shown in Table 4.5.11, the SA well estimates the maximum roof drift ($\theta_{\max}$) responses. The underestimations in the maximum base shear ($V_{c,\max}$) and overturning moment ($M_{c,\max}$) by the SA are because the reaction forces are calculated based on the linearly elastic force-deformation relationships and elastic mode shapes. The E_d/E_s values for the BRB_1 (E_{d1}/E_s) and BRB_2 (E_{d2}/E_s), and the corresponding equivalent damping ratios (h_{eq}) are also presented in Table 4.5.11. The E_d/E_s values are zero in the SingleElastic and DualElastic models because the BRB elements deform elastically. The h_{eq} estimated by the SA are similar to the NLRHA results. This suggests that the SA procedure that uses the equivalent damping ratio could properly evaluate the energy dissipation performance of a structure with BRB-outrigger systems. Based on the analysis results, the deformation-related seismic performance indicators ($\theta_{\max}$ and $\gamma_{\max}$) calculated from the SA and NLRHA, the overturning moment at core structure base ($M_{c,\max}$), and the maximum perimeter column axial force ($C_{1,\max}$) obtained from the NLRHA results are adopted as seismic performance indicators for the purpose of parametric study.

As indicated in Table 4.5.11, the value of $\theta_{\max}$ can be reduced by approximately 10% by the SingleElastic and DualElastic models. If the BRB-outrigger system is applied, the reductions in $\theta_{\max}$ can be increased to approximately 12% and 18% for the Single and Dual models, respectively. The conventional outrigger models (SingleElastic and DualElastic) reduce $\theta_{\max}$ by increasing the system stiffness, however, they could also increase the spectral acceleration demand. In the 96-story model, the maximum inter-story drift ($\gamma_{\max}$) occurs at the top story of the building (Figure 4.5.27), and $\gamma_{\max}$ is reduced by approximately 10% to 20% by the outrigger system. However, the NLRHA results indicate that the BRB-outrigger system does not exhibit better reductions in $\gamma_{\max}$ than the conventional outrigger system. This may be because the location where $\gamma_{\max}$ develops is higher than the upper BRB-outrigger, and the conventional outrigger provides greater rotational spring stiffness if compared to the BRB-outrigger after the BRBs yield, which is more effective in reducing inter-story drift responses. In addition, the conventional outrigger system (SingleElastic and DualElastic models) exhibits a 5% to 10% increase in the maximum base shear ($V_{c,\max}$), whereas the BRB-outrigger system (Single and Dual models) reduces the $V_{c,\max}$ by approximately 10%. This is because the conventional outrigger, that keeps the elastic response, could generate greater lateral force demands during an earthquake. In addition, the energy dissipated by BRBs also assists in reducing $V_{c,\max}$ by the increased equivalent damping ratio. For the maximum overturning moment at the core structure base ($M_{c,\max}$), the reductions are approximately 3% and 11% for the SingleElastic and Single models, respectively, and approximately 17% and 16% for the DualElastic and Dual models, respectively. If compared to the single outrigger, the additional lower outrigger in the dual outrigger system applies an additional reaction moment close to the core structure base. Therefore, the $M_{c,\max}$ can more effectively be reduced if compared to the single outrigger system. Table 4.5.11 presents the maximum perimeter column axial force at the base ($C_{1,\max}$) and between the two outrigger levels ($C_{34,\max}$), where the subscripts 1 and 34 refer to the perimeter column in the 1st and 34th stories, respectively. The perimeter column axial forces of the BRB-outrigger (Single and Dual) models are approximately only 20% of those of the conventional outrigger systems (SingleElastic and DualElastic models). This indicates that the yielding of BRBs effectively limits the maximum force developed in the perimeter columns and outrigger truss members. If the dual outrigger system is compared to the single outrigger system, $C_{1,\max}$ in the dual outrigger system is greater than that in the single outrigger system, but $C_{34,\max}$ in the dual outrigger system is smaller than that in the single outrigger system. This is because the lower outrigger applies an additional reaction moment on the core structure by applying additional axial force on the perimeter columns. Compared to the conventional outrigger system, the BRB-outrigger system can achieve better performance by reducing the seismic response without excessively increasing the perimeter column axial forces. In addition, the dual BRB-outrigger system performs better in reducing the overall seismic response by approximately 3% to 7% than the single BRB-outrigger system. Based on the analysis results, the SA and NLRHA exhibit similar trends in the reduction of seismic response, and the differences between the SA and NLRHA results should not affect the aims of parametric study on the optimal design.

Table 4.5.10 SA results of the 96-story Single model

mode	$T_{eq,n}$ (sec)	yield roof drift ratio ($y_{top,n}/h$, % rad.)	ductility, μ_n	E_d/E_s (%)	$h_{eq,n}$	$\theta_{\max}$ (% rad.)
1st	8.234	0.184	3.83	23	0.038	0.707
2nd	1.374	0.044	1.68	9	0.027	0.074
3rd	0.481	0.045	0.20	0	0.020	0.009
4th	0.244	0.011	0.18	0	0.020	0.002

Table 4.5.11 Maximum responses calculated from SA and NLRHA (BCJ-L2 ground motion) results of the 96-story example model

		DM					MBM				
		Core	Single Elastic	Dual Elastic	Single	Dual	Core	Single Elastic	Dual Elastic	Single	Dual
θ_{max} (% rad.)	S	0.907	0.834	0.818	0.736	0.711	-	-	-	-	-
	N	0.918	0.807	0.800	0.807	0.756	0.920	0.810	0.805	0.810	0.759
γ_{max} (% rad.)	S	1.27	1.15	1.14	1.02	1.00	-	-	-	-	-
	N	1.59	1.29	1.21	1.38	1.28	1.62	1.29	1.24	1.42	1.32
$V_{c,max}$ ($\times10^5$ kN)	S	1.30	1.33	1.39	1.29	1.31	-	-	-	-	-
	N	2.43	2.44	2.57	2.19	2.05	2.32	2.36	2.61	2.20	2.13
$M_{c,max}$ ($\times10^7$ kN-m)	S	2.10	2.06	2.03	1.87	1.81	-	-	-	-	-
	N	2.91	2.76	2.55	2.56	2.37	2.86	2.79	2.69	2.57	2.42
E_{d2}/E_s(%)	N	0	0	0	22.0	11.1	0	0	0	15.5	13.2
E_{d1}/E_s(%)	N	0	0	0	0	18.8	0	0	0	0	7.7
h_{eq}(%)	S	0.02	0.02	0.02	0.039	0.045	-	-	-	-	-
	N	0.02	0.02	0.02	0.038	0.044	0.02	0.02	0.02	0.032	0.037
$C_{34,max}$ ($\times10^4$kN)	N	-	6.29	5.55	1.25	1.24	-	6.49	5.64	1.25	1.25
$C_{1,max}$ ($\times10^4$kN)	N	-	6.29	8.90	1.25	2.20	-	6.49	9.01	1.25	2.20

Note: S=SA, N=NLRHA

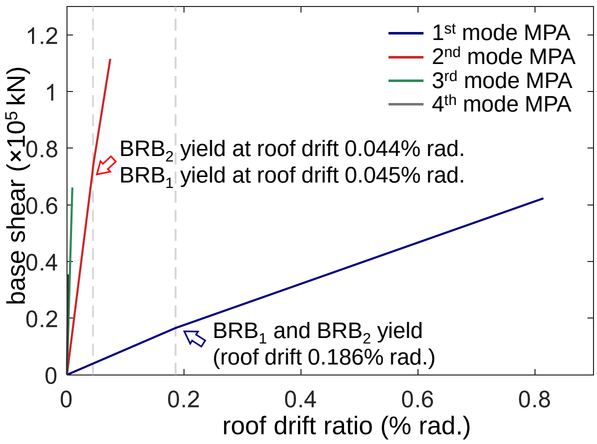


Figure 4.5.34 The MPA curves of the 96-story Dual model

4.6 SUMMARY

This chapter introduces the analysis methods for evaluating the seismic performance of a structure with BRB-outrigger. The effectiveness of using the UM and DM model to represent the structure with BRB-outrigger is demonstrated. The seismic performance of structures with and without outrigger, and with conventional outrigger and BRB-outrigger are compared. The summaries of this chapter are as follows:

(1) The procedure of SA, which incorporates the equivalent damping ratio in order to consider the BRBs inelastic behavior, is introduced. It is anticipated that the yielding of BRBs only causes slight changes in stiffness and mode shape, this, it is assumed that the modal superposition is still applicable after the BRBs yield. The MPA is used to obtain post-yield force and displacement relationship for the models with multiple BRB-outriggers since the BRBs yield in a different time. It is also assumed that the mode shapes are the same before and after the BRBs yield in the MPA. Based on the analysis results, the SA can well estimate the seismic performance such as $\theta_{\max}$ and $\gamma_{\max}$.

(2) The NRLHA with 8 different ground motions is performed in order to confirm the effectiveness of SA and the use of UM and DM models. The analysis results suggest that the UM and DM models with 1-m mass spacing can be used to represent structure with BRB-outrigger system.

(3) Both the structures with conventional outrigger and with BRB-outrigger can mitigate seismic response by increasing system stiffness. However, the conventional outrigger system, that keeps elastic response and lacks energy dissipation mechanism, could amplify seismic demand and significantly increase perimeter column axial force and reaction force demands. Through the yielding of BRB and the increased equivalent damping ratio, the BRB-outrigger system performs better than the conventional outrigger system and avoids excessive increases in reaction forces and perimeter column axial forces.

(4) Based on the analysis results, the dual BRB-outrigger system generally performs better than the single BRB-outrigger system. However, the reductions in seismic response are not proportional to the number of BRB-outrigger layers. In addition, the core structure base overturning moment can be reduced when the lower BRB-outrigger is close to core structure base.

(5) Based on the analysis in this chapter, the DM model without considering the secondary effect is used to perform SA and NLRHA in the parametric study. In addition, the $\theta_{\max}$ and $\gamma_{\max}$ calculated from SA and NLRHA, and the $M_{c,\max}$ calculated from NLRHA are used as indicators to indicate seismic performance in the parametric study.

4.7 REFERENCES

ASCE (2016) *Minimum design loads for buildings and other structures ASCE7-16, ASCE standard*. doi: 10.1061/9780784412916.

Chopra, A. K. and Goel, R. K. (2002) “A modal pushover analysis procedure for estimating seismic demands for buildings,” *Earthquake Engineering and Structural Dynamics*. doi: 10.1002/eqe.144.

Kasai, K., Fu, Y. and Watanabe, A. (1998) “Passive Control Systems for Seismic Damage Mitigation,” *Journal of Structural Engineering*, 124(5), pp. 501–512. doi: 10.1061/(asce)0733-9445(1998)124:5(501).

Nathan M. Newmark and Rosenblueth, E. (1971) *Fundamentals of earthquake engineering newmark*. Englewood Cliffs, N.J., USA: Printice-Hall.

Shome, N. and Cornell, C. A. (1998) “Normalization and scaling accelerograms for nonlinear structural analysis,” *Proceedings of the 6th US National Conference on Earthquake Engineering*.

5

PROGRAMMING FOR PARAMETRIC STUDY

CHAPTER CONTENTS

5.1 Introduction 5-3
5.2 Computer program procedure 5-3
5.3 Main program 5-5
5.4 Input file 5-6

5.1 INTRODUCTION

This chapter introduces the programming of performing the SA and NLRHA which are introduced in Chapter 4. In order to perform a large amount of analysis in the parametric study, a computer program, which supports batch analysis, written by using C++ programming language was developed to perform the parametric study. A brief introduction on the programming is presented in this chapter. The detail of the C++ scripts are shown in Appendix A. The 96-story Dual model introduced in Chapter 4 is used as an example to demonstrate the programming detail. Appendix B and Appendix C show the OpenSees scripts of DM and MBM models of a 32-story single BRB-outrigger model, respectively.

5.2 COMPUTER PROGRAM PROCEDURE

Figure 5.2.1 shows the execution procedure of the computer program. The step-by-step procedure is illustrated as follows:

(1) After the parameters of each analytical model are determined, the parameters of each analytical model are listed in the input file (inpA.txt). The main program reads the parameter values from the input file. The detail and example of the input file are introduced in the following sections.

(2) Based on the given parameters, the main program constructs the UM model and run the calculations for performing modal analysis and SA.

(3) Based on the given parameters, the main program constructs the DM model by generating a Tcl (.tcl) script output file with the commands for performing modal analysis. The main program then calls OpenSees (OpenSees.exe) to run the Tcl script and return the modal analysis results including the vibration periods and mode shapes to the main program.

(4) The main program computes the $u_{d,y}$ for single BRB-outrigger system, or $u_{d,y1}$ and $u_{d,y2}$ for dual BRB-outrigger system (the calculations of $u_{d,y}$, $u_{d,y1}$, and $u_{d,y2}$ are introduced in Chapter 6). Then, create the DM model in Tcl script file with the calculated $u_{d,y}$, or $u_{d,y1}$ and $u_{d,y2}$ for the BRB elements and the commands for performing MPA with the load pattern obtained from the modal analysis results in

Step (3). The main program calls OpenSees to run the Tcl script file and return the pushover curve of each MPA to the main program.

(5) The main program run the SA based on the pushover curves obtained from MPA.

(6) Based on the given parameters, the main program constructs the DM model by generating a Tcl format output file with the commands for performing NLRHA, then calls OpenSees to run the script. The Tcl script of 96-story Dual DM model is shown in Appendix B.

(7) After the OpenSees finishes NLRHA, the analysis results are collected and sorted in the main program.

The above procedure is enclosed in a while loop in the main program. Therefore, the main program could continue running the next analysis until the end of the input file is reached.

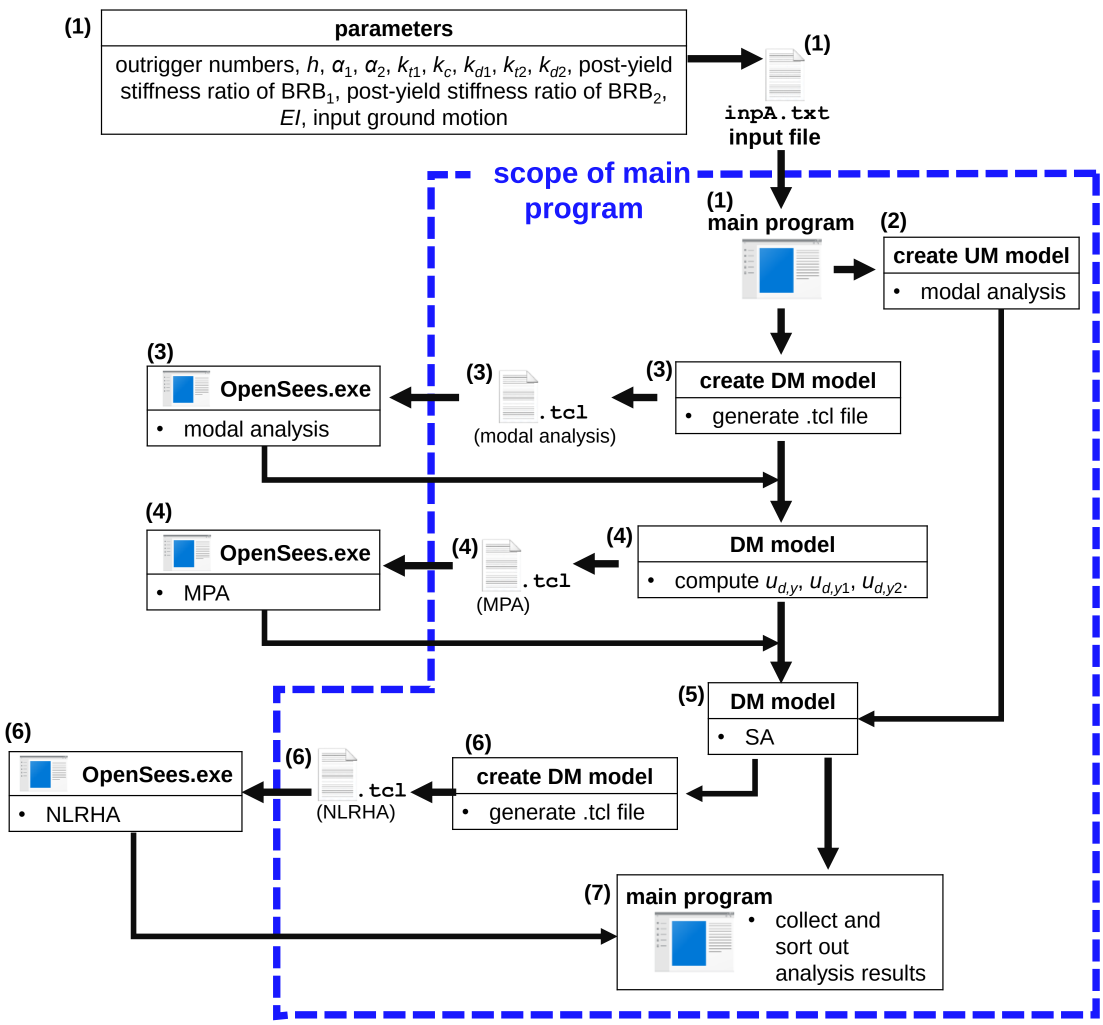


Figure 5.2.1 Procedure of the programming

5.3 MAIN PROGRAM

Figure 5.3.1 shows the framework of the main program. The script of the main program and the associated subroutines are shown in Appendix A. In order to increase the execution efficiency, the main program contains four different classes. Two major subroutines are included in the main part (main.cpp). The subroutine "getL_vec()" is responsible to create the UM model and to run the modal analysis using the UM model. The subroutine "MakeOpenseesTCL()" is responsible to create the DM model and generate the Tcl script with analysis commands for OpenSees to read. The task of calling OpenSees.exe is also done in this subroutine. As the SA contains matrix calculation, the class "matrix" (matrix.h and matrix.cpp) is developed in order to simplify the code and to make the program applicable to structure with multiple outriggers. The matrix class is capable to calculate the inverse, transpose, determinant, and eigenvector of a given square matrix in any dimension. The operator + (plus), − (minus), * (multiplication) were also developed. The class "Section" (Section.h) is to calculate the **A** (Equation 3.2.19) and **B** (Equation 3.2.21) matrices for each segment and store the information of each segment. The class "mode" (mode.h) calculates and stores the modal analysis results, including the modal mass, modal stiffness, mode shapes, vibration periods, and modal participation factors. The class "Spectrum" (Spectrum.h) deals with the design spectrum and the spectrum for the ground motion used in NLRHA.

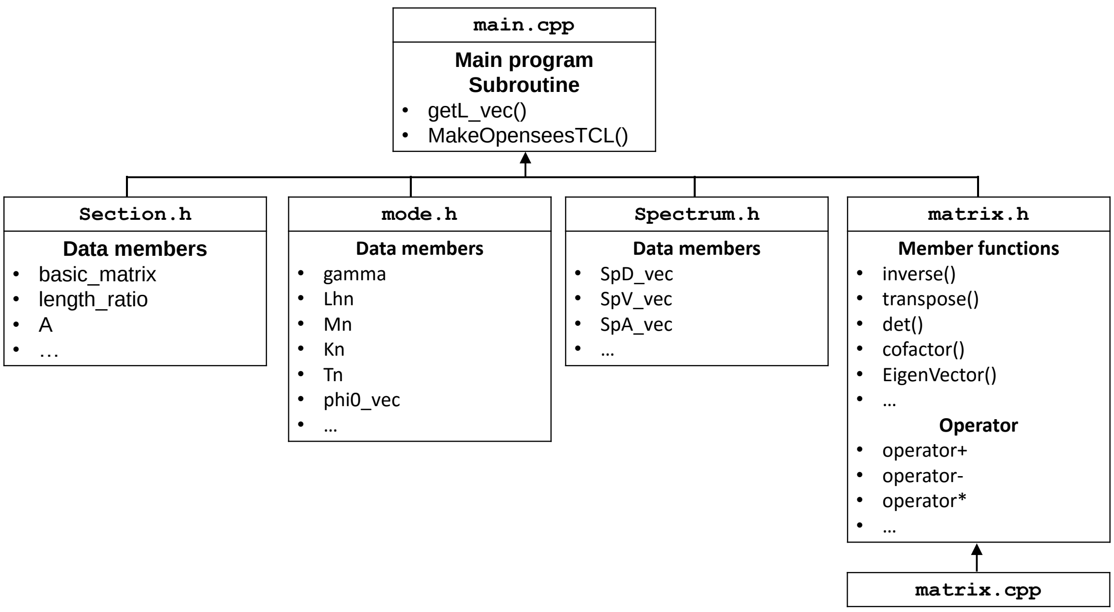


Figure 5.3.1 Framework of the main program

5.4 INPUT FILE

The format of the input file is illustrated in Table 5.4.1. The parameter numbers of α, k_c, k_t, k_d, p, and $u_{d,y}$ should be consistent with the number of outriggers (seg_num). The input file of the 32-story Single model is shown in Table 5.4.2.

Table 5.4.1 Format of the input file

variable name	illustration	variable type in C++ programming
seg_num	number of outrigger	integer
YieldDrift	(=1/θ_r)	double
x1	reserved for spare	double
x2	reserved for spare	double
h	h (unit: m)	double
A1, A2, …	α_1, α_2, …	double
kt1, kc, kd1, kt2, kc, kd2, …	k_{t1}, k_c, k_{d1}, k_{t2}, k_c, k_{d2}, …	double
p1, p2, …	p_1, p_2, …	double
m	mass (unit: kN/m/sec^2)	double
Eib	EI at core structure base (unit: kN-m^2)	double
Eit	EI at core structure top (unit: kN-m^2)	double
lt	l_t (unit: m)	double
tolerance	tolerance for SA iteration	double
filename	model name	std::string
inputEQ	input ground motion	std::string
EQSF	scale factor of input ground motion	double
udy1, udy2, … (optional)	$u_{d,y1}$, $u_{d,y2}$, … (unit: m)	double

Table 5.4.2 Input file of the 32-story Single model

seg_num	YieldDrift	x1	x2	h	A
2	750	0	0	128	0.6875
kt	kc	kd	p	m	Eib
24304687.5	486093.75	2430468.75	0.01	225	2E+10
Eit	lt	tolerance	filename	inputEQ	EQSF
16000000000	16	1	test	BCJL2	1

6

PRELIMINARY ANALYSIS

CHAPTER CONTENTS

6.1 Introduction 6-3
6.2 Modal analysis 6-3
6.3 BRB yield deformation 6-10
6.3.1 Calculation of BRB yield deformation 6-10
6.3.2 Effect of yielding roof drift ratio 6-12
6.3.3 Effect of outrigger truss flexural stiffness 6-16
6.3.4 Effect of BRB yield deformation ratio 6-18
6.4 BRB energy dissipation efficiency based on SA 6-28
6.4.1 Effect of BRB yield deformation ratio and roof yielding drift ratio ... 6-28
6.4.2 Effect of BRB stiffness parameters 6-35
6.5 Summary 6-40
6.6 References 6-41

6.1 INTRODUCTION

This chapter presents the preliminary analysis results which are necessary for the purpose of parametric analysis. The modal analysis results with various design parameters such as outrigger elevation and the stiffness relationships between outrigger truss, BRB, and core structure are presented. In addition, the investigation on the effects of different BRB yield deformation (or yield strength) of the BRB-outrigger systems on seismic performance is demonstrated. The methods in determining the BRB yield deformation are introduced, and the most appropriate method, which is judged by the seismic performance of BRB-outrigger system, is suggested. In addition, the energy dissipation efficiency with different outrigger elevations and BRB yield deformations is studied by performing the SA. The outrigger elevations that are best for developing satisfactory equivalent damping ratio are discussed.

6.2 MODAL ANALYSIS

The modal analysis results can be used to indicate the effectiveness of the outrigger effect. As the BRB-outrigger increases the system stiffness, when compared with the core structure (without outrigger effect), the larger decrease in vibration period suggests greater outrigger effect. Figure 6.2.1 presents the relationships between the 1st to the 3rd mode periods and outrigger elevation (α) for the 16-, 32-, 64-, and 96-story models with single BRB-outrigger calculated from Met. I and Met. II. It should be noted that the parameters R_{db} and R_{dc} are used in Met. I and Met. II, respectively. When α equals 0, it is a core structure without outrigger effect. The BRB-outrigger stiffens the system and thus decrease the vibration period. The larger values of R_{db} and R_{dc} suggest the BRB is stiffer and thus cause larger drops in vibration periods. Although the Met. I and Met. II use different approaches in constructing the models, the models with similar values of R_{db} and R_{dc} have similar ranges of vibration period distributions. As shown in the 1st mode period, the outrigger elevations when the periods are minimum calculated from Met. I are only slightly higher than Met. II. This implies that the outrigger effect should be affected primarily by outrigger elevation. Under the same R_{db} or R_{dc} values, when α is between 0.5 and 0.8, the outrigger effect on the system is the most significant, since the 1st mode period becomes minimum within this range. When α is approximate 0.2 and 0.8, the outrigger affects the 2nd mode response the most. Also, when α is approximate 0.1, 0.5, and 0.9, the outrigger affects the 3rd mode response the most. As shown in Figure 6.2.1, when α is approximate 0.5, the outrigger has almost no effect on the system 2nd mode response as the vibration period remains the same. This is because the elevation of the inflection point of the 2nd mode shape is approximate 0.5 of the building height, and the very small core structure rotation demand would not trigger the outrigger truss to rotate. The same reason applies to the 3rd mode response. As the 1st mode dominates the overall response, the optimal outrigger elevation of the single BRB-outrigger system should be close to α approximate 0.6 to 0.8, where the outrigger effect is the most significant. In addition, the decreases of vibration period as R_{dc} (R_{db}) increases from 0.1 to 1 is greater than when R_{dc} (R_{db}) increases from 1 to 3. This suggests that the magnitude of the outrigger effect is not proportional to the increasing R_{dc} and R_{db}.

Figure 6.2.2 to Figure 6.2.5 present the distributions of the 1st and 2nd mode vibration periods with respect to lower (α_1) and upper (α_2) outrigger elevations for the 16-, 32-, 64-, and 96-story models with dual BRB-outrigger systems. When α_1 and α_2 are 0, it refers to a core structure without outrigger effect. The presence of outriggers increase the system stiffness and decrease the vibration period. If compare the dual to single BRB-outrigger system (only the upper BRB-outrigger exists, α_1=0), the lower outrigger further increases the system stiffness. When α_2 is approximately 0.6 to 0.8 and α_1 is approximately 0.5 to 0.8, the 1st mode vibration period decreases to its minimum. For the 2nd mode response, when α_2 is approximately 0.8 to 0.9 and α_1 is approximately 0.2 to 0.3, the 2nd mode vibration period decreases the most. If considered the first two mode responses, the upper outrigger elevation (α_2) in a dual BRB-outrigger system that results in the greatest outrigger effect is approximate 0.6 to 0.8, which is slightly higher than the single BRB-outrigger system (0.5 to 0.8). In addition, the outrigger effect is more significant when α_1 is approximately 0.6 to 0.8 under the 1st mode deformed shape, and when α_1 is approximately 0.2 to 0.3 under the 2nd mode deformed shape. Therefore, for a dual BRB-outrigger system, as the 1st mode response mainly dominates the overall response, the α_2 locates within 0.6 and 0.8 and α_1 locates within 0.5 and 0.8 should be the optimal configuration that results in greatest outrigger effect in order to minimize seismic response. However, the 2nd mode response could also affect the seismic responses such as inter-story shear and bending moment, the α_1 locates within the range of 0.2 to 0.3 is anticipated to mitigate the inter-story shear and overturning moment responses. As shown in Figure 6.2.2 to Figure 6.2.5, the larger R_{d2c} and R_{kd} values suggest the greater BRB stiffness and thus result in greater outrigger effect. It should be noted that the decreases of vibration period when R_{d2c} increases from 0.1 to 1 are larger than when R_{d2c} increases from 1 to 3. This suggests that the increase of the BRB axial stiffness is not proportional to the increase of outrigger effect. In addition, increasing R_{kd} increases the overall outrigger effect. Also, the increase in R_{kd} and the increase of outrigger effect are not proportional to each other.

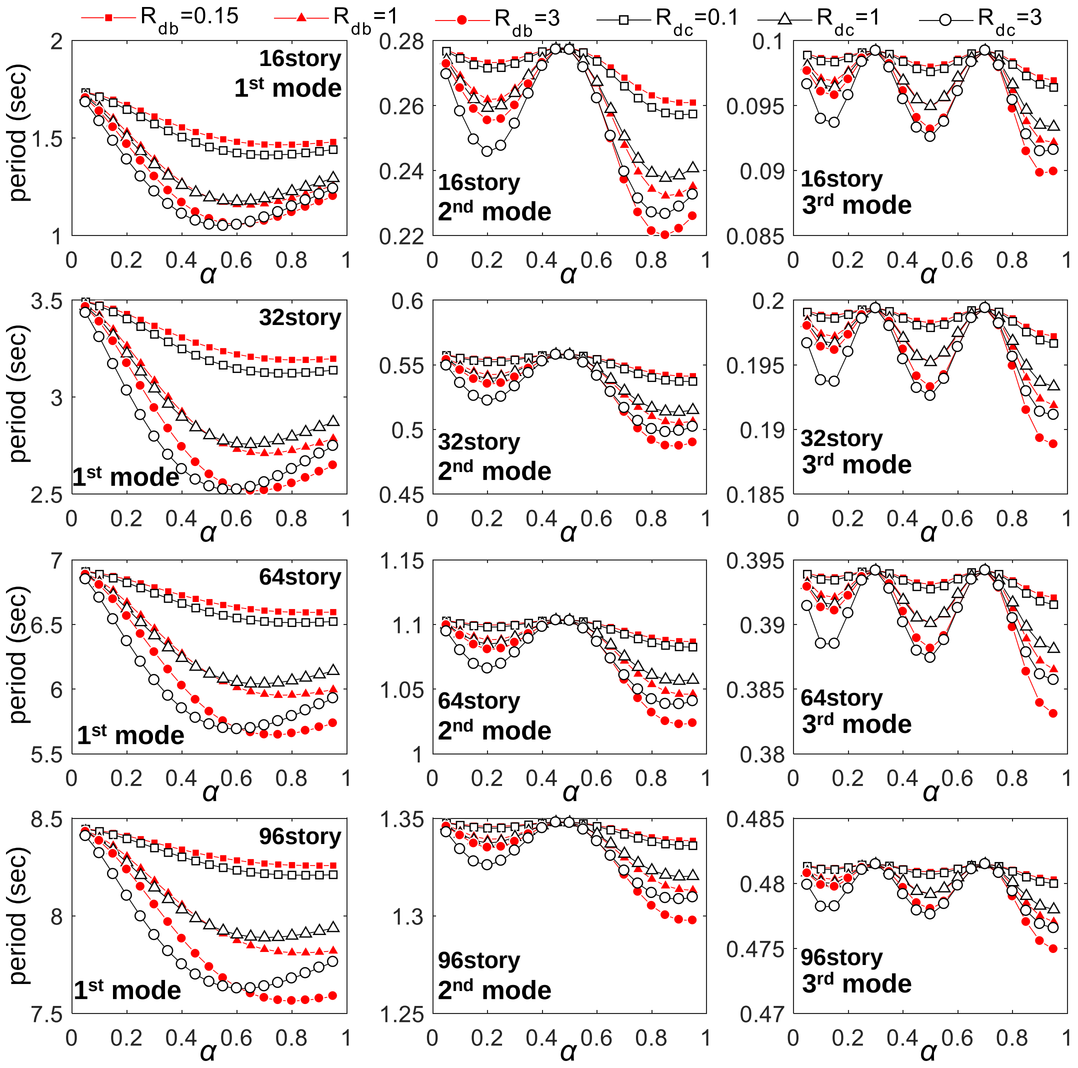


Figure 6.2.1 The vibration periods with respect to α for single BRB-outrigger

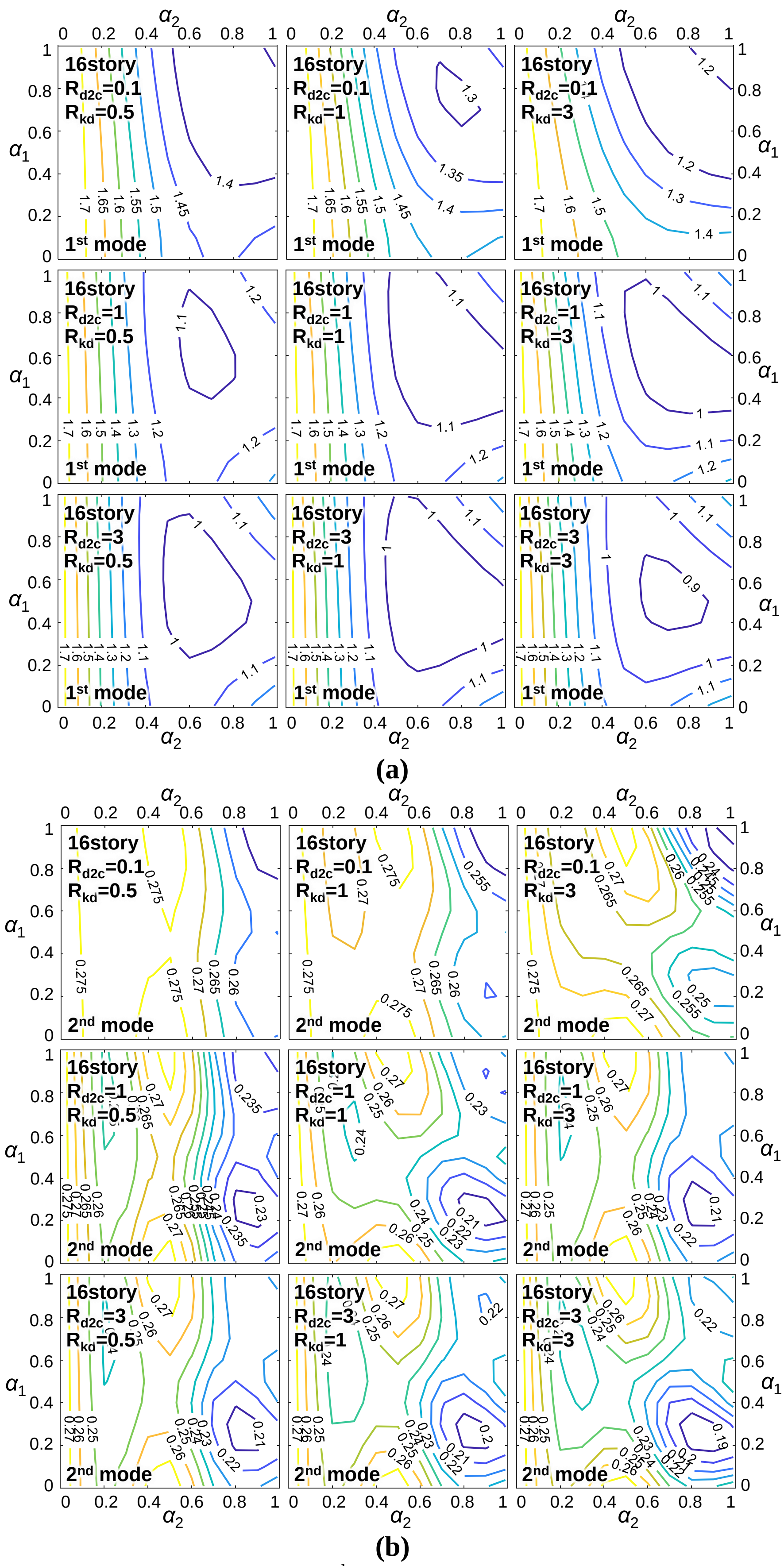
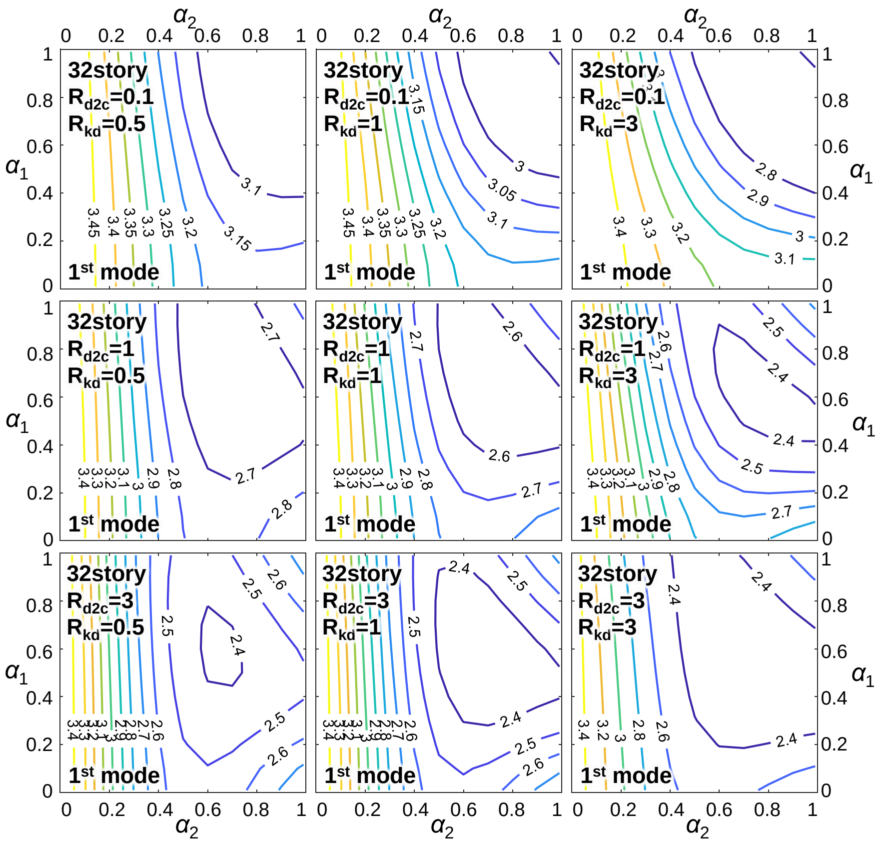
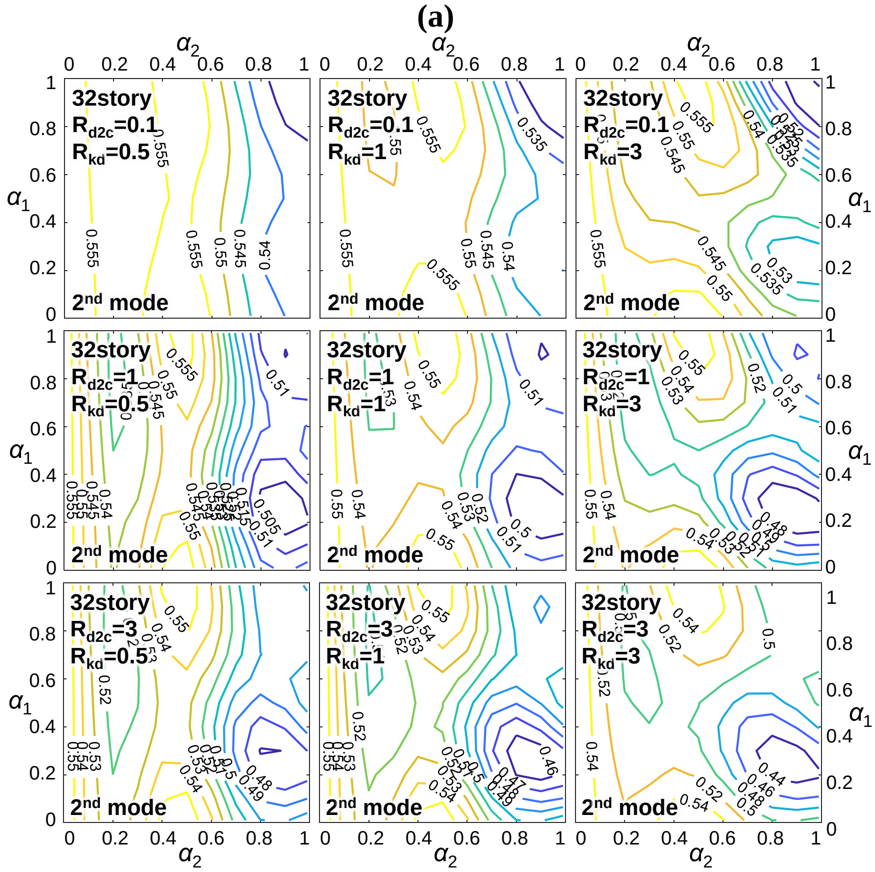


Figure 6.2.2 The (a) 1st and (b) 2nd mode vibration periods for 16-story model

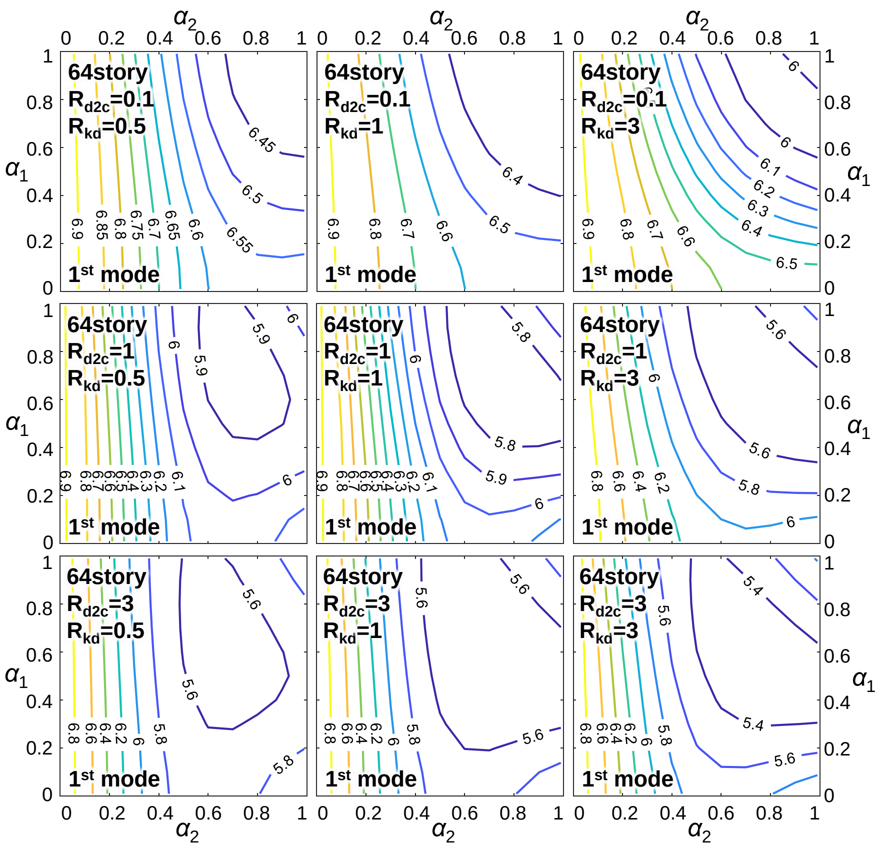


(a)

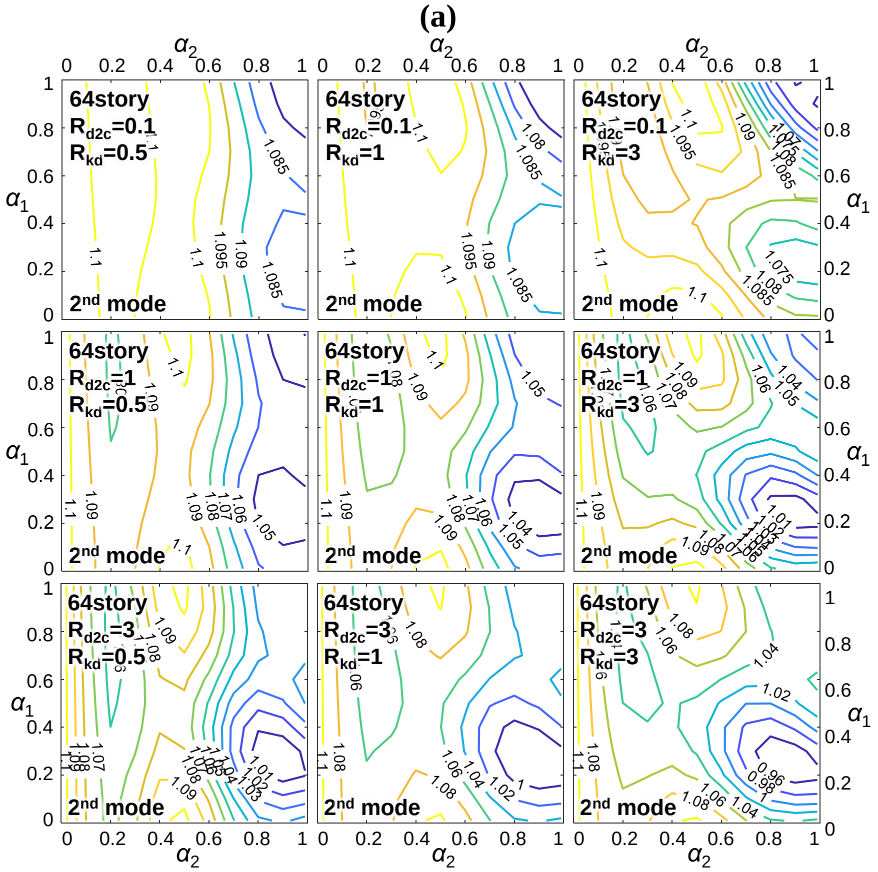


(b)

Figure 6.2.3 The (a) 1st and (b) 2nd mode vibration periods for 32-story model

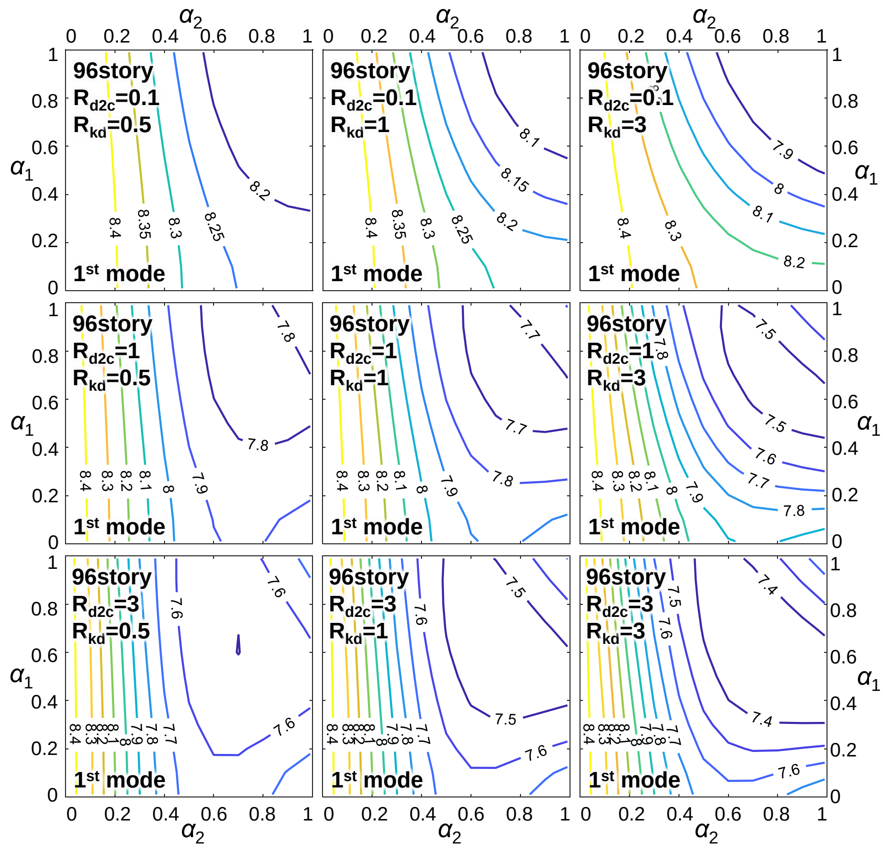


(a)

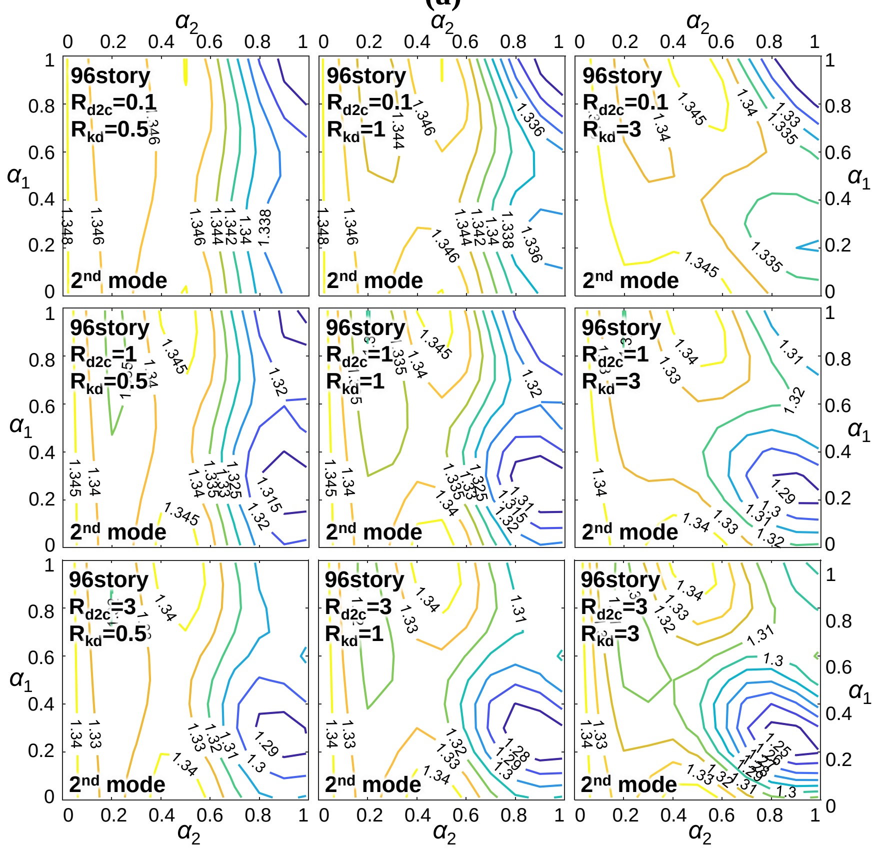


(b)

Figure 6.2.4 The (a) 1st and (b) 2nd mode vibration periods for 64-story model



(a)



(b)

Figure 6.2.5 The (a) 1st and (b) 2nd mode vibration periods for 96-story model

6.3 BRB YIELD DEFORMATION

The yielding deformation of BRB in the BRB-outrigger is critical as it determines when the BRB starts dissipating energy and when the equivalent damping ratio increases. If the yielding deformation of the BRB is too small, the BRB could easily yield during a small earthquake and uses up its ductility capacity or fractures before the end of an earthquake. However, if the yielding deformation of BRB is too large, the BRB could only slightly yield or even remains elastic during a moderate earthquake, and the energy dissipation efficiency would be very low. The methods in determining the BRB yield deformation are discussed.

6.3.1 Calculation of BRB yield deformation

The yield roof drift (θ_r) is the roof drift ratio when the first BRB yields. The θ_r is used to determine the BRB yield deformations. The θ_r could be the allowable elastic roof drift limit (for example 1/750 or 0.2% rad.). The larger θ_r value suggests the BRB yields at relatively large roof drift ratio and therefore leads to large BRB yield deformation, and vice versa. Five cases for calculating BRB yield deformation are proposed in this study, and the most appropriate one is adopted in the parametric study. Figure 6.3.1 illustrates the BRB yield deformations in the Case1 to Case4 for the dual BRB-outrigger system. The Case1, Case2, and Case3 define that the BRBs in two different BRB-outrigger levels yield simultaneously when core structure deforms in the 1st, 2nd, and 3rd mode shapes, respectively. The Case4 defines that the BRBs yield when core structure deforms in the SRSS combined deformed shape (the combination of the first four modes based on the design spectrum) and when roof drift reaches θ_r. For the dual BRB-outrigger system, the BRB yield deformation ratio, R_{udy}, is used to define the relationship between the yield deformations of $u_{d,y1}$ and $u_{d,y2}$, and is calculated as follows:

$$R_{udy} = \left| \frac{u_{d,y1}}{u_{d,y2}} \right| \tag{6.3.1}$$

The Case5, which only applies to dual BRB-outrigger system, assumes the $u_{d,y1}$ and $u_{d,y2}$ are identical to the $u_{d,y2}$ calculated in Case4. For the single BRB-outrigger system, the R_{udy} equals to 0, and for the dual BRB-outrigger system, the R_{udy} equals to 1 in Case5. The procedure of BRB yield deformation calculation is as follows:

(1) Calculate the spectral deformed shape of the first three modes based on the design spectrum, and calculate the R_{udy} for each mode shape (Case1 to Case3).

(2) Superpose the deformed shapes of the first three modes by using the SRSS combination. Scale the roof lateral displacement of the SRSS combined deformed shape until $\theta_r h$.

(3) Calculate both $u_{d,y2}$ and $u_{d,y1}$ (dual BRB-outrigger system) or only $u_{d,y}$ (single BRB-outrigger system) from the SRSS combined deformed shape obtained from Step (2). The BRB yield deformations ($u_{d,y}$ or $u_{d,y1}$ and $u_{d,y2}$) calculated until this step is defined as Case4.

(4) For the dual BRB-outrigger system, the $u_{d,y2}$ obtained in Step (3) is used to calculate the $u_{d,y1}$ for Case1, Case2, Case3, and Case5. The $u_{d,y1}$ in each Case is calculated by multiplying the $u_{d,y2}$ by R_{udy} calculated in Step (1).

Based on the abovementioned calculation procedure, the $u_{d,y2}$ are the same in all five cases, and the R_{udy} controls the $u_{d,y1}$ so that the BRB_1 and BRB_2 could reach yield deformation simultaneously in certain mode shapes. Therefore, for single BRB-outrigger system, the $u_{d,y}$ calculated from the five cases are the same. For the design practices, all BRBs in every BRB-outrigger yield approximately simultaneously is desirable as it prevents weak story or excessive deformation concentrates in certain outrigger level. In addition, as the BRB is displacement-related energy dissipation device, it is anticipated that the SRSS combined deformed shape with a roof drift ratio of θ_r (Case4) can best represent the deformed shape of a building when BRBs yield. The purpose of Case2 and Case3 is to confirm if the BRB yield approximately simultaneously under the higher mode shapes could assist in improving the seismic performance. The effects of θ_r and outrigger elevation on BRB yield deformation are first investigated using the 32-story model with single BRB-outrigger. The effect of R_{udy} on BRB yield deformation and seismic responses is then investigated by using the 16-, 32-, 64-, and 96-story models.

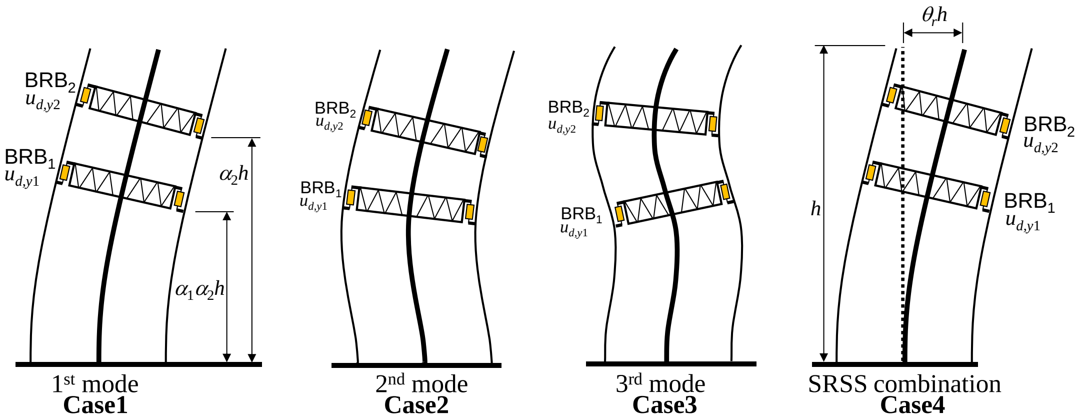


Figure 6.3.1 Illustration of BRBs' yielding under the Case(1) to Case(4)

6.3.2 Effect of yielding roof drift ratio

The seismic performance of the 32-story model with single BRB-outrigger when θ_r equals 1/50, 1/150, 1/350, 1/750, and 1/950 are investigated. The 32-story model has R_{dt}=0.1, R_{db}=3.5 (Met. I), and R_{dc}=5.0 (Met. II). Figure 6.3.2 presents the relationships between the required BRB yield deformation ($u_{d,y}$) and outrigger elevation (α) of the single BRB-outrigger system. A larger θ_r leads to a larger $u_{d,y}$. For a given θ_r, the maximum value of $u_{d,y}$ is obtained when α is approximately 0.6 to 0.7 in Met. I, and 0.5 in Met. II. The larger value of $u_{d,y}$ also suggest that the deformation demand on BRB is larger. For the design practices, the $u_{d,y}$ should lie in a reasonable range (for example, 1/1000 of the BRB length). If the BRB length is 4 m, the $u_{d,y}$ should be approximately 3 mm to 6 mm. Therefore, too large values of θ_r (for example 1/50, 1/150) are undesirable as the required $u_{d,y}$ is difficult to be achieved by conventional BRB or the BRB length would be very long (8 m to 10 m). The range of $u_{d,y}$ between 1/550 and 1/750 for the θ_r should be appropriate for the parametric study.

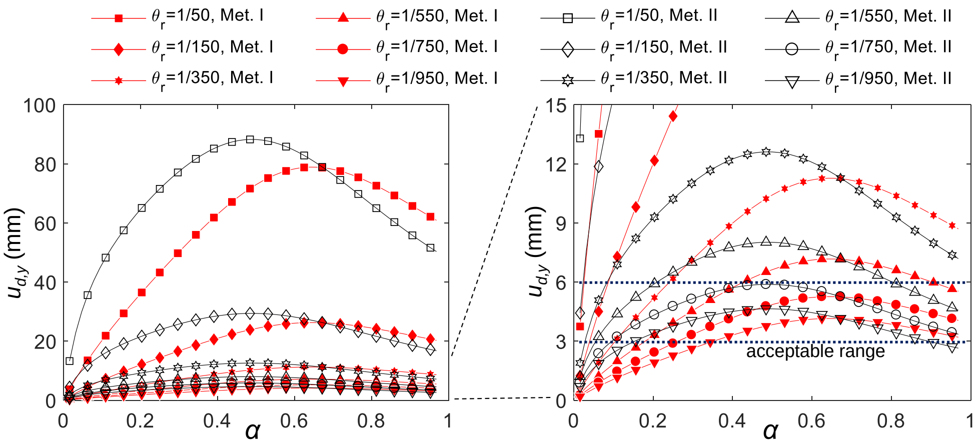


Figure 6.3.2 Illustration of BRBs initially yield under the Case(1) to Case(4)

Figure 6.3.3 and Figure 6.3.4 show the relationships between the maximum roof drift ($\theta_{\max}$) and outrigger elevation (α) calculated from SA and NLRHA (BCJ-L2 waves), respectively. The trends of maximum roof drift ratio variations with respect to α obtained from both SA and NLRHA (BCJ-L2 waves) are similar. However, these trends differ slightly in the 32-story model's analytical results in certain period ranges. This could because that for periods other than the first mode, the BCJL-L2 spectral accelerations are considerably greater than the design spectral acceleration. The responses from higher modes could be large. In addition, the analytical results of θ_r=1/50 and θ_r=1/150 are similar, and the rests are similar to each other. This could be due to that the models with cases of θ_r=1/50 and θ_r=1/150 have very large $u_{d,y}$, and the BRB may remain elastic deformation. Therefore, the single BRB-outrigger system acts as a conventional outrigger system. However, the cases with θ_r smaller than 1/350, the BRB yields and dissipates energy through its hysteretic response. Therefore, the responses can be further reduced if compared with the cases with θ_r = 1/50 and θ_r = 1/150. Figure 6.3.5 presents the relationship between θ_r and optimal α (when $\theta_{\max}$ is minimum). The SA results correlated well with the NLRHA results as shown in Figure 6.3.3 and Figure 6.3.4, both the SA and NLRHA results suggest that a smaller θ_r could better reduce $\theta_{\max}$, and the BRBs could start dissipating energy in a relatively small lateral deformation of the building. However, for θ_r values smaller than 1/550, the reduction in $\theta_{\max}$ was not large enough to significantly improve the seismic performance.

The ratio of BRB's cumulative plastic deformation to axial yield deformation (R_{CPD}) (ANSI/AISC 341-16, 2016) is used to indicate the ductility demand for the BRB. Figure 6.3.6 presents the R_{CPD} calculated from the NLRHA results (BCJ-L2 ground motion). When θ_r is greater than 1/150, the very small R_{CPD} values suggest that the BRBs have low energy dissipation efficiency, or deform elastically as a conventional outrigger system (when R_{CPD} = 0). However, when θ_r is smaller than 1/950, the very large R_{CPD} value may not be practically achieved in conventional BRBs. Therefore, from the above analysis results, θ_r = 1/750 results in an acceptable range of $u_{d,y}$ (from 3 to 6 mm for a 4-m long BRB) and R_{CPD} values for the example model. For the design practices, θ_r can be also determined by code specified maximum allowable elastic roof drift ratio.

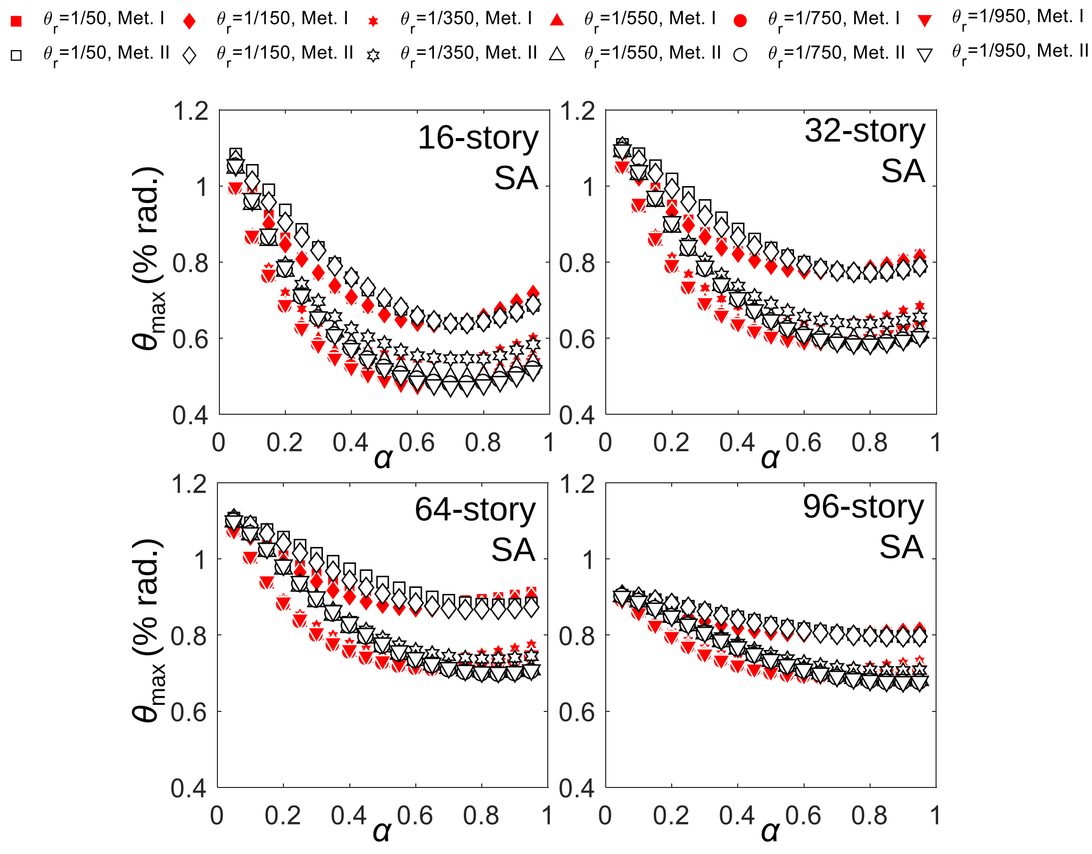


Figure 6.3.3 Relationships between $\theta_{\max}$ and α under various θ_r from SA

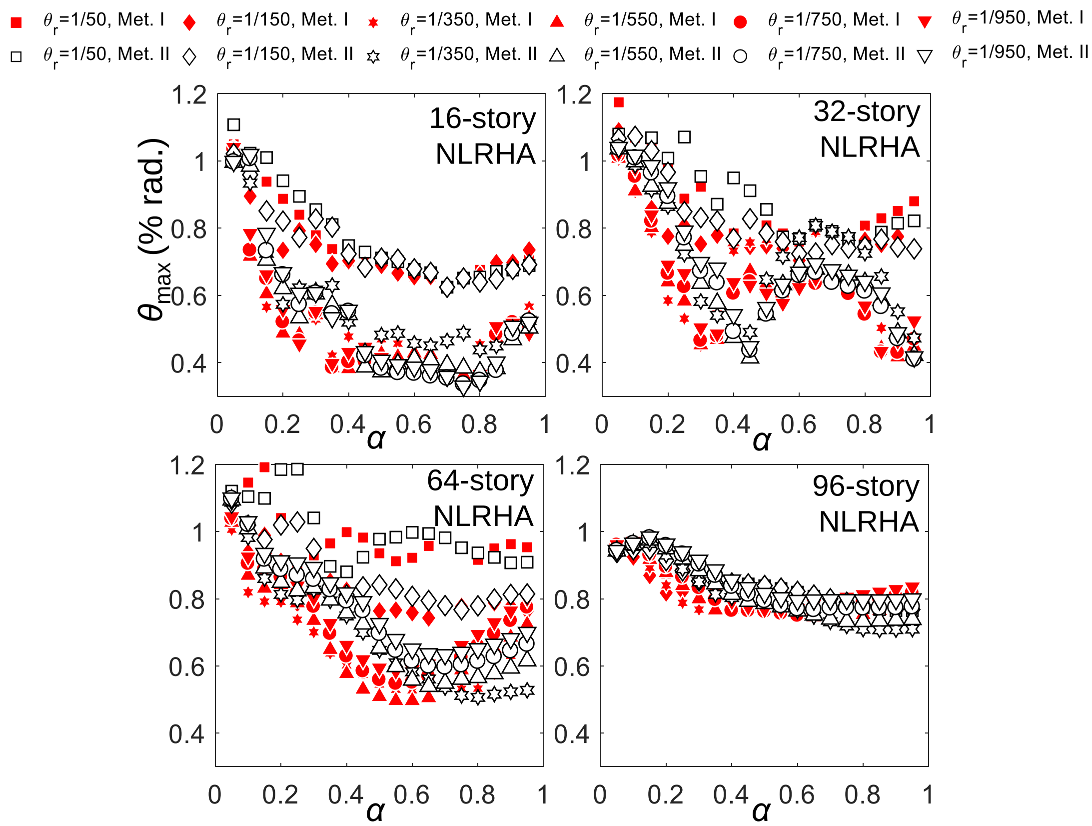


Figure 6.3.4 Relationships between $\theta_{\max}$ and α under various θ_r from NLRHA

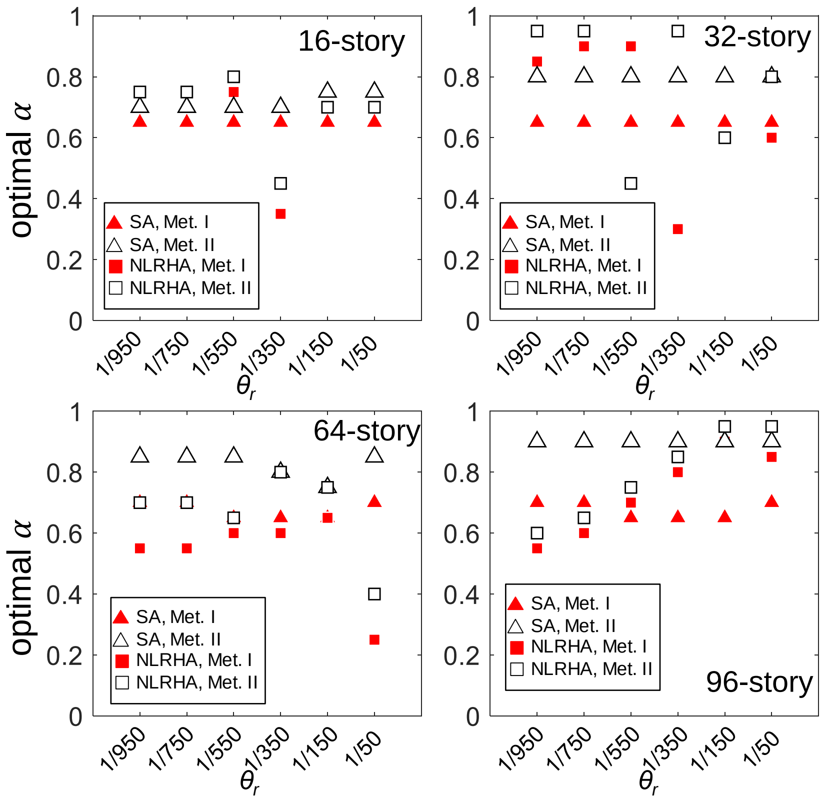


Figure 6.3.5 Relationships between optimal α and θ_r

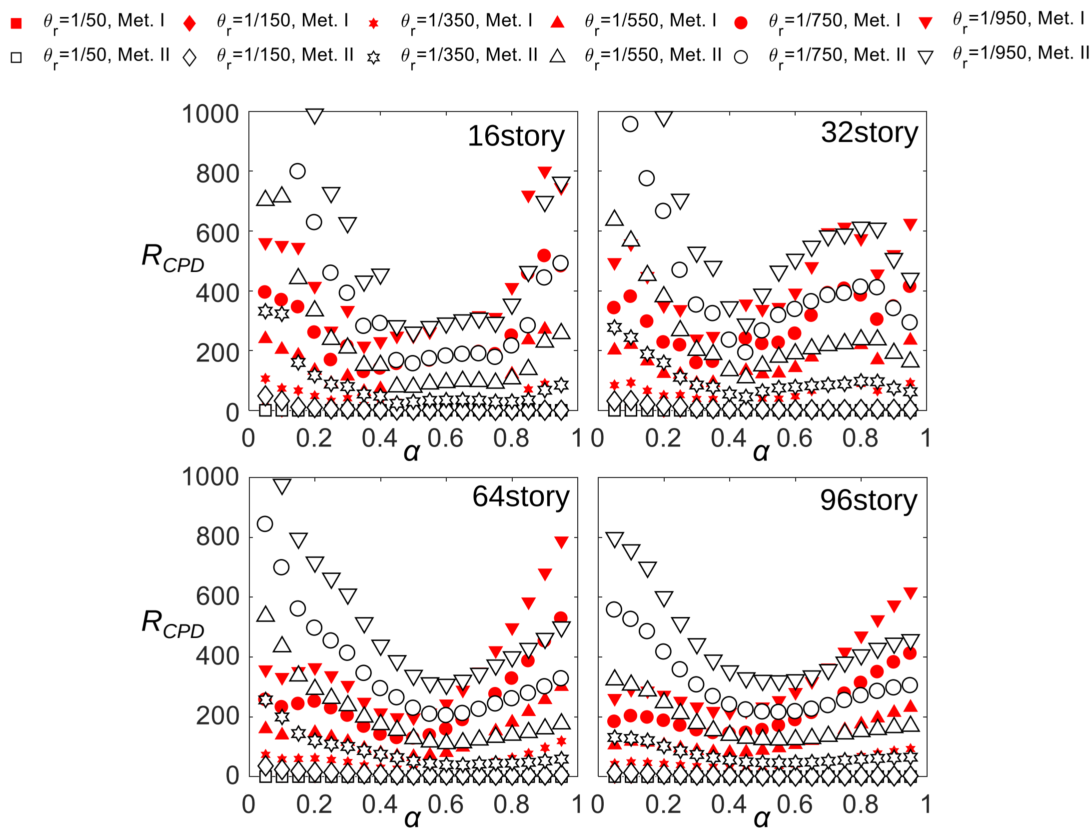


Figure 6.3.6 Relationships between R_{CPD} and α from NLRHA

6.3.3 Effect of outrigger truss flexural stiffness

The smaller R_{dt} value is desirable as it increases the axial deformation demand on the BRB. In the BRB-outrigger system, the outrigger truss should be sufficiently stiff to generate adequate deformation for the BRB. In this section, the analyses with a fixed θ_r at 1/750 and varied R_{dt} between 0.05 and 0.9 are performed in order to investigate the effect of R_{dt} value on seismic response. Figure 6.3.7 presents the relationship between $u_{d,y}$ and α under different R_{dt}. The ranges of $u_{d,y}$ differences (approximate 3 mm) because of varying R_{dt} is much smaller than varying θ_r (greater than 20 mm as shown in Figure 6.3.2). Figure 6.3.8 shows the relationships between the 1st to the 4th mode vibration periods and α with different R_{dt}. Figure 6.3.9 shows the relationships between $\theta_{\max}$ and α calculated from SA and NLRHA. Since k_t is much greater than k_d and k_c, changing R_{dt} only slightly affects the rotational stiffness provided by the BRB-outrigger (k_g). Therefore, the changes in dynamic characteristics and the maximum responses of the overall system because of the variation in R_{dt} (ranging from 0.05 to 0.9) are insignificant. For the design practices, the $u_{d,y}$ can be fine-tuned by changing R_{dt} so that $u_{d,y}$ lies within a desirable range. As the analysis results do not significantly change when R_{dt} varies between 0.05 and 0.9, the value of R_{dt} is fixed at 0.1 in the parametric study.

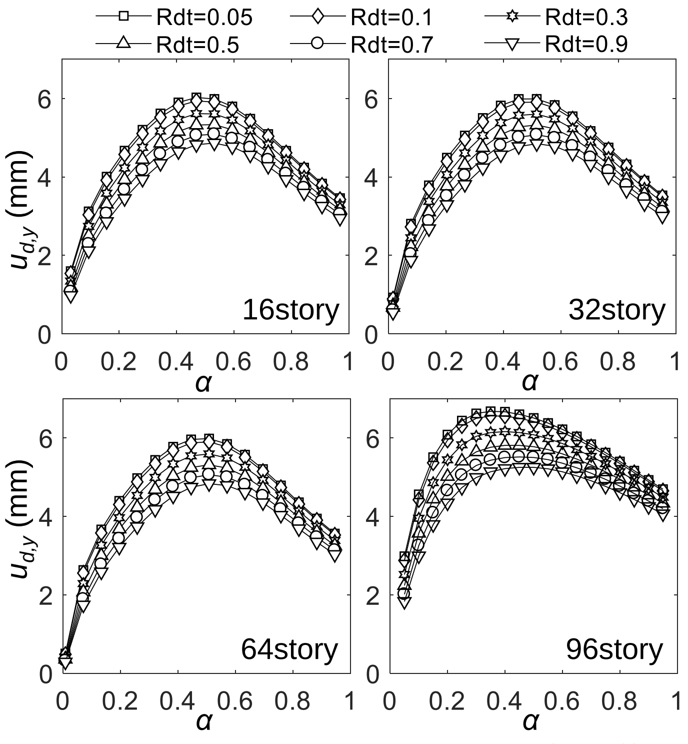


Figure 6.3.7 Relationship between $u_{d,y}$ and α for different R_{dt}

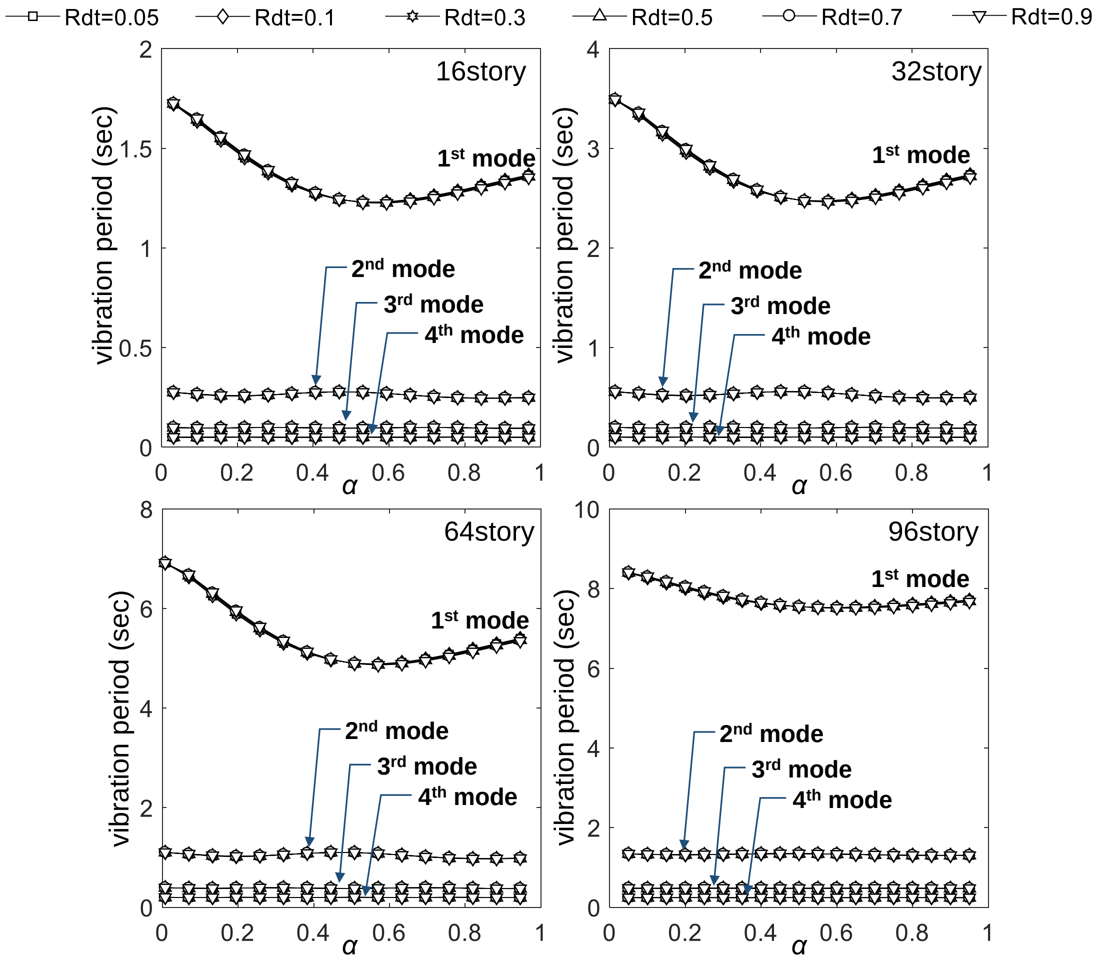


Figure 6.3.8 Relationship between vibration period and α for different R_{dt}

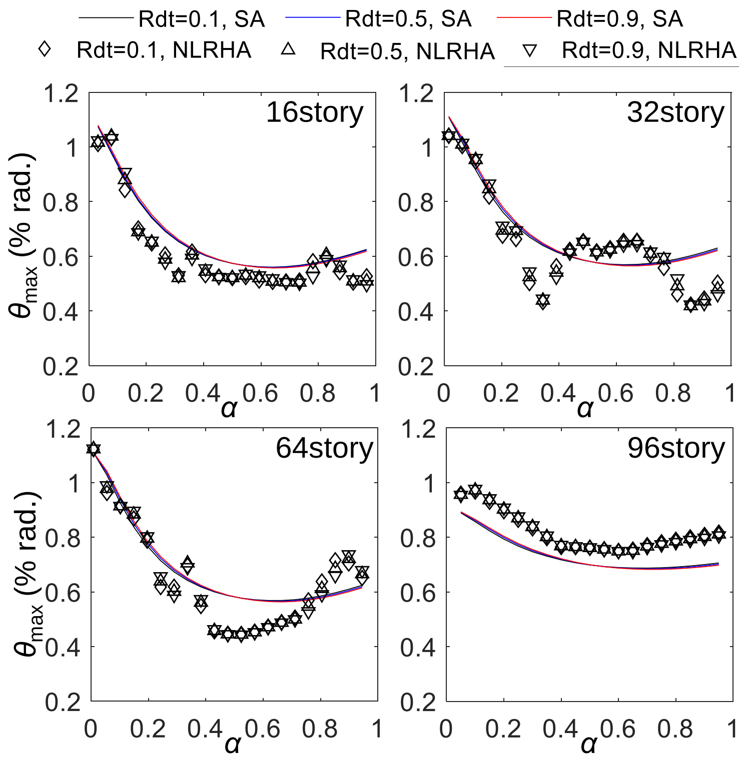


Figure 6.3.9 Relationship between $\theta_{\max}$ and α with different R_{dt}

6.3.4 Effect of BRB yield deformation ratio

The 32-story dual BRB-outrigger model shown in Figure 6.3.10a is used to demonstrate the relationship between different values of R_{udy} and seismic response. The α_2 and α_1 are 0.78 and 0.7, respectively. The core structure flexural rigidity (EI) is 1.6×10^{10} kN-m^2, the upper (k_{t2}) and lower (k_{t1}) outrigger truss flexural stiffnesses are 2.43×10^7 kN/m, and the perimeter column size is Box 1000×1000×85 mm. The other parameters are θ_r = 1/750, R_{d2c} = 5.0, R_{dt1} = R_{dt2} = 0.1, R_{kd} = 1.0. Figure 6.3.10a to Figure 6.3.10c show the 1st to the 3rd mode shapes of the DM model.

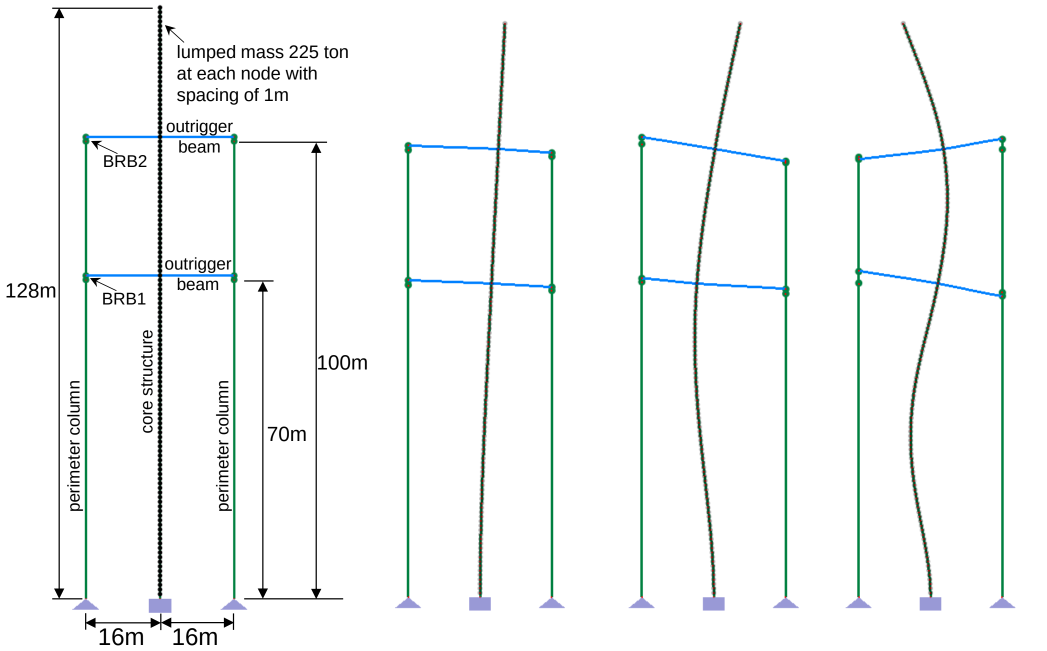


Figure 6.3.10 (a) The DM model of the 32-story dual BRB-outrigger system, and (b) the 1st, (c) the 2nd, and (d) the 3rd mode shapes

Table 6.3.1 shows the BRB yield deformations in the upper ($u_{d,y2}$) and the lower ($u_{d,y1}$) outriggers, the corresponding R_{udy} value, and the $\theta_{\max}$ calculated from both SA and NLRHA with BCJ-L2 ground motion for each Case. As the first mode dominates the overall response, the R_{udy} for Case1 and Case4 are very close to each other, and therefore the responses of Case1 and Case4 are similar. In addition, The Case3 results in a very large $u_{d,y1}$ as the core rotation is large at the lower outrigger elevation as shown in Figure 6.3.10d. The large $u_{d,y1}$ could make the BRB in lower outrigger to remain elastic deformation during a major earthquake. This could be the reason that the $\theta_{\max}$ of Case3 calculated from both SA and NLRHA are greater than the others. The Case2 results in a very small $u_{d,y1}$ as the lower outrigger is close to the inflection point of the 2nd mode shape. The small $u_{d,y1}$ could make the BRB in lower outrigger to yield easily during a minor earthquake. Figure 6.3.11 shows the 1st and 2nd mode MPA curves for the models with five different BRB yield cases. The 1st mode MPA curves of the models with Case1 and Case4 are very close to each other, and the BRBs in upper and lower outriggers yield approximately simultaneously. Case2 results in a very small $u_{d,y1}$, therefore, the BRB in lower outrigger yields under a relatively small roof drift if compared with Case1 and Case4. Case3 results in a very large $u_{d,y1}$, thus, the BRB in lower outrigger remains elastic deformation and yields until the roof drift reaches 0.5% rad. The $u_{d,y1}$ of Case5 is smaller than the $u_{d,y1}$ of Case1 and Case4, therefore, the BRB of Case5 in lower outrigger yields first, and the overall lateral strength of the system is smaller than Case1 and Case4 under the 1st mode MPA. Similar responses can be observed in the 2nd mode MPA. The 2nd mode MPA curve of Case2 indicates that the BRBs in upper and lower outriggers yield approximately simultaneously.

Table 6.3.1 R_{udy} and θ_{max} calculated from each Case

BRB yield case	$u_{d,y1}$ (mm)	$u_{d,y2}$ (mm)	R_{udy}	θ_{max}	
				SA	NLRHA
Case1	4.52	2.65	1.7	0.547	0.579
Case2	0.59	2.65	0.2	0.636	0.538
Case3	19.31	2.65	7.3	0.634	0.590
Case4	4.27	2.65	1.6	0.543	0.578
Case5	2.65	2.65	1.0	0.552	0.544

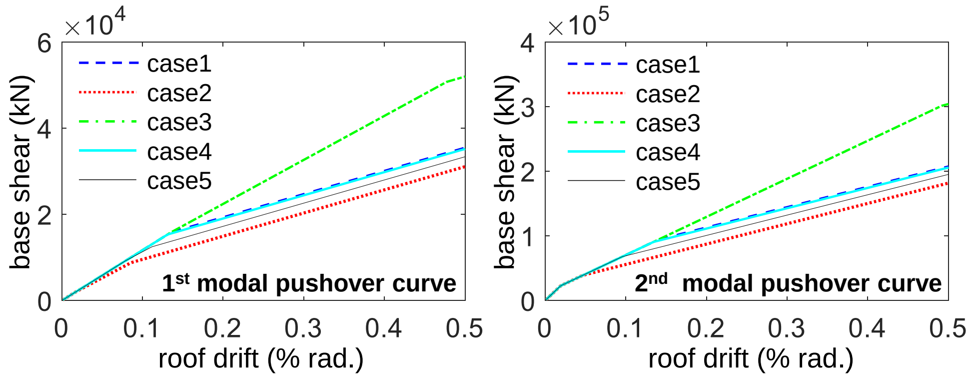


Figure 6.3.11 The 1st and 2nd mode MPA curves under five BRB yield cases

In addition to the analysis performed on the models with 5 different BRB yield cases, the SA and NLRHA were also performed on the same 32-story model with dual BRB-outrigger by varying R_{udy} from 0 to 20 (R_{udy} = 0 refers to single BRB-outrigger system). Figure 6.3.12 the relationships between θ_{max} and R_{udy} calculated from SA and NLRHA, respectively. The ground motions used in NLRHA are scaled as described in Chapter 4. Figure 6.3.12 also shows the R_{udy} value for each of the 5 BRB yield cases. Based on the analysis results, the larger value of R_{udy} delays the yielding of BRB in the lower outrigger. Therefore, the BRB in lower outrigger may only slightly yield or remain elastic deformation during an earthquake. The energy dissipation efficiency of the BRB in lower outrigger is low if R_{udy} is too large. The NLRHA results with ground motion Sendai, KobeJMA, and Tohoku show that the θ_{max} remains the same when R_{udy} is greater than approximate 4. This is because that the BRB in lower outrigger remains in elastic deformation under these ground motions when R_{udy} is greater than 4. As shown in Figure 6.3.12, the θ_{max} can be best reduced when R_{udy} is approximately 1.5 to 3.0, which is close to the R_{udy} values of Case1 and Case4. As the SRSS deformed shape used in Case4 could best represent the deformed shape when the BRBs in upper and lower outriggers yield approximately simultaneously. Therefore, Case4 is used in the parametric study.

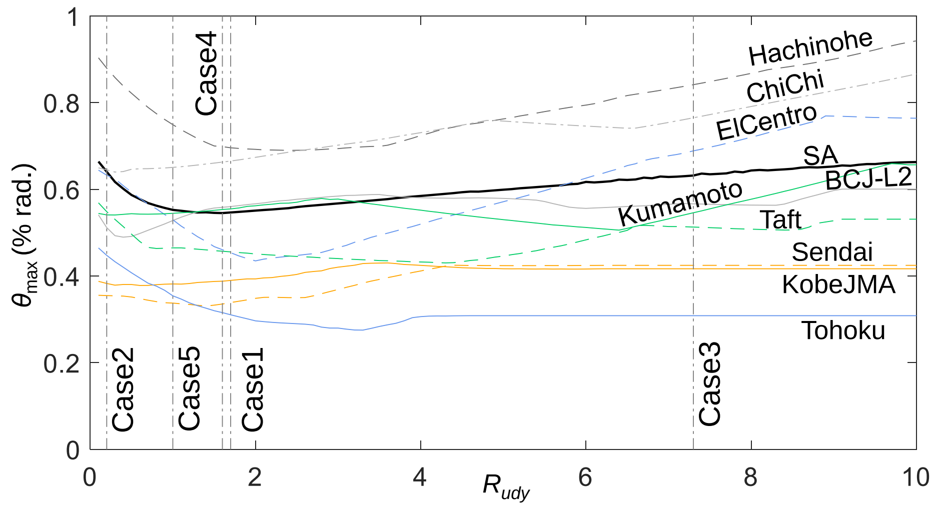


Figure 6.3.12 Relationships between θ_{max} and R_{udy} calculated from SA and NLRHA

In addition to the 32-story model example, the effect of R_{udy} is also investigated on the 16-, 32-, 64-, and 96-story models by varying α_2 between 0.55, 0.65, 0.75, 0.85, and 0.95 varying α_1 from 0 to 1. The BRB parameters R_{d2c} is fixed at 1, and the R_{kd} varies between 0.5, 1, and 3. Figure 6.3.13 presents the relationships between R_{udy} and α_1, and Figure 6.3.14 presents the relationships between $u_{d,y1}$ and α_1. It should be noted that $u_{d,y2}$ are the same in all five BRB yield cases. When the lower outrigger is approaching the upper outrigger (when α_1 is close to 1), the $u_{d,y1}$ and $u_{d,y2}$ become similar to each other. The system acts as a single BRB-outrigger system. In addition, the very large varieties of $u_{d,y1}$ in Case2 and Case3 are due to the more complicated deform shapes. The very large differences between $u_{d,y1}$ and $u_{d,y2}$ may be impractical in design practices.

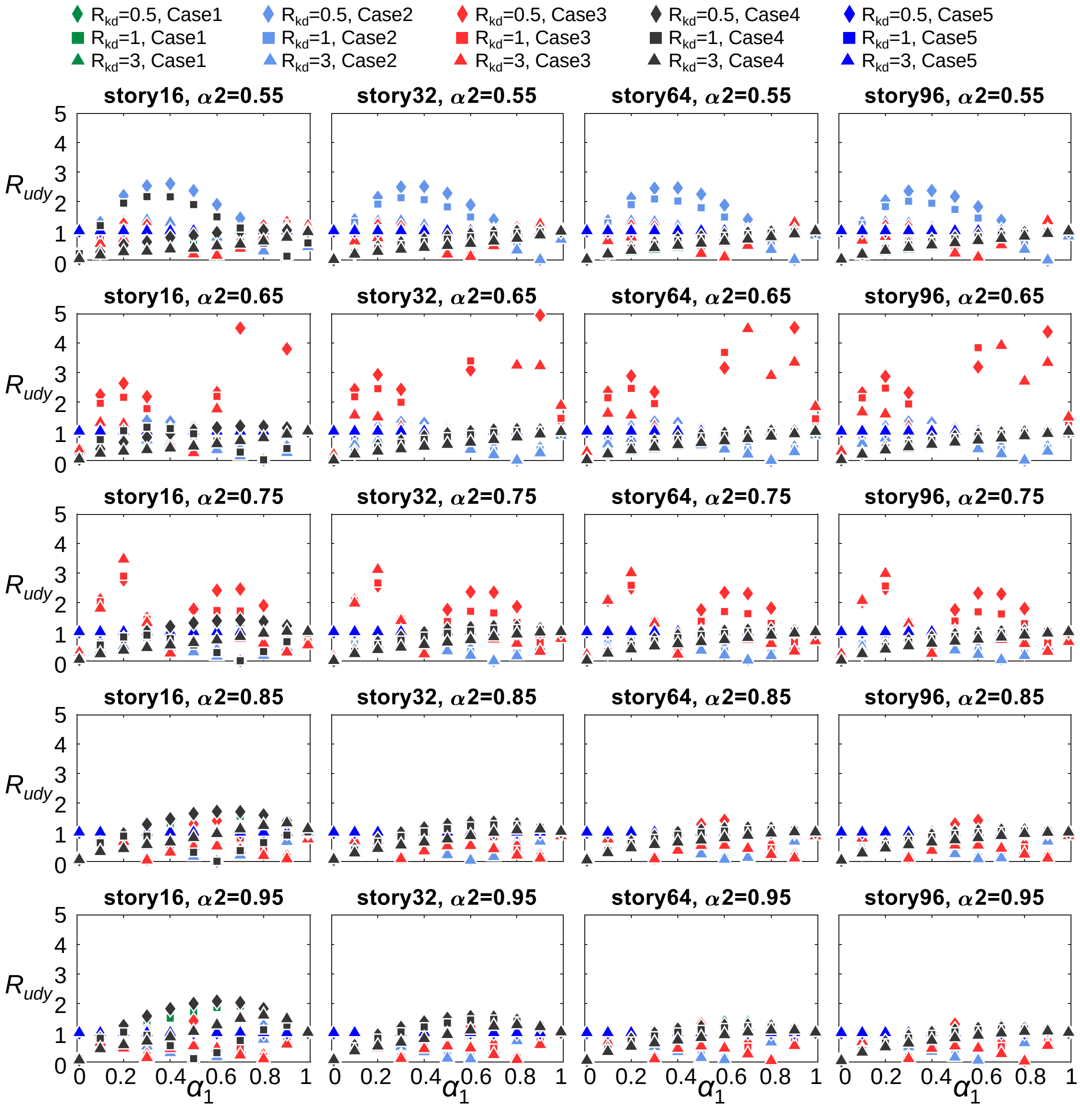


Figure 6.3.13 Relationship between R_{udy} and α_1 with various R_{kd} and BRB yield cases

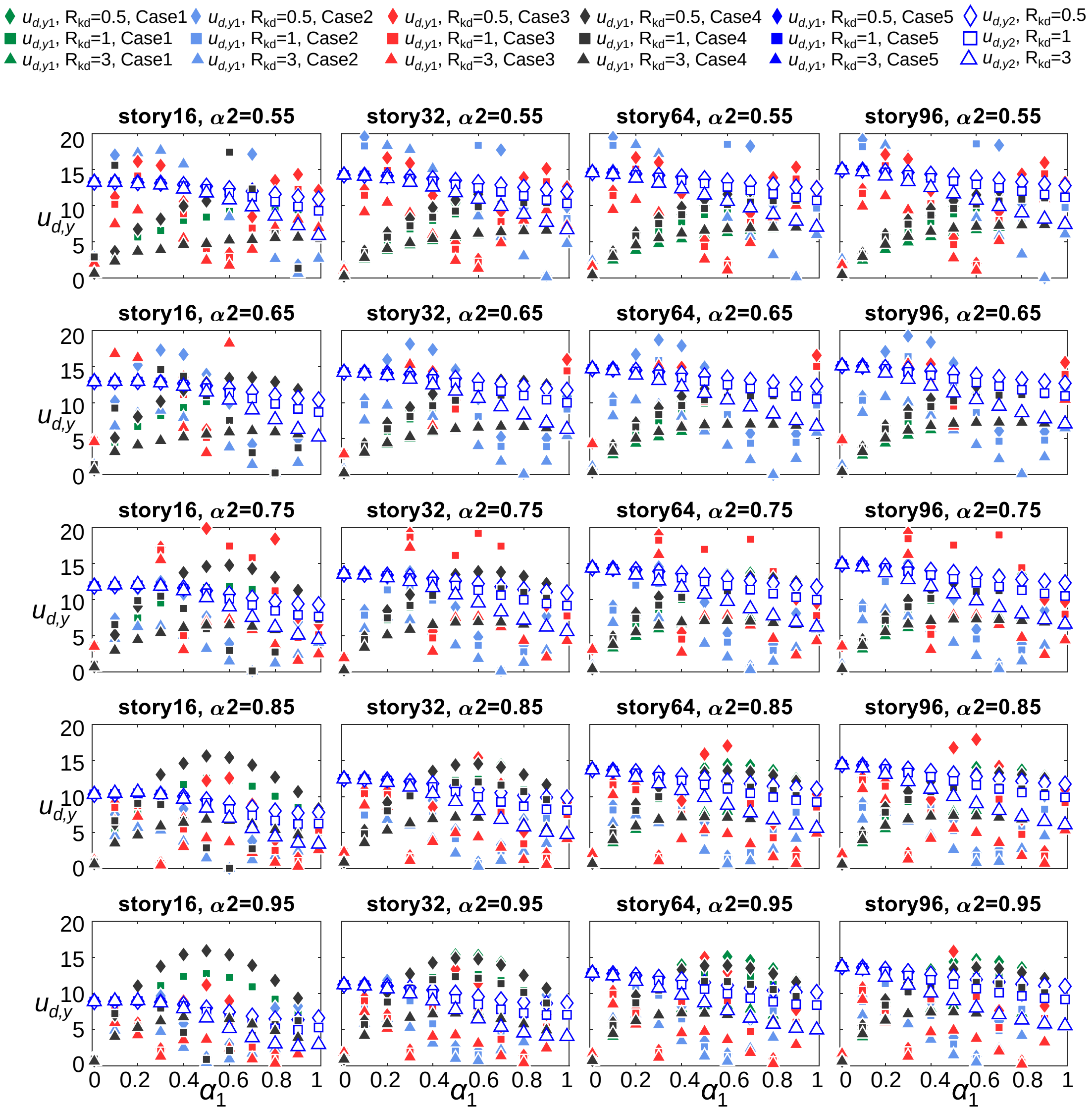


Figure 6.3.14 Relationships between $u_{d,y1}$, $u_{d,y2}$ and α_1 under different R_{kd} and BRB yield cases

Figure 6.3.15 and Figure 6.3.16 show the relationships between $\theta_{\max}$ and α_1 calculated from SA and NLRHA (BCJ-L2 ground motion), respectively. Based on the SA results, when $\alpha_2 = 0.85$ and 0.95, the reductions in $\theta_{\max}$ are the most. In addition, the larger R_{kd} value (stronger lower BRB stiffness) enhances the outrigger effect in reducing $\theta_{\max}$. Since the R_{udy} of Case3 sometimes gives very large value of $u_{d,y1}$, the BRB_1 could remain in elastic deformation or only slightly yield during an earthquake. Therefore, the effect in reducing seismic response of Case3 is weaker if compared with the other BRB yield cases. As seen from the Case3 responses, the $\theta_{\max}$ increases as the α_1 is close to 1, which indicates the BRB_1 that deforms elastically could reduce the energy dissipation efficiency of BRB_2 when they are close to each other. The Case2 gives large $u_{d,y1}$ when α_2 is low (0.65) and very small $u_{d,y1}$ when α_1 is large (0.6

to 0.8). Too large $u_{d,y1}$ could result in very low efficiency in reducing $\theta_{\max}$, which is similar to the Case3. However, too small $u_{d,y1}$ would result in low energy dissipation efficiency and less capable of reducing $\theta_{\max}$. For all the cases, the larger R_{kd} value leads to a samller $\theta_{\max}$ responses found from both SA and NLRHA. However, the amount of $\theta_{\max}$ reduction is not proportional to the increased R_{kd} value. Since the R_{udy} value of Case5 is similar to Case4, the Case5 shows similar responses in reducing $\theta_{\max}$ with Case1 and Case4. It should be noted that when the R_{kd} of the 32-story model is large, the trends of NLRHA results are different from the SA results. This could due to that the stiffer BRB_1 may amplify the responses from the modes other than the 1st mode, and the corresponding spectral accelerations of BCJ-L2 could be larger than that of design spectrum. Figure 6.3.17 and Figure 6.3.18 show the relationships between R_{CPD} value and α_1. The very small $u_{d,y1}$ values of Case2 and Case3 result in very large R_{CPD} values which may be difficult to be achieved by conventional BRB (R_{CPD} ratio approximates 600 to 800). The R_{CPD} ratios for BRB1 of Case5 are very small when α_1 is smaller than 0.5. This suggests that the Case5 suggests too large $u_{d,y1}$ for BRB_1, so that the BRB_1 could not properly yield if compared with Case 1 and Case4. As shown in Figure 6.3.17, when the lower outrigger is close to the upper one (when α_1 is close to 1), the R_{CPD} ratio of BRB_2 increases. This could be due to that after the BRB_1 with small $u_{d,y1}$ (small strength) of Case2 and Case3 yields soon after the earthquake begins, the BRB_2 is required to sustain the majority earthquake force demands, and thus gains a high value of R_{CPD} ratio. In summary, as the Case2 and Case3 do not show better seismic performance than the others, and the Case1 and Case4 generally show good responses in reducing $\theta_{\max}$ and comparable R_{CPD} ratios between BRB_1 and BRB_2, and the Case4 is recommended for design practices.

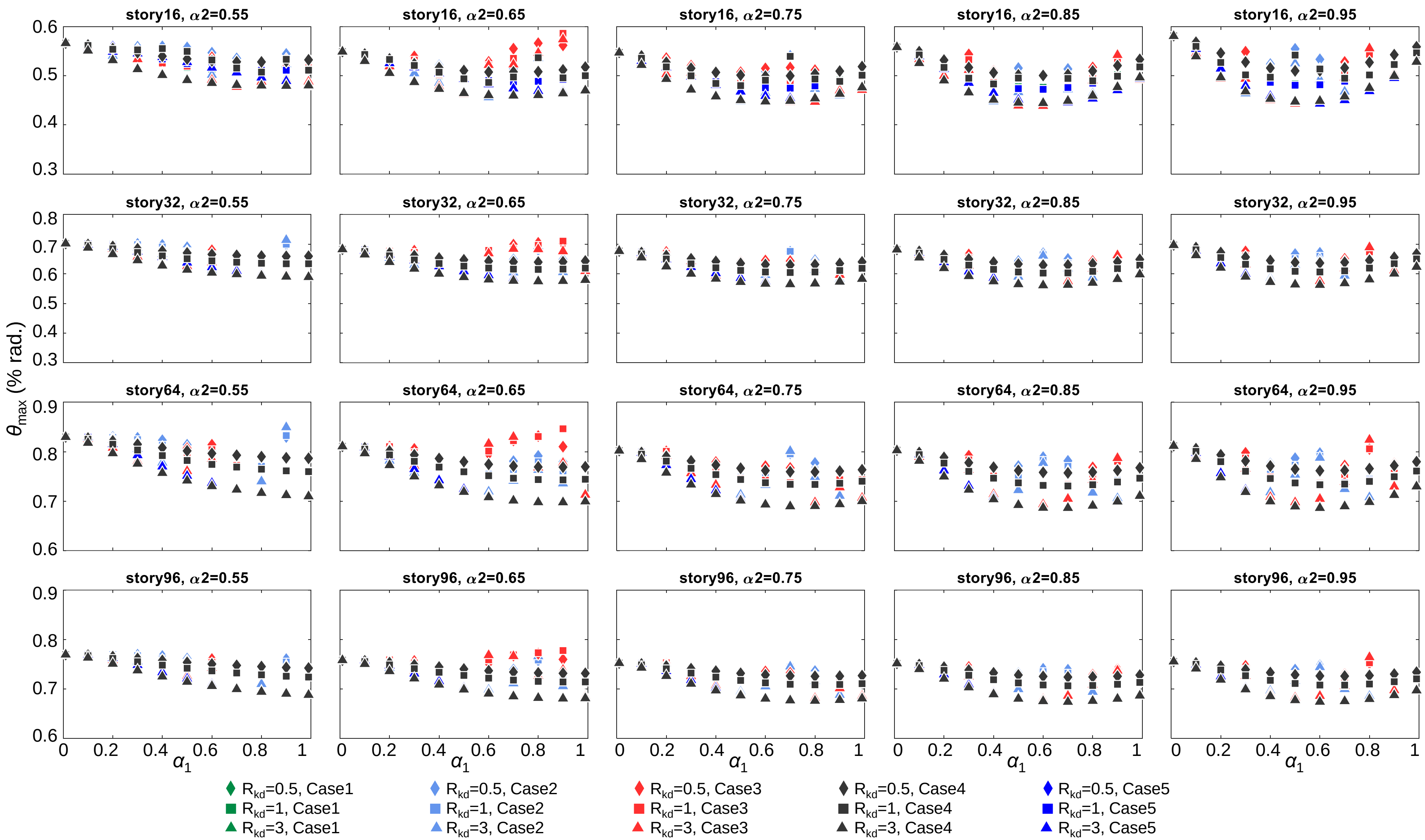


Figure 6.3.15 Relationships between θ_{max} and α_1 with various R_{kd} and BRB yield cases calculated from SA

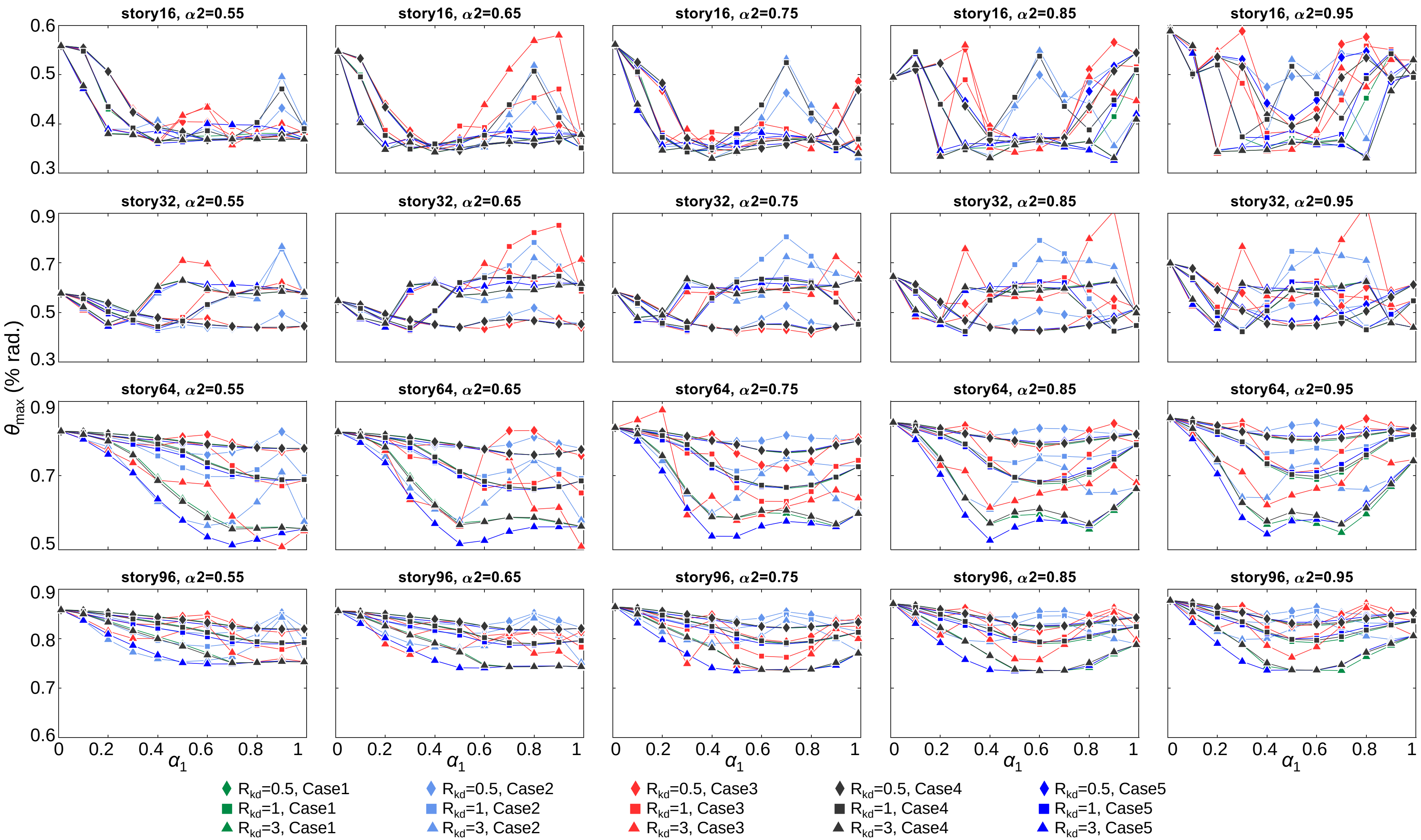


Figure 6.3.16 Relationships between $\theta_{\max}$ and α_1 with various R_{kd} and BRB yield cases calculated from NLRHA

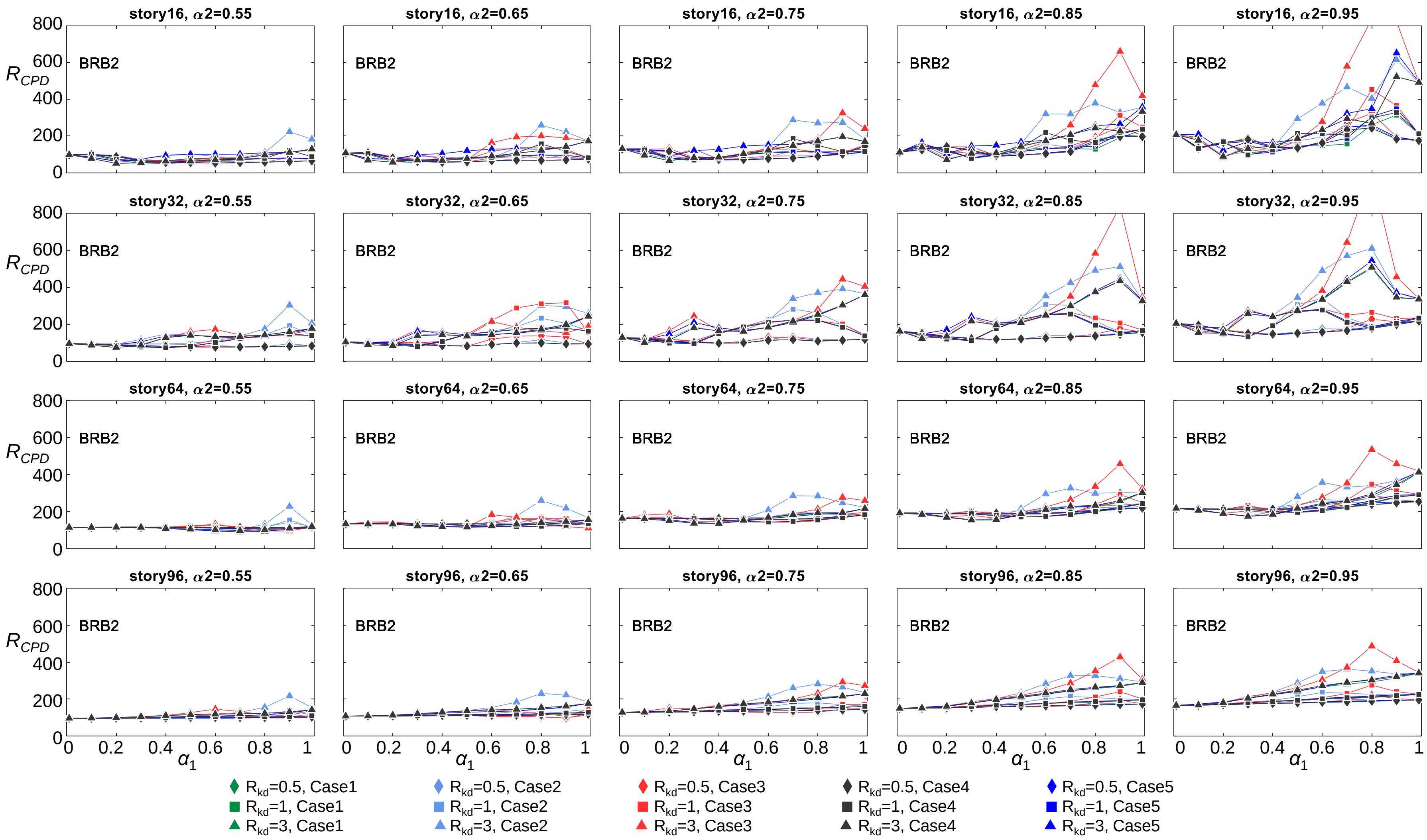


Figure 6.3.17 R_{CPD} ratio of BRB_2 calculated from NLRHA

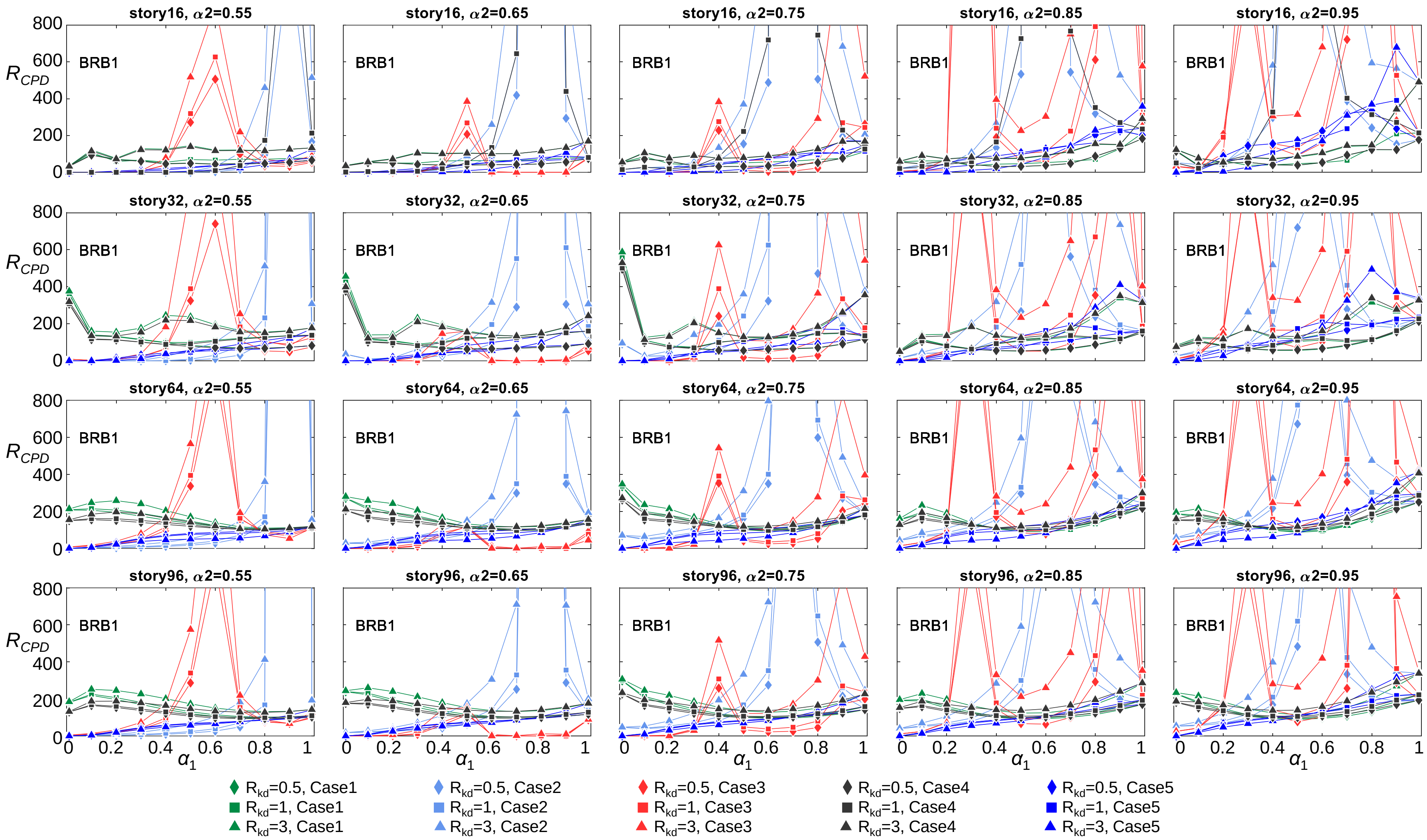


Figure 6.3.18 R_{CPD} ratio of BRB_1 calculated from NLRHA

6.4 BRB ENERGY DISSIPATION EFFICIENCY BASED ON SA

The outrigger elevations and BRB yield deformations affect the seismic response by the BRB's energy dissipation efficiency. For example, when the outrigger is very close to the structure base, the small core structure rotation is not able to generate sufficient axial deformation demand on BRB so that the energy dissipation efficiency is low. Also, if the BRB yield deformation is too large so that the BRB remains in elastic deformation during an earthquake, the energy dissipation efficiency would be low. The ratio of E_d to E_s (E_d/E_s) is used to indicate the BRB energy efficiency. In the SA procedure, E_d and E_s are the energy dissipated by the BRB-outrigger system per loop and the total strain energy of the system, respectively, as shown in Figure 4.3.2. In the NLRHA procedure, E_d and E_s are the energy dissipated by BRBs and the total input energy, respectively. The larger E_d/E_s value suggests the BRB dissipates a larger portion of input seismic energy, thus the energy dissipation efficiency is high. The E_d is 0 if the BRB remains elastic deformation. The larger value of E_d/E_s also suggests a larger equivalent damping ratio is developed and thus is capable to dissipate a larger amount of energy through the BRB's hysteretic response. As described in Chapter 4, in the SA procedure, the responses of each mode are calculated first, then superposed by SRSS combination. Therefore, the E_d/E_s value under each mode responses can be calculated. The effects of outrigger elevation, θ_r, BRB yield cases, R_{kd}, and R_{d2c} on the E_d/E_s value are studied from SA procedure in the following sections.

6.4.1 Effect of BRB yield deformation ratio and roof yielding drift ratio

The effects of R_{udy} of the five cases and θ_r on seismic response are investigated in this section. The E_d/E_s values of a 32-story model with θ_r equal to 1/150, 1/350, 1/550, 1/750, and 1/950 with five BRB yield deformation cases are calculated by SA procedure. Meanwhile, the BRB parameters are fixed as $R_{d2c} = 1$, $R_{kd} = 1$, and $R_{dt1} = R_{dt2} = 0.1$. Figure 6.4.1 and Figure 6.4.2 show the yield deformations of BRB in the upper ($u_{d,y2}$) and lower ($u_{d,y1}$) outriggers, respectively. The BRBs in the upper and lower outriggers are knowns as BRB_2 and BRB_1, respectively. The $u_{d,y2}$ are the same in every BRB yield case and becomes smaller when θ_r is smaller. The $u_{d,y2}$ reaches the maximum when α_2 is approximately 0.6 and when α_1 is 0, and becomes smaller when the lower outrigger is approaching the upper outrigger (when α_1 is close to 1). This is because that although the core structure generates the greatest rotation at building top in the SRSS deformed shape, however, the effective perimeter column axial stiffness for the upper outrigger (k_c/α_2) becomes smaller when α_2 is larger. The $u_{d,y1}$ distributions of Case1 and Case4 are similar to the 1st mode deformed shape is similar to the SRSS combined deformed shape. For the Case1 and Case4, the $u_{d,y1}$ becomes smaller when θ_r decreases and reaches the maximum when α_2 is 1 and α_1 is approximately 0.5. The contour of $u_{d,y1}$ = 0 for Case2 and Case3 indicates that the lower outrigger locates at the inflection point. The negative values suggest that directions of BRB_1 and BRB_2 deformations are opposite to each other.

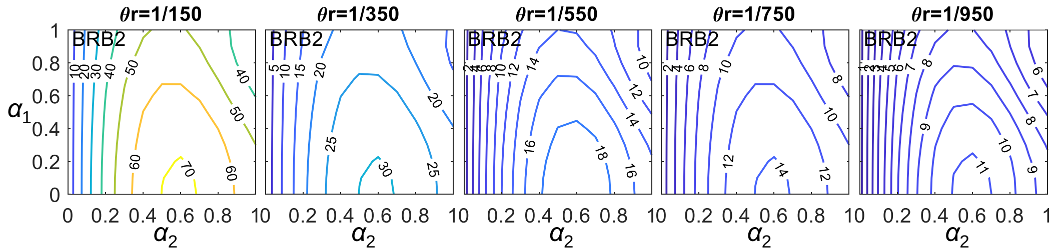


Figure 6.4.1 The $u_{d,y2}$ under different yield cases and θ_r (unit: mm)

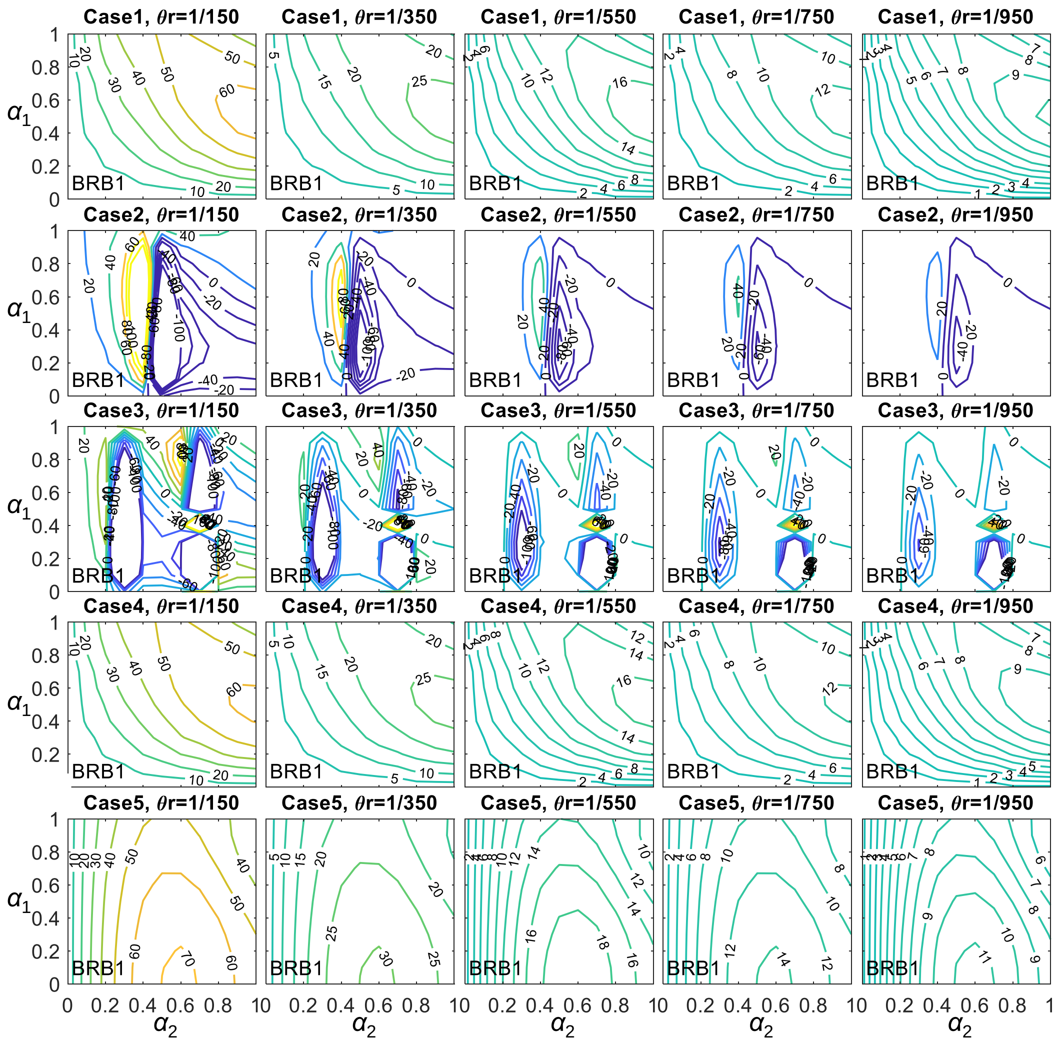


Figure 6.4.2 The $u_{d,y1}$ under different yield cases and θ_r (unit: mm)

Figure 6.4.3 to Figure 6.4.6 present the E_d/E_s values (in percentage) of the BRB_2 (E_{d2}/E_s) and BRB_1 (E_{d1}/E_s) calculated from the 1st and 2nd mode responses with different yield cases and θ_r. For the 1st mode response (Figure 6.4.3 and Figure 6.4.4), the E_{d2}/E_s distributions only slightly change between different yield cases and θ_r, as the $u_{d,y2}$ is the same in every yield case. The E_{d2}/E_s reaches the maximum when α_2 is approximately 0.6 to 0.8 and becomes smaller when α_1 is close to 1. When θ_r equals 1/350, the large E_{d2}/E_s indicates the energy dissipation efficiency is high. The low E_{d2}/E_s values in the cases of θ_r = 1/150 suggest that the $u_{d,y2}$ is large, and the BRB_2 may only slightly yield. The low E_{d2}/E_s values are also found in the cases of θ_r = 1/950. However, this is because the small $u_{d,y2}$ with small yield force capacity results in a small amount of E_{d2}/E_s even the BRB_2 would easily yield. The shapes of E_{d1}/E_s distribution under different θ_r and BRB yield cases are similar. The E_{d1}/E_s reaches the maximum when α_2 is 1 and α_1 is approximately 0.5 to 0.7 for the Case1 and Case4. The Case2 and Case3 result in more complicate E_{d1}/E_s distributions as the corresponding $u_{d,y1}$ and $u_{d,y2}$ are in wide varieties. The Case5 forces $u_{d,y1}$ to equal $u_{d,y2}$. The $u_{d,y1}$ of Case5 is generally larger than the Case1 and Case4. Therefore, the E_{d1}/E_s of Case5 is smaller than Case1 and Case4, which suggests the BRB_1 in Case5 may be overdesigned. For the 2nd mode responses (Figure 6.4.5 and Figure 6.4.6), the E_{d2}/E_s values when θ_r = 1/150 indicate the BRB_2 remains in elastic deformation due to that the large θ_r results in large $u_{d,y2}$. The E_{d2}/E_s is larger when θ_r decreases. In addition, when α_2 is within 0.2 and 0.6, the E_{d2}/E_s is 0. This is because the upper outrigger elevation is near the inflection point of the core structure and thus results in a small BRB axial deformation. For the 2nd mode responses, the E_{d2}/E_s values are larger when α_2 is approximate 0.9 and α_1 is approximately 0.5, and the E_{d1}/E_s values are larger when α_2 and α_1 are approximate 0.5. In addition, the E_{d2}/E_s and E_{d1}/E_s values are larger when θ_r is smaller. This suggests that the contribution of equivalent damping ratio from higher mode responses can be increased only when the $u_{d,y1}$ and $u_{d,y2}$ are small enough. For the 3rd mode responses, the $E_{d2}/E_s = 0$ and $E_{d2}/E_s = 0$ (therefore no figure is shown) suggest that both BRB_1 and BRB_2 do not dissipate energy under the 3rd mode response. Figure 6.4.7 shows the $\theta_{\max}$ distributions with different BRB yield cases and θ_r. When θ_r is smaller than 1/550, the $\theta_{\max}$ can be reduced to lower than 0.65% rad. for each yield case. The models with θ_r = 1/150 result in relatively large $u_{d,y2}$ and $u_{d,y1}$ so that the BRBs only develop slight inelastic deformations. The low energy dissipation efficiency (low E_{d2}/E_s and E_{d1}/E_s values) could not effectively reduce the seismic response. Although the θ_r = 1/350 has the greatest amount of E_{d2}/E_s and E_{d1}/E_s values among the others, the $\theta_{\max}$ response is not the smallest. The large values of E_{d1}/E_s and E_{d2}/E_s generally result in good reductions in $\theta_{\max}$. In summary, for single BRB-outrigger system, the α locates approximate 0.6 to 0.7 can result in good energy dissipation efficiency. For dual BRB-outrigger system, the α_1 and α_2 locate approximate 0.5 and 0.7, respectively, can result in good energy dissipation efficiency. The BRB yield deformation ratio does not significantly affect the energy dissipation efficiency. However, the Case1 and Case4 are recommended as the $u_{d,y1}$ lie in a reasonable range. The smaller θ_r value makes it easier to yield te BRB. However, too large and too small θ_r would reduce the BRB energy dissipation efficiency. Based on the analysis results, the θ_r = 1/350 and 1/750 can lead to good energy dissipation efficiencies.

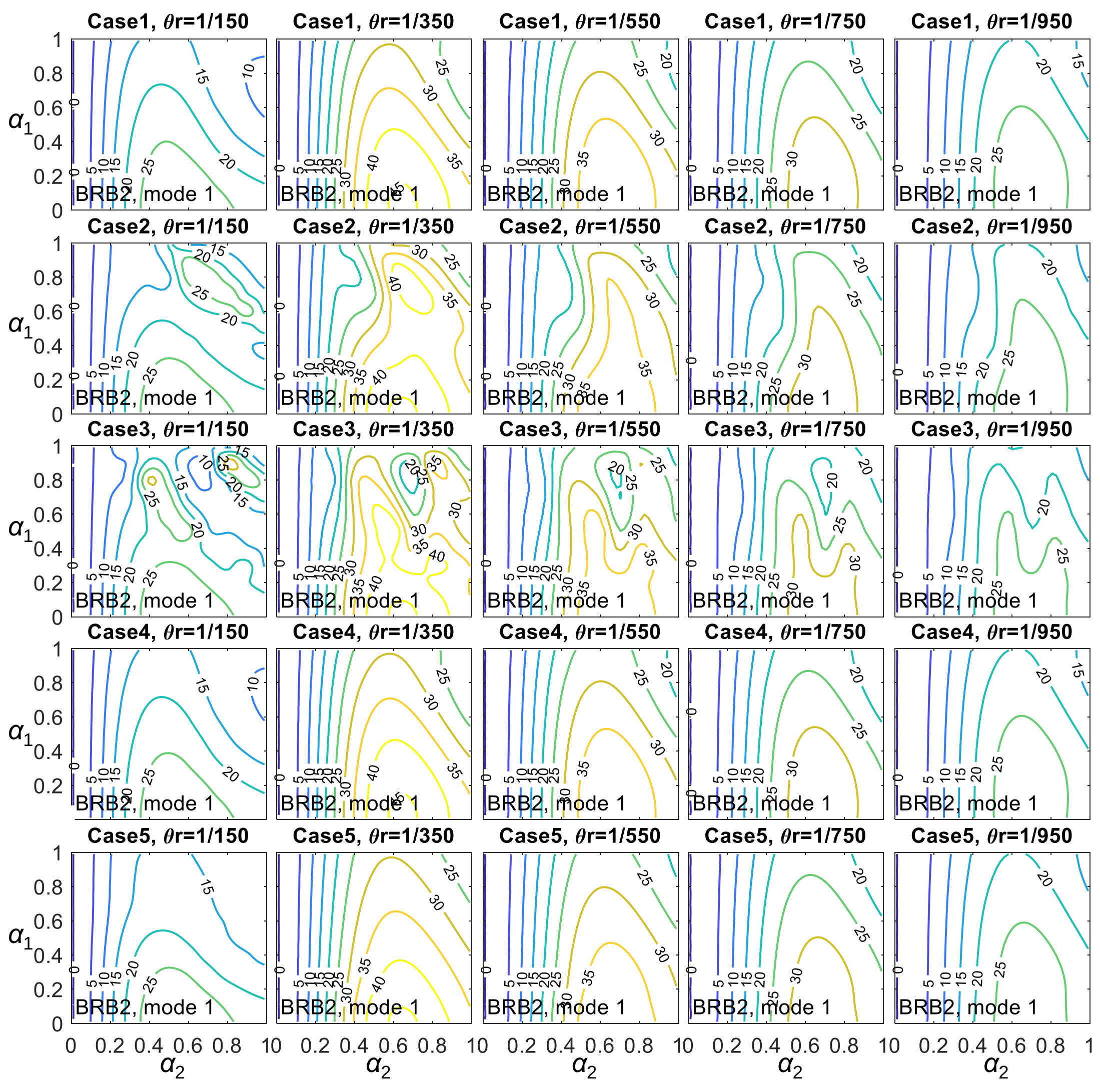


Figure 6.4.3 The E_{d2}/E_s distributions of the 1st mode response under different BRB yield cases and θ_r

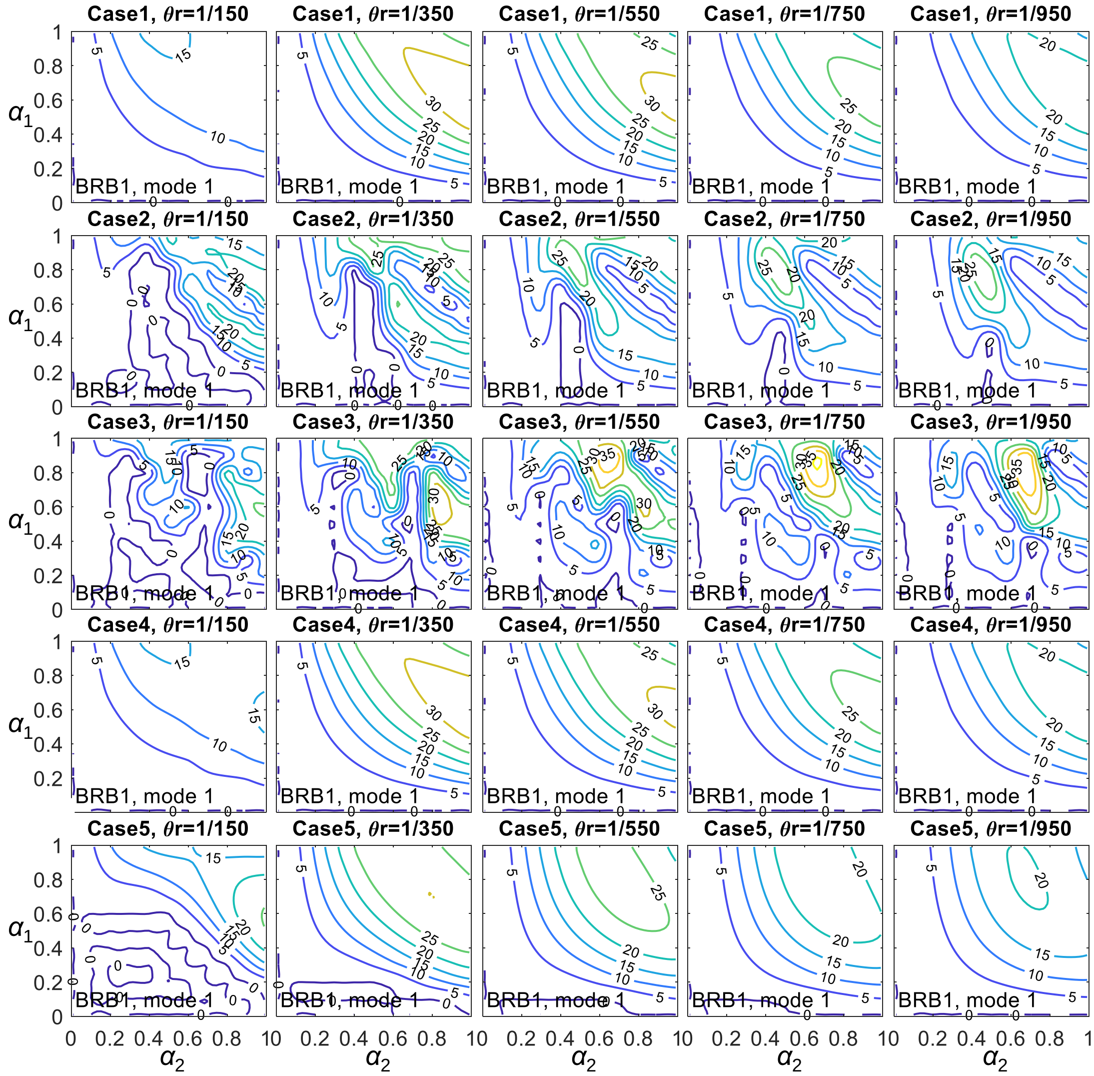


Figure 6.4.4 The E_{d1}/E_s distributions of the 1st mode response under different BRB yield cases and θ_r

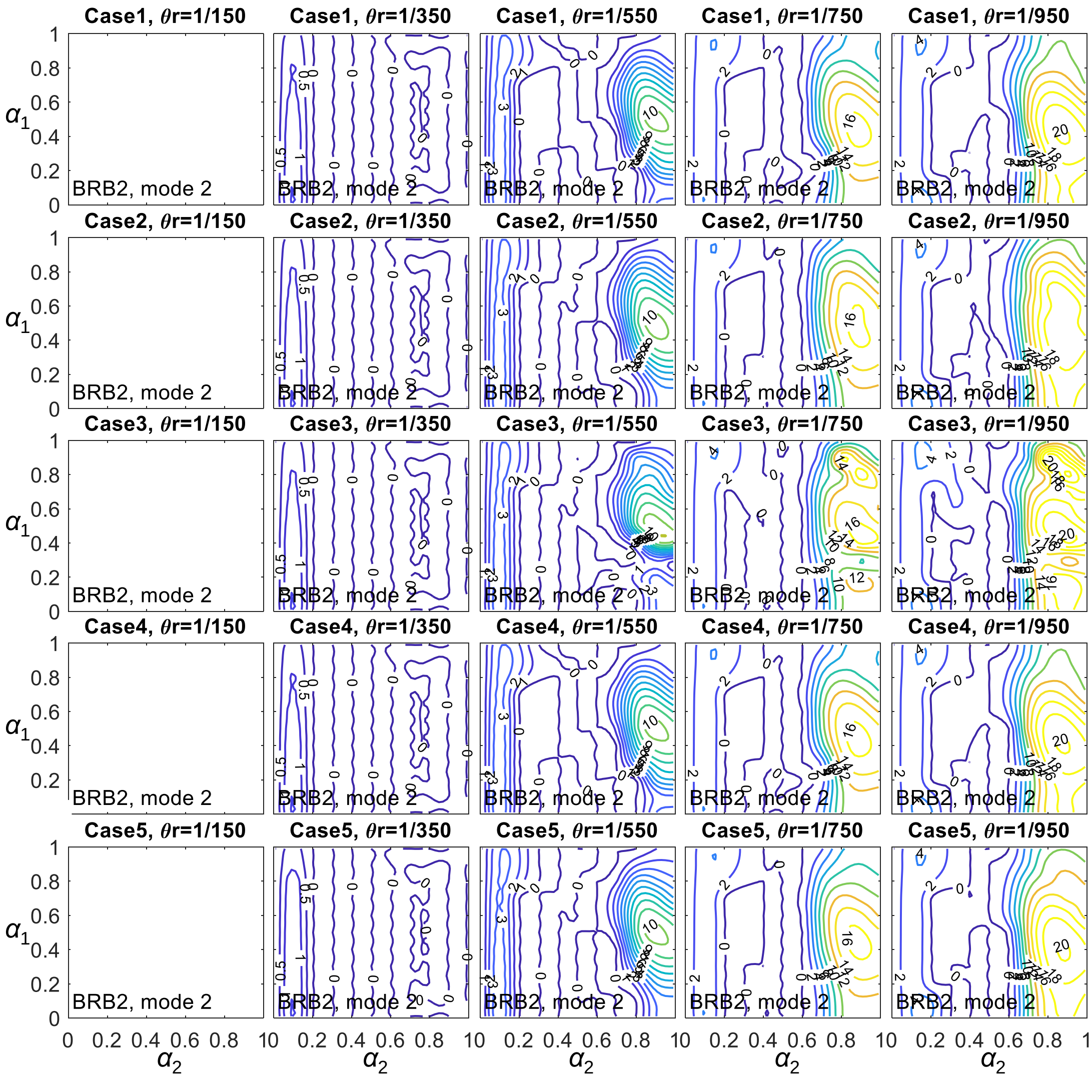


Figure 6.4.5 The E_{d2}/E_s distributions of the 2[nd] mode response under different BRB yield cases and θ_r

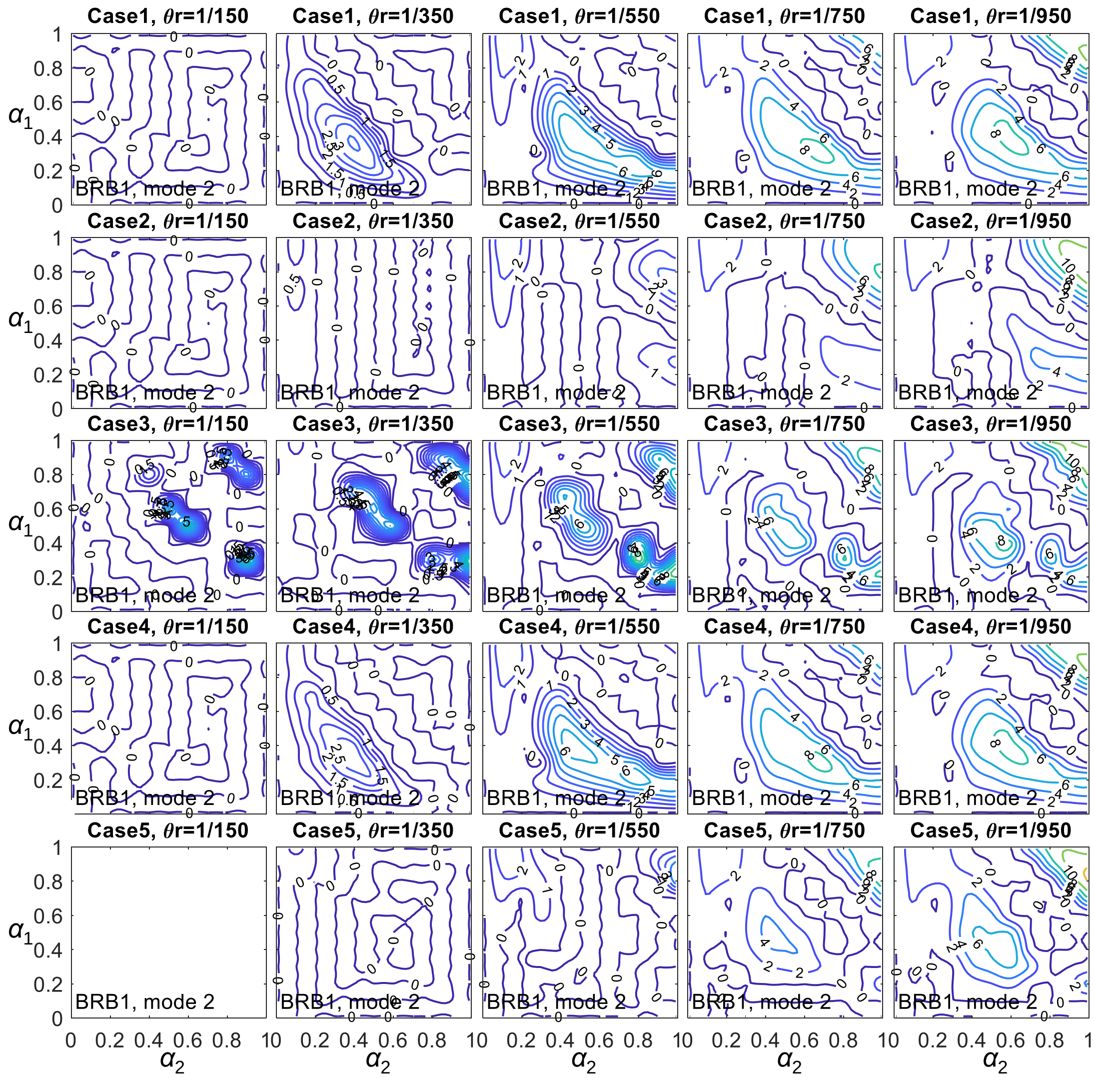


Figure 6.4.6 The E_{d1}/E_s distributions of the 2nd mode response under different BRB yield cases and θ_r

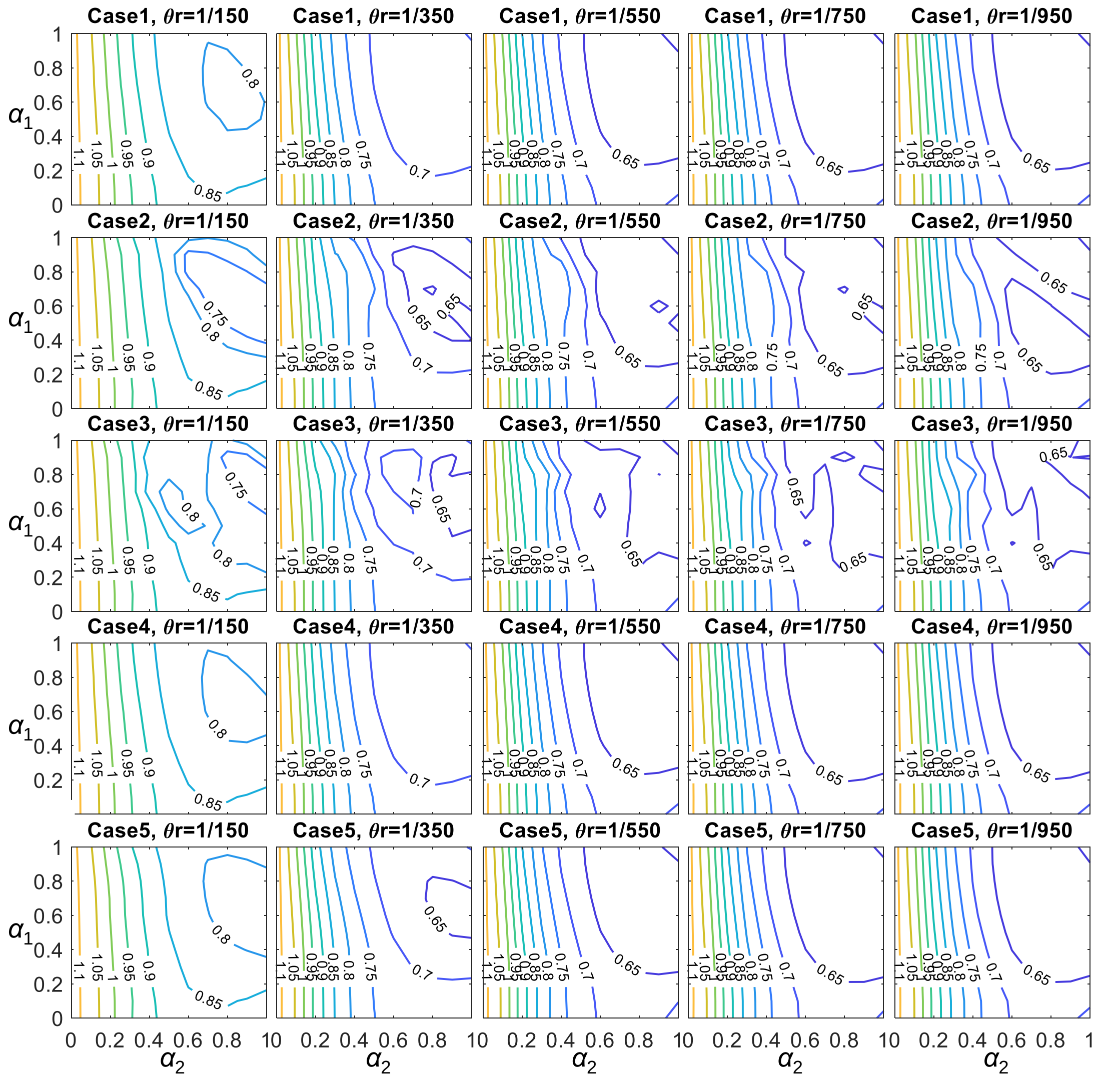
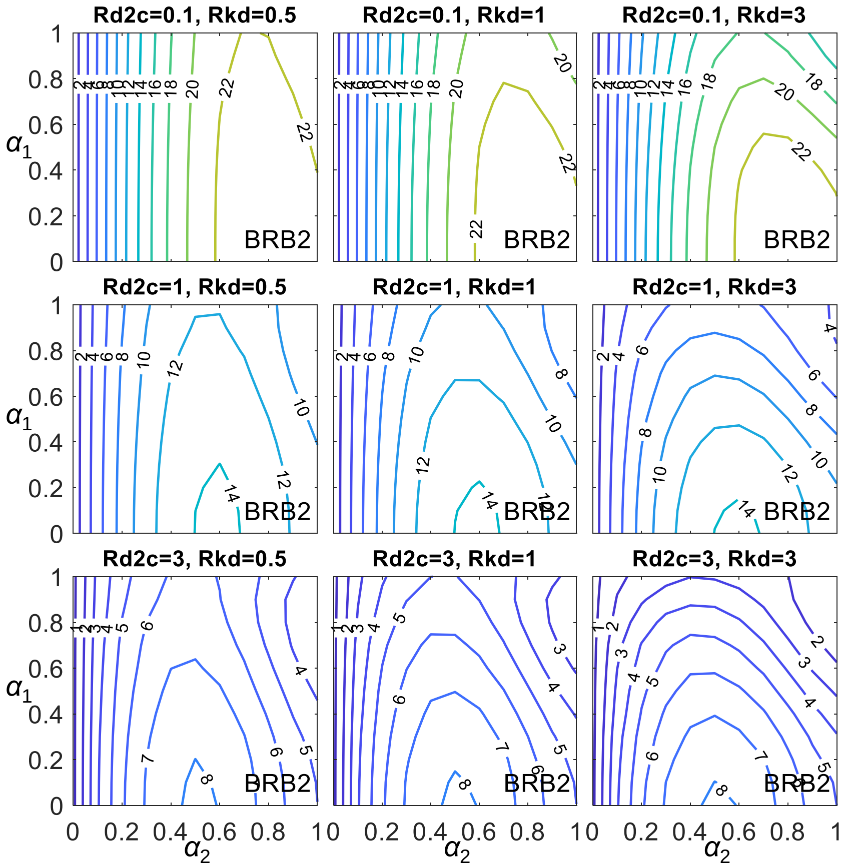


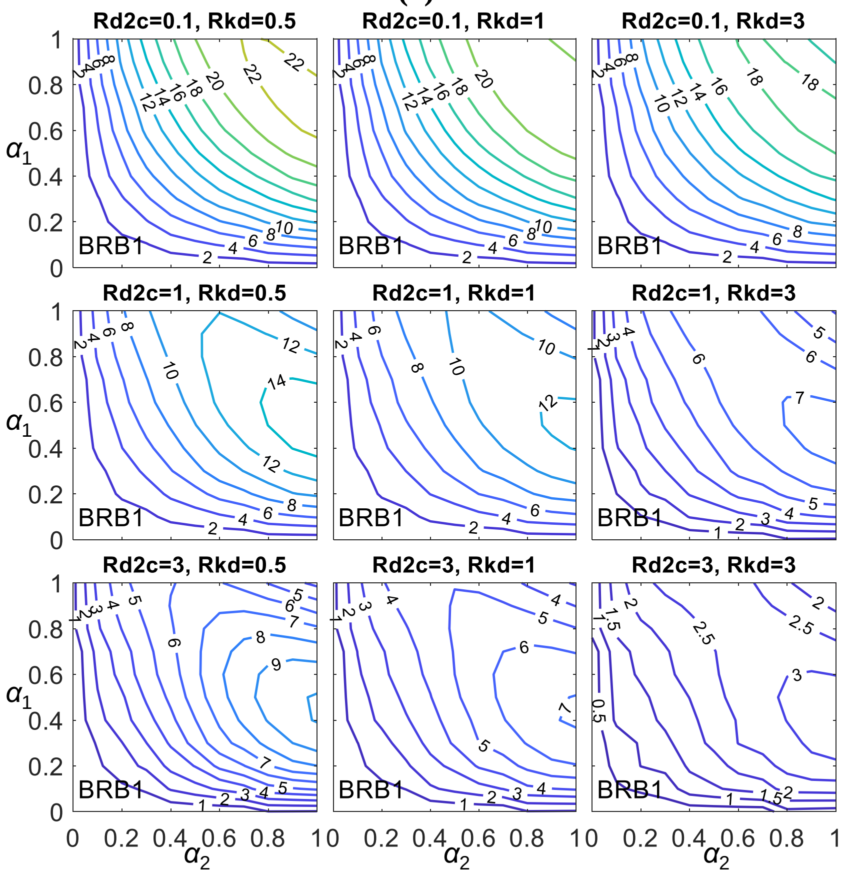
Figure 6.4.7 The θ_{max} distributions under different BRB yield cases and θ_r

6.4.2 Effect of BRB stiffness parameters

The 32-story model with θ_r = 1/750 and BRB yield Case4 is used to investigate the effect of BRB stiffness parameters on energy dissipation efficiency by varying R_{d2c} between 0.1, 1, and 3 and, R_{kd} between 0.5, 1, and 3. Figure 6.4.8a and Figure 6.4.8b show the $u_{d,y2}$ and $u_{d,y1}$ distributions, respectively, with various R_{d2c} and R_{kd}. The $u_{d,y2}$ varies primary with α_2, and reaches the maximum when α_2 is approximately 0.5 to 0.8. The $u_{d,y2}$ becomes smaller when α_1 is close to 1. The $u_{d,y1}$ reaches the maximum when α_2 is 1 and α_1 is approximately 0.5 to 0.8. The larger value of R_{d2c} results in smaller $u_{d,y2}$ and $u_{d,y1}$, and the larger R_{kd} results in smaller $u_{d,y1}$. When R_{kd} = 1, the $u_{d,y1}$ and $u_{d,y2}$ are similar.



(a)

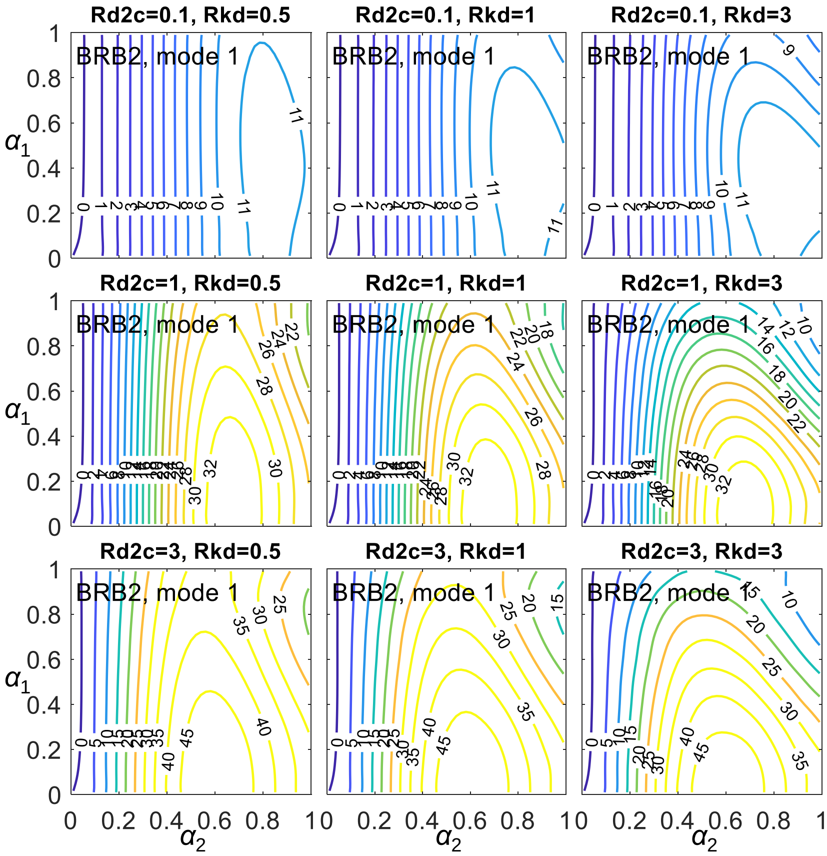


(b)

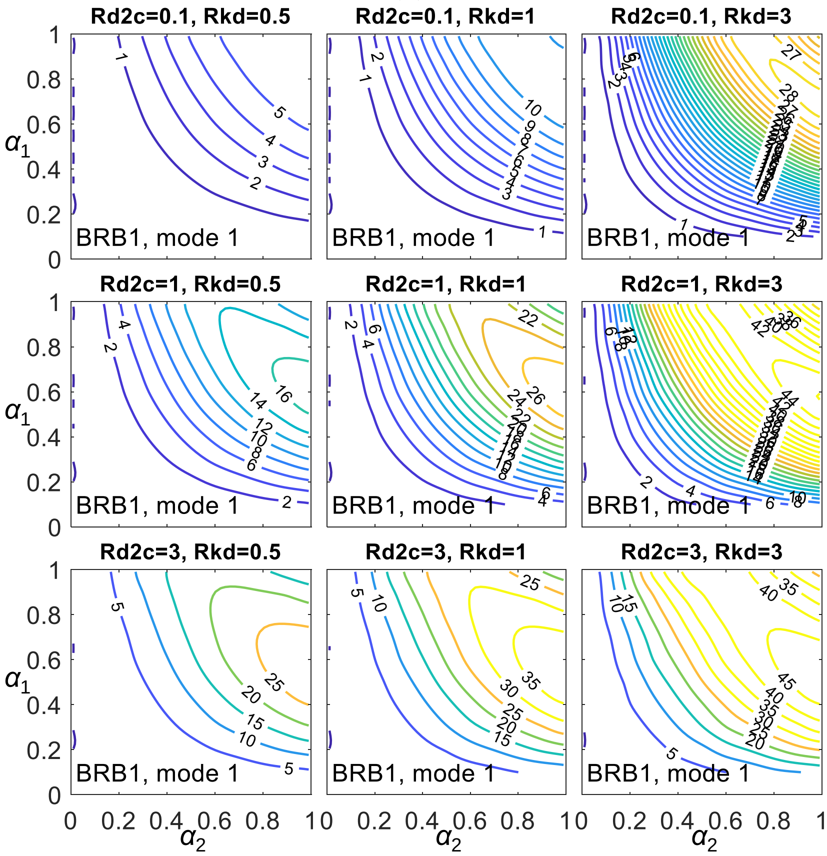
Figure 6.4.8 The (a) $u_{d,y2}$ and (b) $u_{d,y2}$ under different R_{d2c} and R_{kd} (unit: mm)

Figure 6.4.9 and Figure 6.4.10 show the E_{d2}/E_s and E_{d1}/E_s (in percentage) distributions of the 1st and 2nd mode responses, respectively, when R_{d2c} varies between 0.1, 1, and 3 and R_{kd} varies between 0.5, 1, and 3. For the 1st mode response (Figure 6.4.9), The E_{d2}/E_s primarily changes with α_2, and reaches the maximum when α_2 is approximately 0.6 to 0.8 and α_1 is 0. Although the core structure rotation is larger at a higher elevation, the effective column axial stiffness (k_c/α_2) becomes smaller when outrigger elevation is higher. This could explain that the maximum E_{d2}/E_s does not occur at the top of the core structure. In addition, the stiffer lower BRB-outrigger (larger R_{kd}) decreases the values of E_{d2}/E_s. When the lower BRB-outrigger is approaching to the upper outrigger (when α_1 is close to 1), the E_{d2}/E_s decreases and the E_{d1}/E_s increases. The stiffer lower BRB-outrigger (larger R_{kd}) results in larger E_{d1}/E_s value. As shown in Figure 6.4.9, when α_2 is smaller than 0.2, the BRB_1 almost remains elastic deformation, which suggests that when α_2 is smaller than 0.2, the BRB_1 would not develop energy dissipation mechanism. Based on the analysis results, under the 1st mode response, the best outrigger elevations to mobilize the BRB hysteretic responses are approximately α_2 = 0.6 and α_1 = 0.6. For the 2nd mode responses (Figure 6.4.10), the E_{d2}/E_s is 0 when α_2 is approximately 0.2 to 0.6. This is because the α_2 is close to the inflection point of the core structure. The stiffer lower BRB-outrigger (larger R_{kd}) increases the values of E_{d2}/E_s, the E_{d2}/E_s reaches the maximum when α_2 is approximately 0.9 and α_1 is approximately 0.4 to 0.5. Under the 2nd mode response, the best outrigger elevations to mobilize the BRB hysteretic responses are approximately α_2 = 0.6 to 0.8 and α_1 = 0.4 to 0.5. For the 3rd mode responses, the E_{d2}/E_s is 0 and only a very small amount of E_{d1}/E_s appear in certain cases (therefore the figures are not shown).

Figure 6.4.11 shows the relationship between $\theta_{\max}$ and outrigger elevations with different R_{d2c} and R_{kd}. The larger R_{d2c} and R_{kd} result in a smaller $\theta_{\max}$ response. The outrigger elevations that have the smallest $\theta_{\max}$ (α_2=0.6~0.8, α_1=0.5~0.6) agree with the elevations that result in large E_{d2}/E_s and E_{d1}/E_s values. The larger values of R_{d2c} and R_{kd} (larger BRB axial stiffness) generally result in greater energy dissipation efficiency and reduction in seismic response. As the 1st mode dominates the overall response, selecting the α_2 and α_1 that result in large E_{d2}/E_s and E_{d1}/E_s could also result in the smallest seismic response.

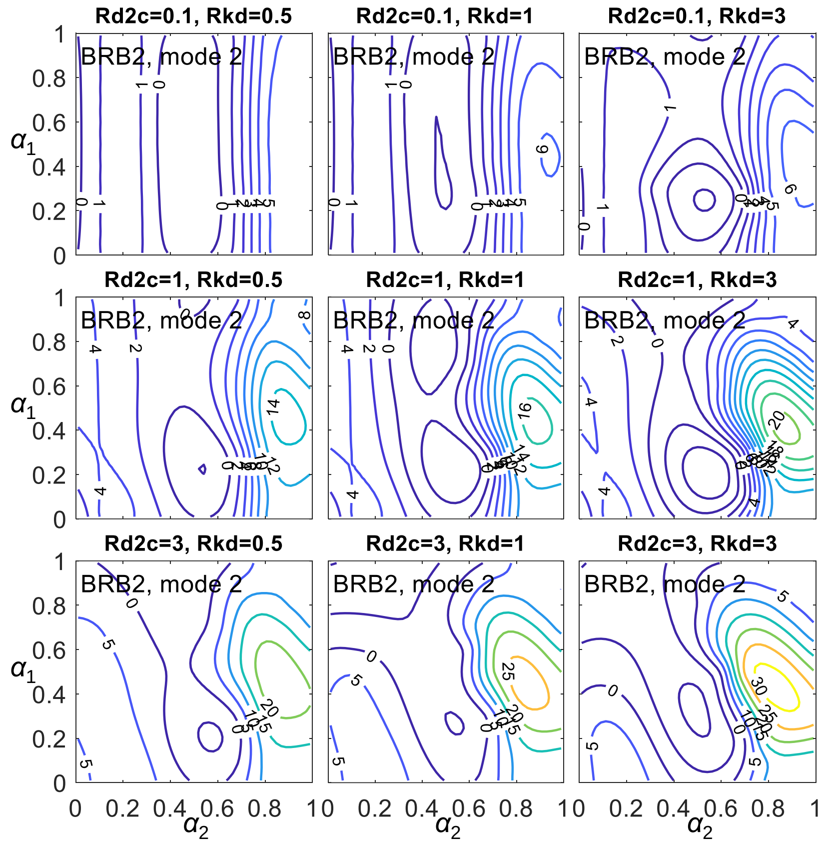


(a)

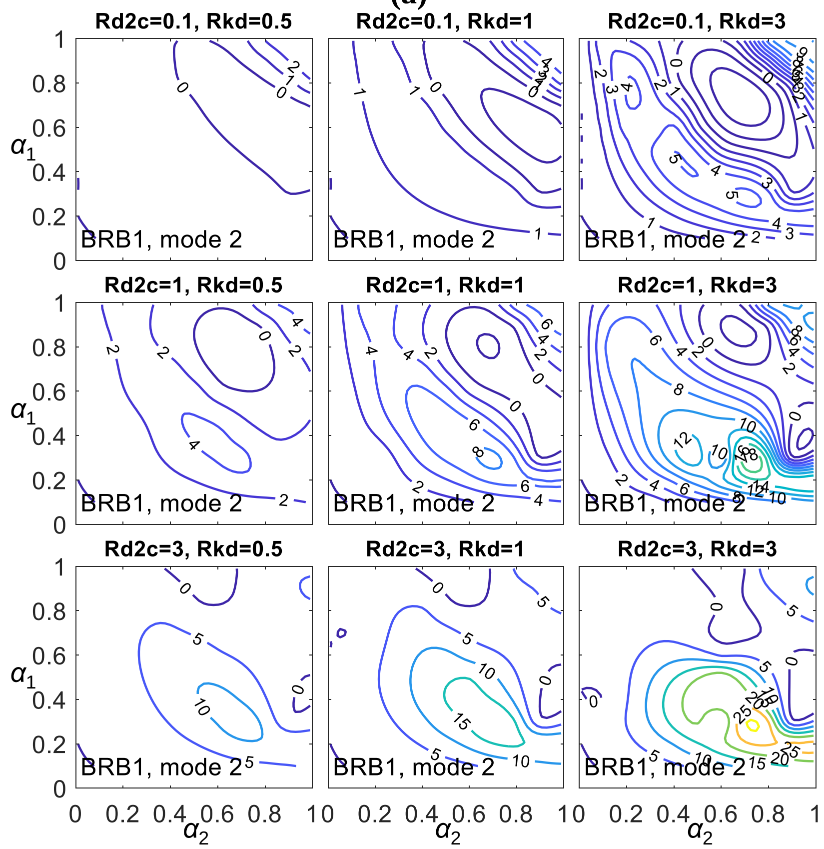


(b)

Figure 6.4.9 The (a) E_{d2}/E_s and (b) E_{d1}/E_s distributions of the 1st mode response under different BRB yield cases and θ_r



(a)



(b)

Figure 6.4.10 The (a) E_{d2}/E_s and (b) E_{d1}/E_s distributions of the 2nd mode response under different BRB yield cases and θ_r

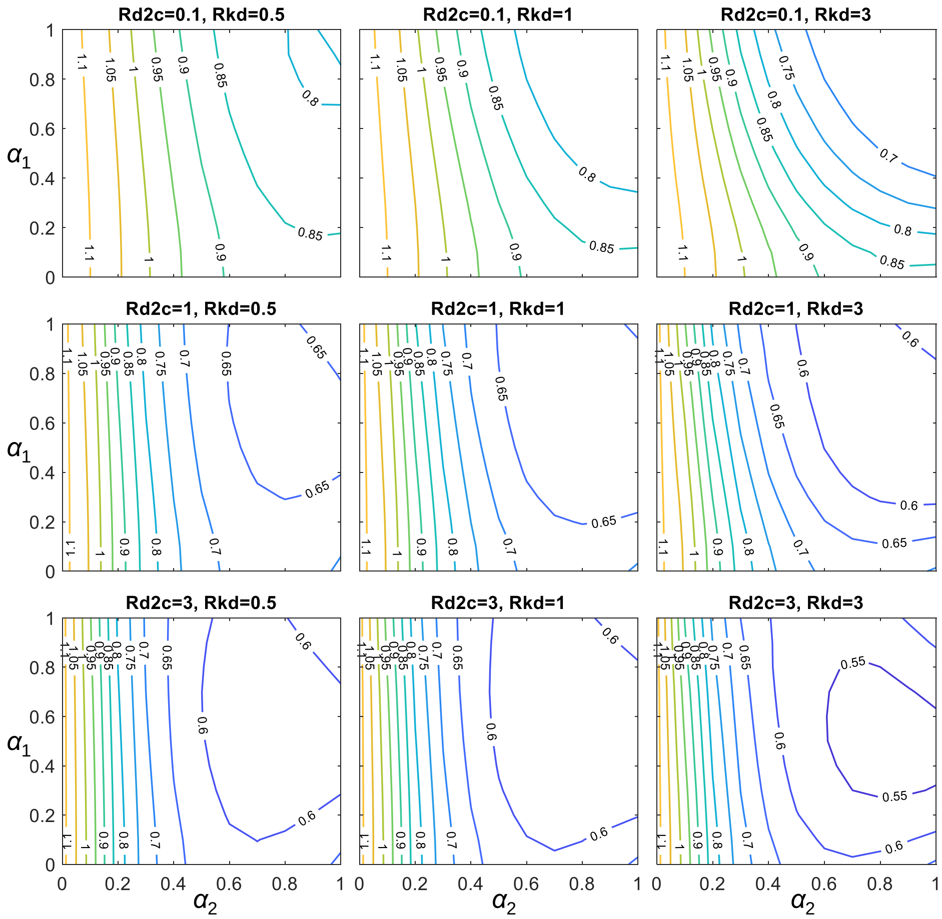


Figure 6.4.11 The θ_{max} distribution with different R_{d2c} and R_{kd} (unit: % rad.)

6.5 SUMMARY

This chapter investigates the outrigger effect when outrigger elevations vary. Five different cases in calculating BRB yield deformation are introduced. The BRB energy dissipation efficiencies with different outrigger elevations, BRB parameters, BRB yield deformations, and θ_r are discussed. The summaries are as follows:

(1) For the structure with dual BRB-outrigger, the Case1 and Case4 methods, which determine the R_{udy} (BRB yield deformation ratio) based on the 1st and the SRSS combined deformed shape, are recommended, so that BRB_1 and BRB_2 could yield approximately simultaneously.

(2) The θ_r is used to determine the amounts of $u_{d,y}$ and $u_{d,y2}$, and the $u_{d,y1}$ is determined by the R_{udy} accordingly. The larger θ_r results in greater BRB yield deformations, and vice versa. Too large θ_r could keep the BRB in elastic or only slight inelastic deformation, and thus lower the energy dissipation efficiency. Too small θ_r could make the BRB yield easily, and the BRB may use up its ductility life before the

end of the earthquake. Based on the analysis results, the θ_r values range between 1/350 and 1/750 are suggested.

(3) The outrigger truss should be stiff enough in order to provide sufficient axial deformation demands on the BRBs. Based on the analysis results, the seismic response is not significantly affected when R_{dt} ranges between 0.05 and 0.9. The R_{dt} = 0.1 is used in the parametric study.

(4) If compared with the structure without outrigger effect, the drops of vibration period are used to indicate the magnitude of the outrigger effect of structure with BRB-outriggers. Table 6.5.1 shows the optimal α, α_1, and α_2 in achieving minimum T_1 and T_2, and achieving maximum outrigger effects.

(5) The values of $u_{d,y}$, $u_{d,y2}$, and $u_{d,y1}$ indicate the deformation demand on the BRBs. The larger values of $u_{d,y}$, $u_{d,y2}$, and $u_{d,y1}$ suggest the outrigger configuration could apply the BRB deformation demands in a more efficient way. Table 6.5.1 shows the optimal α, α_1, and α_2 in achieving maximum BRB yield deformations.

(6) The vales of E_{d2}/E_s, E_{d1}/E_s (dual BRB-outrigger), and E_d/E_s (single BRB-outrigger) are used to indicate the BRB energy dissipation efficiency by performing SA. The larger values of E_{d2}/E_s, E_{d1}/E_s, and E_d/E_s suggest the energy dissipation efficiency is high. The values equal to zero if the BRB deforms elastically. Based on the SA results, the optimal α, α_1, and α_2 in achieving maximum values of E_{d2}/E_s, E_{d1}/E_s, and E_d/E_s are shown in Table 6.5.1.

Table 6.5.1 Optimal outrigger elevations in different indicators

optimal	min. T_1	min. T_2	max. $u_{d,y2}$ or $u_{d,y}$ (Case4)	max. $u_{d,y1}$ (Case4)	max. E_{d2}/E_s or E_d/E_s		max. E_{d1}/E_s	
					1st mode	2nd mode	1st mode	2nd mode
α	0.5~0.8	0.2, 0.8	0.5~0.7	-	0.6~0.8	0.9	-	-
α_1	0.8~0.9	0.2~0.3	0	0.5	0	0.5	0.5~0.7	0.5
α_2	0.6~0.8	0.5~0.8	0.6	1	0.6~0.8	0.9	1	0.5

6.6 REFERENCES

ANSI/AISC 341-16 (2016) *Seismic Provisions for Structural Steel Buildings, American Institute of Steel Construction*. doi: 111.

7

ANALYSIS RESULTS FOR OPTIMAL DESIGN

CHAPTER CONTENTS

7.1 Introduction 7-3
7.2 Single BRB-outrigger system..7-3
7.2.1 Maximum roof drift 7-3
7.2.2 Maximum inter-story drift 7-6
7.2.3 Maximum overturning moment 7-8
7.2.4 Maximum perimeter column axial force..7-11
7.2.5 BRB energy dissipation efficiency 7-13
7.2.6 Summary of optimal design for single BRB-outrigger system...........7-15
7.3 Dual BRB-outrigger system...7-18
7.3.1 Maximum roof drift 7-18
7.3.2 Maximum inter-story drift 7-29
7.3.3 Maximum overturning moment 7-39
7.3.4 Maximum perimeter column axial force..7-45
7.3.5 BRB energy dissipation efficiency 7-50
7.3.6 Summary of optimal design for dual BRB-outrigger system 7-60
7.4 Summary 7-60
7.5 References 7-62

7.1 INTRODUCTION

This chapter presents the analysis results of the optimal design for structures with single or dual BRB-outrigger system. The optimal design, in order to achieve minimum seismic responses, includes the optimal outrigger elevations and the relationships between BRB axial stiffness, perimeter column's axial stiffness, and core structure flexural stiffness. The maximum roof drift ($\theta_{\max}$), maximum inter-story drift ($\gamma_{\max}$), maximum overturning moment at core structure base ($M_{c,\max}$), and maximum perimeter column axial force ($C_{1,\max}$) are used to evaluate the seismic performance. Both the UM and DM models with the 16-, 32-, 64-, and 96-story models are used to perform SA and NLRHA. The detail of the analytical models and the analysis procedures are described in Chapter 3 and Chapter 4, respectively. The analysis results and the conclusions of optimal design are presented in this Chapter.

7.2 SINGLE BRB-OUTRIGGER SYSTEM

As introduced in Chapter 3, two methods are used in constructing analytical models with single BRB-outrigger system. The Met. I fixes R_{db} value as constant while changing α, therefore, the corresponding R_{dc} value varies with the changing α. The Met. II fixes R_{dc} as constant, therefore, the corresponding R_{db} varies with α. The analysis results from Met. I and Met. II are presented in the following sections.

7.2.1 Maximum roof drift

Figure 7.2.1 shows the results of $\theta_{\max}$ calculated from SA and NLRHA with BCJ-L2 ground motion. Both the SA and NLRHA show similar trends, and the SA well estimates the results of $\theta_{\max}$ compared to the NLRHA results. When α equals to 0, it is a core structure model without outrigger effect. Based on the analysis results, for a given value of R_{db} or R_{dc}, the $\theta_{\max}$ becomes minimum when α is approximately between 0.7 and 0.8 for Met. I, and from 0.5 to 0.7 for Met. II. In addition, both the Met. I and Met. II results indicate that the optimal α value is higher for the taller structure model (when S_{bc} is smaller). The trend of $\theta_{\max}$ with respect to α is similar to the 1st mode period trend as shown in Figure 6.2.1. This suggests that the outrigger elevation that has greatest outrigger effect on the system is approximately the optimal elevation in order to achieve minimum θ_{max}. Figure 7.2.1 also indicates that the larger R_{db} and R_{dc} lead to the smaller θ_{max} as the outrigger effect is more significant. However, both SA and NLRHA show that the θ_{max} when R_{db} equals to 5 and 10, and when R_{dc} equals to 5 and 10, are very close to each other. This suggests that a large R_{db} or R_{dc} (stiffer BRB) does not guarantee better performance in reducing θ_{max}. Figure 7.2.2 shows the θ_{max} reduction factors (reduction in θ_{max} if compared with the structure without an outrigger, in percentage). The NLRHA results are calculated from the average of the analysis using 8 scaled ground motions as described in Chapter 4. The θ_{max} reduction factors calculated from SA well agree with the NLRHA results. The θ_{max} reduction factor reaches the minimum (smallest θ_{max}) when α is approximately 0.6 to 0.8. The reduction in θ_{max} is less for the taller structure model with smaller S_{bc} value. In addition, the larger R_{db} or R_{dc} value leads to a smaller θ_{max} reduction factor. However, the reduction in θ_{max} is not proportional to the increase of R_{db} or R_{dc}. Figure 7.2.3 shows the relationships between the θ_{max} reduction factors and R_{db} or R_{dc} when α equals to 0.7. The SA results are close to the NLRHA results. The larger value of R_{db} or R_{dc} results in greater reductions in θ_{max}. However, the reductions in θ_{max} stop increasing when R_{db} or R_{dc} is greater than around 2 to 5. Since the BRB, perimeter column, and outrigger truss act in series, the increase of the ratio of BRB axial stiffness to the stiffness of perimeter column and outrigger truss could not efficiently increase the rotational stiffness provided by the BRB-outrigger.

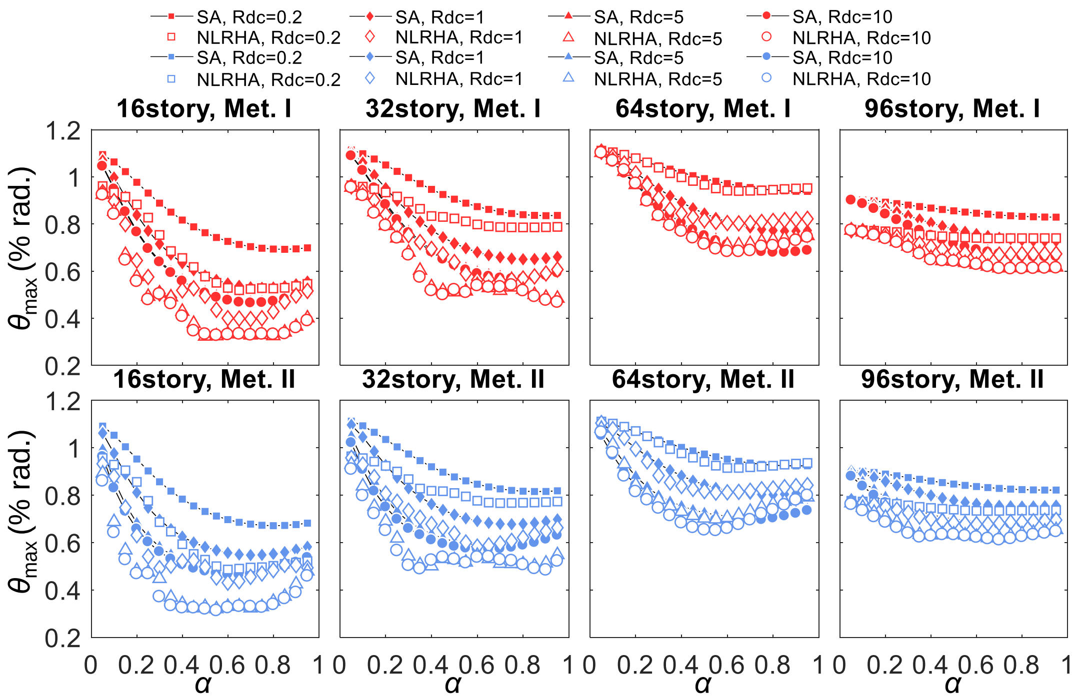


Figure 7.2.1 The θ_{max} distributions calculated from SA and NLRHA with BCJ-L2 ground motion

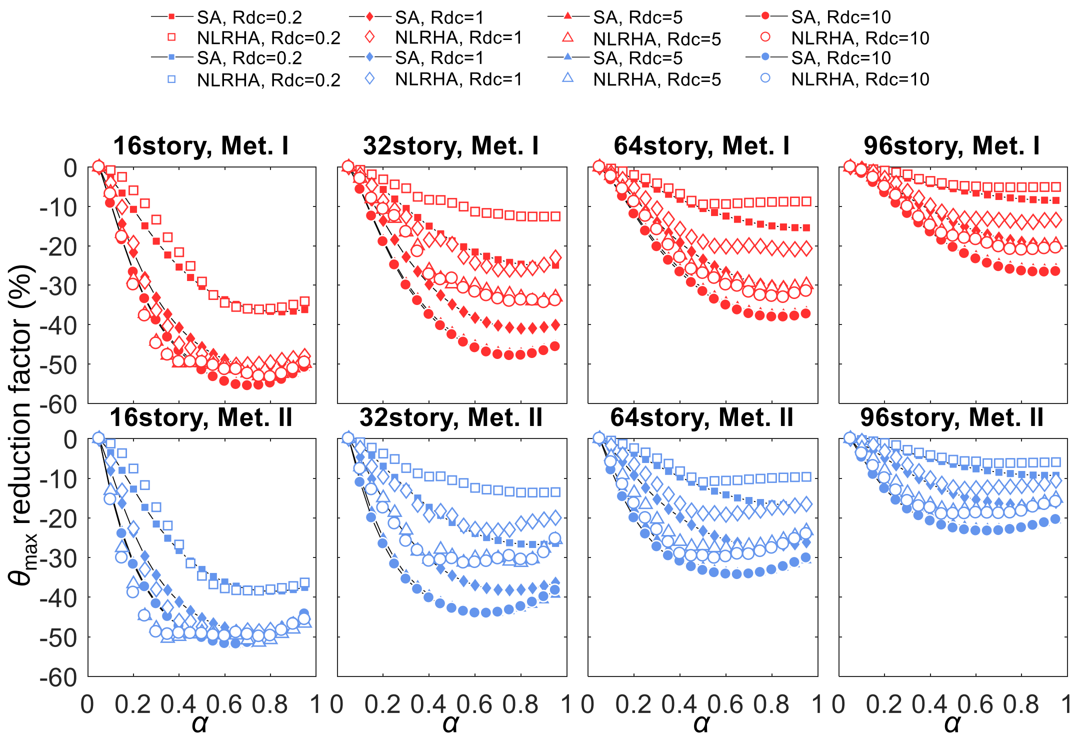


Figure 7.2.2 The $\theta_{\max}$ reduction factors calculated from SA and NLRHA

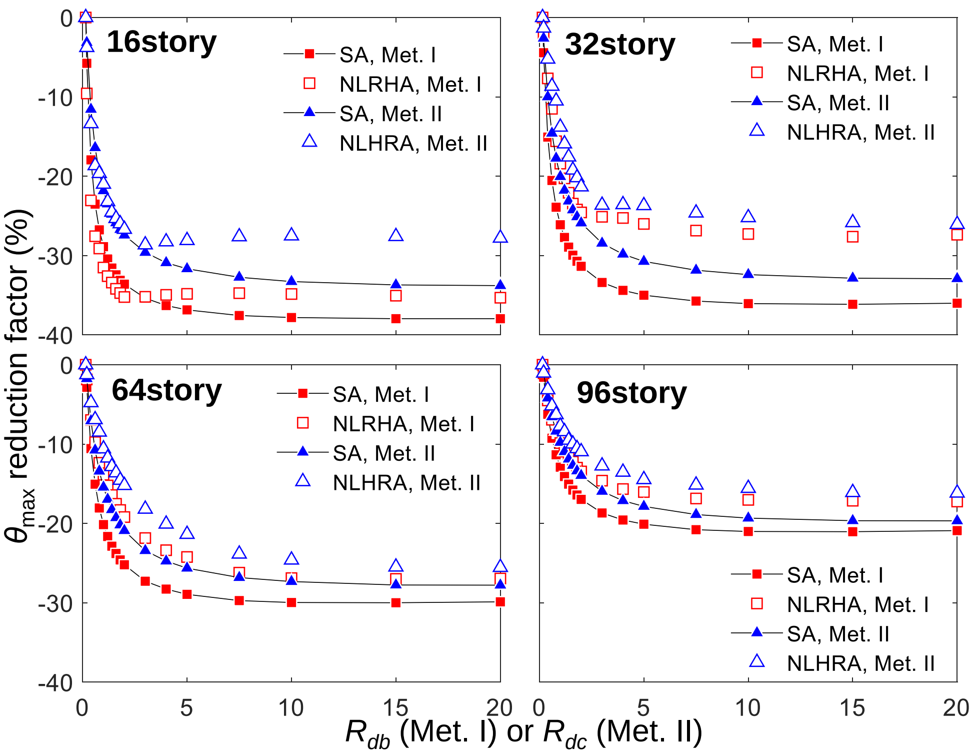


Figure 7.2.3 The relationship between $\theta_{\max}$ reduction factors and R_{db} or R_{dc} when α equals to 0.7 calculated from SA and NLRHA

7.2.2 Maximum inter-story drift

Figure 7.2.4 shows the relationships between the maximum inter-story drift (γ_{max}) and the outrigger elevation (α) calculated from SA and NLRHA using the BCJ-L2 ground motion. Figure 7.2.5 shows the relationships between γ_{max} reduction factor (if compared with the core structure without outrigger effect) and α calculated from SA and the results of the average of NLRHA with 8 ground motions. The SA results well agree with the NLRHA results. The γ_{max} obtained from NLRHA is slightly larger than the one obtained from SA in taller structure models. This could be due to that the effect from the higher mode responses is more significant in taller structures, and the SA procedure using the SRSS to calculate the deformed shape may be less capable to clarify this effect. The trends of γ_{max} are similar to the θ_{max} responses. The γ_{max} is minimum when α is approximately 0.7 to 0.8. The larger value of R_{db} or R_{dc} results in smaller γ_{max}. However, the responses when R_{db} or R_{dc} equals to 5 and 10 are very close to each other. This suggests that the reduction in γ_{max} is not proportional to the increase of R_{db} or R_{dc}. Figure 7.2.6 shows the relationships between γ_{max} reduction factor and R_{db} (Met. I) or R_{dc} (Met. II) when α equals to 0.7. Both the SA and NLRHA results show that the rate of change in γ_{max} becomes slow or even stops when R_{db} or R_{dc} is greater than 5, which is similar to the response of θ_{max} reduction factor shown in Figure 7.2.3.

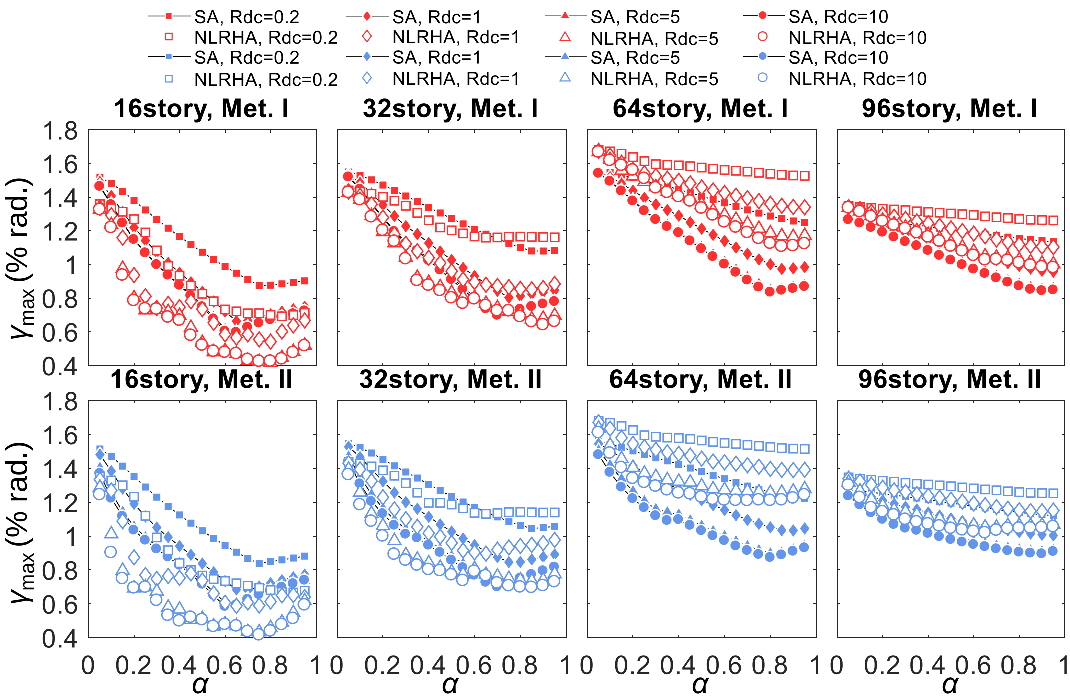


Figure 7.2.4 The γ_{max} distributions calculated from SA and NLRHA with BCJ-L2 ground motion

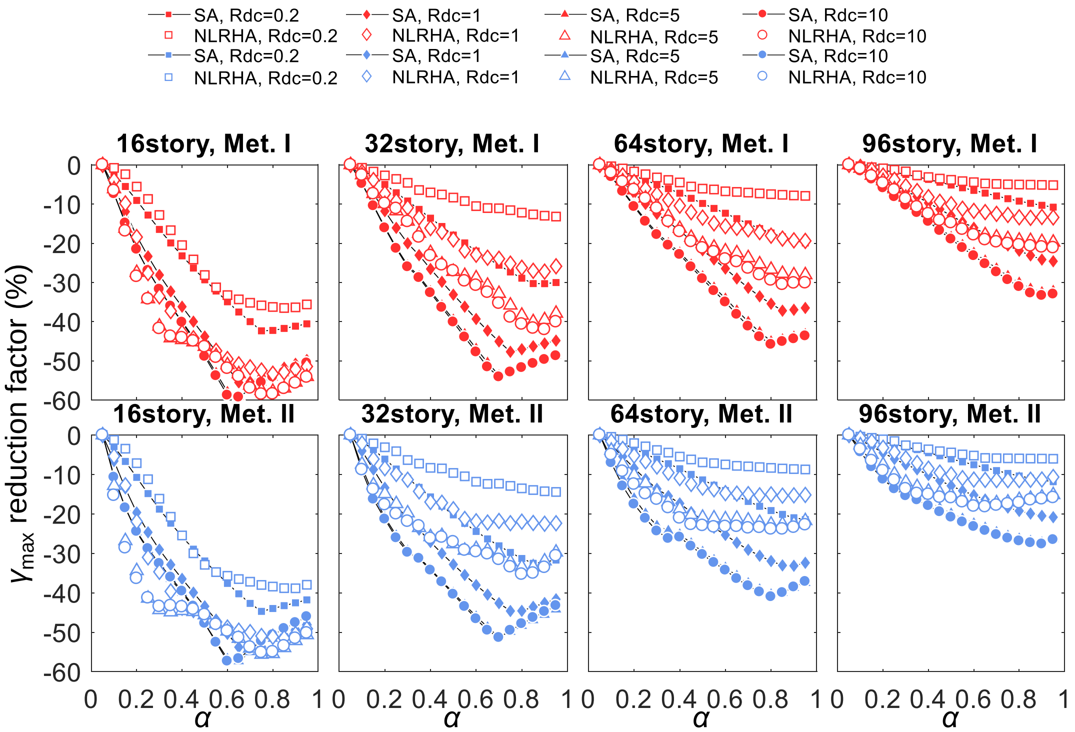


Figure 7.2.5 The $\gamma_{\max}$ reduction factors calculated from SA and NLRHA

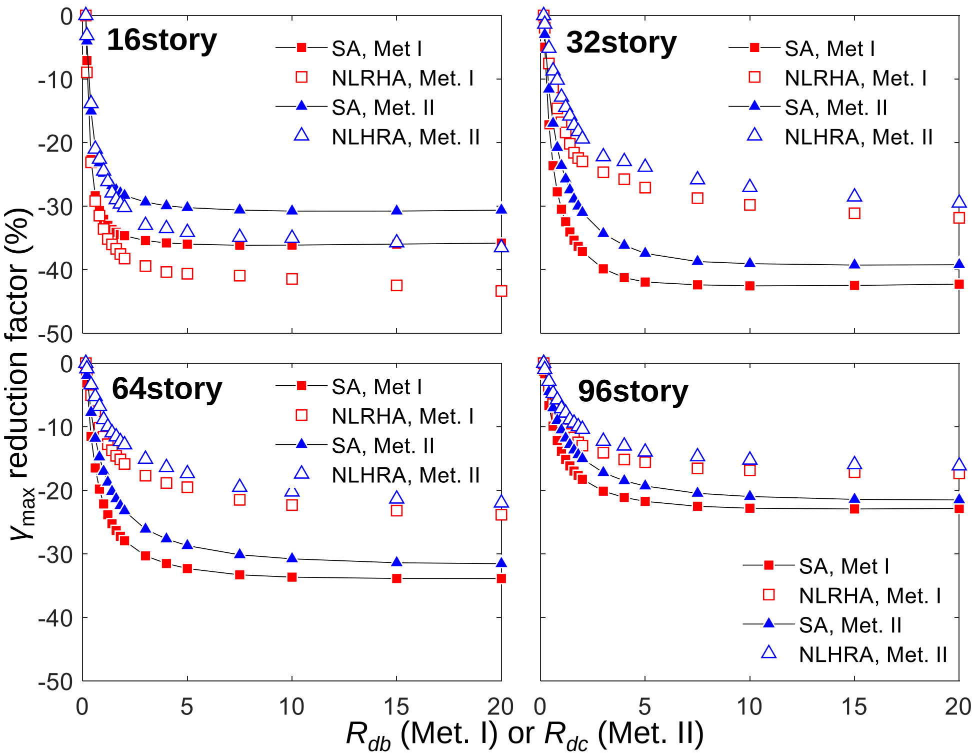


Figure 7.2.6 The relationship between $\gamma_{\max}$ reduction factors and R_{db} or R_{dc} when α equals to 0.7 calculated from SA and NLRHA

7.2.3 Maximum overturning moment

Figure 7.2.7 shows the relationships between the maximum overturning moment at core structure base ($M_{c,\max}$) and outrigger elevation (α) calculated from the average of NLRHA with 8 ground motions. Figure 7.2.8 shows the relationships between $M_{c,\max}$ reduction factor (if compared with the core structure without outrigger effect) and α calculated from the average of NLRHA results using 8 ground motions. When α is at approximately 0.7 to 0.8, the $M_{c,\max}$ can be best reduced. The larger R_{db} or R_{dc} value results in greater outrigger effect and therefore leads to smaller $M_{c,\max}$ response. However, the reduction in $M_{c,\max}$ is not proportional to the increasing R_{db} or R_{dc}. In addition, the low-rise structure models with larger S_{bc} value (more significant outrigger effect) have greater rotational stiffness provided by BRB-outrigger, and thus are more efficient in reducing the $M_{c,\max}$ responses. Figure 7.2.9 shows the relationships between $M_{c,\max}$ reduction factor and R_{db} or R_{dc} when α equals to 0.7. The rate of change in $M_{c,\max}$ reduction factor becomes slow or stops when R_{db} or R_{dc} is greater than 5. However, the $M_{c,\max}$ reduction factors of the 16-story model increase ($M_{c,\max}$ becomes larger) when R_{db} or R_{dc} is greater than 5. Figure 7.2.10 shows the 1st mode vibration periods of the analytical models, and Figure 7.2.11 shows the corresponding ranges of the 1st mode vibration periods on the design acceleration spectrum. When the value of R_{db} or R_{dc} increases, the vibration period becomes shorter and the seismic demand increases. The increases in seismic demand of the 16- and 32-story models are much larger than the 64- and 96-story models. Although the increases in R_{db} or R_{dc} value also increase the outrigger effect in mitigating seismic response, however, the greater outrigger effect cannot compensate the enlarged seismic response due to amplified seismic demand. The 32-story model also shows similar responses that the $M_{c,\max}$ reduction factor increases when R_{db} or R_{dc} is greater than approximately 10.

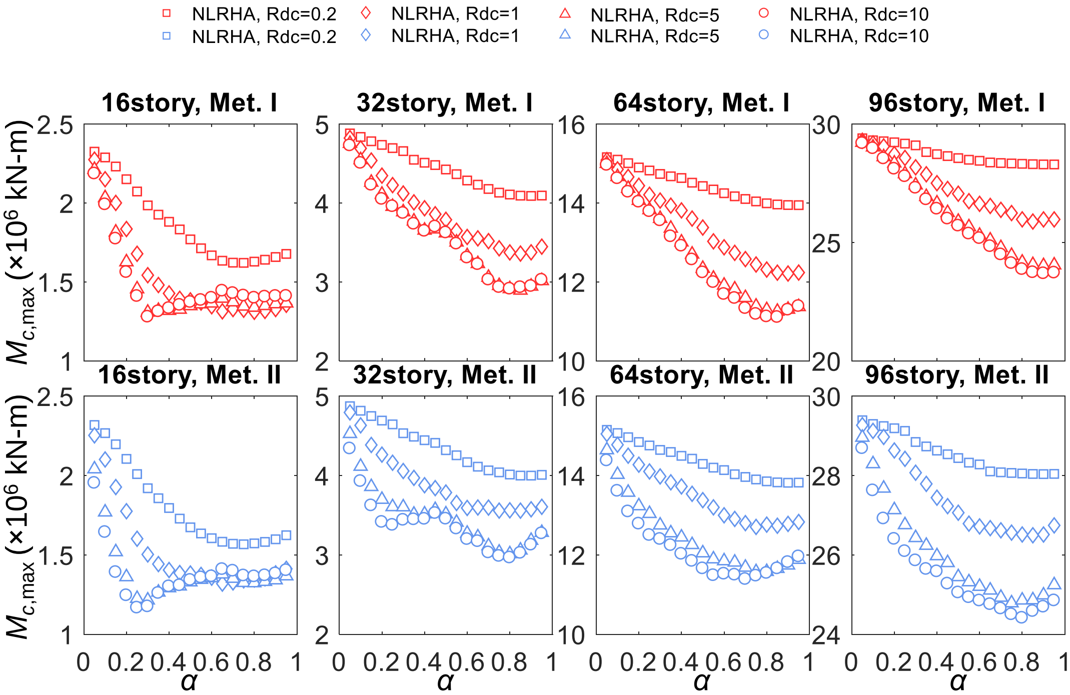


Figure 7.2.7 The $M_{c,\max}$ distributions calculated from NLRHA

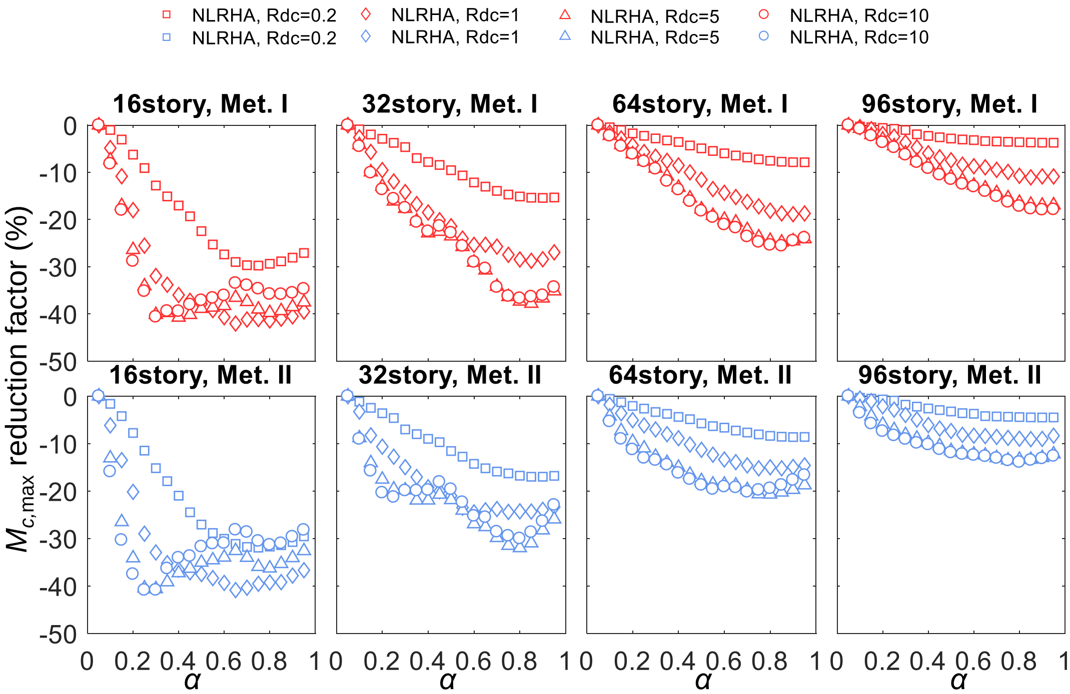


Figure 7.2.8 The $M_{c,\max}$ reduction factors calculated from NLRHA

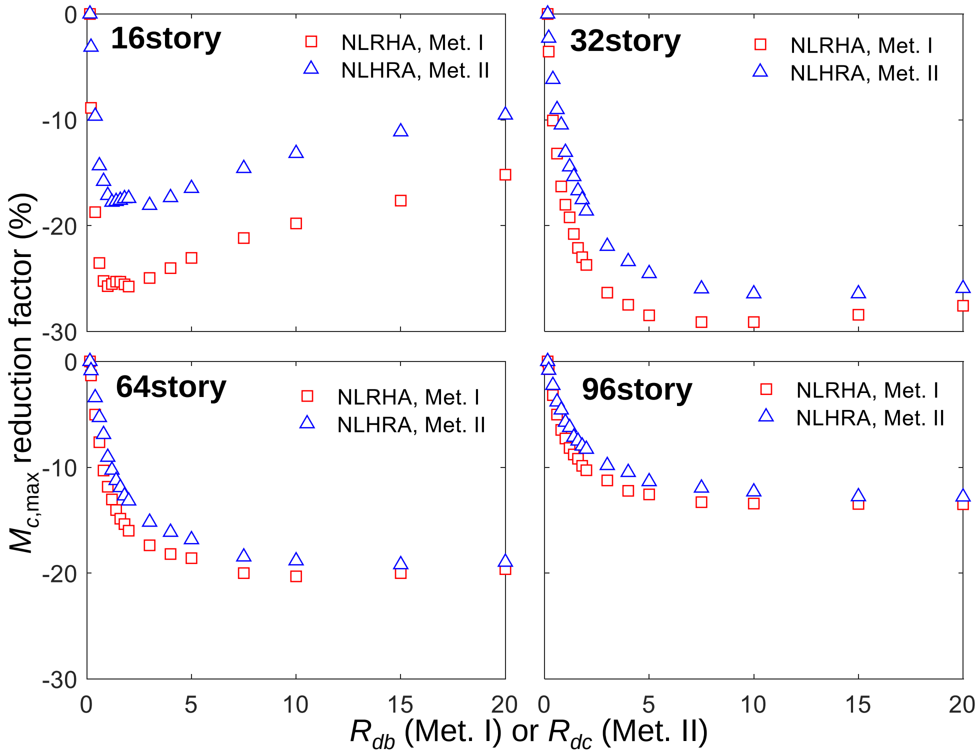


Figure 7.2.9 The relationship between $M_{c,\max}$ reduction factors and R_{db} or R_{dc} when α equals to 0.7 calculated from NLRHA

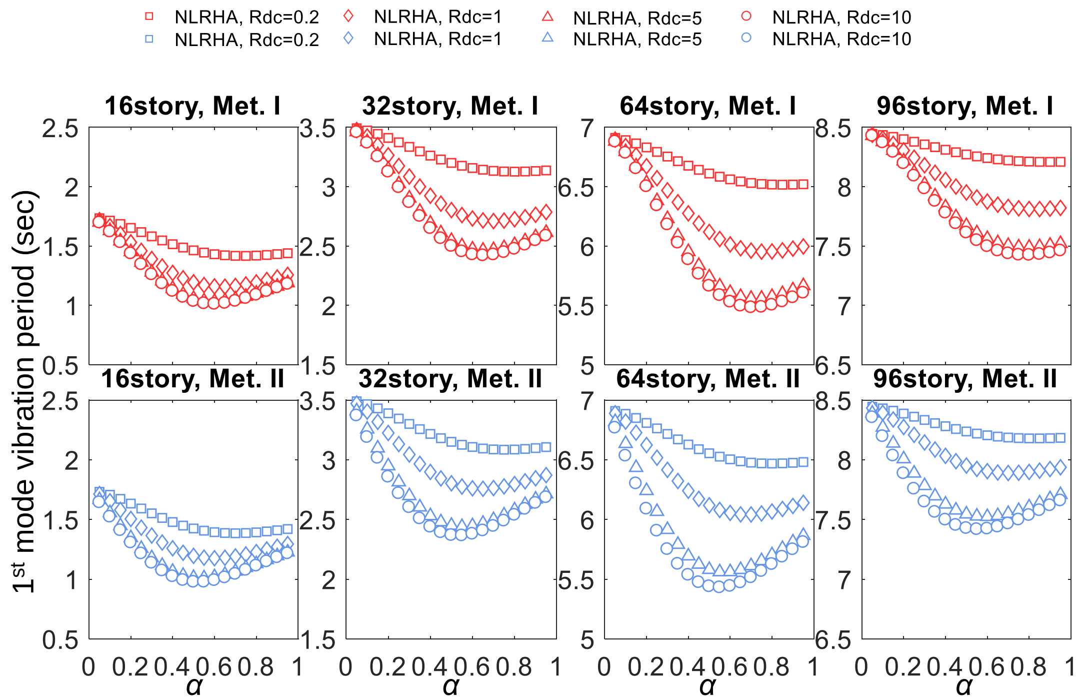


Figure 7.2.10 The relationship 1st mode vibration period and α

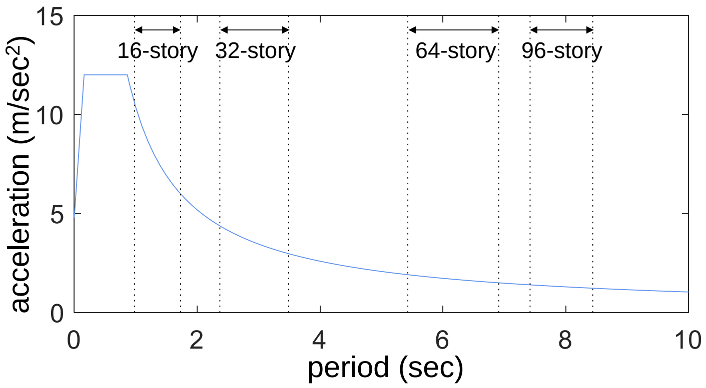


Figure 7.2.11 The design acceleration spectrum and the ranges of 1st mode period of the 16-, 32-, 64-, and 96-story models

7.2.4 Maximum perimeter column axial force

Figure 7.2.12 shows the relationships between the maximum perimeter column axial force at the 1st floor ($C_{1,\max}$) and outrigger elevation (α). The $C_{1,\max}$ is maximum when α is approximately 0.5 to 0.8, which is also within the optimal α in order to minimize $\theta_{\max}$ and $\gamma_{\max}$. This should be straightforward, since the BRB-outrigger system utilizes the perimeter column's axial stiffness to generate a resisting moment on the core structure when the seismic response is best reduced, the perimeter column's axial stiffness should be best utilized. As in the abovementioned sections, the larger value of R_{db} or R_{dc} could result in smaller $\theta_{\max}$, $\gamma_{\max}$, and $M_{c,\max}$ responses. However, the larger value of R_{db} or R_{dc} significantly amplifies the $C_{1,\max}$ as shown in Figure 7.2.12. Figure 7.2.13 shows the relationships between $C_{1,\max}$ and R_{db} or R_{dc} when α equals to 0.7. If compare with Figure 7.2.3, Figure 7.2.6, and Figure 7.2.9, the $C_{1,\max}$ increases with the increasing R_{db} or R_{dc} value. In addition, the rate of increase in $C_{1,\max}$ is almost proportional to the increase of R_{db} or R_{dc} when they range between 0 and 5, and becomes slower when R_{db} or R_{dc} is larger than 5. This suggests that when R_{db} or R_{dc} is greater than 5, the $\theta_{\max}$, $\gamma_{\max}$, and $M_{c,\max}$ cannot be further reduced, but the $C_{1,\max}$ is further amplified. For the design practice, too large $C_{1,\max}$ is not desirable, as it would enlarge the perimeter column size and construction cost. Therefore, the optimal R_{db} or R_{dc} value should be smaller than 5.

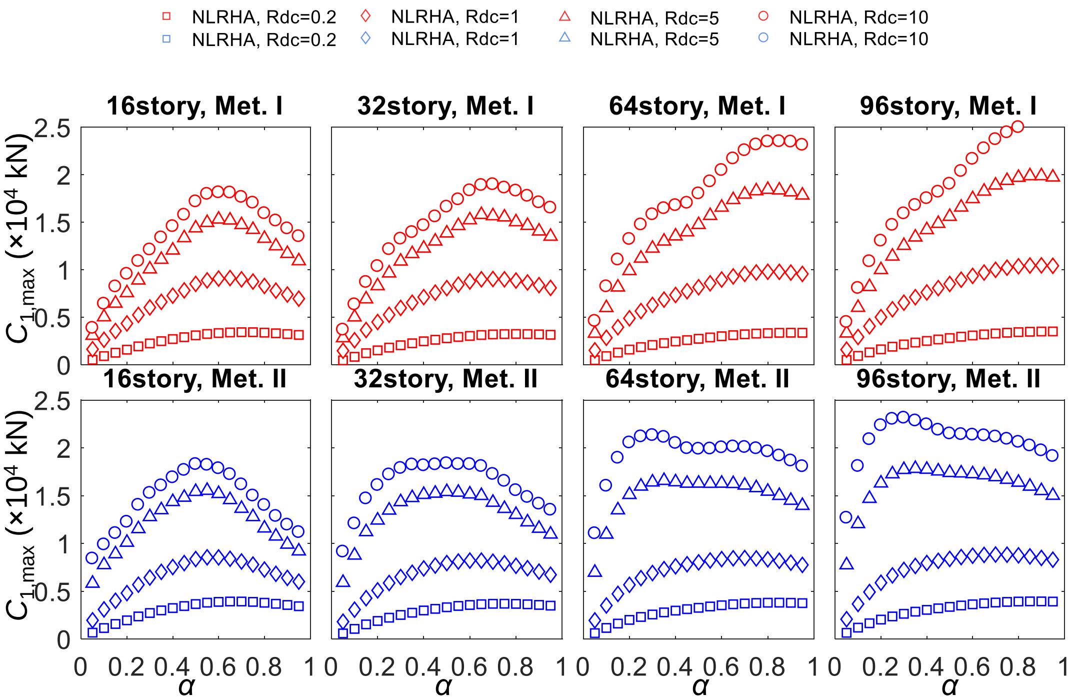


Figure 7.2.12 The $C_{1,max}$ distributions calculated from NLRHA

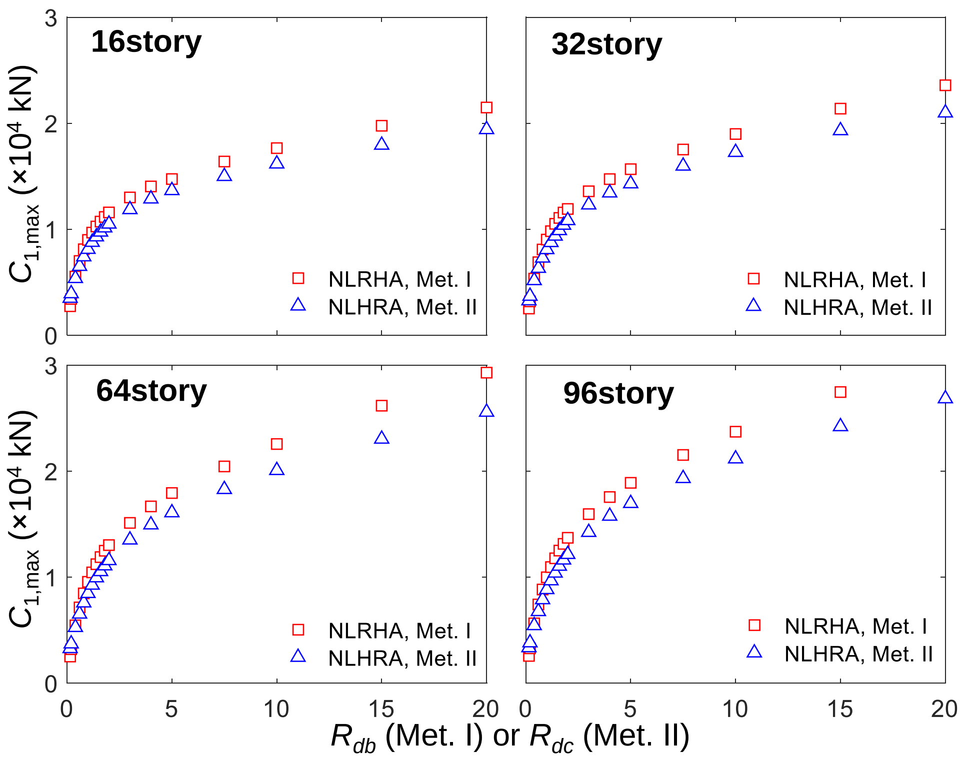


Figure 7.2.13 The relationship between $C_{1,max}$ and R_{db} or R_{dc} when α equals to 0.7 calculated from NLRHA

7.2.5 BRB energy dissipation efficiency

Figure 7.2.14 and Figure 7.2.15 show R_{CPD} (ratio of BRB's cumulative plastic deformation to axial yield deformation) (ANSI/AISC 341-16, 2016) and the relationship between ratios of energy dissipated by BRBs to the total input energy with respect to α calculated from the NLRHA results. The larger value of R_{CPD} suggests the BRB consumes more ductility demand. A very small R_{CPD} value indicates the BRB experiences a small amount of inelastic deformation. Too large R_{CPD} value indicates the BRB core could easily fracture as the BRB uses up its ductility demand. As shown in Figure 7.2.14, the R_{CPD} begins increasing significantly when R_{db} and R_{dc} are greater than 5.0. For the models with large R_{db} or R_{dc} values, once the BRB yields, the drop in BRB stiffness from k_d to relatively small post-yield stiffness (pk_d), if compared with k_c and k_t, can result in a large decrease in k_g and large deformation concentration in the BRB. In addition, as illustrated in Chapter 6, large R_{db} or R_{dc} would result in smaller $u_{d,y}$. Therefore, the energy dissipation efficiencies for the models with large R_{db} or R_{dc} values (greater than 5) accompanies with small $u_{d,y}$ and large BRB axial deformation are similar to those with R_{db} or R_{dc} value that lies between 3 and 5. This helps in explaining that when R_{db} or R_{dc} is larger, $\theta_{\max}$ cannot be proportionally reduced and the BRB energy dissipation efficiency remains almost the same. This also explains that the large R_{CPD} values are found from the models with large R_{db} or R_{dc} values. However, a very large R_{CPD} value indicates that the BRB may use up its ductility capacity and eventually fracture before the end of the earthquake, which is not desirable for engineering practices. For the design purpose, increasing R_{db} or R_{dc} also increases the cost of BRB. Based on the analysis results, in order to reduce $\theta_{\max}$ and meanwhile prevent from excessively gaining R_{CPD} value, it is suggested that the R_{db} and R_{dc} should be greater than 1 and smaller than 5.

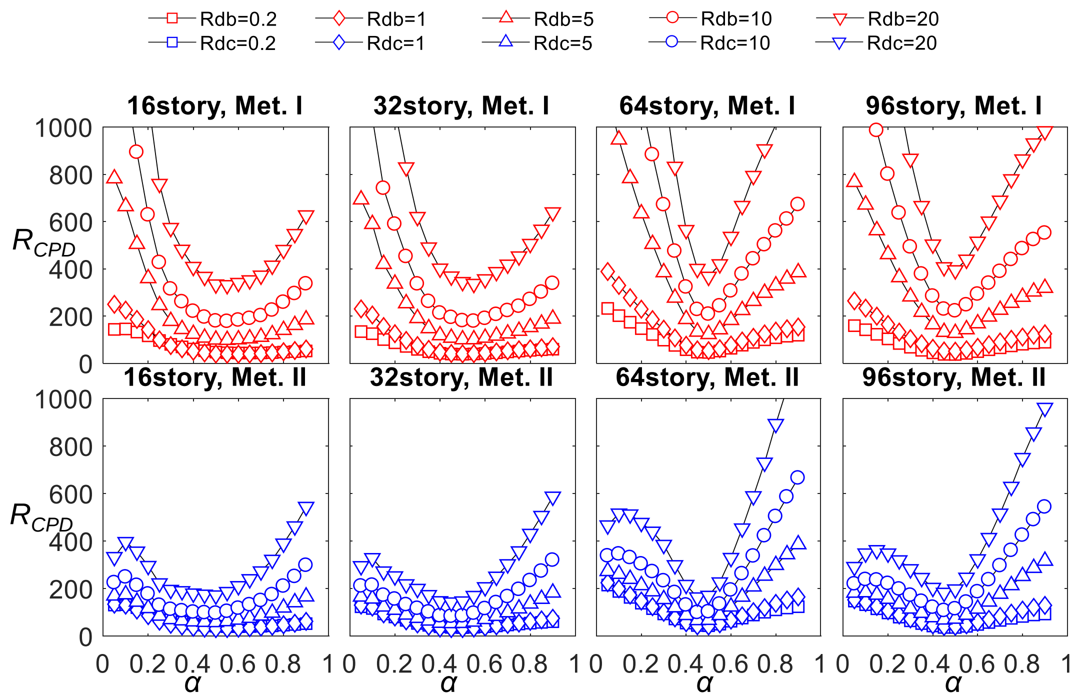


Figure 7.2.14 Relationships between R_{CPD} and α calculated from NLRHA

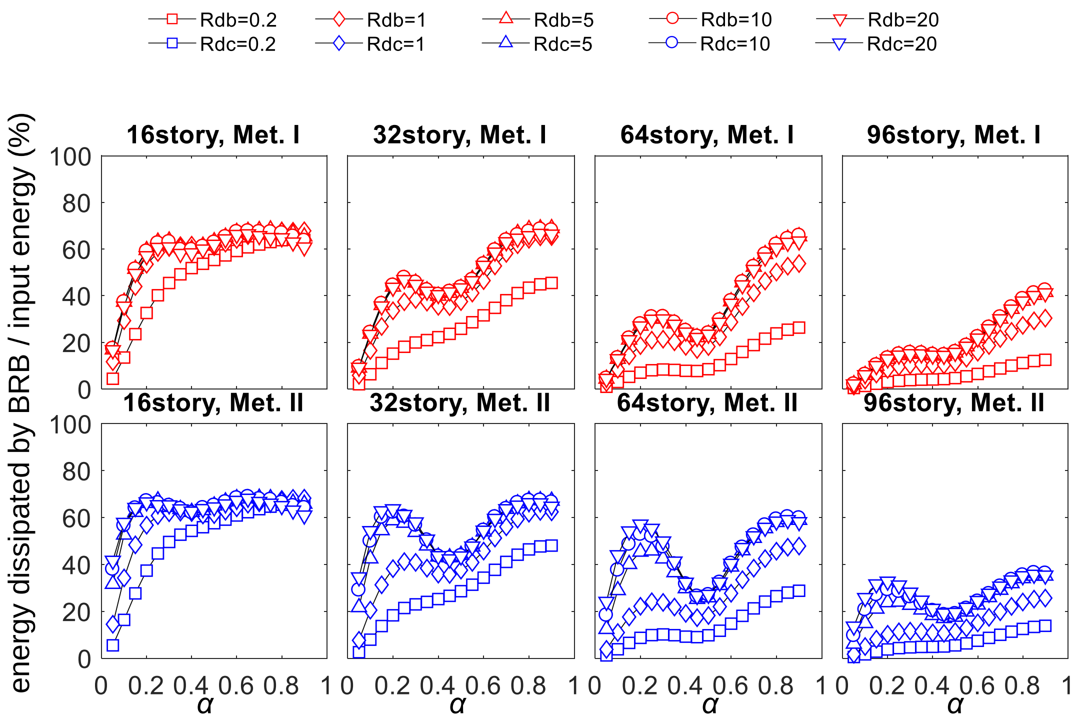


Figure 7.2.15 Relationships between the ratio of energy dissipated by BRB to input energy and α calculated from NLRHA

7.2.6 Summary of optimal design for single BRB-outrigger system

The ratio of $\theta_{\max}$ of a single BRB-outrigger system to the $\theta_{\max}$ of core structure without BRB-outrigger effect is defined as the roof drift reduction ratio (R_d). The ratio of the maximum roof lateral acceleration ($a_{\max}$) of a single BRB-outrigger system to the $a_{\max}$ of core structure without BRB-outrigger effect is defined as the roof acceleration reduction ratio (R_a). The performance curves can be drawn by plotting the relationships between R_a and R_d. Figure 7.2.16 and Figure 7.2.17 show the performance curves calculated from the SA and NLRHA, respectively. The performance curves calculated from NLRHA are based on the average of analysis results using eight ground motions. In each plot, each of the performance curves is drawn by fixing α at either 0.1, 0.3, 0.5, 0.7, or 0.9. The numbers on the performance curves indicate the corresponding R_{db} (Met. I) or R_{dc} (Met. II) value. Both SA and NLRHA analysis results indicate that the R_d and R_a reach minima when α is between 0.7 and 0.9. The Met. I results show that the optimal α, in order to achieve minimum R_a and R_d, increases from approximate 0.7 to 0.9 with increasing building height (decreasing in S_{bc}). In addition, for any fixed α, the $\theta_{\max}$ can be reduced by increasing R_{db} or R_{dc}. However, if R_{db} or R_{dc} is too large resulting in a very stiff system, the R_a could be significantly amplified, and the R_d could increase again. As shown from the performance curves calculated from SA (Figure 7.2.16), the Met. I suggests that the optimal α required to achieve both minimum R_d and R_a is approximately between 0.7 and 0.9 for the 16- and 32-story models, and 0.9 for the 64- and 96-story models, and the Met. II suggests that the optimal α required to achieve both minimum R_d and R_a is approximately 0.7. As shown in Figure 7.2.16, the R_{db} or R_{dc} for achieving minimum R_a could result in an R_d value that is close to its minimum value. Therefore, the optimal R_{db} and R_{dc} could be approximated from the minimum R_a value. If compare the analysis results between the models with different story numbers, increasing S_{bc} value should be more efficient than increasing R_{db} or R_{dc} in order to enhance the outrigger effect, since the large R_{db} and R_{dc} could significantly amplify $a_{\max}$. Based on the analytical results, the design with α is between 0.7 and 0.9, R_{db} (Met. I) and R_{dc} (Met. II) are around 1 and 5, could achieve satisfactory seismic performance in reducing both $\theta_{\max}$ and $a_{\max}$ responses.

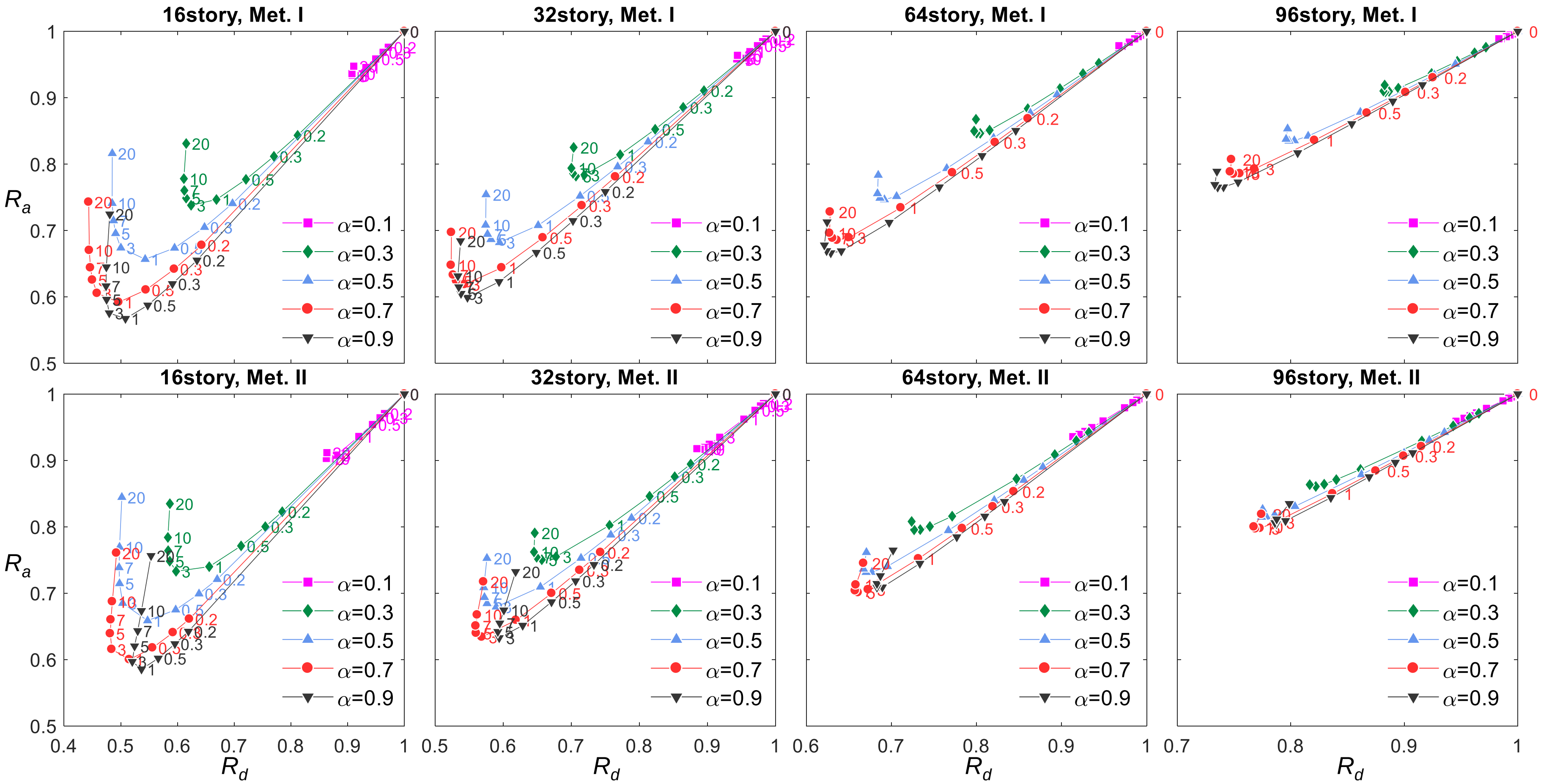


Figure 7.2.16 Performance curves calculated from SA

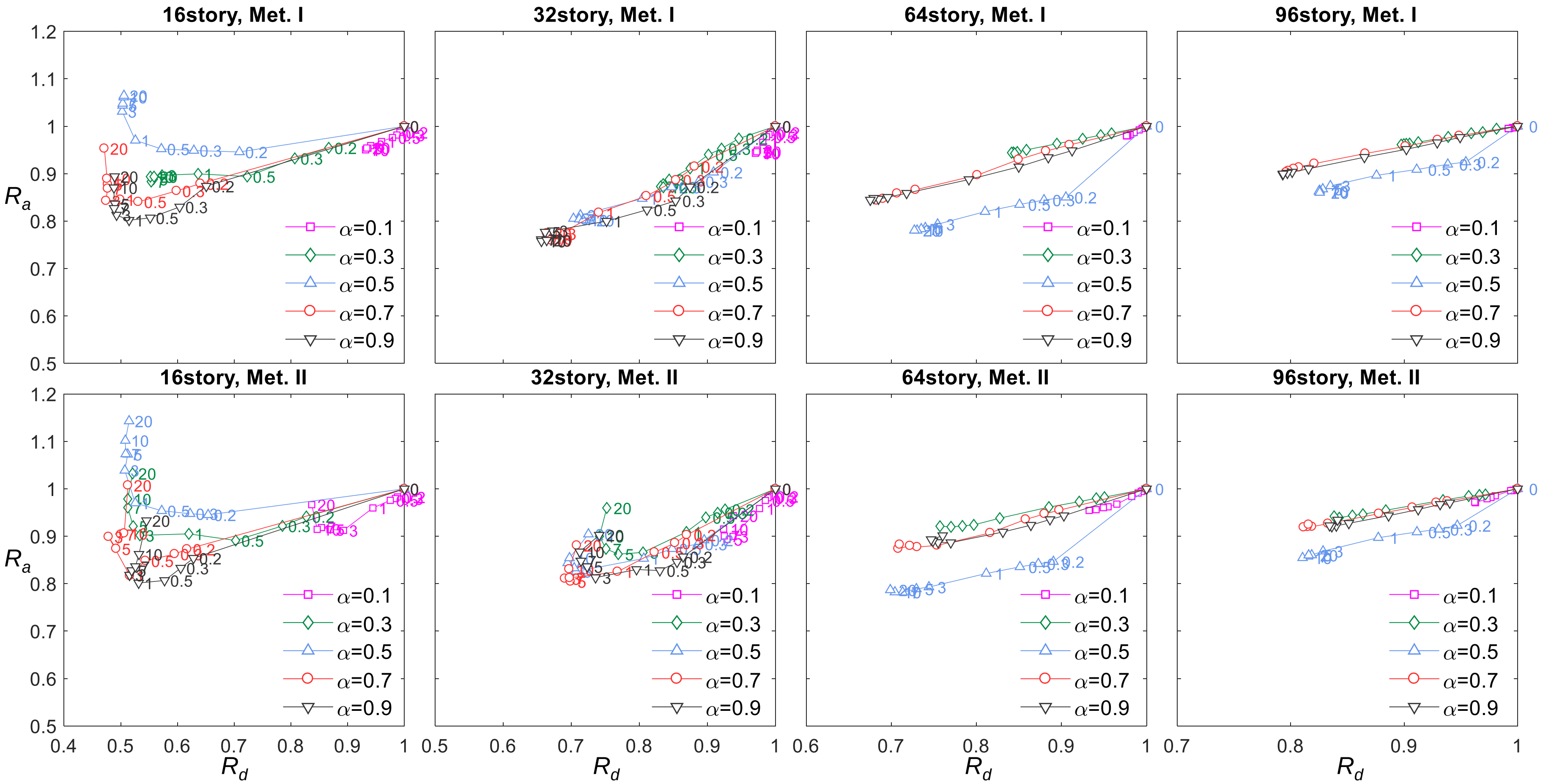


Figure 7.2.17 Performance curves calculated from NLRHA

7.3 DUAL BRB-OUTRIGGER SYSTEM

For the analysis on dual BRB-outrigger system, the BRB stiffness parameters of R_{d2c} equals 0.1, 1, and 3 and R_{kd} equals to 0.5, 1, and 3 are used for the 16-, 32-, 64-, and 96-story models, while the outrigger elevations α_1 and α_2 vary from 0 to 1. The SA and NLRHA results and the optimal designs are shown in the following sections. The NLRHA results are calculated from the average of the analysis results with eight different ground motions as described in Chapter 4.

7.3.1 Maximum roof drift

Figure 7.3.1, Figure 7.3.2, and Figure 7.3.3 show the θ_{max} when α_1 and α_2 vary from 0 to 1 and when R_{kd} equals to 0.5, 1, and 3, respectively, calculated from the SA and NLRHA with BCJ-L2 ground motion. When α_1 equals to 0, it is a single BRB-outrigger system. The SA well estimates the θ_{max} responses if compared with the NLRHA results, and the trends of θ_{max} with respect to the outrigger elevations (α_1 and α_2) calculated from SA and NLRHA are similar. The differences between SA and NLRHA could be due to that the SA uses SRSS to superpose the elastic mode shapes may not perfectly resemble to the NLRHA results. Based on the analysis results, the values of θ_{max} primary change with α_2, and the variations in α_1 only slightly affect the θ_{max} responses. It appears that the upper BRB-outrigger dominates the overall θ_{max} response, and the presence of the lower BRB-outrigger assist in further reducing θ_{max}. The value of θ_{max} reaches the minimum when α_2 and α_1 are approximately 0.7 and 0.6, respectively, and decreases with the increasing R_{d2c}. This suggests that a greater value of R_{d2c} (stiffer BRB) provides a greater outrigger effect in mitigating the seismic response. However, the decrease in θ_{max} when R_{d2c} increases from 0.1 to 1 is much greater than that when R_{d2c} increases from 1 to 3. This suggests that the reduction in θ_{max} is not proportional to the increasing R_{d2c}. The analysis results indicate that the benefit of reducing θ_{max} by increasing R_{d2c} becomes negligible if R_{d2c} is too large. If compare between the cases when R_{kd} equals to 0.5, 1, and 3, the θ_{max} becomes only slightly smaller when R_{kd} increases (stiffer BRB_1). As the upper BRB-outrigger dominates the overall response, the changes in α_2 and R_{d2c} would affect the overall response more than the changes in α_1 and R_{kd}. In addition, the structures with larger S_{bc2} values show higher efficiency in reducing θ_{max}.

Figure 7.3.4 to Figure 7.3.7 show the reduction factors in θ_{max}, which are the reductions (in percentage) if compare with the core structure model without outrigger effects, for the 16-, 32-, 64-, and 96-story models, respectively. The "×" symbol indicates the outrigger elevations with the smallest reduction factor. The reduction factors of the NLRHA results are calculated based on the average of NLRHA with 8 ground motions. Based on the analysis results, the distribution and value of reduction factors obtained from SA and NLRHA are similar. The reduction factors primarily change with α_2, and the θ_{max} reaches the minimum (smallest reduction factor) when α_2 is between 0.7 and 0.8. The effect of varying α_1 is negligible when α_2 is smaller than 0.4. Even when α_2 is at its approximate optimal elevation (between 0.7 and 0.8), the changes in the reduction factor because of varying α_1 is limited to within 10%. This may suggest that, when α_2 is smaller than 0.4, the presence of the lower outrigger has no contribution in achieving better seismic performance in reducing θ_{max}. The analysis results indicate that, when α_2 is at its optimal elevation and α_1 is approximately 0.4 to 0.7, the θ_{max} can be best reduced. In the models with the same number of stories, the increases in R_{d2c} from 0.1 to 1 and from 1 to 3 increase the θ_{max} reduction factors by approximately 10% and 5%, respectively. In addition, for the analysis results in models with different numbers of stories, the reduction factors are larger (smaller reduction in θ_{max}) in taller models that have smaller S_{bc2} values. Based on the analysis results, a greater value of S_{bc2} suggests a greater outrigger effect that, therefore, results in smaller seismic response. In summary, in order to best reduce the θ_{max}, the optimal upper outrigger elevations (α_2) are approximately 0.7 and 0.8. For the lower outrigger elevation (α_1), the optimal α_1 is in the range of 0.4 to 0.7. In addition, in order to achieve a smaller θ_{max} response, increasing the value of S_{bc2} when α_2=0.7 ($S_{bc2,07}$) would be more efficient than increasing the value of R_{d2c}.

Figure 7.3.8 shows the θ_{max} reduction factor (in percentage) distributions with respect to R_{d2c}, S_{bc2}, and R_{kd} for the cases when α_2 is 0.5, 0.7, and 0.9 and when α_1 is 0.3 and 0.6. In addition to the 16-, 32-, 64-, and 96-story models, additional analysis results obtained from two 16-story (16-storyB, 16-storyC) models with outrigger spans (l_t) of 14.5 m and 12.8 m, and a 32-story (32-storyD) model with an l_t of 13.8 m, are included in order to create a denser data distribution of the S_{bc2} value. Table 7.3.1 shows the detials of the additional three analytical models, and the θ_{max} reduction factor plot for the case of α_2 = 0.5, α_1 = 0.3, and R_{kd} =1 shown in Figure 7.3.9 indicates the data distribution. It should be noted that the θ_{max} reduction factor distributions shown in Figure 7.3.8 are based on the SA results. As can be seen in Figure 7.3.8, the shapes of the reduction factor are similar to each other. The greater values of R_{d2c} and S_{bc2} suggest a greater outrigger effect indicating smaller θ_{max} responses (smaller reduction factors). However, the rate of increase in the amount of reduction factors decreases, or even stops, as R_{d2c} increases under a fixed value of S_{bc2}. The optimal value of R_{d2c} should be approximately 0.5 to 1.5. When the value of R_{d2c} is greater than 1.5, the required BRB axial stiffness increases (also increases the cost of BRB), however, the reduction in θ_{max} becomes less efficient. In addition, if the cases when α_2 varies between 0.5, 0.7, and 0.9 are compared, when α_2 is changed from 0.5 to 0.7, the amounts of θ_{max} reduction factor increase by approximately 10%, and when α_2 is changed from 0.7 to 0.9, the amounts of θ_{max} reduction factor increase by only approximately 3%. These results suggest that the optimal upper BRB-outrigger elevation in order to mitigate θ_{max} shoule be between 0.7 to 0.9. In addition, for both the cases when R_{kd} changes from 0.5 to 1 and when R_{kd} changes from 1 to 3, the amounts of θ_{max} reduction factor increase by approximately 3% and 5% when α_1 is 0.3 and 0.6, respectively. Therefore, it appears that the method of increasing R_{kd} is efficient to reduce θ_{max} when α_1 is approximately 0.6.

Table 7.3.1 Detail of the additional analytical models

model	h (m)	l_t (m)	EI (kN-m^2)	$S_{bc2,07}$
16-storyB	64	14.5	4.1×10^9	2.48
16-storyC	64	12.8	4.1×10^9	1.93
32-storyD	128	13.8	1.6×10^{10}	1.02

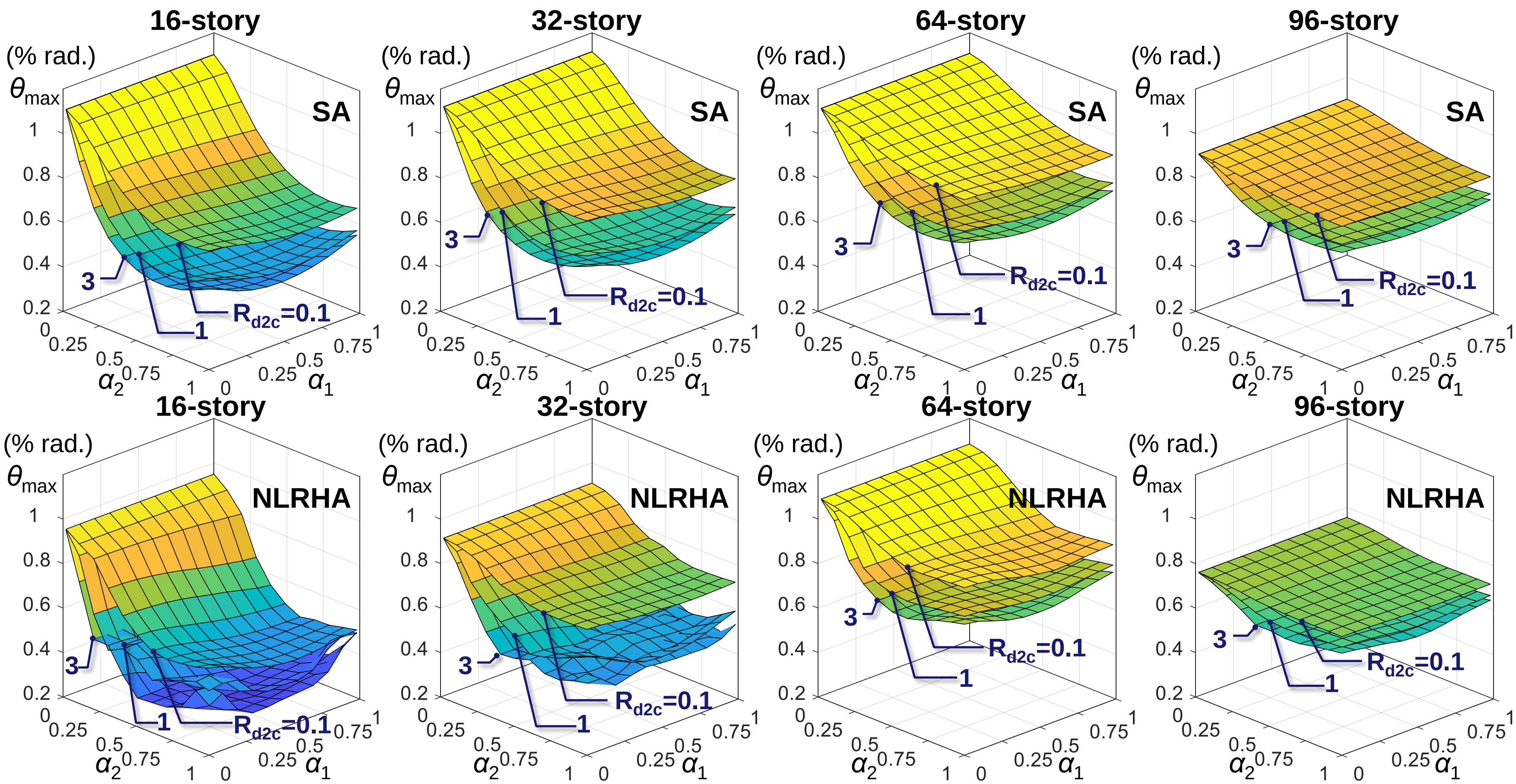


Figure 7.3.1 The θ_{max} distribution when $R_{kd} = 0.5$ calculated from SA and NLRHA with BCJ-L2 ground motion

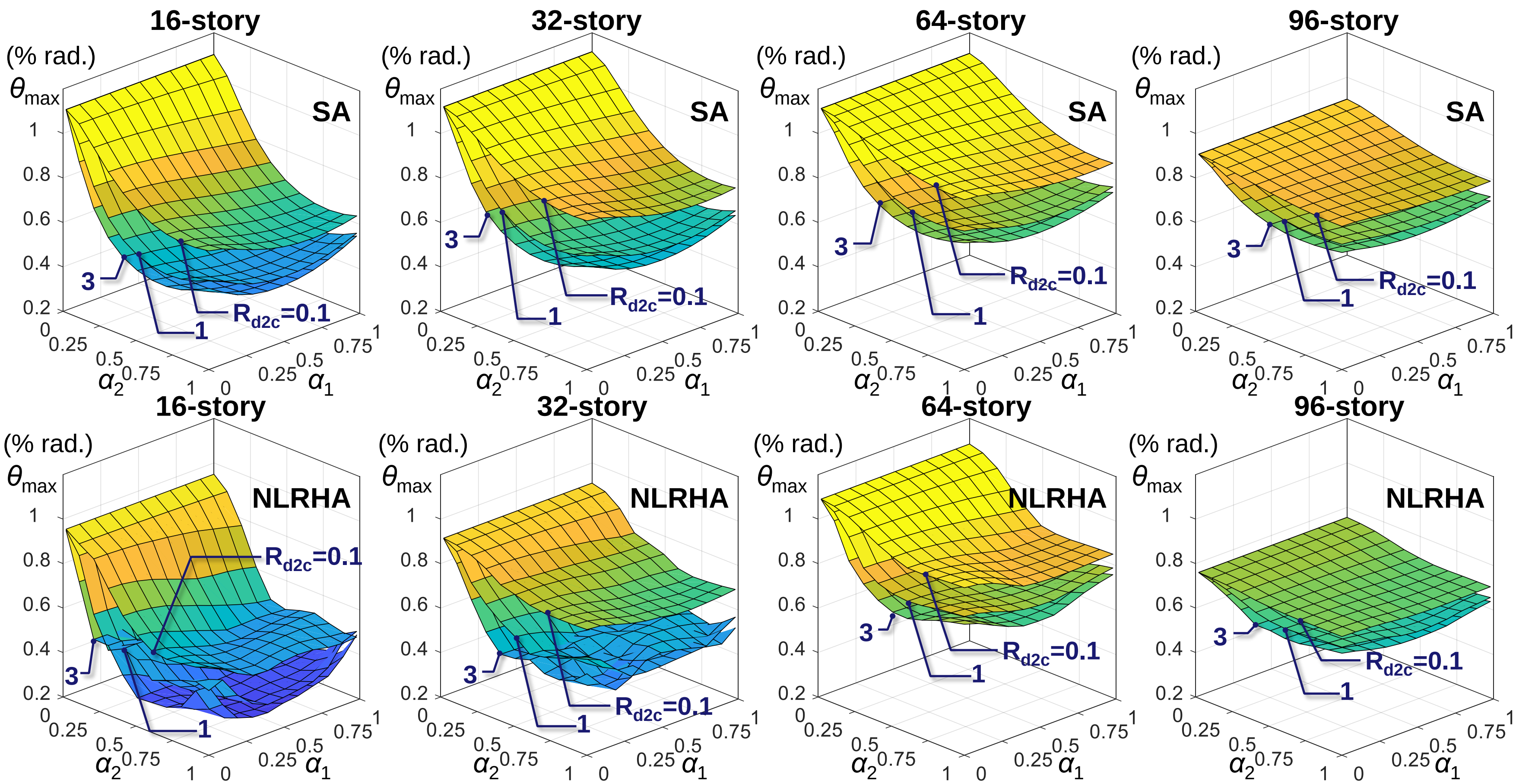


Figure 7.3.2 The θ_{max} distribution when $R_{kd} = 1$ calculated from SA and NLRHA with BCJ-L2 ground motion

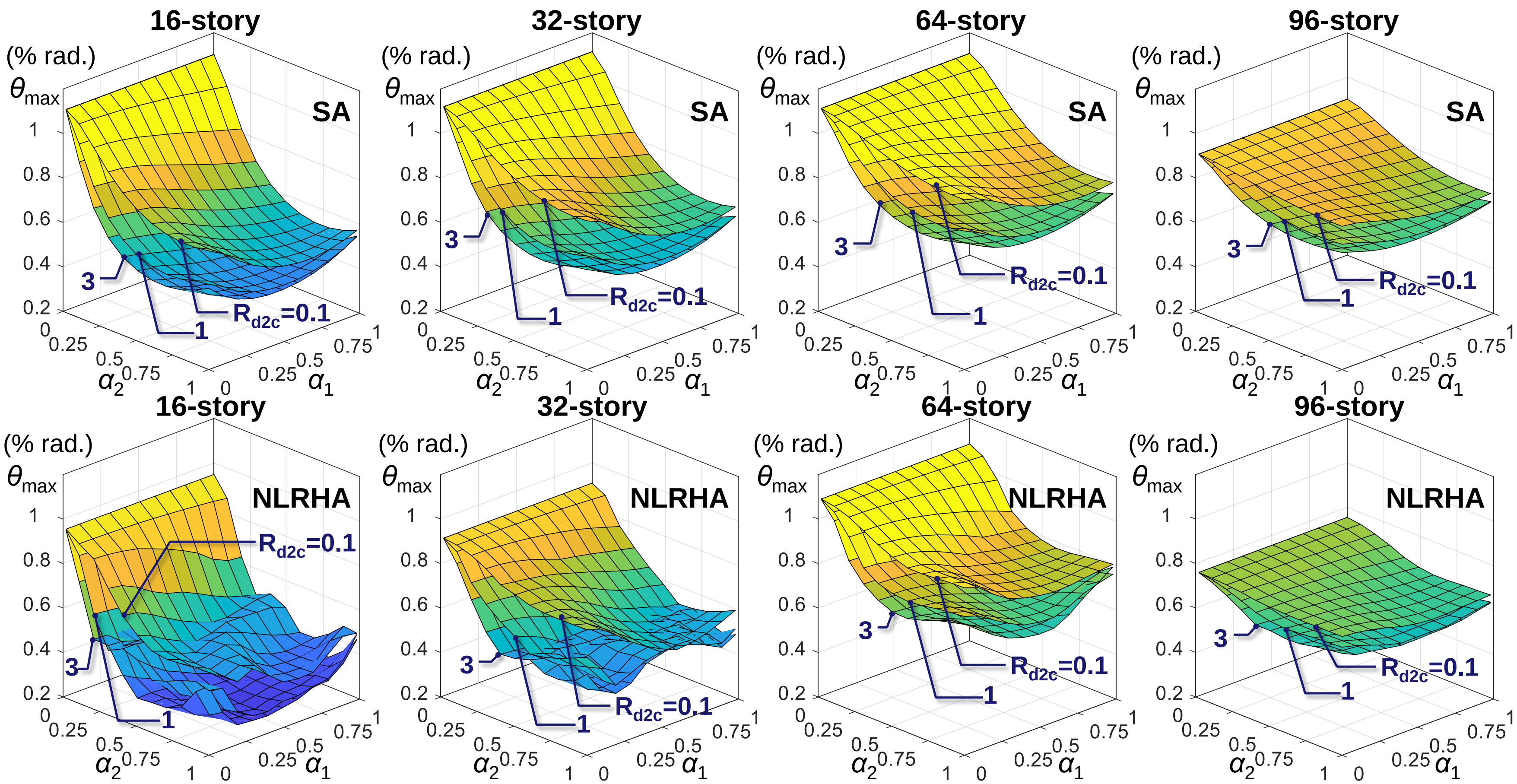


Figure 7.3.3 The θ_{max} distribution when $R_{kd} = 3$ calculated from SA and NLRHA with BCJ-L2 ground motion

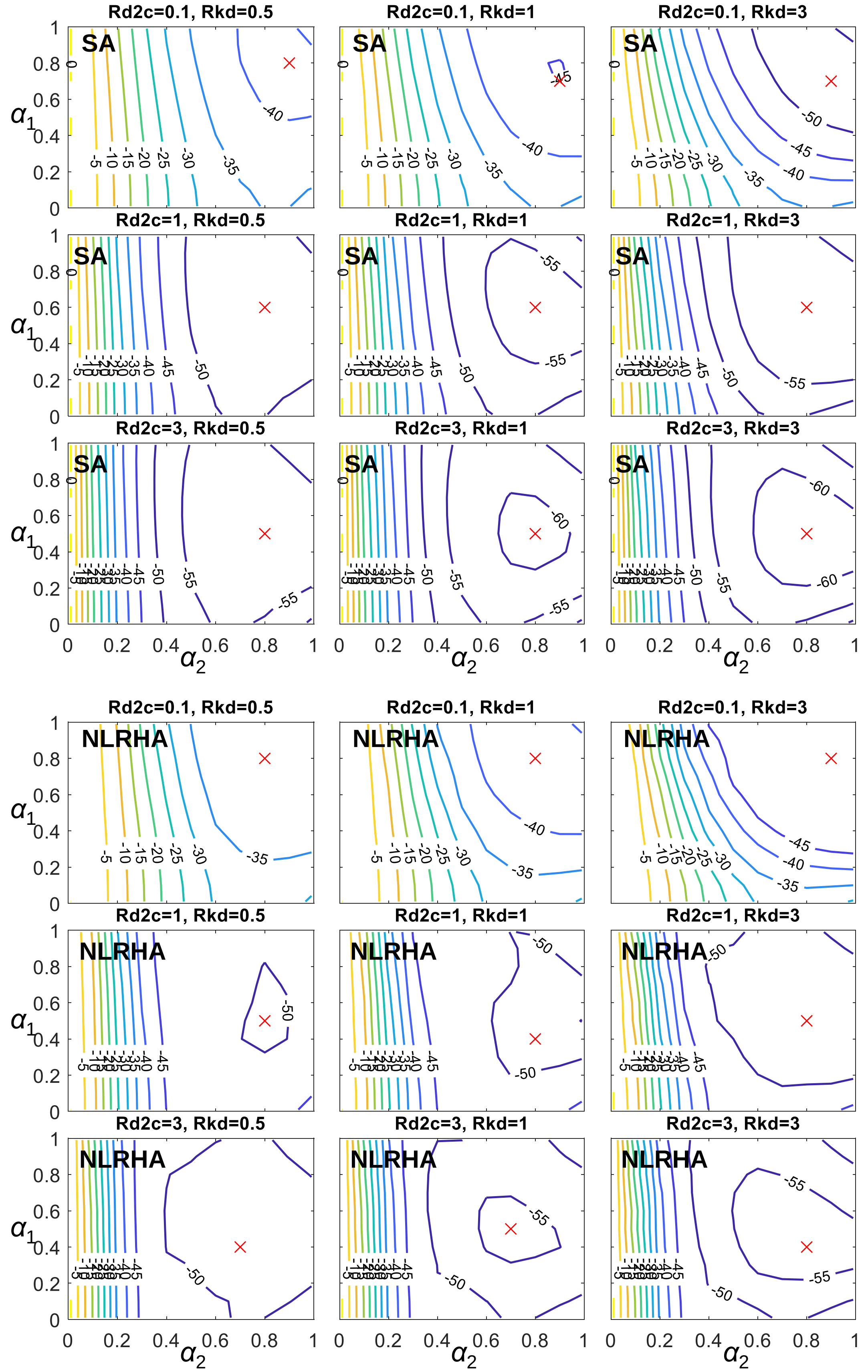


Figure 7.3.4 The θ_{max} reduction factor distributions with various outrigger elevations of the 16-story model

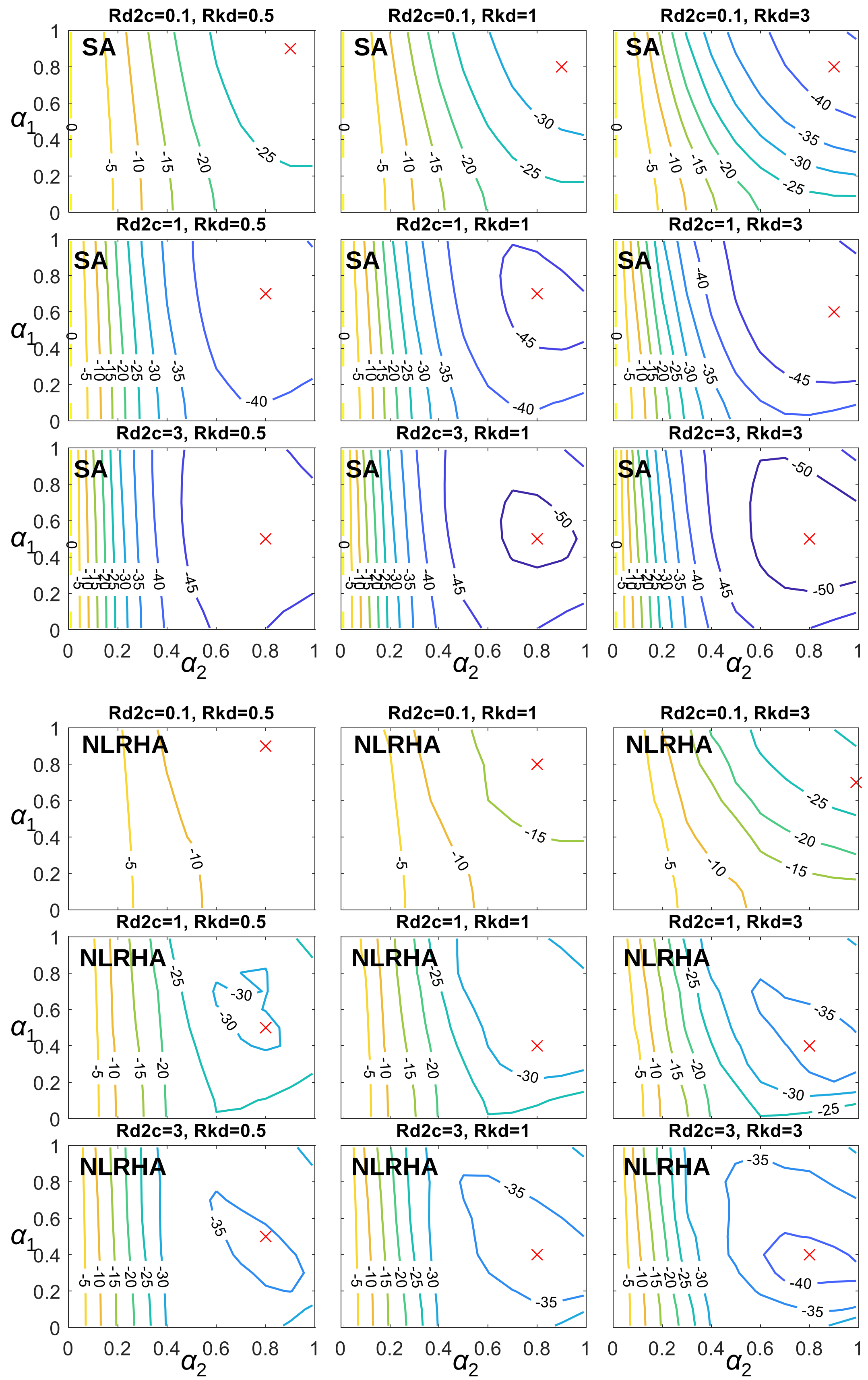


Figure 7.3.5 The θ_{max} reduction factor distributions with various outrigger elevations of the 32-story model

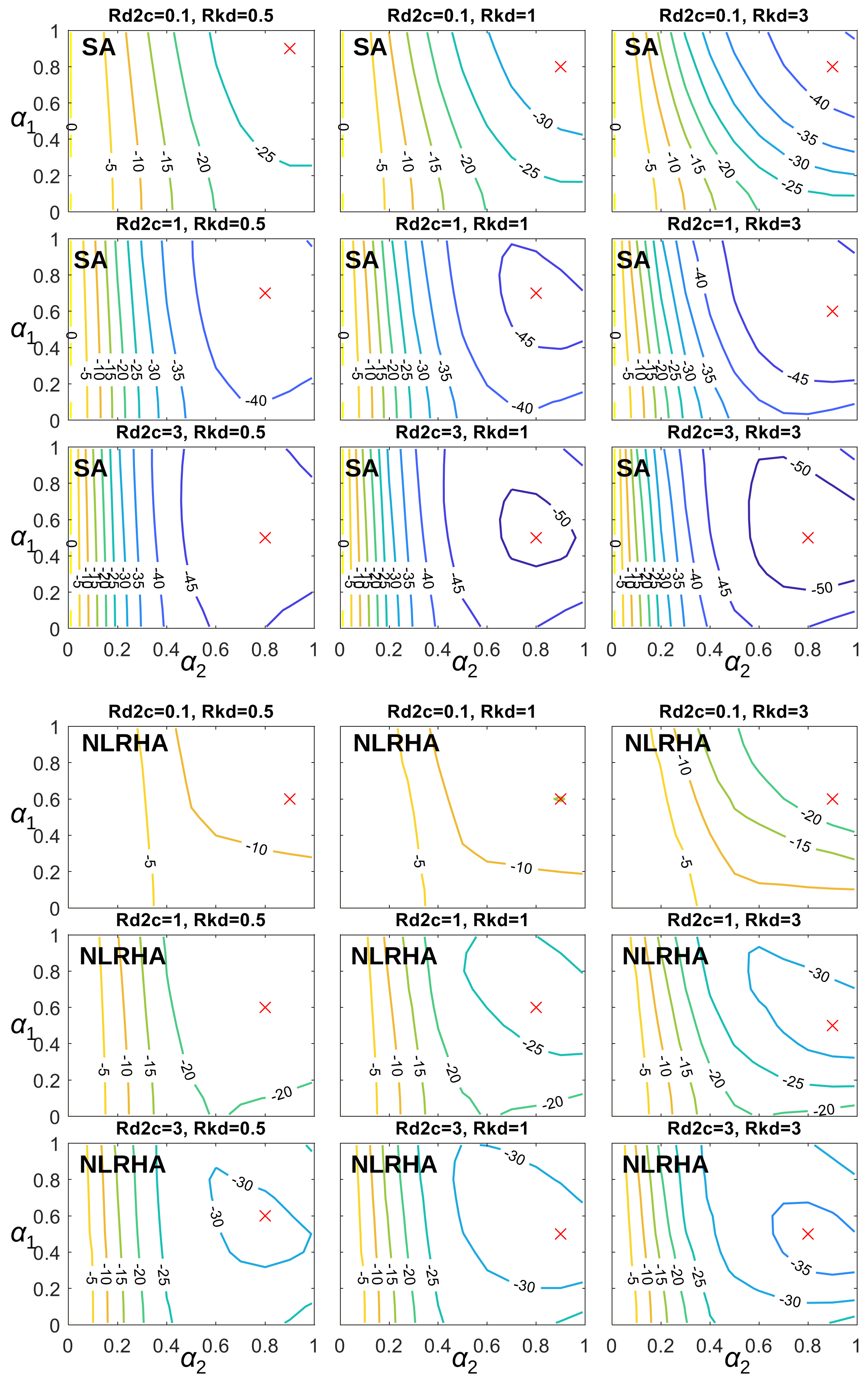


Figure 7.3.6 The θ_{max} reduction factor distributions with various outrigger elevations of the 64-story model

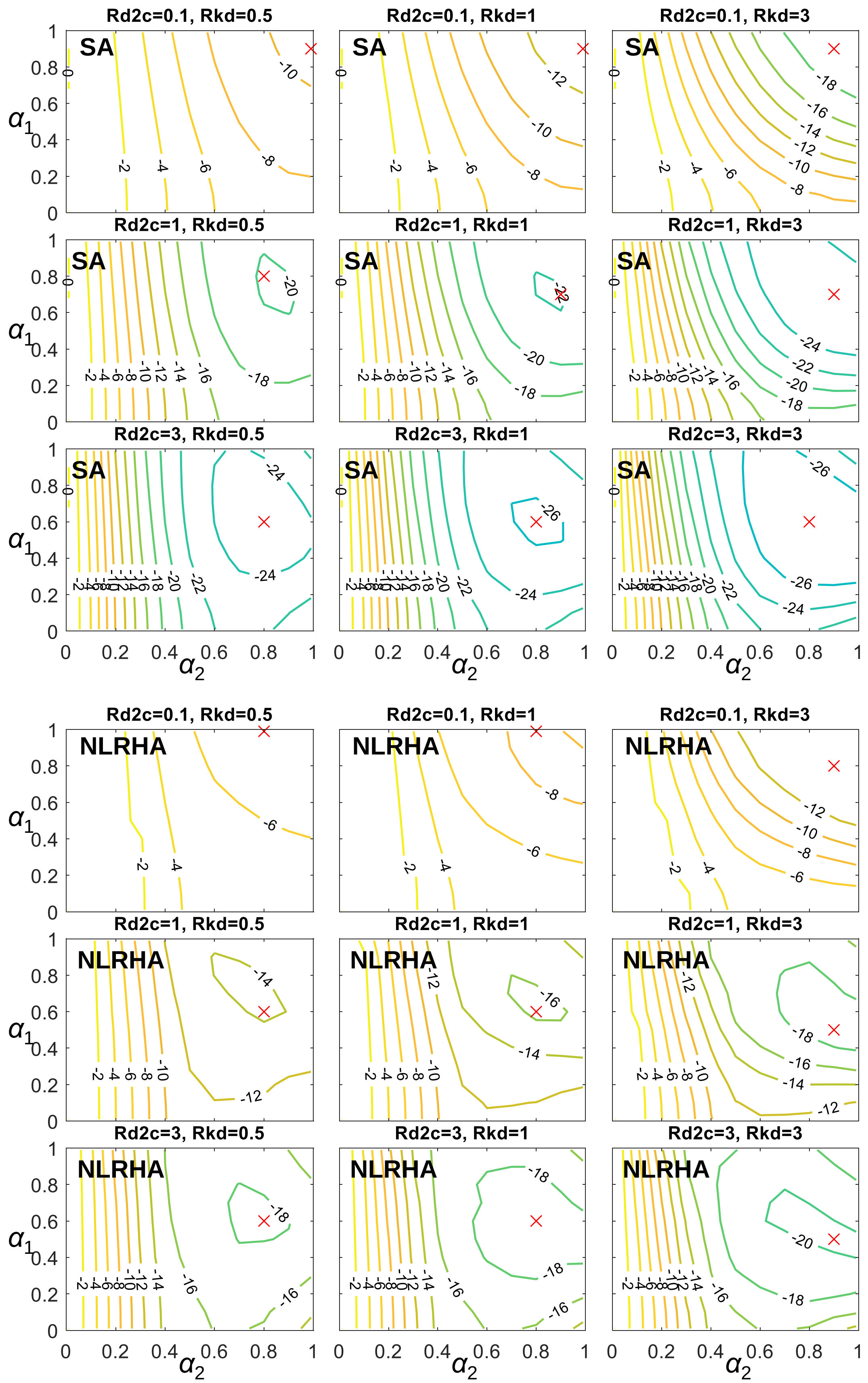


Figure 7.3.7 The $\theta_{\max}$ reduction factor distributions with various outrigger elevations of the 96-story model

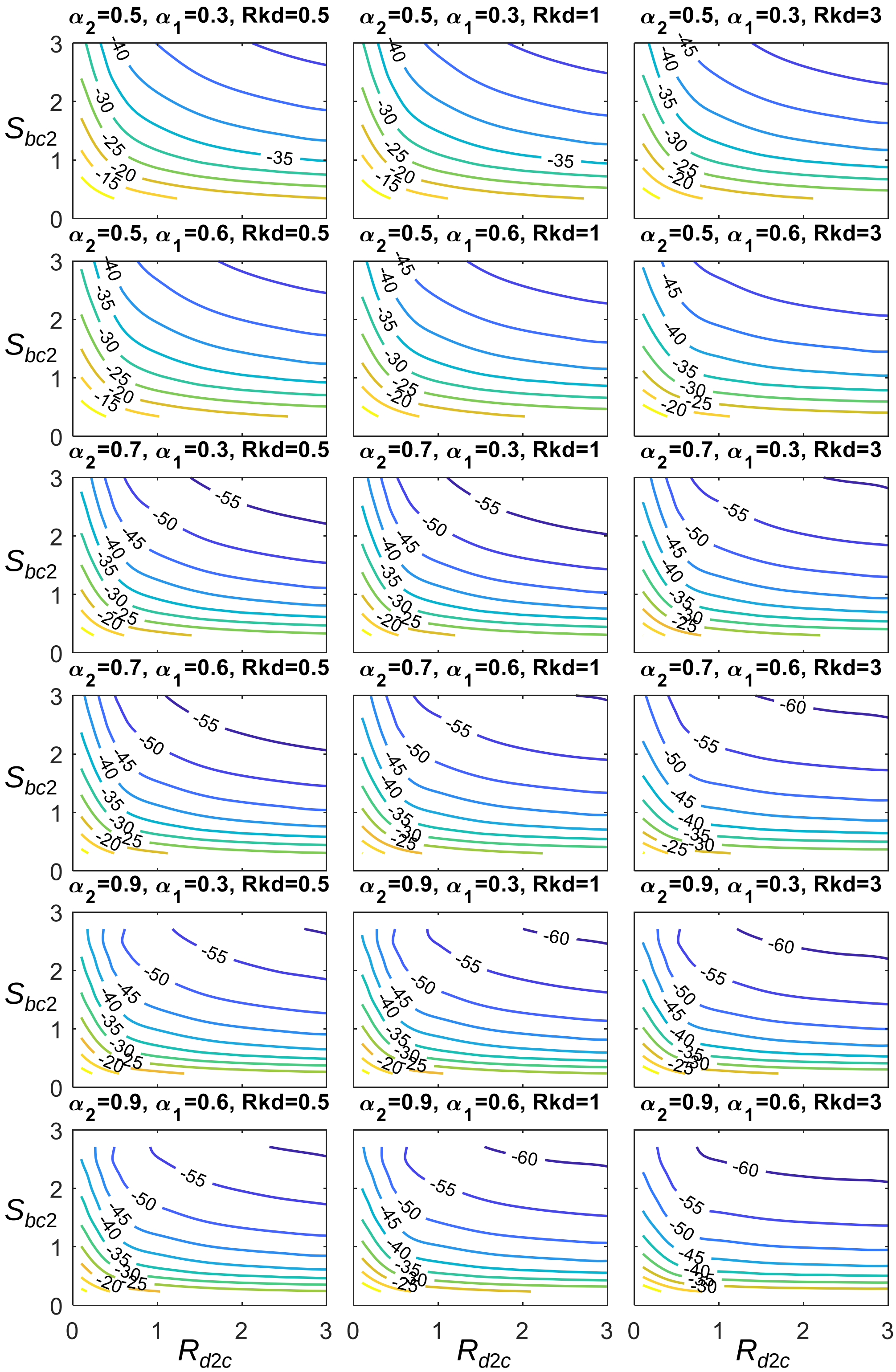


Figure 7.3.8 The $\theta_{\max}$ reduction factor distributions with respect to R_{d2c} and S_{bc2}

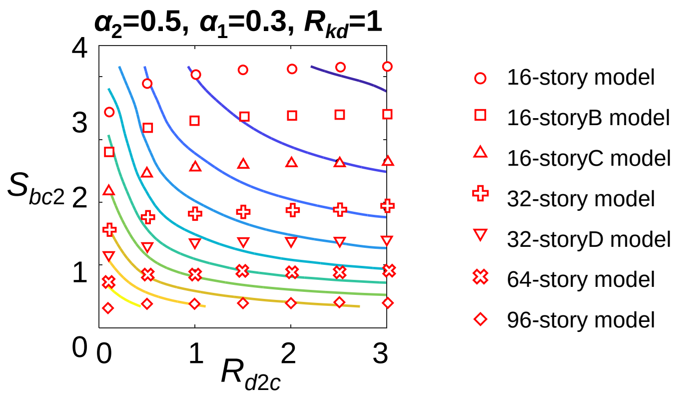


Figure 7.3.9 Illustration of the data distribution with respect to S_{bc2} and R_{d2c}

7.3.2 Maximum inter-story drift

Figure 7.3.10, Figure 7.3.11, Figure 7.3.12 show the analysis results of the maximum inter-story drift (γ_{max}) when α_1 and α_2 vary from 0 to 1 and when R_{kd} equals to 0.5, 1, and 3, respectively, calculated from the SA and NRLHA with BCJ-L2 ground motion. The analysis results obtained from the SA and NLRHA are close to each other, and the distributions of γ_{max} with respect to α_1 and α_2 are similar to θ_{max}. The values of γ_{max} primary change with α_2, and the changes in α_1 only slightly affect the responses. The γ_{max} reaches the minimum when α_2 and α_1 are approximately 0.7 and 0.6, respectively. Similar to the responses of θ_{max}, a greater R_{d2c} value (stiffer BRB) provides a greater outrigger effect. However, the decrease in γ_{max} is not proportional to the increase of R_{d2c}. In addition, the γ_{max} becomes only slightly smaller when R_{kd} is larger. Based on the analysis results, the γ_{max} responses affected by changing α_2 and R_{d2c} are more significant than changing α_1 and R_{kd}.

Figure 7.3.13 to Figure 7.3.16 show the reduction factors in γ_{max}, which are the reductions (in percentage) if compared with the core structure model without outrigger effect, for the 16-, 32-, 64-, and 96-story models, respectively. The reduction factors of the NLRHA results are calculated based on the average of the NLRHA with 8 ground motions. The SA results are close to the NLRHA results. The γ_{max} reduction factors primarily change with α_2, and γ_{max} reaches the minimum (largest reduction in γ_{max}) when α_2 is between 0.7 and 0.8. Similar to the θ_{max} responses, the effect of varying α_1 is negligible when α_2 is smaller than 0.4. Even when α_2 is at its approximate optimal elevation (between 0.7 and 0.8), the changes in the γ_{max} reduction factor because of changing α_1 is limited to within 10%. Based on the analysis results, when α_2 is at its optimal elevation and α_1 is approximately 0.4 to

0.7, the γ_{max} can be best reduced. In the models with the same number of stories, the increases in R_{d2c} from 0.1 to 1 and from 1 to 3 increase the γ_{max} reduction factors by approximately 10% and 5%, respectively. In addition, for the models with different story numbers, the amounts of reduction factor are smaller in taller models that have smaller S_{bc2} values. In summary, in order to best reduce the γ_{max}, the optimal upper outrigger elevations (α_2) are approximately 0.7 and 0.8. For the lower outrigger elevation (α_1), the optimal α_1 is in the range of 0.4 to 0.7.

Figure 7.3.17 shows the γ_{max} reduction factor (in percentage) distributions with respect to R_{d2c}, S_{bc2}, and R_{kd} for the cases when α_2 is 0.5, 0.7, and 0.9 and when α_1 is 0.3 and 0.6 calculated from the SA. The additional analysis results calculated from using the analytical models shown in Figure 7.3.1 are included. The γ_{max} reduction factor distributions are similar to the θ_{max} reduction factor distributions. The greater values of R_{d2c} and S_{bc2} result in smaller γ_{max} responses (smaller reduction factors). However, the rate of increase in the amount of reduction factors decreases or stops, when R_{d2c} increases under a fixed S_{bc2} value. Thus, the optimal value of R_{d2c} is approximately 0.5 to 1.5. When R_{d2c} is greater than 1.5, the reduction in γ_{max} becomes less efficient. In addition, if the cases when α_2 varies between 0.5, 0.7, and 0.9 are compared, when α_2 is changed from 0.5 to 0.7, the γ_{max} reduction factors increase by approximately 10%, and when α_2 is changed from 0.7 to 0.9, the γ_{max} reduction factors increase by approximately 4%. The analysis results suggest that the optimal upper BRB-outrigger elevation in order to mitigate γ_{max} should be between 0.7 to 0.9. In addition, when S_{bc2} is larger than approximate 1.5, the increase in R_{kd} value is more efficient in reducing γ_{max} response.

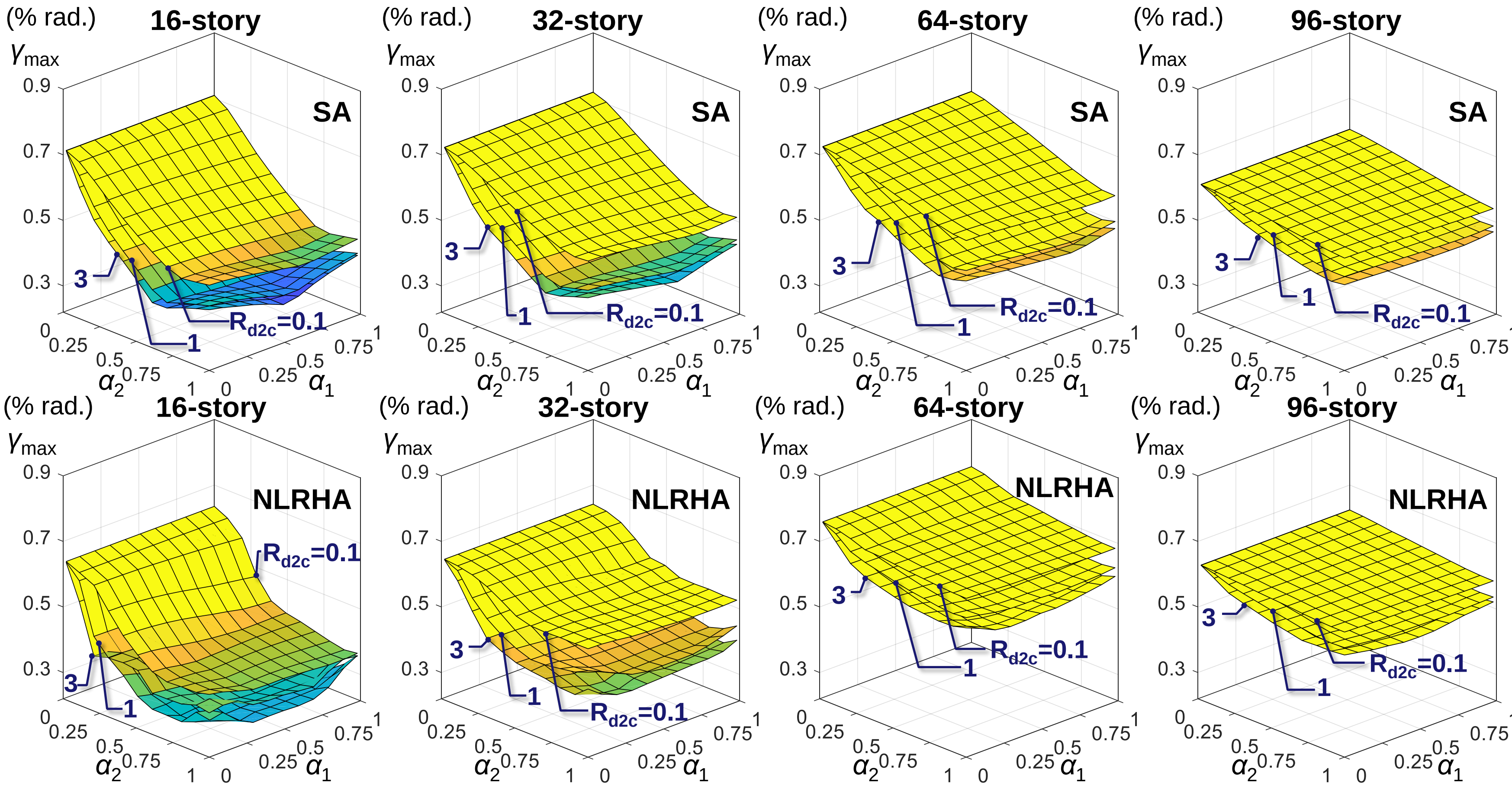


Figure 7.3.10 The $\gamma_{\max}$ distribution when R_{kd} = 0.5 calculated from SA and NLRHA with BCJ-L2 ground motion

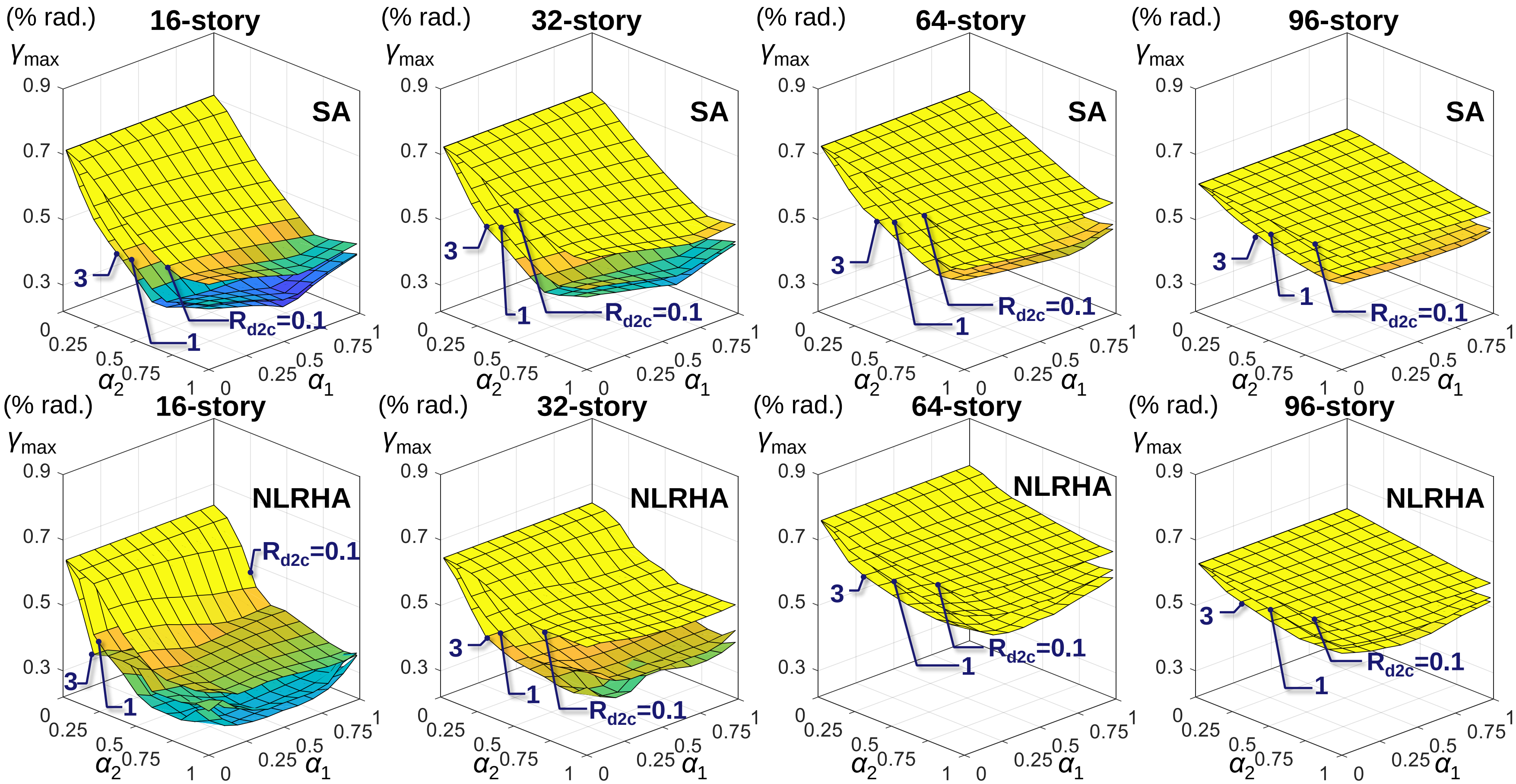


Figure 7.3.11 The γ_{max} distribution when R_{kd}=1 calculated from SA and NLRHA with BCJ-L2 ground motion

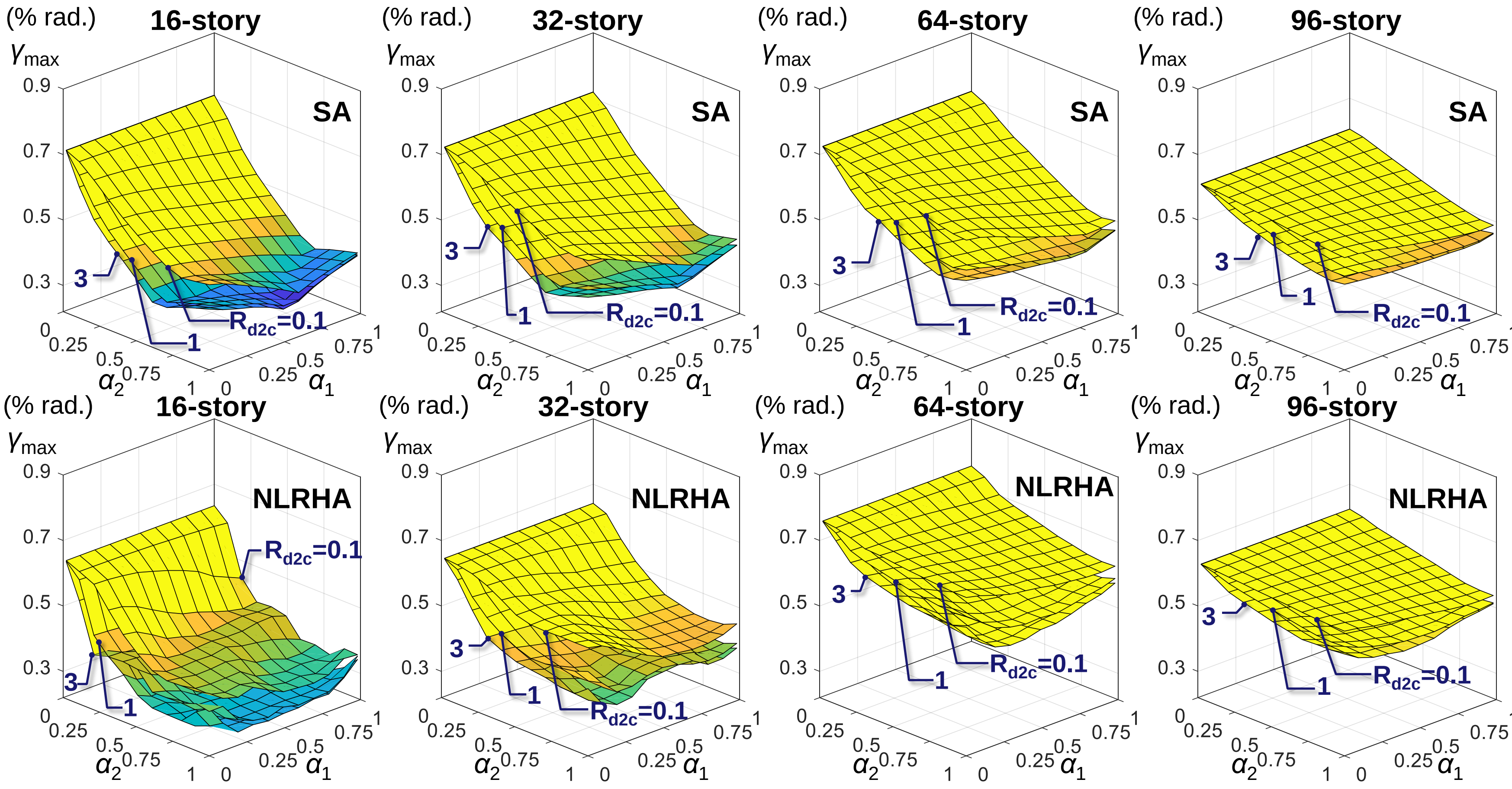


Figure 7.3.12 The γ_{max} distribution when R_{kd}=3 calculated from SA and NLRHA with BCJ-L2 ground motion

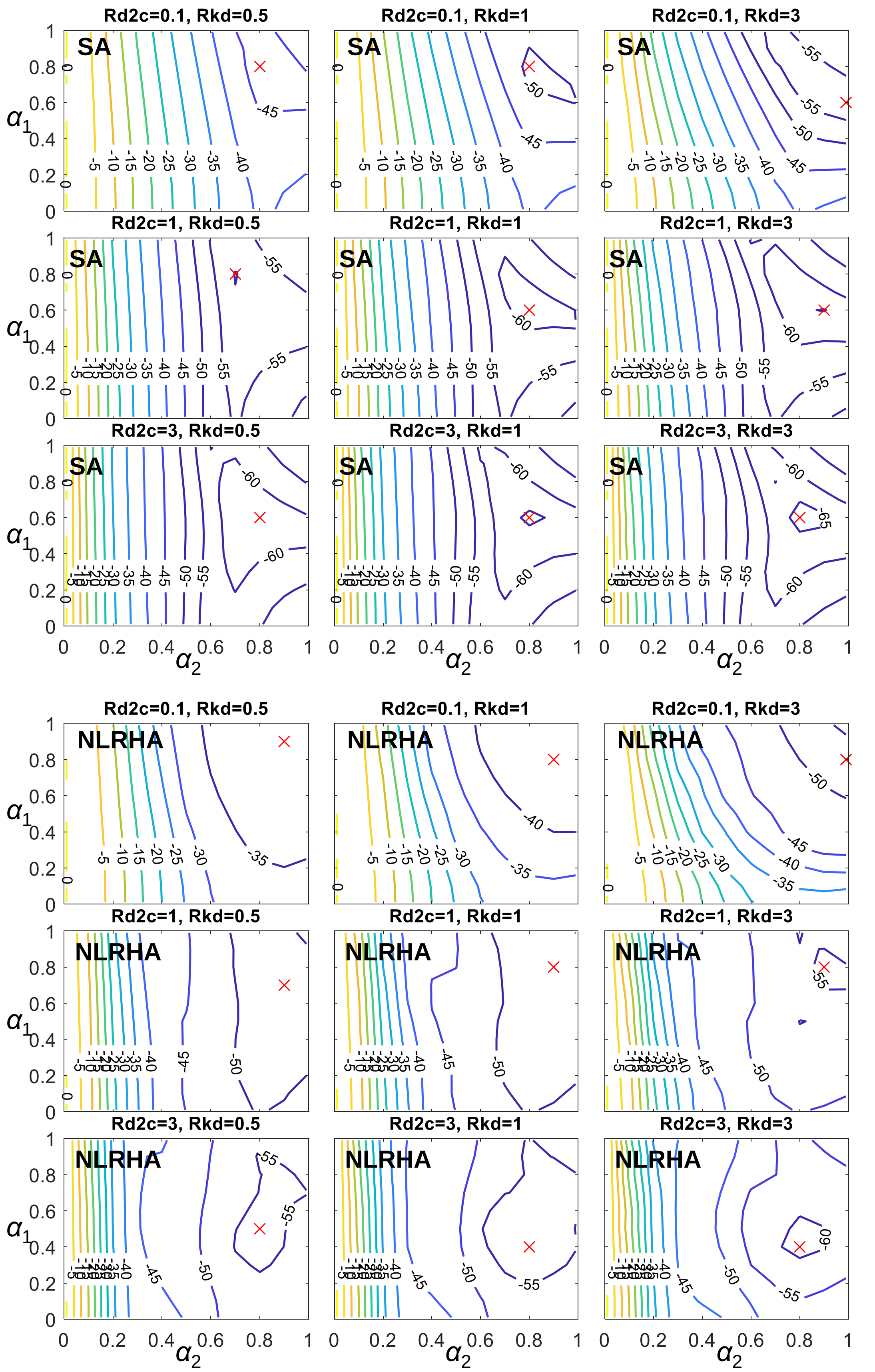


Figure 7.3.13 The $\gamma_{\max}$ reduction factor distributions with various outrigger elevations of the 16-story model

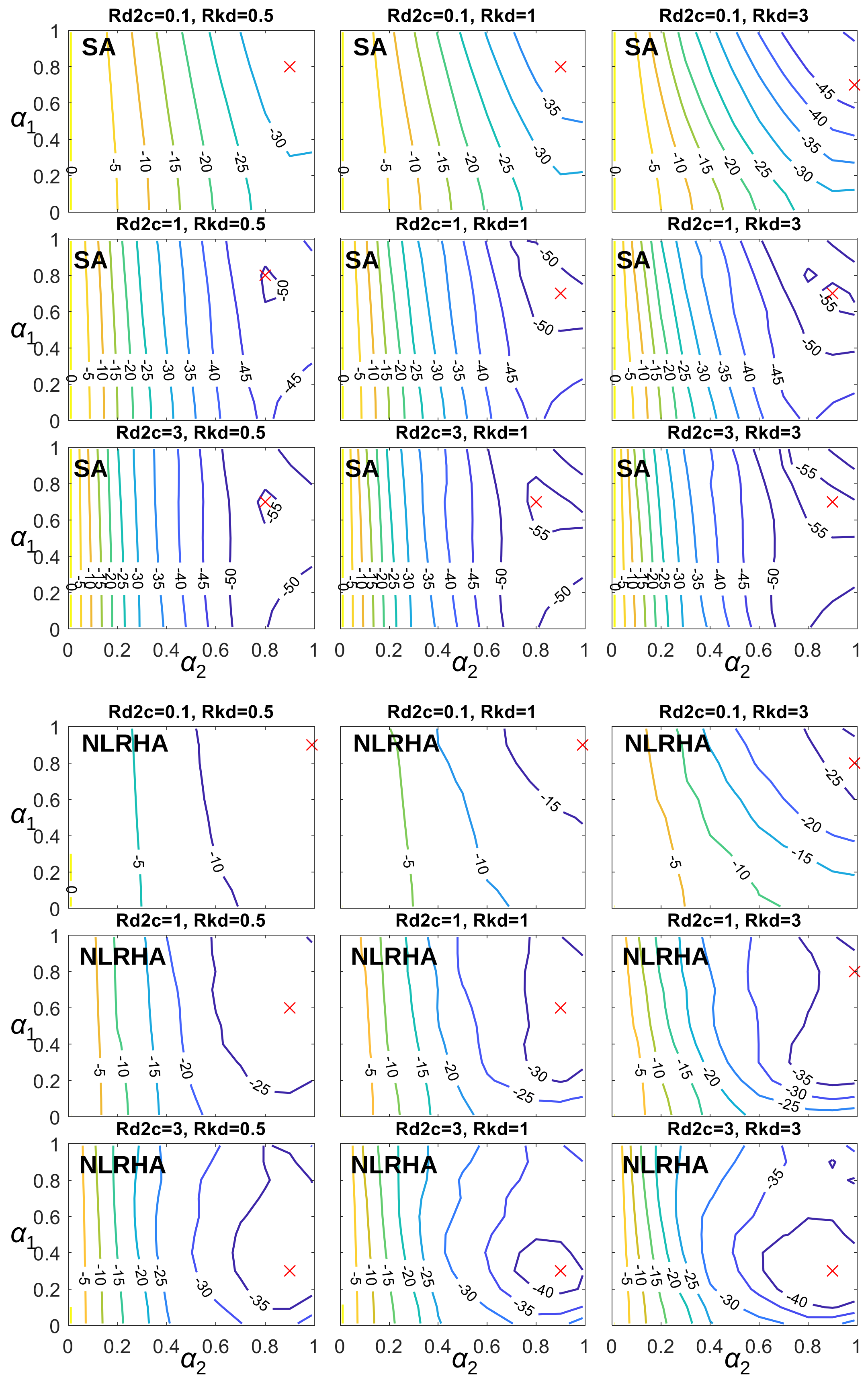


Figure 7.3.14 The $\gamma_{\max}$ reduction factor distributions with various outrigger elevations of the 32-story model

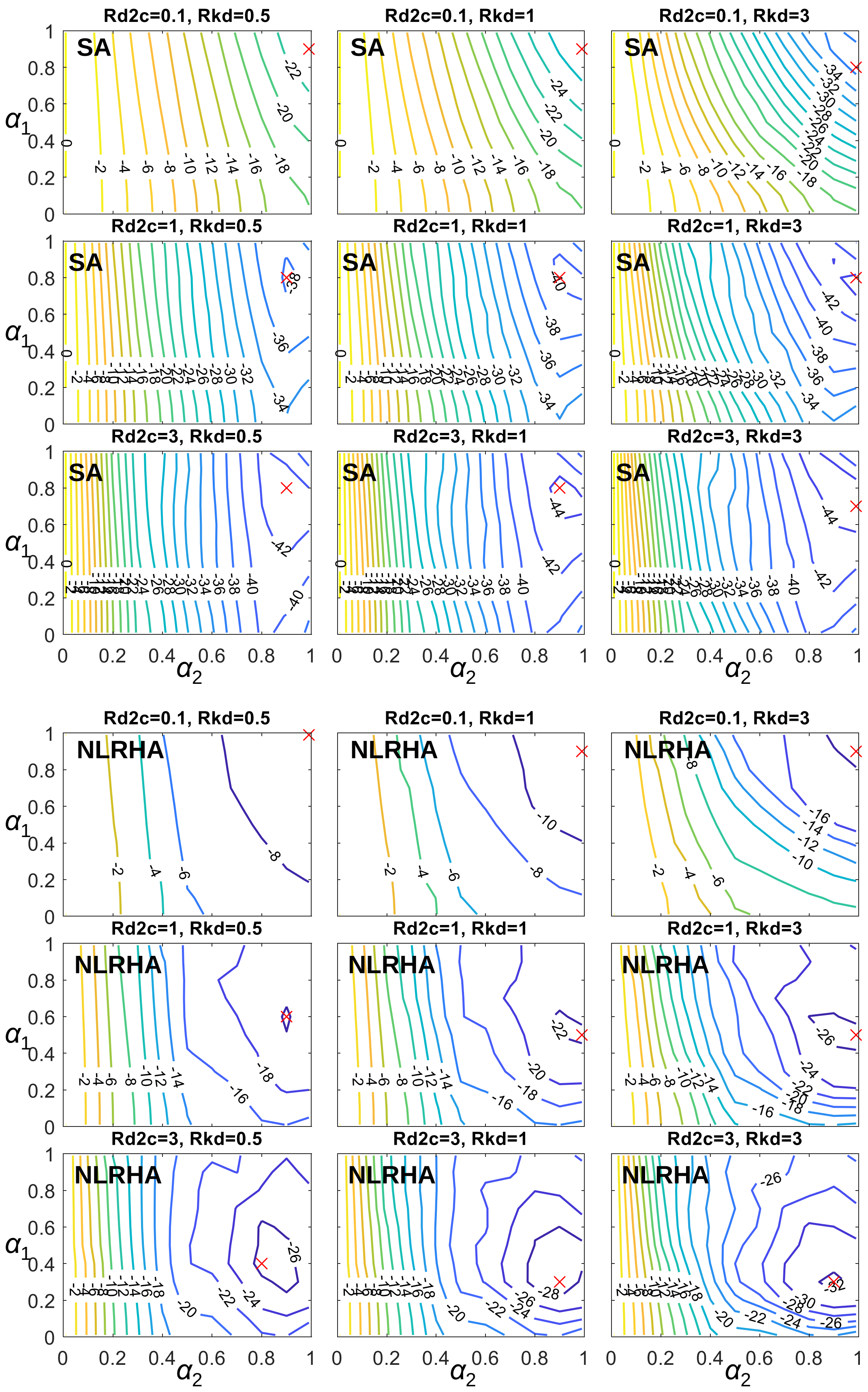


Figure 7.3.15 The γ_{max} reduction factor distributions with various outrigger elevations of the 64-story model

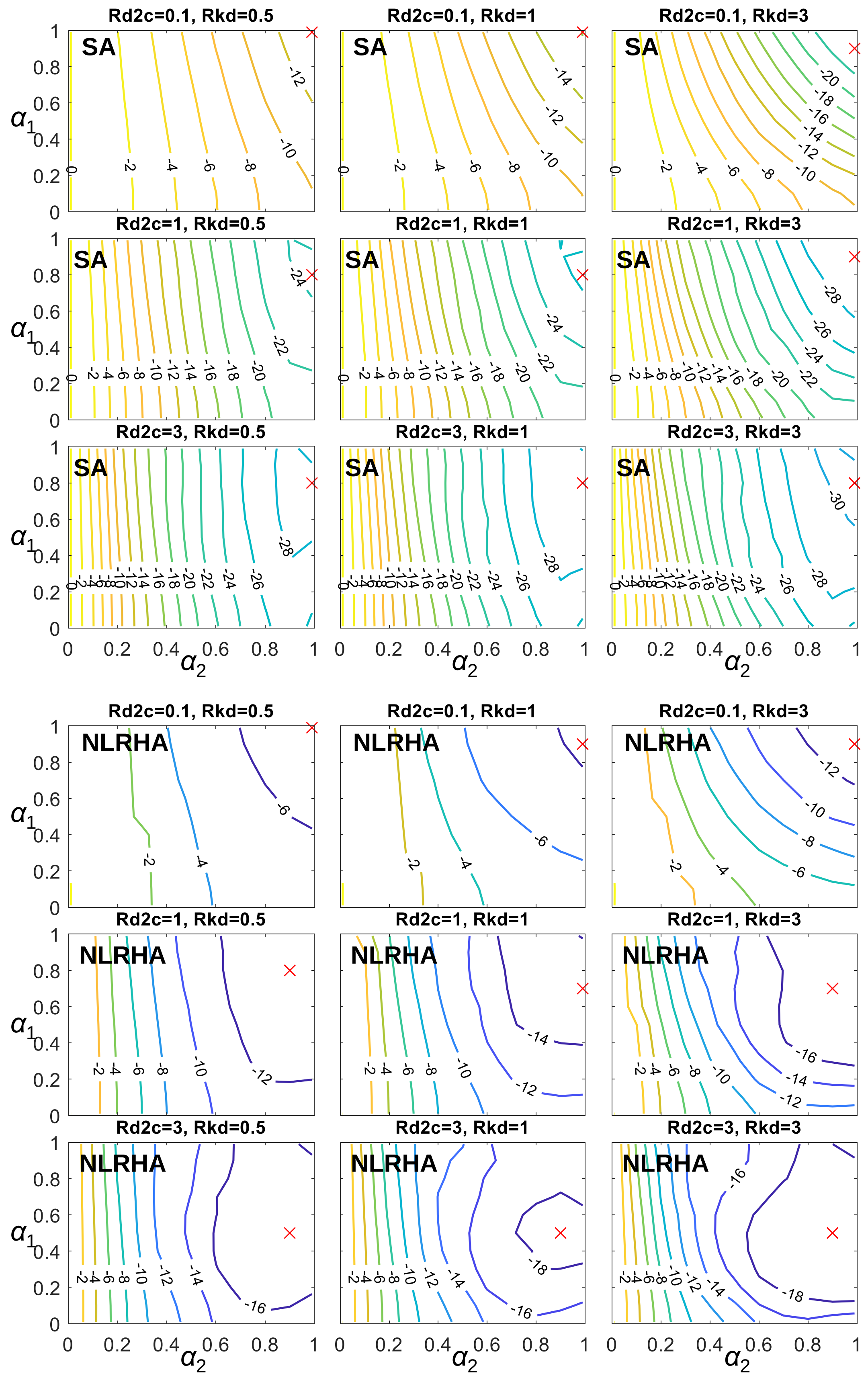


Figure 7.3.16 The γ_{max} reduction factor distributions with various outrigger elevations of the 96-story model

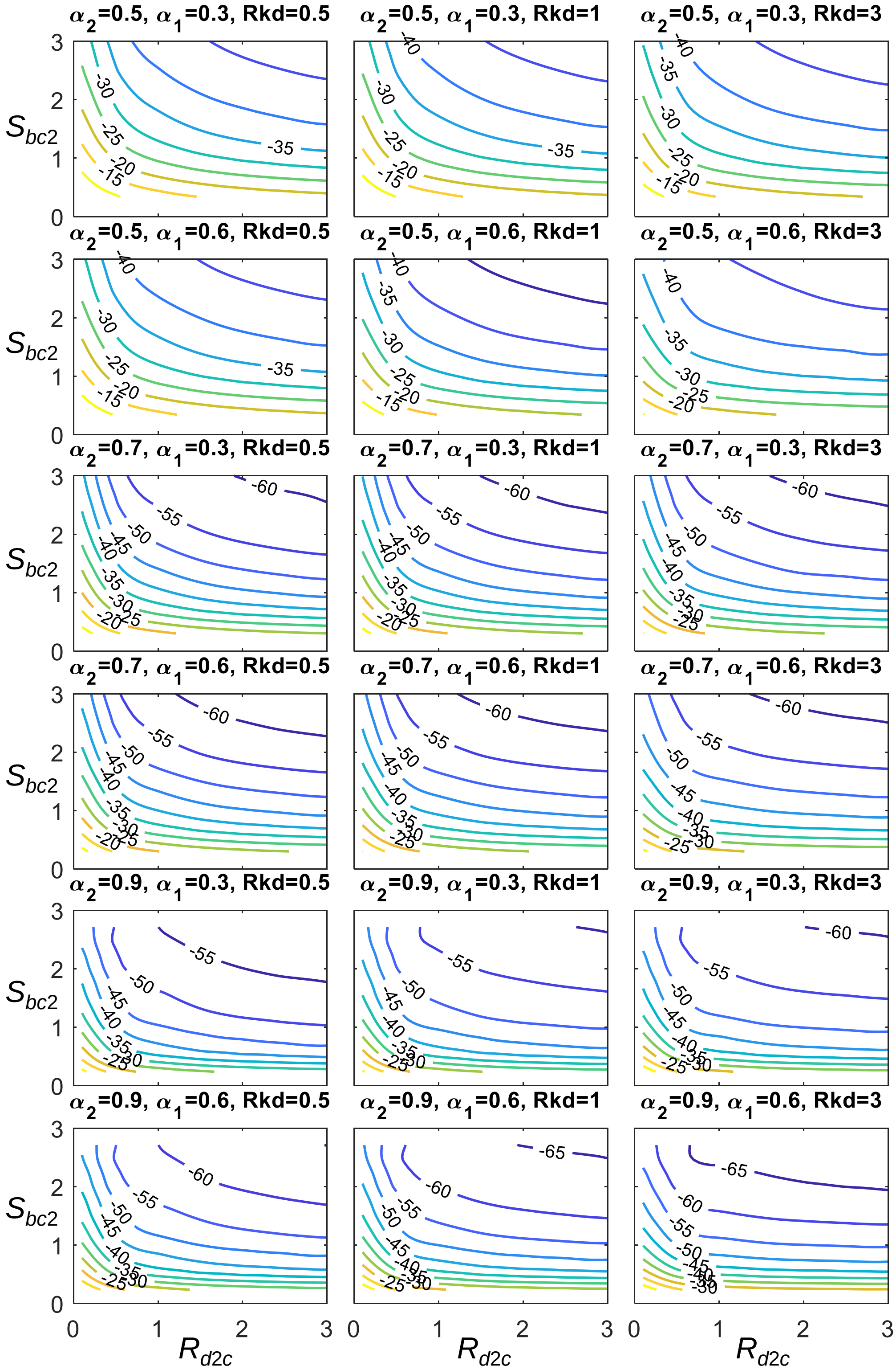


Figure 7.3.17 The $\gamma_{\max}$ reduction factor distributions with various R_{d2c} and S_{bc2}

7.3.3 Maximum overturning moment

Figure 7.3.18 to Figure 7.3.20 show the maximum overturning moment at core structure base ($M_{c,\max}$) calculated from the average of NLRHA with 8 ground motions when R_{kd} equals to 0.5, 1, and 3, respectively. As the SA procedure using the elastic mode shapes and linearly elastic force-deformation relationship to calculate the overturning moment response would be less accurate, only the $M_{c,\max}$ calculated from NLRHA are presented. The trends of $M_{c,\max}$ responses with respect to the outrigger elevations (α_1 and α_2) are similar to the $\theta_{\max}$ and $\gamma_{\max}$ responses. The values of $M_{c,\max}$ primarily change with α_2, and the effect of changing α_1 is less significant as than changing α_2. The values of $M_{c,\max}$ are minima when α_2 and α_1 are approximately 0.7 and 0.6, respectively, and decrease with increasing R_{d2c}. In addition, the decrease in $M_{c,\max}$ responses is not proportional to the increase of R_{d2c}. Furthermore, the improvements in $M_{c,\max}$ reductions because of the increases of R_{kd} are less significant. Similar to the $\theta_{\max}$ and $\gamma_{\max}$ responses, the changes in α_2 and R_{d2c} would affect the $M_{c,\max}$ response more than the changes in α_1 and R_{kd}.

Figure 7.3.21 to Figure 7.3.24 show the $M_{c,\max}$ reduction factors (compare with the core structure without outrigger effect, in percentage) for the 16-, 32-, 64-, and 96-story models, respectively. The shapes of the $M_{c,\max}$ reduction factor distribution are similar to the $\theta_{\max}$ and $\gamma_{\max}$ responses. The $M_{c,\max}$ reduction factors primarily change with α_2, and the values are minimum when α_2 is 0.7 and 0.8. The effect of varying α_1 is negligible when α_2 is smaller than 0.4. Under a given α_2 value, the changes in $M_{c,\max}$ reduction factor because of varying α_1 is less than 5%. Based on the analysis results, when α_2 is at its optimal elevation and α_1 is approximately 0.2 to 0.6, the $M_{c,\max}$ can be best reduced. The larger R_{kd} value results in smaller optimal α_1. In addition, a greater value of S_{bc2} results in smaller $M_{c,\max}$ responses. In summary, the optimal upper outrigger elevations (α_2) are approximate 0.7 and 0.8, and the optimal lower outrigger elevations (α_1) are approximate 0.2 to 0.6 in order to minimize $M_{c,\max}$.

Figure 7.3.25 shows the $M_{c,\max}$ reduction factor (in percentage) distributions with respect to R_{d2c} and S_{bc2}, when α_2 is 0.5, 0.7, and 0.9 and when α_1 is 0.3 and 0.6, calculated from the average of NLRHA with 8 ground motions. The additional analysis results using the analytical models shown in Table 7.3.1 are also included. The $M_{c,\max}$ reduction factor distributions are similar to the $\theta_{\max}$ and $\gamma_{\max}$ responses. The greater values of R_{d2c} and S_{bc2} result in smaller $M_{c,\max}$ responses. However, the rate of increase in the amounts of $M_{c,\max}$ reduction factor decrease or even stop as R_{d2c}

increases under a fixed value of S_{bc2}. The optimal value of R_{d2c} should be approximately 0.5 to 1.5. The change in R_{kd} has limited improvement the $M_{c,\max}$ responses. However, when α_1 decreases from 0.6 to 0.3, the $M_{c,\max}$ reduction factors increase by approximately 5% for the cases when α_2 is 0.5, and increase by approximately 10% for the cases when α_2 is 0.7 and 0.9. Based on the analysis results, in order to mitigate $M_{c,\max}$, the optimal upper BRB-outrigger elevation (α_2) is approximately 0.7 to 0.9. In addition, if α_2 is within this optimal range, the reduction in $M_{c,\max}$ is optimal when the lower BRB-outrigger is approximate 0.2 to 0.6. The optimal α_1 would be lower when the value of R_{kd} is larger.

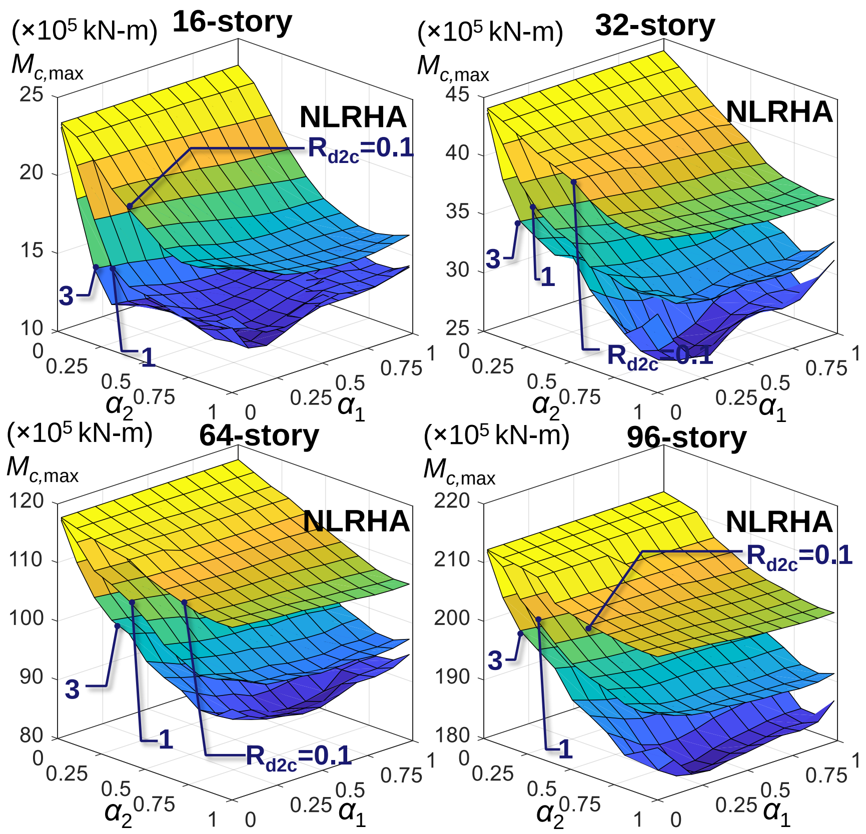


Figure 7.3.18 The $M_{c,\max}$ distribution when R_{kd} = 0.5 calculated from NLRHA

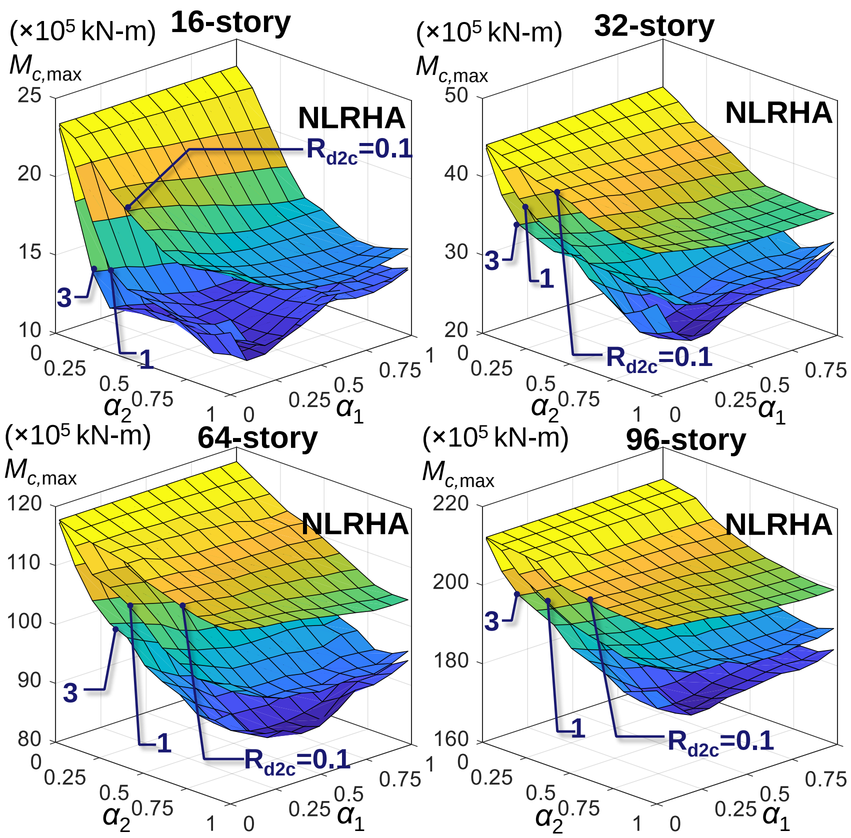


Figure 7.3.19 The $M_{c,\max}$ distribution when R_{kd} = 1 calculated from NLRHA

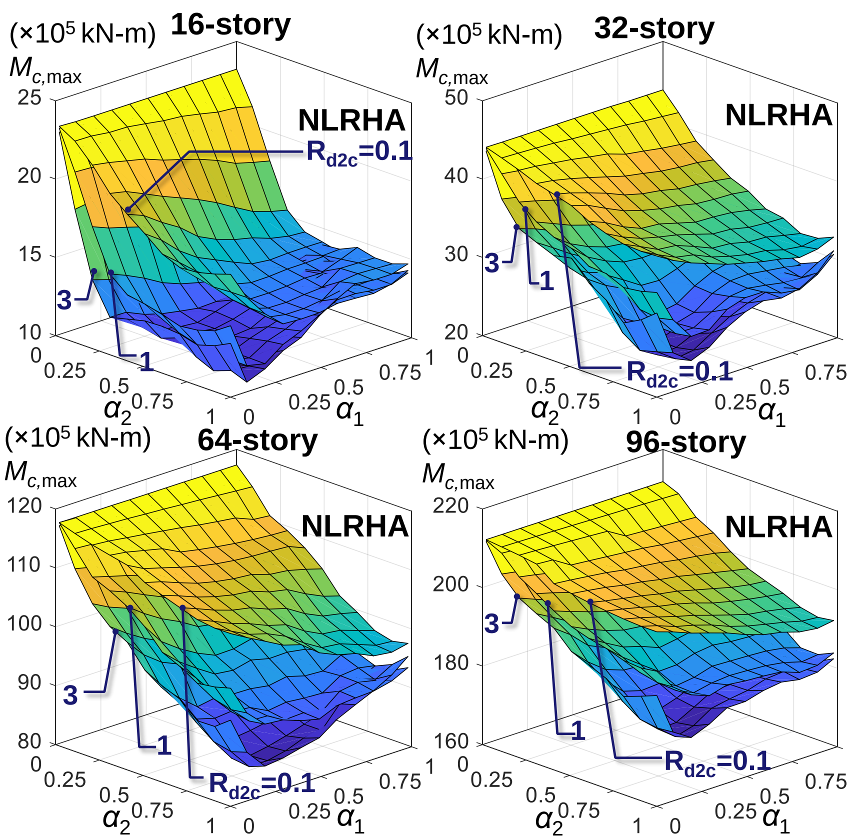


Figure 7.3.20 The $M_{c,\max}$ distribution when R_{kd} = 3 calculated from NLRHA

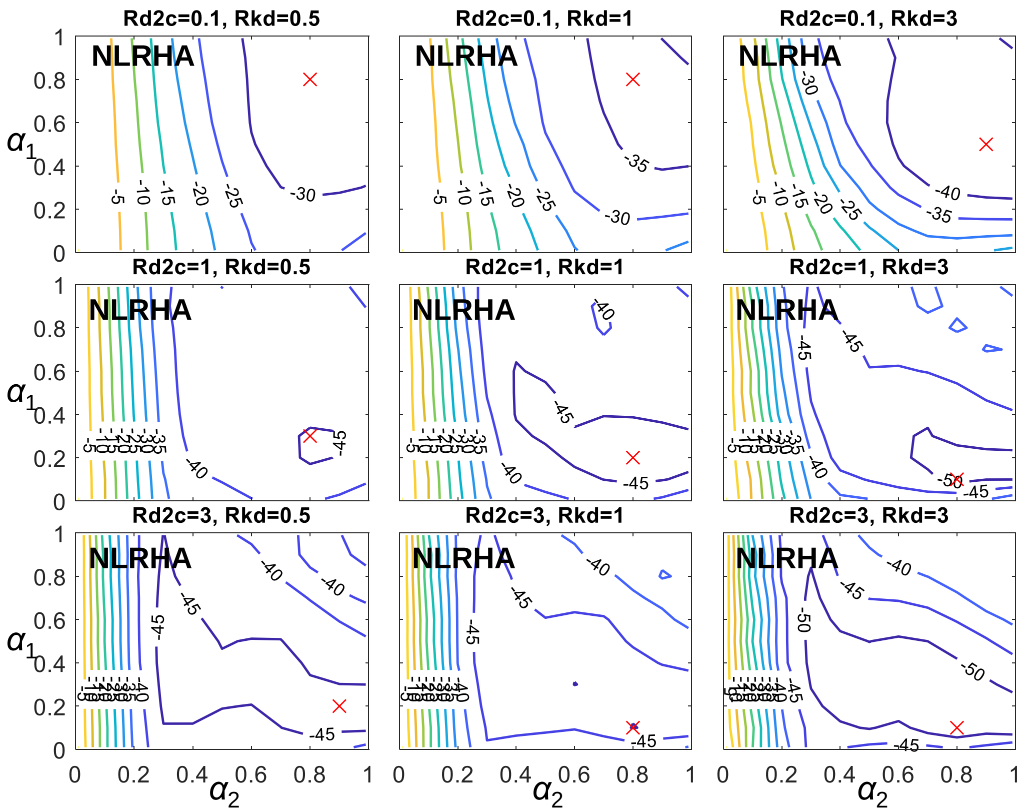


Figure 7.3.21 The $M_{c,\max}$ reduction factor distributions with various outrigger elevations of the 16-story model

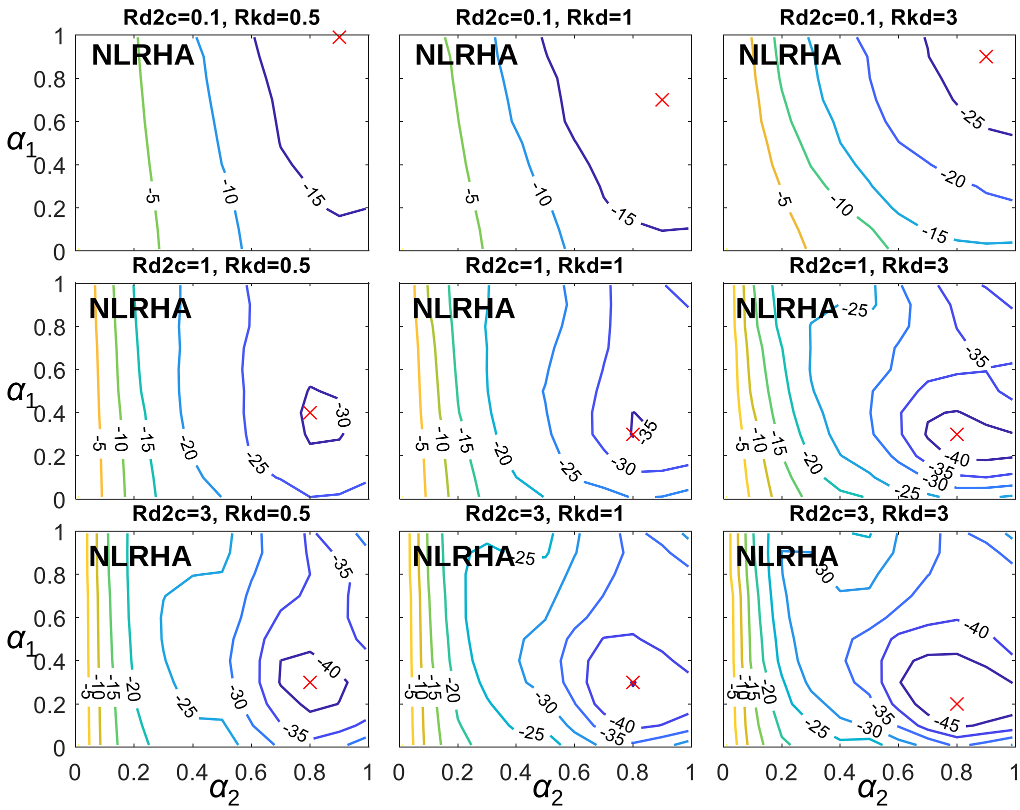


Figure 7.3.22 The $M_{c,\max}$ reduction factor distributions with various outrigger elevations of the 32-story model

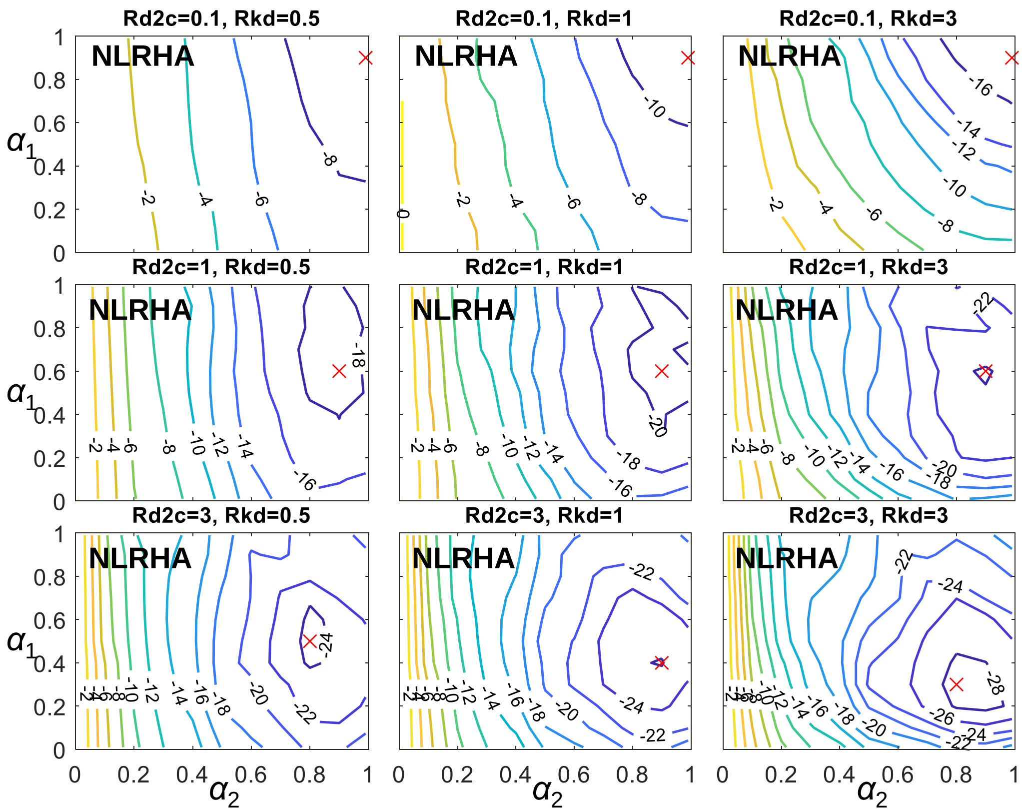


Figure 7.3.23 The $M_{c,\max}$ reduction factor distributions with various outrigger elevations of the 64-story model

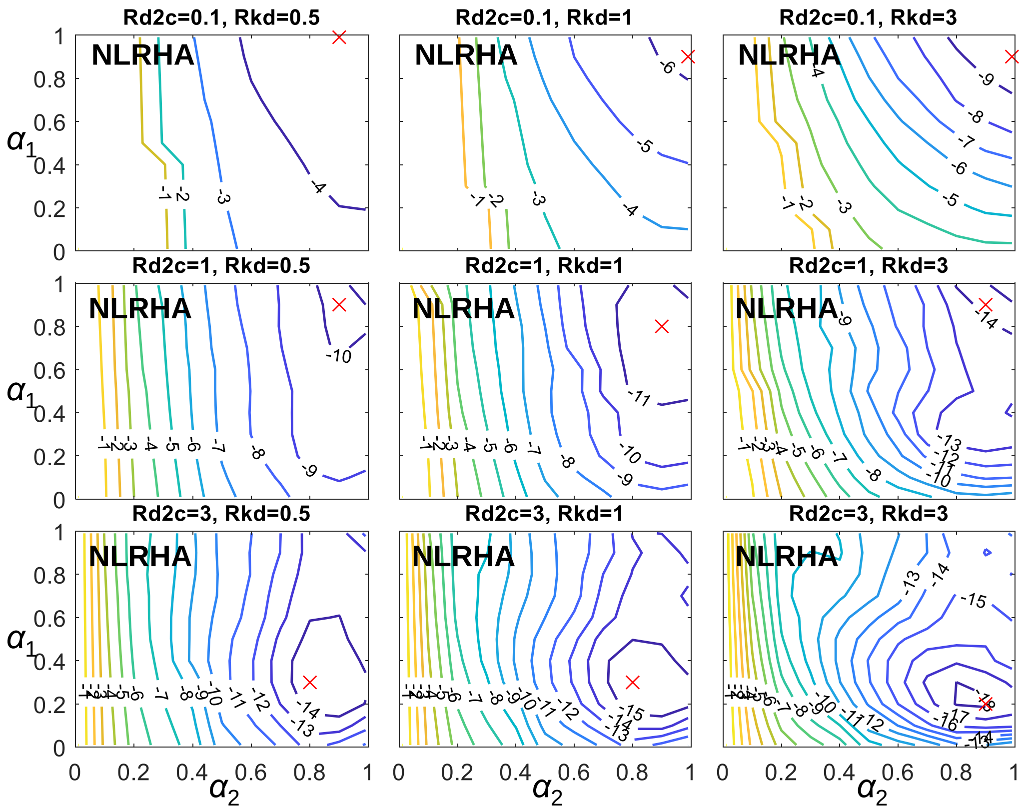


Figure 7.3.24 The $M_{c,\max}$ reduction factor distributions with various outrigger elevations of the 96-story model

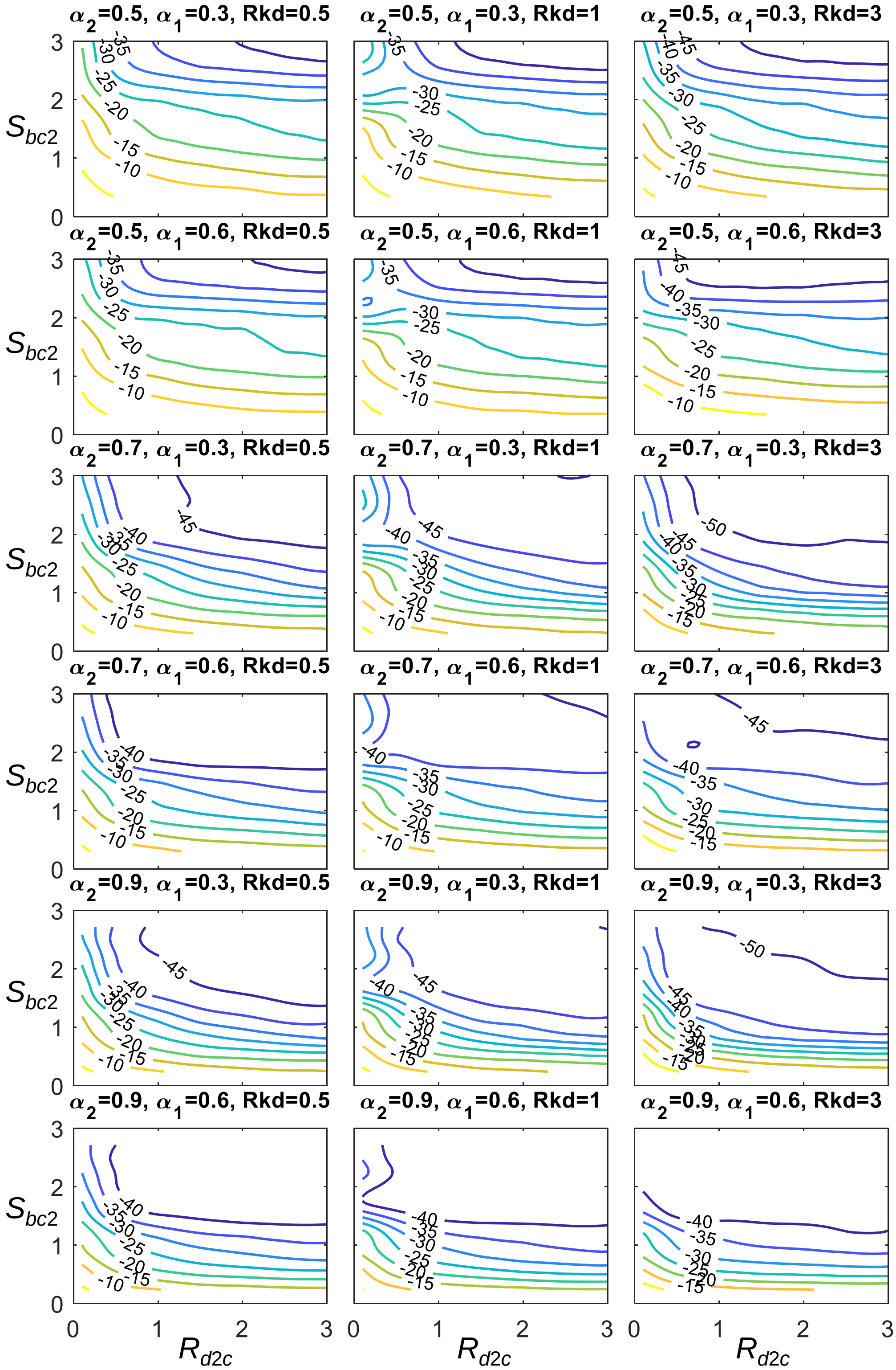


Figure 7.3.25 The $M_{c,\max}$ reduction factor distributions with various R_{d2c} and S_{bc2}

7.3.4 Maximum perimeter column axial force

Figure 7.3.26 to Figure 7.3.28 show the analysis results of maximum perimeter column axial force ($C_{1,max}$) calculated from the average of NLRHA results using 8 ground motions when R_{kd} equals to 0.5, 1, and 3, respectively. The values of $C_{1,max}$ primarily change with α_2. The larger α_1 would result in smaller $C_{1,max}$, and this effect is more significant when R_{kd} is larger. In addition, the larger value of R_{d2c} also results in a larger value of $C_{1,max}$. The $C_{1,max}$ reaches its maximum when α_2 and α_1 are approximately 0.4 to 0.6 and 0, respectively. The $C_{1,max}$ distributions with respect to α_1 and α_2 can be explained by the θ_{max} and γ_{max} responses. As described in the abovementioned sections, the θ_{max} and γ_{max} are minima when α_2 and α_1 are approximately 0.7 and 0.6, respectively, and the larger value of R_{d2c} slows the rate of decreases in θ_{max} and γ_{max}. When θ_{max} and γ_{max} are minimum, the seismic responses are best reduced by utilizing the perimeter columns' axial stiffness. Therefore, the perimeter column's axial force demand is large. In addition, when R_{d2c} increases, the θ_{max} and γ_{max} do not reduce proportionally, therefore, the core structure rotation at the outrigger elevations do not reduce proportionally with the increases of R_{d2c}, either. If two models with different R_{d2c} values are compared and when the core structure rotations at outrigger elevation are similar, the total axial deformation from the BRB and perimeter column should be similar. However, the model with larger R_{d2c} value exhibits greater outrigger rotational stiffness and thus results in larger axial force in both the BRB and perimeter column. Thus, the $C_{1,max}$ increases nearly proportionally to the increases of R_{d2c} value. Based on the analysis results, the $C_{1,max}$ is almost doubled when R_{d2c} increases from 0.1 to 1 and from 1 to 3. If compare with the responses of θ_{max} and γ_{max}, the benefit of reducing seismic responses by increasing R_{d2c} becomes negligible when R_{d2c} is too large, however, the $C_{1,max}$ keeps increasing at the same rate with increasing R_{d2c}. Too large $C_{1,max}$ is not desirable, as it may increase the perimeter column sizes.

Figure 7.3.29 to Figure 7.3.32 show the distributions of $C_{1,max}$ for the 16-, 32-, 64-, and 96-story models, respectively. The effect of varying α_1 on $C_{1,max}$ is almost negligible when α_2 is smaller than 0.4. If compare with the θ_{max} and γ_{max} responses, the increase in R_{d2c} from 1 to 3 only increases the amount of θ_{max} and γ_{max} reduction factors by approximately 5%, however, the perimeter column axial force demand ($C_{1,max}$) is increased by nearly 50%. In summary, the $C_{1,max}$ could be significantly increased by strong outrigger effect. The larger R_{d2c} value helps in reducing seismic responses such as θ_{max} and γ_{max}, however, the effect becomes less significant when R_{d2c} is too large, but the $C_{1,max}$ is considerably increased. For the design practices, the perimeter columns are designed primarily based on the gravity load demands, the additional axial force demands due to the outrigger effect must be confirmed.

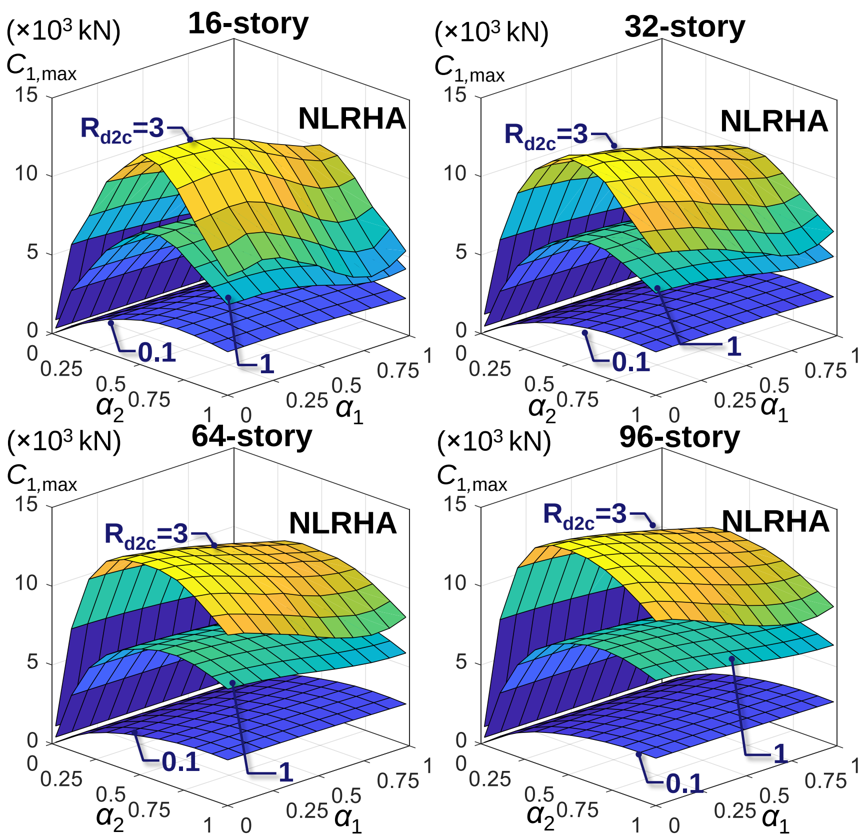


Figure 7.3.26 The $C_{1,max}$ distribution when R_{kd} = 0.5 calculated from NLRHA

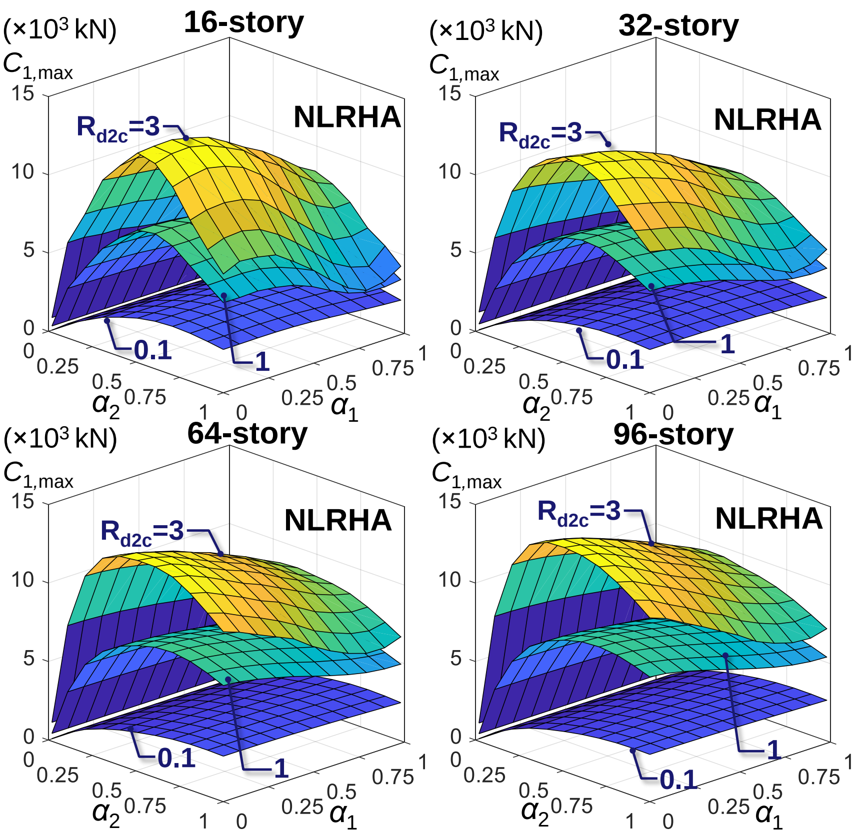


Figure 7.3.27 The $C_{1,\max}$ distribution when R_{kd} = 1 calculated from NLRHA

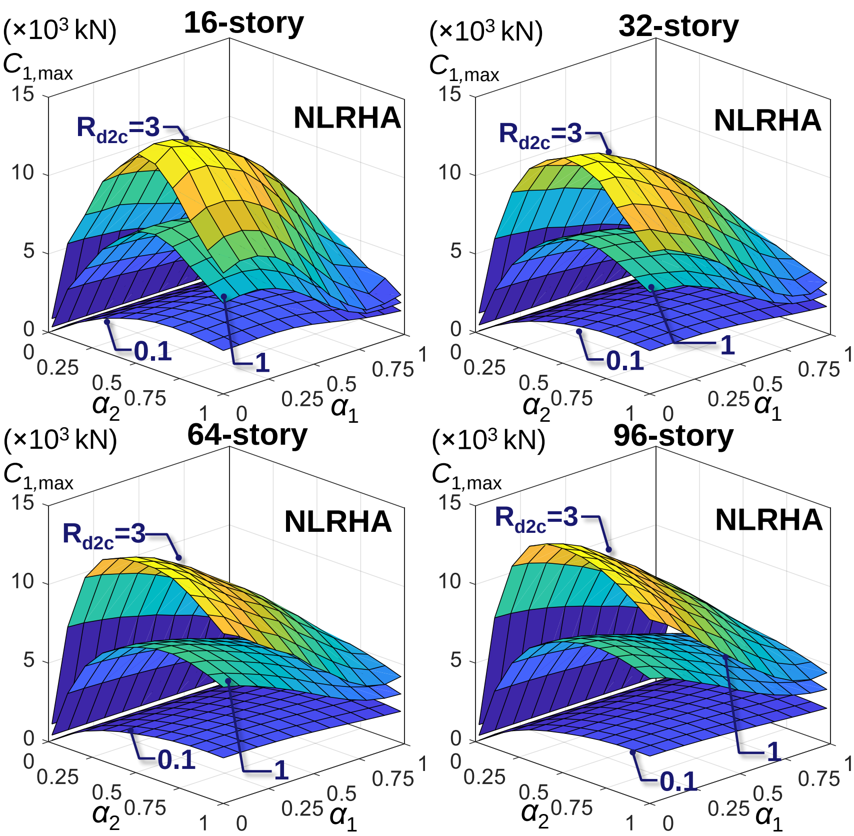


Figure 7.3.28 The $C_{1,\max}$ distribution when R_{kd} = 3 calculated from NLRHA

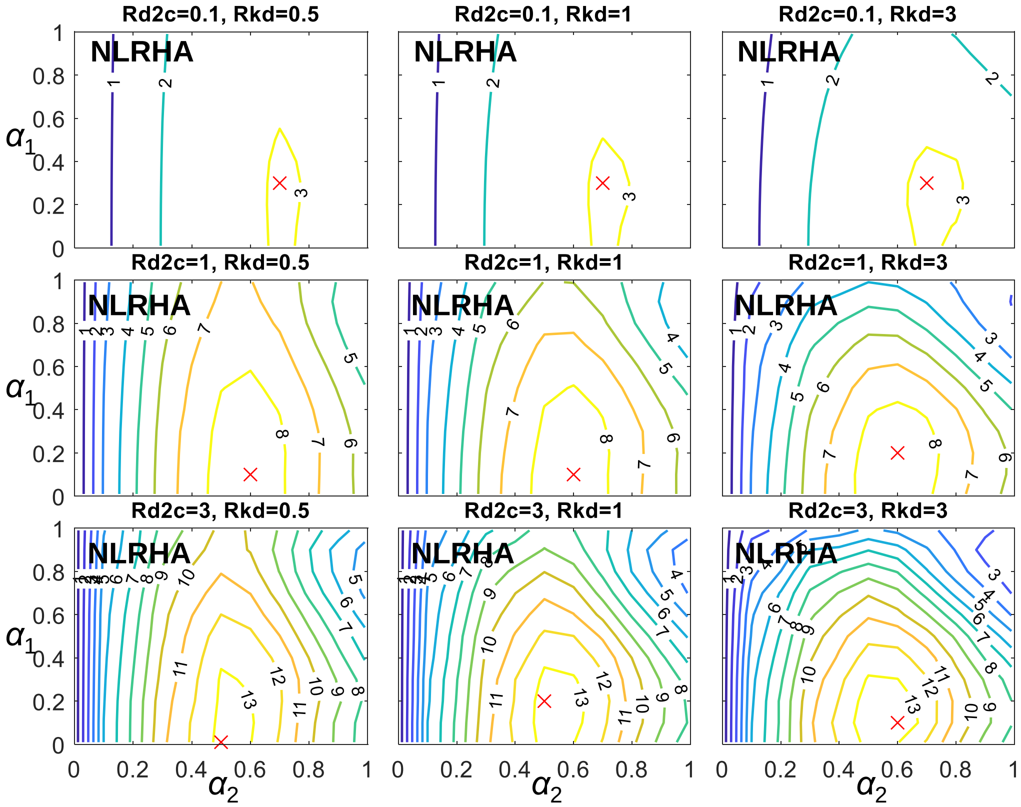


Figure 7.3.29 The $C_{1,max}$ distributions with various outrigger elevations of the 16-story model (unit: $\times 10^3$ kN)

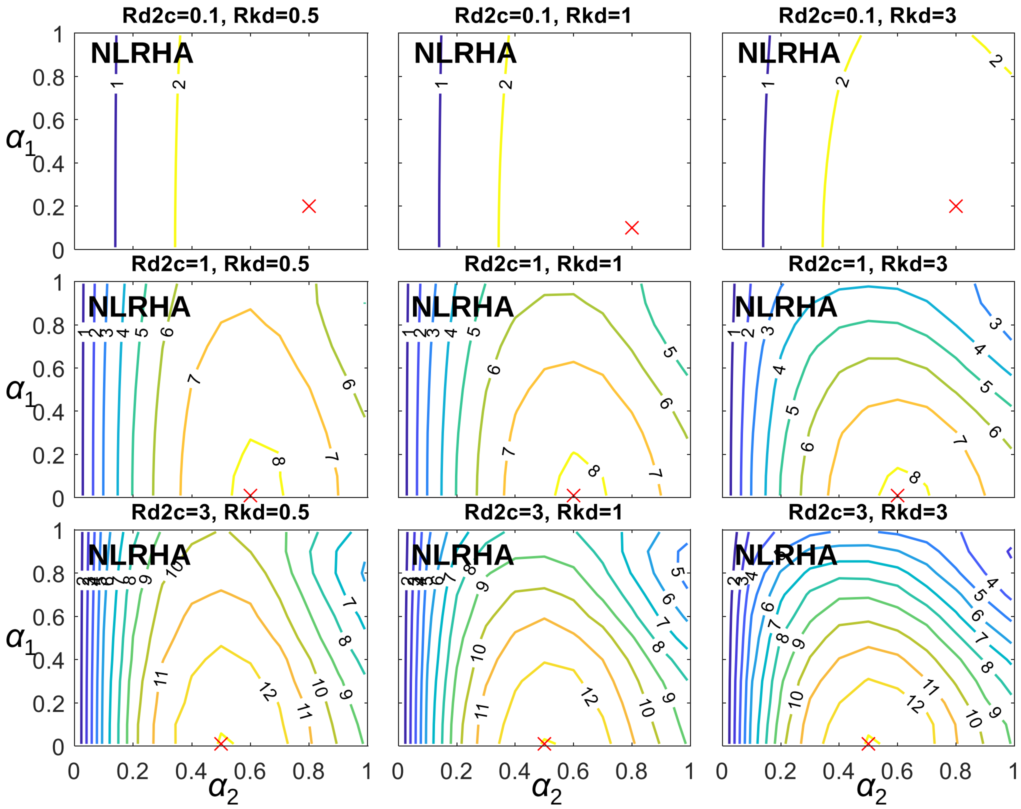


Figure 7.3.30 The $C_{1,max}$ distributions with various outrigger elevations of the 32-story model (unit: $\times 10^3$ kN)

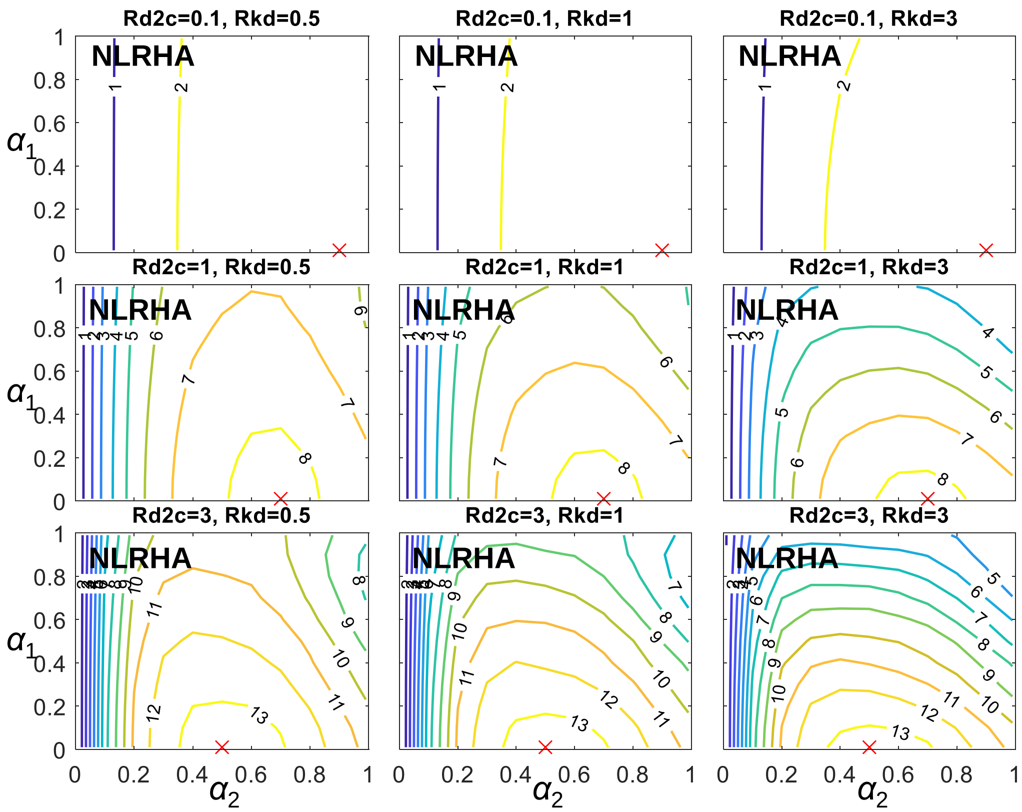


Figure 7.3.31 The $C_{1,\max}$ distributions with various outrigger elevations of the 64-story model (unit: $\times 10^3$ kN)

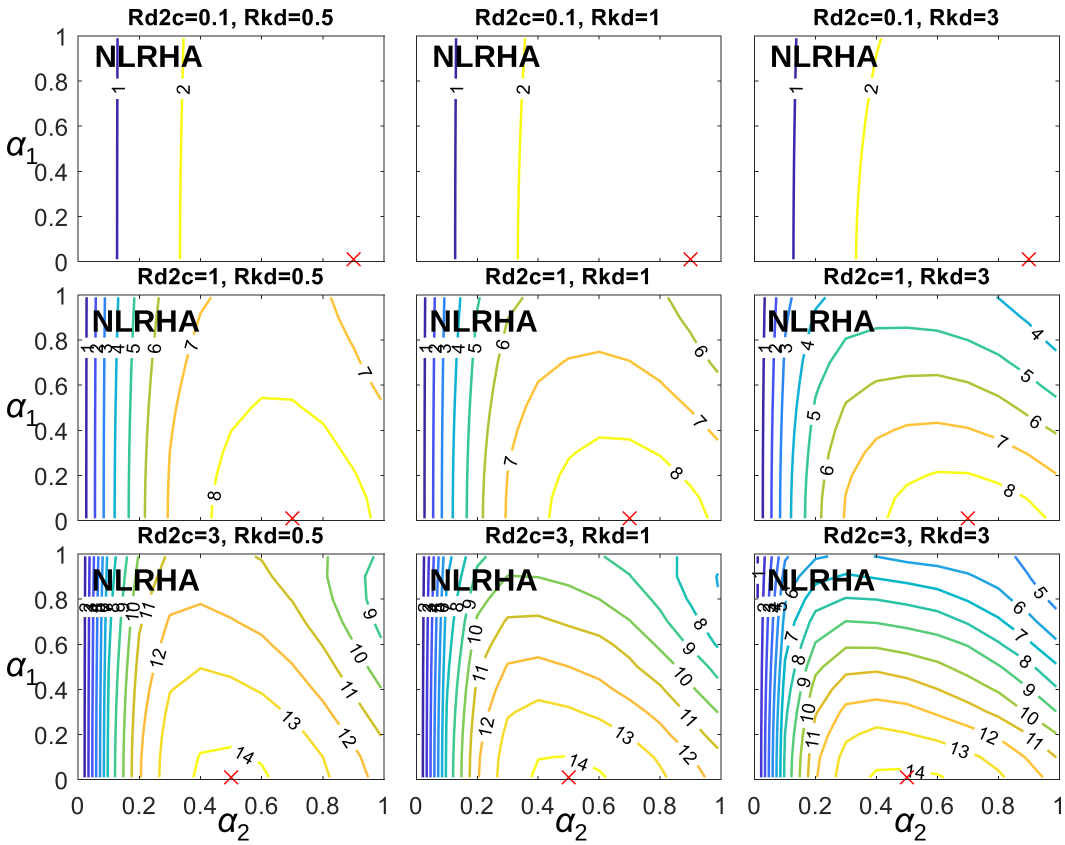


Figure 7.3.32 The $C_{1,\max}$ distributions with various outrigger elevations of the 96-story model (unit: $\times 10^3$ kN)

7.3.5 BRB energy dissipation efficiency

Figure 7.3.33 to Figure 7.3.36 show the R_{CPD} (ratio of BRB's cumulative plastic deformation to axial yield deformation) (ANSI/AISC 341-16, 2016) distributions with respect to α_1 and α_2 of the BRBs in the lower (R_{CPD1}) and upper (R_{CPD2}) outriggers of the 16-, 32-, 64-, and 96-story models, respectively. Figure 7.3.37 to Figure 7.3.40 show the distributions of the ratio of energy dissipated by the BRB in the lower (E_{BRB1}) or upper (E_{BRB2}) outrigger to the total input energy for the 16-, 32-, 64-, and 96-story models, respectively. The R_{CPD1}, R_{CPD2} values, and the energy ratios are calculated from the average of NLRHA with 8 ground motions and are used to indicate the energy dissipation efficiency of the BRBs. The distributions of R_{CPD1} and R_{CPD2} values with respect to α_1 and α_2 are similar to the $u_{d,y1}$ and $u_{d,y2}$ distributions (Figure 6.4.2, Case4 and Figure 6.4.1). When α_1 and α_2 are close to 0, since the yield deformation of the BRB in lower outrigger ($u_{d,y1}$) is smaller (Figure 6.4.2, Case4), the R_{CPD1} values are very large. Both the values of R_{CPD1} and E_{BRB1} vary with α_1 and α_2. The R_{CPD1} is smallest when α_2 and α_1 are approximately 0.7 and 0.6, respectively, and starts increasing when α_1 and α_2 increase. The E_{BRB1} is 0 when both α_1 and α_2 are 0, and starts increasing with the increasing α_1 and α_2. The E_{BRB1} reaches the maximum when α_2 and α_1 are approximately 1 and 0.8 to 0.9, respectively. The analysis results indicate that when the BRB in lower outrigger (BRB_1) is close to the upper outrigger elevation (when α_1 is close to the range between 0.8 and 0.9), the BRB_1 could exhibit the best energy dissipation efficiency without generating a very large value of R_{CPD1}. On the contrary, when α_2 is approximately between 0 and 0.3, the R_{CPD1} values are very large but the corresponding E_{BRB1} are small. The BRB_1 could easily use up its ductility life and eventually fracture without performing a good energy dissipation efficiency. The values of R_{CPD2} primarily change with α_2. The R_{CPD2} reaches the minimum when α_2 is approximately between 0.4 and 0.6. The values of E_{BRB2} change with both α_1 and α_2. When R_{kd} increases, the effect from the lower BRB-outrigger elevation (α_1) on E_{BRB2} becomes more significant. Based on the analysis results, when the lower BRB-outrigger elevation is close to the upper BRB-outrigger (when α_1 is close to 1), the amount of energy dissipated by BRB_1 (E_{BRB1}) decreases. However, the value of R_{CPD2} only slightly increase when α_1 increases under a fixed α_2. In addition, when α_2 is smaller than 0.2, the BRB_1 would not affect the E_{BRB2} and R_{CPD2}. The larger value of R_{d2c} results in greater values of E_{BRB1}, $E_{BRB2,}$ R_{CPD1}, and R_{CPD2}. This is because after the BRBs yield, the drops of stiffness in the models with large R_{d2c} value are greater than the models with small R_{d2c} value. Therefore, the deformation concentration in the BRB would be more severe in the models with large R_{d2c} value and dissipate a greater amount of energy. In addition, the larger value of R_{kd} also increase the values of R_{CPD1} and E_{BRB1}.

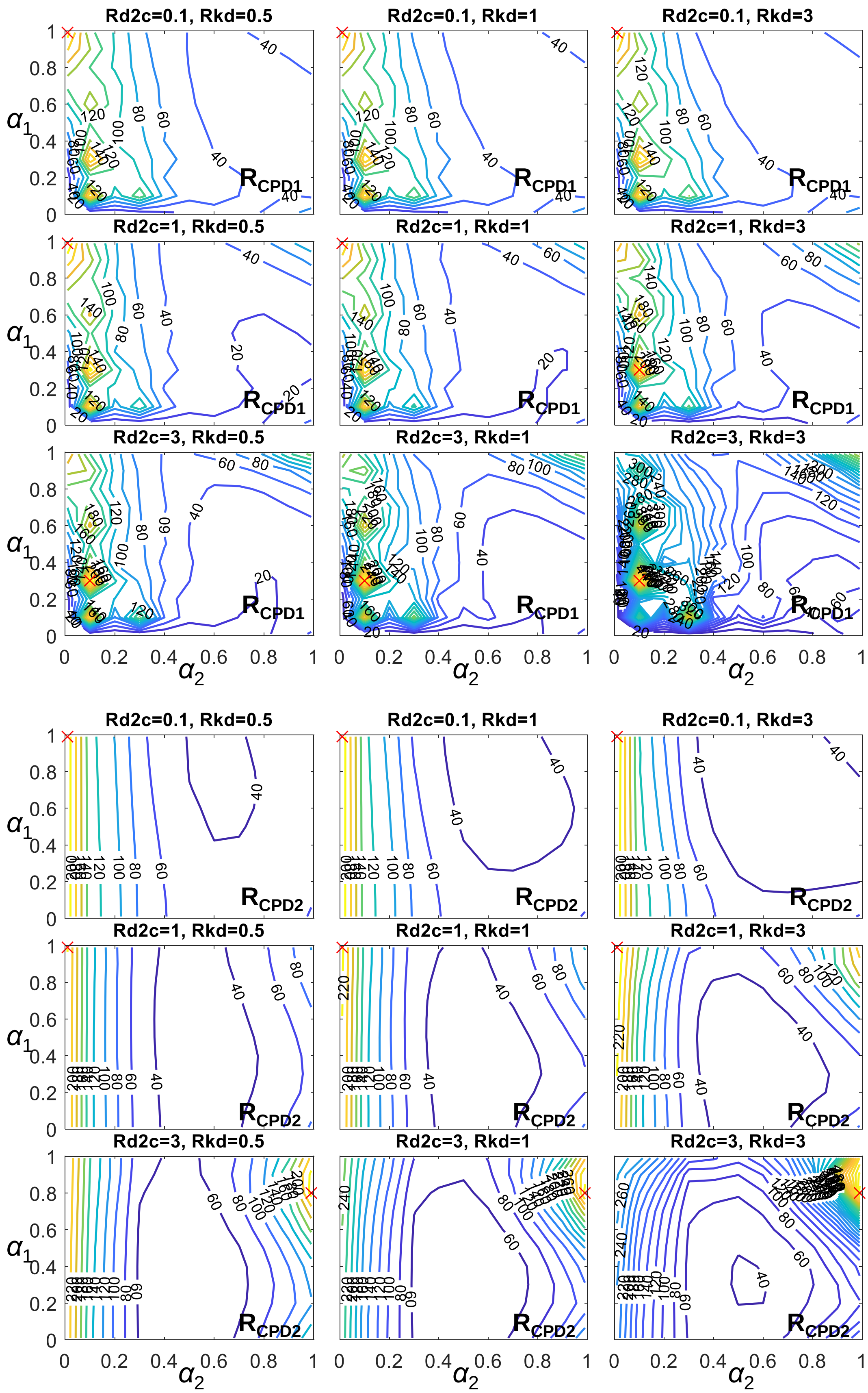


Figure 7.3.33 The CPD_1 and CPD_2 distributions with various outrigger elevations of the 16-story model

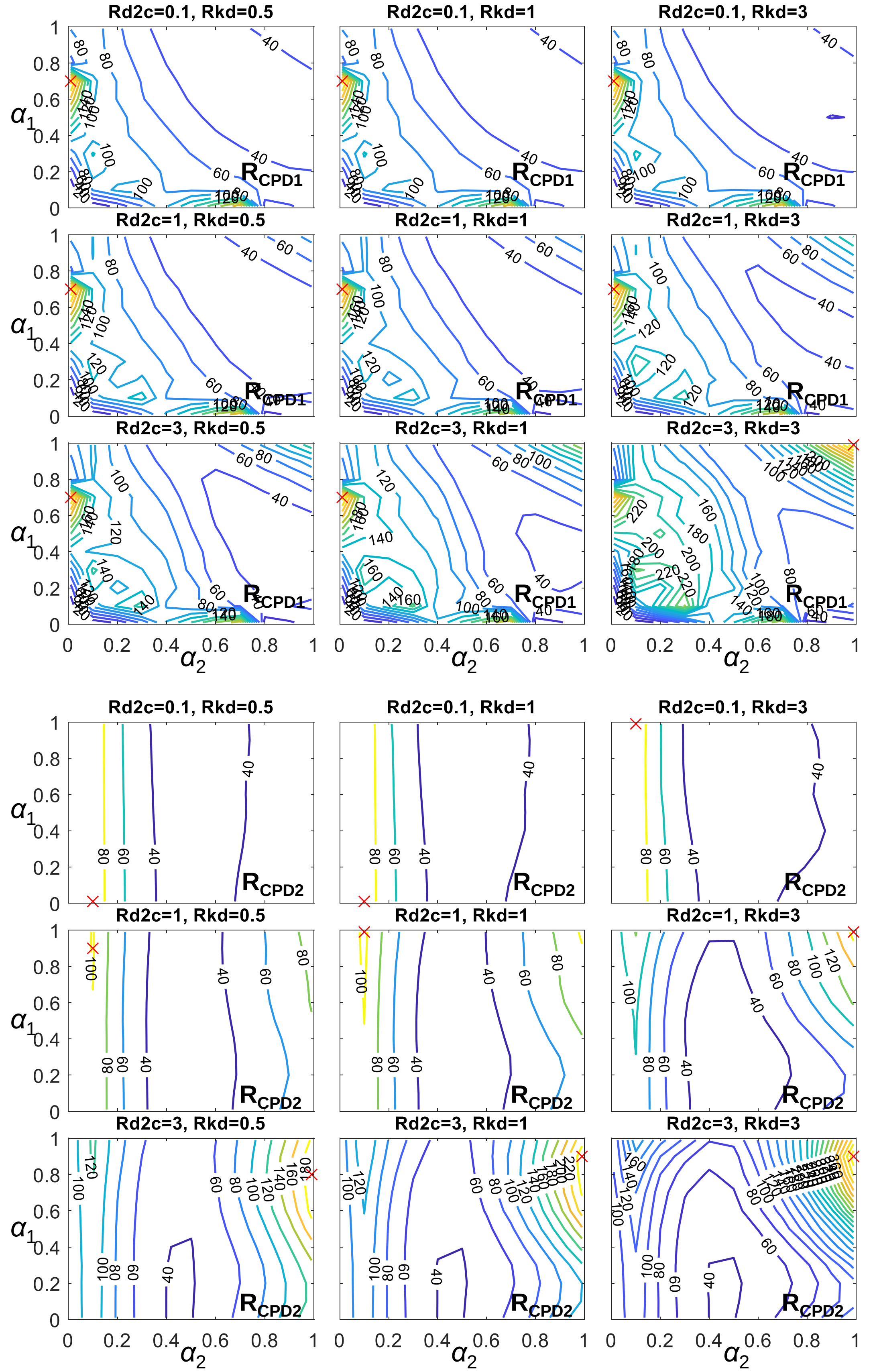


Figure 7.3.34 The CPD_1 and CPD_2 distributions with various outrigger elevations of the 32-story model

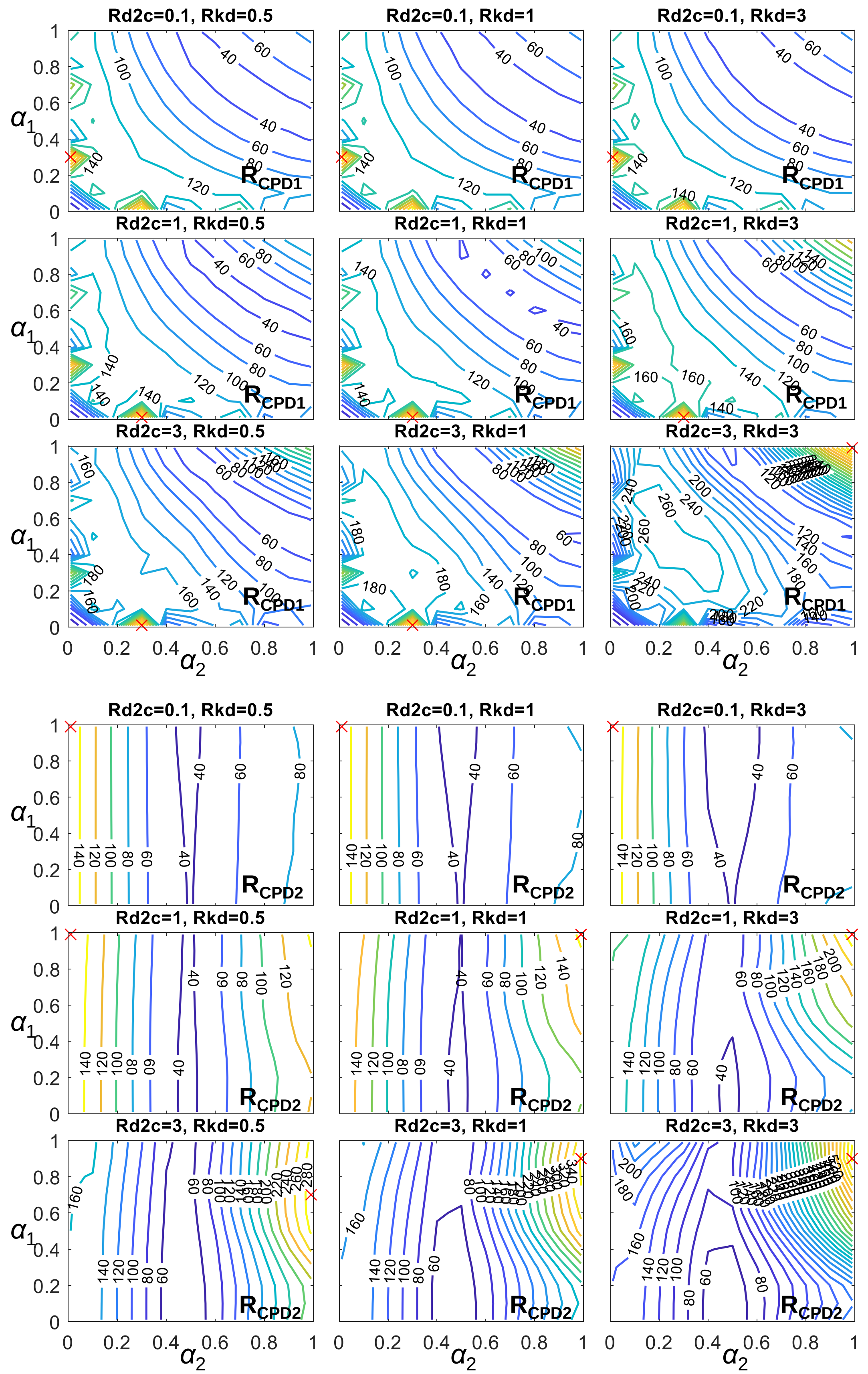


Figure 7.3.35 The CPD_1 and CPD_2 distributions with various outrigger elevations of the 64-story model

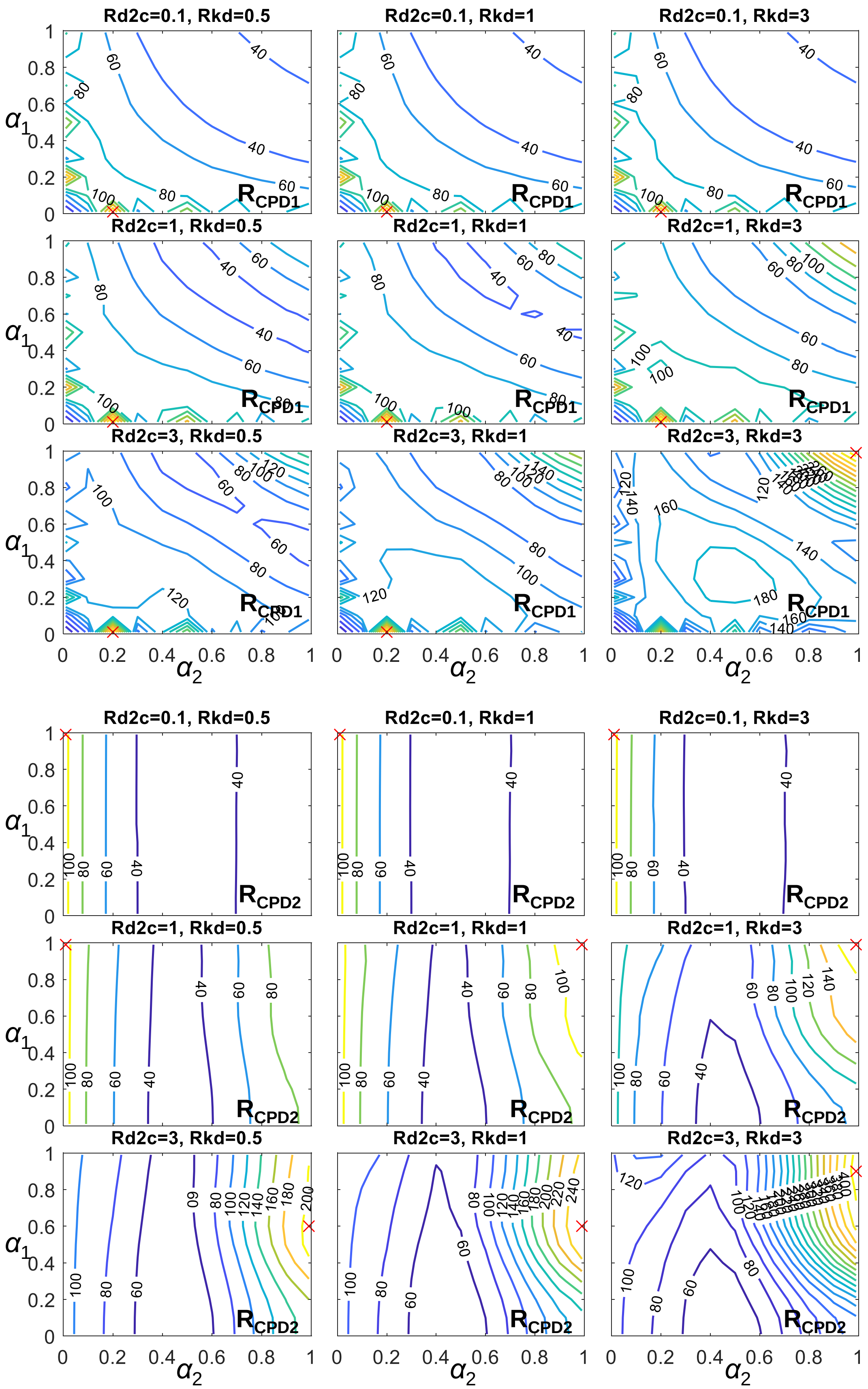


Figure 7.3.36 The CPD_1 and CPD_2 distributions with various outrigger elevations of the 96-story model

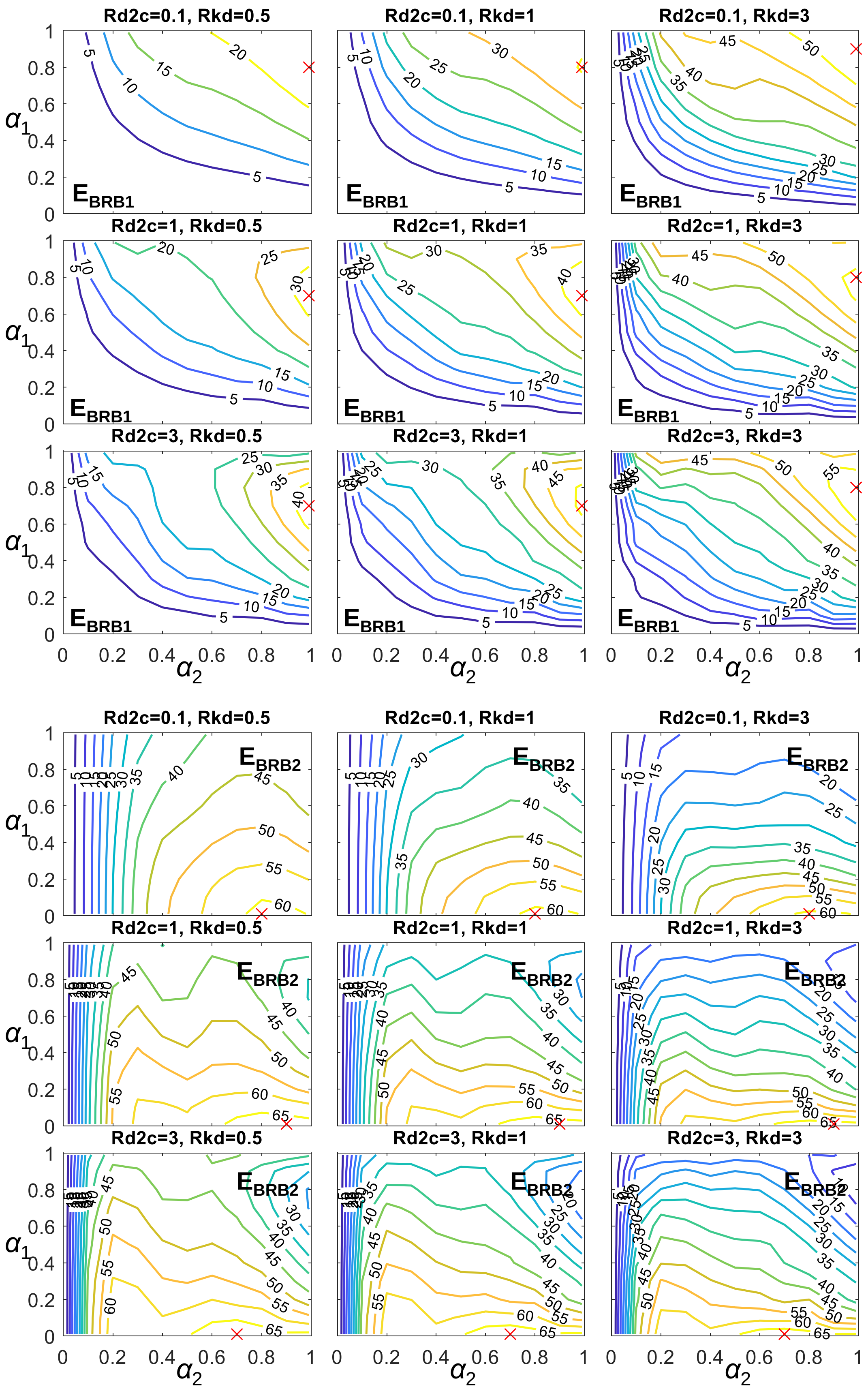


Figure 7.3.37 The ratio of energy dissipated by BRB_1 or BRB_2 to the total input energy with various outrigger elevations of the 16-story model (in percentage)

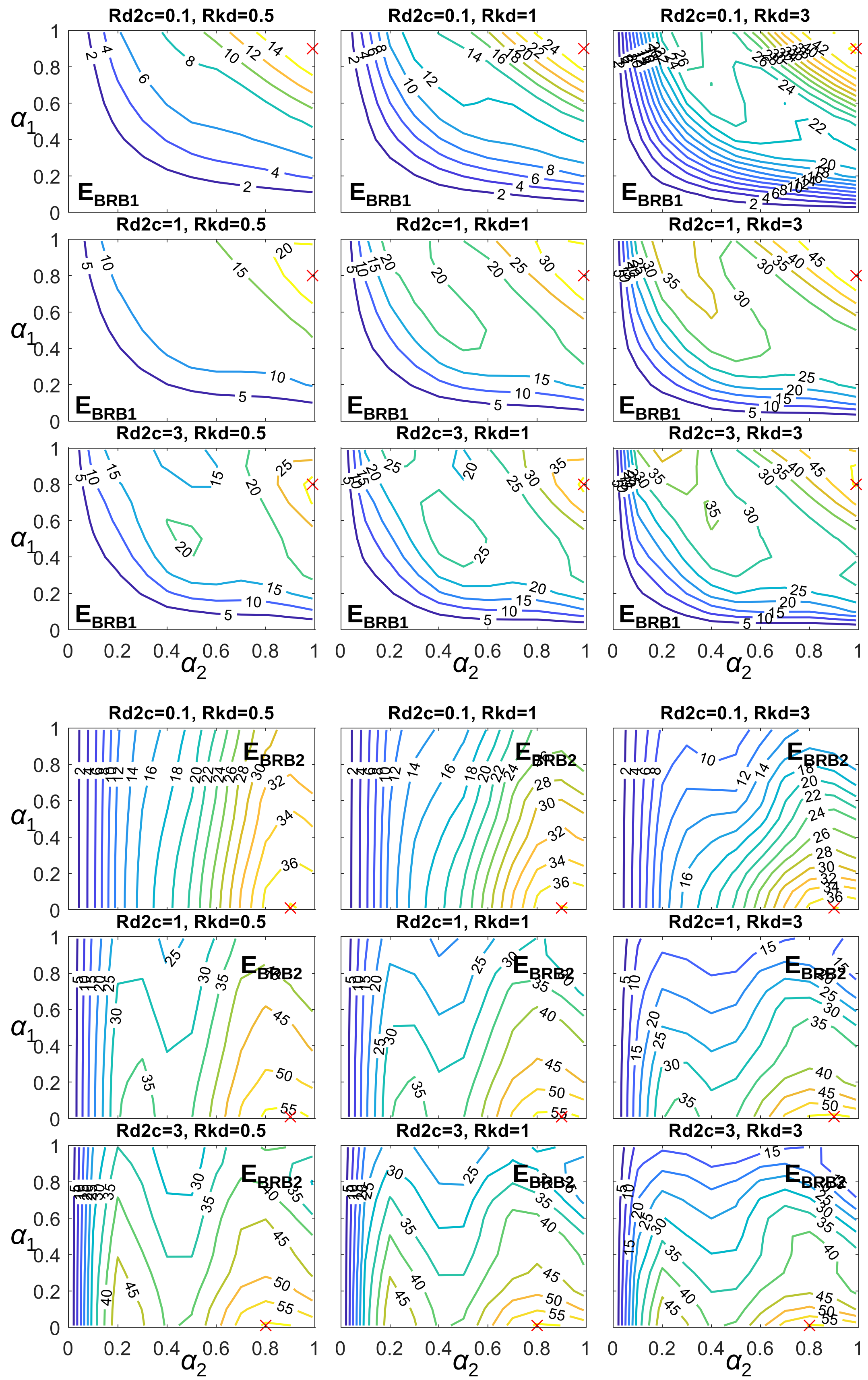


Figure 7.3.38 The ratio of energy dissipated by BRB_1 or BRB_2 to the total input energy with various outrigger elevations of the 32-story model (in percentage)

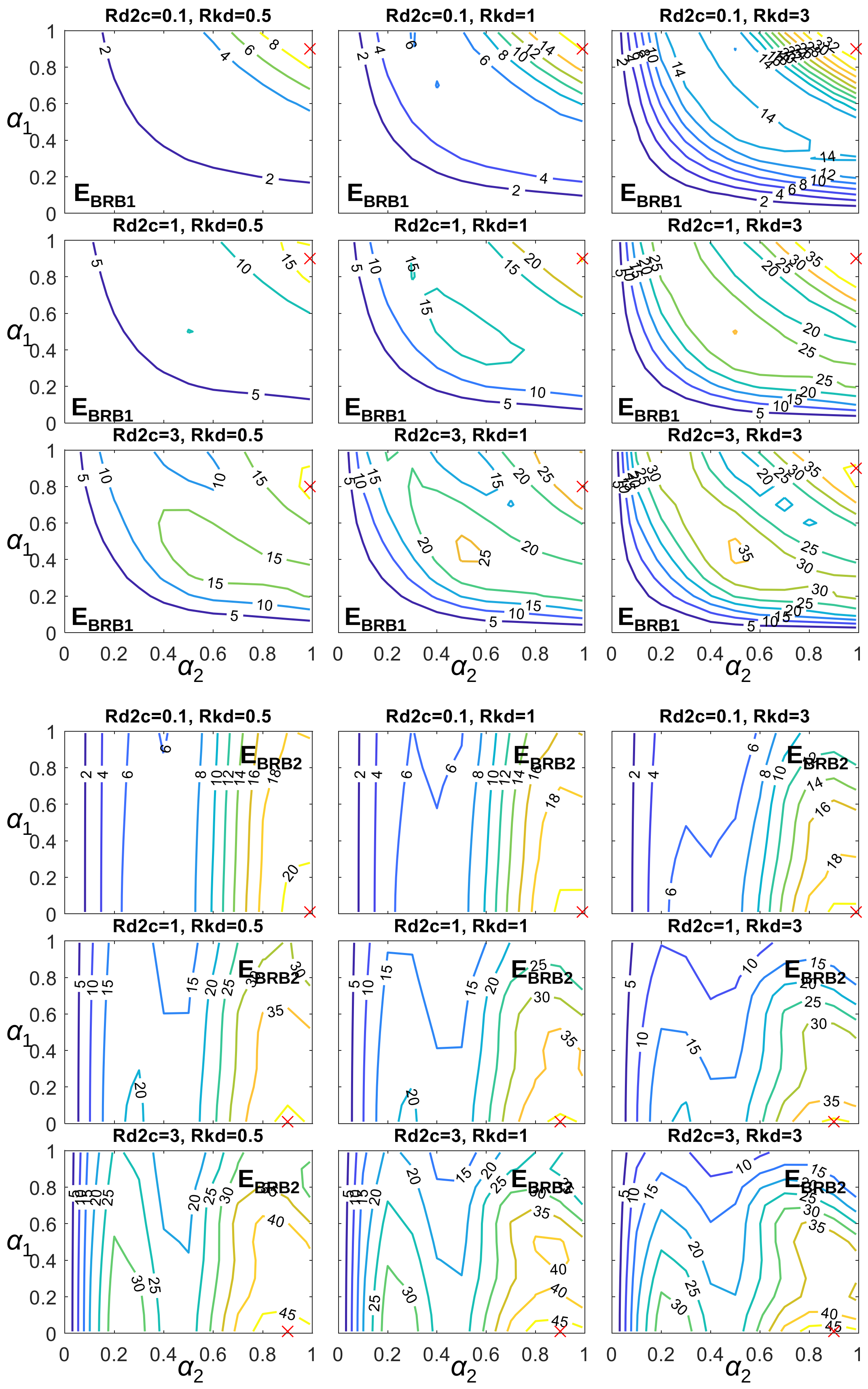


Figure 7.3.39 The ratio of energy dissipated by BRB_1 or BRB_2 to the total input energy with various outrigger elevations of the 64-story model (in percentage)

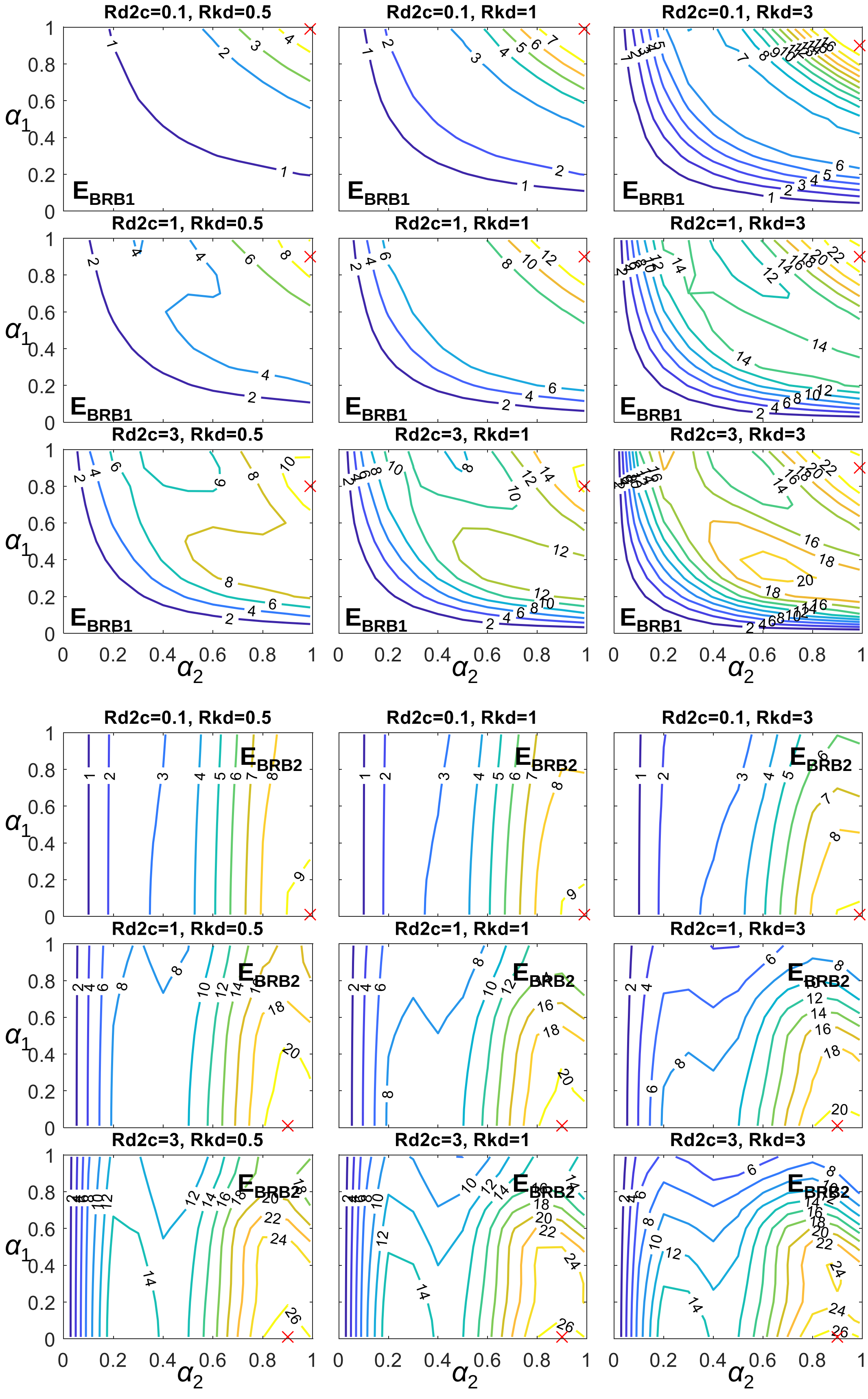


Figure 7.3.40 The ratio of energy dissipated by BRB₁ or BRB₂ to the total input energy with various outrigger elevations of the 96-story model (in percentage)

7.3.6 Summary of optimal design for dual BRB-outrigger system

Based on the analysis results, the optimal upper BRB-outrigger elevation (α_2) in order to mitigate $\theta_{\max}$ and $\gamma_{\max}$ is approximately 0.7 to 0.8. In addition, if α_2 is within its optimal range, the reduction in $M_{c,\max}$ is optimal when the lower BRB-outrigger (α_1) is close to the core structure base (0.4 to 0.6). The method of increasing R_{kd} is efficient in reducing $\theta_{\max}$ and $\gamma_{\max}$ when α_1 is approximately 0.6, and to reduce $M_{c,\max}$ when α_1 is approximately 0.4. The optimal lower outrigger elevations (α_1) are approximately 0.4 to 0.7 to minimize $\theta_{\max}$ and $\gamma_{\max}$, and approximately 0.2 to 0.6 to minimize $M_{c,\max}$. In addition, the optimal range of α_2 (between 0.7 and 0.9) is also the range that the E_{BRB2} could reach its maximum. When α_1 is within 0.4 to 0.7, both E_{BRB1} and E_{BRB2} could exhibit satisfactorily energy dissipation efficiency.

For the design practices, selecting the appropriate outrigger elevations should be the first priority as they affect the overall seismic performance the most. Selecting S_{bc2} as large as possible is also important because it determines the magnitude of the outrigger effect. The value of R_{d2c} should be limited between 0.5 and 1.5, as too large R_{d2c} value increases the cost of the BRB and the seismic response reduction is insignificant, furthermore, the $C_{1,\max}$ could excessively increase. If compare with the single BRB-outrigger system, the lower BRB-outrigger in the dual BRB-outrigger system further improves the seismic performance, and its optimal elevation depends on the design strategy. Placing the lower BRB-outrigger at the elevation where inter-story drift is too large could greatly mitigate the inter-story response. Based on the analysis results, the value of R_{kd} is recommended as 1.0, and it could be used to fine-tune the design as changing it between 1 and 3 only affects the seismic responses within 5% to 10%.

7.4 SUMMARY

This chapter presents the analysis results for the purpose of optimal design for structures with single and dual BRB-outrigger systems. The seismic responses including the maximum roof drift ($\theta_{\max}$), maximum inter-story drift ($\gamma_{\max}$), maximum overturning moment at core structure base ($M_{c,\max}$), maximum perimeter column axial force ($C_{1,\max}$), and the BRB energy dissipation efficiency are used as indicators to indicate the seismic performance. Based on the analysis results, the summaries of this chapter are as follows:

(1) Both the SA and NLRHA are performed on structures with either single or dual BRB-outrigger systems. The SA results generally well agree with the results calculated from NLRHA.

(2) Table 7.4.1 summarizes the parameters of α, R_{db} (Met. I), and R_{dc} (Met. II) when minimum or maximum of the seismic performance indicator is achieved for the single BRB-outrigger system. Based on the analysis results, the larger values of R_{db} and R_{dc} increase the reductions in $\theta_{\max}$ and $\gamma_{\max}$, however, the maximum roof acceleration ($a_{\max}$) and $C_{1,\max}$ can be amplified. For the single BRB-outrigger system, to select the α within the optimal range and increase the S_{bc07} value as large as possible are better strategies than increasing the value of R_{db} and R_{dc} in order to reduce seismic response.

Table 7.4.1 Optimal design parameters for single BRB-outrigger system

seismic response	α		R_{db} (Met. I)	R_{dc} (Met. II)
	Met. I	Met. II		
minimize $\theta_{\max}$	0.7~0.8	0.5~0.7	2~5	
minimize $\gamma_{\max}$	0.7~0.8		0~5	
minimize $M_{c,\max}$	0.7~0.8		0~5	
minimize $a_{\max}$	0.7~0.9	0.7	1~5	
maximum $C_{1,\max}$	0.5~0.8		>5	

(3) Table 7.4.2 summarizes the parameters of α_2, α_1, and R_{d2c} when minimum or maximum of the seismic performance indicator is achieved for the dual BRB-outrigger system. Based on the analysis results, when α_2 is lower than 0.4, the presence of the lower BRB-outrigger has very less contribution in reducing seismic response. Thus, the α_2 must have to be greater than 0.4 for dual BRB-outrigger system. In addition, the larger values of R_{d2c} and R_{kd} increase the reductions in seismic response, however, the increase in seismic response reductions becomes slow or stops when R_{d2c} is greater than 1.5. For the design purpose, the first priority is to select α_1 and α_2 in their optimal ranges and to keep the value of $S_{bc2,07}$ as large as possible. The effects due to changes of α_1 and R_{kd} on seismic response are small but can be used to fine-tune the design.

Table 7.4.2 Optimal design parameters for dual BRB-outrigger system

seismic response	α_2	α_1	R_{d2c}
minimize $\theta_{\max}$	0.7~0.8	0.4~0.7	0.5~1.5
minimize $\gamma_{\max}$	0.7~0.8	0.4~0.7	0.5~1.5
minimize $M_{c,\max}$	0.7~0.8	0.2~0.6	0.5~1.5
maximum $C_{1,\max}$	0.4~0.6	0	-
maximum E_{BRB}	0.7~0.9	0.4~0.7	-

(4) For both the single and dual BRB-outrigger systems, the optimal BRB parameters (R_{db}, R_{dc}, and R_{d2c}) in order to minimize seismic response is smaller when the outrigger effect is stronger. Figure 7.4.1 shows the relationships between optimal R_{db}, R_{dc} (single BRB-outrigger), R_{d2c} (dual BRB-outrigger) and S_{bc07} (single BRB-outrigger) and $S_{bc2,07}$ (dual BRB-outrigger). Based on the analysis results, both the single and dual BRB-outrigger systems can achieve similar seismic response when the BRB parameters are at their optimal values. The optimal R_{d2c} is almost half of the optimal R_{db} and R_{dc}. This suggests that if the BRB is too large to design in the single BRB-outrigger system, the dual BRB-outrigger system with smaller BRB axial stiffness requirements could be an alternative.

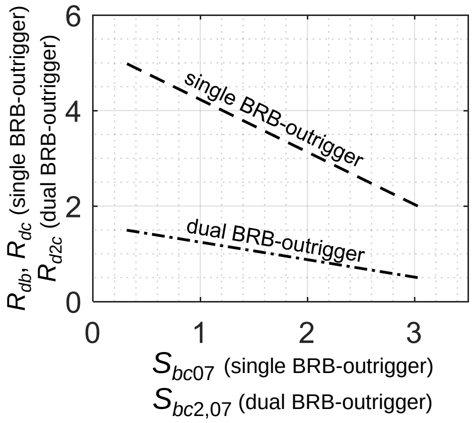


Figure 7.4.1 The optimal R_{db}, R_{dc}, and R_{d2c} values with respect to S_{bc07} and $S_{bc2,07}$

7.5 REFERENCES

ANSI/AISC 341-16 (2016) *Seismic Provisions for Structural Steel Buildings, American Institute of Steel Construction*. doi: 111.

8

DESIGN RECOMMENDATION AND DESIGN EXAMPLES

CHAPTER CONTENTS

8.1 Introduction 8-3
8.2 Optimal design of single BRB-outrigger system 8-3
8.2.1 Introduction of the design example..8-6
8.2.2 Analysis result of the design example 8-9
8.3 Optimal design of dual BRB-outrigger system..8-13
8.3.1 Introduction of the design example..8-16
8.3.2 Design charts..8-16
8.3.3 Analysis result of the design example 8-19
8.4 Summary 8-28

8.1 INTRODUCTION

This chapter presents the design recommendations for structures with single and dual BRB-outrigger systems by using step-to-step flow charts. Based on the analysis results presented in the previous chapters, if compare to the core structure, the dual BRB-outrigger system reduces the overall seismic performance of the single BRB-outrigger system by approximately 5% to 10%. Therefore, for an economical design, it is recommended to start the deisng with single BRB-outrigger configuration. If the siemsic response of a single BRB-outrigger system exceeds allowable limit, then it is suggested to proceed to the dual BRB-outrigger design. The detail of single and dual BRB-outrigger designs are presented in the following sections. A design example is used to demonstrate the optimal design.

8.2 OPTIMAL DESIGN OF SINGLE BRB-OUTRIGGER SYSTEM

The optimal values of α, R_{db}, and R_{dc} in order to minimize the seismic responses of a single BRB-outrigger system are shown in Table 7.4.1. Figure 8.2.1 shows the recommended design flow chart. For design practice, the building lateral stiffness (core structure flexural rigidity, EI) should be mainly determined based on the code specifications. The perimeter column sizes (k_c) should be determined according to the floor framing plan (tributary area) and gravity load demands. Therefore, the determinations of k_d and k_t values are less restrained. The smaller k_d (smaller values of R_{db} and R_{dc}) and k_t are desirable as they reduce material usage and cost. Based on the analytical results, the recommended design procedure is as follows,

(1) If α is not restricted for architectural reasons, select α between 0.6 and 0.8.

(2) Target the S_{bc} as large as possible (approximate 2 to 5), as the larger S_{bc} value could lead to smaller optimal R_{db} and R_{dc}.

(3) Compute the k_t according to the S_{bc} determined in the previous step. If k_t is too large to design the outrigger truss members, reduce k_t until the outrigger truss member sizes are reasonable, and update the S_{bc}.

(4) Select the Met. I approach if the perimeter column sizes are adjustable, otherwise, select Met. II. Target the optimal R_{db} (Met. I) or R_{dc} (Met. II) in the range of 1 to 5.

(5) Calculate the k_d according to the selected R_{db} or R_{dc} value.

(6) Design the BRB detail based on the k_d calculated in Step (5). Determine appropriate $u_{d,y}$ based on the actual BRB configuration and calculate the corresponding θ_r by performing 1st mode modal pushover analysis (MPA). Decrease k_t if θ_r is too large (when $\theta_r > 1/300$), or increase k_t if θ_r is too small (when $\theta_r < 1/800$), until the θ_r is within a reasonable range (1/300 ~ 1/800). At this stage, the $u_{d,y}$ can also be calculated by using the SRSS deformed shape when the roof drift reaches θ_r (Case4) as described in Chapter 6.

(7) After all the parameters are determined, perform the analysis and proceed to member design. As the outrigger effect results in additional force demands on the perimeter column, the perimeter column axial force demand should include the maximum BRB axial force capacity. If the BRB spans one story height as shown in Figure 3.5.2, the additional bending moment demand due to the BRB should be also considered. If the BRB spans more than one story height as shown in Figure 3.5.6, the strength of the floor beam below the BRB should be sufficiently strong to support the BRB's maximum axial force capacity.

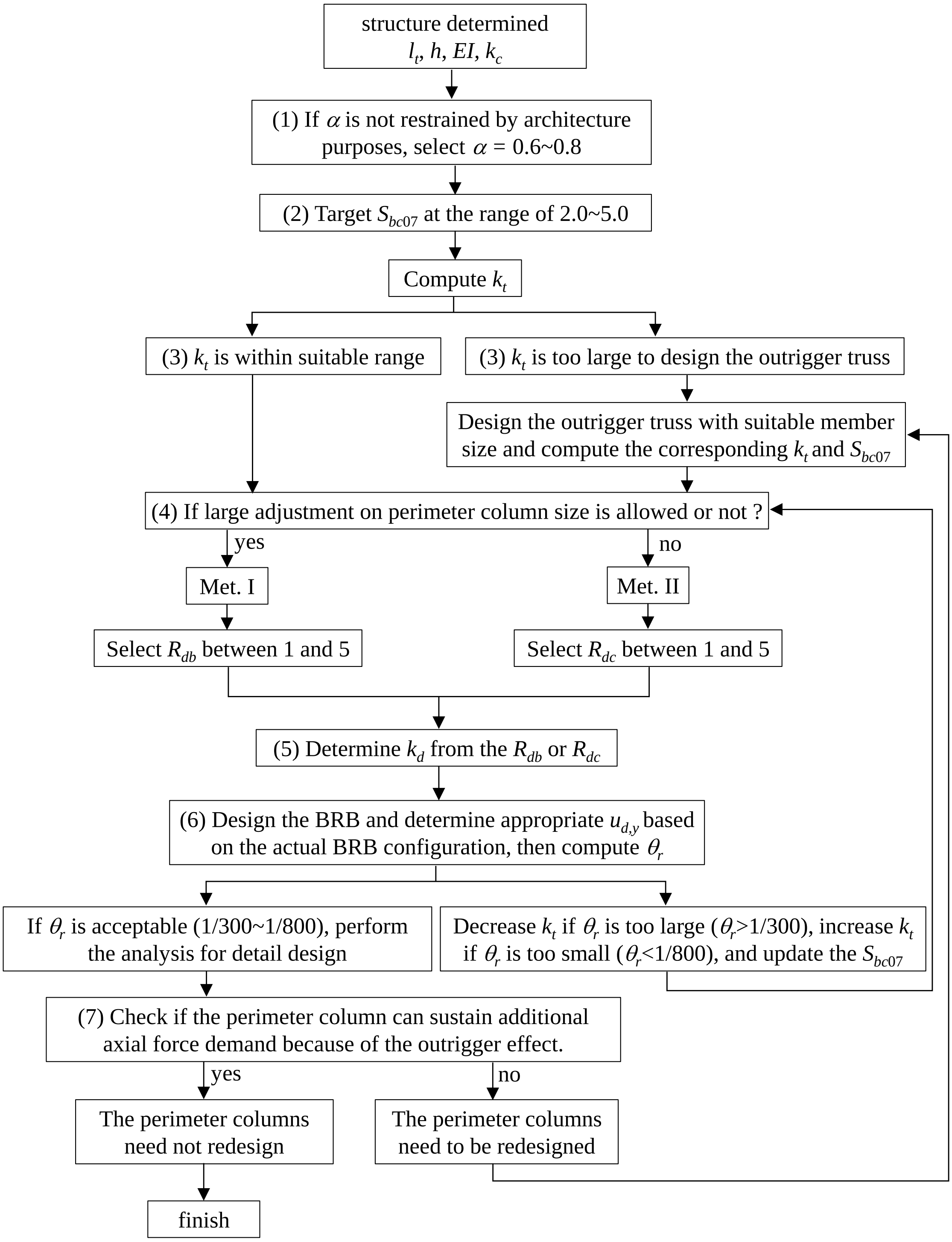


Figure 8.2.1 Flow chart of design recommendation for single BRB-outrigger system

8.2.1 Introduction of the design example

A 40 story (h = 160 m) building is used as the design example for the single BRB-outrigger system. Figure 8.2.2 shows the floor framing plan of the outrigger floor of the example. The two BRB-outrigger elevations are responsible to resist seismic force in the east-west direction. The dead and live loads are 0.8 and 0.3 tonf/m^2, respectively. As only half of the model is analyzed, the mass is 232 ton for each floor and 58 ton for each node in the DM model. The core structure flexural rigidity (EI) is 4×10^9 kN-m^2, so that the fundamental vibration period of the core structure is approximate 5.54 sec. The outrigger is arranged at α = 0.7, and the perimeter column size is Box 900×900×75 made by SN490 grade steel (material yield stress = 325 MPa). Therefore, the value of k_c is 309375 kN/m. Figure 8.2.3 shows the detail of the outrigger truss design. The member of the top chord, bottom chord, and the columns between top and bottom chords are BH 700×500×50×70 mm, and the size of the brace between top and bottom chords are BH 500×500×50×50 mm. All the connections are made by moment connection detail. In order to calculate the flexural stiffness of the outrigger truss (k_t), the analytical model of the outrigger truss is constructed by using OpenSees as shown in Figure 8.2.4. All the members are modeled using beam column elements. The translational and rotational degrees of freedom of Node 1 and Node 7 are restrained, while the others are free to translate and rotate. A unit force in the direction parallel to y-axis is applied at Node 6. The analysis result shows that the k_t is 187987 kN/m. The S_{bc} can be calculated as follows:

$$S_{bc} = \frac{l_t^2 h}{EI\left(\frac{1}{k_t} + \frac{0.7}{k_c}\right)} = \frac{12^2 \times 160}{4\times10^9\left(\frac{1}{187987} + \frac{0.7}{309375}\right)} = 0.76 \tag{8.2.1}$$

The Met. II is adopted for the design example. Based on the analysis results shown in Chapter 7, the optimal R_{dc} ranges from 1 to 5, two BRB designs considering the BRB parameter R_{dc} equals to 1 and 3 are considered. The models are known as Single1 and Single3 when R_{dc} equals to 1 and 3, respectively. The BRB is arranged vertically across one story height as shown in Figure 8.2.3. Figure 8.2.5 and Figure 8.2.6 show the detail of the BRBs with R_{dc} = 1 and R_{dc} = 3, respectively. The BRBs when R_{dc} = 1 and R_{dc} = 3 are known as BRB_Rdc1 and BRB_Rdc3. Table 8.2.1 summarizes the design of the BRBs. The weights of BRB include the weight of steel core and steel tube. For the Single1 model, the values of R_{dc} and R_{dt} are 0.97 and 1.60, respectively. For the Single3 model, the values of R_{dc} and R_{dt} are 3.21 and 5.28, respectively. The cruciform steel cores are made by SN490 grade steel plate (yield strength = 325 MPa). The steel castings for the BRBs in Single1 and Single3 models are made by Box 300×300×12 mm and Box 400×400×16 mm steel tube with infill mortar with a compressive strength of 8000 psi (55 MPa), respectively. The BRB yield force capacity (N_y) can be calculated from multiplying the cross-sectional area of the steel core by the material yield stress, and the yield deformation of the BRB can be calculated from dividing the N_y by BRB axial stiffness (k_d). The N_y of the BRBs in Single1 and Single3 models are 1690 and 4607 kN, respectively. The maximum axial force capacity of the BRB (N_{cu}) can be calculated as follows (ANSI/AISC 341-16, 2016):

$$N_{cu} = \beta \omega_h R_y N_y \tag{8.2.2}$$

Where β, ω_h, and R_y are the compression strength adjustment, strain hardening factors, and the ratio of the expected yield stress to the specified minimum yield stress, respectively. If β = 1.15, ω = 1.3, and R_y =1.2, the N_{cu} of the BRBs in the Single1 and Single3 models are 3032 and 8265 kN, respectively. Therefore, the maximum axial force demand for the perimeter column in the 1st story of Single3 model can be calculated as follow:

$$P_u = \left[1.2 \times \underset{(\text{dead load})}{0.8}\left(\text{tonf/m}^2\right) + 1.6 \times \underset{(\text{live load})}{0.3}\left(\text{tonf/m}^2\right)\right] \times \underset{(\text{tributary area})}{30}\left(\text{m}^2\right) \times 40 + N_{cu} = 25217 \text{ kN} \tag{8.2.3}$$

The compressive capacity of the perimeter column (P_n) can be calculated as follows (AISC (American Institute of Steel Construction), 2016):

$$\frac{L_c}{r} = \frac{4000}{338} = 11.8 < 4.71\sqrt{\frac{E}{F_y}} = 117 \tag{8.2.4}$$

$$F_e = \frac{\pi^2 E}{\left(\frac{L_c}{r}\right)^2} = \frac{\pi^2 \times 200}{\left(\frac{4000}{338}\right)^2} = 14.1 \text{ kN/mm}^2 \tag{8.2.5}$$

$$F_{cr} = \left(0.658^{\frac{F_y}{F_e}}\right) F_y = \left(0.658^{0.325/14.1}\right) 0.325 = 0.322 \text{ kN/mm}^2 \tag{8.2.6}$$

$$P_n = F_{cr} A_g = 0.322 \times 247500 = 79695 \text{ kN} \tag{8.2.7}$$

Where L_c (4000 mm) and r (338 mm) are the effective length and radius of gyration of the perimeter column in the 1st story, respectively. F_e and F_{cr} are elastic buckling and critical stress, respectively. A_g (247500 mm^2) is the gross cross-sectional area of the perimeter column. If considering a strength reduction factor, $\phi = 0.9$, the demand-to-capacity ratio (DCR) of the perimeter column is 0.35.

Table 8.2.1 Detail of BRB design for the Single1 and Single3 models

BRB	cross-sectional area (mm^2)			segment length (mm)			k_d (kN/m)	$u_{d,y}$ (mm)	BRB weight (*kgf*)
	core	transition	joint	core	transition	joint			
BRB_Rdc1	5200	6600	8000	2400	35	765	301480	5.6	589
BRB_Rdc2	14175	22275	30375	1800	90	1010	992226	4.6	1408

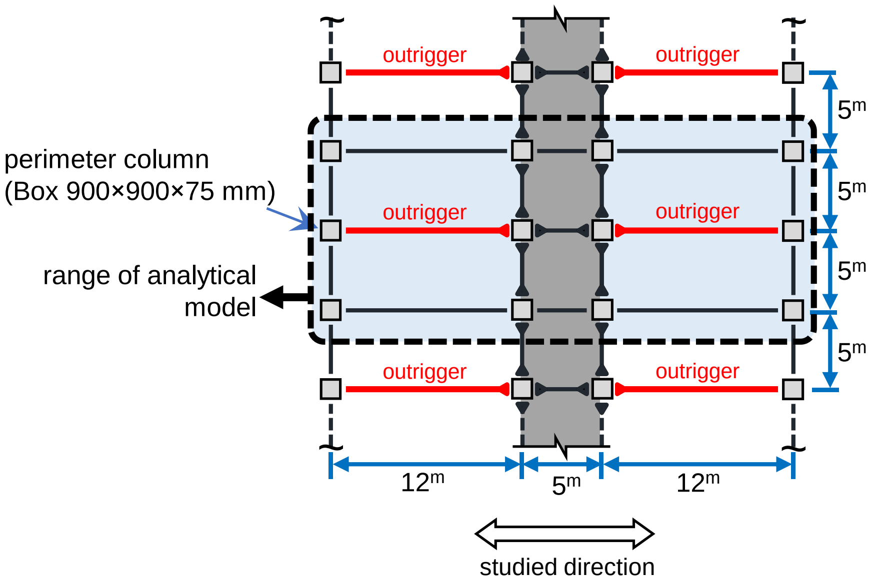


Figure 8.2.2 Floor framing plan of the outrigger elevation

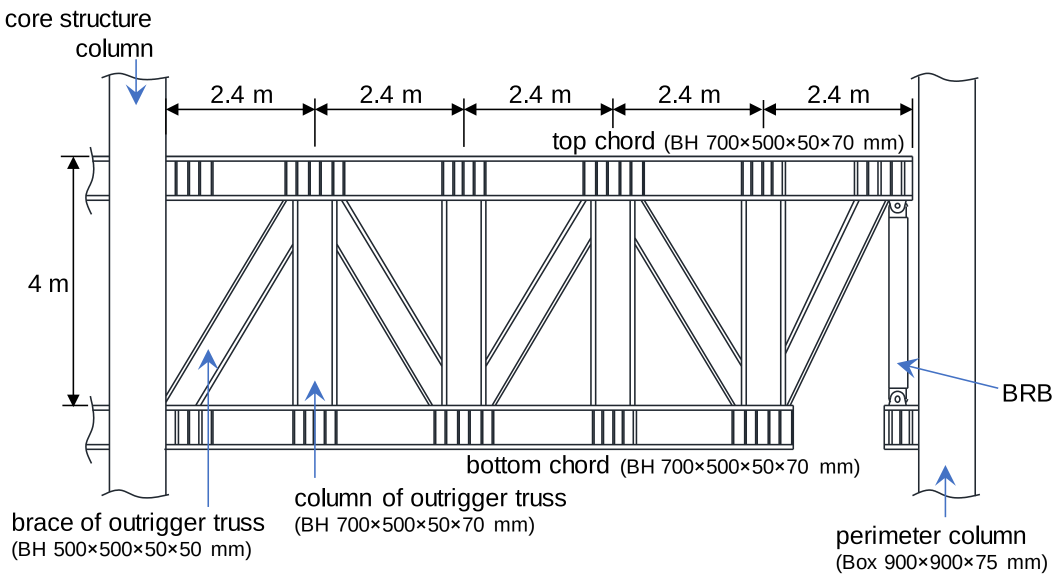


Figure 8.2.3 Detail of outrigger truss design

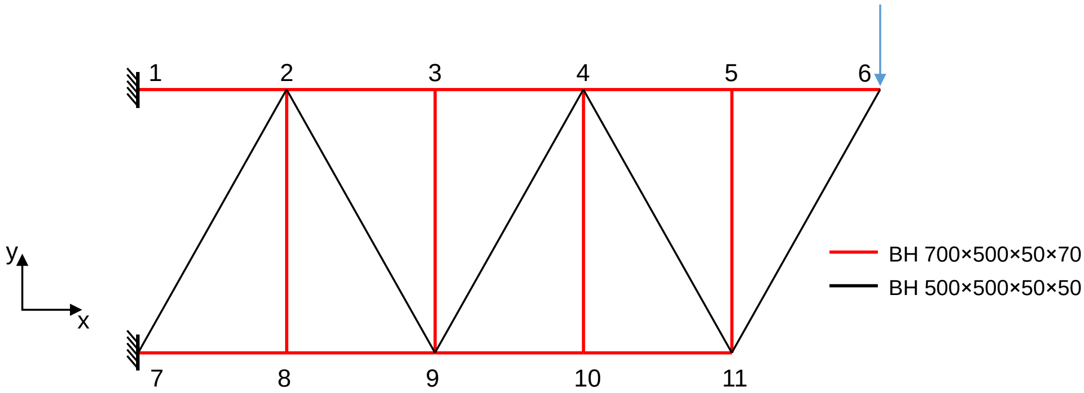


Figure 8.2.4 Analytical model of the outrigger truss

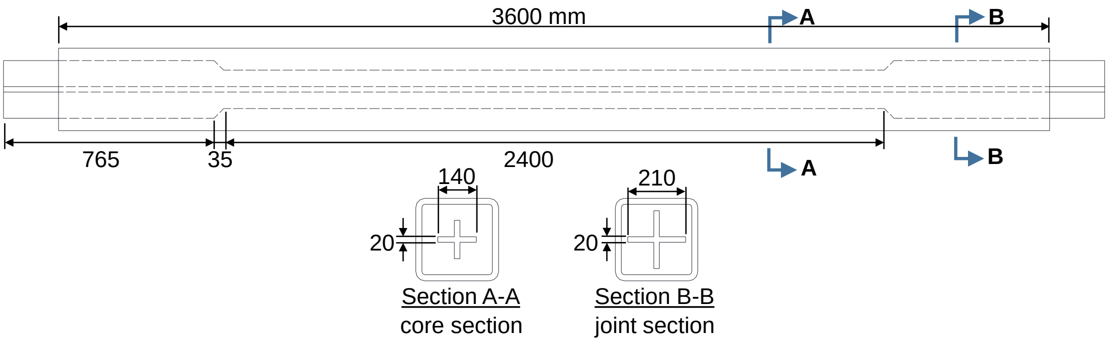


Figure 8.2.5 Detail of the BRB_Rdc1 design (R_{dc} = 1)

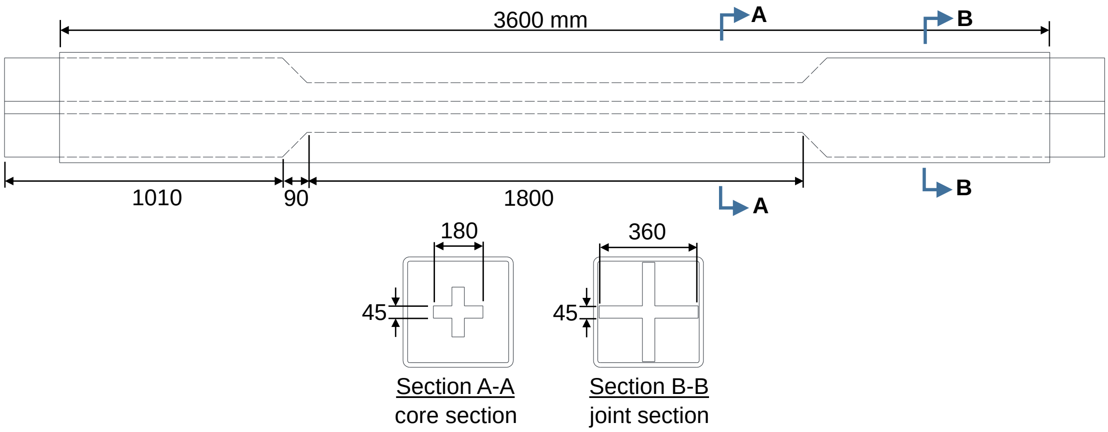


Figure 8.2.6 Detail of the BRB_Rdc3 design (R_{dc} = 3)

8.2.2 Analysis result of the design example

Table 8.2.2 shows the modal analysis results of the design example models and the core structure without outrigger effect (Core) by using the DM models. The drops of the vibration period show the outrigger effect stiffens the system. Figure 8.2.7 shows the 1st mode MPA curve and the relationship between the BRB axial deformation and roof drift. For the Single1 model, when the BRB axial deformation reaches the $u_{d,y}$ (5.6 mm), the corresponding roof drift is 0.131% rad. (θ_r = 1/763 rad.). For the Single3 model, the BRB yields ($u_{d,y}$ = 4.6 mm) when the roof drift reaches

0.280% rad. (θ_r = 1/357 rad.). As the θ_r is within an acceptable range, the SA and NLRHA are performed. Table 8.2.3 and Table 8.2.4 show the SA and NLRHA results, respectively. Figure 8.2.8 shows the maximum responses calculated from the NLRHA. The NLRHA is performed by using the originally observed ground motions. The numbers in Figure 8.2.8 indicate the ratios of the peak responses (θ_{max}, γ_{max}, $V_{c,max}$, and $M_{c,max}$) if compared with the Core model. The Single1 and Single3 models effectively reduce the seismic responses. In addition, the ground motions with larger seismic intensity generally result in greater reduction because of greater BRB energy dissipation. The Single3 with stiffer BRB only results in slightly smaller θ_{max} and γ_{max} than Single1 in most of the cases. However, the Single3 can amplify the $V_{c,max}$, $M_{c,max}$, and $C_{1,max}$ responses if compared with the Single1 model. The SA results (Table 8.2.3) suggest that the Single3 model (R_{dc} = 3) performs better in reducing θ_{max} and γ_{max} than the Single1 (R_{dc} = 1). The NLRHA results (Table 8.2.4) also indicate that the Single3 generally performs better than Single1 in mitigating θ_{max} and γ_{max}. As the Single3 model has greater θ_r, the R_{CPD} of the Single3 model are smaller than Single1 model. In addition, the BRBs of Single3 model stay almost elastic deformation and exhibit very low energy dissipation efficiency under the Tohoku, El Centro, Taft, and Kumamoto ground motions. During small earthquakes, the Single3 model tends to stay elastic deformation. Therefore, the $V_{c,max}$ and $M_{c,max}$ responses are greater than the Single1 model. The $C_{1,max}$ responses indicate the large value of R_{dc} significantly increase the perimeter column axial force demands. As indicated in Table 8.2.1, the weight of BRB in Single3 model is approximately 3 times the weight of BRB in Single1 model. However, the seismic responses Single3 do not significantly better than Single1. Thus, the Single1 (R_{dc} = 1) model should be an economical solution.

Table 8.2.2 modal analysis result of the design example

model	vibration period (sec)				mass participation ratio (%)			
	1st mode	2nd mode	3rd mode	4th mode	1st mode	2nd mode	3rd mode	4th mode
Core	5.543	0.884	0.316	0.161	68.2	20.9	7.2	3.7
Single1	4.483	0.845	0.316	0.160	70.7	18.4	7.2	3.7
Single3	4.316	0.836	0.316	0.160	71.3	17.8	7.2	3.7

Table 8.2.3 SA results of the Core, Single1, and Single3 models

	Core	Single1	Single3
θ_{max} (% rad.)	1.424	0.908	0.876
γ_{max} (% rad.)	2.00	1.20	1.16
h_{eq}	0.02	0.063	0.059

Table 8.2.4 NLRHA results of the Core, Single1, and Single3 models

	model	1	2	3	4	5	6	7	8
θ_{max} (% rad.)	Core	0.274	0.248	0.264	0.299	0.366	0.385	1.380	1.461
	Single1	0.244	0.211	0.177	0.265	0.355	0.356	1.174	0.632
	Single3	0.279	0.234	0.147	0.250	0.356	0.346	1.041	0.587
γ_{max} (% rad.)	Core	0.718	0.532	0.372	0.679	1.210	1.077	2.061	2.138
	Single1	0.598	0.494	0.275	0.573	1.110	0.897	1.738	1.040
	Single3	0.641	0.495	0.310	0.566	1.076	0.816	1.507	1.001
$V_{c,max}$ ($\times 10^4$kN)	Core	2.07	1.48	0.79	2.01	5.09	4.32	3.23	2.04
	Single1	1.38	1.38	0.88	1.77	4.47	3.35	3.03	1.83
	Single3	1.44	1.44	0.92	1.86	3.83	2.93	2.60	2.11
$M_{c,max}$ ($\times 10^6$kN-m)	Core	0.70	0.46	0.26	0.60	1.36	1.28	1.62	1.43
	Single1	0.43	0.40	0.29	0.46	1.13	0.95	1.30	0.77
	Single3	0.50	0.45	0.32	0.54	0.99	0.93	1.02	0.90
E_d/E_s (%)	Core	0	0	0	0	0	0	0	0
	Single1	22	17	12	16	20	23	59	40
	Single3	7	3	0	3	20	12	68	34
h_{eq}	Core	0.020	0.020	0.020	0.020	0.020	0.020	0.020	0.020
	Single1	0.037	0.033	0.030	0.033	0.036	0.039	0.067	0.052
	Single3	0.026	0.022	0.020	0.022	0.036	0.030	0.074	0.047
R_{CPD}	Core	0	0	0	0	0	0	0	0
	Single1	29	22	6	20	115	192	241	313
	Single3	4	2	0	1	55	43	110	114
$C_{1,max}$ ($\times 10^3$kN)	Core	0	0	0	0	0	0	0	0
	Single1	1.82	1.81	1.73	1.79	1.93	1.88	2.21	1.96
	Single3	4.79	4.69	3.23	4.65	5.15	4.92	5.78	5.21

ground motion: 1=Tohoku, 2=El Centro, 3=Taft, 4=Kumamoto, 5= KobeJMA, 6=Sendai, 7=ChiChi, 8=BCJL2

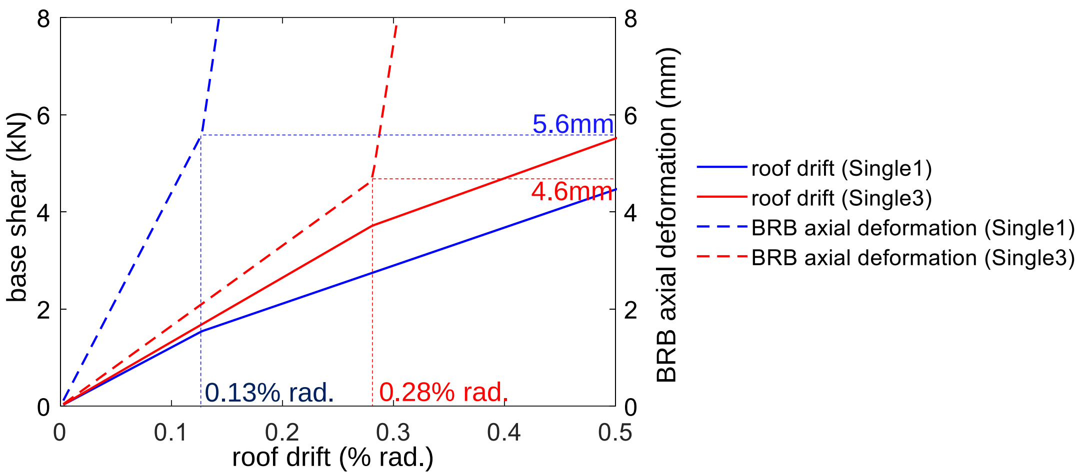
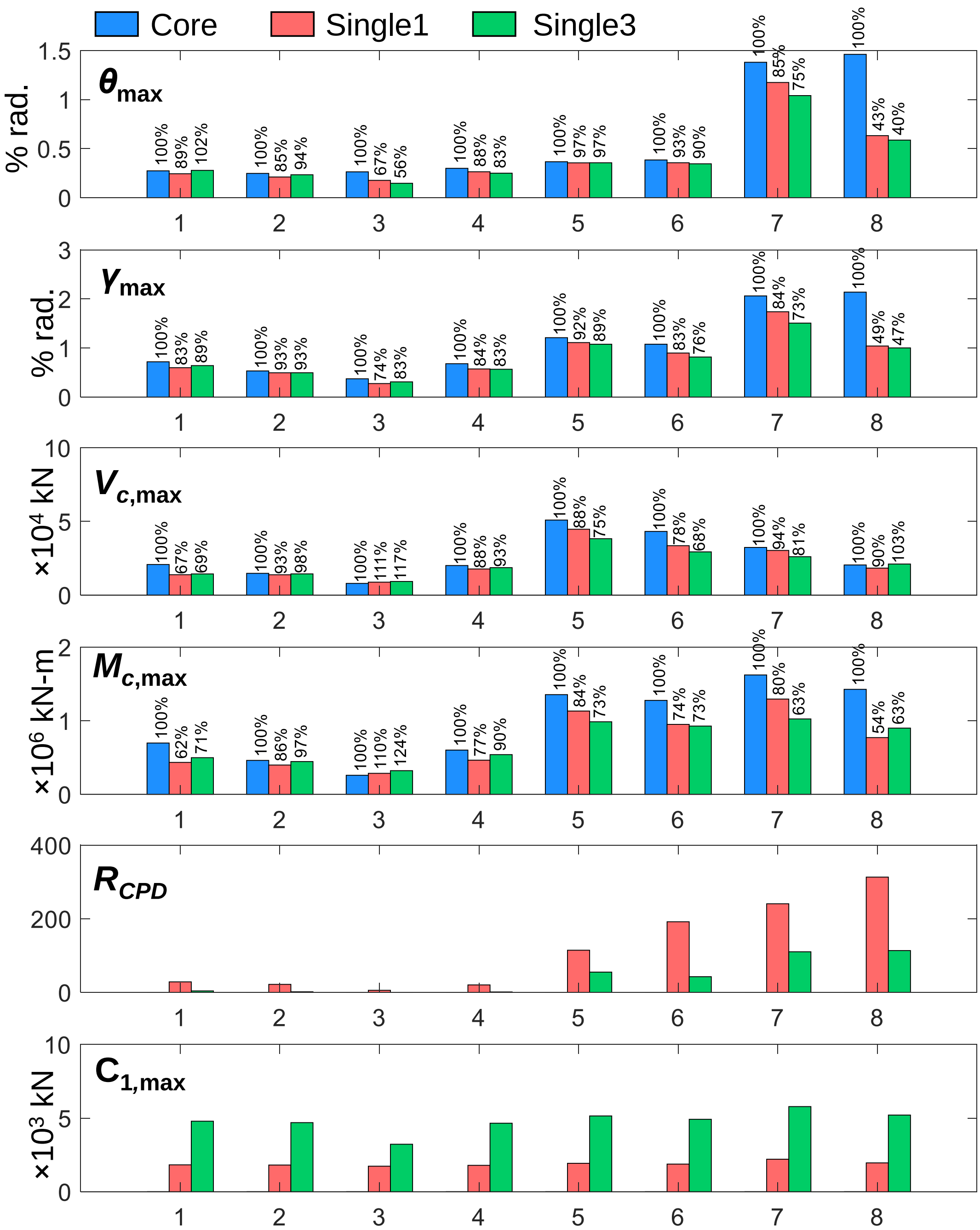


Figure 8.2.7 1st MPA curve and the relationship between BRB axial deformation and roof drift of the single BRB-outrigger example model



ground motion: 1=Tohoku, 2=El Centro, 3=Taft, 4=Kumamoto, 5= KobeJMA, 6=Sendai, 7=ChiChi, 8=BCJL2

Figure 8.2.8 Maximum responses of NLRHA results of the Core, Single1, and Single3 models

8.3 OPTIMAL DESIGN OF DUAL BRB-OUTRIGGER SYSTEM

The recommended design flow chart for the dual BRB-outrigger system is shown in Figure 8.3.1. For the design practice, h and l_t are fixed after the architectural plan and elevation have been determined, and the perimeter column size (k_c) and EI can be first designed based on the gravity load and seismic demands. For an economical design, it is recommended to use the single BRB-outrigger design. If the seismic response exceeds the allowable limits, or the required member sizes are too large to design with the single BRB-outrigger solution, proceed to the dual BRB-outrigger design. The recommended design procedure is as follows:

(1) If the upper outrigger elevation can be changed, place the upper outrigger at α_2 approximate 0.7 to 0.8. Place an additional lower BRB-outrigger based on the following conditions:

   A. If $\theta_{\max}$ is too large, place lower BRB-outrigger with $R_{kd} = 1$ at $\alpha_1 = 0.4 \sim 0.7$.

   B. If $M_{c,\max}$ is too large, place lower BRB-outrigger with $R_{kd} = 1$ at $\alpha_1 = 0.4 \sim 0.7$.

   C. Place lower BRB-outrigger with $R_{kd} = 1$ at the elevation where inter-story drift is too large.

   D. If the required BRB axial stiffness and outrigger truss flexural stiffness are too large to design the members, place the lower BRB-outrigger at $\alpha_1 = 0.4 \sim 0.7$, decrease the value of R_{d2c} to be smaller than 1, and increase the value of R_{kd} to be greater than 1.

(2) Increase the $S_{bc2,07}$ value as large as possible by varying α_2, k_c, and k_t values and meanwhile keeping α_2, k_c, and k_t within acceptable ranges.

(3) Select the value of R_{d2c} to be close to 1, and then calculate k_{d2}.

(4) Design the detail, including the yielding deformations of the BRBs in the lower (BRB_1) and upper outriggers (BRB_2), based on k_{d2} and R_{kd} obtained from the previous steps. Calculate the maximum allowable elastic roof drift limit (θ_r) by performing the 1st mode MPA. If θ_r is too small (<1/800), increase the $u_{d,y}$. If θ_r is too large (>1/300), decrease the $u_{d,y}$. Then, redesign the BRBs until θ_r is within a suitable range.

(5) Estimate the maximum seismic response by performing the SA or NLRHA. If the maximum seismic response exceeds the allowable limits, proceed to either step (6.1) or steep (6.2). Otherwise, proceed to step (7).

(6.1) If the perimeter column size can be changed, increase the value of $S_{bc2,07}$ by increasing k_c. Update the $S_{bc2,07}$ value and repeat from step (2) with the updated k_c.

(6.2) If the perimeter column size cannot be changed, increase the value of R_{kd} to be greater than 1, and repeat from step (4) to redesign the BRB_1.

(7) If the seismic response is within allowable limits, check the maximum axial force demand for the perimeter column. If the demand-to-capacity ratio (DCR) of the perimeter column is greater than 1.0, increase the size of the perimeter column. Update the $S_{bc2,07}$ value and repeat from step (2) to redesign BRB_1 and BRB_2 with the updated k_c. If the DCR of the perimeter column is smaller than 1.0, then the design is finished. If the BRB spans one story height as shown in Figure 3.5.2, the additional bending moment demand due to the BRB should be also considered. If the BRB spans more than one story height as shown in Figure 3.5.6, the strength of the floor beam below the BRB should be sufficiently strong to support the BRB's maximum axial force capacity.

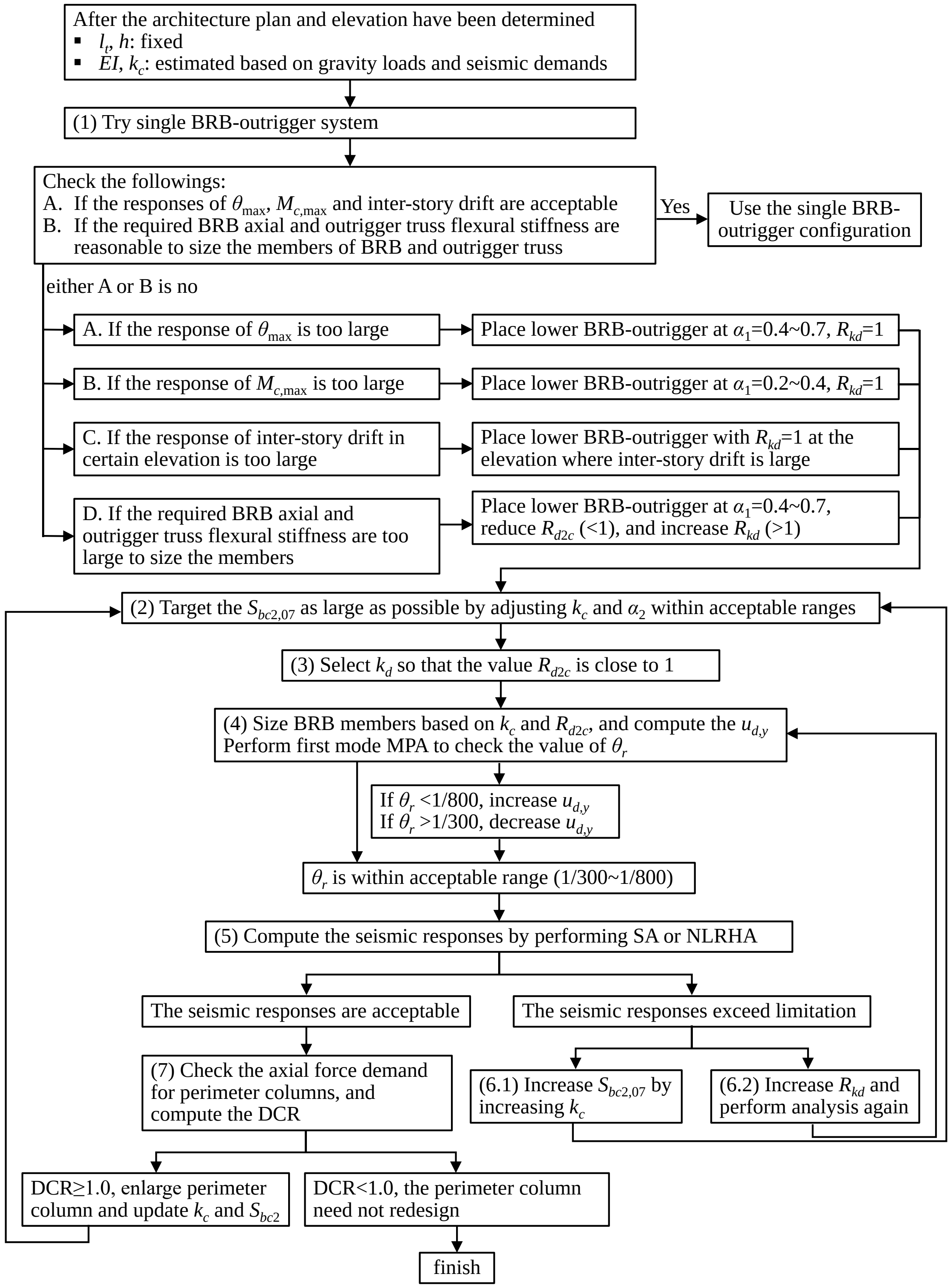


Figure 8.3.1 Flow chart of design recommendation for dual BRB-outrigger system

8.3.1 Introduction of the design example

The 40-story model used in the single BRB-outrigger example is also used to illustrate the design of dual BRB-outrigger system. It is assumed that the seismic performance of single BRB-outrigger design presented in the previous section does not satisfy allowable limits, the design of dual BRB-outrigger system then proceeds. The two BRB designs (BRB_Rdc1 and BRB_Rdc3) shown in Table 8.2.1 are used. A total of 6 configurations of dual BRB-outrigger system are considered, and the details are shown in Table 8.3.1. As the optimal upper outrigger elevation is approximately 0.7 to 0.8, the models with both α_2 = 0.7 and α_2 = 0.8 are considered. As the Dual0731 and Dual0831 model adopt BRB_Rdc3 for both the upper and lower BRB-outriggers, the axial force demand for the perimeter column increases. The axial force demand is calculated as follows:

$$P_u = \left[1.2\times \underset{\text{(dead load)}}{0.8}\left(\text{tonf/m}^2\right)+1.6\times \underset{\text{(live load)}}{0.3}\left(\text{tonf/m}^2\right)\right]\times \underset{\text{(tributary area)}}{30}\left(\text{m}^2\right)\times 40+2N_{cu}=33482\text{ kN} \quad (8.3.1)$$

The compressive strength DCR of the perimeter column (Box 900×900×75) is 0.47.

Table 8.3.1 Configurations of dual BRB-outrigger system design example

model	Dual0711	Dual0713	Dual0731	Dual0811	Dual0813	Dual0831
α_2	0.7	0.7	0.7	0.8	0.8	0.8
R_{d2c}	1	1	3	1	1	3
R_{kd}	1	3	1	1	3	1

8.3.2 Design charts

The design recommendation and flow chart shown in Figure 8.3.1 assists engineers by providing a clear procedure to design the dual BRB-outrigger system. In addition, the design charts, which are based on the analysis results presented in Chapter 7, provide engineers with a quick and efficient way to select the lower BRB-outrigger elevation (α_1) in order to achieve the desired seismic response. Figure 8.3.2 to Figure 8.3.7 show the design charts for the 6 configurations. Each design chart shows the $\theta_{\max}$, $\gamma_{\max}$, and $M_{c,\max}$ reduction factor distributions with respect to S_{bc2} and α_1. Table 8.3.2 shows reduction factors of the seismic response if compared with the single BRB-outrigger system, and the corresponding α_1 for each configuration. The dual BRB-outrigger system generally performs better than the single BRB-outrigger by reducing seismic response by 5% to 10%. The $\theta_{\max}$ can be best reduced when α_1 is approximate 0.6 to 0.8. The $\gamma_{\max}$ can also be best reduced when α_1 is between 0.6 and

0.8. The $M_{c,\max}$ can be best reduced when α_1 is approximate between 0.3 and 0.8, in general. In addition, the models with larger R_{kd} values achieve greater reductions in seismic response. Based on the design charts and the summary shown in Table 8.3.2, the additional lower outrigger could be placed at α_1 = 0.7 for reducing the $\theta_{\max}$ and $\gamma_{\max}$, and can be placed at α_1 = 0.3~0.4 for reducing $M_{c,\max}$. The α_1 equals to 0.3 and 0.7 for the 6 configurations are further analyzed. The identification of the model is added by "U" for the case when α_1 = 0.7, and is added by "L" for the case when α_1 = 0.3. The analysis results are shown in the next section.

Table 8.3.2 Reductions of seismic response if compared with single BRB-outrigger system and the corresponding α_1

	$\theta_{\max}$		$\gamma_{\max}$		$M_{c,\max}$	
	RF*	α_1	RF	α_1	RF	α_1
Dual0711	-6%	0.8	-6%	1.0	-5%	0.8
Dual0713	-10%	0.8	-11%	1.0	-8%	0.3
Dual0731	-4%	0.6	-4%	1.0	-8%	0.4
Dual0811	-6%	0.7	-6%	1.0	-5%	0.7
Dual0813	-11%	0.7	-10%	1.0	-11%	0.4
Dual0831	-6%	0.6	-3%	0.9	-9%	0.3

*RF = reduction factor

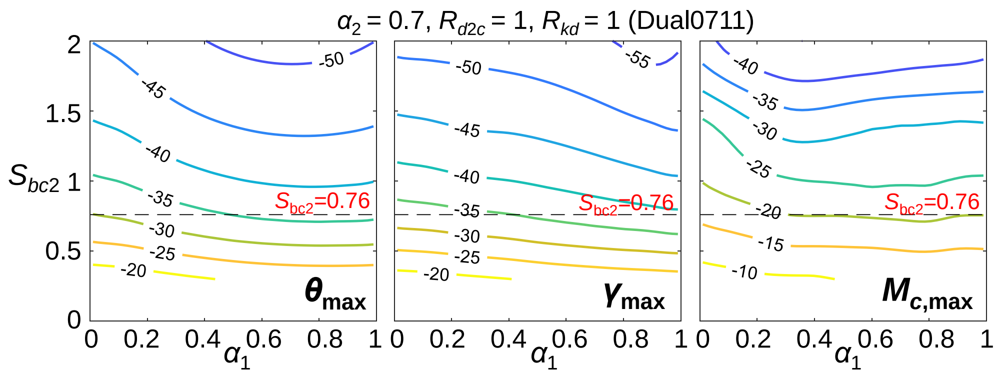


Figure 8.3.2 Reduction factors when α_2=0.7, R_{d2c}=1, and R_{kd}=1 (Dual0711)

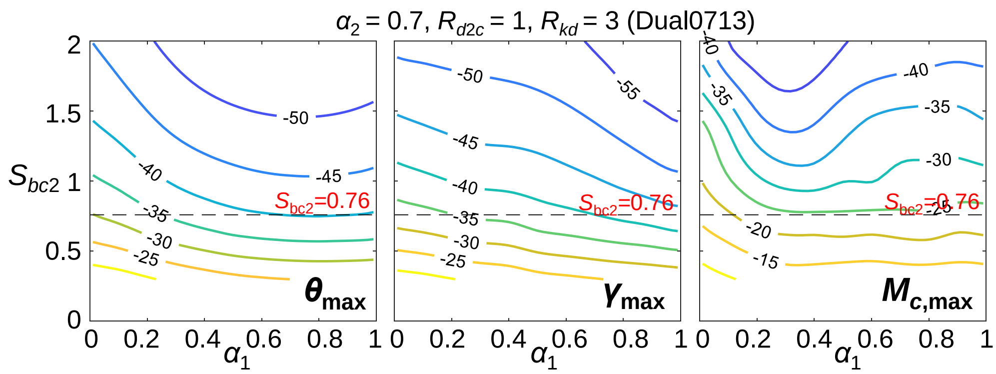


Figure 8.3.3 Reduction factors when α_2=0.7, R_{d2c}=1, and R_{kd}=3 (Dual0713)

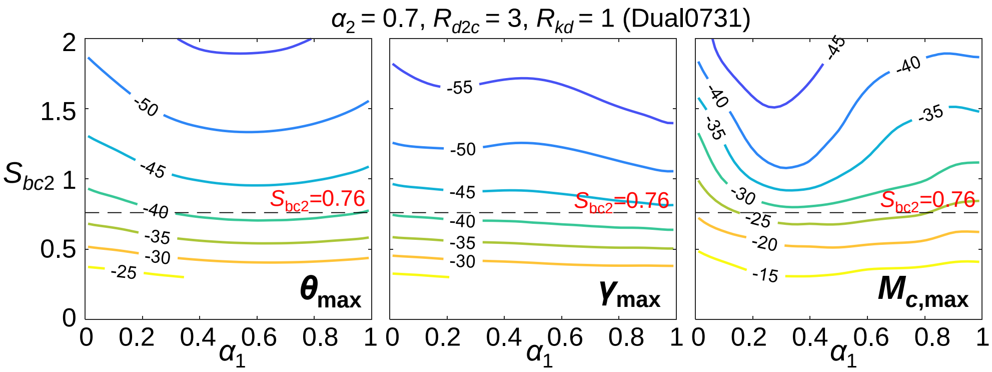


Figure 8.3.4 Reduction factors when α_2=0.7, R_{d2c}=3, and R_{kd}=1 (Dual0731)

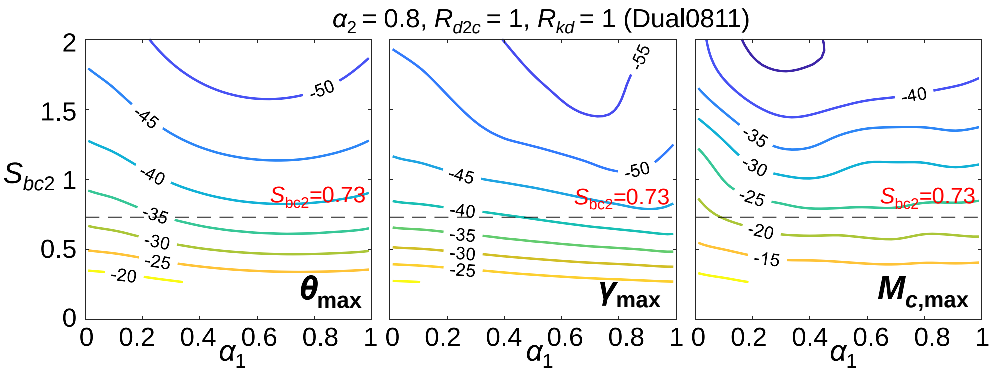


Figure 8.3.5 Reduction factors when α_2=0.8, R_{d2c}=1, and R_{kd}=1 (Dual0811)

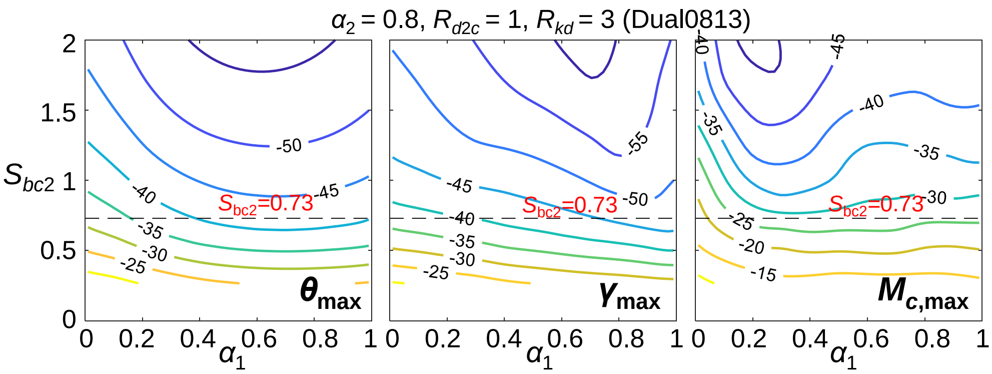


Figure 8.3.6 Reduction factors when α_2=0.8, R_{d2c}=1, and R_{kd}=3 (Dual0813)

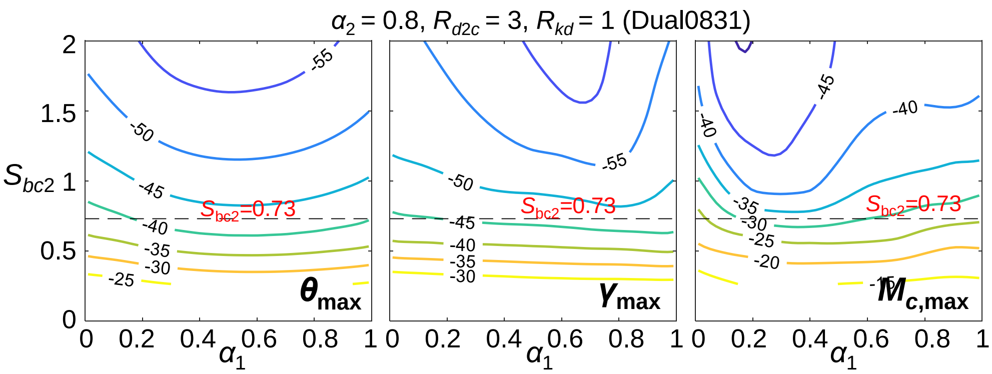


Figure 8.3.7 Reduction factors when α_2=0.8, R_{d2c}=3, and R_{kd}=1 (Dual0831)

8.3.3 Analysis result of the design example

Table 8.3.3 shows the modal analysis result of the design example models. The case when α_1 = 0.7 generates stronger outrigger effect and decrease the vibration period more than the case when α_1 = 0.3. The models with α_1 = 0.7 also show greater 1st mode mass participation factor than the models with α_1 = 0.3. Table 8.3.4 shows the SA results and the θ_{max} reduction factors. All the 12 models (dual BRB-outrigger systems) perform better than Single1 and Single3 models (single BRB-outrigger systems) in reducing the θ_{max} and most of the γ_{max} responses. Figure 8.3.8 and Figure 8.3.9 show the θ_{max} and γ_{max} calculated from the NLRHA (originally observed ground motions). The numbers shown in Figure 8.3.8 and Figure 8.3.9 indicate the reduction factors in θ_{max} and γ_{max}. The cases when α_2 = 0.8 generally perform better than the cases when α_2 = 0.7 in mitigating θ_{max} and γ_{max} responses. In addition, placing the BRB_Rdc3 in the upper outrigger could exhibit a smaller θ_{max} and γ_{max} responses. Figure 8.3.10 and Figure 8.3.11 show the maximum base shear ($V_{c,max}$) and maximum overturning moment at core structure base ($M_{c,max}$) responses. The numbers shown in Figure 8.3.10 and Figure 8.3.11 indicate the reduction factors in $V_{c,max}$ and $M_{c,max}$. The trends of $M_{c,max}$ and $V_{c,max}$ responses are similar to the θ_{max} responses. In addition, when α_1 is at 0.3, the $M_{c,max}$ can be better reduced than the models with α_1 = 0.7 under some ground motions. The $M_{c,max}$ can be reduced efficiently when R_{kd} is large (strong lower BRB-outrigger). Figure 8.3.12 and Figure 8.3.13 show the R_{CPD} and the E_d/E_s ratio, respectively. The BRB in upper outrigger (BRB$_2$) cumulative more R_{CPD} and dissipate more amount of energy than the BRB in lower outrigger (BRB$_1$). The BRB$_1$ could stay elastic deformation (R_{CPD} = 0 and E_d/E_s = 0) in small earthquakes for the cases when R_{d2c} = 3 and R_{kd} = 3. In addition, the BRB$_2$ in the cases when α_2 = 0.8 generally exhibit better energy dissipation performance (larger R_{CPD} or larger E_d/E_s values). Figure 8.3.14 shows the maximum axial force demand of the perimeter column ($C_{1,max}$). When R_{d2c} = 3 or R_{kd} = 3, the $C_{1,max}$ is significantly amplified. In summary, for the single BRB-outrigger design model, if the upper outrigger can be moved to α_2 = 0.8, the overall seismic responses can be slightly reduced. If the $M_{c,max}$ response is critical, placing a strong lower BRB-outrigger (R_{kd} = 3) at α_1 around 0.3 would be helpful. If the θ_{max} and γ_{max} are critical, placing the lower BRB-outrigger at α_2 around 0.8 is recommended. The large R_{kd} and R_{d2c} values are not recommended as they increase the BRB cost and significantly amplify the perimeter column axial force demands. In addition, the strong lower BRB may be prone to stay elastic deformation during earthquakes.

Table 8.3.3 modal analysis result of the design example

model	vibration period (sec)				mass participation ratio (%)			
	1st mode	2nd mode	3rd mode	4th mode	1st mode	2nd mode	3rd mode	4th mode
Core	5.543	0.884	0.316	0.161	68.2	20.9	7.2	3.7
Dual0711L	4.280	0.822	0.314	0.160	69.5	19.2	7.6	3.7
Dual0711U	4.055	0.845	0.310	0.160	70.9	18.5	6.9	3.7
Dual0713L	4.226	0.815	0.314	0.160	69.2	19.5	7.7	3.6
Dual0713U	3.958	0.845	0.309	0.160	70.9	18.5	6.8	3.8
Dual0731L	4.083	0.805	0.314	0.160	69.8	18.9	7.7	3.6
Dual0731U	3.869	0.835	0.309	0.160	71.4	18.0	6.8	3.8
Dual0811L	4.259	0.802	0.312	0.161	69.9	19.4	7.2	3.5
Dual0811U	4.058	0.823	0.307	0.161	71.8	17.7	6.9	3.6
Dual0813L	4.194	0.795	0.312	0.161	69.5	19.7	7.2	3.6
Dual0813U	3.959	0.822	0.306	0.160	72.0	17.5	6.9	3.6
Dual0831L	4.073	0.782	0.311	0.161	70.1	19.2	7.2	3.5
Dual0831U	3.889	0.810	0.304	0.160	72.5	17.2	6.9	3.4

Table 8.3.4 SA results of the Core, Single1, and Single3 models

Model	θ_{max}		γ_{max} (% rad.)	h_{eq}
	θ_{max} (% rad.)	reduction factor		
Core	1.424	-	2	0.02
Single1	0.908	-36.2%	1.2	0.063
Single3	0.876	-38.5%	1.16	0.059
Dual0711L	0.854	-40.0%	1.163	0.07
Dual0711U	0.791	-44.5%	1.08	0.082
Dual0713L	0.869	-39.0%	1.192	0.063
Dual0713U	0.796	-44.1%	1.094	0.072
Dual0731L	0.846	-40.6%	1.155	0.058
Dual0731U	0.804	-43.5%	1.099	0.061
Dual0811L	0.833	-41.5%	1.072	0.072
Dual0811U	0.776	-45.5%	0.984	0.083
Dual0813L	0.846	-40.6%	1.099	0.065
Dual0813U	0.783	-45.0%	0.995	0.072
Dual0831L	0.837	-41.2%	1.068	0.057
Dual0831U	0.799	-43.9%	0.999	0.059

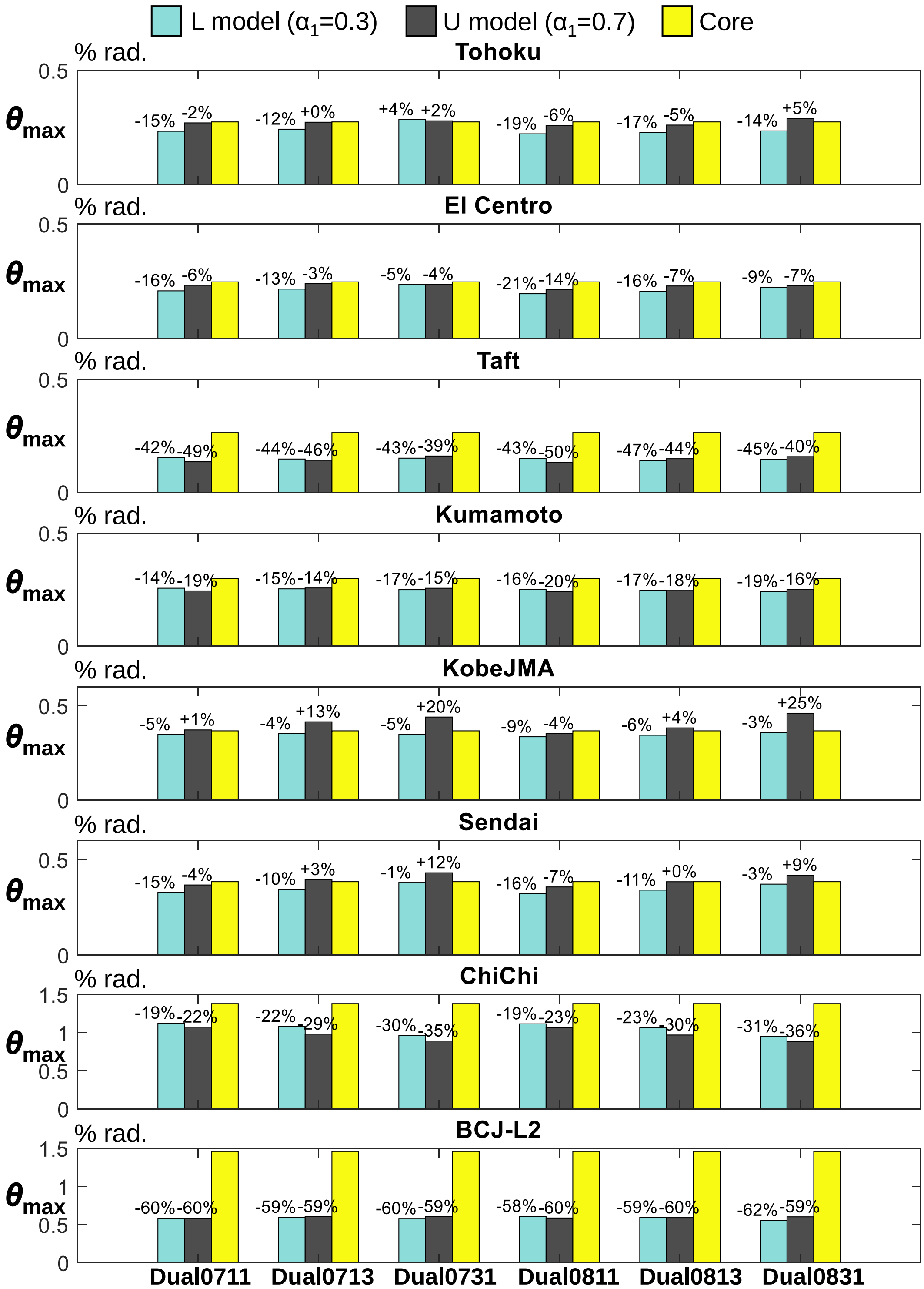


Figure 8.3.8 The θ_{max} responses of the dual BRB-outrigger system calculated from the NLRHA

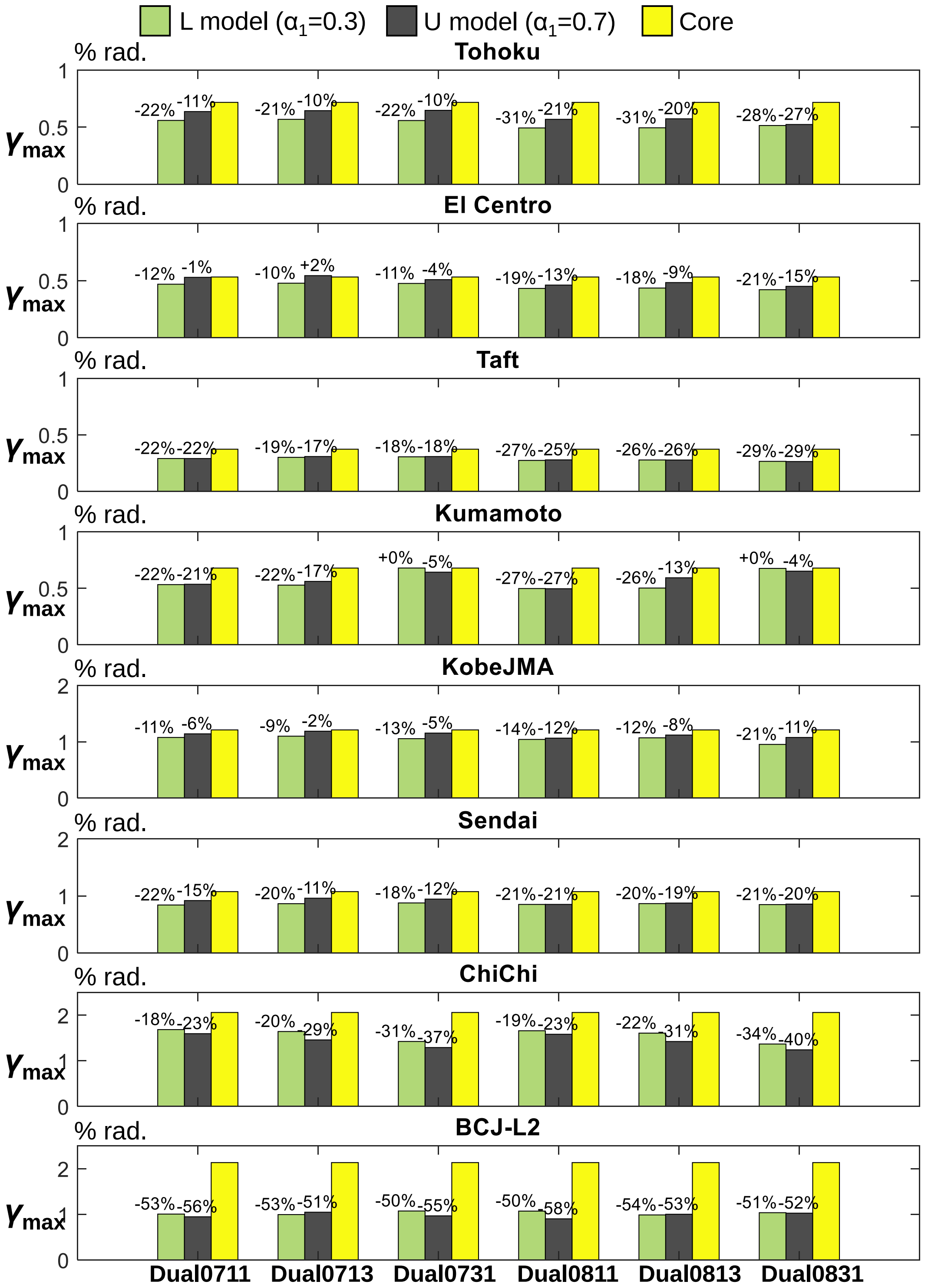


Figure 8.3.9 The γ_{max} responses of the dual BRB-outrigger system calculated from the NLRHA

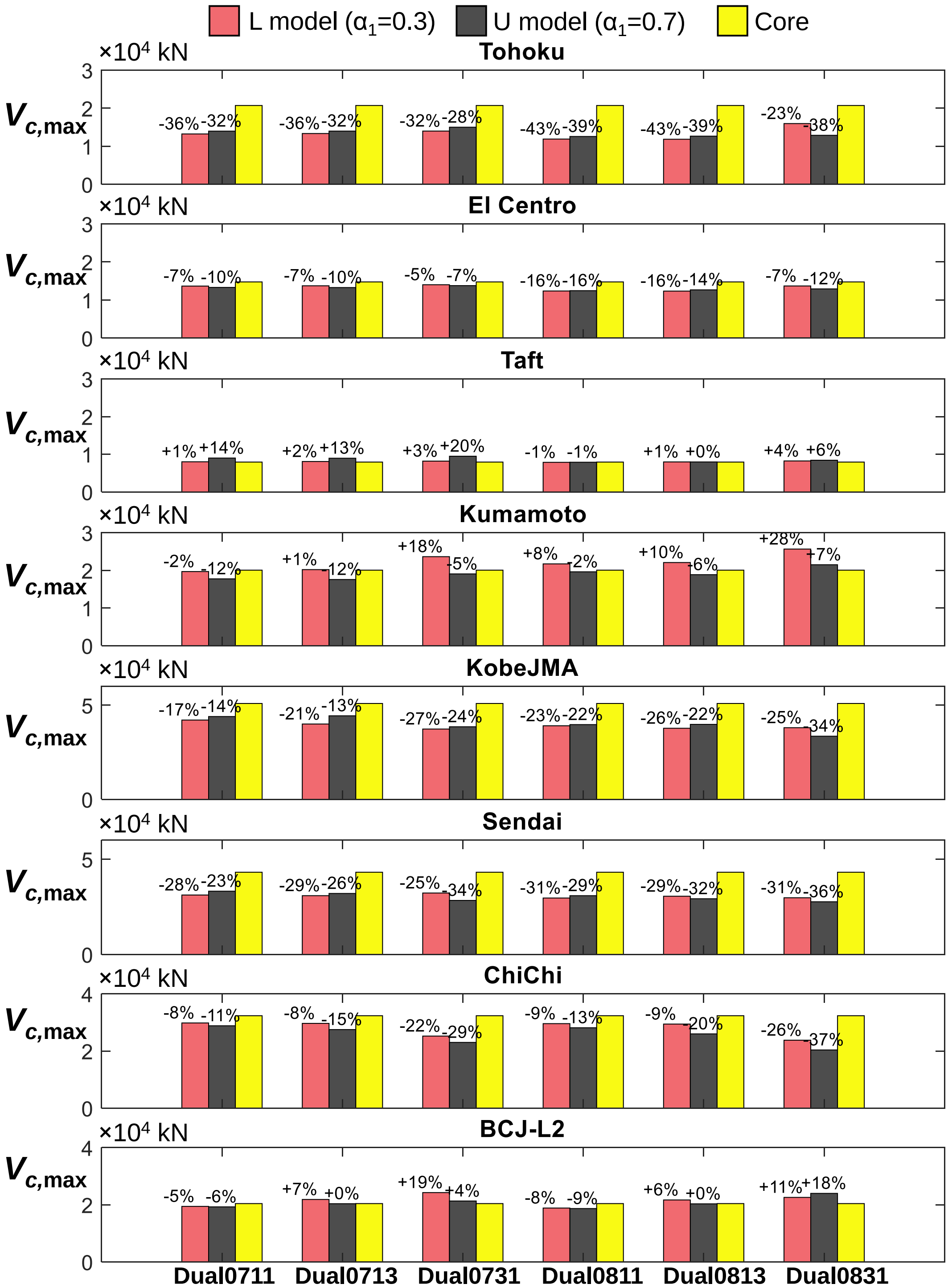


Figure 8.3.10 The $V_{c,\max}$ responses of the dual BRB-outrigger system calculated from the NLRHA

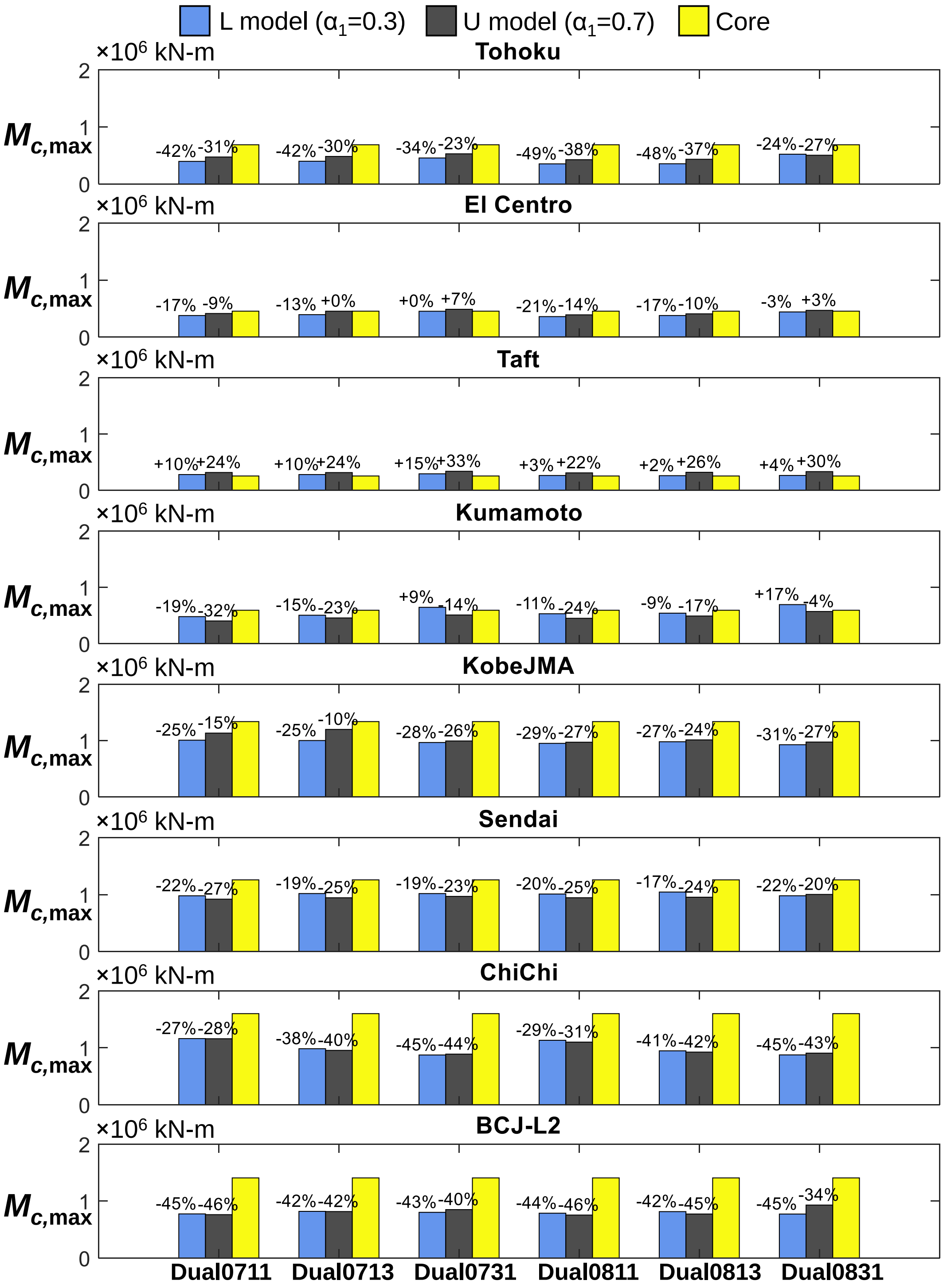


Figure 8.3.11 The $M_{c,\max}$ responses of the dual BRB-outrigger system calculated from the NLRHA

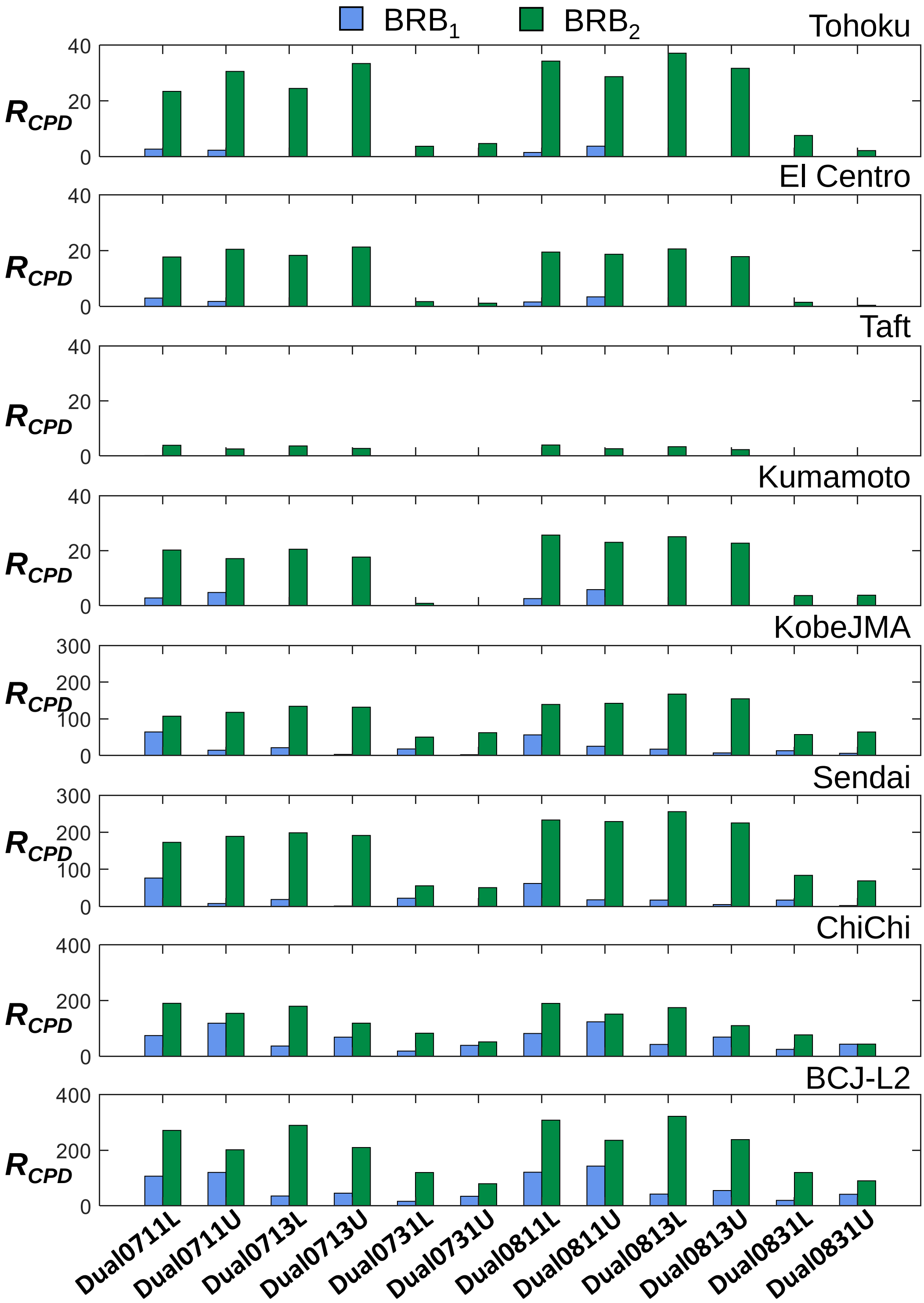


Figure 8.3.12 The R_{CPD} responses of the dual BRB-outrigger system calculated from the NLRHA

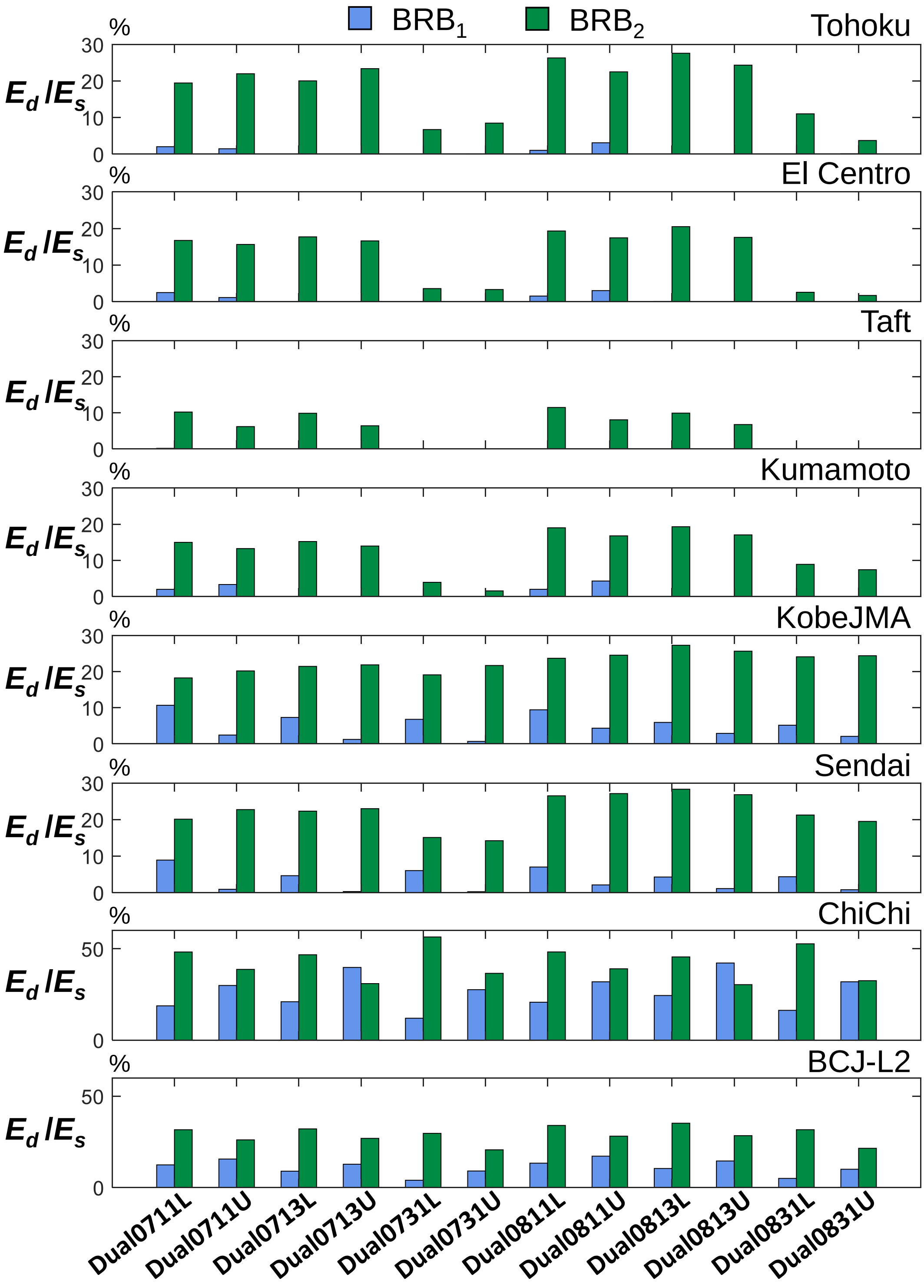


Figure 8.3.13 The E_d/E_s responses of the dual BRB-outrigger system calculated from the NLRHA

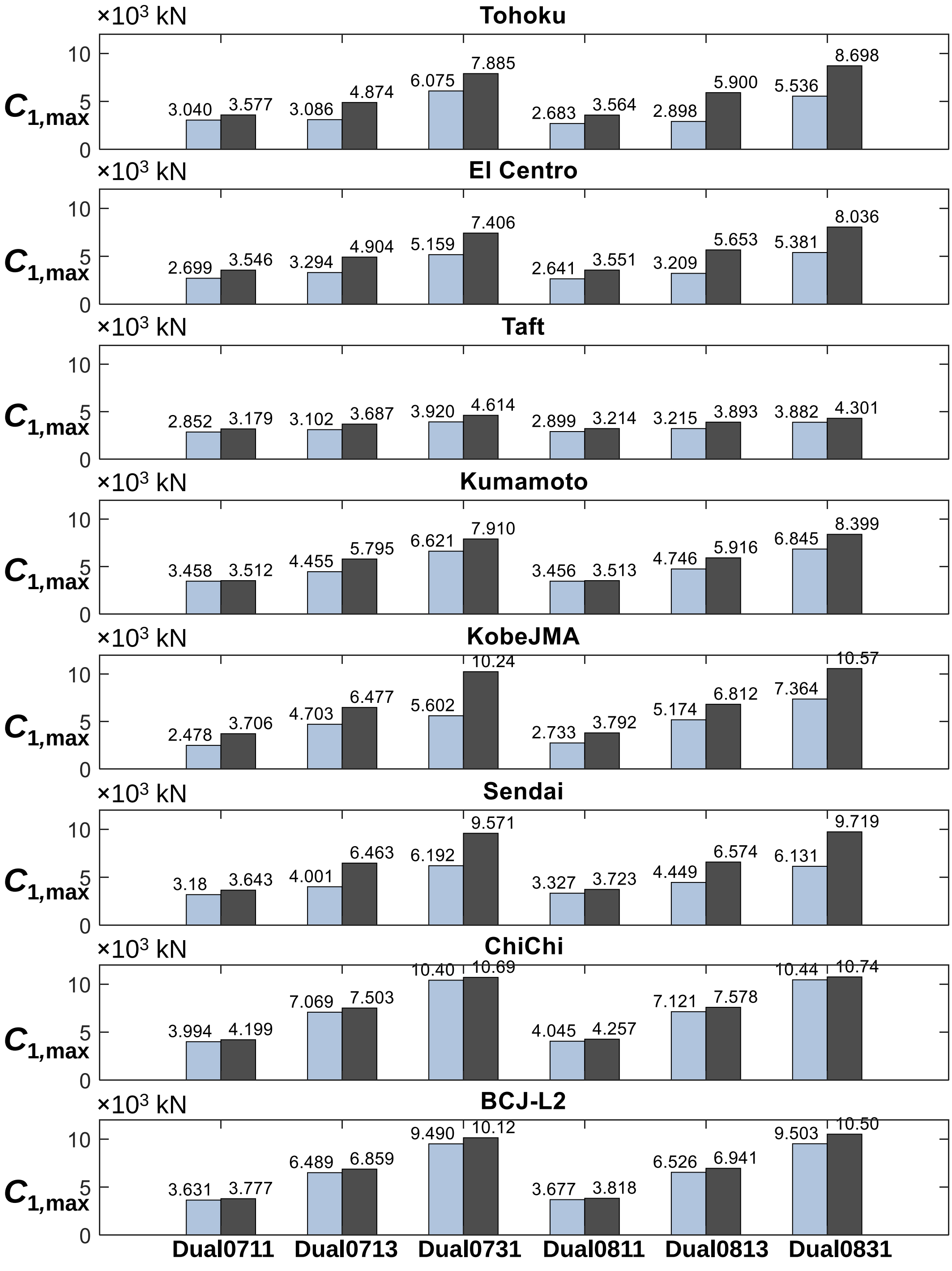


Figure 8.3.14 The $C_{1,\max}$ responses of the dual BRB-outrigger system calculated from the NLRHA

8.4 SUMMARY

This chapter summarizes the analysis results for optimal design and presents the design flow chart for designers. The design examples on single and dual BRB-outrigger systems are presented. The summaries of this chapter are as follows:

(1) Based on the analysis results, the single and dual BRB-outrigger system with similar S_{bc07} and $S_{bc2,07}$ values exhibit similar seismic responses. The dual BRB-outrigger system generally results in 5% to 10% response smaller than the single BRB-outrigger system. Therefore, the single BRB-outrigger system should be a more economical solution than the dual BRB-outrigger system.

(2) Based on the design example, the larger BRB stiffness parameters (R_{dc}, R_{db}, R_{d2c}, R_{kd}) lead to smaller θ_{max} and γ_{max} responses, but they could excessively amplify the perimeter column axial force demand. For the design practices, it is suggested to start the BRB design with the small BRB stiffness parameters.

(3) The dual BRB-outrigger design examples show that the upper BRB-outrigger dominates the seismic response, and the presence of the lower BRB-outrigger further improved the seismic response by reducing θ_{max}, γ_{max}, and $M_{c,max}$ if compared with the single BRB-outrigger system by around 10%. For economical design purpose, it is suggested to start with single BRB-outrigger system design.

(4) For engineering practices, the single BRB-outrigger system should be a more economical solution, as the dual BRB-outrigger system could provide only marginal improvements in seismic response over the single BRB-outrigger system. However, the improvements could be necessary and critical. In the design practices, for both single and dual BRB-outrigger systems, choosing BRB-outrigger elevations at their optimal locations and selecting $S_{bc2,07}$ as large as possible are critical as they could greatly enhance the overall outrigger effect.

(5) Based on the analysis results of this study, the reductions in seismic response are not proportional to the number of BRB-outrigger, but the more layers of BRB-outrigger can lead to a larger perimeter column axial force demand. Therefore, the single BRB-outrigger system should be a more economical solution. In addition, the taller building has a smaller outrigger stiffness parameter because of the longer perimeter columns. It would be less efficient for BRB-outrigger system in mitigating seismic response for a very tall building. Therefore, the maximum of

two layers of BRB-outrigger is considered in this study. If the seismic response of the dual BRB-outrigger system is greater than allowable limit, the engineers should consider applying additional energy dissipating devices in the core structure or increasing the core structure stiffness and strength.

8.5 REFERENCES

AISC (American Institute of Steel Construction) (2016) "Specifications for Structural Steel Buildings, ANSI-AISC 360-16," in *AISC (American Institute of Steel Construction)*. doi: 111.

ANSI/AISC 341-16 (2016) *Seismic Provisions for Structural Steel Buildings, American Institute of Steel Construction*. doi: 111.

9

BRB-OUTRIGGER CONFIGURATIONS

CHAPTER CONTENTS

9.1 Introduction 9-3
9.2 BRB-outrigger configurations 9-3
9.2.1 Ordinary BRB-outrigger system 9-3
9.2.2 BRB-truss outrigger system 9-4
9.2.3 Giant-BRB outrigger system 9-4
9.3 Parameter definitions and analytical models 9-6
9.4 Analysis results 9-8
9.4.1 Optimal outrigger elevations 9-8
9.4.2 Effects of outrigger effect factor and outrigger stiffness ratio 9-14
9.5 Design Examples 9-18
9.5.1 Introduction of the example models 9-18
9.5.2 Seismic response of the example models 9-22
9.5.3 Comparison between OB, BT, and GB outrigger configurations 9-35
9.6 Summary 9-36

9.1 INTRODUCTION

This chapter presents the applications of BRB-outrigger in various configurations in order to fit individual architecture requirements for design practices. In addition to the ordinary BRB-outrigger (OB outrigger) introduced in the previous chapters, two different BRB-outrigger configurations are introduced. The proposed analysis method and optimal designs for structure with single BRB-outrigger system are modified in order to apply on various BRB-outrigger configurations. The design examples of structure with different BRB-outrigger configurations are introduced and their analysis results are compared.

9.2 BRB-OUTRIGGER CONFIGURATIONS

9.2.1 Ordinary BRB-outrigger system

The design of OB outrigger could be uneconomical as the required outrigger truss member could be very large in order to provide sufficient outrigger truss flexural stiffness (k_t) when outrigger span (l_t) is very long. In addition, when the required BRB axial yield deformation ($u_{d,y}$) is large, the required BRB length could be longer than one story height. Therefore, two different BRB-outrigger configuration alternatives are introduced. In order to continue using the proposed UM and DM models, new parameters are introduced. As indicated in Equation (3.2.7), the moment applied by the OB outrigger ($M_{o,OB}$) is modified as follows:

$$M_{o,OB} = \frac{2l_t^2}{\frac{\alpha}{k_c} + \frac{1}{k_d} + \frac{1}{k_t}}\theta_1 = \frac{2l_t^2}{\frac{\alpha}{k_c} + \frac{1}{k_{og,OB}}}\theta_1 = k_{rg,OB}\theta_1 \tag{9.2.1}$$

Where $k_{og,OB}$ is the stiffness combined with k_d and k_t, and is known as the outrigger stiffness of the OB outrigger. $k_{rg,OB}$ is the rotational stiffness applied by OB outrigger. When the BRB yields, the core structure rotation at outrigger elevation θ_y is calculated as follows:

$$\theta_y = \frac{k_d}{l_t}\left(\frac{\alpha}{k_c} + \frac{1}{k_t} + \frac{1}{k_d}\right)u_{d,y} = \frac{k_d}{l_t}\left(\frac{\alpha}{k_c} + \frac{1}{k_{og,OB}}\right)u_{d,y} \tag{9.2.2}$$

Based on this OB outrigger model, the UM and DM models have been developed and used in SA and NLRHA as described in previous chapters.

9.2.2 BRB-truss outrigger system

Figure 9.2.1 shows the BRB-truss outrigger (BT outrigger) configuration. The BRBs are used as braces in the outrigger truss. The ends of the top and bottom chords that connect with core and perimeter column can be designed with either shear or moment connection detail. As the BRBs and the outrigger truss act in parallel, slight plastic deformation in the outrigger truss member might be allowed in order to prevent too large outrigger truss member size. Therefore, if compared with the OB outrigger configuration, the outrigger truss member and the BRB sizes could be reduced. In addition, it is anticipated that the increased number of BRB could provide a more stable energy dissipation mechanism. As shown in Figure 9.2.1, the core structure rotation at outrigger elevation (θ_1) can be calculated as follows:

$$\theta_1 = \frac{1}{l_t}\left(u_{c,BT} + u_{bt}\right) \tag{9.2.3}$$

Where $u_{c,BT}$ and u_{bt} are the perimeter column axial deformation and the flexural deformation of the BT outrigger, respectively. Therefore, if k_{bt} is the flexural stiffness of the BT outrigger, the moment applied by the BT outrigger ($M_{o,BT}$) can be calculated as follows:

$$M_{o,BT} = \frac{2l_t^2}{\frac{\alpha}{k_c} + \frac{1}{k_{bt}}}\theta_1 = \frac{2l_t^2}{\frac{\alpha}{k_c} + \frac{1}{k_{og,BT}}}\theta_1 = k_{rg,BT}\theta_1 \tag{9.2.4}$$

Where $k_{rg,BT}$ is the rotational stiffness provided by the BT outrigger system. If compare with Equation (9.2.2), the outrigger stiffness $k_{og,BT}$ equals to k_{bt} in the BT outrigger configuration. If $u_{bt,y}$ is the BT outrigger flexural deformation when the first BRB yields, the θ_y can be expressed as follows:

$$\theta_y = \frac{k_{bt}}{l_t}\left(\frac{\alpha}{k_c} + \frac{1}{k_{bt}}\right)u_{bt,y} \tag{9.2.5}$$

In order to use the DM model with OB outrigger configuration to analyze the structure with BT outrigger, the k_d, k_t, and $u_{d,y}$ in Equation (9.2.1) and Equation (9.2.2) have to be replaced by k_{bt}, an infinite value, and $u_{bt,y}$, respectively.

9.2.3 Giant-BRB outrigger system

Figure 9.2.2 shows the Giant-BRB outrigger (GB outrigger) system. The GB outrigger removes the outriggers but connects the core structure and the perimeter column by using a giant BRB. When BRB axial force of P_{gb} is developed, the BRB and the floor beam below the BRB act a force couple on the core structure as shown in Figure 9.2.2. The force couple applies a resisting moment ($M_{o,gb}$) on the core structure. It should be noted that both the BRB and the floor beam are acting as truss elements. In addition, the floor beam is usually strong enough to resist the maximum force developed by the BRB, and is sufficiently stiff so that the shortening or elongation of the floor beam can be neglected. If $u_{c,GB}$ and $u_{d,gb,v}$ are the vertical deformations of the perimeter column below the outrigger elevation and the BRB in the GB outrigger configuration (BRB_GB), and $k_{d,gb}$ is the axial stiffness of BRB_GB, the corresponding core structure rotation at the outrigger elevation (θ_1) and $M_{o,gb}$ can be calculated as follows:

$$\theta_1 = \frac{1}{l_t}\left(u_{c,GB} + u_{d,gb,v}\right) = \frac{P_{gb}}{l_t}\left(\frac{\alpha \sin\eta}{k_c} + \frac{1}{k_{d,gb}\sin\eta}\right) \tag{9.2.6}$$

$$M_{o,GB} = 2P_{gb}h_t\cos\eta = 2P_{gb}l_t\sin\eta = \frac{2l_t^2}{\frac{\alpha}{k_c} + \frac{1}{k_{d,gb}\sin^2\eta}}\theta_1 = \frac{2l_t^2}{\frac{\alpha}{k_c} + \frac{1}{k_{og,GB}}}\theta_1 = k_{rg,GB}\theta_1 \tag{9.2.7}$$

Where h_t and η (=$\tan^{-1}(h_t/l_t)$) are the vertical span and inclined angle of BRB_GB, respectively. $k_{rg,GB}$ is the rotational stiffness provided by the GB outrigger system. The outrigger stiffness $k_{og,GB}$ equals to $k_{d,gb}\sin^2\eta$ in the GB outrigger configuration. If $u_{d,ygb}$ is axial yield deformation of BRB_GB, the θ_y can be expressed as follows:

$$\theta_y = \frac{k_{d,gb}u_{d,ygb}}{l_t}\left(\frac{\alpha\sin\eta}{k_c} + \frac{1}{k_{d,gb}\sin\eta}\right) = \frac{k_{d,gb}\sin^2\eta}{l_t}\left(\frac{\alpha}{k_c} + \frac{1}{k_{d,gb}\sin^2\eta}\right)\left(\frac{u_{d,ygb}}{\sin\eta}\right) \tag{9.2.8}$$

If compared with Equation (9.2.1) and Equation (9.2.2), the DM model can be used to model the structure with GB outrigger by replacing k_d, k_t, and $u_{d,y}$ by $k_{d,gb}\sin^2\eta$, an infinite value, and $u_{d,ygb}/\sin\eta$, respectively. It is anticipated that the GB outrigger could save steel usage as the outrigger truss is not necessary. In addition, when the required BRB yield deformation is large, the long BRB in GB outrigger could be easily applied. However, the BRB_GB may be required to span across more than one story, which may reduce usable floor areas. As indicated in Equation (9.2.7), the larger η, the greater $M_{o,GB}$ is. The seismic performance of structures with OB, BT, and GB outrigger can be estimated by using the DM model (with OB outrigger configuration) with modified parameters as shown in Table 9.2.1, and its effectiveness is verified by the analysis results calculated by using MBM models, which follow individual outrigger configuration detail. The detail of MBM model for each BRB-outrigger configuration is introduced in the following sections.

Table 9.2.1 The parameters used in DM model for each outrigger configuration

configuration	outrigger stiffness (k_{og})	k_t	k_d	$u_{d,y}$
OB outrigger	$k_{og,OB} = \dfrac{k_d k_t}{k_d + k_t}$	k_t	k_d	$u_{d,y}$
BT outrigger	$k_{og,BT} = k_{bt}$	∞	k_{bt}	$u_{bt,y}$
GB outrigger	$k_{og,GB} = k_{d,gb} \sin^2 \eta$	∞	$k_{d,gb} \sin^2 \eta$	$\dfrac{u_{d,ygb}}{\sin \eta}$

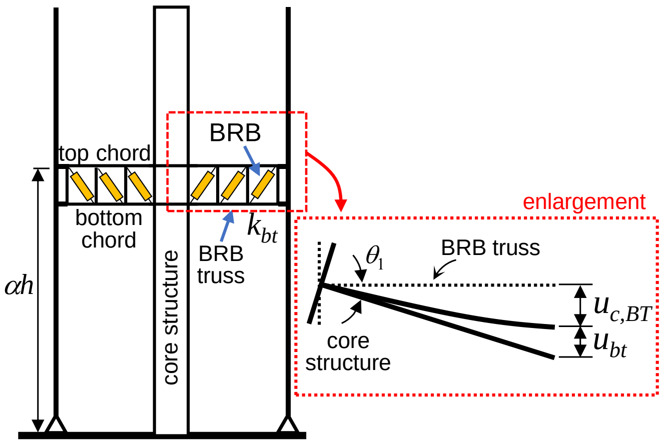


Figure 9.2.1 The simplified structure with BT outrigger

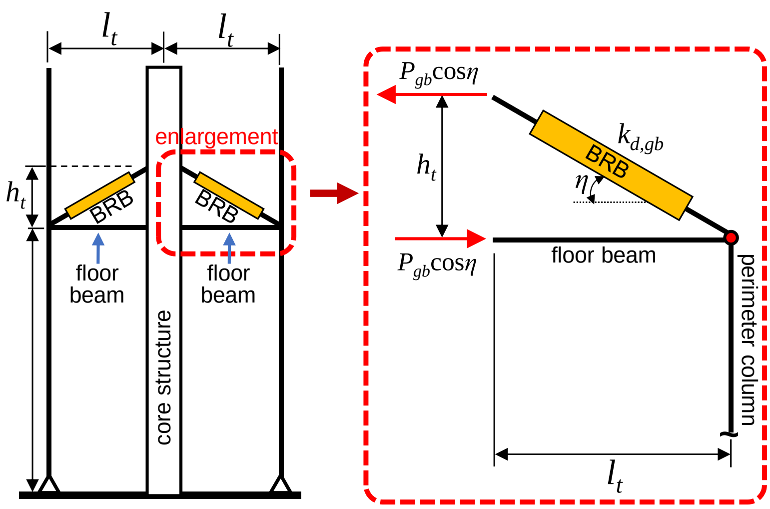


Figure 9.2.2 The simplified structure with GB outrigger

9.3 PARAMETER DEFINITIONS AND ANALYTICAL MODELS

In each BRB-outrigger configuration, the relationship between outrigger stiffness including BRB axial stiffness and perimeter column stiffness is critical. Therefore, two dimensionless indexes are newly defined for the purpose of parametric study for structure with either OB, BT, or GB outrigger. The outrigger effect (S_{cc}) is defined as the ratio of rotational stiffness provided by the outrigger when k_t and k_d are infinity ($k_c l_t^2/\alpha$) to the core structure's rotational stiffness (EI/h), and can be expressed as follows:

$$S_{cc}(\alpha)=\frac{l_t^2 h k_c}{\alpha EI},\ S_{cc07}(0.7)=\frac{l_t^2 h k_c}{0.7EI} \tag{9.3.1}$$

The value of S_{cc} when α equals to 0.7 (S_{cc07}, outrigger effect factor) is used to indicate the magnitude of the outrigger effect. The stiffer perimeter column (greater k_c) and longer outrigger span (greater l_t) can enhance the outrigger effect. For taller structures, the EI could significantly increase because of the larger seismic demand. Therefore, the S_{cc07} would be smaller for taller structures. In the design practices, the k_c should be primarily determined by the gravity load demands, and the l_t is determined from the architectural plan. Therefore, the outrigger effect factor S_{cc07} also reflects the suitability of adopting outrigger in particular buildings. The structure with larger S_{cc07} value suggests that the efficiency in mitigating seismic response would be higher when the outrigger system is adopted. In addition, the outrigger stiffness ratio (R_{oc}) is also newly defined as the ratio of outrigger stiffness, k_{og}, to the perimeter column axial stiffness, k_c, and is expressed as follows:

$$R_{oc}=\frac{k_{og}}{k_c},\ \begin{cases} k_{og}=k_{og,OB}=1/\left(1/k_d+1/k_t\right), & \text{for OB outrigger configuration} \\ k_{og}=k_{og,BT}=k_{bt}, & \text{for BT outrigger configuration} \\ k_{og}=k_{og,GB}=k_{d,gb}\sin^2\eta, & \text{for GB outrigger configuration} \end{cases} \tag{9.3.2}$$

The R_{oc} indicates the stiffness provided by the outrigger system. After the perimeter column size is determined, the R_{oc} could provide engineers a rough estimation on the required BRB sizes and outrigger truss stiffness according to the selected R_{oc} value.

The SA and NLRHA analysis results calculated from using the analytical models shown in Table 3.4.1 and Table 7.3.1 (θ_r=1/750) are utilized to investigate the optimal design when the new indexes of S_{cc07} and R_{oc} are used. Table 9.3.1 shows the detail of the analytical models and the ranges of S_{cc07} and R_{oc}. If the value of R_{dt} equals to 0.1, the relationship between R_{oc} and R_{d2c} can be expressed as follows:

$$R_{oc}=\frac{R_{d2c}}{1+R_{dt}}=\frac{R_{d2c}}{1.1} \tag{9.3.3}$$

Figure 9.3.1 shows the distribution of S_{cc07} value with respect to R_{oc}. Based on Figure 9.3.1, the valid ranges of S_{cc07} and R_{oc} are from 0 to 4 and from 0 to 3, respectively.

Table 9.3.1 The parameters used in DM model for each outrigger configuration

model	h (m)	EI (kN-m^2)	l_t (m)	S_{bc2}	S_{cc07}	fundamental period of core structure (sec)	R_{oc}
16-story	64	4.1×10^9	16	3.03	3.17 – 7.36	1.74	0.09,
16-storyB	64	4.1×10^9	14.5	2.48	2.60 – 6.02	1.74	0.45,
16-storyC	64	4.1×10^9	12.8	1.93	2.02 – 4.69	1.74	0.91,
32-story	128	1.6×10^{10}	16	1.38	1.45 – 3.35	3.50	1.36,
32-storyD	128	1.6×10^{10}	13.8	1.02	1.07 – 2.48	3.50	1.82,
64-story	256	6.5×10^{10}	16	0.66	0.69 – 1.60	6.92	2.27,
96-story	384	2.2×10^{11}	16	0.30	0.31 – 0.73	9.76	2.73

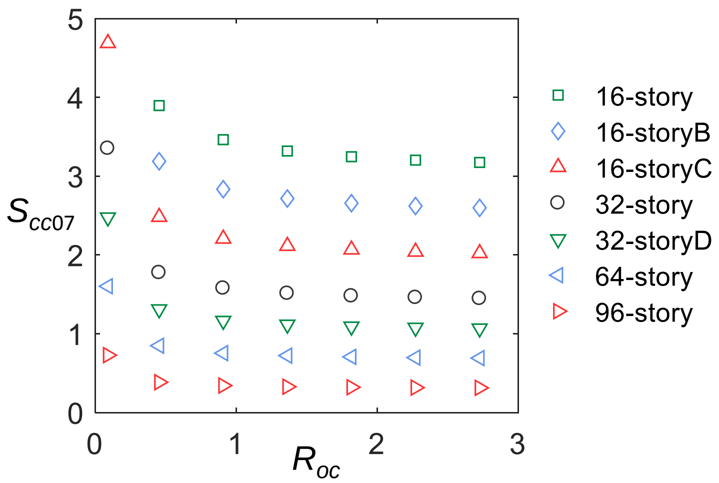


Figure 9.3.1 Distribution of S_{cc07} value with respect to R_{oc} for each analytical model

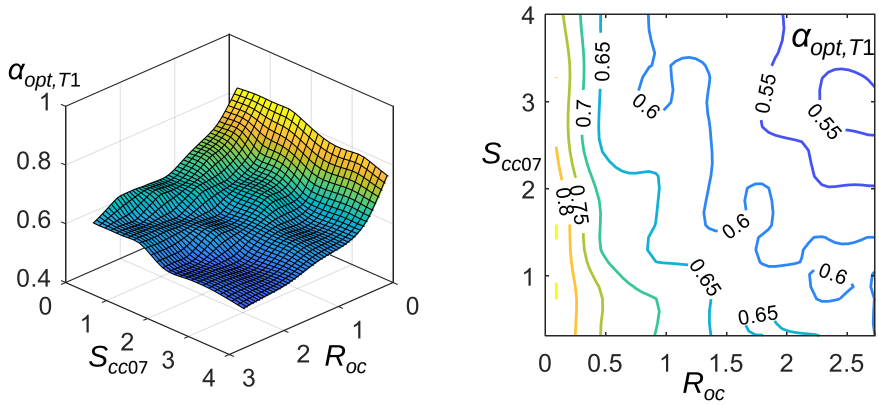
9.4 ANALYSIS RESULTS

9.4.1 Optimal outrigger elevations

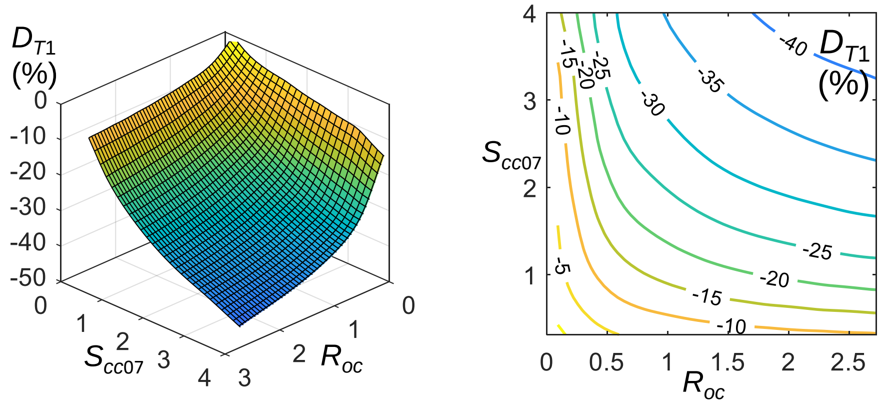
As the BRB-outrigger applies a resisting moment on the core structure, the structure becomes stiffer when the outrigger effect is more significant. Therefore, the drop of the first mode vibration period of the elastic system if compared with the core model without outrigger effect (D_{T1}) is used to estimate the effectiveness of BRB-outrigger. The smaller value of D_{T1} suggests the outrigger effect is more significant. Figure 9.4.1a shows the distribution of α when the value of D_{T1} is smallest ($\alpha_{opt,T1}$) with respect to R_{oc} and S_{cc07}. The distribution of D_{T1} when the outrigger locates at $\alpha_{opt,T1}$ is shown in Figure 9.4.1b. When the values of S_{cc07} and R_{oc} are larger, the outrigger effect is more significant as the D_{T1} value is smaller. The $\alpha_{opt,T1}$ ranges from 0.5 to 0.8, and is lower when the values of R_{oc} and S_{cc07} are larger. Based on the analysis results, the relationship between $\alpha_{opt,T1}$, R_{oc}, and S_{cc07} can be approximately fitted by using the following polynomial:

$$\alpha_{opt,T1}\left(R_{oc},S_{cc07}\right)=0.9165-0.3672R_{oc}-0.05033S_{cc07}+0.188R_{oc}^2-0.006691R_{oc}S_{cc07}$$
$$+0.005755S_{cc07}^2-0.03384R_{oc}^3+0.002815R_{oc}^2S_{cc07}+0.0004949R_{oc}S_{cc07}^2 \quad (9.4.1)$$
$$-0.0001824S_{cc07}^3 \quad \left(0<S_{cc07}\le 4,\ 0<R_{oc}\le 3\right)$$

Within the valid ranges of S_{cc07} and R_{oc}, the coefficient of determination of Equation (9.4.1) is 0.95.



(a)



(b)

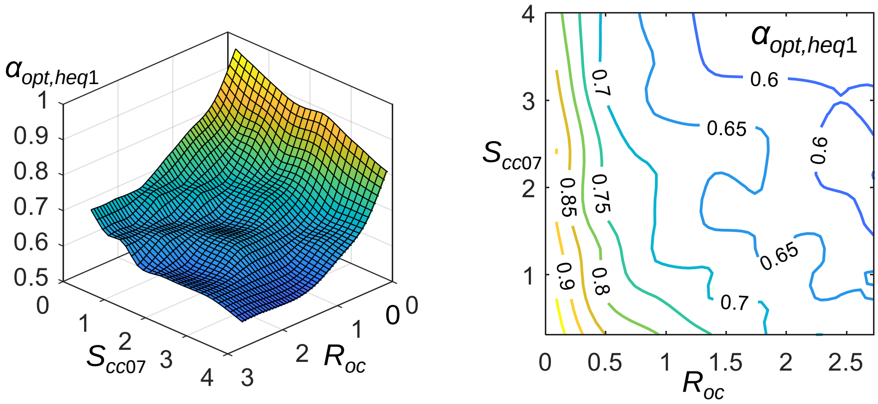
Figure 9.4.1 The (a) $\alpha_{opt,T1}$ and (b) D_{T1} distributions with respect to S_{cc07} and R_{oc}

After the BRB yields, the BRB dissipates energy through its hysteretic response. As the first mode dominates the seismic response, the value of the equivalent damping ratio calculated from the 1st mode response ($h_{eq,1}$) is used to identify the energy dissipation efficiency of the BRB-outrigger system. The larger value of $h_{eq,1}$ developed by the BRB-outrigger suggests the energy dissipation efficiency is higher. Figure 9.4.2a shows the distribution of the outrigger elevation α when the value of $h_{eq,1}$ is maximum ($\alpha_{opt,heq1}$) with respect to S_{cc07} and R_{oc}. The distribution of $h_{eq,1}$ when the outrigger locates at $\alpha_{opt,heq1}$ is shown in Figure 9.4.2b. The BRB-outrigger with larger R_{oc} and S_{cc07} values imposes greater outrigger effect and thus results in greater

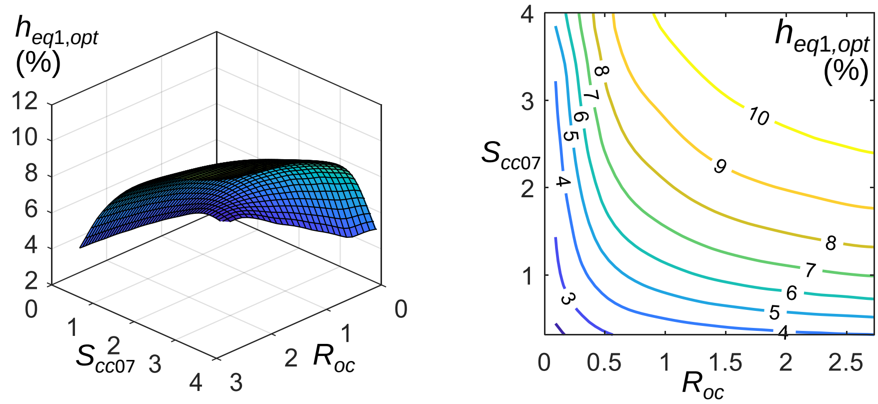
$h_{eq,1}$ value. The distribution of $\alpha_{opt,heq1}$ is similar to $\alpha_{opt,T1}$. The $\alpha_{opt,heq1}$ ranges from 0.6 to 0.9, and is lower when the values of R_{oc} and S_{cc07} are larger. The relationship between $\alpha_{opt,heq1}$, R_{oc}, and S_{cc07} can be fitted by using the following polynomial based on the analysis results:

$$\begin{aligned}\alpha_{opt,heq1}\left(R_{oc},S_{cc07}\right)=&1.028-0.4112R_{oc}-0.06694S_{cc07}+0.2054R_{oc}^2-0.01297R_{oc}S_{cc07}\\&+0.007359S_{cc07}^2-0.03614R_{oc}^3+0.002813R_{oc}^2S_{cc07}+0.003196R_{oc}S_{cc07}^2\\&-0.0002686S_{cc07}^3\quad\left(0<S_{cc07}\le 4,\ 0<R_{oc}\le 3\right)\end{aligned} \tag{9.4.2}$$

Within the valid ranges of S_{cc07} and R_{oc} values, the coefficient of determination of Equation (9.4.2) is 0.96.



(a)



(b)

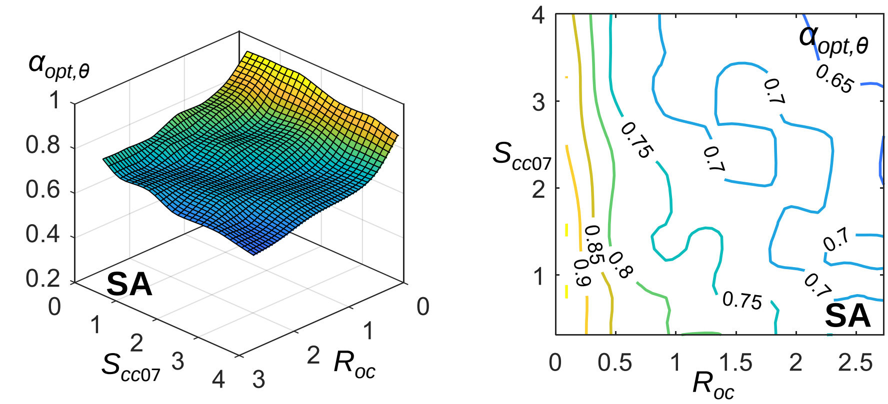
Figure 9.4.2 The (a) $\alpha_{opt,heq1}$ and (b) $h_{eq,1}$ distributions with respect to S_{cc07} and R_{oc}

Figure 9.4.3 shows the distributions of outrigger elevation α when $\theta_{\max}$ is minimum ($\alpha_{opt,\theta}$) with respect to R_{oc} and S_{cc07} calculated from SA and NLRHA. The distributions of $\theta_{\max}$ when outrigger locates at $\alpha_{opt,\theta}$ calculated from SA and NLRHA are shown in Figure 9.4.4. Both the SA and NLRHA results indicate that the larger values of R_{oc} and S_{cc07} result in a smaller $\theta_{\max}$ response. The distribution of $\theta_{\max}$

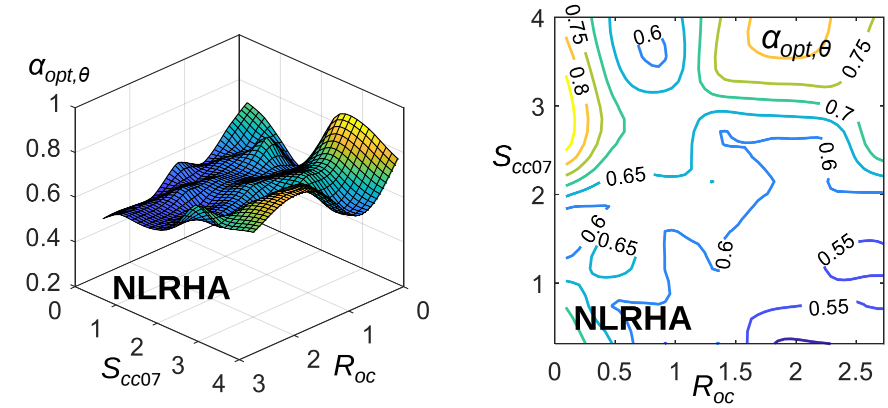
calculated from NLRHA and SA are similar. As the NLRHA results are sensitive to different ground motions, the $\alpha_{opt,\theta}$ calculated from NLRHA results does not exhibit similar distribution as SA results. However, both the SA and NLRHA results suggest that the $\alpha_{opt,\theta}$ ranges approximately from 0.6 to 0.8. The $\alpha_{opt,\theta}$ is smaller when the values of R_{oc} and S_{cc07} are larger, which is similar to the $\alpha_{opt,T1}$ and $\alpha_{opt,heq1}$. Based on the SA results, the relationship between $\alpha_{opt,\theta}$, R_{oc}, and S_{cc07} are fitted by using the following polynomial:

$$\begin{aligned}\alpha_{opt,\theta}\left(R_{oc},S_{cc07}\right) = {} & 1.016-0.3824R_{oc}-0.04082S_{cc07}+0.2075R_{oc}^2-0.007177R_{oc}S_{cc07} \\ & +0.002457S_{cc07}^2-0.03809R_{oc}^3+8.228\times10^{-5}R_{oc}^2S_{cc07}+0.002544R_{oc}S_{cc07}^2 \\ & +6.748\times10^{-5}S_{cc07}^3 \quad \left(0<S_{cc07}\le 4,\ 0<R_{oc}\le 3\right)\end{aligned} \tag{9.4.3}$$

Within the valid ranges of S_{cc07} and R_{oc} values, the coefficient of determination of Equation Figure 9.4.4 is 0.95.

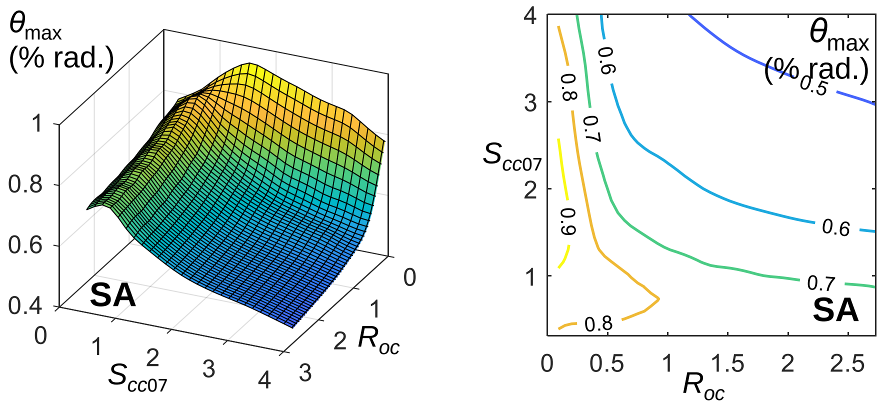


(a)

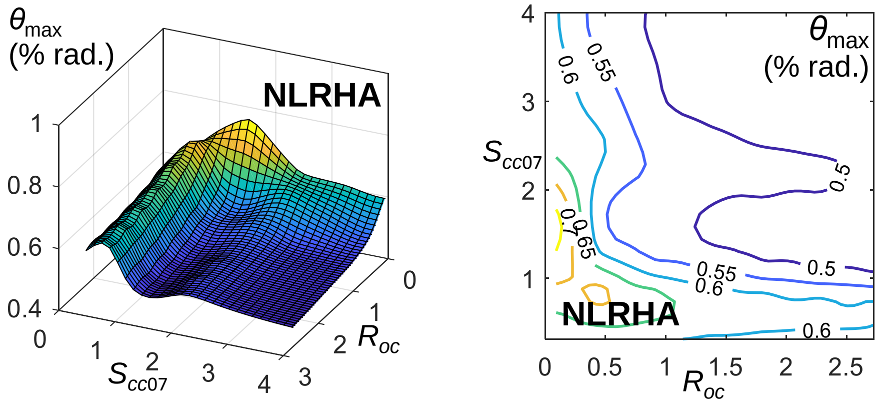


(b)

Figure 9.4.3 Distribution of $\alpha_{opt,\theta}$ with respect to S_{cc07} and R_{oc} calculated from (a) SA and (b) NLRHA



(a)



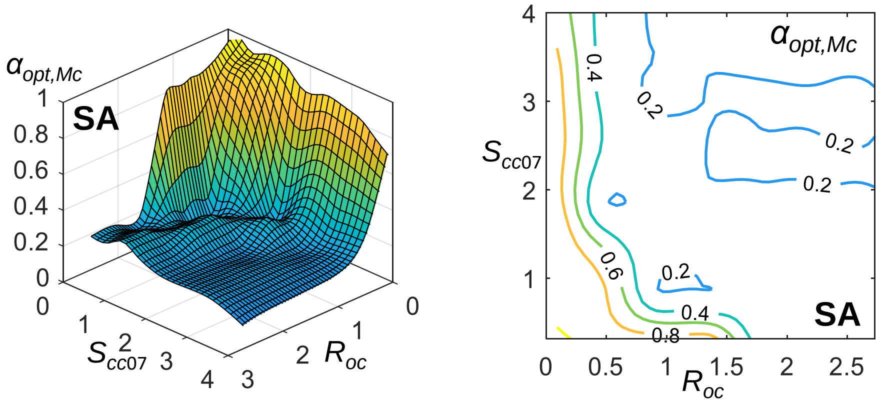
(b)

Figure 9.4.4 Distribution of $\theta_{\max}$ with respect to S_{cc07} and R_{oc} calculated from (a) SA and (b) NLRHA

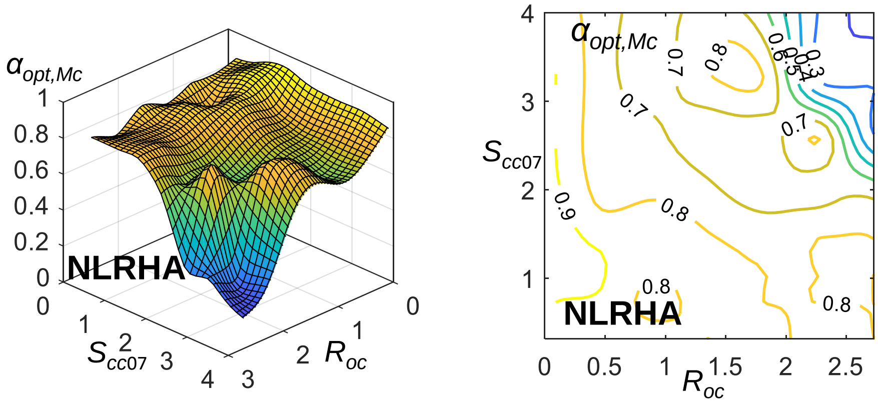
Figure 9.4.5 shows the distributions of outrigger elevation α when $M_{c,\max}$ is minimum ($\alpha_{opt,Mc}$) with respect to R_{oc} and S_{cc07} calculated from SA and NLRHA. The distributions of $M_{c,\max}$ when outrigger locates at $\alpha_{opt,Mc}$ calculated from SA and NLRHA are shown in Figure 9.4.6. The $M_{c,\max}$ calculated from NLRHA is similar to SA. The $M_{c,\max}$ decreases with the increasing S_{cc07} and R_{oc} values. However, the $M_{c,\max}$ stops decreasing when S_{cc07} is greater around 1 to 2 and when R_{oc} is greater than 1. The SA results indicate the $\alpha_{opt,Mc}$ drops to around 0.2 when R_{oc} and S_{cc07} are greater than approximate 0.5. However, the NLRHA results suggest that $\alpha_{opt,Mc}$ drops to around 0.3 when R_{oc} and S_{cc07} are greater than around 2 and 3, respectively. The differences between the SA and NRLHA results could be due to the SA calculation of $M_{c,\max}$ is based on elastic mode shape and linearly elastic force-deformation relation as shown in Equation (4.3.12), and the NRLHA results could be sensitive to different ground motions. However, both the SA and NLRHA show the trend that the $\alpha_{opt,Mc}$ would be lower when the values of S_{cc07} and R_{oc} are larger. Based on the SA results, the relationship between $\alpha_{opt,Mc}$, R_{oc}, and S_{cc07} can be fitted by using the following polynomial:

$$\begin{aligned}\alpha_{opt,Mc}\left(R_{oc},S_{cc07}\right)=&1.583-1.421R_{oc}-0.5673S_{cc07}+0.6559R_{oc}^{2}+0.1473R_{oc}S_{cc07}\\&+0.1358S_{cc07}^{2}-0.1134R_{oc}^{3}+0.00461R_{oc}^{2}S_{cc07}-0.02445R_{oc}S_{cc07}^{2}\\&-0.01047S_{cc07}^{3}\quad\left(0<S_{cc07}\le 4,\ 0<R_{oc}\le 3\right)\end{aligned}\tag{9.4.4}$$

Within the valid ranges of S_{cc07} and R_{oc} values, the coefficient of determination of Equation (9.4.4) is 0.81.

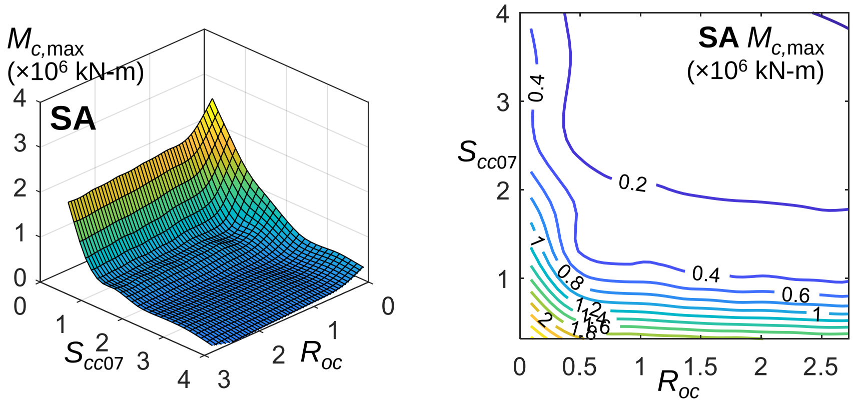


(a)

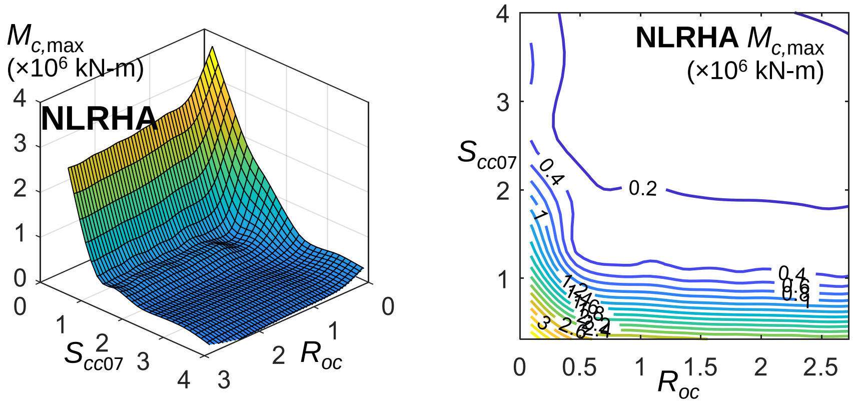


(b)

Figure 9.4.5 Distribution of $\alpha_{opt,Mc}$ with respect to S_{cc07} and R_{oc} calculated from (a) SA and (b) NLRHA



(a)



(b)

Figure 9.4.6 Distribution of $M_{c,\max}$ with respect to S_{cc07} and R_{oc} calculated from (a) SA and (b) NLRHA

9.4.2 Effects of outrigger effect factor and outrigger stiffness ratio

Figure 9.4.7 shows the reductions of $\theta_{\max}$ (D_θ) if compared with the core model without outrigger effect when R_{oc} equals to 0.5, 1, 2, and 3 calculated from SA and NLRHA. The SA and NLRHA results are similar and indicate that the α that best reduces $\theta_{\max}$ is approximately between 0.6 and 0.8 and is lower when the value of R_{oc} is larger (stronger outrigger effect). Under a fixed R_{oc} value, the larger S_{cc07} value leads to greater reductions of $\theta_{\max}$ when α is higher than 0.5. This suggests that when α is lower than 0.5, the $\theta_{\max}$ could not be effectively reduced by increasing the S_{cc07} value. The dashed lines in Figure 9.4.7 show the $\alpha_{opt,T1}$, $\alpha_{opt,heq1}$, $\alpha_{opt,\theta}$ and $\alpha_{opt,Mc}$ calculated from Equations (9.4.1) to (9.4.4). The similar distributions of $\alpha_{opt,T1}$, $\alpha_{opt,heq1}$, and $\alpha_{opt,\theta}$ indicate that the optimal α to minimize outrigger effect, equivalent damping ratio, and to minimize $\theta_{\max}$ are approximate 0.7 to 0.9. In addition, the $\alpha_{opt,T1}$, $\alpha_{opt,heq1}$, and $\alpha_{opt,\theta}$ distributions calculated from SA well match the D_θ distributions calculated from both SA and NLRHA. Figure 9.4.8 shows the reductions of $M_{c,\max}$ (D_{Mc}) if compared with the core structure without outrigger effect when R_{oc} equals to 0.5, 1, 2, and 3. Both the SA and NLRHA suggest that the larger S_{cc07} value results in a greater reduction in $M_{c,\max}$. The α that results in the greatest reduction in $M_{c,\max}$ calculated from SA is lower than the one calculated from NLRHA. However, the $\alpha_{opt,Mc}$ could still provide satisfactory outrigger elevation in order to reduce the $M_{c,\max}$ response according to the NLRHA results. In addition, as shown in Figure 9.4.8b, the D_{Mc} is also close to its minimum value along with the $\alpha_{opt,\theta}$, $\alpha_{opt,heq1}$, and $\alpha_{opt,T1}$ distributions. Based on the analysis results, both the D_{θ} and D_{Mc} responses indicate that the BRB-outrigger with larger R_{oc} value offers a stiffer outrigger (stronger outrigger effect) and results in smaller seismic response. However, the reductions of seismic response are not proportional to the increasing R_{oc}. For example, if compare the cases when R_{oc} increases from 1 to 2, the required outrigger stiffness k_{og} is doubled, however, the D_{θ} and D_{Mc} are increased by approximate 5% only. Therefore, in order to efficiently reduce seismic response by utilizing BRB-outrigger, to select an α approximate between 0.6 to 0.8, and an S_{cc07} value greater than 1 would be more efficient than increasing R_{oc}. In summary, based on the analysis results, the optimal α is around 0.6 to 0.8, which is similar to the results described in the previous sections, and the recommended values of S_{cc07} and R_{oc} are greater than 1 and around 0.5 and 1, respectively, in order to efficiently mitigate seismic response. Figure 9.4.7 and Figure 9.4.8 can be used as design charts to assist the designer to select outrigger elevation and the outrigger stiffness ratio in order to achieve the desired seismic response at the preliminary design stage.

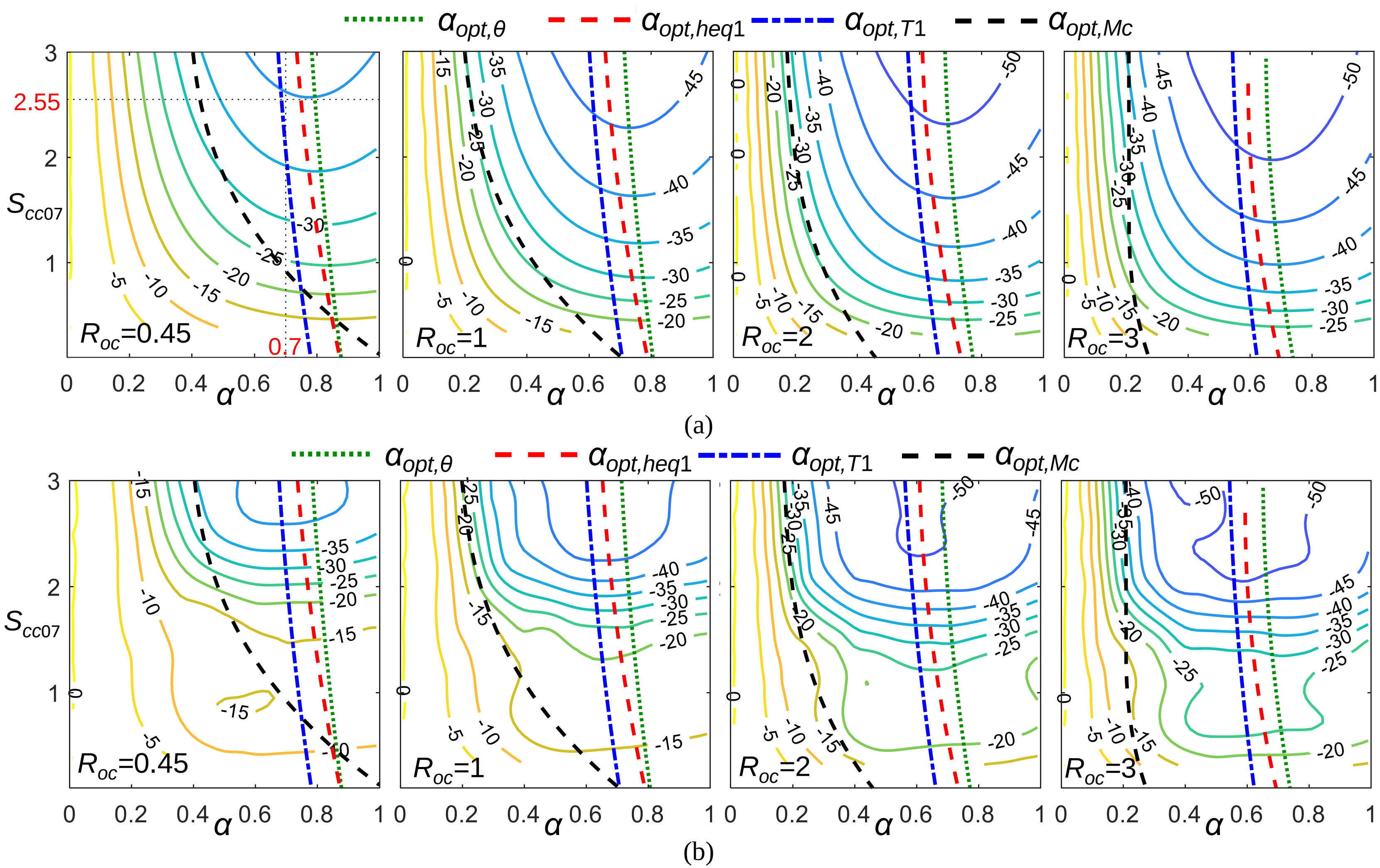
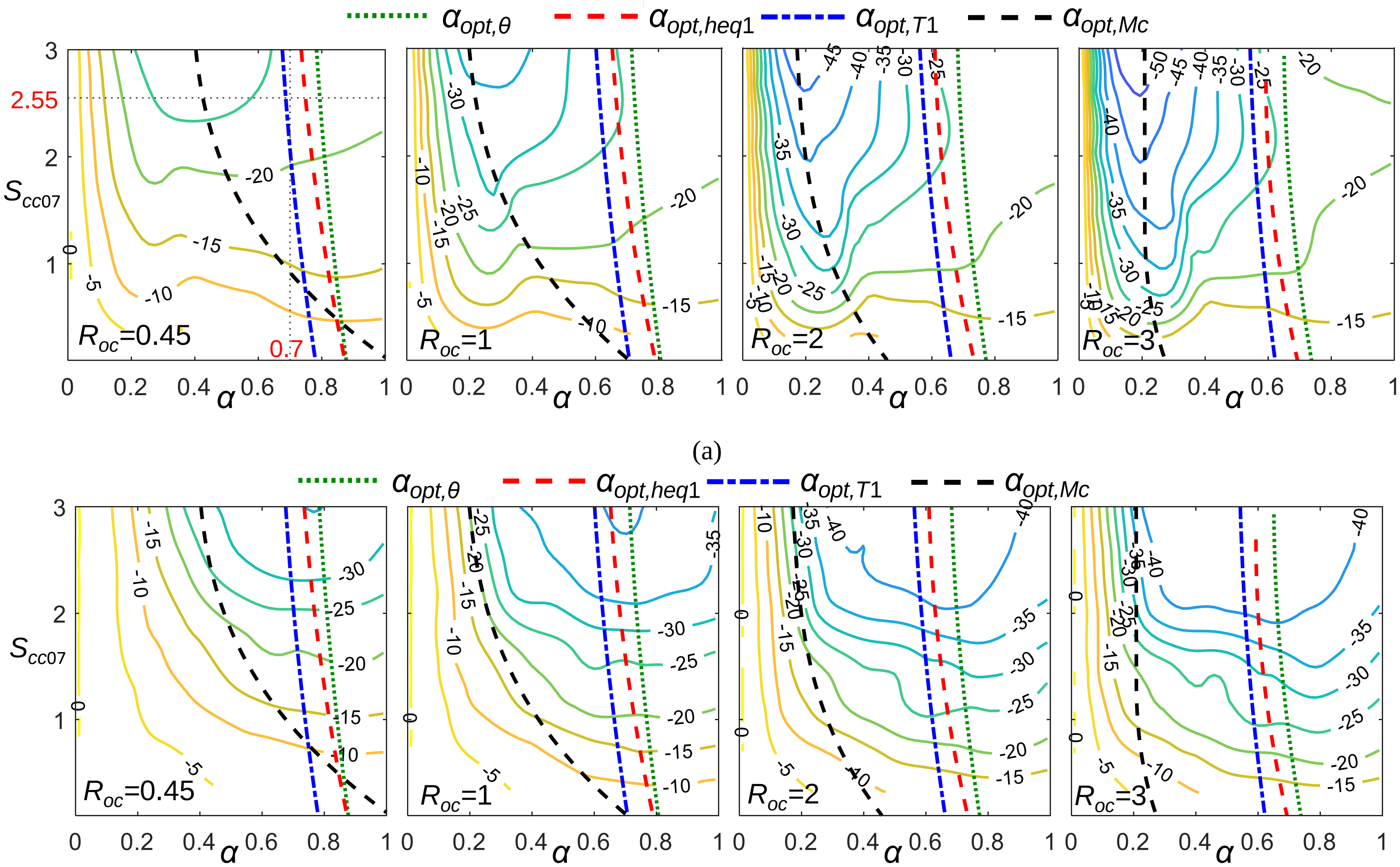


Figure 9.4.7 The D_θ distribution calculated from (a) SA and (b) NLRHA



(a)

(b)

Figure 9.4.8 The D_{Mc} distribution calculated from (a) SA and (b) NLRHA

9.5 DESIGN EXAMPLES

9.5.1 Introduction of the example models

The 40-story model (h=160m) introduced in Chapter 8 is used to demonstrate as design examples for structures with OB, BT, and GB outriggers. The perimeter column size is Box 900×900×75 mm. The value of k_c is 309375 kN/m, and the outrigger effect factor S_{cc07} of 2.25 can be calculated from Equation (9.3.1). Based on the analysis results, the value of R_{oc} is set to be approximate 0.45, and the α is set at 0.7. Therefore, the required outrigger stiffness (k_{og}) is approximate 139219 kN/mm and the required $u_{d,y}$ is approximate 16 to 18 mm. As shown in Figure 9.4.7a and Figure 9.4.8a, the case of R_{oc}=0.45 and S_{cc07}=2.55 when α=0.7 suggests that the reductions in $\theta_{\max}$ and $M_{c,\max}$ are approximate 39% and 23%, respectively. With this condition, the seismic performance of structures with different BRB-outrigger configurations of OB, BT, and GB are compared.

Figure 9.5.1a shows the design detail of the OB outrigger. The top and bottom chords of the outrigger truss locate at the 28th and 27th floors, respectively. The two ends of the braces and column members in the outrigger truss are designed with moment connection detail. The connections between the top and bottom chords to the core structure are rigid connections. The top chord end near the perimeter column connections with the BRB, which is arranged vertically with a length of 8 m. The bottom end of the BRB connects to the perimeter column at the 26th floor. Both the two ends of the BRB are pinned-connections. The 26th floor beam is spliced adjacent to the lower BRB end. The value of k_t is 187987 kN/m, which is calculated by using OpenSees. Table 9.5.1 shows the detail of the BRB design in the OB outrigger configuration (BRB_OB). Figure 9.5.1b shows the detail of the OB outrigger in the MBM model. The members in the outrigger truss are modeled by using beam column element. The BRB_OB, which is modeled by using truss element, connects to the outrigger truss end as Node A and to the perimeter column at Node C. The perimeter column at the 28th floor level is separated at Node B, which shares the same coordinate with Node A, but moves independently from Node A. The bilinear material model with a post-yield stiffness ratio of 0.01 was used for all the outrigger truss and BRB members.

Table 9.5.1 The parameters used in DM model for each outrigger configuration

		BRB_OB	BRB_BT	BRB_GB
core material		SN490	SN490	SM570
cross-sectional area (mm^2)	core	19500	22275	27600
	transition	30500	34925	40080
	joint	41500	47575	52560
segment length (mm)	core	6500	2500	6500
	transition	110	115	104
	joint	640	1135	3846
axial stiffness (kN/m)		538518	1200962	516737
yield deformation (mm)		11.8	6.0	22.4
yield axial force (kN)		6338	7239	11592

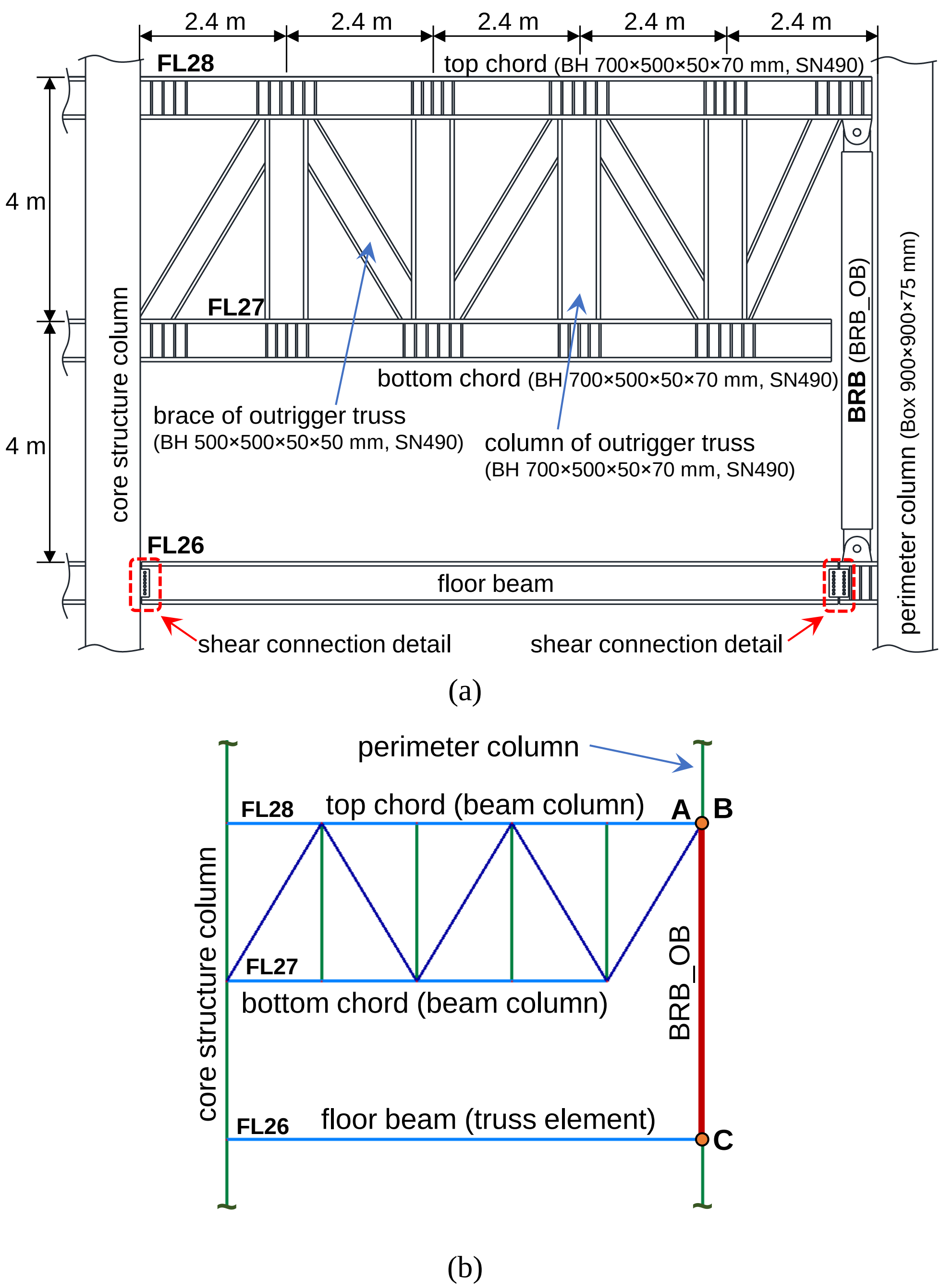
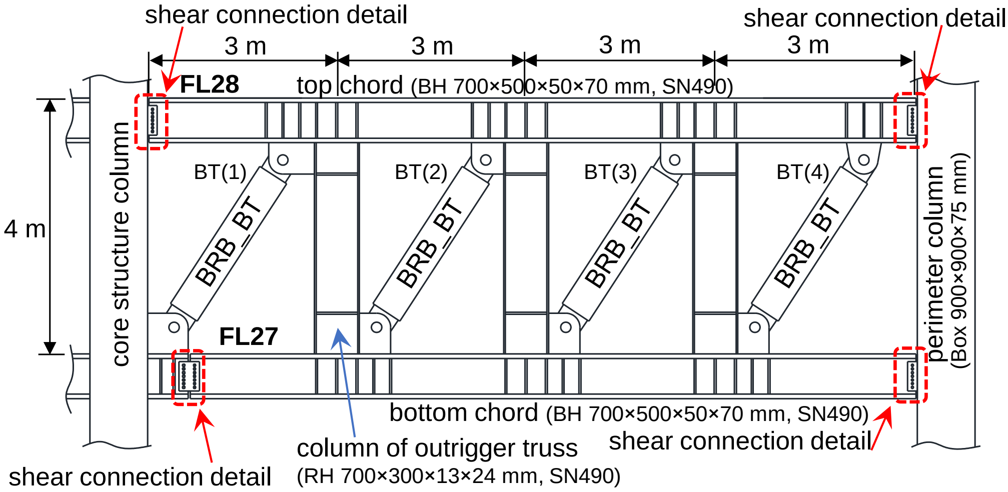
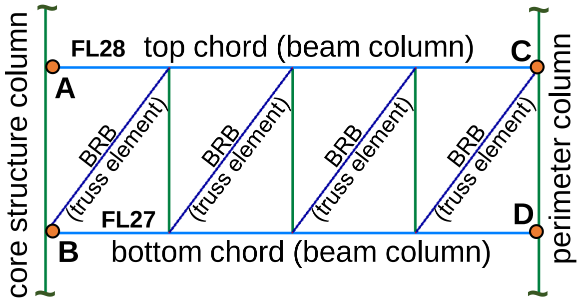


Figure 9.5.1 The (a) design detail and the (b) MBM model of the OB outrigger

Figure 9.5.2a shows the detail of the BT outrigger. The top and bottom chords with both ends of shear connection detail located at the 28th and 27th floors, respectively. The member size of the top and bottom chords are the same as the OB outrigger (BH 700×500×50×70 mm), but the column size in the outrigger is designed to be smaller (RH 700×300×13×24 mm). The two ends of the outrigger truss columns are designed with moment connection detail. Four identical BRBs (BRB_BT), are arranged along the BT outrigger with an equal span of 3 m as shown in Figure 9.5.2a. The design detail of the BRB_BT is shown in Table 9.5.1. Figure 9.5.2b shows the BT outrigger in the MBM model. The top and bottom chords are modeled using beam column elements. The ends at Node A, B, C, and D are free to rotate about the out-of-plane direction by using the equalDOF command in the OpenSees. The ideal inelastic behavior of the BR outrigger is to let the BRB yield first and dissipate the majority of the input seismic energy. Slight inelastic deformations in the top and bottom chords and the columns in the BT outrigger would be permitted. The bilinear material model with a post-yield stiffness ratio of 0.01 was used for all the elements in the BT outrigger.



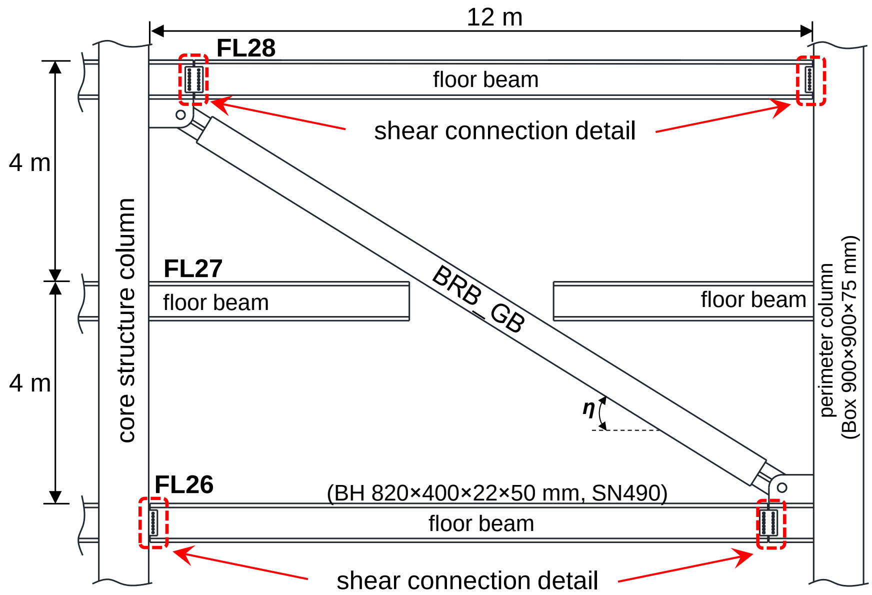
(a)



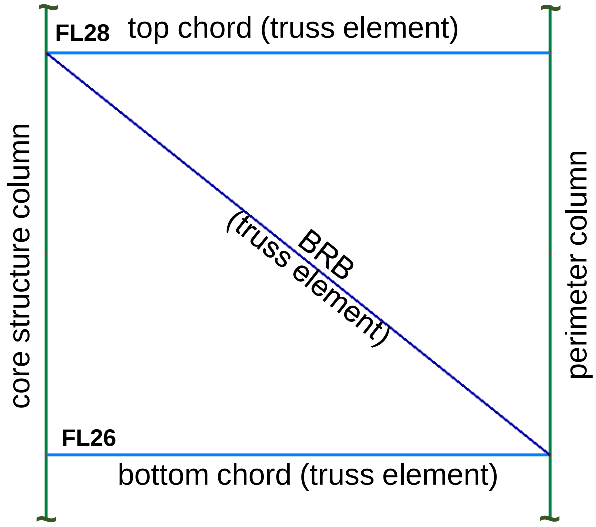
(b)

Figure 9.5.2 The (a) design detail and the (b) MBM model of the BT outrigger

Figure 9.5.3a shows the detail of the GB outrigger. The top and bottom ends of the BRB (BRB_GB) connect to the core structure at the 28th floor and the perimeter column at the 26th floor, respectively. Both the BRB_GB ends are pinned connections. The floor beams at the 26th and 28th floors are spliced adjacent to the BRB connections. The detail of the BRB_GB design is shown in Table 9.5.1. As the floor beam at the 26th floor is required to sustain the maximum axial force developed in the BRB_GB, had has to be sufficiently stiff in order to prevent excessive axial deformation. The axial force demand for the 26th floor beam is estimated as 15861 kN (=N_y×1.1×1.3×1.15×cosη, where N_y is the axial yield force of the BRB_GB, and the parameters 1.1, 1.3, and 1.15 are the factors to account for material overstrength, strain hardening, and compression strength adjustment, respectively). It is assumed that sufficient lateral support is provided on the 26th floor beam. Therefore, the size of BH 820×400×22×50 mm, made by SN490 grade steel, can be designed with a compression DCR of 0.97. The axial stiffness of the 26th floor beam is 2.6 times of the axial stiffness of BRB_GB in the horizontal direction. Thus, it should be sufficiently stiff to prevent excessive axial deformation. Figure 9.5.3b shows the GB outrigger in the MBM model. The 27th floor beam is not included. Both the BRB and floor beams are modeled by using truss elements. The bilinear material model with a post-yield stiffness ratio of 0.01 was used for the BRB member.



(b)



(b)

Figure 9.5.3 The (a) design detail and the (b) MBM model of the GB outrigger

9.5.2 Seismic response of the example models

Figure 9.5.4a shows the relationship between the vertical force applied at the outrigger end (P_v) and the corresponding vertical deformation (u_v) in each outrigger configuration calculated from the vertical pushover analysis by using OpenSees. Figure 9.5.4b illustrates the vertical pushover analysis of each configuration. Based on the analysis results, the elastic outrigger stiffness (k_{og}) of the OB, BT, and GB outriggers are 139344, 132384, and 159077 kN/m, respectively. The vertical deformation when BRB yields are 46, 51, and 40 mm for the OB, BT, and GB outriggers, respectively. Figure 9.5.4a and Figure 9.5.4c show the sequence of BRB yielding and the flexural plastic hinges form in the outrigger truss columns. The BRB1 and BRB4 (Figure 9.5.4c) yield first when the vertical deformation (u_v) reaches 51 mm (0.4% rad. deflection). The BRB2 and BRB3 yield when u_v reaches 90 mm (0.8% rad. deflection). The flexural plastic hinges form at the two ends of outrigger truss columns when u_v reaches 120 and 159 mm (1% and 1.3% rad. deflection), respectively. The post-yield stiffness of the OB outrigger is slightly larger than the others. However, as the elastic stiffness and the yield deformation of OB, BT, and GB outriggers are similar, it is anticipated that the structure with the different outrigger configurations would exhibit similar seismic response. Table 9.5.2 shows the parameters used in the DM model for the structure with different outrigger configurations. The value of 9×10^9 is used for the k_t in the DM model for the BT and GB outrigger configurations.

Table 9.5.3 shows the first four mode vibration periods calculated from using DM and MBM models. The vibration periods calculated from using DM model are slightly larger than MBM model. Figure 9.5.5, Figure 9.5.6, and Figure 9.5.7 show the roof drift histories of the structures with OB, BT, and GB outriggers, respectively, calculated by using DM and MBM models under the originally observed ground motions. The roof drift history results obtained from using DM and MBM models are close to each other. In addition, the roof drift responses between the three structures with different outrigger configurations are similar to each other. The differences between the analysis results calculated by using DM and MBM models could due to that the span of the core structure is not included in the DM model, the cantilever column in the DM model could not perfectly resemble the braced core structure in the MBM model, and the different distribution of mass along the building height in the DM and MBM model. The modal analysis and roof drift history results suggest that the DM model with modified parameters shown in Table 9.2.1 could be used to model the structure with BT and GB outrigger configurations. In addition, the structures with different BRB-outrigger configurations but sharing the same S_{cc07} and R_{oc} values exhibit very close seismic response. This indicates that the proposed parameters (S_{cc07} and R_{oc}) could effectively reflect seismic performance for a structure when any one of the OB, BT, or GB outrigger configuration is adopted.

Table 9.5.2 The parameters used in DM model for design example with different outrigger configurations

configuration	k_{og} (kN/m)	k_t (kN/m)	k_d (kN/m)	$u_{d,y}$ (mm)	p
OB outrigger	139344	187987	538518	11.8	0.01
BT outrigger	132384	∞	132384	51.0	0.01
GB outrigger	159077	∞	159077	40.4	0.01

Table 9.5.3 The vibration periods calculated by using DM and MBM models for the design example with different outrigger configurations

configuration	model	vibration period (sec)			
		1st mode	2nd mode	3rd mode	4th mode
OB outrigger	DM	4.383	0.840	0.316	0.160
	MBM	4.252	0.880	0.344	0.178
BT outrigger	DM	4.410	0.841	0.316	0.160
	MBM	4.390	0.890	0.344	0.178
GB outrigger	DM	4.312	0.835	0.316	0.160
	MBM	4.184	0.880	0.344	0.178

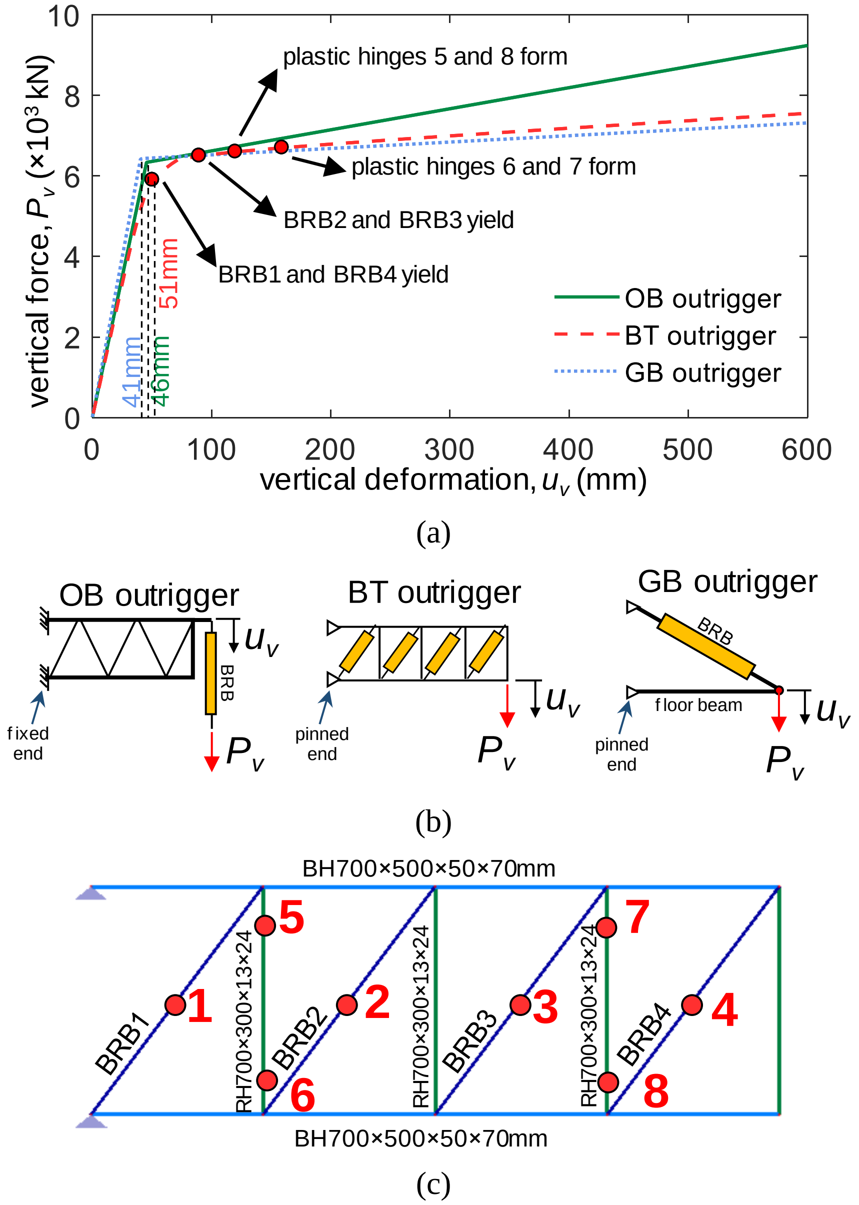


Figure 9.5.4 The (a) vertical force and deformation relationship, the (b) illustrations for performing vertical pushover analysis for the OB, BT, and GB outriggers, and (c) illustration of plastic hinge locations of the BT outrigger.

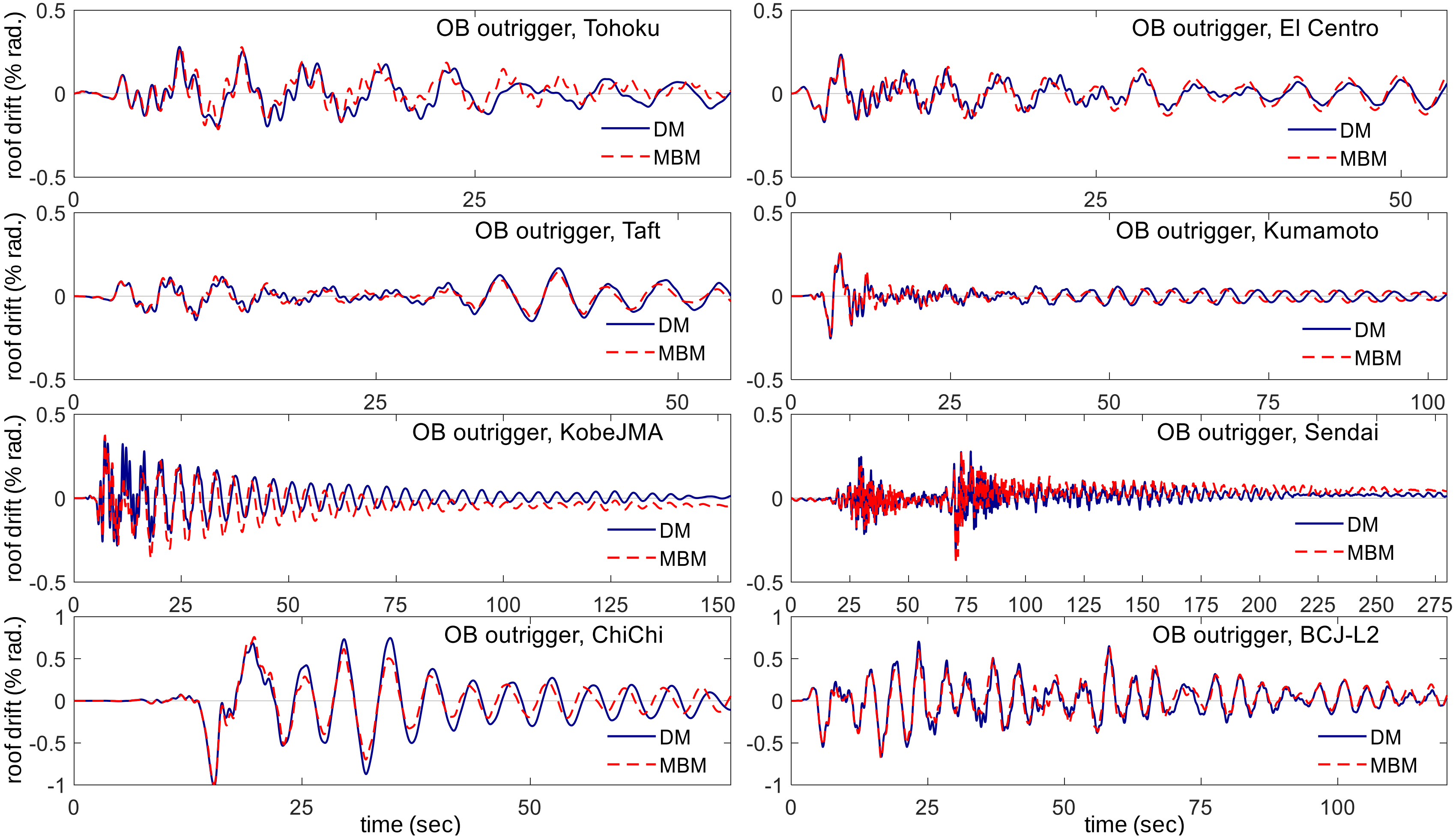


Figure 9.5.5 Roof drift history of the models with OB outrigger calculated by using DM and MBM models

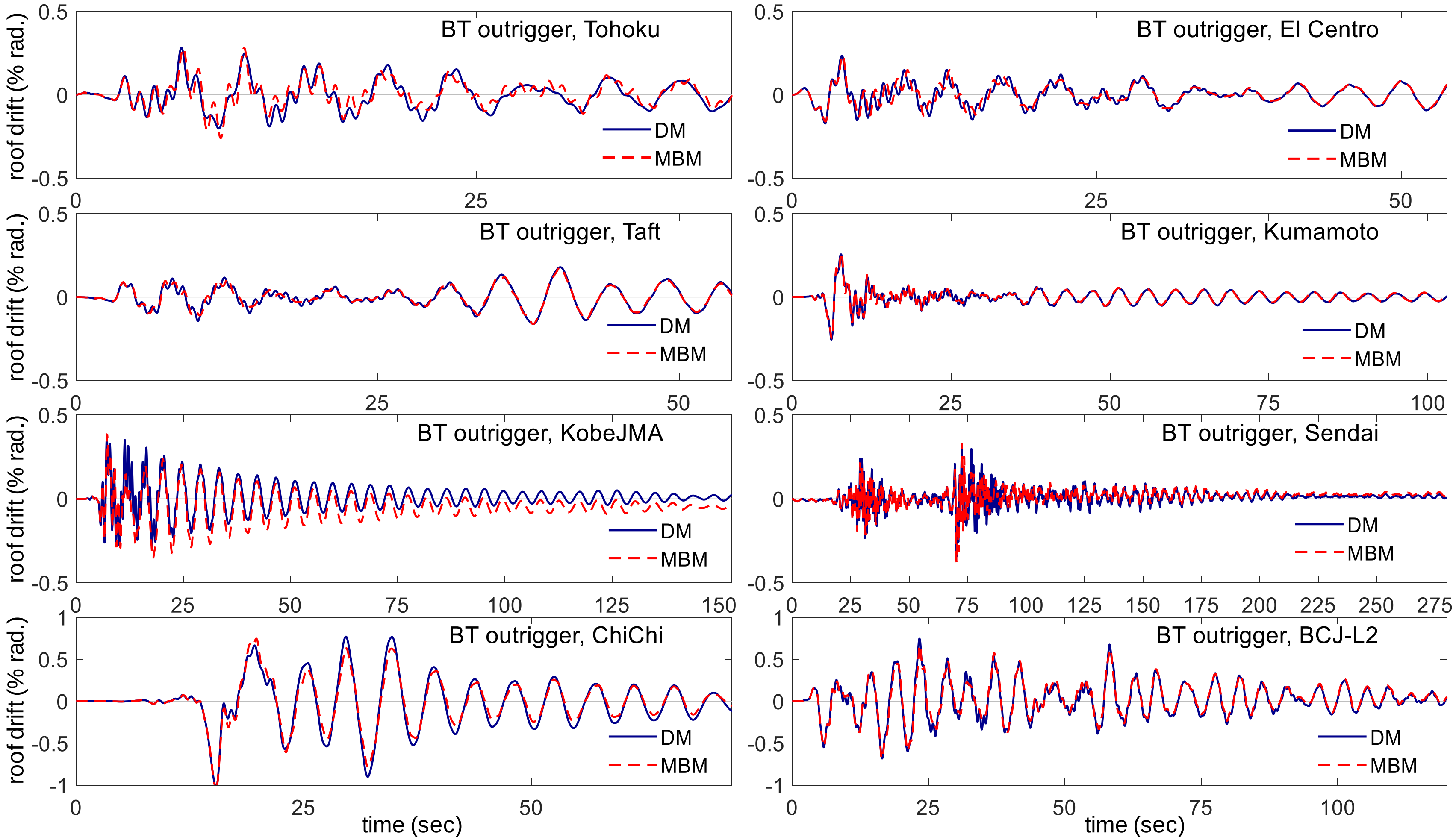


Figure 9.5.6 Roof drift history of the models with BT outrigger calculated by using DM and MBM models

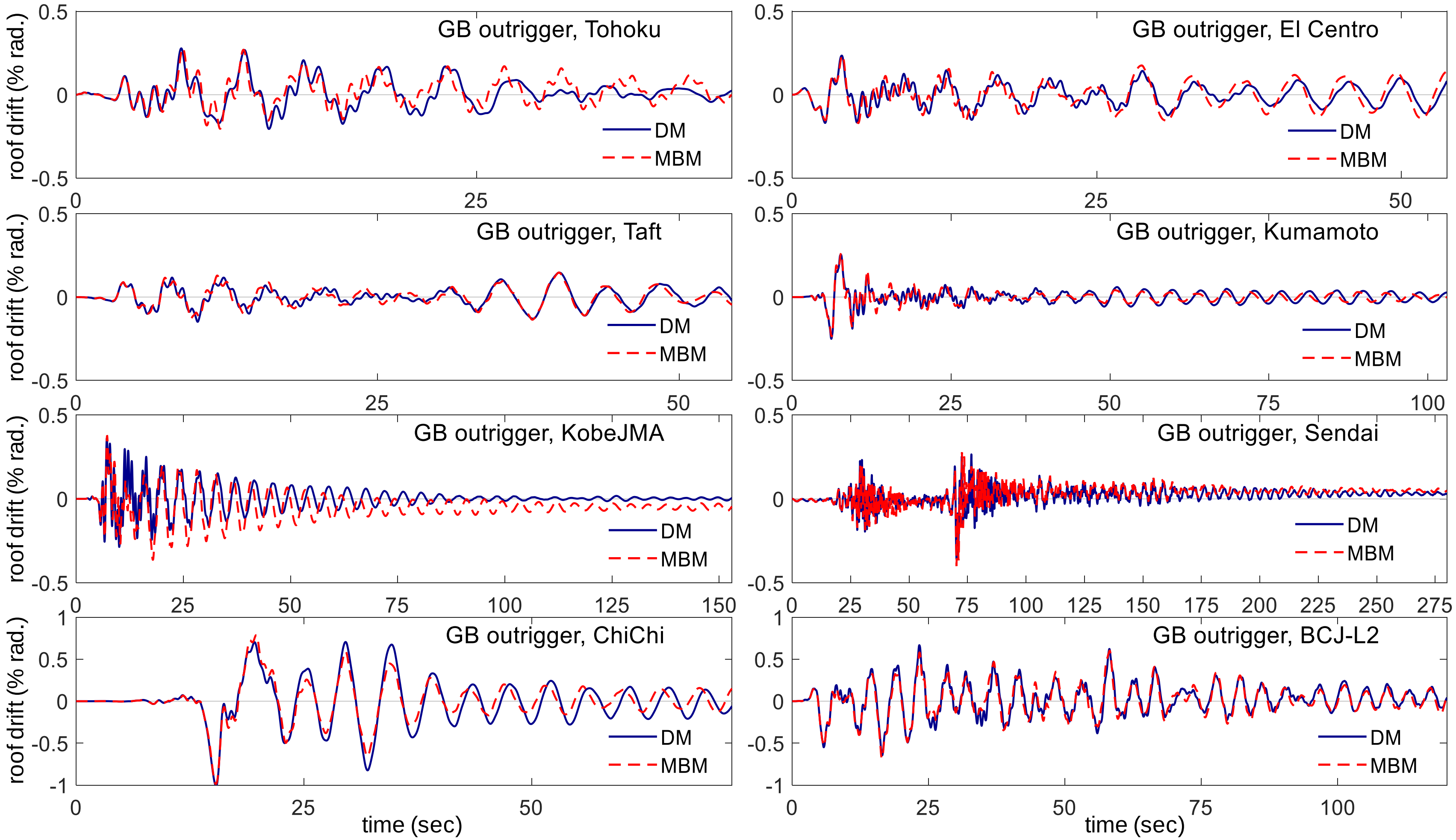


Figure 9.5.7 Roof drift history of the models with GB outrigger calculated by using DM and MBM models

Figure 9.5.8 and Figure 9.5.9 show the maximum lateral deformation and inter-story responses under all the 8 original observed ground motions, respectively. Both the analysis results calculated from using DM and MBM model show that all the three BRB-outrigger could effectively mitigate the lateral deformation and inter-story responses. Figure 9.5.10, Figure 9.5.11 and Figure 9.5.12 show the maximum roof drift ($\theta_{\max}$), the maximum overturning moment at core structure base ($M_{c,\max}$) , and the maximum perimeter column axial force ($C_{1,\max}$) calculated from NLRHA with the original observed ground motions by using DM and MBM models, respectively. Figure 9.5.13 shows the relationships between normalized axial force and core strain for the BRB members in OB, BT, and BT outriggers calculated from using MBM model. The labels for the four BRBs in BT outrigger are shown in Figure 9.5.2b. Figure 9.5.14 shows the cumulative plastic deformation ratio (R_{CPD}) for the BRB in each outrigger configuration. The zero values of R_{CPD} indicate the BRB deform elastically. The analysis result shows that the $\theta_{\max}$ and $M_{c,\max}$ responses between the structures with OB, BT, and GB outrigger configurations are only slightly different. The models without outrigger (Core) generally exhibit greater $\theta_{\max}$ and $M_{c,\max}$ than the models with BRB-outrigger. However, under the Tohoku, El Centro, Taft, and Kumamoto ground motions, the BRB-outrigger only slightly improves the seismic response if compared with the Core model. This is because that the increased stiffness resulted from the outrigger effect might increase the seismic demand, and the BRB deforms elastically (R_{CPD}=0 as shown in Figure 9.5.14) or exhibits only slight inelastic deformation (Figure 9.5.13) and thus results in a low energy dissipation efficiency. Figure 9.5.15 show the percentages of energy dissipated by the BRB (E_{BRB}) to the total input energy. The reductions of the $\theta_{\max}$ and $M_{c,\max}$ are greater when the value of E_{BRB} is larger. The reductions in $\theta_{\max}$ (the average of OB, BT, and GB outriggers) if compared to the Core model are approximate 27% and 50% under the ChiChi and BCJ-L2 ground motions, respectively. In addition, the reductions in $M_{c,\max}$ (the average of OB, BT, and GB outriggers) if compared to the Core model are approximately 40% and 30% under the ChiChi and BCJ-L2 ground motions, respectively. According to Figure 9.4.7 and Figure 9.4.8, the SA results indicate that the reductions in $\theta_{\max}$ and $M_{c,\max}$ are around 39% and 23%, respectively. This suggests that the SA could provide appropriate estimations on seismic response of a structure with BRB-outrigger if the BRBs develop sufficient hysteretic responses. As shown in Figure 9.5.14, the R_{CPD} values also indicate the ductility demand for the BRB. The

BRB_OB exhibits the largest R_{CPD} value as the vertical BRB arrangement imposes a large amount of axial deformation demand on the BRB_OB. The R_{CPD} values of the BRB_GB are smaller than the BRB_OB. However, the seismic response and E_{BRB} of the models with OB and GB configuration are similar. This suggests that the GB outrigger configuration could be a better alternative in order to prevent excessive ductility demand on the BRB, as the BRB with too large R_{CPD} value could easily fracture before the end of an earthquake. For the BT outrigger, because of the outrigger arrangement, the ductility demands for the BRB near the two outrigger truss ends (BRB_BT(1) and BRB_BT(4)) are greater than the BRB in the mid-span of outrigger truss (BRB_BT(2) and BRB_BT(3)). In the design practices, the sizes of BRB in the BT outrigger configuration could be properly adjusted in order to reduce steel usage. For instance, the BRB_BT(2) and BRB_BT(3) with low R_{CPD} values in the design example could be replaced by ordinary elastic steel braces.

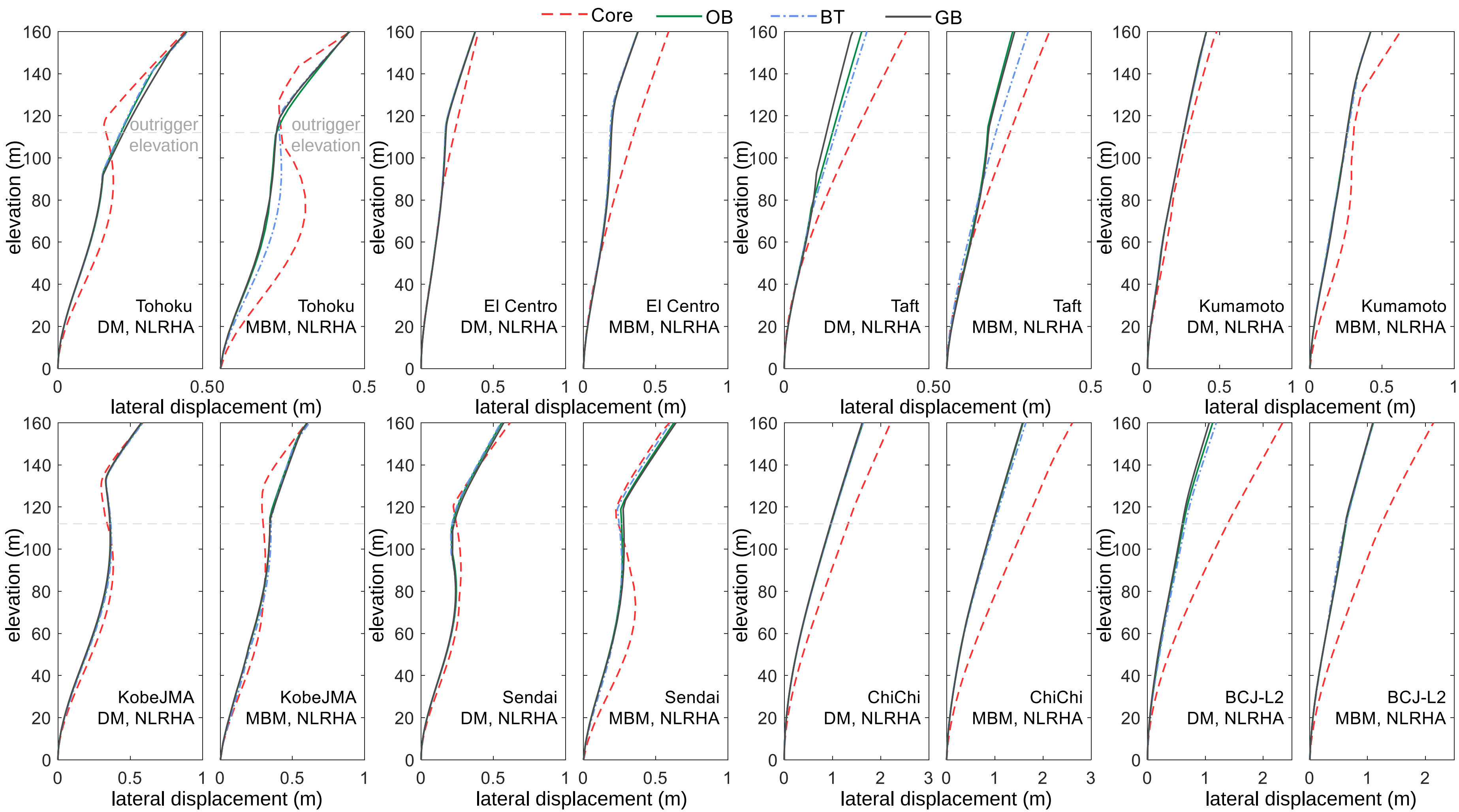


Figure 9.5.8 Roof drift history of the models with GB outrigger calculated by using DM and MBM models

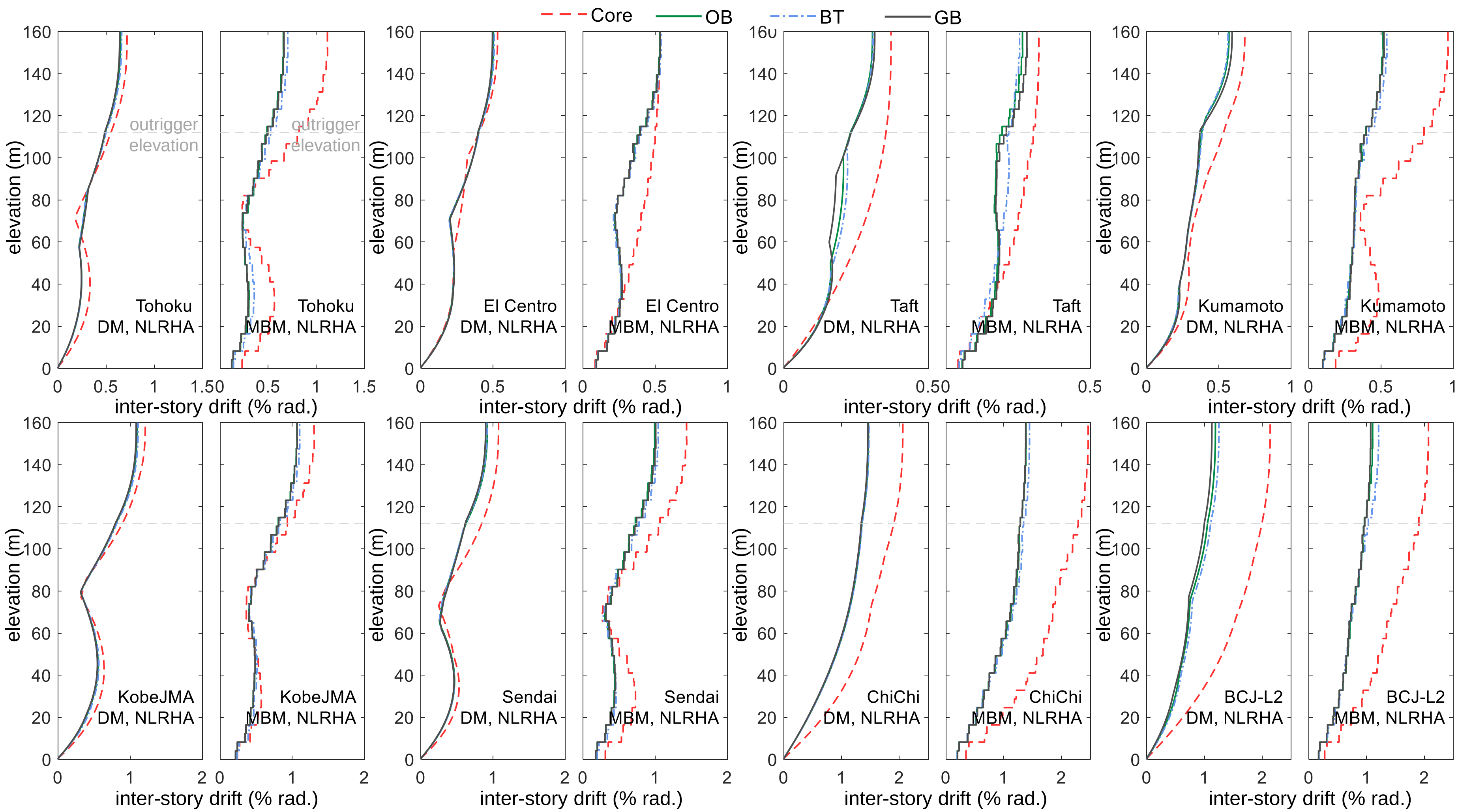


Figure 9.5.9 Inter-story drift history of the models with GB outrigger calculated by using DM and MBM models

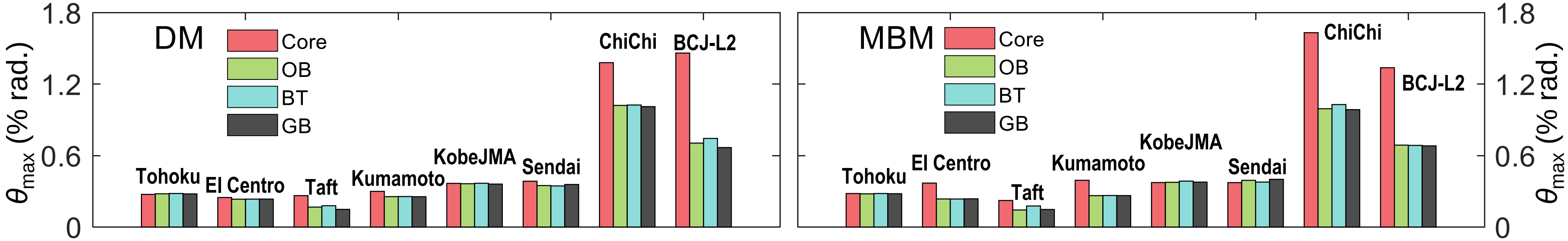


Figure 9.5.10 The θ_{max} responses calculated from NLRH with original observed ground motions

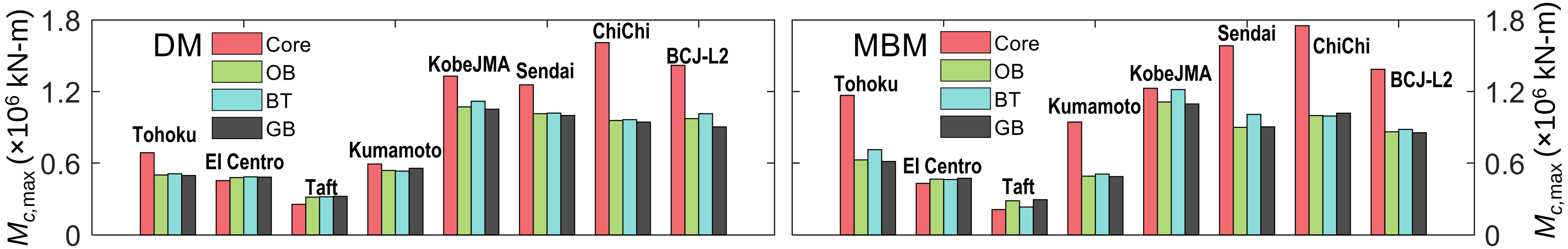


Figure 9.5.11 The $M_{c,max}$ responses calculated from NLRH with original observed ground motions

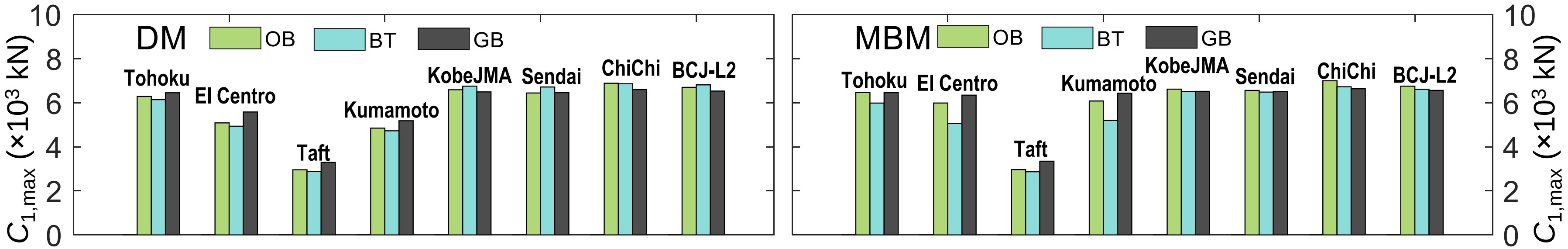


Figure 9.5.12 The $C_{1,max}$ responses calculated from NLRHA with original observed ground motions

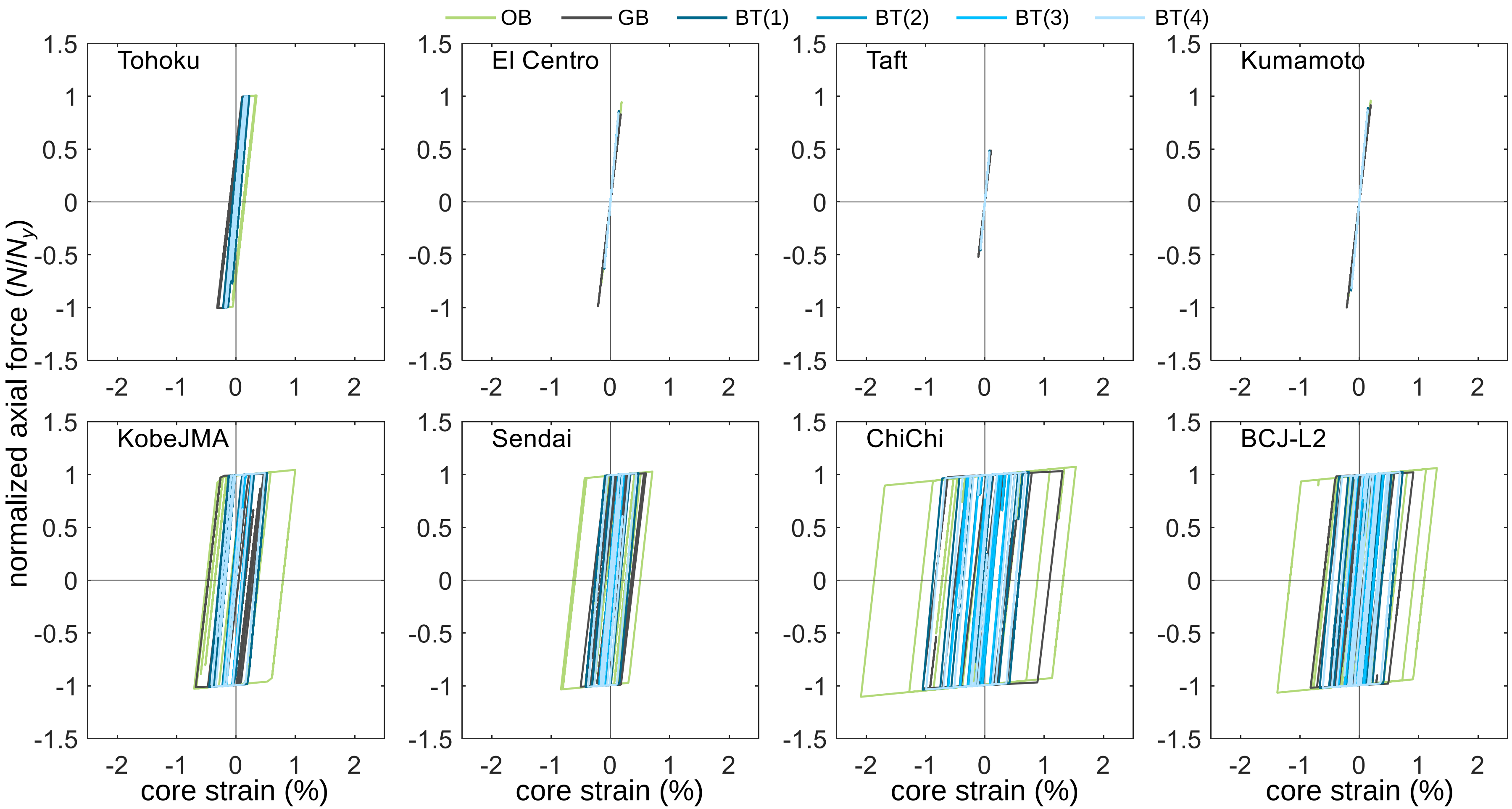


Figure 9.5.13 The relationships between normalized axial force and core strain of the BRBs in OB, BT, and GB outriggers calculated from NLRHA

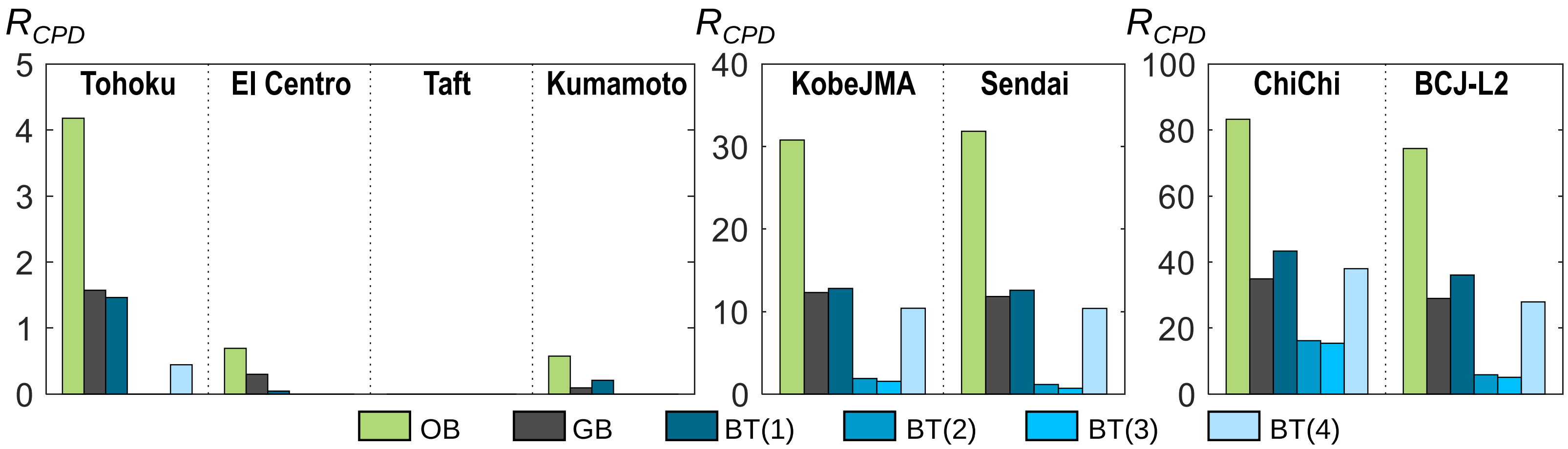


Figure 9.5.14 The R_{CPD} calculated from NLRHA with original observed ground motions

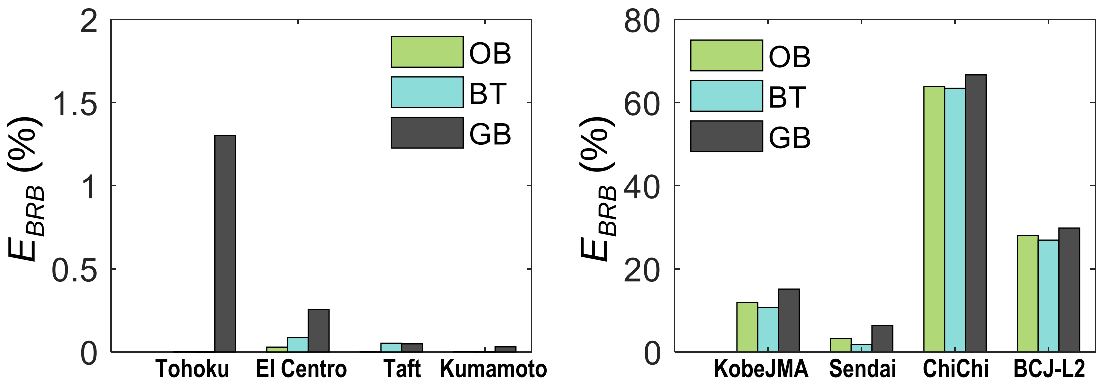


Figure 9.5.15 The E_{BRB} calculated from NLRH with original observed ground motions

9.5.3 Comparison between OB, BT, and GB outrigger configurations

Table 9.5.4 shows the steel usage of the OB, BT, and GB outriggers (single span within the core structure and perimeter column) for the design example models. The weight of the 26th floor beam in the GB outrigger, which is designed to sustain the maximum force developed by the BRB, is included. The OB outrigger consumes the heaviest steel. Although the force demands for the OB outrigger truss members can be effectively limited by the maximum force capacity of the BRB_OB, the OB outrigger truss member sizes could be determined in order to create sufficient k_t value, instead of determined by force demands. In addition, when the outrigger span (l_t) becomes longer, the required outrigger truss member size must be sharply increased. Therefore, the OB outrigger configuration would be suitable when the outrigger truss span is short. However, the OB outrigger truss can be designed to occupy only one story and only one BRB is required. The BT outrigger configuration requires more than one BRBs. Slight plastic deformations are allowed in the BT outrigger truss members in order to prevent too large member size. Therefore, the design of the BT outrigger configuration is more flexible but more complicated than the OB outrigger configuration. In addition, as the BRB and outrigger truss act in parallel, the post-yield stiffness of the BT outrigger could be properly adjusted by selecting different sizes of the outrigger truss member. The larger post-yield stiffness ratio would be beneficial, as it avoids the sudden stiffness drops such as the OB and GB outrigger configurations when BRB yields. The GB outrigger consumes the smallest amount of steel. However, the GB outrigger requires a very long BRB when the l_t is long, and the BRB_GB may be required to span more than two-story height in order to generate sufficient outrigger stiffness (k_{og}), which could reduce usable floor area. Based on the analysis results, all three outrigger configurations could achieve a satisfactory seismic response and could be selected by designers to fit individual architecture requirements by using the proposed indexes and design charts.

Table 9.5.4 Steel usage for the OB, BT, and GB outriggers with single outrigger span

configuration	outrigger truss (tonf)	BRB (tonf)	total (tonf)
OB outrigger	43.6	3.8	47.4
BT outrigger	20.6	11.2 (2.8 tonf for each BRB_BT)	31.8
GB outrigger	5.3 (including the 12m-long floor beam)	9.2	14.5

9.6 SUMMARY

In this chapter, three different BRB-outrigger configurations were proposed, and their seismic performances are discussed. The design indexes and modified parameters are proposed in order to use the DM and UM models to analyze the structure with BT and GB outriggers. The summaries of this chapter are as follows:

(1) For a building incorporating one layer of BRB-outrigger with either OB, GB, or BT outrigger configuration, the optimal outrigger elevation (α) should be approximate 0.6 to 0.8.

(2) The outrigger effect factor S_{cc07} can be used to indicate the efficiency of utilizing the BRB-outrigger with either OB, BT, or GB outrigger configuration as seismic resistance system to improve seismic performance. The larger S_{cc07} value suggests the efficiency in mitigating seismic response by the BRB-outrigger is higher. Based on the analysis results, the recommended value of S_{cc07} should be larger than 1.0.

(3) When the outrigger locates at approximate α=0.6 to 0.8, and when the values of S_{cc07} and R_{oc} are greater than 1.0 and 0.5, respectively, both the maximum roof drift ratio and the maximum overturning moment at core structure base can be reduced by approximately 20% to 30%, if the BRBs in either OB, BT, or GB outrigger configuration develop full hysteretic responses.

(4) From the viewpoint of BRB design, the OB and GB outrigger configurations are suitable when the outrigger span is short. The OB and BT outrigger configurations best utilize the building interior space. The GB outrigger could be the most economical solution as the outrigger truss members are not necessary. All the three BRB-outrigger configurations are capable to achieve the wanted seismic response. The designers can select suitable BRB-outrigger configurations to fulfill both architectural requirements and economical designs using the proposed design charts and indexes.

10

APPLICATION OF BRB-OUTRIGGER

CHAPTER CONTENTS

10.1 Introduction 10-3
10.2 Inconsistent perimeter column axial stiffness 10-3
10.2.1 Design examples 10-5
10.3 Linearly changed core rigidity and perimeter column axial stiffness...10-12
10.3.1 Single BRB-outrigger system 10-13
10.3.2 Dual BRB-outrigger system...10-18
10.4 Summary..10-30

10.1 INTRODUCTION

In the previous chapters, it is assumed that the perimeter columns on two sides of the structure are identical to each other, and the core structure flexural rigidity (*EI*) is constant along with the building height. However, those simplifications may be different from the cases for real buildings. This chapter discusses the effects when the perimeter column has different axial stiffness in tension and compression, and when *EI* varies along with the building height on the optimal design for structure with BRB-outrigger system. The modifications of the proposed analysis method are introduced.

10.2 INCONSISTENT PERIMETER COLUMN AXIAL STIFFNESS

While the axial tensile and compressive stiffness of the perimeter columns on the two sides of the building are different, the proposed analytical model has to be modified. This occurs when the perimeter column is a concrete filled tube (CFT) column. As the infill concrete does not develop tensile strength, the perimeter column develops different axial stiffness when it is subjected to tension and compression. During small earthquakes, because of the gravity load demands, the perimeter columns could be subjected to compression only. However, when the gravity load demands are small, or during large earthquakes, the perimeter column could be subjected to tension, and develops different tensile and compressive axial stiffness. Figure 10.2.1 shows a structure with *n* layers of BRB-outrigger. When the gravity loads are ignored, if the core structure deformed toward the right and in the first mode deformed shape, the left and right perimeter columns are in tension and compression, respectively. If $k_{c,c}$ and $k_{c,t}$ are axial compressive and tensile stiffness of the perimeter columns with a height of *h*, respectively, the Equation (3.2.2) to Equation (3.2.4) are modified as follows:

$$\frac{1}{h}\mathbf{M_o} = \begin{bmatrix} M_{o1}/h \\ M_{o2}/h \\ \vdots \\ M_{on-1}/h \\ M_{on}/h \end{bmatrix} = \frac{l_t}{h^2}\left[\mathbf{C_c}\mathbf{D_c}^{-1} + \mathbf{C_t}\mathbf{D_t}^{-1}\right]\boldsymbol{\theta} h = \frac{1}{h}\mathbf{k_g^{'}}\boldsymbol{\theta} \qquad (10.2.1)$$

$$\mathbf{C_c}=\begin{bmatrix} k_{c,c1}+k_{c,c2} & -k_{c,c2} & \cdots & 0 & 0 \\ -k_{c,c2} & k_{c,c2}+k_{c,c3} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & k_{c,c(n-1)}+k_{c,cn} & -k_{c,cn} \\ 0 & 0 & \cdots & -k_{c,cn} & k_{c,cn} \end{bmatrix} \tag{10.2.2}$$

$$\mathbf{C_t}=\begin{bmatrix} k_{c,t1}+k_{c,t2} & -k_{c,t2} & \cdots & 0 & 0 \\ -k_{c,t2} & k_{c,t2}+k_{c,t3} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & k_{c,t(n-1)}+k_{c,tn} & -k_{c,tn} \\ 0 & 0 & \cdots & -k_{c,tn} & k_{c,tn} \end{bmatrix} \tag{10.2.3}$$

$$\mathbf{D_c}=\begin{bmatrix} \frac{1}{l_t}+\frac{k_{c,c1}+k_{c,c2}}{l_t}\left(\frac{1}{k_{d1}}+\frac{1}{k_{t1}}\right) & -\frac{k_{c,c2}}{l_t}\left(\frac{1}{k_{d1}}+\frac{1}{k_{t1}}\right) & \cdots & 0 & 0 \\ -\frac{k_{c,c2}}{l_t}\left(\frac{1}{k_{d2}}+\frac{1}{k_{t2}}\right) & \frac{1}{l_t}+\frac{k_{c,c2}+k_{c,c3}}{l_t}\left(\frac{1}{k_{d2}}+\frac{1}{k_{t2}}\right) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{l_t}+\frac{k_{c,c(n-1)}+k_{c,cn}}{l_t}\left(\frac{1}{k_{d(n-1)}}+\frac{1}{k_{t(n-1)}}\right) & -\frac{k_{c,cn}}{l_t}\left(\frac{1}{k_{d(n-1)}}+\frac{1}{k_{t(n-1)}}\right) \\ 0 & 0 & \cdots & -\frac{k_{c,cn}}{l_t}\left(\frac{1}{k_{dn}}+\frac{1}{k_{tn}}\right) & \frac{1}{l_t}+\frac{k_{c,cn}}{l_t}\left(\frac{1}{k_{dn}}+\frac{1}{k_{tn}}\right) \end{bmatrix} \tag{10.2.4}$$

$$\mathbf{D_t}=\begin{bmatrix} \frac{1}{l_t}+\frac{k_{c,t1}+k_{c,t2}}{l_t}\left(\frac{1}{k_{d1}}+\frac{1}{k_{t1}}\right) & -\frac{k_{c,t2}}{l_t}\left(\frac{1}{k_{d1}}+\frac{1}{k_{t1}}\right) & \cdots & 0 & 0 \\ -\frac{k_{c,t2}}{l_t}\left(\frac{1}{k_{d2}}+\frac{1}{k_{t2}}\right) & \frac{1}{l_t}+\frac{k_{c,t2}+k_{c,t3}}{l_t}\left(\frac{1}{k_{d2}}+\frac{1}{k_{t2}}\right) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \frac{1}{l_t}+\frac{k_{c,t(n-1)}+k_{c,tn}}{l_t}\left(\frac{1}{k_{d(n-1)}}+\frac{1}{k_{t(n-1)}}\right) & -\frac{k_{c,tn}}{l_t}\left(\frac{1}{k_{d(n-1)}}+\frac{1}{k_{t(n-1)}}\right) \\ 0 & 0 & \cdots & -\frac{k_{c,tn}}{l_t}\left(\frac{1}{k_{dn}}+\frac{1}{k_{tn}}\right) & \frac{1}{l_t}+\frac{k_{c,tn}}{l_t}\left(\frac{1}{k_{dn}}+\frac{1}{k_{tn}}\right) \end{bmatrix} \tag{10.2.5}$$

Where $k_{c,cn}$ and $k_{c,tn}$ are the compressive and tensile axial stiffness of the perimeter column within the n^{th} segment (Figure 10.2.1), respectively, and $\mathbf{k'_g}$ is the rotational stiffness applied by the BRB-outriggers. For the single BRB-outrigger system incorporating perimeter columns with different axial tensile and compressive stiffness, the Equation (3.2.7) can be modified as follows:

$$M_o=l_t^2\left(\frac{1}{\alpha/k_{c,c}+1/k_d+1/k_t}+\frac{1}{\alpha/k_{c,t}+1/k_d+1/k_t}\right)\theta=k_g^{'}\theta \tag{10.2.6}$$

In design practices, the perimeter column cross-section size would become smaller with the increasing elevation. Therefore, the values of $k_{c,c}$ and $k_{c,t}$ are equivalent axial stiffness considering various cross-section sizes within the column length.

10.2.1 Design examples

The 40-story single BRB-outrigger example model introduced in Chapter 8 is used to demonstrate the effects when the perimeter column develops different tensile and compressive axial stiffness. The perimeter column cross-section is replaced by Box 850×850×22 mm made by SN490 (yield stress = 325 MPa) with infill concrete, which has a compressive strength (f'_c) of 8000 psi (55 MPa). When the perimeter column is in tension, the tensile axial stiffness ($k_{c,t}$) considers the contribution from steel only and can is calculated as follows:

$$k_{c,t} = \frac{EA_s}{h} = 91080 \text{ kN/m} \tag{10.2.7}$$

Where A_s (72864 mm^2) is the cross-sectional area of the steel box. The compressive stiffness ($k_{c,c}$) can be calculated as follows:

$$k_{c,c} = \frac{EA_s}{h} + \frac{E_c A_c}{h} = 233801 \text{ kN/m} \tag{10.2.8}$$

Where A_c (649636 mm^2) is the cross-sectional area of the infill concrete. E_c is the modulus of elasticity of the concrete and is evaluated from Equation 19.2.2.1b in the reference (American Concrete Institute (ACI), 2014):

$$E_c = 57000\sqrt{f_c^{'}(\text{in psi})} = 5.098\times10^6 \text{ psi} = 3.515\times10^7 \text{ kN/m}^2 \tag{10.2.9}$$

The designs of outrigger truss shown in Figure 8.2.3 and the BRB_Rdc1 (Figure 8.2.5) are used for the example model. The k_t and k_d are 187987 kN/m and 301480 kN/m, respectively. The SA was performed by using DM model, and the NLRHA was performed by using both DM and MBM models. The perimeter columns with different axial stiffness were modeled by using elastic material with different Young's modulus in tension and compression. Two cases were considered, the models with different perimeter column axial tensile and compressive axial stiffness are known as Case2. The Case1 ignores the contribution from concrete even the perimeter column is in compressive strain, thus, the $k_{c,t}$ and $k_{c,c}$ equal to 91080 kN/m. For conservative estimation, the gravity load demands were ignored. Figure 10.2.2 shows the material model for modeling the CFT column in both the DM and MBM models, where $E_{s,t}$ and $E_{c,t}$ are modified modulus of elasticity for the perimeter column subjected to tension and compression, respectively.

Table 10.2.1 shows the modal analysis results. Both the analysis results of DM and MBM models suggest that the system stiffness increases when the contribution of concrete in CFT column is included (Case2). Figure 10.2.3 and Figure 10.2.4 show the roof drift (θ) and perimeter column axial force (C_1) histories, respectively, calculated from NLRHA with originally observed ground motions. The differences between using DM and MBM are slightly larger than the differences between Case1 and Case2. This indicates that the effects of different $k_{c,c}$ and $k_{c,t}$ values on seismic performance would be marginal, for the example model, and it would not affect the effectiveness of using DM model to analyze the structure with BRB-outrigger system, either. Figure 10.2.5 shows the maximum seismic response of the Case1 and Case2 DM and MBM models. Figure 10.2.6 shows the relations between BRB axial force and deformation for Case1 and Case2 models calculated from NLRHA using DM and MBM models. The $\theta_{\max}$ calculated from SA are 0.992 and 0.944% rad. for the Case1 and Case2 models, respectively. The $\gamma_{\max}$ calculated from SA are 1.344 and 1.260% rad. for the Case1 and Case2, respectively. Both the SA and NLRHA indicate that the $\theta_{\max}$, $\gamma_{\max}$, and $V_{c,\max}$ are generally slightly smaller in the Case2 (when greater $k_{c,c}$ due to the concrete in CFT column is considered). However, as the fundamental periods of the Case2 models become smaller, the $\theta_{\max}$ and $\gamma_{\max}$ can be amplified during a smaller earthquake because of the BRB keeps deforming elastically or undergoes very small inelastic deformation (such as the Taft ground motion). In addition, the Case2 models generally result in greater $C_{1,\max}$ because of larger perimeter column axial stiffness. However, the increasing of $C_{1,\max}$ is marginal, the compressive capacity of the CFT column should be sufficient to sustain the increased $C_{1,\max}$.

Based on the analysis results on the example models. The proposed method can be modified to analyze the structure with BRB-outrigger and CFT perimeter columns. When the larger $k_{c,c}$ because of the contribution from the infill concrete in CFT columns is considered, the seismic response could be reduced. This calculation procedure can be used to analyze the structure with BRB-outrigger incorporating reinforced concrete column. For the design practice, it is recommended to include the effect of different $k_{c,c}$ and $k_{c,t}$ for the structure incorporates reinforced concrete or CFT as the perimeter columns.

Table 10.2.1 Vibration periods of the Case1 and Case2 configurations calculated from using DM and MBM models

	Case1 ($k_{c,t} = k_{c,c}$ = 91080 kN/m)				Case2 ($k_{c,t}$ = 91080 kN/m, $k_{c,c}$ = 233801 kN/m)			
	1st mode	2nd mode	3rd mode	4th mode	1st mode	2nd mode	3rd mode	4th mode
DM	4.739	0.857	0.316	0.161	4.528	0.847	0.316	0.160
MBM	4.726	0.903	0.345	0.180	4.474	0.887	0.345	0.179

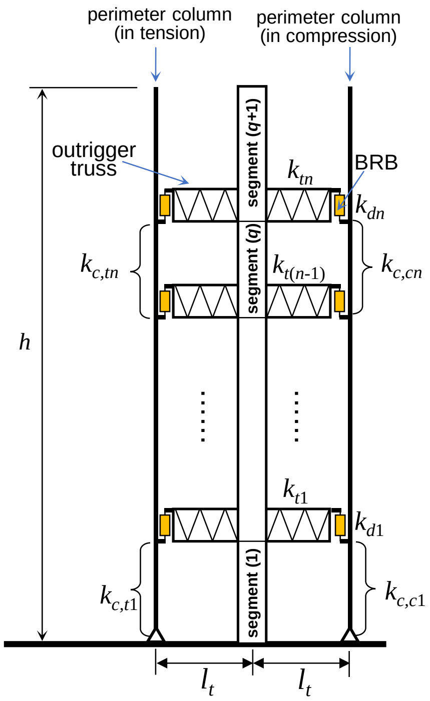


Figure 10.2.1 Illustration of the single BRB-outrigger system with the perimeter columns have different axial tensile and compressive stiffness

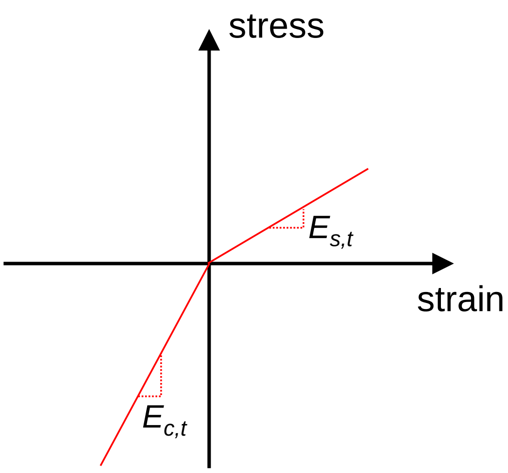


Figure 10.2.2 Material model of the CFT column used in the OpenSees model

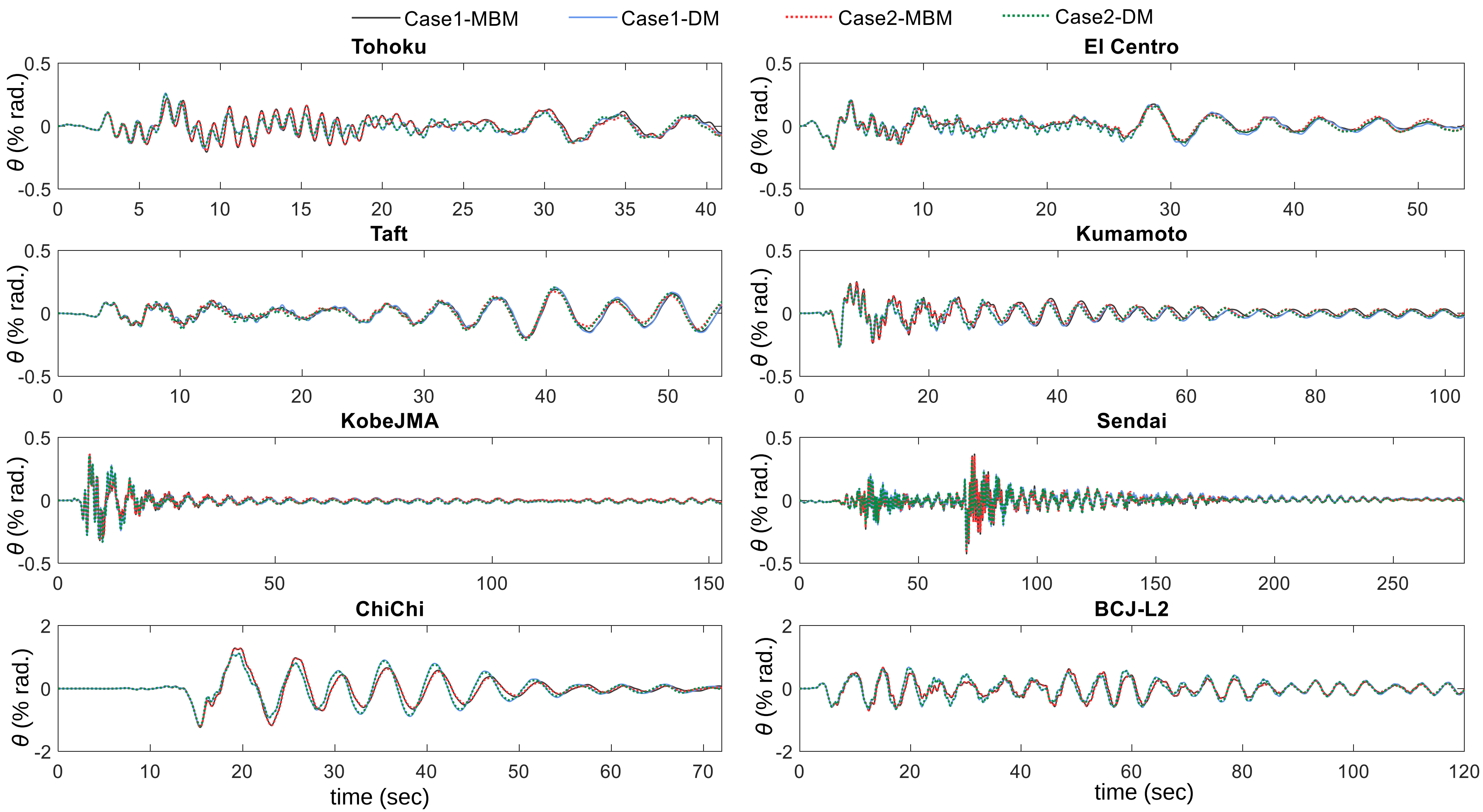


Figure 10.2.3 The roof drift (θ) histories of the Case1 and Case2 models calculated from using DM and MBM models

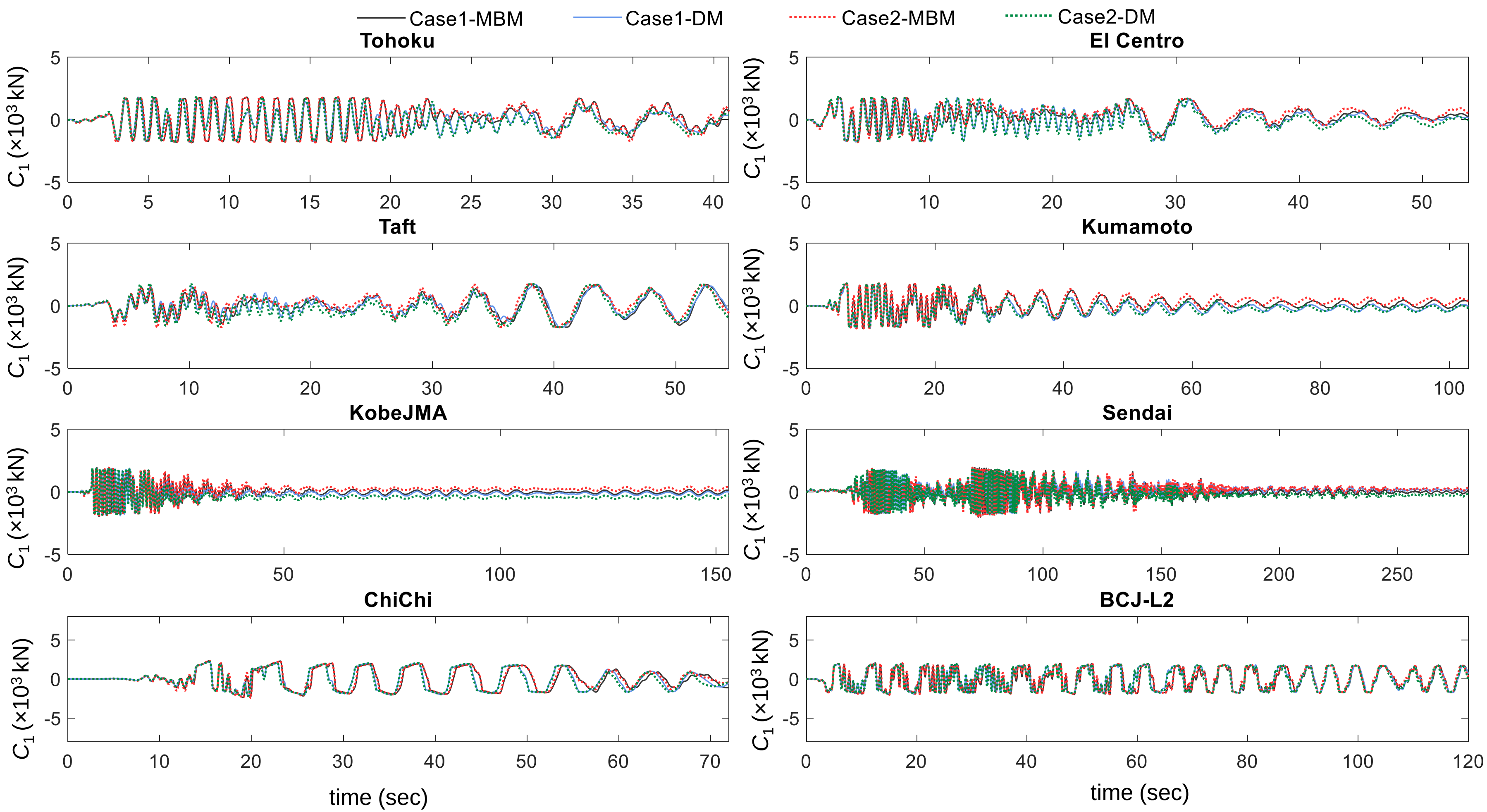


Figure 10.2.4 The perimeter column axial force (C_1) histories of the Case1 and Case2 models calculated from using DM and MBM models

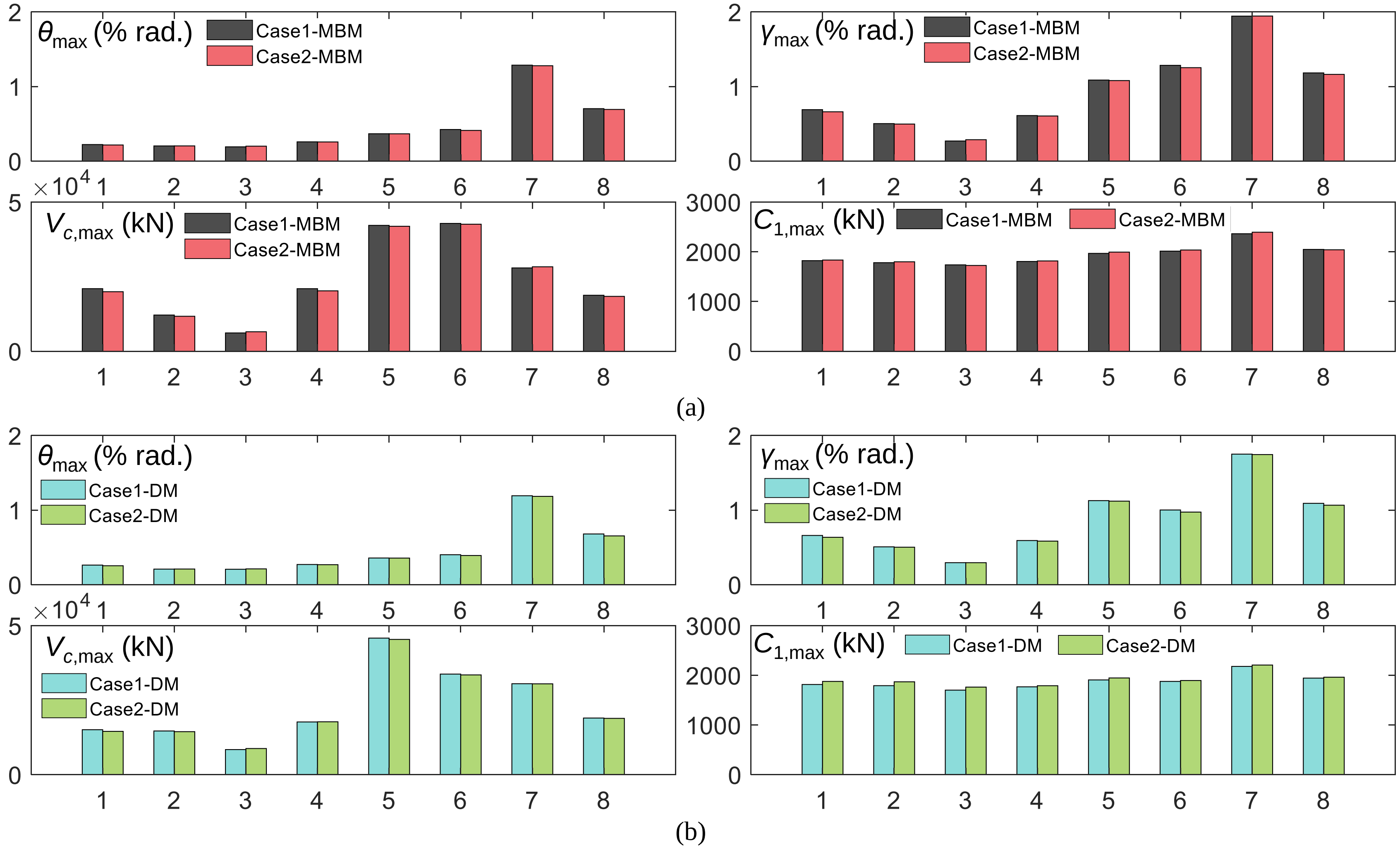
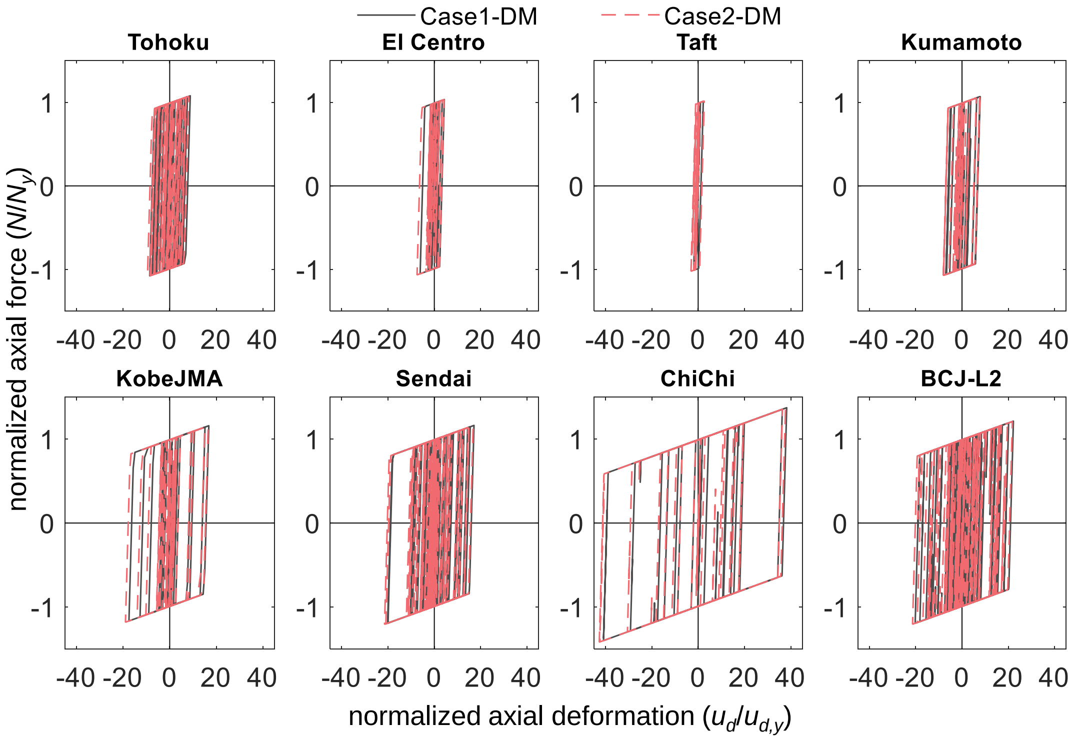
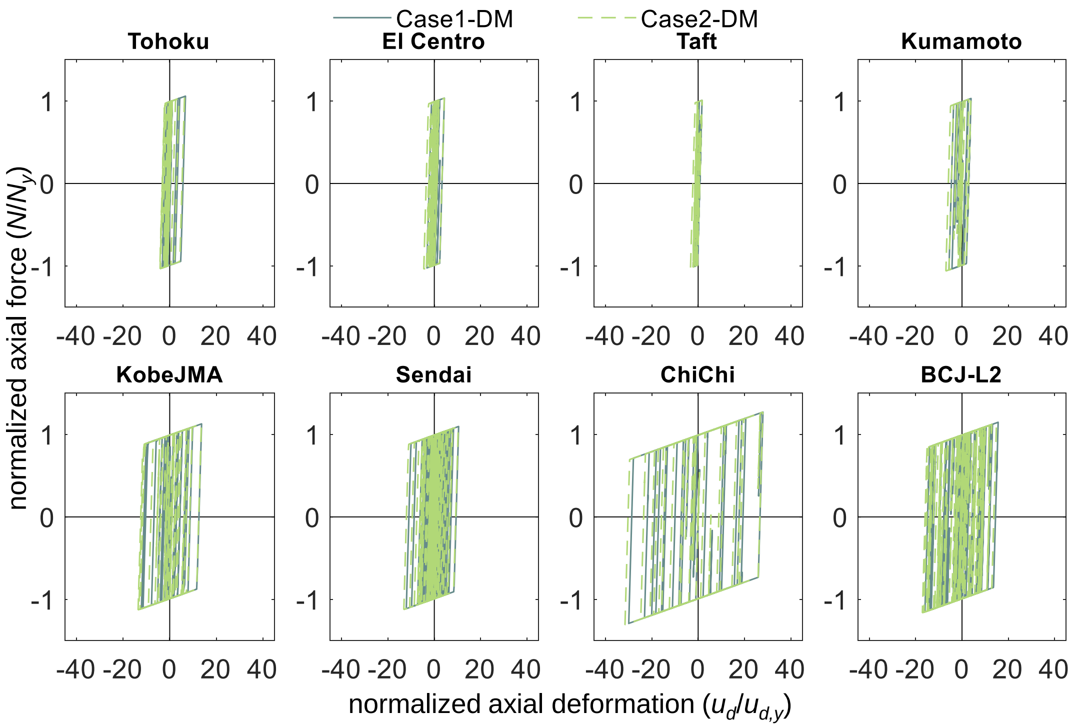


Figure 10.2.5 The NLRHA results of the maximum seismic response of Case1 and Case2 calculated by using (a) MBM and (b) DM models ground motion: 1=Tohoku, 2=El Centro, 3=Taft, 4=Kumamoto, 5= KobeJMA, 6=Sendai, 7=ChiChi, 8=BCJL2



(a)



(b)

Figure 10.2.6 The axial force and deformation relation of the BRB calculated from NLRHA by using (a) DM and (b) MBM models

10.3 LINEARLY CHANGED CORE RIGIDITY AND PERIMETER COLUMN AXIAL STIFFNESS

In the design practices, the core structure flexural rigidity and the perimeter column size are smaller with the increasing elevation for economical design reasons. This section investigates the influences on the seismic response when core structure flexural rigidity and perimeter column sizes vary with the elevation as shown in Figure 10.3.1. If the core structure flexural rigidity at the base and top of the core structure are EI and μEI, the core structure flexural rigidity at an elevation of x measured from the core structure base can be expressed as follows:

$$EI(x) = \frac{EI(\mu - 1)}{h} x + EI \tag{10.3.1}$$

In addition, if the effective cross-sectional areas of the perimeter column at bottom and top of the building are A_{pc} and τA_{pc}, the cross-sectional area of perimeter column at an elevation of x measured from core structure base can be expressed as follows:

$$A_{pc}(x) = \frac{A_{pc}(\tau - 1)}{h} x + A_{pc} \tag{10.3.2}$$

The axial stiffness of the perimeter column ranges from elevations of x_1 to x_2 ($k_{c,12}$) can be calculated as follows:

$$k_{c,12} = \frac{1}{f_{c,12}}, \text{ where } f_{c,12} = \frac{1}{E}\int_{x_1}^{x_2} \frac{h}{A_{pc}\left[(\tau-1)x+h\right]} dx = \left.\frac{h \ln\left[(\tau-1)x+h\right]}{EA_{pc}(\tau-1)}\right|_{x=x_1}^{x=x_2} \text{ (when } \tau<1) \tag{10.3.3}$$

For simplicity, the tensile and compressive axial stiffnesses of the perimeter column are assumed to be identical to each other.

The 40-story example model introduced in Chapter 8 is used to demonstrate the effect of varying EI and A_{pc} along the building elevation on seismic response. For simplicity, the values of outrigger truss flexural stiffness (k_t), the BRB axial stiffness (k_d for single BRB-outrigger systems, k_{d1} and k_{d2} for dual BRB-outrigger systems), and the BRB yield deformation ($u_{d,y}$ for single BRB-outrigger systems, $u_{d,y1}$ and $u_{d,y2}$ for dual BRB-outrigger systems) are kept as constants while the outrigger elevation (α for single BRB-outrigger systems, α_1 and α_2 for dual BRB-outrigger systems) moves from 0 to 1. The design detail of outrigger truss and BRB (BRB_Rdc1) are shown in Figure 8.2.3 and Figure 8.2.5. The values of EI and A_{pc} at the building base are 4×10^9 kN-m^2 and 0.2475 m^2 (cross-sectional area of Box 900×900×75), respectively. The values of k_t (k_t for single BRB-outrigger systems, k_{t1} and k_{t2} for dual BRB-outrigger systems), k_d (k_d for single BRB-outrigger systems, k_{d1} and k_{d2} for dual BRB-outrigger systems), and $u_{d,y}$ ($u_{d,y}$ for single BRB-outrigger systems, $u_{d,y1}$ and $u_{d,y2}$ for dual BRB-outrigger systems) are 187987 kN/m, 301480 kN/m, and 5.6 mm, respectively. Table 10.3.1 shows the identifications of the analytical models with a different combination of μ and τ factors. In each analytical model, the value of α (α for single BRB-outrigger systems, α_1 and α_2 for dual BRB-outrigger systems) varies from 0 to 1.

Table 10.3.1 The identifications of analytical model with various μ and τ factors

		μ					
		1	0.8	0.6	0.4	0.2	0
τ	1	μ1 τ1	μ08 τ1	μ06 τ1	μ04 τ1	μ02 τ1	μ0 τ1
	0.8	μ1 τ08	μ08 τ08	-	-	-	-
	0.6	μ1 τ06	-	μ06 τ06	-	-	-
	0.4	μ1 τ04	-	-	μ04 τ04	-	-
	0.2	μ1 τ02	-	-	-	μ02 τ02	-
	0	μ1 τ0	-	-	-	-	μ0 τ0

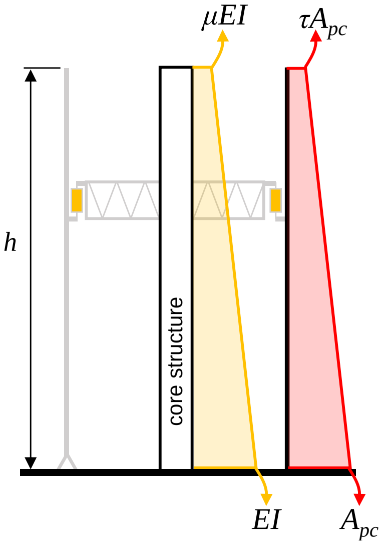


Figure 10.3.1 Illustration of the core structure flexural rigidity and perimeter column cross-sectional area vary with the building elevation

10.3.1 Single BRB-outrigger system

Figure 10.3.2 shows the relationships between the 1st to the 3rd mode vibration periods and outrigger elevation (α) with various values of μ and τ. It is found that when the value of EI at the building top is smaller (when μ is smaller), the overall vibration periods (the 1st to the 3rd modes) increase because of the core structure is less stiff. In addition, when the value of A_{pc} at the building top is smaller (when τ is smaller), the vibration periods also increase because the perimeter column is less stiff. However, the effect on the vibration periods because of changing τ is much less than changing μ. In addition, the effect of changing τ on vibration periods is obvious only when the value of μ is small and when α is larger than 0.5 for the 1st and 2nd modes. Figure 10.3.3 and Figure 10.3.4 show the responses of maximum roof drift ratio ($\theta_{\max}$) and maximum inter-story drift ratio ($\gamma_{\max}$) with respect to α, respectively. In general, the models with smaller values of μ (weaker core structure) and τ (weaker perimeter column) exhibit larger $\theta_{\max}$ and $\gamma_{\max}$ responses. When the μ value decreases, the overall $\theta_{\max}$ and $\gamma_{\max}$ responses increase. However, when the τ value decreases, the $\theta_{\max}$ and $\gamma_{\max}$ responses increase only when α is greater than around 0.6. The responses could be explained by the analysis results shown in Chapter 7. The smaller values of μ and τ suggests the S_{bc} and R_{dc} values are smaller, respectively. Therefore, the outrigger effect becomes smaller and the reduction on seismic response decreases. In addition, the changes in S_{bc} value (changes of μ) affect the seismic response more than the changes in R_{dc} value (changes of τ). Figure 10.3.5 and Figure 10.3.6 show the maximum overturning moment at the core structure base ($M_{c,\max}$) and the maximum perimeter column axial force ($C_{1,\max}$) with respect to α. As the responses of reaction forces ($M_{c,\max}$ and $C_{1,\max}$) are affected by both the deformation responses ($\theta_{\max}$ and $\gamma_{\max}$) and the stiffness factors (EI and k_c), the larger deformation responses do not necessarily lead to greater reaction forces ($M_{c,\max}$ and $C_{1,\max}$). For example, under the BCJ-L2 ground motion, both the models $\mu 0 \tau 0$ and $\mu 02 \tau 02$ exhibit the largest $\theta_{\max}$ and $\gamma_{\max}$ responses, and thus result in the greatest $C_{1,\max}$ response among all the models. However, model $\mu 06 \tau 06$ exhibits the maximum $M_{c,\max}$ response among all the models, although its $\theta_{\max}$ and $\gamma_{\max}$ responses are not the largest. When the values of μ and τ are smaller, the outrigger effects are smaller and the reductions on seismic response are less efficient. However, the seismic demands could be smaller due to a less stiff structure. Based on the analysis results, the case when μ and τ equal to 1 is the most efficient in reducing seismic response, although the $C_{1,\max}$ and $M_{c,\max}$ could be slightly amplified. As shown in Figure 10.3.3 and Figure 10.3.4, the optimal outrigger elevation (α) in order to minimize $\theta_{\max}$ and $\gamma_{\max}$ is approximately 0.7 to 0.8. The changes in μ and τ values do not significantly affect the optimal α. In addition, when the outrigger is at its optimal elevation, the $M_{c,\max}$ is close to its minimum value, and the $C_{1,\max}$ reaches its maximum when α is approximately 0.8 to 1.0.

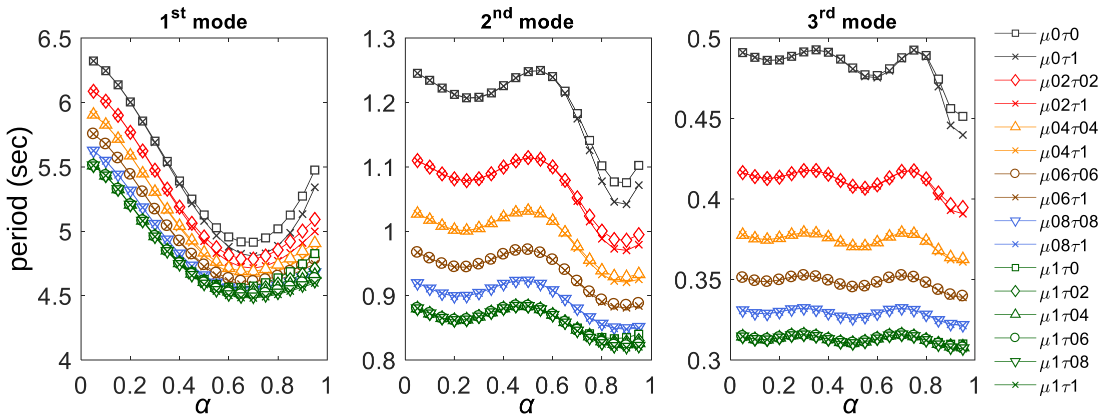


Figure 10.3.2 The 1st to the 3rd mode vibration periods of the 40-story single BRB-outrigger example model with various μ and τ values

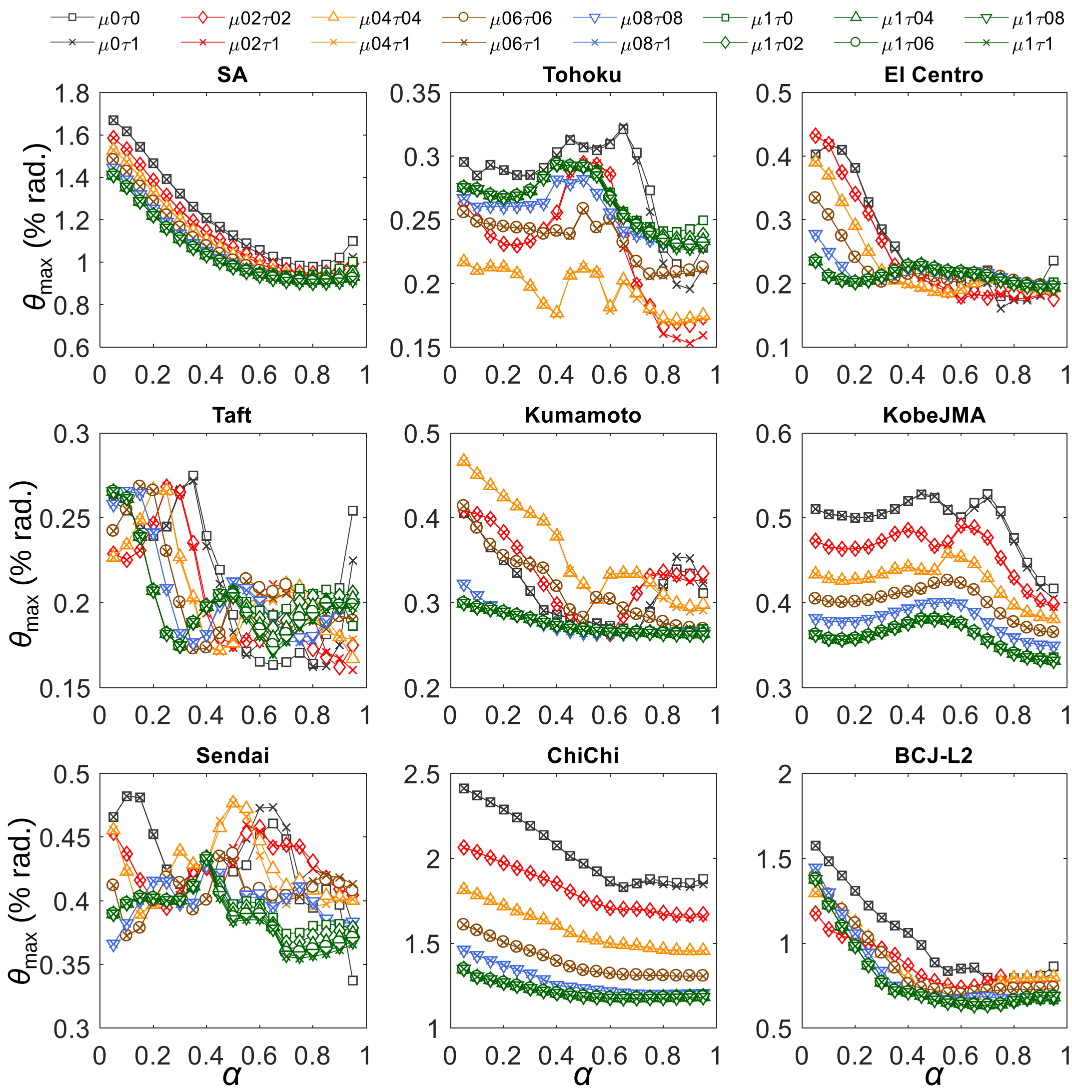


Figure 10.3.3 Relationship between θ_{max} and α of the 40-story single BRB-outrigger example model with various μ and τ values

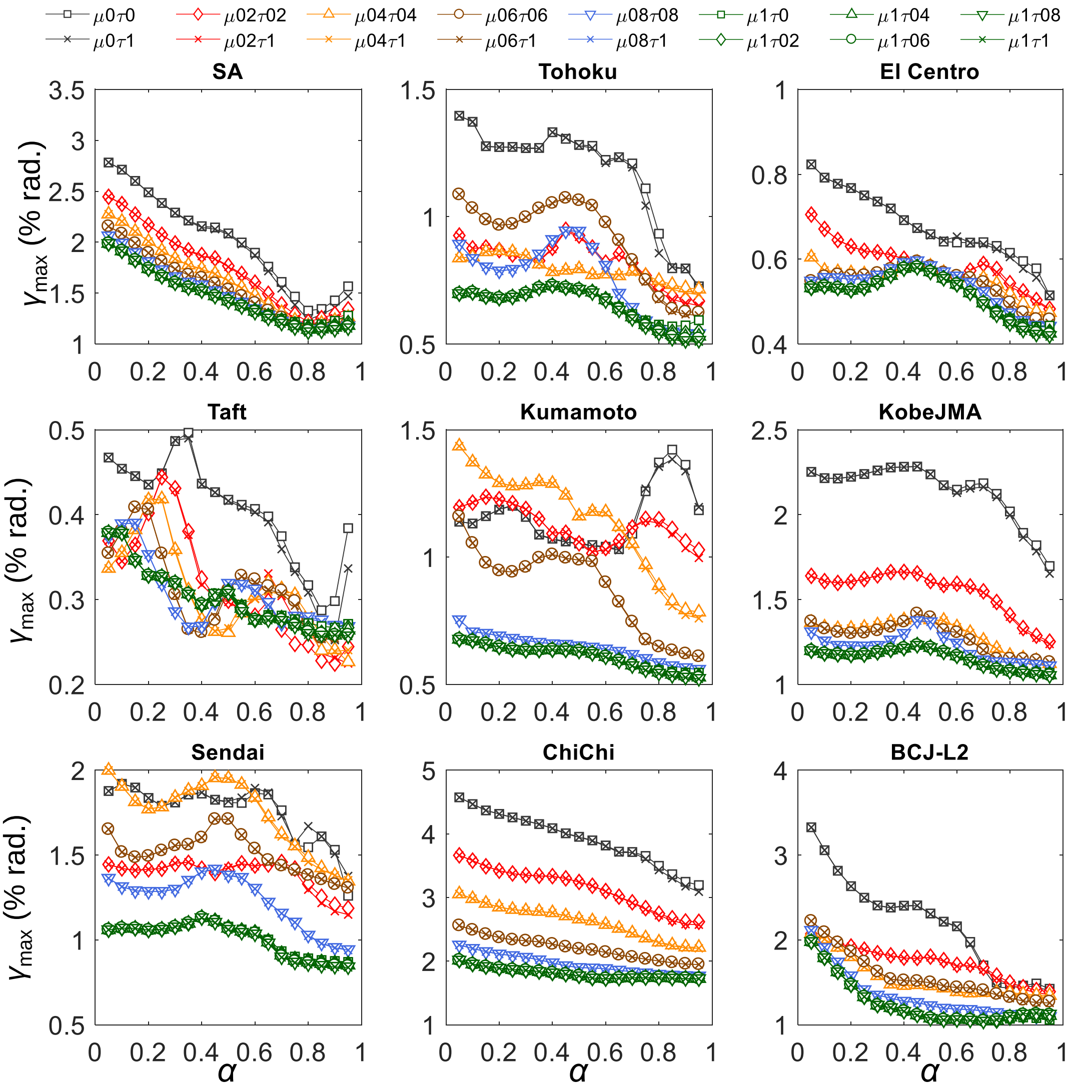


Figure 10.3.4 Relationship between γ_{max} and α of the 40-story single BRB-outrigger example model with various μ and τ values

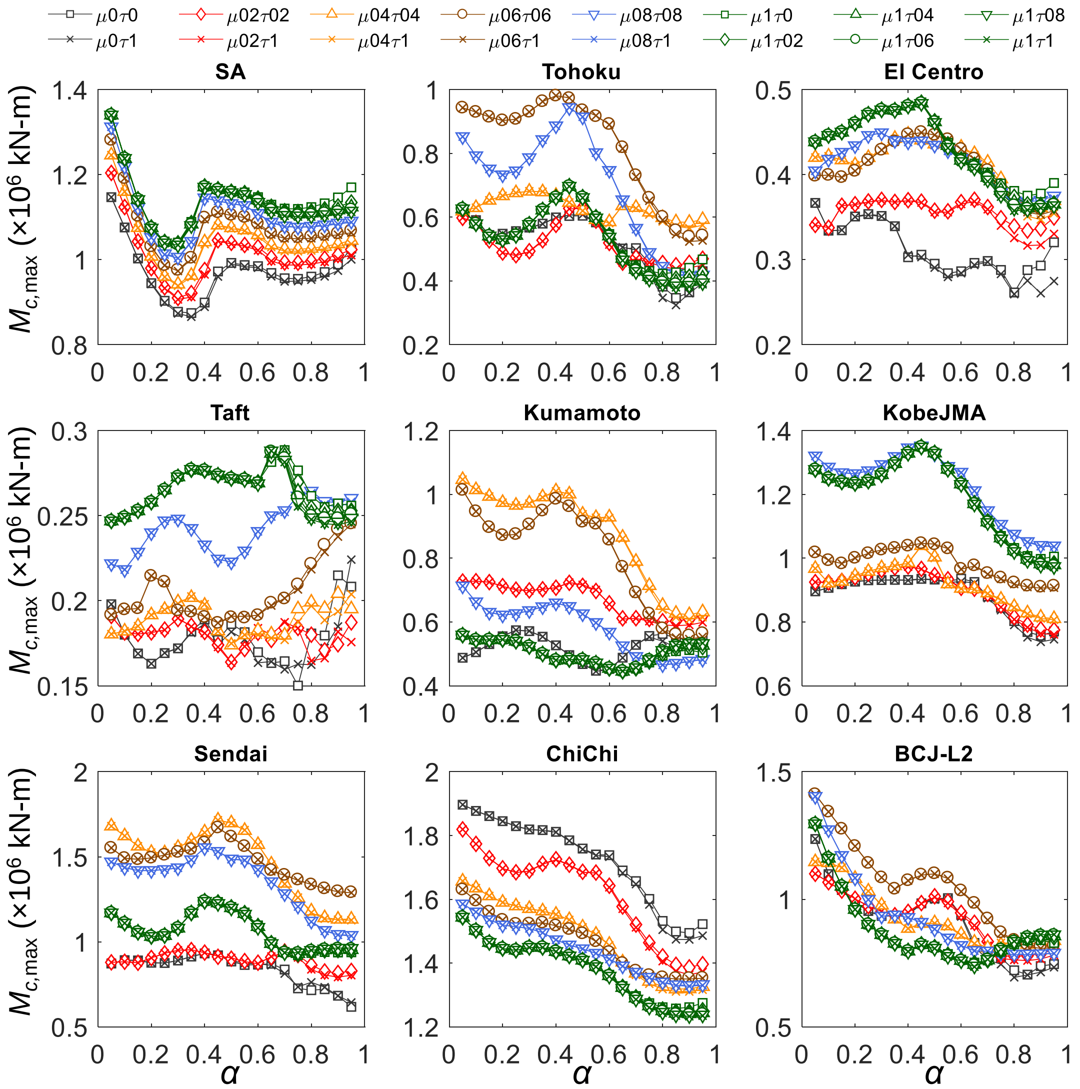


Figure 10.3.5 Relationship between $M_{c,\max}$ and α of the 40-story single BRB-outrigger example model with various μ and τ values

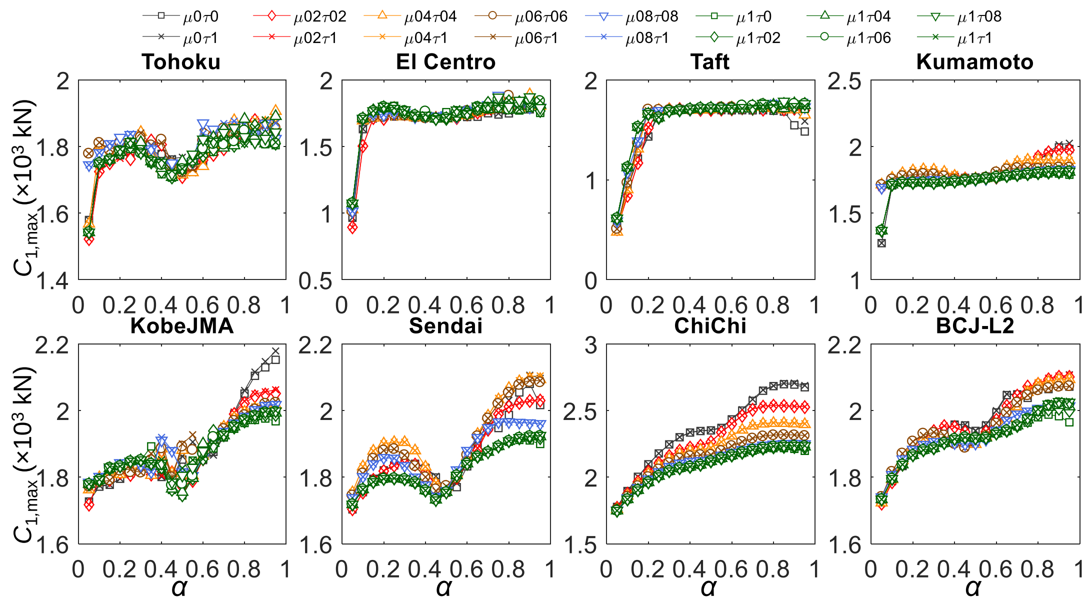


Figure 10.3.6 Relationship between $C_{1,\max}$ and α of the 40-story single BRB-outrigger example model with various μ and τ values

10.3.2 Dual BRB-outrigger system

Figure 10.3.7 and Figure 10.3.8 show the distributions of the 1st (T_1) and the 2nd (2nd) mode vibration periods with respect to the outrigger elevations (α_1 and α_2) under various factors of μ and τ. The vibration periods are longer when the values of μ and τ are smaller. If compare the case when μ=0 and τ=0 to the case when μ=1 and τ=1, the maximum possible increases of the 1st and the 2nd vibration periods are 9% and 31%, respectively. Based on the modal analysis results, the effect on vibration periods because of changing μ (EI) is more significant than changing τ (k_c). This is because the value of S_{bc2} (changed with μ) has a greater influence on seismic response than the value of R_{d2c} (changed with τ) as discussed in Chapter 7. The effect of changing τ value on the 2nd mode vibration period is marginal as the outrigger effect functions more efficient in the 1st mode deformed shape. The distributions of T_1 and T_2 are similar to the analytical results shown in Chapter 7. The T_1 majorly changes with α_2, and reaches the minimum when α_2 is around 0.6 to 0.8. The T_2 also majorly changes with α_2. The T_2 reaches the minimum when α_2 is around 0.8 to 0.9 and when α_1 is approximate 0.9. Figure 10.3.9 and Figure 10.3.10 show the maximum roof drift ratio (θ_{max}) distribution with respect to α_1 and α_2 calculated from SA and NLRHA with BCJ-L2 ground motion, respectively. Figure 10.3.11 and Figure 10.3.12 show the maximum inter-story drift ratio (γ_{max}) with respect to α_1 and α_2 calculated from SA and NLRHA with BCJ-L2 ground motion, respectively. Similar to the single BRB-outrigger case, the larger values of μ and τ lead to the slightly smaller θ_{max} and γ_{max} responses. If compare the case when μ=0 and τ=0 to the case when μ=1 and τ=1, the θ_{max} are only increased by 10% and 2% based on the SA and NLRHA results, respectively, and the γ_{max} are increased by 22% and 43% based on the SA and NLRHA results, respectively. In addition, the reductions in θ_{max} and γ_{max} because of the increased μ is more effective than increasing τ, which is similar to the T_1 and T_2 responses.

Based on the SA and NLRHA results, the optimal outrigger elevations in order to reach minimum θ_{max} and γ_{max} values are approximately 0.7 and 0.8 for α_2 and α_1, respectively. The optimal outrigger elevations do not significantly vary with different values of μ and τ. Figure 10.3.13 and Figure 10.3.14 show the maximum overturning moment at the core structure base ($M_{c,max}$) and the maximum axial force of the perimeter column ($C_{1,max}$) calculated from NLRHA with BCJ-L2 ground motion, respectively. Similar to the single BRB-outrigger example, the responses of reaction forces ($M_{c,\max}$ and $C_{1,\max}$) are affected by both the deformation responses ($\theta_{\max}$ and $\gamma_{\max}$) and the stiffness factors (EI and k_c). The larger deformation responses do not necessarily result in greater reactions. The model $\mu04\tau04$ exhibits the maximum $M_{c,\max}$ response among all the models, although its $\theta_{\max}$ and $\gamma_{\max}$ responses are not the greatest. The $C_{1,\max}$ responses shown in Figure 10.3.14 indicate that the value of $C_{1,\max}$ reaches its maximum limit at approximate 3800 kN when α_1 and α_2 are approximately 0.2, and almost stop changing when α_1 and α_2 keep increasing. Figure 10.3.15 and Figure 10.3.16 show the R_{CPD} values of the BRB in the lower (R_{CPD1}) and upper (R_{CPD2}) outriggers, respectively. The value of R_{CPD} greater than 1 indicates the BRB yields. The R_{CPD} responses suggest the BRB in the lower outrigger yields when α_1 and α_2 are greater than 0.2, and the BRB in the upper outrigger yields when α_2 is larger than 0.1. When both the BRBs in the upper and lower outrigger yield, the $C_{1,\max}$ reaches its maximum. This could explain the distribution of $C_{1,\max}$ response with respect to α_1 and α_2. Based on the analysis results, after the BRBs yield, the values of μ and τ have almost no effect on the $C_{1,\max}$ response. The $C_{1,\max}$ response shown in Figure 10.3.14 is different from the $C_{1,\max}$ response shown in Figure 7.3.26 to Figure 7.3.32, this is because that in this section, the values of k_{d1}, k_{d2}, $u_{d,y1}$, and $u_{d,y2}$ are set to be constants so that the maximum BRB axial force capacity is fixed and does not change with α_1 and α_2. However, the maximum axial force capacities of the BRBs shown in Figure 7.3.26 to Figure 7.3.32 vary with α_1 and α_2 because of the $u_{d,y1}$ and $u_{d,y2}$ vary with outrigger elevations. Figure 10.3.17 and Figure 10.3.18 show the energy dissipated by BRB_1 (E_{BRB1}) and BRB_2 (E_{BRB2}), respectively. The R_{CPD1} and E_{BRB1} become larger when both α_2 and α_1 are close to 1.0. The R_{CPD2} and E_{BRB2} become larger when α_2 and α_1 are close to 0.8 and 0, respectively. Based on the R_{CPD} and E_{BRB} responses, the energy performance of the BRBs is majorly affected by outrigger elevations, the values of μ and τ have almost no effect on R_{CPD} and E_{BRB}.

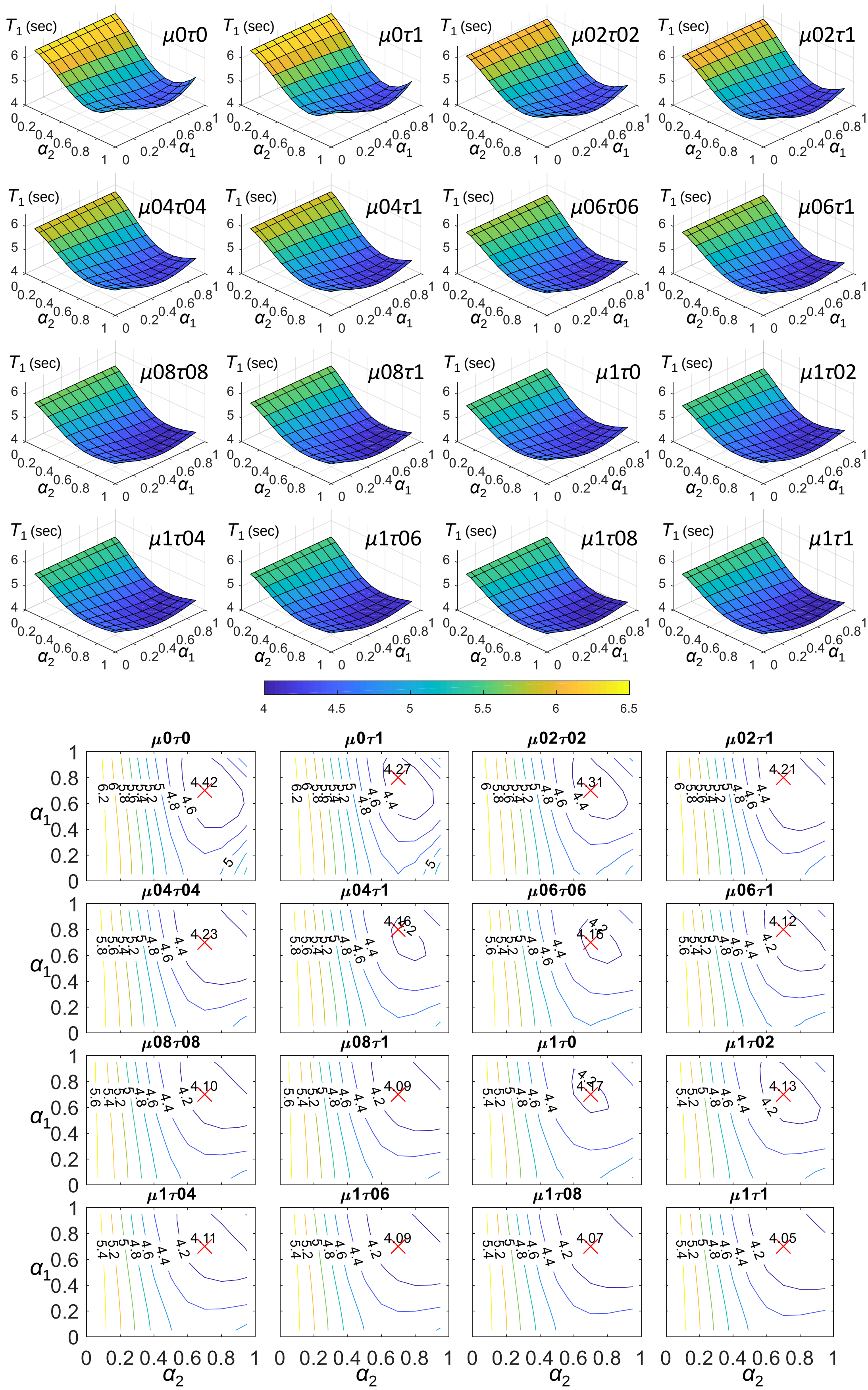


Figure 10.3.7 The 1st mode vibration periods of the 40-story dual BRB-outrigger example model

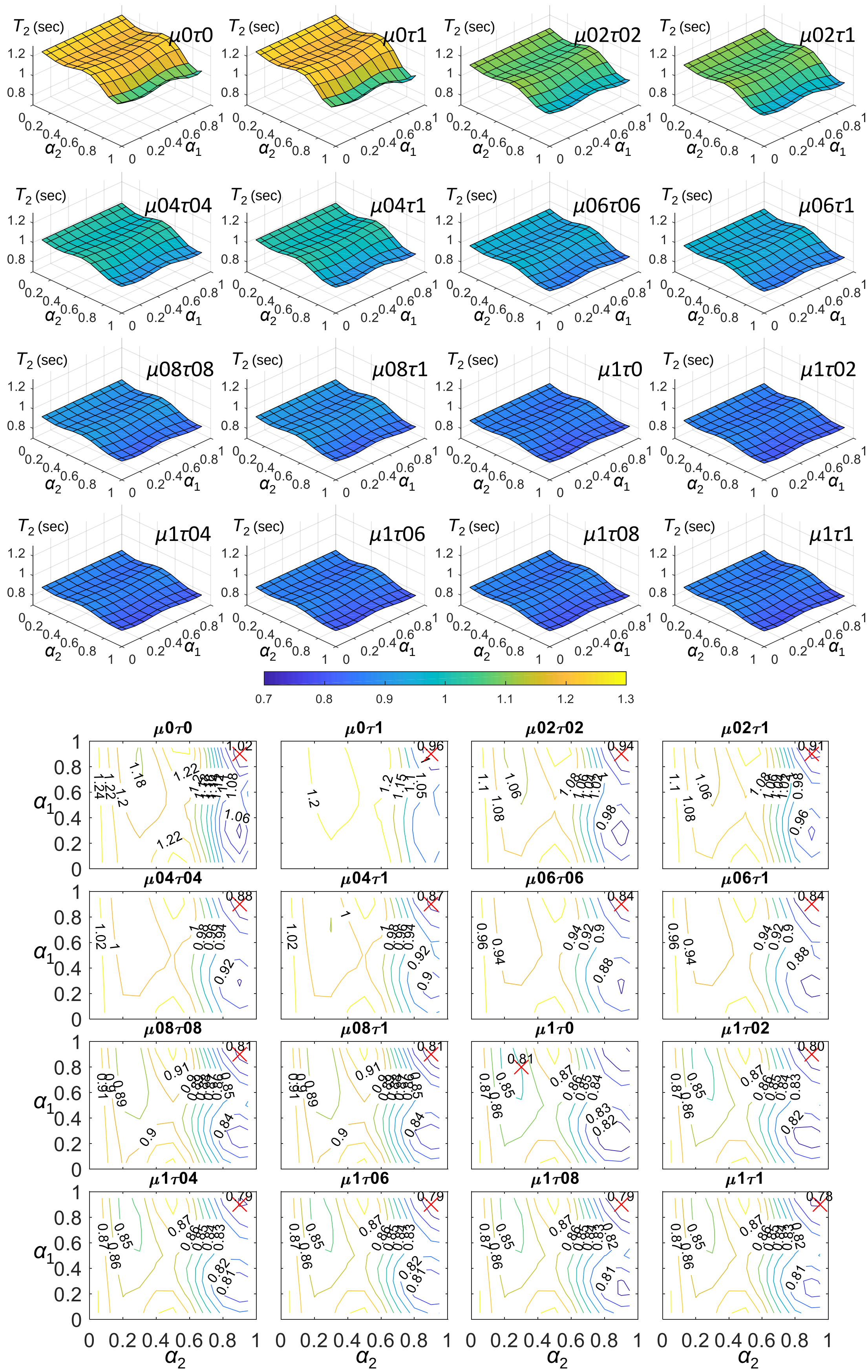


Figure 10.3.8 The 2nd mode vibration periods of the 40-story dual BRB-outrigger example model

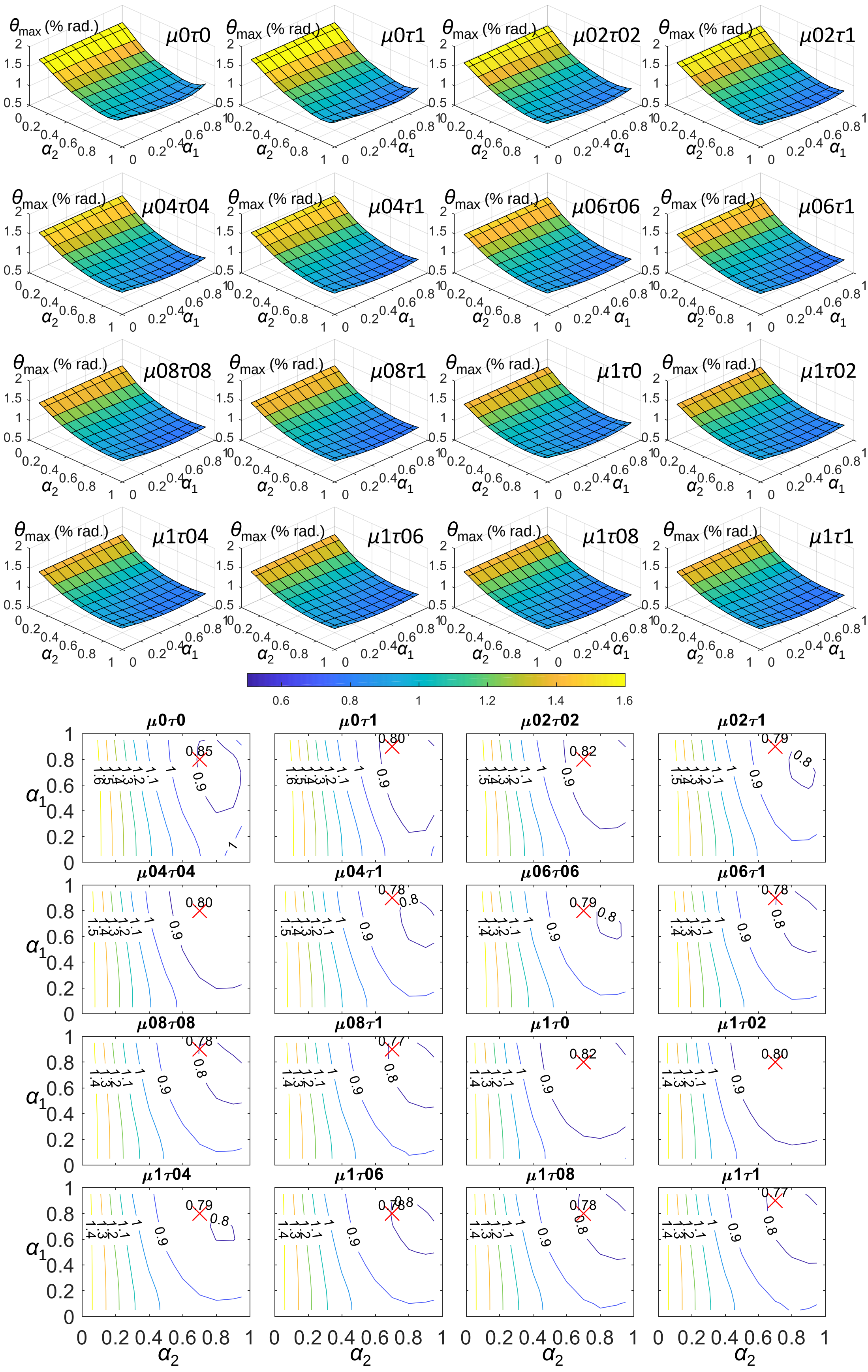


Figure 10.3.9 The $\theta_{\max}$ of the 40-story dual BRB-outrigger example model calculated from the SA (unit: % rad.)

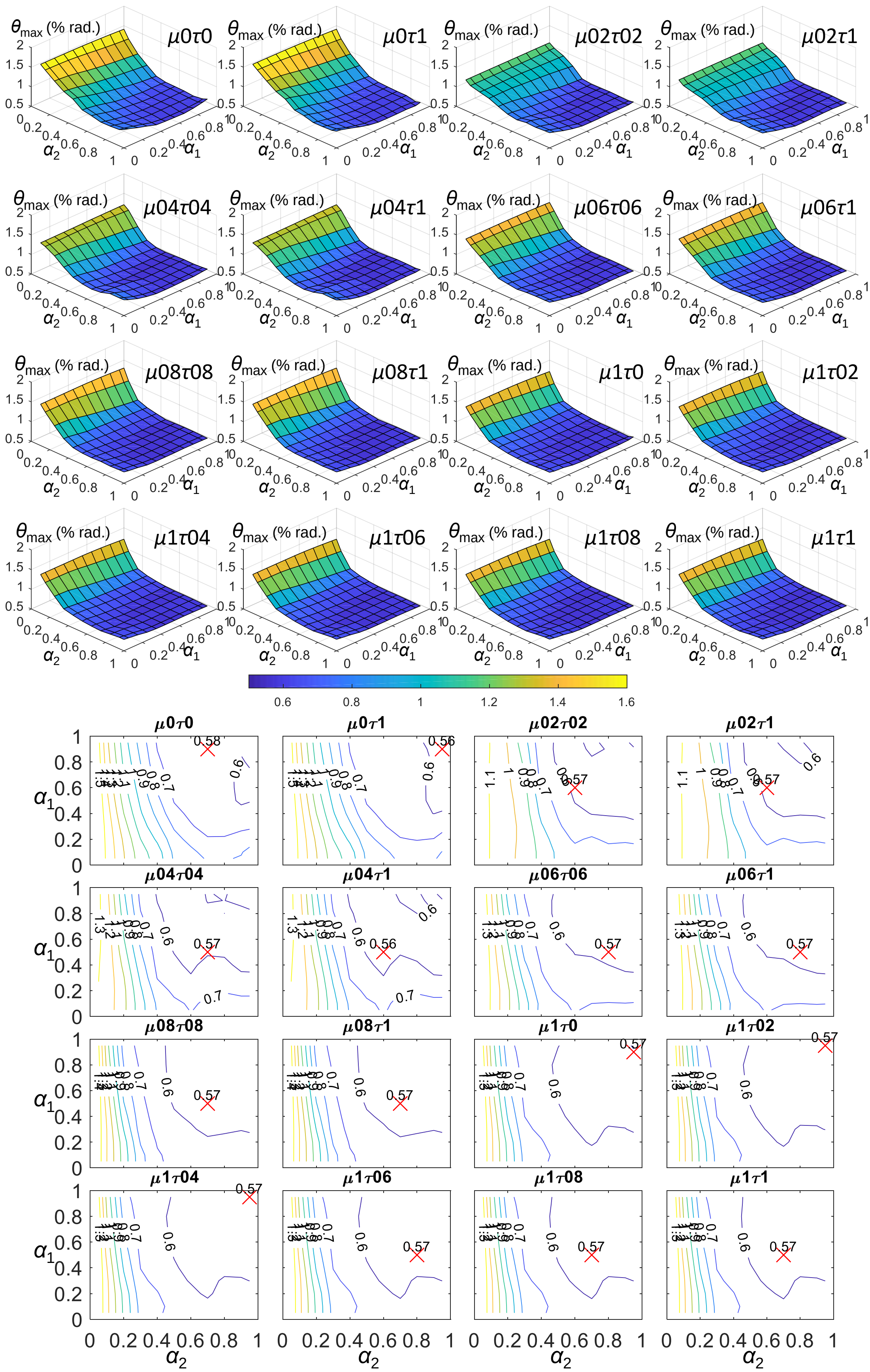


Figure 10.3.10 The θ_{max} of the 40-story dual BRB-outrigger example model calculated from the NLRHA with BCJ-L2 ground motion (unit: % rad.)

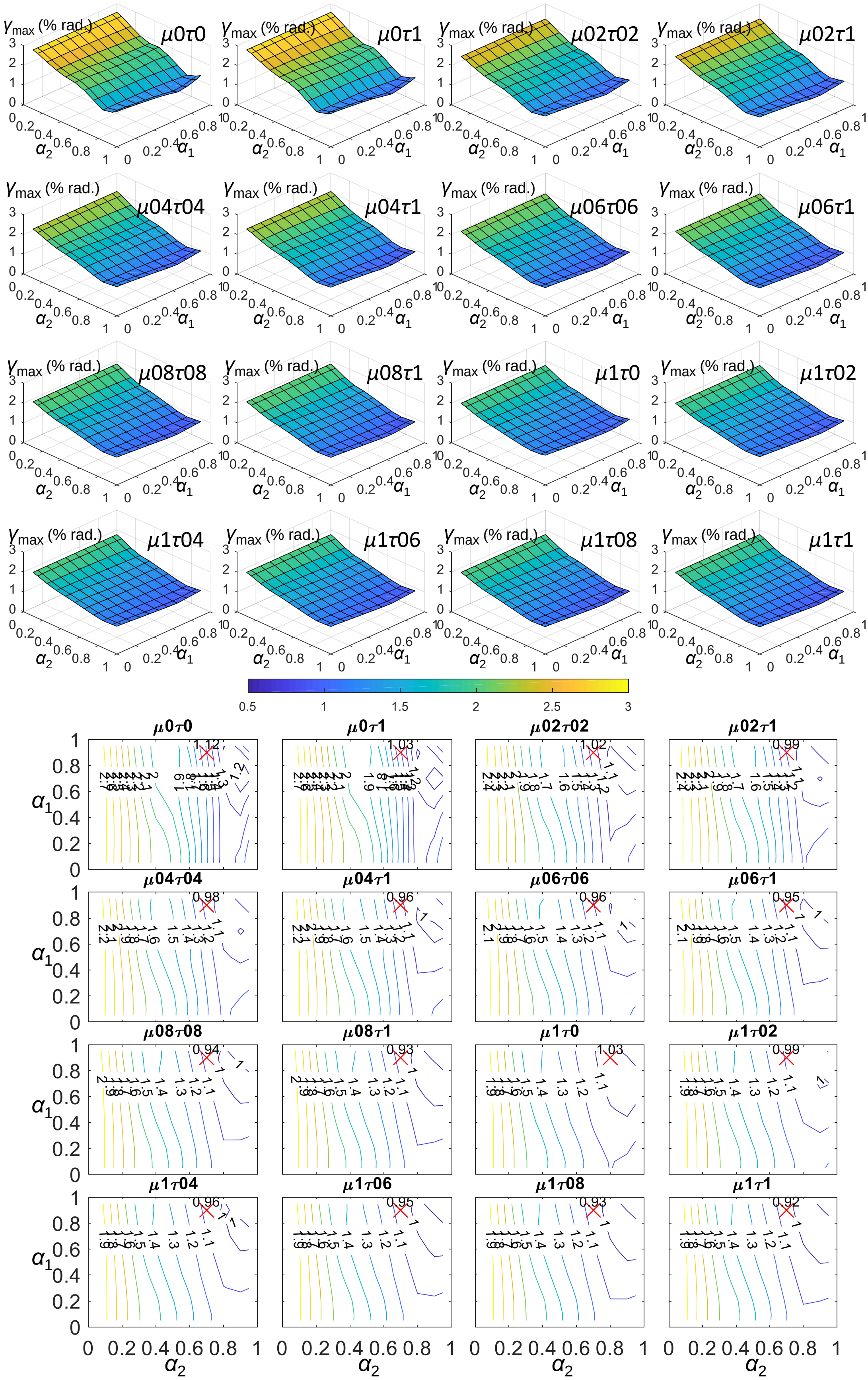


Figure 10.3.11 The γ_{max} of the 40-story dual BRB-outrigger example model calculated from the SA (unit: % rad.)

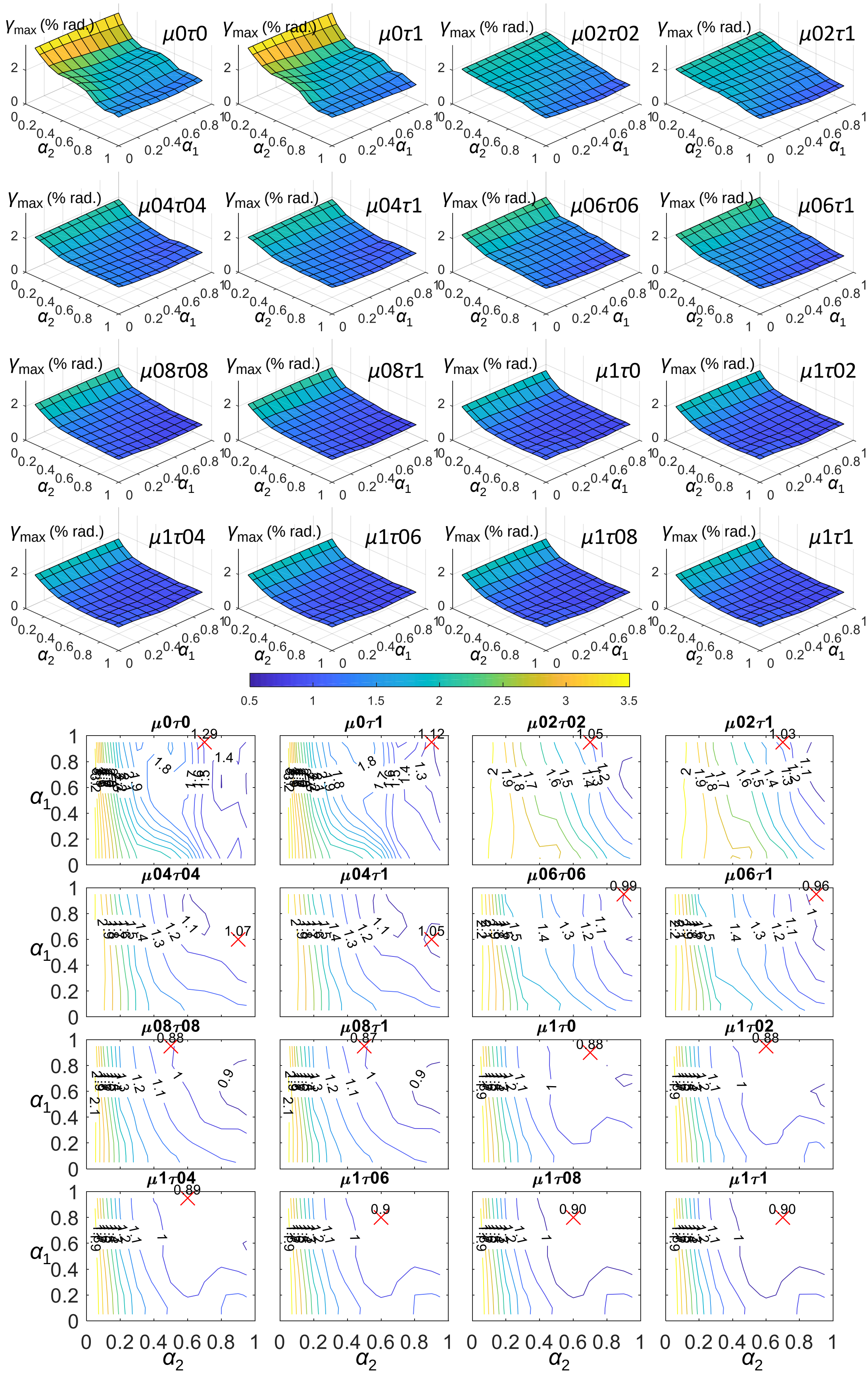


Figure 10.3.12 The γ_{max} of the 40-story dual BRB-outrigger example model calculated from the NLHRA with BCJ-L2 ground motion (unit: % rad.)

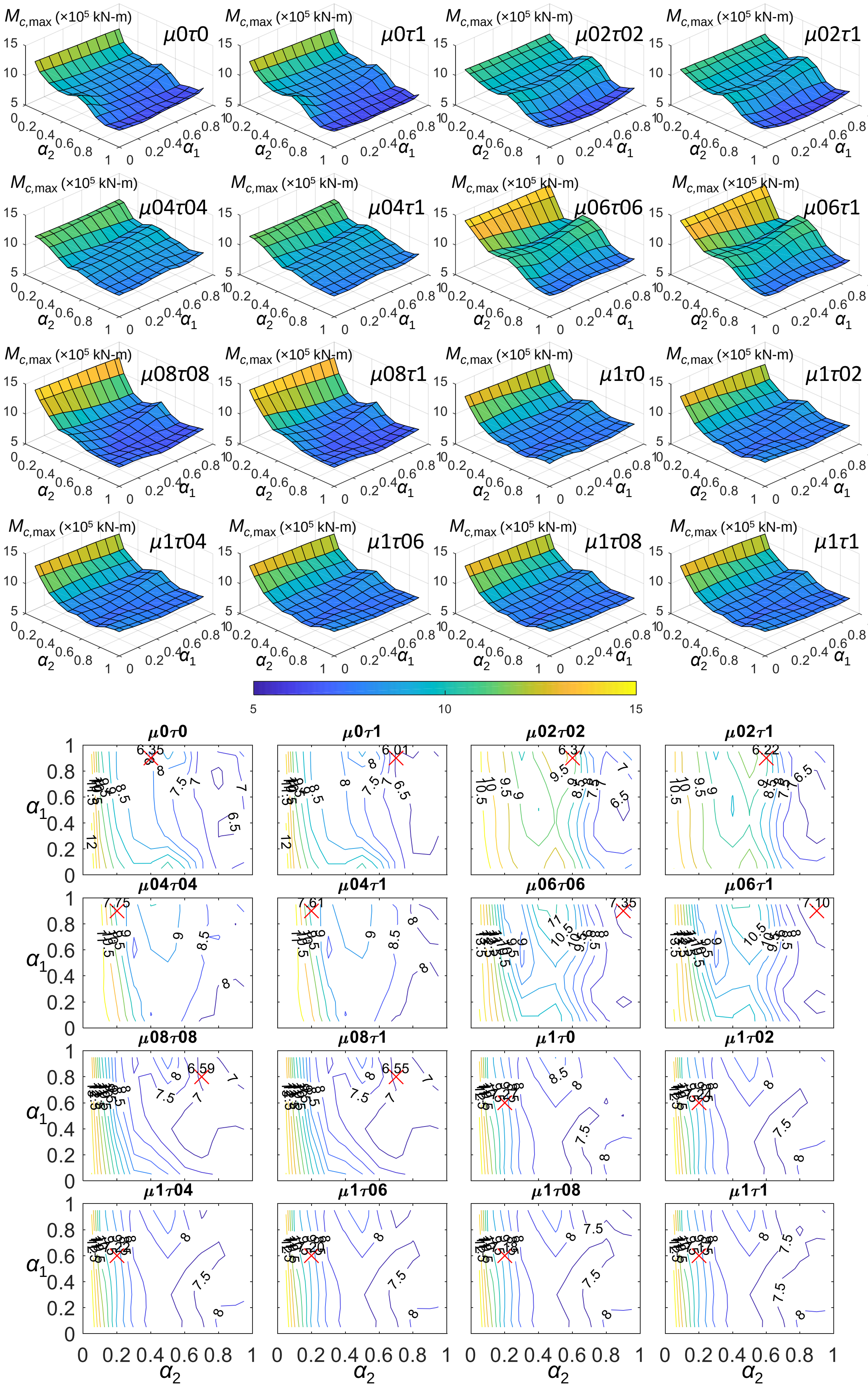


Figure 10.3.13 The $M_{c,\max}$ of the 40-story dual BRB-outrigger example model calculated from the NLHRA with BCJ-L2 ground motion (unit: $\times 10^5$ kN-m)

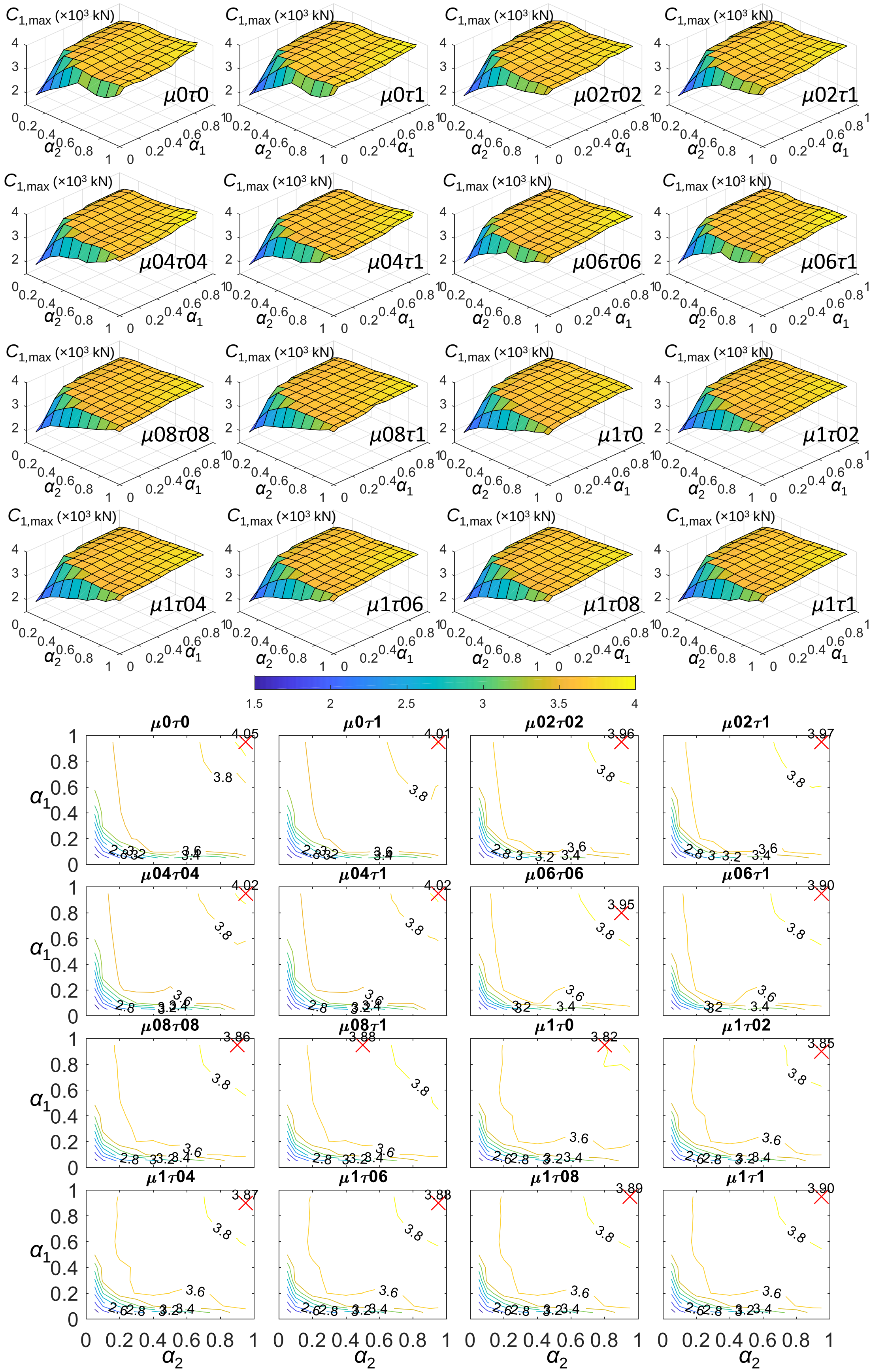


Figure 10.3.14 The $C_{1,\max}$ of the 40-story dual BRB-outrigger example model calculated from the NLHRA with BCJ-L2 ground motion (unit: $\times 10^3$ kN)

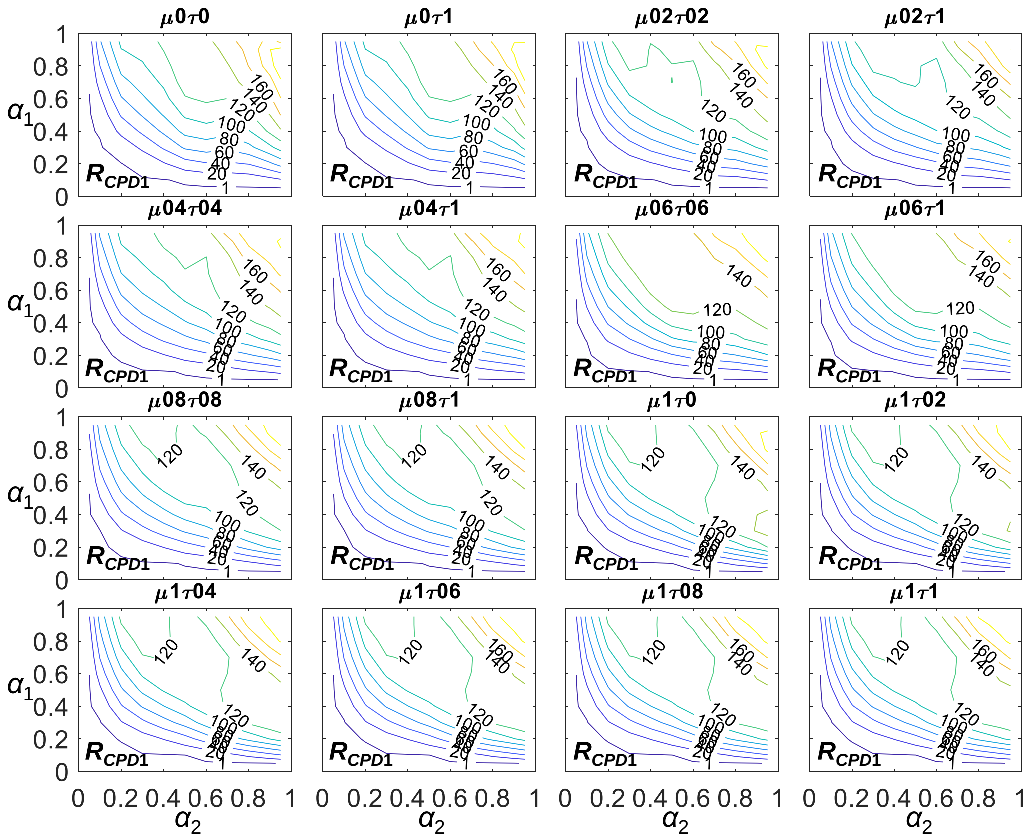


Figure 10.3.15 The R_{CPD1} calculated from the NLHRA with BCJ-L2 ground motion

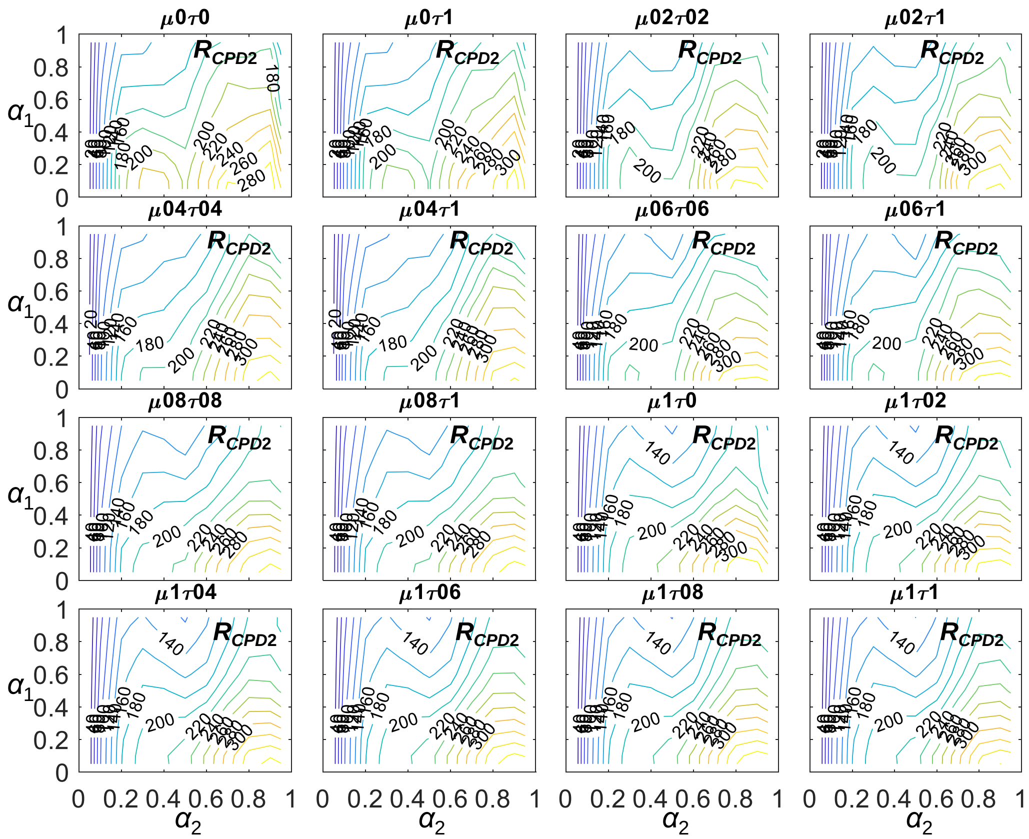


Figure 10.3.16 The R_{CPD2} calculated from the NLHRA with BCJ-L2 ground motion

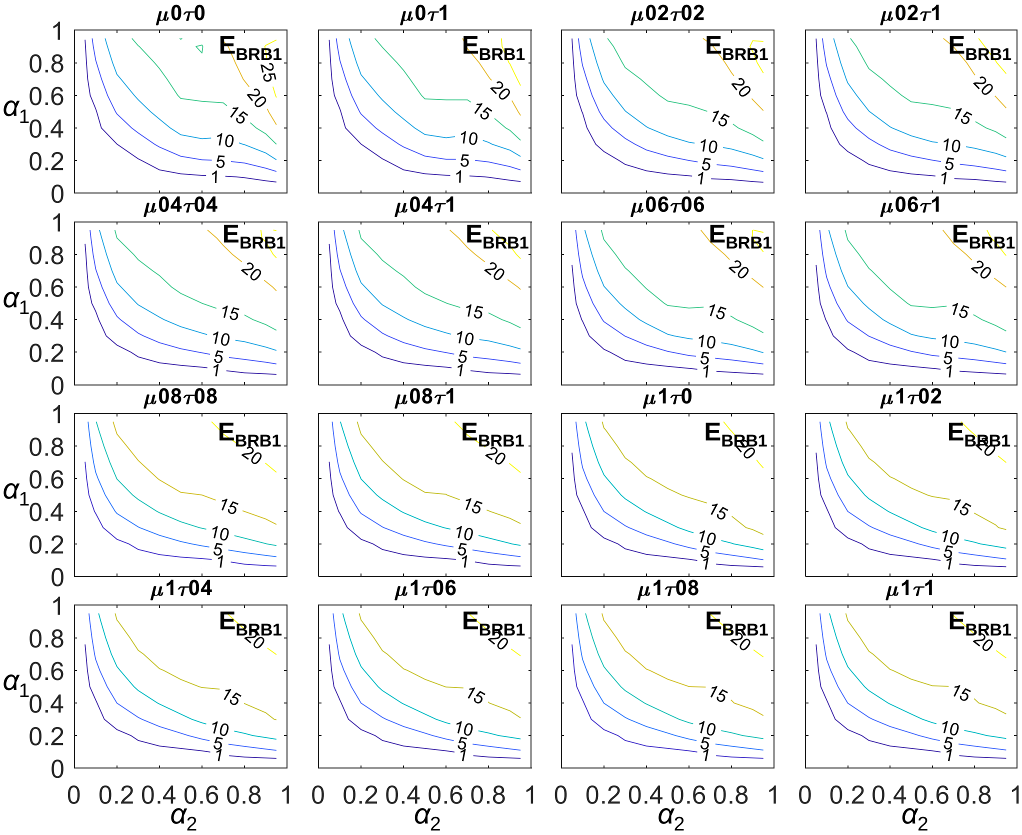


Figure 10.3.17 The E_{BRB1} calculated from the NLHRA with BCJ-L2 ground motion (unit: %)

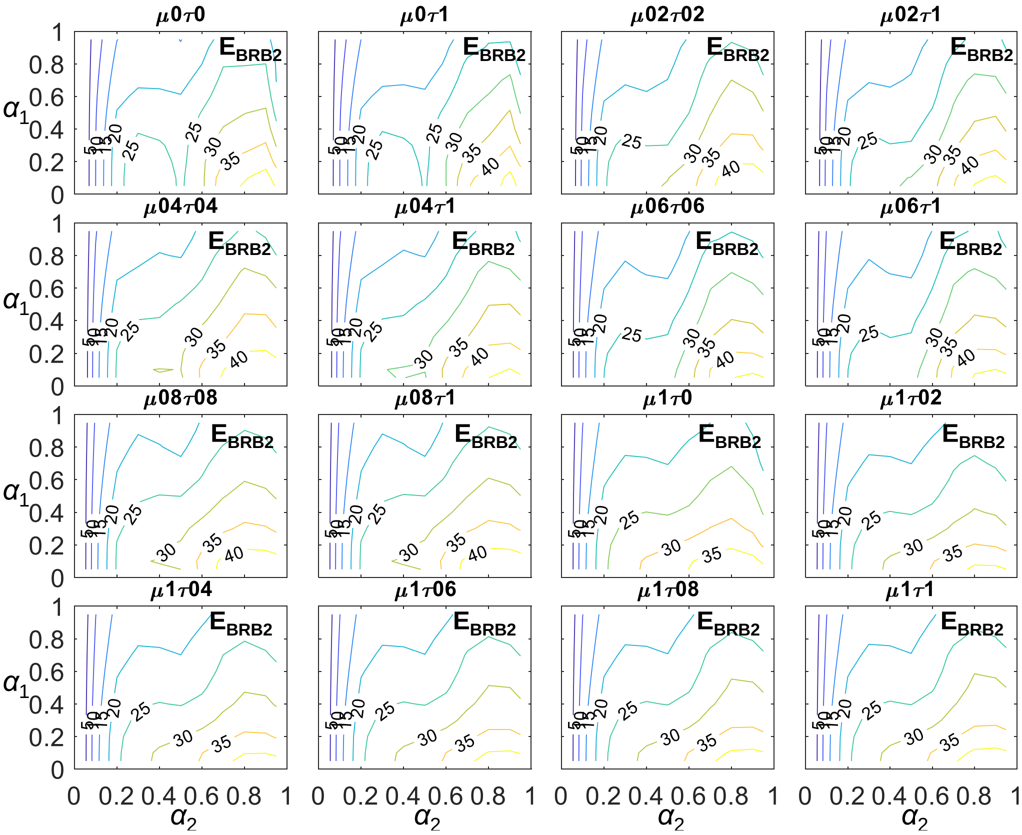


Figure 10.3.18 The E_{BRB2} calculated from the NLHRA with BCJ-L2 ground motion (unit: %)

10.4 SUMMARY

This chapter investigates the influences on seismic response when the perimeter column develops different axial tensile and compressive stiffness, and when *EI* and perimeter column cross-sectional area vary with building elevation, which is closer to the reality of structure design practices. The modified method is proposed in order to estimate seismic response on structures when *EI* and k_c are not uniform. The summaries of this chapter can be drawn as follows:

(1) When the perimeter column develops different axial tensile and compressive stiffness, the proposed analytical model has to be modified. When greater compressive axial stiffness is considered, such as the CFT perimeter columns, the system stiffness could increase and amplify seismic demands. However, the increased perimeter column compressive axial stiffness enhances the outrigger effect and could better mitigate seismic response. Therefore, in order to increase the accuracy of seismic performance evaluation and to achieve a conservative design result, it is recommended to include the effect of different axial compressive and tensile stiffness in the design practice.

(2) The effect of different tensile and compressive perimeter column axial stiffness would affect the maximum seismic response. However, it would not significantly affect the optimal outrigger elevation in order to minimize $\theta_{\max}$ and $\gamma_{\max}$ for both the single and dual BRB-outrigger systems discussed in Chapter 7.

(3) The effects when core structure rigidity and perimeter column's cross-sectional area vary with building elevation were investigated. The smaller μ and τ values decrease outrigger effect and are less efficient in mitigating seismic response. In addition, the decrease of μ value amplifies the overall $\theta_{\max}$ and $\gamma_{\max}$ responses. However, the decrease of τ value amplifies the $\theta_{\max}$ and $\gamma_{\max}$ responses only when α and α_2 are larger than 0.5 for single and dual BRB-outrigger systems. Based on the analysis results, the case when both μ and τ equal to 1 could achieve best seismic performance, although the reaction forces such as $V_{c,\max}$ and $M_{c,\max}$ could be slightly amplified due to shorter fundamental vibration period. However, in the design practices, the values of μ and τ should be smaller than 1 for economical design. The analysis results of using μ and τ equal to 1 could underestimate the maximum seismic responses. Therefore, it is recommended to use the values of μ and τ close to the real design for a conservative design result.

(4) Based on the analysis results, the values of μ and τ would not significantly affect the optimal outrigger elevations in order to minimize θ_{max} and γ_{max} for both the single and dual BRB-outrigger systems.

11

CONCLUSIONS

This research investigated the seismic behavior of structure employing either single or dual BRB-outrigger system through numerical analysis. The optimal design of BRB-outrigger system in order to minimize seismic responses is proposed. Chapter 1 and Chapter 2 of this thesis introduce the background and related past researches. The key summaries of each chapter and the conclusions of this thesis are summarized as follows:

Chapter 3: Analytical models

The structure with BRB-outriggers is simplified for the purpose of performing the parametric study. The simplified structure and analytical model details are introduced in this chapter. The summaries of chapter 3 are as follows:

(1) The simplified structure is constructed by assuming a cantilever column with attached rotational springs to represent the core structure and BRB-outriggers, respectively. The dynamic characteristics of the simplified structure are solved by using D'Alembert's principle. The mass and core structure flexural rigidity are uniformly distributed along with the core structure height. This simplified structure and the solution procedure are known as UM model.

(2) The simplified structure constructed by using OpenSees model in order to perform NLRHA. The mass is concentrated on mass nodes, which are evenly distributed along with the core structure height with a spacing of 1 m. This analytical is known as DM model.

(3) The MBM model, which follows all structure details, is constructed using OpenSees. The analysis results obtained from using MBM model show good agreements with the analysis results from using UM and MBM models.

(4) Two sets of dimensionless parameters are proposed for single and dual BRB-outrigger systems for the purpose of the parametric study. The outrigger stiffness parameter defines the magnitude of how outrigger affects the structure. The BRB stiffness parameter defines the relationships between BRB axial stiffness, perimeter column axial stiffness, and outrigger truss flexural stiffness.

Chapter 4: Analysis methods

The spectral analysis (SA) and nonlinear response history analysis (NLRHA) procedures are used to study the seismic behavior of BRB-outrigger systems. The summaries of this chapter are as follows:

(1) In the SA procedure, the effect of reduction in the response spectrum because of the increased damping ratio due to BRB yielding is incorporated by using equivalent damping. The modal pushover analysis (MPA) is applied in order to obtain the base shear and roof displacement relationship. The responses of the first four modes are calculated separately and then superposed using SRSS method.

(2) The NLARH is performed using eight ground motions, which are scaled according to the design spectrum used in SA. The NLRHA and SA show good agreements.

Chapter 5: Programming for parametric study

In order to perform a large amount of SA and NLRHA, a computer program is developed using C++ programming. The program generates OpenSees script based on the input parameters and runs SA and NLRHA automatically.

Chapter 6: Preliminary analysis

This chapter presents the preliminary analysis results which are necessary for the purpose of parametric analysis. The summaries of this chapter are as follows:

(1) The BRB yield deformation is crucial as it determines when the BRB starts dissipating energy. Too small BRB yield deformation can cause an early fracture, and too large BRB yield deformation can result in low energy dissipation efficiency.

(2) In this research, it is defined that the BRBs yield when roof drift reaches θ_r in the deformed shape calculated from the combination of the first four mode shapes using SRSS method. The θ_r is the maximum elastic deformation limit.

(3) The outrigger truss should be stiff enough to provide sufficient axial deformation demand on the BRB. The analysis results show that when the flexural stiffness of the outrigger truss is more than 110% of the BRB axial stiffness, the influence of changing outrigger truss flexural stiffness is marginal.

(4) The drop of the fundamental vibration period if compared with the core structure without outrigger effect is used to judge the magnitude of outrigger effect. For the single BRB-outrigger system, the fundamental vibration drops the most when α is between 0.5 and 0.8. For the dual BRB-outrigger system, the fundamental

vibration period drops the most when α_2 is between 0.6 and 0.8, and when α_1 is between 0.8 and 0.9.

(5) The outrigger elevation has a significant relation to the energy dissipated by BRB-outrigger system. The energy dissipation efficiency for the first two mode responses are investigated using SA procedure. For the single BRB-outrigger system, the energy dissipation is the most efficient when α is 0.6~0.8 and 0.9 for the 1st and the 2nd more responses, respectively.

(6) For the energy dissipation efficiency of the upper outrigger in a dual BRB-outrigger system, it is the most efficient when α_1 is 0 and 0.5 for the 1st and 2nd mode responses, respectively, and when α_2 is 0.6~0.8 and 0.9 for the 1st and 2nd mode responses, respectively. For the energy dissipation efficiency of the lower outrigger in a dual BRB-outrigger system, it is the most efficient when α_1 is 0.5~0.7 and 0.5 for the 1st and 2nd mode responses, respectively, and when α_2 is 1 and 0.5 for the 1st and 2nd more responses, respectively.

Chapter 7: Analysis results for optimal design

This chapter presents the analysis results for the purpose of optimal design, the summaries of this chapter are as follows:

(1) The maximum roof drift ($\theta_{\max}$), the maximum inter-story drift ($\gamma_{\max}$), the maximum overturning moment at core structure base ($M_{c,\max}$), and the maximum axial force in the perimeter column ($C_{1,\max}$) are used as indicators to indicate seismic response.

(2) For the single BRB-outrigger system, when α is 0.5~0.8, the $\theta_{\max}$ is best reduced, however, the $C_{1,\max}$ also reaches its maximum value. When α is 0.7~0.8, the $\gamma_{\max}$, $M_{c,\max}$, and the maximum roof acceleration can be best reduced.

(3) For the single BRB-outrigger system, the optimal BRB parameters (R_{db} and R_{d2c}) in order to minimize seismic responses are between 1 and 5. The reductions in seismic response stop increasing when the values of R_{db} and R_{dc} are greater than 5, and the $C_{1,\max}$ grows almost proportionally with the increasing R_{db} and R_{dc} values. The optimal R_{db} and R_{dc} values are smaller when the outrigger effect factor (S_{bc07}) is greater.

(4) For the dual BRB-outrigger system, when α_2 is 0.7~0.8, the seismic response can be best reduced. The $\theta_{\max}$ and $\gamma_{\max}$ can be best reduced when α_1 is 0.4~0.7, and the

$M_{c,\max}$ can be best reduced when α_1 is 0.2~0.6. When α_2 is lower than 0.4, the presence of the lower BRB-outrigger has very less contribution in reducing seismic response.

(5) For the dual BRB-outrigger system, the seismic response decreases with increasing R_{d2c} value. However, the reduction in seismic response stops growing when R_{d2c} is greater than approximate 1.5. The optimal R_{d2c} ranges between 0.5 and 1.5 and is smaller when the outrigger effect parameter ($S_{bc2,07}$) is greater.

(6) For the design purpose, the first priority is to select α (single BRB-outrigger), α_1, and α_2 (dual BRB-outrigger) in their optimal ranges and to target the S_{bc07} (single BRB-outrigger) or $S_{bc2,07}$ (dual BRB-outrigger) value as large as possible. Then, select R_{db}, R_{dc} (single BRB-outrigger), and R_{d2c} (dual BRB-outrigger) within their optimal range. The values of R_{dt} and R_{kd} can be used to fine-tune the design.

Chapter 8: Design recommendation and design examples

This chapter introduces the design recommendations for BRB-outrigger system based on the analysis results. Design examples are used to demonstrate the use of design charts. The summaries of this chapter are as follows:

(1) A step-by-step design procedure is proposed. The design charts can assist the designer in selecting proper lower BRB-outrigger elevation.

(2) When the single and dual BRB-outrigger systems are compared and when their outrigger effect parameters (S_{bc07} and $S_{bc2,07}$) are similar, the dual BRB-outrigger system generally results in 5% to 10% response smaller than the single BRB-outrigger system.

(3) Considering both the cost and seismic performance, the single BRB-outrigger system should be a more economical solution in controlling seismic response. However, the dual BRB-outrigger system can be a better alternative than single BRB-outrigger system when the overturning moment at core structure base is too large, when the inter-story response is too large below the upper BRB-outrigger, and when the required BRB axial stiffness or the outrigger truss flexural stiffness is too large to design.

Chapter 9: BRB-outrigger configurations

Three BRB-outrigger configurations in order to meet different architectural requirements are discussed. The summaries of this chapter are as follows:

(1) The outrigger effect parameter and BRB stiffness parameter are modified in order to fit different BRB-outrigger configurations. According to the analysis results, the optimal design of BRB-outrigger system would not be affected by the type of BRB-outrigger configuration.

(2) From the viewpoint of BRB design, the OB and GB outrigger configurations are suitable when the outrigger span is short, the OB and BT outrigger configurations best utilize the building interior space. The GB outrigger is the most economical solution as the outrigger truss members are not necessary.

(3) All the OB, BT, and GB configurations are capable to achieve the desired seismic response. The designers can select suitable BRB-outrigger configurations to fulfill both architectural requirements and economical designs.

Chapter 10: Application of BRB-outrigger

This chapter discusses the effects when core structure flexural rigidity and perimeter column axial stiffness vary with building height on seismic response. The summaries of this chapter are as follows:

(1) When the perimeter column is composed of a steel box with infill concrete, its axial stiffness in tension and compression are different. Based on analysis results, if compared with the case without considering the additional contribution from infill concrete, while the larger compressive axial stiffness of perimeter column is considered, the maximum seismic response can be further reduced and the optimal outrigger elevation is not significantly affected.

(2) The greater amount of decreases in EI and A_{pc} along with the building height, the less significant outrigger effect, and the seismic response is underestimated if compared with the case when EI and A_{pc} are assumed to be uniformly distributed along with the building height. Based on the analysis results, the optimal outrigger elevations are very similar no matter when EI and A_{pc} are linearly or uniformly distributed along with the building height. For the design purpose, the actual distribution of EI and A_{pc} should be considered for conservative design purpose.

Publications related to this doctoral thesis

Journal papers:

1. **Pao-Chun Lin,** Toru Takeuchi, and Ryota Matsui, "Seismic performance evaluation of single damped-outrigger system incorporating buckling-restrained braces", *Earthquake Engineering and Structural Dynamics*, 2018; 47(12): 2343-2365. (*chapters* 2 *to* 8)
2. **Pao-Chun Lin**, Toru Takeuchi, and Ryota Matsui, "Optimal design of multiple damped-outrigger system incorporating buckling-restrained braces", *Engineering Structures*, 2019; 194: 441-457. (*chapters* 2 *to* 8)
3. **Pao-Chun Lin** and Toru Takeuchi, "Seismic performance of buckling-restrained brace outrigger system in various configurations", *Japan Architectural Review*, 2019, accepted. (*chapter* 9)

International conference papers:

1. **Pao-Chun Lin**, Toru Takeuchi, and Ryota Matsui, "Seismic performance of outrigger system incorporating buckling-restrained braces", *Proceedings of the* 7*th Asia Conference on Earthquake Engineering*, Bangkok, Thailand, 2018.11.
2. **Pao-Chun Lin**, Toru Takeuchi, and Ryota Matsui, "Seismic performance of damped-outrigger system incorporating buckling-restrained braces", *Proceedings of* 12*th Pacific Structural Steel Conference*, Tokyo, Japan, 2019.11.

Domestic conference papers:

1. **Pao-Chun Lin**, Toru Takeuchi, and Ryota Matsui, "Seismic evaluation of damped-outrigger incorporating buckling-restrained braces", *Proceedings of Architecture Institute Japan Annual Conference*, paper no. 22428-22429, Sendai, Japan, 2018.9.
2. **Pao-Chun Lin** and Toru Takeuchi, "Seismic performance of BRB-outrigger system in various configurations", *Proceedings of Architecture Institute Japan Annual Conference*, paper no. 22578-22579, Kanazawa, Japan, 2019.9.

Other publications

Journal papers:

1. Chao-Hsien Li, Keh-Chyuan Tsai, Lei Su, **Pao-Chun Lin**, and Te-Hung Lin, "Experimental investigations on seismic behavior and design of bottom vertical boundary elements in multi-story steel plate shear walls", *Earthquake Engineering and Structural Dynamics*, 2018; 47(14): 2777-2801.
2. **Pao-Chun Lin**, Keh-Chyuan Tsai, Chieh-An Chang, Yu-Yun Hsiao, and An-Chien Wu, "Seismic design and testing of buckling-restrained braces with a thin profile", *Earthquake Engineering and Structural Dynamics*, 2016; 45(3): 339-358.
3. **Pao-Chun Lin**, Keh-Chyuan Tsai, An-Chien Wu, Ming-Chieh Chuang, Chao-Hsien Li, and Kung-Juin Wang, "Seismic design and experiment of single and coupled corner gusset connections in a full-scale two-story buckling-restrained braced frame", *Earthquake Engineering and Structural Dynamics*, 2015; 44(13): 2177-2198.
4. Ming-Chieh Chuang, Keh-Chyuan Tsai, **Pao-Chun Lin**, and An-Chien Wu, "Critical limit states in seismic buckling-restrained brace and connection designs", *Earthquake Engineering and Structural Dynamics*, 2015; 44(10): 1559-1579.
5. **Pao-Chun Lin**, Keh-Chyuan Tsai, An-Chien Wu, and Ming-Chieh Chuang, "Seismic design and test of gusset connections for buckling-restrained braced frames", *Earthquake Engineering and Structural Dynamics*, 2014; 43(4): 565-587.
6. Christoph Mahrenholtz, **Pao-Chun Lin**, An-Chien Wu, Keh-Chyuan Tsai, Shyh-Jiann Hwang, Ruei-Yan Lin, and Muhammad Y. Bhayusukma, "Retrofit of reinforced concrete frames with buckling-restrained braces", *Earthquake Engineering and Structural Dynamics*, 2014; 44(1): 59-78.
7. An-Chien Wu, **Pao-Chun Lin**, and Keh-Chyuan Tsai, "High-mode buckling responses of buckling-restrained brace core plates", *Earthquake Engineering and Structural Dynamics*, 2014; 43(3): 375-393.
8. Keh-Chyuan Tsai, An-Chien Wu, Chih-Yu Wei, **Pao-Chun Lin**, Ming-Chieh Chuang, and Yi-Jer Yu, "Welded end-slot connection and debonding layers for buckling-restrained braces", *Earthquake Engineering and Structural Dynamics*, 2014; 43(12): 1785-1807.

9. Ching-Yi Tsai, Keh-Chyuan Tsai, **Pao-Chun Lin**, Wai-Hang Ao, Charles W. Roeder, Stephen A. Mahin, Chih-Han Lin, Yi-Jer Yu, Kung-Juin Wang, An-Chien Wu, Jia-Chian Chen, and Te-Hung Lin, "Seismic design and hybrid tests of a full-scale three-story concentrically braced frame using in-plane buckling braces", *Earthquake Spectra*, 2013; 29(3): 1043-1067.
10. **Pao-Chun Lin**, Keh-Chyuan Tsai, Kung-Juin Wang, Yi-Jer Yu, Chih-Yu Wei, An-Chien Wu, Ching-Yi Tsai, Chih-Han Lin, Jia-Chian Chen, Andreas H. Schellenberg, Stephen A. Mahin, and Charles W. Roeder, "Seismic design and hybrid tests of a full-scale three-story buckling-restrained braced frame using welded end connections and thin profile", *Earthquake Engineering and Structural Dynamics*, 2012; 41(5):1001-1020.

International conference papers:

1. **Pao-Chun Lin**, Toru Takeuchi, Ryota Matsui and Ben Sitler, "Seismic design of buckling-restrained in preventing local bulging failure", *Proceedings of the* 9*th* *International Conference on Behavior of Steel Structures in Seismic Areas*, Christchurch, New Zealand, 2018.2.
2. **Pao-Chun Lin**, Toru Takeuchi, Ryota Matsui and Ben Sitler, "Seismic design of buckling-restrained brace in preventing the local bulging failure", *Proceedings of Architecture Institute Japan Annual Conference*, paper no. 22592, Hiroshima, Japan, 2017.9.
3. **Pao-Chun Lin**, Toru Takeuchi, Ryota Matsui and Ben Sitler, "Seismic design of buckling-restrained brace in preventing the local bulging failure", *Proceedings of the* 4*th* *Joint Workshop on Building/Civil Engineering between Tongji and Tokyo Tech*, Shanghai, China, 2017.6.
4. **Pao-Chun Lin**, An-Chien Wu, Chao-Hsien Li, Kung-Juin Wang and Keh-Chyuan Tsai, "Design and testing of buckling-restrained brace gusset connections", *Proceedings of the* 5*th* *Asia Conference on Earthquake Engineering*, Taipei, Taiwan, 2014.
5. Christoph Mahrenholtz, **Pao-Chun Lin**, An-Chien Wu, Keh-Chyuan Tsai and Shyh-Jiann Hwang, "Seismic retrofit of reinforced concrete frame with buckling-restrained braces", *Proceedings of the* 5*th* *Asia Conference on Earthquake Engineering*, Taipei, Taiwan, 2014.
6. **Pao-Chun Lin**, Keh-Chyuan Tsai, Ming-Chieh Chuang and An-Chien Wu, "Seismic responses of buckling-restrained brace to gusset connections in framed structures", *Proceedings of the* 5*th* *International Conference on Advances in Experimental Structural Engineering*, Taipei, Taiwan, 2013.
7. **Pao-Chun Lin**, Keh-Chyuan Tsai, Ming-Chieh Chuang and An-Chien Wu, "Seismic design and test of buckling-restrained brace connections", *Proceedings of the* 10*th* *International Conference on Urban Earthquake Engineering*, Tokyo, Japan, 2013.
8. An-Chien Wu, **Pao-Chun Lin** and Keh-Chyuan Tsai, "A type of buckling restrained brace for convenient inspection and replacement", *Proceedings of the* 15*th* *World Conference on Earthquake Engineering*, Lisbon, Portugal, 2012.
9. **Pao-Chun Lin**, Keh-Chyuan Tsai, Kung-Juin Wang, Chih-Yu Wei, An-Chien Wu, Ching-Yi Tsai and Jia-Chian Chen, "Seismic performance of a gull scale three-story buckling-restrained braced frame specimen", *Proceedings of the* 6*th* *International Symposium on Steel Structures*, Seoul, Korea, 2011.
10. An-Chien Wu, Chih-Yu Wei, **Pao-Chun Lin**, and Keh-Chyuan Tsai, "Experimental investigations of welded end-slot connection and unbonding layers for buckling restrained braces", *Proceedings of the* 6*th* *International Symposium on Steel Structures*, Seoul, Korea, 2011.

Honor and Distinctions

1. 2019 年日本建築学会著作賞: 竹内徹, 和田章, 松井良太, Sitler Ben, Pao-Chun Lin, Sutcu Fatih, 坂田弘安, 曲哲, "Buckling-Restrained Braces and Applications", 日本免震構造協会, 2019/6.
2. Research fellowship of the Japan Society for the Promotion of Science (日本学術振興会), DC2, 4/2018-3/2020.
3. Scholarship: Japan-Taiwan Exchange Association (公益財団法人日本台湾交流協会), 4/2017-3/2018.

Appendix A

C++ script of computer program for parametric study

CONTENT

main.cpp…… A-2
getL_vec()…… A-45
MakeOpenseesTCL()…… A-51
Section.h…… A-64
mode.h…… A-67
Spectrum.h…… A-70
matrix.h…… A-74
matrix.cpp…… A-75

```cpp
#include<iostream>
#include<cmath>
#include<fstream>
#include<vector>
#include<string>
#include<time.h>
#include<map>
#include"Spectrum.h"
#include"Section.h"
#include"matrix.h"
#include"mode.h"

int seg_num, sol_num, iteration_num, iteration_limit, IfDoSA_NUM, IfDoSA_OpenseesApproah, DesignSpecType, BRByieldCase;
int IfDoNLRHA, tired, SAmode, ifpause, inputType, MPAmodesNUM, MPAmodeNow, DesignSpecOREQspec, IfAssignUy, SRSSmodeShapeType;
int IfRemoveOpenseesResultFile, IfDisplayDetail, IfEQT1scale, IfwriteModeShapeToTxtFile;
double YieldDrift, h, m, EIb, EIt, lt, slopeEI, h0, kc, dx, MPAtargetdisp, MPAincr, MPAstep, Es;
double OutriggerElevation, tolerance, dt, EQt, EQscalefactor, T1, T1EFF, alpha_in_Dh, dL, value, KE, K1, K2, MassSpacing;
double pi = 3.141592654;
std::string tcl_file_name, EQname;
std::vector<double> EQtime, EQacc, kt, kd, kd_plastic, kc_seg, kb, kg, Sbc, Rdt, Rdc, Rdb, yieldDisp, disp, force;
std::vector<double> thetaY, alpha, OutriggerElevationVec, Ytop_mode, uBRBy, uBRByOpensees, Es_vec, Ed;
std::vector<double> thetaYOpensees, Fy_BRB, BRBductilityOpensees, BRBPostYieldStiffnessRatio;
std::vector<Section> segment_vec;
std::vector<mode> mode_vec;
std::vector<double> L_vec, Lp_vec, CPD, BRBmaxDeformation, BRBenergy;
std::vector<double> CoreNodeElevation;
std::vector<int> CoreMassID, yieldStep;
std::vector<std::vector<double>> Ed_vec;
Spectrum DesignSpec, EQspec;

double EI(double x) {
    return (EIt - EIb) / h*x + EIb;
}

double heq_p(int i) { // i, the ith element
    double x1, x2, x3, x4, x5, x6, x7, x8, x9, x10;
    double y1, y2, y3, y4, y5, y6, y7, y8, y9, y10;
    if (disp[i] < yieldDisp[0]) { // elastic case
        x1 = 0;
        x2 = disp[i];
        x3 = disp[i];
        x4 = disp[i];
        x5 = -1 * disp[i];
        x6 = -1 * disp[i];
        x7 = -1 * disp[i];
        x8 = 0;
        x9 = 0;
        x10 = 0;
```

```
        y1 = 0;
        y2 = y1 + (x2 - x1)*KE;
        y3 = y2 + (x3 - x2)*K1;
        y4 = y3 + (x4 - x3)*K2;
        y5 = y4 + (x5 - x4)*KE;
        y6 = y5 + (x6 - x5)*K1;
        y7 = y6 + (x7 - x6)*K2;
        y8 = y7 + (x8 - x7)*KE;
        y9 = y8 + (x9 - x8)*K1;
        y10 = y9 + (x10 - x9)*K2;
    }
    else if (disp[i] >= yieldDisp[0] && disp[i] <= yieldDisp[1]) {
        x1 = 0;
        x2 = yieldDisp[0];
        x3 = disp[i];
        x4 = disp[i];
        x5 = x4 - 2 * yieldDisp[0];
        x6 = -1 * disp[i];
        x7 = -1 * disp[i];
        x8 = x7 + 2 * yieldDisp[0];
        x9 = 0;
        x10 = 0;
        if (x8 >= 0) {
            x8 = 0;
        }
        y1 = 0;
        y2 = y1 + (x2 - x1)*KE;
        y3 = y2 + (x3 - x2)*K1;
        y4 = y3 + (x4 - x3)*K2;
        y5 = y4 + (x5 - x4)*KE;
        y6 = y5 + (x6 - x5)*K1;
        y7 = y6 + (x7 - x6)*K2;
        y8 = y7 + (x8 - x7)*KE;
        y9 = y8 + (x9 - x8)*K1;
        y10 = y9 + (x10 - x9)*K2;
    }
    else {
        x1 = 0;
        y1 = 0;
        x2 = yieldDisp[0];
        y2 = x2 * KE;
        x3 = yieldDisp[1];
        y3 = y2 + (x3 - x2)*K1;
        x4 = disp[i];
        y4 = y3 + (x4 - x3)*K2;
        x5 = x4 - 2 * yieldDisp[0];
        y5 = y4 + (x5 - x4)*KE;
        x7 = -1 * disp[i];
        y7 = -1 * y4;
        x6 = (-1 * K1*x5 + y5 + K2 * x7 - y7) / (K2 - K1);
        y6 = K1 * (x6 - x5) + y5;
        x8 = x7 + 2 * yieldDisp[0];
        y8 = y7 + (x8 - x7)*KE;
        x9 = (-1 * K2*x4 + y4 - y8 + K1 * x8) / (K1 - K2);
        y9 = y8 + K1 * (x9 - x8);
        x10 = 0;
```

```
        y10 = y9 + (x10 - x9)*K2;

        if (x8 >= 0) {
            x8 = 0;
            x9 = 0;
            y8 = y7 + (x8 - x7)*KE;
            y9 = y8 + K1 * (x9 - x8);
        }
        if (x9 >= 0) {
            x9 = 0;
            y9 = y8 + K1 * (x9 - x8);
        }
    }

    double Ed = 0;
    Ed += (y1 + y2)*(x2 - x1)*0.5;
    Ed += (y2 + y3)*(x3 - x2)*0.5;
    Ed += (y3 + y4)*(x4 - x3)*0.5;
    Ed += (y4 + y5)*(x5 - x4)*0.5;
    Ed += (y5 + y6)*(x6 - x5)*0.5;
    Ed += (y6 + y7)*(x7 - x6)*0.5;
    Ed += (y7 + y8)*(x8 - x7)*0.5;
    Ed += (y8 + y9)*(x9 - x8)*0.5;
    Ed += (y9 + y10)*(x10 - x9)*0.5;

    double Es = force[i] * disp[i] * 0.5;
    return std::abs(Ed / 4 / pi / Es);
}

matrix CreateCmatrix() {
    matrix C(seg_num - 1, seg_num - 1);
    for (int k = 0; k != seg_num - 1; k++) {
        for (int j = 0; j != seg_num - 1; j++) {
            if (k == j && k != seg_num - 2) {
                C.assign(k, j, (kc_seg[k] + kc_seg[k + 1]));
            }
            else if (k == j && k == seg_num - 2) {
                C.assign(k, j, kc_seg[k]);
            }
            else if (k - j == 1) {
                C.assign(k, j, -1 * kc_seg[k]);
                C.assign(j, k, -1 * kc_seg[k]);
            }
        }
    }
    return C;
}

matrix CreateDmatrix(std::vector<double> kdSRSS) {
    matrix D(seg_num - 1, seg_num - 1);
    for (int k = 0; k != seg_num - 1; k++) {
        for (int j = 0; j != seg_num - 1; j++) {
            if (k == j && k != seg_num - 2) {
                D.assign(k, j, (1 + (kc_seg[k] + kc_seg[k + 1])*(1 / kdSRSS[k] + 1 / kt[k])) / lt);
            }
```

```
            else if (k == j && k == seg_num - 2) {
                D.assign(k, j, (1 + kc_seg[k] * (1 / kdSRSS[k] + 1 / kt[k])) / lt);
            }
            else if (k - j == 1) {
                D.assign(k, j, (-1 * kc_seg[k] * (1 / kdSRSS[k] + 1 / kt[k])) / lt);
            }
            else if (j - k == 1) {
                D.assign(k, j, (-1 * kc_seg[j] * (1 / kdSRSS[k] + 1 / kt[k])) / lt);
            }
        }
    }
    return D;
}

int main()
{
    time_t StartTime, EndTime;
    StartTime = time(NULL);
    remove("status.txt");
    std::string firstline;
    std::ifstream insetup;
    insetup.open("C:\\Dropbox\\TokyoTech\\ProgramRun\\OutriggerEXEsetup\\OutriggerEXEsetupV6.txt");
    if (!insetup) {
        std::cerr << "[error][insetup]" << std::endl;
        system("pause");
        return -1;
    }

    IfDisplayDetail = 1; //1: yes, 0: no
    IfDoSA_NUM = 1; // 1: yes, 0: no
    IfDoSA_OpenseesApproah = 0; // 1:yes, 0: no
    SAmode = 0; // use the nth mode to perform spectral analysis
                        // 0: consider all the modes
    IfDoNLRHA = 1;  // 1: yes, 0: no
    IfRemoveOpenseesResultFile = 1; // 1: yes, 0: no
    IfEQT1scale = 0; // 1:yes, 0: no
    h0 = 0.02; // the inherent damping ratio
    alpha_in_Dh = 75.0; // 75 for artificial eq, 25 for real eq
    iteration_limit = 100;
    dL = 0.01;
    dx = 0.01;
    IfwriteModeShapeToTxtFile = 0;

    std::string sss;
    insetup >> sss;
    insetup >> IfDisplayDetail;
    insetup >> sss;
    insetup >> IfDoSA_NUM;
    insetup >> sss;
    insetup >> IfDoSA_OpenseesApproah;
    insetup >> sss;
    insetup >> SAmode;
```

```
    insetup >> sss;
    insetup >> IfDoNLRHA;
    insetup >> sss;
    insetup >> IfRemoveOpenseesResultFile;
    insetup >> sss;
    insetup >> IfEQT1scale;
    insetup >> sss;
    insetup >> h0;
    insetup >> sss;
    insetup >> alpha_in_Dh;
    insetup >> sss;
    insetup >> iteration_limit;
    insetup >> sss;
    insetup >> ifpause;
    insetup >> sss;
    insetup >> DesignSpecType;
    insetup >> sss;
    insetup >> inputType;
    insetup >> sss;
    insetup >> MPAmodesNUM;
    insetup >> sss;
    insetup >> MPAstep;
    insetup >> sss;
    insetup >> DesignSpecOREQspec;
    insetup >> sss;
    insetup >> BRByieldCase;
    insetup >> sss;
    insetup >> IfAssignUy;
    insetup >> sss;
    insetup >> MassSpacing;
    insetup >> sss;
    insetup >> SRSSmodeShapeType;

    if (IfDisplayDetail == 1) {
        std::cout << "IfDisplayDetail=" << IfDisplayDetail << std::endl;
        std::cout << "IfDoSA_NUM=" << IfDoSA_NUM << std::endl;
        std::cout << "IfDoSA_OpenseesApproah=" << IfDoSA_OpenseesApproah << std::endl;
        std::cout << "SAmode=" << SAmode << std::endl;
        std::cout << "IfDoNLRHA=" << IfDoNLRHA << std::endl;
        std::cout << "IfRemoveOpenseesResultFile=" << IfRemoveOpenseesResultFile << std::endl;
        std::cout << "IfEQT1scale=" << IfEQT1scale << std::endl;
        std::cout << "h0=" << h0 << std::endl;
        std::cout << "alpha_in_Dh=" << alpha_in_Dh << std::endl;
        std::cout << "iteration_limit=" << iteration_limit << std::endl;
        std::cout << "ifpause=" << ifpause << std::endl;
        std::cout << "DesignSpecType=" << DesignSpecType << std::endl;
        std::cout << "inputType=" << inputType << std::endl;
    }

    //Spectrum DesignSpec, EQspec;
    std::cout << "reading spectrum file..." << std::endl;
    if (DesignSpecType == 1) {
        if (DesignSpecOREQspec == 1) {
            DesignSpec.ReadSpaFromFile("C:\\Dropbox\\TokyoTech\\ProgramRun\
```

```
            \OutriggerEXEsetup\\AccelerationSpectrum.txt"); // file name: AccelerationSpectrum.txt
        }
        else if (DesignSpecOREQspec == 2) {
            DesignSpec.ReadSpaFromFile("C:\\Dropbox\\TokyoTech\\ProgramRun\\OutriggerEXEsetup\\SP_BCJL2.txt");
            IfEQT1scale = 0;
        }
        if (IfDisplayDetail == 1) {
            std::cout << "read from Spa" << std::endl;
        }
    }
    else if (DesignSpecType == 2) {
        DesignSpec.ReadSpvFromFile("C:\\Dropbox\\TokyoTech\\ProgramRun\\OutriggerEXEsetup\\VelocitySpectrum.txt"); // file name: AccelerationSpectrum.txt
        if (IfDisplayDetail == 1) {
            std::cout << "read form Spv" << std::endl;
        }
    }

    std::ofstream SANoutput, SAOPoutput, NLRHAoutput, MPAoutput, SAOPoutput1, InterStorySA, MomentSA, InterStoryNLRHA, DriftSA, DriftNLRHA, MomentNLRHA;
    if (IfDoSA_NUM == 1) {
        SANoutput.open("SA_NUM_result.txt");
        if (!SANoutput) {
            std::cerr << "[error][SANoutput.open]" << std::endl;
            system("pause");
            return -1;
        }
    }
    if (IfDoSA_OpenseesApproah == 1) {
        SAOPoutput.open("SA_OP_result.txt", std::ios::out);
        SAOPoutput1.open("SA_OP_result1.txt");
        MPAoutput.open("MPArecord.txt");
        InterStorySA.open("SA_InterStory.txt");
        MomentSA.open("SA_Moment.txt");
        DriftSA.open("SA_Drift.txt");
        if (!SAOPoutput || !MPAoutput || !SAOPoutput1 || !InterStorySA || !DriftSA || !MomentSA) {
            std::cerr << "[error][SAOPoutput.open][MPAoutput.open][SAOPoutput.open][InterStorySA.open][DriftSA.open][MomentSA.open]" << std::endl;
            system("pause");
            return -1;
        }
    }
    if (IfDoNLRHA == 1) {
        NLRHAoutput.open("NLRHAresult.txt");
        InterStoryNLRHA.open("NLRHA_InterStory.txt");
        DriftNLRHA.open("NLRHA_Drift.txt");
        MomentNLRHA.open("NLRHA_moment.txt");
        if (!NLRHAoutput || !InterStoryNLRHA || !DriftNLRHA || !MomentNLRHA) {
            std::cerr << "[error][SANoutput.open][InterStoryNLRHA.open]
```

```
                    [DriftNLRHA.open][MomentNLRHA.open]" << std::endl;
                system("pause");
                return -1;
            }
        }

        std::ifstream input;
        if (inputType == 1) {
            input.open("inp.txt");
        }
        else if (inputType == 2) {
            input.open("inpA.txt");
        }

        if (!input) {
            std::cerr << "[error][ifstream: input the inp.txt]";
            system("pause");
            return -1;
        }
        std::getline(input, firstline);
        tired = 1;

    // WHILE LOOP STARTS
        while (input >> seg_num) {
            std::ofstream runtime;
            runtime.open("runtime.txt");
            EndTime = time(NULL);
            runtime << "start: " << EndTime - StartTime << " sec." << std::endl;
            EQtime.resize(0);
            EQacc.resize(0);
            kt.resize(0);
            kd.resize(0);
            kd_plastic.resize(0);
            kc_seg.resize(0);
            kb.resize(0);
            kg.resize(0);
            Sbc.resize(0);
            Rdt.resize(0);
            Rdc.resize(0);
            Rdb.resize(0);
            thetaY.resize(0);
            alpha.resize(0);
            OutriggerElevationVec.resize(0);
            Ytop_mode.resize(0);
            uBRBy.resize(0);
            uBRByOpensees.resize(0);
            thetaYOpensees.resize(0);
            Fy_BRB.resize(0);
            BRBductilityOpensees.resize(0);
            segment_vec.resize(0);
            mode_vec.resize(0);
            L_vec.resize(0);
            Lp_vec.resize(0);
            CPD.resize(0);
            BRBmaxDeformation.resize(0);
            BRBenergy.resize(0);
```

```cpp
        Ed_vec.resize(0);
        Es_vec.resize(0);
        BRBPostYieldStiffnessRatio.resize(0);
        std::cout << "================================ " << tired << " ================================" << std::endl;
        sol_num = seg_num * 2;
        iteration_num = 0;
        segment_vec.resize(seg_num);

        if (tired == 1) {
            for (int i = 1; i != seg_num; i++) {
                if (IfDoSA_NUM == 1) {
                    SANoutput << "alpha_" << i << "\t";
                }
                if (IfDoSA_OpenseesApproah == 1) {
                    SAOPoutput << "alpha_" << i << "\t";
                }
            }
            for (int i = 1; i != seg_num; i++) {
                if (IfDoSA_NUM == 1) {
                    SANoutput << "uBRBy_" << i << "\t";
                }
                if (IfDoSA_OpenseesApproah == 1) {
                    SAOPoutput << "uBRBy_" << i << "\t";
                }
            }
            for (int i = 0; i != sol_num; i++) {
                if (IfDoSA_NUM == 1) {
                    SANoutput << "T" << i + 1 << "\t";
                }
                if (IfDoSA_OpenseesApproah == 1) {
                    SAOPoutput << "T" << i + 1 << "\t";
                }
            }
            for (int i = 0; i != sol_num; i++) {
                if (IfDoSA_NUM == 1) {
                    SANoutput << "MassRatio" << i + 1 << "\t";
                }
                if (IfDoSA_OpenseesApproah == 1) {
                    SAOPoutput << "MassRatio" << i + 1 << "\t";
                }
            }
            for (int i = 0; i != MPAmodesNUM; i++) {
                SAOPoutput1 << "roofDrift_mode" << i + 1 << "\t";
            }
            for (int i = 0; i != MPAmodesNUM; i++) {
                SAOPoutput1 << "roofAcc_mode" << i + 1 << "\t";
            }
            for (int i = 0; i != MPAmodesNUM; i++) {
                SAOPoutput1 << "Es_mode" << i + 1 << "\t";
                for (int j = 1; j != seg_num; j++) {
                    SAOPoutput1 << "Ed" << j << "_mode" << i + 1 << "\t";
                }
            }
            SAOPoutput1 << "SRSSroofDrift" << "\t" << "SRSSmaxInterStroyDrift" << "\t" << "LocOFmaxInterStoryDrift"
```

```cpp
                    << "\t";
                SAOPoutput1 << "SRSSroofAcc" << "\t" << "SRSSbaseShear" << "\t" << "SRSSoverTurningMoment" << "\t" << "SRSSmaxColForce" << std::endl;
                int ModeNumOutput = MPAmodesNUM;
                for (int i = 0; i != ModeNumOutput; i++) {
                    if (IfDoSA_NUM == 1) {
                        SANoutput << "Teqbar_mode" << i + 1 << "     mu_mode" << i + 1 << "    heq_mode" << i + 1 << " maxDriftRatio_mode" << i + 1 << " yieldRoofDrift_mode" << i + 1 << "\t";
                    }
                }
                for (int i = 0; i != MPAmodesNUM; i++) {
                    if (IfDoSA_OpenseesApproah == 1) {
                        SAOPoutput << "Teqbar_mode" << i + 1 << "    mu_mode" << i + 1 << "    heq_mode" << i + 1 << " maxDriftRatio_mode" << i + 1 << " yieldRoofDrift_mode" << i + 1 << "\t";
                    }
                }
                for (int i = 1; i != seg_num; i++) {
                    if (IfDoSA_NUM == 1) {
                        SANoutput << "Rdt_" << i << "\t" << "Rdc_" << i << "\t" << "Rdb_" << i << "\t" << "Sbc_" << i << "\t";
                    }
                    if (IfDoSA_OpenseesApproah == 1) {
                        SAOPoutput << "Rdt_" << i << "\t" << "Rdc_" << i << "\t" << "Rdb_" << i << "\t" << "Sbc_" << i << "\t";
                    }
                }
                if (IfDoSA_NUM == 1) {
                    SANoutput << "MaxRoofDriftSRSS" << std::endl;
                }
                if (IfDoSA_OpenseesApproah == 1) {
                    SAOPoutput << "MaxRoofDriftSRSS" << "\t";
                    for (int i = 0; i != MPAmodesNUM; i++) {
                        SAOPoutput << "IterationNum_Mode" << i + 1 << "\t";
                        for (int j = 1; j != seg_num; j++) {
                            MPAoutput << "YieldStep_" << j << "_mode" << i + 1 << "\t";
                        }
                        for (int j = 1; j != seg_num; j++) {
                            MPAoutput << "YieldDisp_" << j << "_mode" << i + 1 << "\t";
                        }
                        MPAoutput << "IterNum_mode" << i + 1 << "\t";
                    }
                    SAOPoutput << std::endl;
                    MPAoutput << std::endl;
                }

                if (IfDoNLRHA == 1) {
                    for (int i = 1; i != seg_num; i++) {
                        NLRHAoutput << "OutriggerElevation_" << i << "\t";
                    }
                    NLRHAoutput << "h" << "\t";
                    for (int i = 1; i != seg_num; i++) {
```

```
                NLRHAoutput << "alpha_" << i << "\t";
            }
            NLRHAoutput << "DynamicResult" << "\t";
            for (int i = 0; i != sol_num; i++) {
                NLRHAoutput << "T" << i + 1 << "\t";
            }
            for (int i = 1; i != seg_num; i++) {
                NLRHAoutput << "uBRBy" << i << "\t";
            }
            for (int i = 1; i != seg_num; i++) {
                NLRHAoutput << "BRBductility" << i << "\t";
            }
            for (int i = 1; i != seg_num; i++) {
                NLRHAoutput << "CPD" << i << "\t";
            }
            NLRHAoutput << "MaxRoofDrift" << "\t";
            NLRHAoutput << "MaxRoofDisp" << "\t";
            NLRHAoutput << "MaxDisp" << "\t" << "MaxDispLocation" << "\t";
            NLRHAoutput << "RelMaxRoofAcceleration" << "\t" << "AbsMaxRoofAcceleration" << "\t";
            NLRHAoutput << "MaxAcceleration" << "\t" << "MaxAccelerationLocation" << "\t";
            for (int i = 1; i != seg_num; i++) {
                NLRHAoutput << "BRBenergy" << i << "\t";
            }
            NLRHAoutput << "InputEnergy" << "\t";
            for (int i = 1; i != seg_num; i++) {
                NLRHAoutput << "Rdt" << i << "\t" << "Rdc" << i << "\t" << "Rdb" << i << "\t" << "Sbc" << i << "\t";
            }
            NLRHAoutput << "EQname" << "\t" << "EQscalefactor" << "\t";
            for (int i = 1; i != seg_num; i++) {
                NLRHAoutput << "BRB" << i << "/TotalEnergy(%)" << "\t";
            }
            NLRHAoutput << "MaxBaseShear" << "\t" << "MaxOTmoment" << "\t";
            for (int i = 1; i != seg_num; i++) {
                NLRHAoutput << "PerColForce_" << i << "\t";
            }
            NLRHAoutput << std::endl;
        }
    }

    if (IfDisplayDetail == 1) {
        std::cout << "number of segment: ";
        std::cout << seg_num << std::endl;
    }
    input >> YieldDrift;
    double uBRBy_max, uBRBy_min;
    input >> uBRBy_max >> uBRBy_min;
    input >> h;
    if (IfDisplayDetail == 1) {
        std::cout << "Building height: ";
        std::cout << h << "m" << std::endl;
    }
```

```
        OutriggerElevationVec.resize(seg_num - 1);
        alpha.resize(seg_num - 1);
        if (inputType == 1) {
            for (int i = 0; i != seg_num - 1; i++) {
                input >> OutriggerElevationVec[i];
            }
        }
        else if (inputType == 2) {
            std::vector<double> Aalpha;
            Aalpha.resize(seg_num - 1);
            for (int i = 0; i != seg_num - 1; i++) {
                input >> Aalpha[i];
            }
            alpha[1] = Aalpha[1];
            alpha[0] = Aalpha[0] * alpha[1];
            OutriggerElevationVec.resize(0);
            for (int i = 0; i != alpha.size(); i++) {
                OutriggerElevationVec.push_back(alpha[i] * h);
            }
        }
        kt.resize(seg_num - 1);
        kd.resize(seg_num - 1);
        kd_plastic.resize(seg_num - 1);
        kb.resize(seg_num - 1);
        kg.resize(seg_num - 1);
        Sbc.resize(seg_num - 1);
        Rdt.resize(seg_num - 1);
        Rdc.resize(seg_num - 1);
        Rdb.resize(seg_num - 1);
        BRBPostYieldStiffnessRatio.resize(seg_num - 1);

        kc_seg.resize(seg_num - 1);
        Ytop_mode.resize(sol_num);
        thetaY.resize(seg_num - 1);
        uBRBy.resize(seg_num - 1);
        Es_vec.resize(MPAmodesNUM);
        Ed_vec.resize(MPAmodesNUM);
        for (int i = 0; i != Ed_vec.size(); i++) {
            Ed_vec[i].resize(seg_num - 1);
        }

        for (int i = 0; i != seg_num - 1; i++) {
            input >> kt[i];
            input >> kc;
            input >> kd[i];
        }

        for (int i = 0; i != BRBPostYieldStiffnessRatio.size(); i++) {
            input >> value;
            BRBPostYieldStiffnessRatio[i] = value;
        }
        input >> m;
        input >> EIb;
        input >> EIt;
        input >> lt;
```

```
        input >> tolerance;
        input >> tcl_file_name;
        tcl_file_name = tcl_file_name + std::to_string(tired);
        input >> EQname;
        input >> EQscalefactor;

        if (IfDisplayDetail == 1) {
            if (inputType == 1) {
                std::cout << "outrigger location (m): ";
                for (int i = 0; i != seg_num - 1; i++) {
                    std::cout << OutriggerElevationVec[i] << "\t";
                }
            }
            else if (inputType == 2) {
                std::cout << "outrigger location (alpha):";
                for (int i = 0; i != alpha.size(); i++) {
                    std::cout << alpha[i] << "\t";
                }
            }
            std::cout << std::endl << "kt = ";
            for (int i = 0; i != seg_num - 1; i++) {
                std::cout << kt[i] << " kN/m" << "\t";
            }
            std::cout << std::endl << "kc = " << kc << " kN/m" << std::endl << "kd = ";
            for (int i = 0; i != seg_num - 1; i++) {
                std::cout << kd[i] << " kN/m" << "\t";
            }
            std::cout << std::endl << "BRB post-yield stiffness ratio = ";
            for (int i = 0; i != BRBPostYieldStiffnessRatio.size(); i++) {
                std::cout << BRBPostYieldStiffnessRatio[i] << "\t";
            }
            std::cout << std::endl;
            std::cout << "m = " << m << "(ton, mass per unit building height)" << std::endl;
            std::cout << "EIb = " << EIb << std::endl;
            std::cout << "EIt = " << EIt << std::endl;
            std::cout << "lt = " << lt << "m" << std::endl;
            std::cout << "tolerance for spectral analysis = " << tolerance << std::endl;
            std::cout << "tcl file name: " << tcl_file_name << std::endl;
            std::cout << "earthquake: " << EQname << std::endl;
            if (IfEQT1scale != 1) {
                std::cout << "earthquake scale factor: " << EQscalefactor << std::endl;
            }
        }

        if (inputType == 1) {
            for (int i = 0; i != seg_num - 1; i++) {
                alpha[i] = OutriggerElevationVec[i] / h;
            }
        }

        double start_pos = 0;
        double last_seg = 1;
```

```
        double LengthRatio = 0;
        for (int i = 0; i != seg_num - 1; i++) {
            if (i == 0) {
                kc_seg[i] = kc / alpha[i];
                segment_vec[i].AssignLengthRatio(alpha[i]);
                LengthRatio = alpha[i];
            }
            else {
                kc_seg[i] = kc / (alpha[i] - alpha[i - 1]);
                segment_vec[i].AssignLengthRatio(alpha[i] - alpha[i - 1]);
                LengthRatio = alpha[i] - alpha[i - 1];
            }
            segment_vec[i].StartPos = start_pos;
            segment_vec[i].SolutionNumber = sol_num;
            segment_vec[i].au.resize(sol_num);
            start_pos = start_pos + LengthRatio;
            last_seg = last_seg - LengthRatio;
        }
        segment_vec[segment_vec.size() - 1].AssignLengthRatio(last_seg);
        segment_vec[segment_vec.size() - 1].StartPos = start_pos;
        segment_vec[segment_vec.size() - 1].SolutionNumber = sol_num;
        segment_vec[segment_vec.size() - 1].au.resize(sol_num);


        if (IfDisplayDetail == 1) {
            for (int i = 0; i != seg_num - 1; i++) {
                std::cout << "alpha" << i << " = " << alpha[i] << "\t";
            }
            std::cout << std::endl;
        }

        for (int i = 0; i != seg_num - 1; i++) {
            kb[i] = 1 / (alpha[i] / kc + 1 / kt[i]);
            Sbc[i] = lt*lt*h*kb[i] / EIb;
            Rdt[i] = kd[i] / kt[i];
            Rdc[i] = kd[i] / kc;
            Rdb[i] = kd[i] / kb[i];
            kd_plastic[i] = kd[i] * BRBPostYieldStiffnessRatio[i];
        }

        if (IfDoSA_NUM == 1) {
            EndTime = time(NULL);
            runtime << "numerical method start: " << EndTime - StartTime << " 
              sec." << std::endl;
            getL_vec();
            if (IfDisplayDetail == 1) {
                std::cout << "core rotation at outrigger when BRB yields: ";
                for (int i = 0; i != thetaY.size(); i++) {
                    std::cout << thetaY[i] << "\t";
                }
                std::cout << std::endl << "uBRBy: ";
                for (int i = 0; i != uBRBy.size(); i++) {
                    std::cout << uBRBy[i] * 1000 << "mm" << "\t";
                }
                std::cout << std::endl;
            }
```

```
        std::vector<double> NumericalPeriod;
        NumericalPeriod.resize(L_vec.size());
        for (int i = 0; i != L_vec.size(); i++) {
            double lam = L_vec[i];
            double ans = std::pow((lam*lam*lam*lam*EIb / m / h / h / h / h), 0.5);
            ans = 2 * pi / ans;
            NumericalPeriod[i] = ans;
        }

        if (IfDoSA_OpenseesApproah == 0) {
            T1 = NumericalPeriod[0];
        }

        if (IfwriteModeShapeToTxtFile == 1) {
            std::ofstream cout_modeshape;
            cout_modeshape.open((tcl_file_name + "_modeshape.txt").c_str());
            std::cout << "output mode shape to file: " << tcl_file_name << "_modeshape.txt" << std::endl;
            if (!cout_modeshape) {
                std::cerr << "[error][cout_modeshape]" << std::endl;
                system("pause");
                return -1;
            }
            cout_modeshape << "Height" << "\t";
            for (int i = 0; i != mode_vec.size(); i++) {
                cout_modeshape << "mode-" << i + 1 << "phi0" << "\t";
            }
            cout_modeshape << std::endl;
            for (int i = 0; i != mode_vec[0].x_vec.size(); i++) {
                for (int j = 0; j != mode_vec.size(); j++) {
                    if (j == 0) {
                        cout_modeshape << mode_vec[j].x_vec[i] << "\t";
                    }
                    cout_modeshape << mode_vec[j].phi0_vec[i] << "\t";
                }
                cout_modeshape << std::endl;
            }
            std::cout << "Done." << std::endl;
        }

        for (int i = 0; i != alpha.size(); i++) {
            SANoutput << alpha[i] << "\t";
        }
        for (int i = 0; i != uBRBy.size(); i++) {
            SANoutput << uBRBy[i] * 1000 << "\t"; // unit: mm
        }
        for (int i = 0; i != NumericalPeriod.size(); i++) {
            SANoutput << NumericalPeriod[i] << "\t";
        }
        for (int i = 0; i != mode_vec.size(); i++) {
            SANoutput << mode_vec[i].MassParticipationRatio * 100 << "\t";
        }
        std::vector<double> Sd_vec;
```

```cpp
        double maxRoofDriftCombine = 0;
        if (IfDisplayDetail == 1) {
            std::cout << "SPECTRUM ANALYSIS STARTS, ";
        }
        std::vector <double> Mn_vec; // Mn of each mode
        std::vector <double> Kbar_vec; // Kbar of each mode
        std::vector <double> Tbar_vec; // Tbar of each mode

        for (int i = 0; i != mode_vec.size(); i++) {
            Mn_vec.push_back(mode_vec[i].Mn);
            Kbar_vec.push_back(mode_vec[i].Kn);
            Tbar_vec.push_back(mode_vec[i].Tn);
        }
        int ModeNum = MPAmodesNUM;
        for (int i = 0; i != ModeNum; i++) {
            double heq, p, Dh;
            p = (std::pow(Lp_vec[i], 4.0)) / (std::pow(L_vec[i], 4.0));
            double Kbarp = Kbar_vec[i] * p;
            double u0 = DesignSpec.GetSpD(Tbar_vec[i])*mode_vec[i].gamma*Ytop_mode[i];
            if (IfDisplayDetail == 1) {
                std::cout << "mode " << i + 1 << ": Tbar = " << Tbar_vec[i] << " sec" << "\t" << "p = " << p << "\t" << "maxDrift = " << u0 / h << std::endl;
            }
            double prev_u;
            double mu = std::abs(u0 / Ytop_mode[i]);
            double muCout = mu;
            if (mu < 1) {
                mu = 1;
                if (IfDisplayDetail == 1) {
                    std::cout << "mu is less than 1.0 for mode " << i + 1 << std::endl;
                }
            }
            double Keqbar = Kbarp + (Kbar_vec[i] - Kbarp) / mu;
            heq = h0 + 2 / pi / p / mu*log((1 - p + p*mu) / (std::pow(mu, p)));
            Dh = std::pow((1.0 + alpha_in_Dh*h0) / (1.0 + alpha_in_Dh*heq), 0.5);
            double Teqbar = Tbar_vec[i] * std::sqrt(Kbar_vec[i] / Keqbar);
            double new_u = u0*Dh*sqrt(Kbar_vec[i] / Keqbar)*DesignSpec.GetSpV(Teqbar) / DesignSpec.GetSpV(Tbar_vec[i]);
            double error = std::abs((new_u - u0) / u0 * 100.0);
            if (IfDisplayDetail == 1) {
                std::cout << "iteration " << 1 << "\t" << "Teq=" << Tbar_vec[i] * sqrt(Kbar_vec[i] / Keqbar) << "\t" << "ductility=" << muCout << "\t" << "max drift ratio=" << new_u / h << "\t" << "heq=" << heq << "\t" << "error=" << error << std::endl;
            }

            int ind = 2;
            while (error >= tolerance) {
```

```
                    mu = std::abs((new_u / Ytop_mode[i]));
                    muCout = mu;
                    if (mu < 1) {
                        std::cout << "mu is less than 1.0 after iteration 1 procedure" << std::endl;
                        system("pause");
                        return -1;
                    }
                    Keqbar = Kbarp + (Kbar_vec[i] - Kbarp) / mu;
                    heq = h0 + 2 / pi / p / mu*log((1 - p + p*mu) / (std::pow(mu, p)));
                    Dh = std::pow((1.0 + alpha_in_Dh*h0) / (1.0 + alpha_in_Dh*heq), 0.5);
                    Teqbar = Tbar_vec[i] * sqrt(Kbar_vec[i] / Keqbar);
                    prev_u = new_u;
                    new_u = u0*Dh*std::sqrt(Kbar_vec[i] / Keqbar)*DesignSpec.GetSpV(Teqbar) / DesignSpec.GetSpV(Tbar_vec[i]);
                    error = std::abs((new_u - prev_u) / prev_u*100.0);
                    if (IfDisplayDetail == 1) {
                        std::cout << "iteration " << ind << "\t" << "Teq=" << Tbar_vec[i] * std::sqrt(Kbar_vec[i] / Keqbar) << "\t" << "ductility=" << muCout << "\t" << "max drift ratio=" << new_u / h << "\t" << "heq=" << heq << "\t" << "error=" << error << std::endl;
                    }
                    ind++;
                }
                Sd_vec.push_back(new_u);
                SANoutput << Tbar_vec[i] * std::sqrt(Kbar_vec[i] / Keqbar) << "\t" << muCout << "\t" << heq << "\t" << new_u / h << "\t" << std::abs(Ytop_mode[i] / h) << "\t";
                maxRoofDriftCombine += (new_u / h)*(new_u / h);
            }
            for (int i = 0; i != kd.size(); i++) {
                SANoutput << kd[i] / kt[i] << "\t" << kd[i] / kc << "\t" << kd[i] * (1 / kt[i] + alpha[i] / kc) << "\t" << lt*lt*h*kb[i] / EIb << "\t";
            }
            maxRoofDriftCombine = std::sqrt(maxRoofDriftCombine);
            SANoutput << maxRoofDriftCombine << "\t" << std::endl;
        }

        if (IfDoSA_OpenseesApproah == 1) {
            EndTime = time(NULL);
            runtime << "Opensees method start: " << EndTime - StartTime << " sec." << std::endl;
            if (IfDisplayDetail == 1) {
                std::cout << "building Opensees model..." << std::endl;
            }
            MakeOpenseesTCL(tcl_file_name, "modal1", 0);

            std::ifstream inOpenseesPeriod;
            inOpenseesPeriod.open("OpenseesPeriod.txt");
            std::vector<double> OpenseesPeriod;
            OpenseesPeriod.resize(0);
```

```
        double value;
        while (inOpenseesPeriod >> value) {
            OpenseesPeriod.push_back(value);
        }
        inOpenseesPeriod.close();
        if (IfRemoveOpenseesResultFile == 1) {
            remove("OpenseesPeriod.txt");
        }

        T1 = OpenseesPeriod[0];

        std::vector<std::vector<double>> ModeShapeVec;
        for (int i = 1; i <= sol_num; i++) {
            std::ifstream inOpenseesModeShape;
            inOpenseesModeShape.open((tcl_file_name + "ModeShape" + std::to_string(i) + ".txt").c_str());
            if (!inOpenseesModeShape) {
                std::cerr << "[error][inOpenseesModeShape]: open file fail";
                system("pause");
                return -1;
            }
            std::vector<double> vec;
            while (inOpenseesModeShape >> value) {
                vec.push_back(value);
            }
            ModeShapeVec.push_back(vec);
            inOpenseesModeShape.close();
        }

        for (int i = 0; i != ModeShapeVec.size(); i++) {
            for (int j = 0; j != ModeShapeVec[0].size(); j++) {
                if (ModeShapeVec[i][ModeShapeVec[i].size() - 1] < 0) {
                    ModeShapeVec[i][j] = ModeShapeVec[i][j] * -1;
                }
            }
        }

        std::vector<double> MnVec;
        MnVec.resize(sol_num);
        for (int i = 0; i != sol_num; i++) {
            value = 0;
            for (int j = 0; j != ModeShapeVec[i].size(); j++) {
                value = value + m * ModeShapeVec[i][j] * ModeShapeVec[i][j];
            }
            MnVec[i] = value;
        }
        std::vector<double> GammaNVec;
        GammaNVec.resize(sol_num);
        for (int i = 0; i != sol_num; i++) {
            value = 0;
            for (int j = 0; j != ModeShapeVec[i].size(); j++) {
                value += m * ModeShapeVec[i][j];
            }
            GammaNVec[i] = value / MnVec[i];
```

```
        }
        std::vector<double> MassPartRatioVec;
        MassPartRatioVec.resize(sol_num);
        for (int i = 0; i != sol_num; i++) {
            MassPartRatioVec[i] = MnVec[i] * GammaNVec[i] * GammaNVec[i];
        }
        double TotalMassPartRatio = 0;
        for (int i = 0; i != sol_num; i++) {
            TotalMassPartRatio += MassPartRatioVec[i];
        }
        for (int i = 0; i != sol_num; i++) {
            MassPartRatioVec[i] = MassPartRatioVec[i] / TotalMassPartRatio * 100;
        }

        std::vector<std::vector<double>> ModeShapeUdy;
        ModeShapeUdy.resize(3);
        for (int i = 0; i != 3; i++) {
            ModeShapeUdy[i].resize(ModeShapeVec[i].size());
        }
        value = 0;
        for (int i = 0; i != 3; i++) {
            for (int j = 0; j != ModeShapeUdy[i].size(); j++) {
                ModeShapeUdy[i][j] = DesignSpec.GetSpD(OpenseesPeriod[i]) *GammaNVec[i] * ModeShapeVec[i][j];
                if (j == ModeShapeUdy[i].size() - 1) {
                    value += ModeShapeUdy[i][j] * ModeShapeUdy[i][j];
                }
            }
        }

        double factor;
        value = std::sqrt(value);
        factor = h / YieldDrift / value;
        for (int i = 0; i != 3; i++) {
            for (int j = 0; j != ModeShapeUdy[i].size(); j++) {
                ModeShapeUdy[i][j] = ModeShapeUdy[i][j] * factor;
            }
        }

        std::vector<double> Rudy;
        Rudy.resize(5);

        for (int i = 0; i != 4; i++) {
            std::vector<double> ShapeUdy;
            ShapeUdy.resize(ModeShapeUdy[0].size());
            if (i < 3) {
                for (int j = 0; j != ShapeUdy.size(); j++) {
                    ShapeUdy[j] = ModeShapeUdy[i][j];
                }
            }
            if (i == 3) { // case 4
                for (int j = 0; j != ShapeUdy.size(); j++) {
                    value = 0;
                    for (int k = 0; k != 3; k++) {
                        value += ModeShapeUdy[k][j] * ModeShapeUdy[k][j];
```

```
                }
                ShapeUdy[j] = std::sqrt(value);
            }
        }
        thetaYOpensees.resize(0);
        for (int k = 0; k != OutriggerElevationVec.size(); k++) {
            for (int j = 1; j != ShapeUdy.size(); j++) {
                if (OutriggerElevationVec[k] > (double)(j - 1) && OutriggerElevationVec[k] < (double)j) {
                    thetaYOpensees.push_back(ShapeUdy[j] - ShapeUdy[j - 1]);
                }
                else if (OutriggerElevationVec[k] == (double)j && j < ShapeUdy.size() - 1) {
                    thetaYOpensees.push_back(((ShapeUdy[j + 1] - ShapeUdy[j]) + (ShapeUdy[j] - ShapeUdy[j - 1]))*0.5);
                }
                else if (OutriggerElevationVec[k] == (double)j && j == ShapeUdy.size() - 1) {
                    thetaYOpensees.push_back(ShapeUdy[j] - ShapeUdy[j - 1]);
                }
            }
        }

        matrix udyOpenseesMatrix;
        matrix Cd(seg_num - 1, seg_num - 1);
        matrix D(seg_num - 1, seg_num - 1);
        matrix thetaYOpenseesMatrix(seg_num - 1, 1);
        uBRByOpensees.resize(seg_num - 1);
        for (int j = 0; j != thetaYOpensees.size(); j++) {
            thetaYOpenseesMatrix.assign(j, 0, thetaYOpensees[j]);
        }
        for (int k = 0; k != seg_num - 1; k++) {
            for (int j = 0; j != seg_num - 1; j++) {
                if (k == j && k != seg_num - 2) {
                    Cd.assign(k, j, (kc_seg[k] + kc_seg[k + 1]) / kd[k]);
                    D.assign(k, j, (1 + (kc_seg[k] + kc_seg[k + 1])*(1 / kd[k] + 1 / kt[k])) / lt);
                }
                else if (k == j && k == seg_num - 2) {
                    Cd.assign(k, j, kc_seg[k] / kd[k]);
                    D.assign(k, j, (1 + kc_seg[k] * (1 / kd[k] + 1 / kt[k])) / lt);
                }
                else if (k - j == 1) {
                    Cd.assign(k, j, -1 * kc_seg[k] / kd[k]);
                    Cd.assign(j, k, -1 * kc_seg[k] / kd[j]);
                    D.assign(k, j, (-1 * kc_seg[k] * (1 / kd[k] + 1 / kt[k])) / lt);
                }
                else if (j - k == 1) {
                    D.assign(k, j, (-1 * kc_seg[j] * (1 / kd[k] + 1 / kt[k])) / lt);
                }
```

```
                    }
                }
                udyOpenseesMatrix = Cd * D.inverse()*thetaYOpenseesMatrix;

                if (seg_num == 3) {
                    Rudy[i] = udyOpenseesMatrix.Cell(0, 0) / udyOpenseesMatrix.Cell(1, 0);
                }
                else if (seg_num == 2){
                    Rudy[i] = 1;
                }

                Rudy[4] = 1; // case 5
                if (i == 3) {
                    for (int j = 0; j != udyOpenseesMatrix.get_row(); j++) {
                        uBRByOpensees[j] = udyOpenseesMatrix.Cell(j, 0);
                    }
                }
            }

            if (seg_num == 3) {
                if (BRByieldCase == 1 || BRByieldCase == 2 || BRByieldCase == 3 || BRByieldCase == 5) {
                    uBRByOpensees[0] = uBRByOpensees[1] * Rudy[BRByieldCase - 1];
                }
            }

            matrix udyOpenseesMatrix(seg_num - 1, 1);
            matrix Cd(seg_num - 1, seg_num - 1);
            matrix D(seg_num - 1, seg_num - 1);
            matrix thetaYOpenseesMatrix1(seg_num - 1, 1);
            for (int i = 0; i != seg_num - 1; i++) {
                udyOpenseesMatrix.assign(i, 0, uBRByOpensees[i]);
            }
            for (int k = 0; k != seg_num - 1; k++) {
                for (int j = 0; j != seg_num - 1; j++) {
                    if (k == j && k != seg_num - 2) {
                        Cd.assign(k, j, (kc_seg[k] + kc_seg[k + 1]) / kd[k]);
                        D.assign(k, j, (1 + (kc_seg[k] + kc_seg[k + 1])*(1 / kd[k] + 1 / kt[k])) / lt);
                    }
                    else if (k == j && k == seg_num - 2) {
                        Cd.assign(k, j, kc_seg[k] / kd[k]);
                        D.assign(k, j, (1 + kc_seg[k] * (1 / kd[k] + 1 / kt[k])) / lt);
                    }
                    else if (k - j == 1) {
                        Cd.assign(k, j, -1 * kc_seg[k] / kd[k]);
                        Cd.assign(j, k, -1 * kc_seg[k] / kd[j]);
                        D.assign(k, j, (-1 * kc_seg[k] * (1 / kd[k] + 1 / kt[k])) / lt);
                    }
                    else if (j - k == 1) {
                        D.assign(k, j, (-1 * kc_seg[j] * (1 / kd[k] + 1 / kt[k])) / lt);
```

```
                    }
                }
            }
            thetaYOpenseesMatrix1 = D * (Cd.inverse())*udyOpenseesMatrix;
            for (int i = 0; i != thetaYOpenseesMatrix1.get_row(); i++) {
                thetaYOpenseesMatrix1.assign(i, 0, std::abs
                  (thetaYOpenseesMatrix1.Cell(i, 0)));
            }

            Fy_BRB.resize(seg_num - 1);
            for (int i = 0; i != Fy_BRB.size(); i++) {
                Fy_BRB[i] = std::abs(uBRByOpensees[i])*kd[i];
            }

            if (seg_num == 2) {
                IfAssignUy == 1;
            }
            if (IfAssignUy == 2) {
                double RudyAssign;
                input >> RudyAssign;
                uBRByOpensees[0] = RudyAssign;
                input >> RudyAssign;
                uBRByOpensees[1] = RudyAssign;
            }

            if (IfDisplayDetail == 1) {
                std::cout << "BRB axial deformation(s) = ";
                for (int i = 0; i != uBRByOpensees.size(); i++) {
                    std::cout << uBRByOpensees[i] * 1000 << "mm" << "\t";
                }
                std::cout << std::endl;
            }

            if (IfDisplayDetail == 1) {
                for (int i = 0; i != sol_num; i++) {
                    std::cout << "mode " << i + 1 << ", mass participation
                   ratio = " << MassPartRatioVec[i] << "%, participation
                   factor = " << GammaNVec[i] << std::endl;
                }
            }

            for (int i = 0; i != alpha.size(); i++) {
                SAOPoutput << alpha[i] << "\t";
            }
            for (int i = 0; i != uBRByOpensees.size(); i++) {
                SAOPoutput << uBRByOpensees[i] * 1000 << "\t"; //unit: mm
            }
            for (int i = 0; i != OpenseesPeriod.size(); i++) {
                SAOPoutput << OpenseesPeriod[i] << "\t";
            }
            for (int i = 0; i != MassPartRatioVec.size(); i++) {
                SAOPoutput << MassPartRatioVec[i] << "\t";
            }

            if (IfDisplayDetail == 1) {
                std::cout << "SPECTRUM ANALYSIS STARTS WITH MPA (Opensees
```

```
                  METHOD)" << std::endl;
            }
            double heq, Dh;

            std::vector<double> MaxRoofDriftCombined;
            std::vector<double> MaxRoofAccCombined;
            std::vector<std::vector<double>> DeformShape;
            std::vector<std::vector<double>> NodeAcc;
            std::vector<double> StoryShear;
            std::vector<double> StoryMoment;
            DeformShape.resize(MPAmodesNUM);
            NodeAcc.resize(MPAmodesNUM);
            StoryShear.resize(ModeShapeVec[0].size());
            StoryMoment.resize(ModeShapeVec[0].size());

            std::ifstream readPushoverResult;
            std::vector<int> IterationCountForMode;
            IterationCountForMode.resize(sol_num);
            for (int nthmode = 0; nthmode != MPAmodesNUM; nthmode++) {
                remove("OpenseesPushoverResult.txt");
                MPAmodeNow = nthmode;
                iteration_num = 0;

                double u0 = DesignSpec.GetSpD(OpenseesPeriod[nthmode])
                  *GammaNVec[nthmode] * ModeShapeVec[nthmode][ModeShapeVec
                  [nthmode].size() - 1];
                u0 = std::abs(u0);
                if (IfDisplayDetail == 1) {
                    std::cout << "maxDriftRatio_mode" << nthmode + 1 << " = "
                     << u0 / h << std::endl;
                }
                MPAtargetdisp = u0;
                MPAincr = MPAtargetdisp / MPAstep;

                MakeOpenseesTCL(tcl_file_name, "MPA", nthmode + 1);
                readPushoverResult.open("OpenseesPushoverResult.txt");
                if (!readPushoverResult) {
                    std::cerr << "[error][readPushoverResult]" << std::endl;
                    system("pause");
                    return -1;
                }
                std::string POresult;
                readPushoverResult >> POresult;
                readPushoverResult.close();
                if (POresult == "success" && KE != 0) {
                    double integrate_heq_p = 0;
                    for (int k = 1; k != disp.size(); k++) {
                        integrate_heq_p += (heq_p(k) + heq_p(k - 1))*(disp[k]
                     - disp[k - 1])*0.5;
                    }
                    if (integrate_heq_p >= 0) {
                        heq = h0 + integrate_heq_p / disp[disp.size() - 1];
                        double mu = MPAtargetdisp / yieldDisp[0];
                        double Keqbar = force[force.size() - 1] / disp
                    [disp.size() - 1];
                        Dh = pow((1.0 + alpha_in_Dh * h0) / (1.0 +
```

```
alpha_in_Dh * heq), 0.5);
        double Teqbar = OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar);
        double new_u = u0 * Dh*sqrt(KE / Keqbar) *DesignSpec.GetSpV(Teqbar) / DesignSpec.GetSpV(OpenseesPeriod[nthmode]);
        mu = new_u / yieldDisp[0];
        double error = std::abs((new_u - u0) / u0 * 100);
        if (IfDisplayDetail == 1) {
            std::cout << "iteration " << 1 << "\t" << "Teq=" << OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar) << "\t";
            std::cout << "ductility=" << mu << "\t" << "max drift ratio=" << new_u / h << "\t";
            std::cout << "heq=" << heq << "\t" << "error=" << error << std::endl;
        }
        int ind = 2;

        while (error >= tolerance && iteration_num <= iteration_limit) {
            if (std::abs(new_u) > disp[disp.size() - 1]) {
                MPAtargetdisp = std::abs(new_u);
                MPAincr = MPAtargetdisp / MPAstep;
                remove("OpenseesPushoverResult.txt");
                MakeOpenseesTCL(tcl_file_name, "MPA", nthmode + 1);
                readPushoverResult.open("OpenseesPushoverResult.txt");
                if (!readPushoverResult) {
                    std::cerr << "[error][readPushoverResult]" << std::endl;
                    system("pause");
                    return -1;
                }
                std::string POresult;
                readPushoverResult >> POresult;
                readPushoverResult.close();
                integrate_heq_p = 0;
                for (int k = 1; k != disp.size(); k++) {
                    integrate_heq_p += (heq_p(k) + heq_p(k - 1))*(disp[k] - disp[k - 1])*0.5;
                }
                if (POresult == "fail" || KE == 0 || integrate_heq_p < 0) {
                    if (IfDisplayDetail == 1) {
                        std::cout << "pushover for mode " << nthmode + 1 << " fails." << std::endl;
                    }
                    iteration_num = iteration_limit + 1;
                }

            }
            else {
                std::vector<double> new_disp;
                std::vector<double> new_force;
```

```
                for (int i = 0; i != disp.size(); i++) {
                    if (disp[i] < std::abs(new_u)) {
                        new_disp.push_back(disp[i]);
                        new_force.push_back(force[i]);
                    }
                }
                if (new_disp.size() <=2) {
                    iteration_num = iteration_limit + 1;
                }
                else {
                    new_disp.push_back(std::abs(new_u));

                    new_force.push_back((new_force[new_force.size() - 1] - new_force[new_force.size() - 2])*(new_disp[new_disp.size() - 1] - new_disp[new_disp.size() - 3]) / (new_disp[new_disp.size() - 2] - new_disp[new_disp.size() - 3]) + new_force[new_force.size() - 2]);

                    disp.resize(0);
                    force.resize(0);
                    for (int i = 0; i != new_disp.size(); i++) {
                        disp.push_back(new_disp[i]);
                        force.push_back(new_force[i]);
                    }
                    integrate_heq_p = 0;

                    for (int k = 1; k != disp.size(); k++) {
                        integrate_heq_p += (heq_p(k) + heq_p(k - 1))*(disp[k] - disp[k - 1])*0.5;
                    }
                    if (integrate_heq_p < 0) {
                        iteration_num = iteration_limit + 1;
                    }
                }
            }

            if (iteration_num <= iteration_limit && integrate_heq_p >= 0) {
                heq = h0 + integrate_heq_p / disp[disp.size() - 1];
                Dh = std::pow((1.0 + alpha_in_Dh * h0) / (1.0 + alpha_in_Dh * heq), 0.5); // new Dh
                Teqbar = OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar);
                double prev_u = new_u;
                new_u = u0 * Dh*sqrt(KE / Keqbar)*DesignSpec.GetSpV(Teqbar) / DesignSpec.GetSpV(OpenseesPeriod[nthmode]);
                error = std::abs((new_u - prev_u) / prev_u * 100.0);
                mu = new_u / yieldDisp[0];
                if (IfDisplayDetail == 1) {
                    std::cout << "iteration " << ind << "\t" << "Teq=" << OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar) << "\t";
```

```cpp
                        std::cout << "ductility=" << mu << "\t" << "max drift ratio=" << new_u / h << "\t";
                        std::cout << "heq=" << heq << "\t" << "error=" << error << std::endl;
                    }
                    ind++;
                    iteration_num++;
                }
            }
            if (iteration_num >= iteration_limit) {
                ind = iteration_num;
                std::cout << "number of iteraion (" << iteration_num << ") exceeds " << iteration_limit << ", force iteration." << std::endl;
                error = 10000;
                tolerance = tolerance / 10;
                u0 = ModeShapeVec[0][ModeShapeVec[0].size() - 1];
                double incr = ModeShapeVec[0][ModeShapeVec[0].size() - 1] / 100.0;
                std::vector<double> orig_disp;
                std::vector<double> orig_force;
                for (int i = 0; i != disp.size(); i++) {
                    orig_disp.push_back(disp[i]);
                    orig_force.push_back(force[i]);
                }

                while (error >= tolerance) {
                    u0 = u0 + incr;
                    std::cout << u0 << "\t" << incr << std::endl;
                    MPAtargetdisp = u0;
                    MPAincr = MPAtargetdisp / 100.0;
                    remove("OpenseesPushoverResult.txt");
                    MakeOpenseesTCL(tcl_file_name, "MPA", nthmode + 1);
                    readPushoverResult.open("OpenseesPushoverResult.txt");
                    if (!readPushoverResult) {
                        std::cerr << "[error][readPushoverResult]" << std::endl;
                        system("pause");
                        return -1;
                    }
                    readPushoverResult >> POresult;
                    readPushoverResult.close();

                    KE = 0;

                    if (POresult == "fail" || KE == 0) {
                        if (IfDisplayDetail == 1) {
                            std::cerr << "pushover for mode " << nthmode + 1 << " fails, analysis for this mode terminated." << std::endl;
                        }
                        if (KE == 0) {
                            std::cerr << "KE = 0" << std::endl;
                        }
```

```
                error = 0;
            }
            else {
                integrate_heq_p = 0;
                for (int k = 1; k != disp.size(); k++) {
                    integrate_heq_p += (heq_p(k) + heq_p(k - 1))*(disp[k] - disp[k - 1])*0.5;
                }
                heq = h0 + integrate_heq_p / disp[disp.size() - 1];
                Keqbar = force[force.size() - 1] / disp[disp.size() - 1];
                Dh = pow((1.0 + alpha_in_Dh * h0) / (1.0 + alpha_in_Dh * heq), 0.5);
                Teqbar = OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar);
                new_u = u0 * Dh*sqrt(KE / Keqbar)*DesignSpec.GetSpV(Teqbar) / DesignSpec.GetSpV(OpenseesPeriod[nthmode]);
                error = std::abs((new_u - u0) / u0 * 100.0);
                if (IfDisplayDetail == 1) {
                    std::cout << "iteration " << ind << "\t" << "Teq=" << OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar) << "\t";
                    std::cout << "ductility=" << mu << "\t" << "max drift ratio=" << new_u / h << "\t";
                    std::cout << "heq=" << heq << "\t" << "error=" << error << std::endl;
                }
                ind++;
            }
        }
    }

    if ((OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar)) > 0) {
        for (int i = 0; i != ModeShapeVec[nthmode].size(); i++) {
            double Teq;
            Teq = OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar);
            NodeAcc[nthmode].push_back(DesignSpec.GetSpA(OpenseesPeriod[nthmode])*GammaNVec[nthmode]*ModeShapeVec[nthmode][i]*Dh*OpenseesPeriod[nthmode]/Teq*DesignSpec.GetSpV(Teq)/DesignSpec.GetSpV(OpenseesPeriod[nthmode]));
        }
    }
    else {
        MaxRoofAccCombined.push_back(0);
        for (int i = 0; i != ModeShapeVec[nthmode].size(); i++) {
            NodeAcc[nthmode].push_back(0);
        }
    }
```

```cpp

            if (SRSSmodeShapeType == 1) {
                for (int i = 0; i != ModeShapeVec[nthmode].size(); i++) {
                    DeformShape[nthmode].push_back(ModeShapeVec[nthmode][i] / ModeShapeVec[nthmode][ModeShapeVec[nthmode].size() - 1] * new_u);
                }
                MPAmodeNow = nthmode;
                MPAtargetdisp = new_u;
                MPAincr = MPAtargetdisp / MPAstep;
                if (IfDisplayDetail == 1) {
                    std::cout << "Running pushover for inelastic mode shape of mode" << nthmode + 1;
                    std::cout << " target disp = " << MPAtargetdisp << std::endl;
                }
                MakeOpenseesTCL(tcl_file_name, "MPA", nthmode + 1);
                Es_vec[nthmode] = Es;
                for (int i = 0; i != seg_num - 1; i++) {
                    Ed_vec[nthmode][i] = Ed[i];
                }
                if (IfRemoveOpenseesResultFile == 1)
                {
                    remove((tcl_file_name + "MPA_LateralDisplacement_Mode_" + std::to_string(nthmode + 1) + ".txt").c_str());
                }
            }

            else if (SRSSmodeShapeType == 2) {
                MPAmodeNow = nthmode;
                MPAtargetdisp = new_u;
                MPAincr = MPAtargetdisp / MPAstep;
                if (IfDisplayDetail == 1) {
                    std::cout << "Running pushover for inelastic mode shape of mode" << nthmode + 1;
                    std::cout << " target disp = " << MPAtargetdisp << std::endl;
                }
                MakeOpenseesTCL(tcl_file_name, "MPA", nthmode + 1);
                Es_vec[nthmode] = Es;
                for (int i = 0; i != seg_num - 1; i++) {
                    Ed_vec[nthmode][i] = Ed[i];
                }

                DeformShape[nthmode].resize(CoreMassID.size());
                std::ifstream inMPAdrift;
                inMPAdrift.open((tcl_file_name + "MPA_LateralDisplacement_Mode_" + std::to_string(nthmode + 1) + ".txt").c_str());
                if (!inMPAdrift) {
                    std::cerr << "[error][inMPAdrift]" << std::endl;
```

```cpp
        }
        int indd = 0;
        while (inMPAdrift >> value) {
            DeformShape[nthmode][indd] = value;
            indd = indd + 1;
            if (indd == CoreMassID.size()) {
                indd = 0;
            }
        }
        inMPAdrift.close();
        if (IfRemoveOpenseesResultFile == 1)
        {
            remove((tcl_file_name + "MPA_LateralDisplacement_Mode_" + std::to_string(nthmode + 1) + ".txt").c_str());
        }
    }

    if (KE == 0) {
        SAOPoutput << 0 << "\t" << 1 << "\t" << h0 << "\t" << 0 << "\t" << 0 << "\t";
        if (nthmode == 0) {
            T1EFF = 0;
        }
    }
    else {
        SAOPoutput << OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar) << "\t" << mu << "\t" << heq << "\t" << new_u / h << "\t" << std::abs(yieldDisp[0] / h) << "\t";
        MaxRoofDriftCombined.push_back(new_u / h);
        if (nthmode == 0) {
            T1EFF = OpenseesPeriod[nthmode] * std::sqrt(KE / Keqbar);
        }
    }
    for (int i = 0; i != kd.size(); i++) {
        MPAoutput << yieldStep[i] << "\t";
    }
    for (int i = 0; i != kd.size(); i++) {
        MPAoutput << yieldDisp[i] << "\t";
    }
    MPAoutput << ind << "\t";
    if (IfDisplayDetail == 1) {
        std::cout << "*********************************************" << std::endl;
        std::cout << "mode " << nthmode + 1 << " Done." << std::endl;
        std::cout << "*********************************************" << std::endl;
    }
    IterationCountForMode[nthmode] = ind;
}
else {
    if (IfDisplayDetail == 1) {
```

```
                            std::cout << "initial pushouver for mode " << nthmode + 1 << " strange result." << std::endl;
                        }
                        SAOPoutput << 0 << "\t" << 1 << "\t" << h0 << "\t" << 0 << "\t" << 0 << "\t";
                        for (int i = 0; i != kd.size(); i++) {
                            MPAoutput << "fail" << "\t";
                        }
                        for (int i = 0; i != kd.size(); i++) {
                            MPAoutput << "fail" << "\t";
                        }
                        MPAoutput << "fail" << "\t";
                        IterationCountForMode[nthmode] = 0;

                    }
                }
                else {
                    if (IfDisplayDetail == 1) {
                        std::cout << "initial pushouver for mode " << nthmode + 1 << " fails." << std::endl;
                        if (KE == 0) {
                            std::cout << "KE = 0" << std::endl;
                        }
                    }
                    SAOPoutput << 0 << "\t" << 1 << "\t" << h0 << "\t" << 0 << "\t" << 0 << "\t";
                    for (int i = 0; i != kd.size(); i++) {
                        MPAoutput << "fail" << "\t";
                    }
                    for (int i = 0; i != kd.size(); i++) {
                        MPAoutput << "fail" << "\t";
                    }
                    MPAoutput << "fail" << "\t";
                    IterationCountForMode[nthmode] = 0;
                }
                if (IfRemoveOpenseesResultFile == 1) {
                    remove((tcl_file_name + "MPAmono_RoofDisp_Mode_" + std::to_string(nthmode + 1) + ".txt").c_str());
                    remove((tcl_file_name + "MPAmono_BaseShear_Mode_" + std::to_string(nthmode + 1) + ".txt").c_str());
                    for (int i = 0; i != kd.size(); i++) {
                        remove((tcl_file_name + "MPA_BRBdeform" + std::to_string(i + 1) + "_Mode_" + std::to_string(nthmode + 1) + ".txt").c_str());
                        remove((tcl_file_name + "MPA_BRBaxialForce" + std::to_string(i + 1) + "_Mode_" + std::to_string(nthmode + 1) + ".txt").c_str());
                    }
                }
            }

            std::vector<double> SRSSdeformshape;

            for (int i = 0; i != DeformShape.size(); i++) {
                if (DeformShape[i].size() == 0) {
                    for (int j = 0; j != ModeShapeVec[0].size(); j++) {
```

```
            DeformShape[i].push_back(0);
        }
    }
    if (NodeAcc[i].size() == 0) {
        for (int j = 0; j != ModeShapeVec[0].size(); j++) {
            NodeAcc[i].push_back(0);
        }
    }
}

for (int i = 0; i != DeformShape[0].size(); i++) {
    double value = 0;
    for (int j = 0; j != DeformShape.size(); j++) {
        value += DeformShape[j][i] * DeformShape[j][i];
    }
    SRSSdeformshape.push_back(std::sqrt(value));
    DriftSA << std::sqrt(value) << "\t";
}
DriftSA << std::endl;

for (int i = 0; i != DeformShape.size(); i++) {
    SAOPoutput1 << DeformShape[i][DeformShape[i].size() - 1] / h << "\t";
    gg << "roofdisp" << "\t";
}

StoryShear.resize(SRSSdeformshape.size());
StoryMoment.resize(SRSSdeformshape.size());
for (int i = 0; i != StoryShear.size(); i++) {
    StoryShear[i] = EI(i*MassSpacing)*Derivate3(SRSSdeformshape)[i];
    StoryMoment[i] = EI(i*MassSpacing)*Derivate2(SRSSdeformshape)[i];
}

std::vector<double> SRSSacc;

for (int i = 0; i != NodeAcc[0].size(); i++) {
    double value = 0;
    for (int j = 0; j != NodeAcc.size(); j++) {
        value += NodeAcc[j][i] * NodeAcc[j][i];
    }
    SRSSacc.push_back(std::sqrt(value));
}

for (int i = 0; i != NodeAcc.size(); i++) {
    SAOPoutput1 << NodeAcc[i][NodeAcc[i].size() - 1] << "\t";
    gg << "roofacc" << "\t";
}

for (int i = 0; i != Es_vec.size(); i++) {
    SAOPoutput1 << Es_vec[i] << "\t";
    for (int j = 0; j != Ed_vec[i].size(); j++) {
        SAOPoutput1 << Ed_vec[i][j] << "\t";
    }
}
```

```
            SAOPoutput1 << SRSSdeformshape[SRSSdeformshape.size() - 1] / h << "\t";
            gg << "roofdrift" << "\t";

            std::vector<double> SRSSinterStoryDrift;
            double SRSSmaxInterStoryDrift = 0;
            double LocSRSSmaxInterStoryDrift = 0;
            SRSSinterStoryDrift.push_back(0);
            for (int i = 1; i != SRSSdeformshape.size(); i++) {
                SRSSinterStoryDrift.push_back(std::abs((SRSSdeformshape[i] - SRSSdeformshape[i - 1]) / MassSpacing));
            }
            for (int i = 0; i != SRSSinterStoryDrift.size(); i++) {
                InterStorySA << SRSSinterStoryDrift[i] << "\t";
            }
            InterStorySA << std::endl;

            for (int i = 0; i != SRSSinterStoryDrift.size(); i++) {
                if (SRSSinterStoryDrift[i] >= SRSSmaxInterStoryDrift) {
                    SRSSmaxInterStoryDrift = SRSSinterStoryDrift[i];
                    LocSRSSmaxInterStoryDrift = h / ((double) SRSSinterStoryDrift.size() - 1.0)*(double)i;
                }
            }
            SAOPoutput1 << SRSSmaxInterStoryDrift << "\t" << LocSRSSmaxInterStoryDrift << "\t";
            gg << "interstorydrift" << "\t" << "location" << "\t";
            SAOPoutput1 << SRSSacc[SRSSacc.size() - 1] << "\t";
            gg << "SRSSroofacceleration" << "\t";

            std::vector<double> SRSSmoment;
            std::vector<double> tempInterstorydrift;
            SRSSmoment.resize(SRSSdeformshape.size());
            tempInterstorydrift.resize(SRSSdeformshape.size());
            tempInterstorydrift[0] = 0;
            for (int i = 1; i != tempInterstorydrift.size() - 1; i++) {
                tempInterstorydrift[i] = ((SRSSdeformshape[i] - SRSSdeformshape[i - 1])*0.5 + (SRSSdeformshape[i + 1] - SRSSdeformshape[i])*0.5) / MassSpacing;
            }
            tempInterstorydrift[tempInterstorydrift.size() - 1] = (SRSSdeformshape[SRSSdeformshape.size() - 1] - SRSSdeformshape[SRSSdeformshape.size() - 2]) / MassSpacing;

            SRSSmoment[0] = (tempInterstorydrift[1] - tempInterstorydrift[0]) / MassSpacing;
            for (int i = 1; i != tempInterstorydrift.size() - 1; i++) {
                SRSSmoment[i] = ((tempInterstorydrift[i] - tempInterstorydrift[i - 1])*0.5 + (tempInterstorydrift[i + 1] - tempInterstorydrift[i])*0.5) / MassSpacing;
            }

            for (int i = 0; i != SRSSmoment.size(); i++) {
                MomentSA << SRSSmoment[i]*EI((double)i*MassSpacing) << "\t";
            }
```

```
            MomentSA << std::endl;

            matrix SRSthetaAtOutrigger(seg_num - 1, 1);
            for (int i = 0; i != OutriggerElevationVec.size(); i++) {
                for (int j = 1; j != SRSinterStoryDrift.size(); j++) {
                    if ((double)j*MassSpacing >= OutriggerElevationVec[i] && 
                    ((double)j-1)*MassSpacing <= OutriggerElevationVec[i]) {
                        double a, b, e, x, y;
                        a = ((double)j - 1)*MassSpacing;
                        b = (double)j*MassSpacing;
                        e = OutriggerElevationVec[i];
                        x = SRSinterStoryDrift[j - 1];
                        y = SRSinterStoryDrift[j];
                        SRSthetaAtOutrigger.assign(i, 0, (e - a)*(y - x) / 
                    (b - a) + x);
                    }
                }
            }

            matrix FMatrix;
            double SRSmaxColForce;
            if (seg_num == 3) {
                matrix Cm;
                matrix D_elas;
                matrix D_inelas;
                Cm = CreateCmatrix();
                D_elas = CreateDmatrix(kd);
                std::vector<double> kdSRSS;
                if (SRSthetaAtOutrigger.Cell(0, 0) <= 
                  thetaYOpenseesMatrix1.Cell(0, 0) && 
                  SRSthetaAtOutrigger.Cell(1,0)<=thetaYOpenseesMatrix1.Cell 
                  (1,0)) {
                    FMatrix = Cm*D_elas.inverse()*SRSthetaAtOutrigger;
                }
                else if (SRSthetaAtOutrigger.Cell(0, 0) <= 
                  thetaYOpenseesMatrix1.Cell(0, 0) && 
                  SRSthetaAtOutrigger.Cell(1, 0) > 
                  thetaYOpenseesMatrix1.Cell(1, 0)) {
                    kdSRSS.resize(kd.size());
                    kdSRSS[0] = kd[0];
                    kdSRSS[1] = kd[1]*BRBPostYieldStiffnessRatio[1];
                    D_inelas = CreateDmatrix(kdSRSS);
                    matrix S1;
                    S1 = SRSthetaAtOutrigger;
                    S1.assign(1, 0, thetaYOpenseesMatrix1.Cell(1, 0));
                    FMatrix = Cm*D_elas.inverse()*S1 + Cm*D_inelas.inverse()* 
                    (SRSthetaAtOutrigger - S1);

                    std::cout << "case1" << std::endl;
                    std::cout << SRSthetaAtOutrigger << std::endl;
                    std::cout << S1 << std::endl;
                }
                else if (SRSthetaAtOutrigger.Cell(0, 0) > 
                  thetaYOpenseesMatrix1.Cell(0, 0) && 
                  SRSthetaAtOutrigger.Cell(1, 0) <= 
                  thetaYOpenseesMatrix1.Cell(1, 0)) {
```

```cpp
            kdSRSS.resize(kd.size());
            kdSRSS[0] = kd[0] * BRBPostYieldStiffnessRatio[0];
            kdSRSS[1] = kd[1];
            D_inelas = CreateDmatrix(kdSRSS);
            matrix S1;
            S1 = SRSSthetaAtOutrigger;
            S1.assign(0, 0, thetaYOpenseesMatrix1.Cell(0, 0));
            FMatrix = Cm*D_elas.inverse()*S1 + Cm*D_inelas.inverse()*(SRSSthetaAtOutrigger - S1);
            std::cout << "case2" << std::endl;
            std::cout << SRSSthetaAtOutrigger << std::endl;
            std::cout << S1 << std::endl;
        }
        else if (SRSSthetaAtOutrigger.Cell(0, 0) > thetaYOpenseesMatrix1.Cell(0, 0) && SRSSthetaAtOutrigger.Cell(1, 0) > thetaYOpenseesMatrix1.Cell(1, 0)) {
            kdSRSS.resize(kd.size());
            kdSRSS[0] = kd[0] * BRBPostYieldStiffnessRatio[0];
            kdSRSS[1] = kd[1] * BRBPostYieldStiffnessRatio[1];
            D_inelas = CreateDmatrix(kdSRSS);
            matrix S1;
            S1 = SRSSthetaAtOutrigger;
            S1.assign(0, 0, thetaYOpenseesMatrix1.Cell(0, 0));
            FMatrix = Cm*D_elas.inverse()*S1 + Cm*D_inelas.inverse()*(SRSSthetaAtOutrigger - S1);
            std::cout << "case3" << std::endl;
            std::cout << SRSSthetaAtOutrigger << std::endl;
            std::cout << S1 << std::endl;
        }

        SRSSmaxColForce = 0;
        for (int i = 0; i != FMatrix.get_row(); i++) {
            SRSSmaxColForce += FMatrix.Cell(i, 0);
        }
        SRSSmaxColForce = std::abs(SRSSmaxColForce);
    }

    double MaxBaseShear = 0;
    MaxBaseShear = std::sqrt(StoryShear[0]* StoryShear[0]);

    double MaxBaseMoment = 0;
    MaxBaseMoment = std::sqrt(StoryMoment[0] * StoryMoment[0]);

    SAOPoutput1 << MaxBaseShear << "\t" << MaxBaseMoment << "\t" << SRSSmaxColForce << std::endl;
    gg << "baseshear" << "\t" << "moment" << "\t" << "colforce" << "\t";

    double MaxRoofDriftSRSS = 0;
    for (int i = 0; i != MaxRoofDriftCombined.size(); i++) {
        MaxRoofDriftSRSS += MaxRoofDriftCombined[i] * MaxRoofDriftCombined[i];
    }
    MaxRoofDriftSRSS = std::sqrt(MaxRoofDriftSRSS);

```

```
            for (int i = 0; i != kd.size(); i++) {
                SAOPoutput << kd[i] / kt[i] << "\t" << kd[i] / kc << "\t" << kd[i] * (1 / kt[i] + alpha[i] / kc) << "\t" << lt*lt*h*kb[i] / EIb << "\t";
            }
            SAOPoutput << MaxRoofDriftSRSS <<"\t";
            for (int i = 0; i != IterationCountForMode.size(); i++) {
                SAOPoutput << IterationCountForMode[i] << "\t";
            }
            SAOPoutput << std::endl;
            MPAoutput << std::endl;
        }

        if (IfRemoveOpenseesResultFile == 1 && IfDoNLRHA == 0) {
            remove((tcl_file_name + ".tcl").c_str());
        }
        for (int i = 1; i <= sol_num; i++) {
            if (IfRemoveOpenseesResultFile == 1) {
                remove((tcl_file_name + "ModeShape" + std::to_string(i) + ".txt").c_str());
            }
        }

        if (IfRemoveOpenseesResultFile == 3) {
            std::ofstream op3;
            for (int i = 0; i != MPAmodesNUM; i++) {
                op3.open(("mode"+std::to_string(i+1)+"YieldPoint.txt").c_str());
                MPAmodeNow = i;
                MPAtargetdisp = h/YieldDrift * 10;
                MPAincr = MPAtargetdisp / MPAstep;
                std::cout << "RUNNING PUSHOVER FOR MODE " << i + 1;
                std::cout << "  TARGET DISP = " << MPAtargetdisp << std::endl;
                MakeOpenseesTCL(tcl_file_name, "MPA", i + 1);
                for (int j = 0; j != yieldStep.size(); j++) {
                    op3 << yieldStep[j] << "\t";
                }
                for (int j = 0; j != yieldDisp.size(); j++) {
                    op3 << yieldDisp[j] << "\t";
                }
                op3.close();
            }
        }

        if (IfDoSA_OpenseesApproah == 0 && IfAssignUy == 2) {
            uBRByOpensees.resize(seg_num - 1);
            if (seg_num == 2) {
                IfAssignUy == 1;
            }
            if (IfAssignUy == 2) {
                double RudyAssign;
                input >> RudyAssign;
                uBRByOpensees[0] = RudyAssign; // unit:m
                input >> RudyAssign;
                uBRByOpensees[1] = RudyAssign; // unit:m
```

```
            }
        }

        if (IfDoNLRHA == 1) {
            EndTime = time(NULL);
            runtime << "NLRHA start: " << EndTime - StartTime << " sec." << std::endl;
            if (IfEQT1scale == 1) {
                std::string SP_EQname = "SP_" + EQname + ".txt";
                EQspec.ReadSpaFromFile(("C:\\Dropbox\\TokyoTech\\ProgramRun\\OutriggerEXEsetup\\" + SP_EQname).c_str());
                double Sa_Design = DesignSpec.GetSpA(T1);
                double Sa_EQ = EQspec.GetSpA(T1);
                EQscalefactor = Sa_Design / Sa_EQ;
                std::cout << "EQ scale factor (T1 scale) = " << EQscalefactor << std::endl;
            }

            if (IfEQT1scale == 3) {
                std::string SP_EQname = "SP_" + EQname + ".txt";
                EQspec.ReadSpaFromFile(("C:\\Dropbox\\TokyoTech\\ProgramRun\\OutriggerEXEsetup\\" + SP_EQname).c_str());
                double Sa_Design = DesignSpec.GetSpA(T1EFF);
                double Sa_EQ = EQspec.GetSpA(T1EFF);
                EQscalefactor = Sa_Design / Sa_EQ;
                std::cout << "EQ scale factor (T1 scale) = " << EQscalefactor << std::endl;
            }

            if (IfEQT1scale == 2) {
                remove("info.txt");
                std::ofstream info;
                info.open("info.txt");
                if (!info) {
                    std::cerr << "[error][info][outrigger6.exe]";
                    return -1;
                }
                info << EQname << "\t" << T1;
                info.close();

                std::string filepath;
                filepath = "C:\\Dropbox\\TokyoTech\\Program\\GMscaling_outriggerEXEuse\\Debug\\GMscaling_outriggerEXEuse.exe";
                system(filepath.c_str());

                std::ifstream inEQscalefactor;
                inEQscalefactor.open("EQscalefactor.txt");
                if (!inEQscalefactor) {
                    std::cerr << "[error][inEQscalefactor]" << std::endl;
                    return -1;
                }
                inEQscalefactor >> EQscalefactor;
                inEQscalefactor.close();
            }

```

```
            BRBmaxDeformation.resize(kd.size());
            BRBenergy.resize(kd.size());
            CPD.resize(kd.size());
            BRBductilityOpensees.resize(kd.size());
            std::string filepath;
            if (IfDisplayDetail == 1) {
                std::cout << "NLRHA starts" << std::endl;
                std::cout << "reading ground motion..." << std::endl;
            }
            std::ifstream InputEQ;
            InputEQ.open(("C:\\Dropbox\\TokyoTech\\ProgramRun\\OutriggerEXEsetup\\EQ_" + EQname + ".txt").c_str());
            if (!InputEQ) {
                std::cerr << "[error][InputEQ]: EQ file open fail, ";
                std::cerr << "file name: " << EQname << std::endl;
                system("pause");
                return -1;
            }
            EQtime.resize(0);
            EQacc.resize(0);
            while (InputEQ >> value) {
                EQtime.push_back(value);
                InputEQ >> value;
                EQacc.push_back(value);
            }
            std::ofstream OpenseesEQ;
            OpenseesEQ.open("OpenseesEQ.txt");
            for (int i = 0; i != EQacc.size(); i++) {
                OpenseesEQ << EQacc[i] * EQscalefactor << std::endl;
            }
            if (IfDisplayDetail == 1) {
                std::cout << "finish reading ground acceleration, data number = " << EQtime.size() << "\t";
                std::cout << "eqrthquake time = " << EQtime[EQtime.size() - 1] << std::endl;
            }
            dt = EQtime[1] - EQtime[0];
            EQt = EQtime[EQtime.size() - 1];
            std::cout << uBRByOpensees.size();

            MakeOpenseesTCL(tcl_file_name, "NLRHA", 0);
            std::ifstream inOpenseesPeriod;
            std::vector<double> OpenseesPeriod;
            inOpenseesPeriod.open("OpenseesPeriod.txt");
            if (!inOpenseesPeriod) {
                std::cerr << "[error][inOpenseesPeriod]";
                system("pause");
                return -1;
            }
            OpenseesPeriod.resize(0);
            double value;
            while (inOpenseesPeriod >> value) {
                OpenseesPeriod.push_back(value);
            }
            if (IfRemoveOpenseesResultFile == 1) {
                remove("OpenseesPeriod.txt");
```

```
            }
            std::cout << "processing data: " << tcl_file_name << std::endl;
            EndTime = time(NULL);
            runtime << "NLRHA reading LateralVelocity.txt: " << EndTime - StartTime << " sec." << std::endl;

            double InputEnergy = 0;
            value = 0;
            std::ifstream inLateralVelo;
            filepath = tcl_file_name + "LateralVelocity.txt";
            inLateralVelo.open(filepath.c_str());
            for (int t = 0; t != EQtime.size(); t++) {
                for (int i = 0; i <= h; i++) {
                    inLateralVelo >> value;
                    InputEnergy = InputEnergy + (-1.0)*m*EQacc[t] * value*dt;
                }
            }
            inLateralVelo.close();
            if (IfRemoveOpenseesResultFile == 1) {
                std::remove(filepath.c_str());
            }
            std::ifstream inBRBdisp;
            std::ifstream inBRBfor;

            EndTime = time(NULL);
            runtime << "NLRHA reading BRBdeform.txt and BRBforce.txt: " << EndTime - StartTime << " sec." << std::endl;

            for (int i = 0; i != kd.size(); i++) {
                std::string number = std::to_string(i + 1);
                filepath = tcl_file_name + "_BRBdeform" + number + ".txt";
                inBRBdisp.open(filepath.c_str());
                filepath = tcl_file_name + "_BRBforce" + number + ".txt";
                inBRBfor.open(filepath.c_str());
                double BRBd, BRBd_prev, BRBf, BRBf_prev, uBRByPos, uBRByNeg;
                double uBRBrange, CPDi, BRBdimax, BRBenergyi;
                uBRByPos = std::abs(uBRByOpensees[i]);
                uBRByNeg = -1.0*std::abs(uBRByOpensees[i]);
                uBRBrange = 2.0*std::abs(uBRByOpensees[i]);
                BRBd_prev = 0;
                BRBf_prev = 0;
                CPDi = 0;
                BRBdimax = 0;
                BRBenergyi = 0;
                while (inBRBdisp >> BRBd) {
                    inBRBdisp >> value;
                    inBRBfor >> BRBf;
                    inBRBfor >> value;
                    BRBenergyi = BRBenergyi + ((BRBf + BRBf_prev)*(BRBd - BRBd_prev)*0.5);
                    if (BRBd >= uBRByPos) {
                        uBRByPos = BRBd;
                        uBRByNeg = uBRByPos - uBRBrange;
                        CPDi = CPDi + std::abs(BRBd - BRBd_prev);
                    }
                    if (BRBd <= uBRByNeg) {
```

```
            uBRByNeg = BRBd;
            uBRByPos = uBRByNeg + uBRBrange;
            CPDi + CPDi + std::abs(BRBd - BRBd_prev);
        }
        BRBf_prev = BRBf;
        BRBd_prev = BRBd;
        if (std::abs(BRBd) >= BRBdimax) {
            BRBdimax = std::abs(BRBd);
        }
    }
    CPDi = CPDi / std::abs(uBRByOpensees[i]);
    BRBmaxDeformation[i] = BRBdimax;
    BRBductilityOpensees[i] = BRBdimax / std::abs(uBRByOpensees[i]);
    CPD[i] = CPDi;
    BRBenergy[i] = BRBenergyi;
    inBRBdisp.close();
    inBRBfor.close();
    if (IfRemoveOpenseesResultFile == 1) {
        filepath = tcl_file_name + "_BRBdeform" + number + ".txt";
        std::remove(filepath.c_str());
        filepath = tcl_file_name + "_BRBforce" + number + ".txt";
        std::remove(filepath.c_str());
    }
} // end of for loop

EndTime = time(NULL);
runtime << "NLRHA start reading LateralDisp.txt: " << EndTime - StartTime << " sec." << std::endl;

std::ifstream inLateralDisp;
filepath = tcl_file_name + "LateralDisp.txt";
inLateralDisp.open(filepath.c_str());

std::vector<double> NLRHAinterStoryDrift;
std::vector<double> NLRHAlateralDisp;
std::vector<double> NLRHAmoment;

double opMaxRoofDisp = 0;
double opMaxRoofAcc = 0;
double opMaxRoofAccb = 0;
double opMaxDisp = 0;
double opMaxAcc = 0;
int opPosMaxDisp = 0;
int opPosMaxAcc = 0;

NLRHAinterStoryDrift.resize(CoreMassID.size());
NLRHAlateralDisp.resize(CoreMassID.size());
NLRHAmoment.resize(CoreMassID.size());
for (int i = 0; i != NLRHAmoment.size(); i++) {
    NLRHAmoment[i] = 0;
}

while (inLateralDisp>>value){
    double last_value = value;
```

```
                if (std::abs(value) >= NLRHAlateralDisp[0]) {
                    NLRHAlateralDisp[0] = std::abs(value);
                }
                for (int j = 1; j != NLRHAlateralDisp.size(); j++) {
                    inLateralDisp >> value;
                    if (std::abs(value) >= NLRHAlateralDisp[j]) {
                        NLRHAlateralDisp[j] = std::abs(value);
                    }
                    if (std::abs(value - last_value) / MassSpacing >= NLRHAinterStoryDrift[j]) {
                        NLRHAinterStoryDrift[j] = std::abs(value - last_value) / MassSpacing;
                    }
                    if (std::abs(value) >= opMaxDisp) {
                        opMaxDisp = std::abs(value);
                        opPosMaxDisp = j;
                    }
                    last_value = value;
                }
            }
            opMaxRoofDisp = NLRHAlateralDisp[NLRHAlateralDisp.size() - 1];
            inLateralDisp.close();

            EndTime = time(NULL);
            runtime << "NLRHA reading rotation.txt:" << EndTime - StartTime << "sec." << std::endl;

            std::ifstream inRotation;
            filepath = tcl_file_name + "Rotation.txt";
            inRotation.open(filepath.c_str());
            if (!inRotation) {
                std::cerr << "error";
            }
            while(inRotation >> value){
                double last_value = value;
                for (int j = 1; j != NLRHAmoment.size(); j++) {
                    inRotation >> value;
                    if (std::abs(value - last_value) / MassSpacing *EI((double)j*MassSpacing)>= NLRHAmoment[j-1]) {
                        NLRHAmoment[j - 1] = std::abs(value - last_value) / MassSpacing * EI((double)j*MassSpacing);
                    }
                    last_value = value;
                }
            }
            inRotation.close();

            EndTime = time(NULL);
            runtime << "NLRHA reading Acceleration.txt: " << EndTime - StartTime << " sec." << std::endl;

            std::ifstream inLateralAcc;
            filepath = tcl_file_name + "LateralAcceleration.txt";
            inLateralAcc.open(filepath.c_str());
            for (int i = 0; i != EQtime.size(); i++) {
                for (int j = 0; j != CoreNodeElevation.size(); j++) {
```

```
                    inLateralAcc >> value;
                    if (std::abs(value) >= opMaxAcc) {
                        opMaxAcc = std::abs(value);
                        opPosMaxAcc = j;
                    }
                    if (j == CoreNodeElevation.size() - 1) {
                        if (std::abs(value) >= opMaxRoofAcc) {
                            opMaxRoofAcc = std::abs(value);
                        }
                        if (std::abs(value + EQacc[i] * EQscalefactor) >= opMaxRoofAccb) {
                            opMaxRoofAccb = std::abs(value + EQacc[i] * EQscalefactor);
                        }
                    }
                }
            }
            inLateralAcc.close();

            EndTime = time(NULL);
            runtime << "NLRHA finish reading LateralAcceleration.txt: " << EndTime - StartTime << " sec." << std::endl;

            if (IfRemoveOpenseesResultFile == 1) {
                filepath = tcl_file_name + "LateralDisp.txt";
                std::remove(filepath.c_str());
                filepath = tcl_file_name + "LateralAcceleration.txt";
                std::remove(filepath.c_str());
                filepath = tcl_file_name + "Rotation.txt";
                std::remove(filepath.c_str());
            }

            for (int i = 0; i != NLRHAinterStoryDrift.size(); i++) {
                InterStoryNLRHA << NLRHAinterStoryDrift[i] << "\t";
            }
            InterStoryNLRHA << std::endl;

            for (int i = 0; i != NLRHAlateralDisp.size(); i++) {
                DriftNLRHA << NLRHAlateralDisp[i] << "\t";
            }
            DriftNLRHA << std::endl;

            for (int i = 0; i != NLRHAmoment.size(); i++) {
                MomentNLRHA << NLRHAmoment[i] << "\t";
            }
            MomentNLRHA << std::endl;

            std::ifstream inOpenseesDynamicTestResult;
            inOpenseesDynamicTestResult.open("OpenseesDynamicResult.txt");
            std::string OpSresult;
            inOpenseesDynamicTestResult >> OpSresult;
            inOpenseesDynamicTestResult.close();
            if (IfRemoveOpenseesResultFile == 1) {
                std::remove("OpenseesDynamicResult.txt");
                std::remove(tcl_file_name.c_str());
            }
```

```
            EndTime = time(NULL);
            runtime << "NLRHA reading BaseShear.txt: " << EndTime - StartTime << " sec." << std::endl;

            std::ifstream inBaseShear;
            filepath = tcl_file_name + "BaseShear.txt";
            inBaseShear.open(filepath.c_str());
            double b1, b2, b3;
            double MaxBaseShear = 0;
            while (inBaseShear >> b1) {
                inBaseShear >> b2 >> b3;
                if (std::abs(b1 + b2 + b3) >= MaxBaseShear) {
                    MaxBaseShear = std::abs(b1 + b2 + b3);
                }
            }
            inBaseShear.close();
            if (IfRemoveOpenseesResultFile == 1) {
                std::remove(filepath.c_str());
            }

            EndTime = time(NULL);
            runtime << "NLRHA reading BaseOTmoment.txt: " << EndTime - StartTime << " sec." << std::endl;

            std::ifstream inOTmoment;
            filepath = tcl_file_name + "BaseOTmoment.txt";
            inOTmoment.open(filepath.c_str());
            double MaxOTmoment = 0;
            while (inOTmoment >> value) {
                if (std::abs(value) >= MaxOTmoment) {
                    MaxOTmoment = std::abs(value);
                }
            }
            inOTmoment.close();
            if (IfRemoveOpenseesResultFile == 1) {
                std::remove(filepath.c_str());
            }

            EndTime = time(NULL);
            runtime << "NLRHA reading Column.txt: " << EndTime - StartTime << " sec." << std::endl;

            std::vector<double> NLRHAcolForce; // from bottom to top
            NLRHAcolForce.resize(kd.size());
            for (int i = 0; i != kd.size(); i++) {
                std::ifstream inColForce;
                std::string number = std::to_string(i + 1);
                filepath = tcl_file_name + "_Column" + number + ".txt";
                inColForce.open(filepath.c_str());
                double maxcolforce = 0;
                while (inColForce >> value) {
                    if (std::abs(value) >= maxcolforce) {
                        maxcolforce = std::abs(value);
                    }
                    for (int j = 0; j != 23; j++) {
```

```
                    inColForce >> value;
                }
            }
            NLRHAcolForce[i] = maxcolforce;
            inColForce.close();
            if (IfRemoveOpenseesResultFile == 1) {
                std::remove(filepath.c_str());
            }
        }

        EndTime = time(NULL);
        runtime << "NLRHA start outputresults: " << EndTime - StartTime << " sec." << std::endl;

        for (int i = 0; i != OutriggerElevationVec.size(); i++) {
            NLRHAoutput << OutriggerElevationVec[i] << "\t";
        }
        NLRHAoutput << h << "\t";
        for (int i = 0; i != alpha.size(); i++) {
            NLRHAoutput << alpha[i] << "\t";
        }
        NLRHAoutput << OpSresult << "\t";
        for (int i = 0; i != OpenseesPeriod.size(); i++) {
            NLRHAoutput << OpenseesPeriod[i] << "\t";
        }
        for (int i = 0; i != uBRByOpensees.size(); i++) {
            NLRHAoutput << uBRByOpensees[i] * 1000 << "\t"; // unit: mm
        }
        for (int i = 0; i != BRBductilityOpensees.size(); i++) {
            NLRHAoutput << BRBductilityOpensees[i] << "\t";
        }
        for (int i = 0; i != CPD.size(); i++) {
            NLRHAoutput << CPD[i] << "\t";
        }
        NLRHAoutput << opMaxRoofDisp / h << "\t" << opMaxRoofDisp << "\t" << opMaxDisp << "\t" << opPosMaxDisp << "\t";
        NLRHAoutput << opMaxRoofAcc <<"\t"<< opMaxRoofAccb << "\t" << opMaxAcc << "\t" << opPosMaxAcc << "\t";
        for (int i = 0; i != BRBenergy.size(); i++) {
            NLRHAoutput << BRBenergy[i]*2.0 << "\t"; // two BRBs, two sides
        }
        NLRHAoutput << InputEnergy << "\t";
        for (int i = 0; i != kd.size(); i++) {
            NLRHAoutput << kd[i] / kt[i] << "\t" << kd[i] / kc << "\t" << kd[i] * (1 / kt[i] + alpha[i] / kc) << "\t" << lt*lt*h*kb[i] / EIb << "\t";
        }
        NLRHAoutput << EQname << "\t" << EQscalefactor << "\t";
        for (int i = 0; i != BRBenergy.size(); i++) {
            NLRHAoutput << BRBenergy[i] * 2.0 / InputEnergy * 100 << "\t";
        }
        NLRHAoutput << MaxBaseShear << "\t" << MaxOTmoment << "\t";
        for (int i = 0; i != NLRHAcolForce.size(); i++) {
            NLRHAoutput << NLRHAcolForce[i] << "\t";
```

```
            }
            NLRHAoutput << std::endl;

            if (IfRemoveOpenseesResultFile == 1) {
                remove((tcl_file_name + ".tcl").c_str());
                remove((tcl_file_name + "RoofDisp.txt").c_str());
                remove((tcl_file_name + "RoofAcc.txt").c_str());
            }

        }

        tired++;
        EndTime = time(NULL);
        runtime << "end: " << EndTime - StartTime << " sec." << std::endl;
    }
    std::cout << "finish" << std::endl;
    std::ifstream inpath;
    inpath.open("path.txt");
    if (!inpath) {
        std::cerr << "[error][inpath]" << std::endl;
    }
    std::string path;
    inpath >> path;
    path = path + "matlabmesh.exe";
    system(path.c_str());

    EndTime = time(NULL);
    std::cout << "Program running time: " << EndTime - StartTime << "sec." << std::endl;

    std::ofstream Status;
    Status.open("status.txt");
    Status << "completed. (" << EndTime - StartTime << " sec.)" << std::endl;
    std::cout << "completed. (" << EndTime - StartTime << " sec.)" << std::endl;
    if (ifpause == 1) {
        system("pause");
    }
    return 0;
}
```

```
void getL_vec()
{
    for (int i = 0; i != segment_vec.size(); i++) {
        segment_vec[i].CleanLambda();
    }
    double yieldisp = h / YieldDrift;
    if (IfDisplayDetail == 1) {
        std::cout << "MODAL ANALYSIS INITIATED (NUMERICAL METHOD)" << std::endl;
    }
    int dof = seg_num * 2;
    double cont = 0; // cont = 0 for the first iteration
    double prev_ans = 0;
    double prev_ans_p = 0;
    double ans = 0;
    double ans_p = 0;
    L_vec.resize(0);
    Lp_vec.resize(0);
    double L = 0;
    double Lp = 0;
    int slope = 0; //1 for positive slope and -1 for negative slope
    int slope_p = 0;
    int prev_slope = 0; //1 for positive slope and -1 for negative slope
    int prev_slope_p = 0;

    // construction force matrix induced from rotational springs
    matrix C(seg_num - 1, seg_num - 1);
    matrix Cd(seg_num - 1, seg_num - 1);
    matrix D(seg_num - 1, seg_num - 1);
    matrix D_plastic(seg_num - 1, seg_num - 1);

    for (int i = 0; i != seg_num - 1; i++) {
        for (int j = 0; j != seg_num - 1; j++) {
            if (i == j && i != seg_num - 2) {
                C.assign(i, j, kc_seg[i] + kc_seg[i + 1]);
                Cd.assign(i, j, (kc_seg[i] + kc_seg[i + 1]) / kd[i]);
                D.assign(i, j, (1 + (kc_seg[i] + kc_seg[i + 1])*(1 / kd[i] + 1 / kt[i])) / lt);
                D_plastic.assign(i, j, (1 + (kc_seg[i] + kc_seg[i + 1])*(1 / kd_plastic[i] + 1 / kt[i])) / lt);
            }
            else if (i == j && i == seg_num - 2) {
                C.assign(i, j, kc_seg[i]);
                Cd.assign(i, j, kc_seg[i] / kd[i]);
                D.assign(i, j, (1 + kc_seg[i] * (1 / kd[i] + 1 / kt[i])) / lt);
                D_plastic.assign(i, j, (1 + kc_seg[i] * (1 / kd_plastic[i] + 1 / kt[i])) / lt);
            }
            else if (i - j == 1) {
                C.assign(i, j, -1 * kc_seg[i]);
                C.assign(j, i, -1 * kc_seg[i]);
                Cd.assign(i, j, -1 * kc_seg[i] / kd[i]);
                Cd.assign(j, i, -1 * kc_seg[i] / kd[j]);
                D.assign(i, j, (-1 * kc_seg[i] * (1 / kd[i] + 1 / kt[i])) / lt);
```

```
                D_plastic.assign(i, j, (-1 * kc_seg[i] * (1 / kd_plastic[i] + 1 / kt[i])) / lt);
            }
            else if (j - i == 1) {
                D.assign(i, j, (-1 * kc_seg[j] * (1 / kd[i] + 1 / kt[i])) / lt);
                D_plastic.assign(i, j, (-1 * kc_seg[j] * (1 / kd_plastic[i] + 1 / kt[i])) / lt);
            }
        }
    }

    matrix kg, kg_plastic;
    kg = (2 * lt / h / h)*C*D.inverse();
    kg_plastic = (2 * lt / h / h)*C*D_plastic.inverse();
    kg = kg.InsertZeroRowColumn();
    kg_plastic = kg_plastic.InsertZeroRowColumn();

    // end of construction of force matrix induced from rotational springs
    matrix k_matrix(dof, dof);
    matrix k_matrix_p(dof, dof);
    matrix thetaYmatrix(seg_num - 1, 1);
    matrix udymatrix(seg_num - 1, 1);
    std::vector<std::vector<double>> thetay_vec;
    thetay_vec.resize(0);
    Ytop_mode.resize(0);

    // while loop to find solutions
    while (L_vec.size() < sol_num || Lp_vec.size() < sol_num) {
        for (int i = 0; i != seg_num; i++) {
            while ((1 - cos(L)*cosh(L)) == 0) {
                L = L + dL; // to avoid denominator being 0
            }
            while ((1 - cos(Lp)*cosh(Lp)) == 0) {
                Lp = Lp + dL;
            }
            segment_vec[i].UnitMatrix(L, h, EIb);
            k_matrix.Merge(segment_vec[i].GetBasicMatrix(), 2 * i - 2, 2 * i - 2);
            k_matrix_p.Merge(segment_vec[i].GetBasicMatrix(), 2 * i - 2, 2 * i - 2);
        }

        k_matrix.Merge(kg, 1, 1);
        k_matrix_p.Merge(kg_plastic, 1, 1);
        ans = k_matrix.det();
        ans_p = k_matrix_p.det();

        if (ans >= prev_ans) {
            slope = 1;
        }
        else {
            slope = -1;
        }
        if (ans_p > prev_ans_p) {
            slope_p = 1;
```

```
        }
        else {
            slope_p = -1;
        }
        k_matrix.Zero();
        k_matrix_p.Zero();

        if (ans*prev_ans <= 0 && slope == prev_slope && cont == 1) {
            double aans = dL * (-1 * prev_ans) / (ans - prev_ans) + L - dL;
            if (L_vec.size() < sol_num) {
                L_vec.push_back(aans);
                if (IfDisplayDetail == 1) {
                    std::cout << aans;
                    std::cout << "\t" << L_vec.size() << " of " << sol_num << " Elastic mode(s) found" << std::endl;
                }
                for (int i = 0; i != seg_num; i++) {
                    segment_vec[i].UnitMatrix(aans, h, EIb);
                    k_matrix.Merge(segment_vec[i].GetBasicMatrix(), 2 * i - 2, 2 * i - 2);
                }

                k_matrix.Merge(kg, 1, 1);
                k_matrix = (EIb / h / h / h)*k_matrix;

                matrix u; // eigenvector
                if (L_vec.size() >= 1) {
                    u = k_matrix.EigenVector(k_matrix.get_row() - 2, yieldisp);
                    Ytop_mode.push_back(u.Cell(u.get_row() - 2, 0));
                    std::vector<double> temp;
                    temp.resize(seg_num - 1);
                    for (int j = 0; j != seg_num - 1; j++) {
                        temp[j] = u.Cell(j * 2 + 1, 0) / h;
                    }
                    thetay_vec.push_back(temp);
                }

                for (int i = 0; i != seg_num; i++) {
                    matrix aau(4, 1);
                    for (int j = 0; j != 4; j++) {
                        if (2 * i - 2 + j >= 0) {
                            aau.assign(j, 0, u.Cell(2 * i - 2 + j, 0));
                        }
                    }
                    segment_vec[i].L.push_back(aans); // pass lambda
                    segment_vec[i].AssignAu(segment_vec[i].L.size() - 1, aau); // pass u matrix (4x1)
                }
            }
        }
        if (ans_p*prev_ans_p <= 0 && slope_p == prev_slope_p && cont == 1) {
            if (Lp_vec.size() < sol_num) {
                double aans = dL * (-1 * prev_ans_p) / (ans_p - prev_ans_p) + Lp - dL;
                Lp_vec.push_back(aans);
```

```cpp
                if (IfDisplayDetail == 1) {
                    std::cout << aans;
                    std::cout << "\t" << Lp_vec.size() << " of " << sol_num << " Plastic mode(s) found" << std::endl;
                }
            }
        }
        if (cont == 0 && IfDisplayDetail == 1) { // first iteration
            std::cout << "\t" << L_vec.size() << " of " << sol_num << " mode(s) found" << std::endl;
        }
        prev_ans = ans;
        prev_slope = slope;
        prev_ans_p = ans_p;
        prev_slope_p = slope_p;
        L = L + dL;
        Lp = Lp + dL;
        cont = 1;
    }

      // compute period
    std::vector<double> period;
    period.resize(L_vec.size());
    for (int i = 0; i != L_vec.size(); i++) {
        double lam = L_vec[i];
        double ans = std::pow((lam*lam*lam*lam*EIb / m / h / h / h / h), 0.5);
        ans = 2 * pi / ans;
        if (IfDisplayDetail == 1) {
            std::cout << "[Elastic] mode " << i + 1 << " vibration period = " << ans << " sec." << std::endl;
        }
        period[i] = ans;
    }
    for (int i = 0; i != Lp_vec.size(); i++) {
        double lam = Lp_vec[i];
        double ans = std::pow((lam*lam*lam*lam*EIb / m / h / h / h / h), 0.5);
        ans = 2 * pi / ans;
        if (IfDisplayDetail == 1) {
            std::cout << "[Plastic] mode " << i + 1 << " vibration period = " << ans << " sec." << std::endl;
        }
    }

    for (int i = 0; i != seg_num; i++) {
        if (IfDisplayDetail == 1) {
            std::cout << "Compute A matrix for segment " << i + 1 << std::endl;
        }
        segment_vec[i].computeA();
    }

    // create the modeshapes
    mode mo;
    mode_vec.resize(0);
    double x = 0;
    double a1, a2, a3, a4, phi0;
```

```
    double TotalModalMass = 0;

    for (int i = 0; i != sol_num; i++) {
        mode_vec.push_back(mo);
        double lam = L_vec[i] / h;
        for (int j = 0; j != seg_num; j++) {
            while (x <= segment_vec[j].length_ratio*h) {
                a1 = segment_vec[j].A[i].Cell(0, 0);
                a2 = segment_vec[j].A[i].Cell(1, 0);
                a3 = segment_vec[j].A[i].Cell(2, 0);
                a4 = segment_vec[j].A[i].Cell(3, 0);
                phi0 = a1 * cosh(lam*x) + a2 * sinh(lam*x) + a3 * cos(lam*x) + a4 * sin(lam*x);
                mode_vec[i].WriteData1(x + segment_vec[j].StartPos*h, phi0);
                x = x + dx;
            }
            x = 0;
        }

        mode_vec[i].ComputeUpdate(m, EIb);
        mo.kill();
        if (IfDisplayDetail == 1) {
            std::cout << "Compute Mn of mode " << i + 1 << std::endl;
        }
        mode_vec[mode_vec.size() - 1].Compute(m, EIb, lam, h0); // compute Mn, Ln, and Participation Factor
        TotalModalMass += mode_vec[mode_vec.size() - 1].ModalMass;
    }

    // compute modal mass participation ratios, then cout and output in file
    for (int i = 0; i != mode_vec.size(); i++) {
        mode_vec[i].AssignMassParticipationRatio(mode_vec[i].ModalMass / TotalModalMass);
        if (IfDisplayDetail == 1) {
            std::cout << "mode " << i + 1 << ", mass participation ratio = " << mode_vec[i].MassParticipationRatio * 100 << " %, participation factor = " << mode_vec[i].gamma << std::endl;
        }
    }
    double sum = 0;

    for (int i = 0; i != sol_num; i++) {
        sum += std::pow(Ytop_mode[i] * mode_vec[i].gamma*DesignSpec.GetSpD(period[i]), 2.0);
    }

    double ratio = yieldisp / std::sqrt(sum);

    for (int i = 0; i != seg_num - 1; i++) {
        sum = 0;
        for (int j = 0; j != thetay_vec.size(); j++) {
            sum += std::pow(thetay_vec[j][i] * mode_vec[j].gamma*DesignSpec.GetSpD(period[j])*ratio, 2.0);
        }
        thetaYmatrix.assign(i, 0, std::sqrt(sum));
    }
```

```
    udymatrix = Cd * D.inverse()*thetaYmatrix;
    uBRBy.resize(0);
    for (int i = 0; i != udymatrix.get_row(); i++) {
        uBRBy.push_back(udymatrix.Cell(i, 0));
    }
}
```

```cpp
void MakeOpenseesTCL(std::string s, std::string analysis, int MPAmode) {
    std::ofstream tcl;
    tcl.open(s + ".tcl");

    // make tcl file
    tcl << "# version6" << std::endl;
    tcl << "# " << s << std::endl;
    tcl << "# seg_num = " << seg_num << std::endl;
    tcl << "# x - axis, z - axis horizontal, y - axis vertival" << std::endl;
    tcl << "# unit: kN m" << std::endl << std::endl;
    tcl << "wipe all;" << std::endl;
    tcl << "model BasicBuilder -ndm 3;" << std::endl;

    // set parameters
    for (int i = 0; i != alpha.size(); i++) {
        tcl << "set alpha" << i + 1 << " " << alpha[i] << std::endl;
    }
    for (int i = 0; i != kt.size(); i++) {
        tcl << "set kt" << i + 1 << " " << kt[i] << std::endl;
    }
    for (int i = 0; i != kc_seg.size(); i++) {
        tcl << "set kc" << i + 1 << " " << kc_seg[i] << std::endl;
    }
    for (int i = 0; i != kd.size(); i++) {
        tcl << "set kd" << i + 1 << " " << kd[i] << std::endl;
    }
    for (int i = 0; i != kd.size(); i++) {
        tcl << "set BRBPostYieldStiffnessRatio" << i + 1 << " " << BRBPostYieldStiffnessRatio[i] << std::endl;
    }
    tcl << "set m " << m * MassSpacing << std::endl;
    tcl << "set h " << h << std::endl;
    tcl << "set lt " << lt << std::endl;
    // compute BRB and column length
    std::vector <double> OpenseesBRBlength, OpenseesColumnLength;
    OpenseesBRBlength.resize(seg_num - 1);
    OpenseesColumnLength.resize(seg_num - 1);

    for (int i = 0; i != OutriggerElevationVec.size(); i++) {
        if (i == 0) {
            if (OutriggerElevationVec[i] <= 1.0) {
                OpenseesBRBlength[i] = OutriggerElevationVec[i] * 0.5;
                OpenseesColumnLength[i] = OutriggerElevationVec[i] * 0.5;
            }
            else {
                OpenseesBRBlength[i] = 1.0;
                OpenseesColumnLength[i] = OutriggerElevationVec[i] - OpenseesBRBlength[i];
            }
        }
        else {
            if (OutriggerElevationVec[i] - OutriggerElevationVec[i - 1] <= 1.0) {
                OpenseesBRBlength[i] = (OutriggerElevationVec[i] - OutriggerElevationVec[i - 1])*0.5;
                OpenseesColumnLength[i] = (OutriggerElevationVec[i] -
```

```
                    OutriggerElevationVec[i - 1])*0.5;
            }
            else {
                OpenseesBRBlength[i] = 1.0;
                OpenseesColumnLength[i] = OutriggerElevationVec[i] - OutriggerElevationVec[i - 1] - OpenseesBRBlength[i];
            }
        }
    }

    for (int i = 0; i != OpenseesBRBlength.size(); i++) {
        tcl << "set BRB" << i + 1 << "_length " << OpenseesBRBlength[i] << std::endl;
    }
    for (int i = 0; i != OpenseesColumnLength.size(); i++) {
        tcl << "set column" << i + 1 << "_length " << OpenseesColumnLength[i] << std::endl;
    }

    tcl << "set numModes " << sol_num << std::endl;
    tcl << "set dampingratio " << h0 << std::endl;

    if (analysis == "modal1" || analysis == "modal2") {
        for (int i = 0; i != seg_num - 1; i++) {
            tcl << "set BRB_udy" << i + 1 << " " << 999999999.9 << std::endl; // FOR MODAL ANALYSIS
        }
    }
    else if (analysis == "NLRHA" || analysis == "MPA") {
        for (int i = 0; i != uBRByOpensees.size(); i++) {
            tcl << "set BRB_udy" << i + 1 << " " << std::abs(uBRByOpensees[i]) << std::endl; // FOR DYNAMIC ANALYSIS AND PUSHOVER ANALYSIS
        }
    }

    // set node - core structure
    std::vector<int> CoreNodeID;
    std::vector<int> CoreNodeIDOutrigger;

    CoreNodeElevation.resize(0);
    CoreNodeID.resize(0);
    CoreNodeIDOutrigger.resize(0);
    CoreMassID.resize(0);
    std::vector<int> CoreAssignMassID;
    CoreAssignMassID.resize(0);

    int count = 0;
    int id = 0;
    double Elev = 0;
    std::map<double, int> CoreNodeElevationMap;

    for (int i = 0; i <= (int)h; i++) {
        CoreNodeElevationMap.insert(std::make_pair((double)i, i));
        CoreMassID.push_back(i);
        if (count == (int)MassSpacing) {
            count = 0;
```

```
        }
        if (count == 0) {
            CoreAssignMassID.push_back(i);
        }
        if (count < (int)MassSpacing) {
            count++;
        }
    }

    for (int i = 0; i != OutriggerElevationVec.size(); i++) {
        if (CoreNodeElevationMap.find(OutriggerElevationVec[i]) == CoreNodeElevationMap.end()) {
            CoreNodeElevationMap.insert(std::make_pair(OutriggerElevationVec[i], (int)h + i + 1));
            CoreNodeIDOutrigger.push_back((int)h + i + 1);
        }
        else {
            CoreNodeIDOutrigger.push_back(CoreNodeElevationMap.find(OutriggerElevationVec[i])->second);
        }
    }
    for (std::map<double, int>::iterator it = CoreNodeElevationMap.begin(); it != CoreNodeElevationMap.end(); it++) {
        CoreNodeElevation.push_back(it->first);
        CoreNodeID.push_back(it->second);
    }

    for (int i = 0; i != CoreNodeElevation.size(); i++) {
        tcl << "node " << CoreNodeID[i] << " 0 " << CoreNodeElevation[i] << " 0" << std::endl;
        if (i == 0) {
            tcl << "fix " << CoreNodeID[i] << " 1 1 1 1 1 1" << std::endl;
        }
        else {
            tcl << "fix " << CoreNodeID[i] << " 1 1 0 0 1 1" << std::endl;
        }
    }
    for (int i = 0; i != CoreAssignMassID.size(); i++) {
        tcl << "mass " << CoreAssignMassID[i] << " 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09" << std::endl;
    }

    // set node - perimeter column and BRB
    Elev = 0;
    int LeftSide = 1000;
    int RightSide = 2000;
    tcl << "node " << LeftSide << " 0 0 " << -1 * lt << std::endl; // left base
    tcl << "node " << RightSide << " 0 0 " << lt << std::endl; // right base
    tcl << "fix " << LeftSide << " 1 1 1 1 1 1" << std::endl;
    tcl << "fix " << RightSide << " 1 1 1 1 1 1" << std::endl;
    for (int i = 0; i != OpenseesColumnLength.size(); i++) {
        Elev = Elev + OpenseesColumnLength[i];
        tcl << "node " << LeftSide + 2 * i + 1 << " 0 " << Elev << " " << -1 * lt << std::endl;
        tcl << "fix " << LeftSide + 2 * i + 1 << " 1 0 1 0 1 1" << std::endl;
```

```cpp
        tcl << "node " << RightSide + 2 * i + 1 << " 0 " << Elev << " " << lt << std::endl;
        tcl << "fix " << RightSide + 2 * i + 1 << " 1 0 1 0 1 1" << std::endl;
        Elev = Elev + OpenseesBRBlength[i];
        tcl << "node " << LeftSide + 2 * i + 2 << " 0 " << Elev << " " << -1 * lt << std::endl;
        tcl << "fix " << LeftSide + 2 * i + 2 << " 1 0 1 0 1 1" << std::endl;
        tcl << "node " << RightSide + 2 * i + 2 << " 0 " << Elev << " " << lt << std::endl;
        tcl << "fix " << RightSide + 2 * i + 2 << " 1 0 1 0 1 1" << std::endl;
    }

    //transformation
    tcl << "geomTransf Linear 1 1 0 0" << std::endl;

    // set parameters
    tcl << "# core structure" << std::endl;
    double E = 200000000;
    tcl << "set E " << E << std::endl;
    tcl << "set nu 0.3" << std::endl;
    tcl << "set SmallA 1e-9" << std::endl;
    tcl << "set G 2e7" << std::endl; //NEGLECT THE SHEAR DEFORMATION
    tcl << "set J 2e7" << std::endl; //NEGLECT THE TORSIONAL DEFOMATION
    std::vector<double> Iout;
    for (int i = 0; i != kt.size(); i++) {
        Iout.push_back(kt[i] * lt*lt*lt / E / 3.0);
        tcl << "set Iout" << i + 1 << " " << Iout[i] << std::endl;
    }
    tcl << "set ColumnMat 1" << std::endl; // perimeter column
    for (int i = 0; i != kd.size(); i++) {
        tcl << "set BRBmat" << i + 1 << " " << i + 2 << std::endl;
    }
    for (int i = 0; i != kd.size(); i++) {
        tcl << "set BRB_A" << i + 1 << " " << kd[i] * OpenseesBRBlength[i] / E << std::endl;
        if (analysis == "modal1" || analysis == "modal2") {
            tcl << "set BRB_Fy" << i + 1 << " " << 999999999.9 * kd[i] / (kd[i] * 1.0 / E) << std::endl;
        }
        else {
            tcl << "set BRB_Fy" << i + 1 << " " << std::abs(uBRByOpensees[i]) * kd[i] / (kd[i] * OpenseesBRBlength[i] / E) << std::endl;
        }
    }
    for (int i = 0; i != OpenseesColumnLength.size(); i++) {
        tcl << "set Column_A" << i + 1 << " " << kc * h / E << std::endl;
    }
    tcl << "uniaxialMaterial Elastic $ColumnMat $E" << std::endl; // the perimeter column
    for (int i = 0; i != kd.size(); i++) {
        tcl << "uniaxialMaterial Steel01 $BRBmat" << i + 1 << " $BRB_Fy" << i + 1 << " $E $BRBPostYieldStiffnessRatio" << i + 1 << std::endl;
    }

    // set element - core structure
    for (int i = 1; i != CoreNodeID.size(); i++) {
```

```cpp
        tcl << "set element_core_" << i << " " << i << std::endl;
    }
    for (int i = 1; i != CoreNodeID.size(); i++) {
        tcl << "element elasticBeamColumn $element_core_" << i << " " << CoreNodeID[i - 1] << " " << CoreNodeID[i] << " $SmallA $E $G $J " << EI(CoreNodeElevation[i] - 0.5) / E << " " << EI(CoreNodeElevation[i] - 0.5) / E << " 1" << std::endl;
    }

    // element column, BRB, outrigger
    int LeftOutrigger = 10000;
    int RightOutrigger = 20000;
    int LeftBRB = 30000;
    int RightBRB = 40000;
    int LeftColumn = 50000;
    int RightColumn = 60000;

    for (int i = 0; i != OpenseesColumnLength.size(); i++) {
        tcl << "set element_columnL" << i + 1 << " " << LeftColumn + i + 1 << std::endl;
        tcl << "set element_columnR" << i + 1 << " " << RightColumn + i + 1 << std::endl;
        tcl << "set element_BRBL" << i + 1 << " " << LeftBRB + i + 1 << std::endl;
        tcl << "set element_BRBR" << i + 1 << " " << RightBRB + i + 1 << std::endl;
        tcl << "set element_outriggerL" << i + 1 << " " << LeftOutrigger + i + 1 << std::endl;
        tcl << "set element_outriggerR" << i + 1 << " " << RightOutrigger + i + 1 << std::endl;
        // column
        if (i == 0) {
            tcl << "element elasticBeamColumn $element_columnL" << i + 1 << " " << LeftSide << " " << LeftSide + 1 << " $Column_A" << i + 1 << " $E $G $J $SmallA $SmallA 1" << std::endl;
            tcl << "element elasticBeamColumn $element_columnR" << i + 1 << " " << RightSide << " " << RightSide + 1 << " $Column_A" << i + 1 << " $E $G $J $SmallA $SmallA 1" << std::endl;
        }
        else {
            tcl << "element elasticBeamColumn $element_columnL" << i + 1 << " " << LeftSide + 2 * i - 1 << " " << LeftSide + 2 * i + 1 << " $Column_A" << i + 1 << " $E $G $J $SmallA $SmallA 1" << std::endl;
            tcl << "element elasticBeamColumn $element_columnR" << i + 1 << " " << RightSide + 2 * i - 1 << " " << RightSide + 2 * i + 1 << " $Column_A" << i + 1 << " $E $G $J $SmallA $SmallA 1" << std::endl;
        }
        // BRB
        tcl << "element truss $element_BRBL" << i + 1 << " " << LeftSide + 2 * i + 1 << " " << LeftSide + 2 * i + 2 << " $BRB_A" << i + 1 << " $BRBmat" << i + 1 << std::endl;
        tcl << "element truss $element_BRBR" << i + 1 << " " << RightSide + 2 * i + 1 << " " << RightSide + 2 * i + 2 << " $BRB_A" << i + 1 << " $BRBmat" << i + 1 << std::endl;
```

```cpp
        // outrigger
        tcl << "element elasticBeamColumn $element_outriggerL" << i + 1 << " " << LeftSide + 2 * i + 2 << " " << CoreNodeIDOutrigger[i] << " $SmallA $E $G $J " << Iout[i] << " " << Iout[i] << " 1" << std::endl;
        tcl << "element elasticBeamColumn $element_outriggerR" << i + 1 << " " << RightSide + 2 * i + 2 << " " << CoreNodeIDOutrigger[i] << " $SmallA $E $G $J " << Iout[i] << " " << Iout[i] << " 1" << std::endl;
    }

    // make recorder - modal analysis
    if (analysis == "modal1") {
        for (int j = 1; j <= sol_num; j++) {
            tcl << "recorder Node -file " << s + "ModeShape" << j << ".txt" << " -node ";
            for (int i = 0; i != CoreMassID.size(); i++) {
                tcl << CoreMassID[i] << " ";
            }
            tcl << "-dof 3 \"eigen " << j << "\"" << std::endl;
        }
    }

    // modal analysis
    if (analysis == "modal1" || analysis == "NLRHA") {
        tcl << "#modal analysis" << std::endl;
        tcl << "set fileperiod [open OpenseesPeriod.txt w]" << std::endl;
        tcl << "set ind 0" << std::endl;
        tcl << "set lambda [eigen $numModes]" << std::endl;
        tcl << "set omega {}" << std::endl;
        tcl << "set f {}" << std::endl;
        tcl << "set T {}" << std::endl;
        tcl << "set pi 3.141592654" << std::endl;
        tcl << "foreach lam $lambda {" << std::endl;
        tcl << "\t" << "lappend omega [expr sqrt($lam)]" << std::endl;
        tcl << "\t" << "lappend f [expr sqrt($lam)/(2*$pi)]" << std::endl;
        tcl << "\t" << "lappend T [expr (2*$pi)/sqrt($lam)]" << std::endl;
        tcl << "}" << std::endl;
        tcl << "puts \"modal analysis done\"" << std::endl;
        tcl << "puts \"vibration period (sec): \"" << std::endl;
        tcl << "foreach t $T {" << std::endl;
        tcl << "puts \" $t\"" << std::endl;
        tcl << "puts $fileperiod \"$t\"" << std::endl;
        tcl << "}" << std::endl;
        tcl << "record" << std::endl;
    }

    //dynamic analysis
    if (analysis == "NLRHA") {
        tcl << "#dynamic analysis" << std::endl;
        tcl << "#rayleigh $alphaM $betaK $betaKinit $betaKcomm" << std::endl;
        tcl << "set w1 [expr 2 * $pi / [lindex $T 0]]" << std::endl;
        tcl << "set w2 [expr 2 * $pi / [lindex $T 1]]" << std::endl;
        tcl << "set alphaM [expr $dampingratio*2.0*$w1*$w2 / ($w1 + $w2)]" << std::endl;
        tcl << "set betaK [expr 2.0*$dampingratio / ($w1 + $w2)]" <<
```

```
        std::endl;
    tcl << "rayleigh $alphaM $betaK 0.0 0.0" << std::endl;
    tcl << "puts \"\"" << std::endl;
    tcl << "puts \"rayleigh damping set\"" << std::endl;
    tcl << "puts \" damping ratio = $dampingratio\"" << std::endl;
    tcl << "puts \" alphaM = $alphaM\"" << std::endl;
    tcl << "puts \" betaK = $betaK\"" << std::endl;
    tcl << "#reading acceleration history" << std::endl;
    tcl << "set accelSeries \"Series -dt " << dt << " -filePath OpenseesEQ.txt -factor 1.0\"" << std::endl;
    tcl << "pattern UniformExcitation 1  3 -accel $accelSeries;" << std::endl;

    // make recorder - lateral disp. // in order to find the maximum lateral disp and inter-story drift
    tcl << "recorder Node -file " << s + "LateralDisp.txt" << " -node ";
    for (int i = 0; i != CoreMassID.size(); i++) {
        tcl << CoreMassID[i] << " ";
    }
    tcl << "-dof 3 disp" << std::endl;

    // make recorder - rotation. // in order to compute moment
    tcl << "recorder Node -file " << s + "Rotation.txt" << " -node ";
    for (int i = 0; i != CoreMassID.size(); i++) {
        tcl << CoreMassID[i] << " ";
    }
    tcl << "-dof 4 disp" << std::endl;

    tcl << "recorder Node -file " << s + "RoofDisp.txt" << " -node " << CoreMassID[CoreMassID.size() - 1] << " -dof 3 disp" << std::endl;

    // make recorder - lateral velocity
    tcl << "recorder Node -file " << s + "LateralVelocity.txt" << " -node ";
    for (int i = 0; i != CoreMassID.size(); i++) {
        tcl << CoreMassID[i] << " ";
    }
    tcl << "-dof 3 vel" << std::endl;

    // make recorder - lateral acceleration
    tcl << "recorder Node -file " << s + "LateralAcceleration.txt" << " -node ";
    for (int i = 0; i != CoreMassID.size(); i++) {
        tcl << CoreMassID[i] << " ";
    }
    tcl << "-dof 3 accel" << std::endl;
    tcl << "recorder Node -file " << s + "RoofAcc.txt" << " -node " << CoreMassID[CoreMassID.size() - 1] << " -dof 3 accel" << std::endl;

    // make recorder - base shear
    tcl << "recorder Node -file " << s + "BaseShear.txt" << " -node 0 " << LeftSide << " " << RightSide << " -dof 3 reaction" << std::endl;

    // make recorder - column axial force
    tcl << "recorder Node -file " << s + "BaseOTmoment.txt" << " -node 0 -dof 4 reaction" << std::endl;
```

```cpp

        for (int i = 0; i != kd.size(); i++) {
            tcl << "recorder Element -file " << s + "_BRBdeform" << i + 1 << ".txt -ele $element_BRBL" << i + 1 << " $element_BRBR" << i + 1 << " deformations" << std::endl;
            tcl << "recorder Element -file " << s + "_BRBforce" << i + 1 << ".txt -ele $element_BRBL" << i + 1 << " $element_BRBR" << i + 1 << " axialForce" << std::endl;
            tcl << "recorder Element -file " << s + "_Column" << i + 1 << ".txt -ele $element_columnL" << i + 1 << " $element_columnR" << i + 1 << " localForce" << std::endl;
        }

        tcl << "source LibAnalysisDynamicParameters.tcl" << std::endl;
        tcl << "set TmaxAnalysis " << EQt << std::endl;
        tcl << "set DtAnalysis " << dt << std::endl;
        tcl << "set Tol 1e-3;" << std::endl;
        tcl << "source DynamicAnalysis.tcl" << std::endl;
        tcl << "record" << std::endl;

        tcl << "puts \"\"" << std::endl;
        tcl << "puts \"dynamic analysis done\"" << std::endl;
    }

    // PUSHOVER
    std::vector<double> LoadPattern;
    LoadPattern.resize(0);
    if (analysis == "MPA") {
        // make recorder - roof disp
        tcl << "recorder Node -file " << s + "MPAmono_RoofDisp_Mode_" << MPAmodeNow + 1 << ".txt" << " -node " << CoreMassID[CoreMassID.size() - 1] << " -dof 3 disp" << std::endl;
        // make recorder - base shear
        tcl << "recorder Node -file " << s + "MPAmono_BaseShear_Mode_" << MPAmodeNow + 1 << ".txt" << " -node 0 " << LeftSide << " " << RightSide << " -dof 3 reaction" << std::endl;
        // make recorder - BRB
        for (int i = 0; i != kd.size(); i++) {
            tcl << "recorder Element -file " << s + "MPA_BRBdeform" << i + 1 << "_Mode_" << MPAmodeNow + 1 << ".txt -ele $element_BRBL" << i + 1 << " deformations" << std::endl;
            tcl << "recorder Element -file " << s + "MPA_BRBaxialForce" << i + 1 << "_Mode_" << MPAmodeNow + 1 << ".txt -ele $element_BRBL" << i + 1 << " axialForce" << std::endl;
            if (IfRemoveOpenseesResultFile == 3) {
                tcl << "recorder Element -file " << s + "MPA_BRBaxialForce" << i + 1 << "_Mode_" << MPAmodeNow + 1 << ".txt -ele $element_BRBL" << i + 1 << " axialForce" << std::endl;
            }
        }
        // make recorder
        if (SRSSmodeShapeType == 2 || SRSSmodeShapeType == 1) {
            tcl << "recorder Node -file " << s + "MPA_LateralDisplacement_Mode_" << MPAmodeNow + 1 << ".txt" << " -node ";
            for (int i = 0; i != CoreMassID.size(); i++) {
```

```
                tcl << CoreMassID[i] << "\t";
            }
            tcl << " -dof 3 disp" << std::endl;
        }

        tcl << "set IDctrlNode " << CoreMassID[CoreMassID.size() - 1] << std::endl;
        tcl << "set IDctrlDOF 3" << std::endl;

        std::ifstream inModeShape;
        inModeShape.open((s + "ModeShape" + std::to_string(MPAmode) + ".txt").c_str());
        if (!inModeShape) {
            std::cerr << "[error][inModeShape]" << std::endl;
            system("pause");
        }

        while (inModeShape >> value) {
            LoadPattern.push_back(-1 * value);
        }
        tcl << "pattern Plain 200 Linear {" << std::endl;
        for (int i = 0; i != CoreMassID.size(); i++) {
            tcl << "\t" << "load " << CoreMassID[i] << " " << "0.0 0.0 " << LoadPattern[i] << " 0.0 0.0 0.0" << std::endl;
        }
        tcl << "}" << std::endl;

        // analysis (monotonic)
        tcl << "set IDctrlNode " << CoreMassID[CoreMassID.size() - 1] << std::endl;
        tcl << "set DOFctrlNode 3" << std::endl;
        tcl << "set Incr " << MPAincr << std::endl;
        tcl << "set Nsteps " << MPAstep << std::endl;
        tcl << "set DmaxPush " << MPAtargetdisp << std::endl;
        tcl << "set PushFile [open OpenseesPushoverResult.txt w]" << std::endl;
        tcl << "constraints Plain" << std::endl;
        tcl << "numberer Plain" << std::endl;
        tcl << "system BandGeneral" << std::endl;
        tcl << "test NormDispIncr 1.0e-8 10" << std::endl;
        tcl << "algorithm Newton" << std::endl;
        tcl << "integrator DisplacementControl $IDctrlNode $DOFctrlNode $Incr" << std::endl;
        tcl << "analysis Static" << std::endl;
        tcl << "source PushoverAnalysis.tcl" << std::endl;
        tcl << "record" << std::endl;
        tcl << "wipe all" << std::endl;
    }

    std::cout << "Opensees file " << s + ".tcl" << " made." << std::endl;
    std::string filepath;

    filepath = "C:\\Opensees\\OpenSees.exe " + s + ".tcl";
    system(filepath.c_str());

    if (analysis == "MPA") {
```

```
        std::ifstream inMPAmonoDisp, inMPAmonoFor, inMPAmonoBRBDisp;
        inMPAmonoDisp.open((tcl_file_name + "MPAmono_RoofDisp_Mode_" + std::to_string(MPAmodeNow + 1) + ".txt").c_str());
        inMPAmonoFor.open((tcl_file_name + "MPAmono_BaseShear_Mode_" + std::to_string(MPAmodeNow + 1) + ".txt").c_str());

        std::vector<std::vector<double>> MPABRBdisp;
        MPABRBdisp.resize(kd.size());
        for (int i = 0; i != kd.size(); i++) {
            inMPAmonoBRBDisp.open((tcl_file_name + "MPA_BRBdeform" + std::to_string(i + 1) + "_Mode_" + std::to_string(MPAmodeNow + 1) + ".txt").c_str());
            if (!inMPAmonoBRBDisp) {
                std::cerr << "[error][inMPAmonoBRBDisp]" << "\t" << (tcl_file_name + "MPA_BRBdeform" + std::to_string(i + 1) + "_Mode_" + std::to_string(MPAmodeNow + 1) + ".txt") << std::endl;
                system("pause");
            }
            while (inMPAmonoBRBDisp >> value) {
                MPABRBdisp[i].push_back(std::abs(value));
            }
            inMPAmonoBRBDisp.close();
        }

        if (!inMPAmonoDisp || !inMPAmonoFor) {
            std::cerr << "[error][inMPAmono]" << std::endl;
            system("pause");
        }

        force.resize(0);
        disp.resize(0);

        while (inMPAmonoDisp >> value) {
            disp.push_back(std::abs(value));
            double core_base, left_bottom, right_bottom;
            inMPAmonoFor >> core_base;
            inMPAmonoFor >> left_bottom;
            inMPAmonoFor >> right_bottom;
            force.push_back(std::abs(core_base + left_bottom + right_bottom));
        }

        inMPAmonoDisp.close();
        inMPAmonoFor.close();

        if (force.size() > 0) {
            if (force[0] < 0) {
                for (int i = 0; i != force.size(); i++) {
                    force[i] = force[i] * -1;
                }
            }
        }

        std::ifstream inLateralDeform;
        inLateralDeform.open((s + "MPA_LateralDisplacement_Mode_" + std::to_string(MPAmodeNow + 1) + ".txt").c_str());
```

```
        std::vector<double> LateralDeform;
        LateralDeform.resize(CoreMassID.size());
        int iid = 0;
        while (inLateralDeform >> value) {
            LateralDeform[iid] = value;
            iid = iid + 1;
            if (iid == LateralDeform.size()) {
                iid = 0;
            }
        }

        double sum_pattern = 0;
        for (std::vector<double>::iterator it = LoadPattern.begin(); it != LoadPattern.end(); it++) {
            sum_pattern = sum_pattern + *it;
        }
        double LateralForceRatio = force[force.size() - 1] / sum_pattern;
        Es = 0;
        for (int i = 0; i != CoreMassID.size(); i++) {
            Es = Es + LoadPattern[i] * LateralDeform[i] * LateralForceRatio*0.5;
        }
        Es = std::abs(Es);
        Ed.resize(MPABRBdisp.size());
        for (int i = 0; i != MPABRBdisp.size(); i++) {
            double umax = std::abs(MPABRBdisp[i][MPABRBdisp[i].size() - 1]);
            if (umax <= std::abs(uBRByOpensees[i])) {
                Ed[i] = 0;
            }
            else {
                Ed[i] = 4 * kd[i] * std::abs(uBRByOpensees[i]) * (umax - BRBPostYieldStiffnessRatio[i] * umax - std::abs(uBRByOpensees[i]) + BRBPostYieldStiffnessRatio[i] * std::abs(uBRByOpensees[i]));
            }
        }

        // identify yield points
        yieldDisp.resize(0);
        yieldStep.resize(0);
        for (int ii = 0; ii != MPABRBdisp.size(); ii++) {
            for (int j = 1; j != MPABRBdisp[ii].size(); j++) {
                if (MPABRBdisp[ii][j] > std::abs(uBRByOpensees[ii]) && MPABRBdisp[ii][j - 1] <= std::abs(uBRByOpensees[ii])) {
                    double a, b, u1, u2;
                    u1 = MPABRBdisp[ii][j - 1];
                    u2 = MPABRBdisp[ii][j];
                    a = disp[j - 1];
                    b = disp[j];
                    yieldDisp.push_back((b - a)*(std::abs(uBRByOpensees[ii]) - u1) / (u2 - u1) + a);
                    yieldStep.push_back(j);
                }
            }
        }

```

```
        if (yieldStep.size() == 2) {
            if (yieldStep[0] >= yieldStep[1]) {
                int a, b;
                a = yieldStep[0];
                b = yieldStep[1];
                yieldStep[0] = b;
                yieldStep[1] = a;
            }
            if (yieldDisp[0] >= yieldDisp[1]) {
                double a, b;
                a = yieldDisp[0];
                b = yieldDisp[1];
                yieldDisp[0] = b;
                yieldDisp[1] = a;
            }
        }
        else if (yieldStep.size() == 1) {
            yieldStep.push_back(yieldStep[0]);
            yieldDisp.push_back(yieldDisp[0]);
        }
        else if (yieldStep.size() == 0) {
            yieldStep.push_back(disp.size() - 1);
            yieldStep.push_back(disp.size() - 1);
            yieldDisp.push_back(disp[disp.size() - 1]);
            yieldDisp.push_back(disp[disp.size() - 1]);
        }

        std::vector<double> KE_vec;
        std::vector<double> K1_vec;
        std::vector<double> K2_vec;

        for (int k = 1; k < yieldStep[0]; k++) {
            if (disp[k] != disp[k - 1]) {
                KE_vec.push_back((force[k] - force[k - 1]) / (disp[k] - disp[k
                  - 1]));
            }
        }
        for (int k = yieldStep[0]; k < yieldStep[1]; k++) {
            if (disp[k] != disp[k - 1]) {
                K1_vec.push_back((force[k] - force[k - 1]) / (disp[k] - disp[k
                  - 1]));
            }
        }
        for (int k = yieldStep[1]; k < disp.size(); k++) {
            if (disp[k] != disp[k - 1]) {
                K2_vec.push_back((force[k] - force[k - 1]) / (disp[k] - disp[k
                  - 1]));
            }
        }
        KE = 0;
        K1 = 0;
        K2 = 0;
        for (int k = 0; k != KE_vec.size(); k++) {
            KE += KE_vec[k];
        }
        for (int k = 0; k != K1_vec.size(); k++) {
```

```
                K1 += K1_vec[k];
            }
            for (int k = 0; k != K2_vec.size(); k++) {
                K2 += K2_vec[k];
            }

            if (KE_vec.size() > 0) {
                KE = KE / KE_vec.size();
            }

            if (K1_vec.size() > 0) {
                K1 = K1 / K1_vec.size();
            }
            else if (K1_vec.size() == 0) {
                K1 = KE;
            }

            if (K2_vec.size() > 0) {
                K2 = K2 / K2_vec.size();
            }
            else if (K2_vec.size() == 0) {
                K2 = KE;
            }
        }
    }
```

```
//Section Class
#ifndef _SECTION_H
#define _SECTION_H

#include <iostream>
#include <vector>
#include <cmath>
#include "matrix.h"

class Section {
private:
    matrix basic_matrix;
    matrix stiffness_matrix;
public:
    Section();
    double StartPos;
    double length_ratio;
    int SolutionNumber;
    void UnitMatrix(double, double);
    void UnitMatrix(double, double, double);
    void AssignLengthRatio(double);
    double GetLengthRatio();
    std::vector<double> L;
    std::vector<matrix> A;
    std::vector<matrix> au;
    void AssignAu(size_t, matrix);
    void computeA();
    void CleanLambda();
    matrix& GetBasicMatrix();
    matrix& GetStiffnessMatrix();
    double u(double);
};

inline void Section::CleanLambda() {
    Section::L.resize(0);
}

inline matrix& Section::GetBasicMatrix() {
    return Section::basic_matrix;
}

inline matrix& Section::GetStiffnessMatrix() {
    return Section::stiffness_matrix;
}

inline void Section::UnitMatrix(double L, double h) {
    Section::basic_matrix.set_dimension(4, 4);
    Section::stiffness_matrix.set_dimension(4, 4);
    double c, C, s, S, mom;
    c = cos(L*Section::length_ratio);
    C = cosh(L*Section::length_ratio);
    s = sin(L*Section::length_ratio);
    S = sinh(L*Section::length_ratio);
    mom = 1 - c * C;
    Section::basic_matrix.assign(0, 0, L*L*L*(S*c + C * s) / mom);
    Section::basic_matrix.assign(0, 1, L*L*S*s / mom);
```

```
    Section::basic_matrix.assign(0, 2, -1 * (S + s)*L*L*L / mom);
    Section::basic_matrix.assign(0, 3, (C - c)*L*L / mom);
    Section::basic_matrix.assign(1, 0, L*L*S*s / mom);
    Section::basic_matrix.assign(1, 1, (C*s - S * c)*L / mom);
    Section::basic_matrix.assign(1, 2, (c - C)*L*L / mom);
    Section::basic_matrix.assign(1, 3, (S - s)*L / mom);
    Section::basic_matrix.assign(2, 0, -1 * (S + s)*L*L*L / mom);
    Section::basic_matrix.assign(2, 1, (c - C)*L*L / mom);
    Section::basic_matrix.assign(2, 2, (S*c + C * s)*L*L*L / mom);
    Section::basic_matrix.assign(2, 3, -1 * S*s*L*L / mom);
    Section::basic_matrix.assign(3, 0, (C - c)*L*L / mom);
    Section::basic_matrix.assign(3, 1, (S - s)*L / mom);
    Section::basic_matrix.assign(3, 2, -1 * S*s*L*L / mom);
    Section::basic_matrix.assign(3, 3, (C*s - S * c)*L / mom);

    double length = h * Section::length_ratio;
    Section::stiffness_matrix.assign(0, 0, 12 / length / length / length);
    Section::stiffness_matrix.assign(0, 1, 6 / h / length / length);
    Section::stiffness_matrix.assign(0, 2, -12 / length / length / length);
    Section::stiffness_matrix.assign(0, 3, 6 / h / length / length);
    Section::stiffness_matrix.assign(1, 0, 6 / h / length / length);
    Section::stiffness_matrix.assign(1, 1, 4 / h / h / length);
    Section::stiffness_matrix.assign(1, 2, -6 / h / length / length);
    Section::stiffness_matrix.assign(1, 3, 2 / h / h / length);
    Section::stiffness_matrix.assign(2, 0, -12 / length / length / length);
    Section::stiffness_matrix.assign(2, 1, -6 / h / length / length);
    Section::stiffness_matrix.assign(2, 2, 12 / length / length / length);
    Section::stiffness_matrix.assign(2, 3, -6 / h / length / length);
    Section::stiffness_matrix.assign(3, 0, 6 / h / length / length);
    Section::stiffness_matrix.assign(3, 1, 2 / h / h / length);
    Section::stiffness_matrix.assign(3, 2, -6 / h / length / length);
    Section::stiffness_matrix.assign(3, 3, 4 / h / h / length);
}

inline Section::Section() {
    Section::basic_matrix.set_dimension(4, 4);
}

inline void Section::AssignLengthRatio(double c) {
    Section::length_ratio = c;
}

inline double Section::GetLengthRatio() {
    return Section::length_ratio;
}

inline void Section::computeA() {
    for (int i = 0; i != Section::SolutionNumber; i++) {
        double lam = Section::L[i];
        double c = cos(lam*Section::length_ratio);
        double C = cosh(lam*Section::length_ratio);
        double s = sin(lam*Section::length_ratio);
        double S = sinh(lam*Section::length_ratio);
        matrix m(4, 4);
        m.assign(0, 0, 1);
        m.assign(0, 1, 0);
```

```
        m.assign(0, 2, 1);
        m.assign(0, 3, 0);
        m.assign(1, 0, 0);
        m.assign(1, 1, lam);
        m.assign(1, 2, 0);
        m.assign(1, 3, lam);
        m.assign(2, 0, C);
        m.assign(2, 1, S);
        m.assign(2, 2, c);
        m.assign(2, 3, s);
        m.assign(3, 0, lam*S);
        m.assign(3, 1, lam*C);
        m.assign(3, 2, -1.0*lam*s);
        m.assign(3, 3, lam*c);
        Section::A.push_back(m.inverse()*Section::au[i]);
    }
}

inline void Section::AssignAu(size_t i, matrix m) {
    Section::au[i] = m;
}

inline void Section::UnitMatrix(double L, double h, double EIb) {
    double c, C, s, S, mom;
    c = cos(L*Section::length_ratio);
    C = cosh(L*Section::length_ratio);
    s = sin(L*Section::length_ratio);
    S = sinh(L*Section::length_ratio);
    mom = 1 - c * C;
    Section::basic_matrix.assign(0, 0, (EIb / h / h / h)*L*L*L*(S*c + C * s) / mom);
    Section::basic_matrix.assign(0, 1, (EIb / h / h / h)*L*L*S*s / mom);
    Section::basic_matrix.assign(0, 2, (EIb / h / h / h)*(-1) * (S + s) *L*L*L / mom);
    Section::basic_matrix.assign(0, 3, (EIb / h / h / h)*(C - c)*L*L / mom);
    Section::basic_matrix.assign(1, 0, (EIb / h / h / h)*L*L*S*s / mom);
    Section::basic_matrix.assign(1, 1, (EIb / h / h / h)*(C*s - S * c)*L / mom);
    Section::basic_matrix.assign(1, 2, (EIb / h / h / h)*(c - C)*L*L / mom);
    Section::basic_matrix.assign(1, 3, (EIb / h / h / h)*(S - s)*L / mom);
    Section::basic_matrix.assign(2, 0, (EIb / h / h / h)*(-1) * (S + s) *L*L*L / mom);
    Section::basic_matrix.assign(2, 1, (EIb / h / h / h)*(c - C)*L*L / mom);
    Section::basic_matrix.assign(2, 2, (EIb / h / h / h)*(S*c + C * s)*L*L*L / mom);
    Section::basic_matrix.assign(2, 3, (EIb / h / h / h)*(-1) * S*s*L*L / mom);
    Section::basic_matrix.assign(3, 0, (EIb / h / h / h)*(C - c)*L*L / mom);
    Section::basic_matrix.assign(3, 1, (EIb / h / h / h)*(S - s)*L / mom);
    Section::basic_matrix.assign(3, 2, (EIb / h / h / h)*(-1) * S*s*L*L / mom);
    Section::basic_matrix.assign(3, 3, (EIb / h / h / h)*(C*s - S * c)*L / mom);
}

#endif#pragma once
```

```
//mode Class
#ifndef _MODE_H
#define _MODE_H

#include <iostream>
#include <vector>
#include <cmath>
#include <map>
#include <string>

class mode {
private:
public:
    mode();
    void WriteData(double, double, double, double);
    void WriteData1(double, double);
    void Compute(double, double, double, double);
    void kill();
    void NormalizeTopDisplacement(double, double, double);
    void AssignMassParticipationRatio(double);
    void ComputeUpdate(double, double);
    double gamma; // modal participation factor
    double Lhn;
    double Mn;
    double Kn; // compute by using wn
    double dampingratio;
    double MassParticipationRatio;
    double wn;
    double lambda;
    double ModalMass;
    double Tn;
    int OutriggerLoc;
    std::vector<double> x_vec;
    std::vector<double> phi0_vec;
    std::vector<double> phi1_vec;
    std::vector<double> phi4_vec;
    std::map<double, double> q; // first: time, second: q(t)
    std::map<double, double> phi; // first: x, second phi(x)
};

inline mode::mode() {
    mode::x_vec.resize(0);
    mode::phi0_vec.resize(0);
    mode::phi4_vec.resize(0);
    mode::q.clear();
}
inline void mode::WriteData(double x, double phi0, double phi1, double phi4) {
    mode::x_vec.push_back(x);
    mode::phi0_vec.push_back(phi0);
    mode::phi1_vec.push_back(phi1);
    mode::phi4_vec.push_back(phi4);
}

inline void mode::WriteData1(double x, double phi0) {
    mode::x_vec.push_back(x);
    mode::phi0_vec.push_back(phi0);
```

```
}

inline void mode::kill() {
    mode::x_vec.resize(0);
    mode::phi0_vec.resize(0);
    mode::phi1_vec.resize(0);
    mode::phi4_vec.resize(0);
    mode::q.clear();
}
inline void mode::Compute(double m, double EI, double lam, double d) {
    double Mn_ans = 0;
    double Ln_ans = 0;
    for (int i = 1; i != mode::x_vec.size(); i++) {
        Mn_ans += (mode::phi0_vec[i - 1] * mode::phi0_vec[i - 1] + mode::phi0_vec[i] * mode::phi0_vec[i])*(mode::x_vec[i] - mode::x_vec[i - 1])*0.5;
        Ln_ans += (mode::phi0_vec[i - 1] + mode::phi0_vec[i])*(mode::x_vec[i] - mode::x_vec[i - 1])*0.5;
    }
    Mn_ans = Mn_ans * m; // only valid when m=constant
    Ln_ans = Ln_ans * m; // only valid when m=constant
    mode::Mn = Mn_ans;
    mode::Lhn = Ln_ans;
    mode::gamma = Ln_ans / Mn_ans;
    mode::lambda = lam;
    mode::wn = pow((lam*lam*lam*lam*EI / m), 0.5);
    mode::ModalMass = mode::Mn*mode::gamma*mode::gamma;
    mode::Kn = mode::wn*mode::wn*mode::Mn;
    mode::Tn = 2 * 3.14159265358979 / mode::wn;
    mode::dampingratio = d;
    //mode::Kn_t = Kn_t_ans;
}

inline void mode::NormalizeTopDisplacement(double NewTopDisp, double m, double EI) {
    // to normalize the building top displacement with the given value
    double OrigTopDisp = mode::phi0_vec[mode::phi0_vec.size() - 1];
    double mod = NewTopDisp / OrigTopDisp;
    for (int i = 0; i != mode::phi0_vec.size(); i++) {
        mode::phi0_vec[i] = mode::phi0_vec[i] * mod;
        mode::phi1_vec[i] = mode::phi1_vec[i] * mod;
    }
    mode::ComputeUpdate(m, EI);
}

inline void mode::AssignMassParticipationRatio(double r) {
    mode::MassParticipationRatio = r;
}

inline void mode::ComputeUpdate(double m, double EI) {
    double Mn_ans = 0;
    double Ln_ans = 0;
    //double Kn_t_ans = 0;
    for (int i = 1; i != mode::x_vec.size(); i++) {
        Mn_ans += (mode::phi0_vec[i - 1] * mode::phi0_vec[i - 1] + mode::phi0_vec[i] * mode::phi0_vec[i])*(mode::x_vec[i] - mode::x_vec
```

```
            [i - 1])*0.5;
        Ln_ans += (mode::phi0_vec[i - 1] + mode::phi0_vec[i])*(mode::x_vec[i] 
            - mode::x_vec[i - 1])*0.5;
    }
    Mn_ans = Mn_ans * m; // only valid when m=constant
    Ln_ans = Ln_ans * m; // only valid when m=constant
    mode::Mn = Mn_ans;
    mode::Lhn = Ln_ans;
    mode::gamma = Ln_ans / Mn_ans;
}

#endif#pragma once
```

```
//Spectrum Class
#ifndef _SPECTRUM_H
#define _SPECTRUM_H

#include <iostream>
#include <vector>
#include <cmath>
#include <fstream>
#include <string>
#include <map>

class Spectrum {
private:
public:
    Spectrum();
    std::vector<double> T_vec;
    std::vector<double> SpD_vec;
    std::vector<double> SpV_vec;
    std::vector<double> SpA_vec;

    void ReadSpvFromFile(std::string);
    void ReadSpaFromFile(std::string);
    double GetSpV(double);
    double GetSpD(double);
    double GetSpA(double);
};

inline double Spectrum::GetSpA(double t) {
    double ans;
    double a, b, x, y;
    int found = 0;
    for (int i = 1; i != Spectrum::T_vec.size(); i++) {
        if (t < Spectrum::T_vec[i] && t >= Spectrum::T_vec[i - 1]) {
            a = Spectrum::T_vec[i - 1];
            b = Spectrum::T_vec[i];
            x = Spectrum::SpA_vec[i - 1];
            y = Spectrum::SpA_vec[i];
            ans = x + (t - a)*(y - x) / (b - a);
            found = 1;
        }
    }
    if (found == 1) {
        return ans;
    }
    else if (found == 0) {
        a = Spectrum::T_vec[0];
        b = Spectrum::T_vec[1];
        x = Spectrum::SpA_vec[0];
        y = Spectrum::SpA_vec[1];
        return -1.0*((a - t)*(y - x) / (b - a) - x);
    }
}

inline double Spectrum::GetSpV(double t) {
    double ans;
    double a, b, x, y;
```

```
    int found = 0;
    for (int i = 1; i != Spectrum::T_vec.size(); i++) {
        if (t < Spectrum::T_vec[i] && t >= Spectrum::T_vec[i - 1]) {
            a = Spectrum::T_vec[i - 1];
            b = Spectrum::T_vec[i];
            x = Spectrum::SpV_vec[i - 1];
            y = Spectrum::SpV_vec[i];
            ans = x + (t - a)*(y - x) / (b - a);
            found = 1;
        }
    }
    if (found == 1) {
        return ans;
    }
    else if (found == 0)
    {
        a = Spectrum::T_vec[0];
        b = Spectrum::T_vec[1];
        x = Spectrum::SpV_vec[0];
        y = Spectrum::SpV_vec[1];
        return -1.0*((a - t)*(y - x) / (b - a) - x);
    }
}

inline double Spectrum::GetSpD(double t) {
    double ans;
    double a, b, x, y;
    int found = 0;
    for (int i = 1; i != Spectrum::T_vec.size(); i++) {
        if (t < Spectrum::T_vec[i] && t >= Spectrum::T_vec[i - 1]) {
            a = Spectrum::T_vec[i - 1];
            b = Spectrum::T_vec[i];
            x = Spectrum::SpD_vec[i - 1];
            y = Spectrum::SpD_vec[i];
            ans = x + (t - a)*(y - x) / (b - a);
            found = 1;
        }
    }
    if (found == 1) {
        return ans;
    }
    else if (found == 0) {
        a = Spectrum::T_vec[0];
        b = Spectrum::T_vec[1];
        x = Spectrum::SpD_vec[0];
        y = Spectrum::SpD_vec[1];
        return -1.0*((a - t)*(y - x) / (b - a) - x);
    }
}

inline Spectrum::Spectrum() {
    Spectrum::T_vec.resize(0);
    Spectrum::SpD_vec.resize(0);
    Spectrum::SpV_vec.resize(0);
    Spectrum::SpA_vec.resize(0);
}
```

```

void Spectrum::ReadSpaFromFile(std::string filename) {
    std::ifstream inSpa;
    inSpa.open(filename.c_str());
    if (!inSpa) {
        std::cerr << "[error][Spectrum]: reading AccelerationSpectrum.txt";
    }

    Spectrum::T_vec.resize(0);
    Spectrum::SpD_vec.resize(0);
    Spectrum::SpV_vec.resize(0);
    Spectrum::SpA_vec.resize(0);

    std::map<double, double> mmm;
    double s;
    double ss;
    while (inSpa >> s) {
        inSpa >> ss;
        mmm.insert(std::make_pair(s, ss));
    }
    for (std::map<double, double>::iterator it = mmm.begin(); it != mmm.end(); it++) {
        Spectrum::T_vec.push_back(it->first);
        Spectrum::SpA_vec.push_back(it->second);
    }

    Spectrum::SpD_vec.resize(Spectrum::T_vec.size());
    Spectrum::SpV_vec.resize(Spectrum::T_vec.size());

    for (int i = 0; i != Spectrum::T_vec.size(); i++) {
        Spectrum::SpD_vec[i] = Spectrum::SpA_vec[i] * Spectrum::T_vec[i] * Spectrum::T_vec[i] * 0.5 * 0.5 / 3.14159265358979 / 3.14159265358979;
        Spectrum::SpV_vec[i] = Spectrum::SpA_vec[i] * Spectrum::T_vec[i] / 2.0 / 3.14159265358979;
    }
    if (Spectrum::T_vec.size() > 0) {
        std::cout << "read acceleration spectrum successfully. (" << Spectrum::T_vec.size() << "data)" << std::endl;
    }

}

void Spectrum::ReadSpvFromFile(std::string filename) {
    std::ifstream in;
    in.open(filename.c_str());
    if (!in) {
        std::cerr << "[error][Spectrum]: reading VelocitySpectrum.txt";
    }

    Spectrum::T_vec.resize(0);
    Spectrum::SpD_vec.resize(0);
    Spectrum::SpV_vec.resize(0);
    Spectrum::SpA_vec.resize(0);

    std::map<double, double> mmm;
```

```
    double s;
    double ss;
    while (in >> s) {
        in >> ss;
        mmm.insert(std::make_pair(s, ss));
    }
    for (std::map<double, double>::iterator it = mmm.begin(); it != mmm.end(); it++) {
        Spectrum::T_vec.push_back(it->first);
        Spectrum::SpA_vec.push_back(it->second);
    }

    Spectrum::SpD_vec.resize(Spectrum::T_vec.size());
    Spectrum::SpA_vec.resize(Spectrum::T_vec.size());

    for (int i = 0; i != Spectrum::T_vec.size(); i++) {
        Spectrum::SpD_vec[i] = Spectrum::SpV_vec[i] * Spectrum::T_vec[i] * 0.5 / 3.14159265358979;
        Spectrum::SpA_vec[i] = Spectrum::SpV_vec[i] * 2 * 3.14159265358979 / Spectrum::T_vec[i];
    }
    if (Spectrum::T_vec.size() > 0) {
        std::cout << "read velocity spectrum successfully. (" << Spectrum::T_vec.size() << "data)" << std::endl;
    }
}

#endif#pragma once
```

```
//New Matrix Class
#ifndef _MATRIX_H
#define _MATRIX_H

#include <iostream>
#include <vector>
#include <cmath>

class matrix {
private:
    // mat in 2D vectors format
    std::vector<std::vector<double>> mat;
    int row_num;
    int col_num;
public:
    matrix();
    matrix(int row, int column);
    matrix(matrix&);
    ~matrix();
    void set_dimension(int, int);
    void assign(int, int, double);
    int get_row();
    int get_col();
    friend std::istream& operator>> (std::istream&, matrix&);
    friend std::ostream& operator<< (std::ostream&, const matrix&);
    friend matrix operator+ (const matrix&, const matrix&);
    friend matrix operator- (const matrix&, const matrix&);
    friend matrix operator* (double, const matrix&);
    friend matrix operator* (const matrix&, double);
    friend matrix operator* (const matrix&, const matrix&);
    matrix& operator= (const matrix&);
    matrix& operator- ();
    matrix transpose();
    double det();
    matrix inverse();
    matrix cofactor();
    void Merge(const matrix&, int, int);
    double Cell(int, int);
    void Zero();
    matrix InsertZeroRowColumn();
    matrix EigenVector();
    matrix EigenVector(int, double);
    void kill();
};

#endif#pragma once
```

```cpp
#include "matrix.h"

// initialize a matrix with zero dimension
matrix::matrix()
{
    matrix::mat.resize(0);
    matrix::col_num = 0;
    matrix::row_num = 0;
}

// initialize a zero matrix with given dimensions
matrix::matrix(int row, int column)
{
    matrix::mat.resize(row);
    for (int i = 0; i != row; i++) {
        matrix::mat[i].resize(column);
    }

    for (int i = 0; i != row; i++) {
        for (int j = 0; j != column; j++) {
            mat[i][j] = 0;
        }
    }

    matrix::col_num = column;
    matrix::row_num = row;
}

// initizlize a matrix with a give matrix
matrix::matrix(matrix& m)
{
    matrix::mat.resize(m.row_num);
    for (int i = 0; i != m.row_num; i++) {
        matrix::mat[i].resize(m.col_num);
    }

    for (int i = 0; i != m.row_num; ++i) {
        for (int j = 0; j != m.col_num; ++j) {
            matrix::mat[i][j] = m.mat[i][j];
        }
    }
    matrix::col_num = m.col_num;
    matrix::row_num = m.row_num;
}

// delete the original matrix and initialize a matrix with a given dimension
void matrix::set_dimension(int row, int column)
{
    matrix::mat.resize(0);
    matrix::mat.resize(row);
    for (int i = 0; i != row; i++) {
        matrix::mat[i].resize(column);
    }

    for (int i = 0; i != row; i++) {
        for (int j = 0; j != column; j++) {
```

```cpp
            mat[i][j] = 0;
        }
    }
    matrix::col_num = column;
    matrix::row_num = row;
}

// return number of rows
int matrix::get_row()
{
    return matrix::row_num;
}

// return number of columns
int matrix::get_col()
{
    return matrix::col_num;
}

// define operator >>, input the matrix elements from screen of file
std::istream& operator>> (std::istream& in, matrix& m)
{
    for (int i = 0; i != m.row_num; ++i) {
        for (int j = 0; j != m.col_num; ++j) {
            double t;
            in >> t;
            m.mat[i][j] = t;
        }
    }
    return in;
}

// define operator <<, output the matrix elements on screen or file
std::ostream& operator<< (std::ostream& out, const matrix& m)
{
    for (int i = 0; i != m.row_num; ++i) {
        for (int j = 0; j != m.col_num; ++j) {
            out << m.mat[i][j] << "\t";
        }
        out << std::endl;
    }
    return out;
}

// define operator +
matrix operator+ (const matrix& m1, const matrix& m2)
{
    if (m1.col_num != m2.col_num || m1.row_num != m2.row_num) {
        std::cout << "[error][matrix +]: matrix dimensions must consist" << std::endl;
        matrix ans;
        return ans;
    }
    else {
        matrix ans(m1.row_num, m1.col_num);
        for (int i = 0; i != m1.row_num; ++i) {
```

```cpp
                for (int j = 0; j != m1.col_num; ++j) {
                    ans.mat[i][j] = m1.mat[i][j] + m2.mat[i][j];
                }
            }
            return ans;
        }
}

// define operator -
matrix operator- (const matrix& m1, const matrix& m2)
{
    if (m1.col_num != m2.col_num || m1.row_num != m2.row_num) {
        std::cout << "[error][matrix -]: matrix dimensions must consist" << std::endl;
        matrix ans;
        return ans;
    }
    else {
        matrix ans(m1.row_num, m1.col_num);
        for (int i = 0; i != m1.row_num; ++i) {
            for (int j = 0; j != m1.col_num; ++j) {
                ans.mat[i][j] = m1.mat[i][j] - m2.mat[i][j];
            }
        }
        return ans;
    }
}

// define operator *, number*matrix
matrix operator* (double c, const matrix& m)
{
    matrix ans(m.row_num, m.col_num);
    for (int i = 0; i != m.row_num; ++i) {
        for (int j = 0; j != m.col_num; ++j) {
            ans.mat[i][j] = c * m.mat[i][j];
        }
    }
    return ans;
}

// define operator *, matrix*number
matrix operator* (const matrix& m, double c)
{
    matrix ans(m.row_num, m.col_num);
    for (int i = 0; i != m.row_num; ++i) {
        for (int j = 0; j != m.col_num; ++j) {
            ans.mat[i][j] = c * m.mat[i][j];
        }
    }
    return ans;
}

// define operator *, matrix*matrix
matrix operator* (const matrix& m1, const matrix& m2)
{
    if (m1.col_num != m2.row_num) {
```

```cpp
        std::cerr << "[error][matrix *]: column and row numbers must consist" << std::endl;
        matrix ans;
        return ans;
    }
    else {
        matrix ans(m1.row_num, m2.col_num);
        double elem = 0;
        for (int i = 0; i != m1.row_num; ++i) {
            for (int j = 0; j != m2.col_num; ++j) {
                for (int k = 0; k != m1.col_num; ++k) {
                    elem += m1.mat[i][k] * m2.mat[k][j];
                }
                ans.mat[i][j] = elem;
                elem = 0;
            }
        }
        return ans;
    }
}

// define operator =, delete the original matrix and assign with matrix m
matrix& matrix::operator= (const matrix& m)
{
    this->set_dimension(m.row_num, m.col_num);
    for (int i = 0; i != m.row_num; ++i) {
        for (int j = 0; j != m.col_num; ++j) {
            this->mat[i][j] = m.mat[i][j];
        }
    }
    return *this;
}

// define operator -, multiply the original matrix with -1
matrix& matrix::operator- ()
{
    for (int i = 0; i != this->row_num; ++i) {
        for (int j = 0; j != this->col_num; ++j) {
            this->mat[i][j] *= -1;
        }
    }
    return *this;
}

// return the transpose of the original matrix
matrix matrix::transpose()
{
    matrix ans(this->col_num, this->row_num);
    for (int i = 0; i != this->col_num; ++i) {
        for (int j = 0; j != this->row_num; ++j) {
            ans.mat[i][j] = this->mat[j][i];
        }
    }
    return ans;
}

```

```cpp
// return the determinate of a matrix
double matrix::det()
{
    if (this->row_num != this->col_num) {
        std::cerr << "[error][matrix det]: must be a square matrix" << std::endl;
        return 0;
    }
    else if (this->row_num == 0) {
        return 0;
    }
    else if (this->row_num == 1) {
        return this->mat[0][0];
    }
    else if (this->row_num == 2) {
        return this->mat[0][0] * this->mat[1][1] - this->mat[0][1] * this->mat[1][0];
    }
    else if (this->row_num == 3) {
        return this->mat[0][0] * this->mat[1][1] * this->mat[2][2]
            + this->mat[1][0] * this->mat[2][1] * this->mat[0][2]
            + this->mat[2][0] * this->mat[0][1] * this->mat[1][2]
            - this->mat[0][2] * this->mat[1][1] * this->mat[2][0]
            - this->mat[1][2] * this->mat[2][1] * this->mat[0][0]
            - this->mat[2][2] * this->mat[0][1] * this->mat[1][0];
    }
    else if (this->row_num == 4) {
        double a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p;
        a = this->mat[0][0];
        b = this->mat[0][1];
        c = this->mat[0][2];
        d = this->mat[0][3];
        e = this->mat[1][0];
        f = this->mat[1][1];
        g = this->mat[1][2];
        h = this->mat[1][3];
        i = this->mat[2][0];
        j = this->mat[2][1];
        k = this->mat[2][2];
        l = this->mat[2][3];
        m = this->mat[3][0];
        n = this->mat[3][1];
        o = this->mat[3][2];
        p = this->mat[3][3];
        return a * f*k*p - a * f*l*o - a * g*j*p + a * g*l*n + a * h*j*o - a * h*k*n - b * e*k*p + b * e*l*o + b * g*i*p - b * g*l*m - b * h*i*o + b * h*k*m + c * e*j*p - c * e*l*n - c * f*i*p + c * f*l*m + c * h*i*n - c * h*j*m - d * e*j*o + d * e*k*n + d * f*i*o - d * f*k*m - d * g*i*n + d * g*j*m;
    }
    else if (this->row_num == 5) {
        double a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y;
        a = this->mat[0][0];
        b = this->mat[0][1];
        c = this->mat[0][2];
```

```
            d = this->mat[0][3];
            e = this->mat[0][4];
            f = this->mat[1][0];
            g = this->mat[1][1];
            h = this->mat[1][2];
            i = this->mat[1][3];
            j = this->mat[1][4];
            k = this->mat[2][0];
            l = this->mat[2][1];
            m = this->mat[2][2];
            n = this->mat[2][3];
            o = this->mat[2][4];
            p = this->mat[3][0];
            q = this->mat[3][1];
            r = this->mat[3][2];
            s = this->mat[3][3];
            t = this->mat[3][4];
            u = this->mat[4][0];
            v = this->mat[4][1];
            w = this->mat[4][2];
            x = this->mat[4][3];
            y = this->mat[4][4];
            return a * g*m*s*y - a * g*m*t*x - a * g*n*r*y + a * g*n*t*w + a *
              g*o*r*x - a * g*o*s*w - a * h*l*s*y + a * h*l*t*x + a * h*n*q*y - a
              * h*n*t*v - a * h*o*q*x + a * h*o*s*v + a * i*l*r*y - a * i*l*t*w -
              a * i*m*q*y + a * i*m*t*v + a * i*o*q*w - a * i*o*r*v - a * j*l*r*x
              + a * j*l*s*w + a * j*m*q*x - a * j*m*s*v - a * j*n*q*w + a *
              j*n*r*v - b * f*m*s*y + b * f*m*t*x + b * f*n*r*y - b * f*n*t*w - b
              * f*o*r*x + b * f*o*s*w + b * h*k*s*y - b * h*k*t*x - b * h*n*p*y +
              b * h*n*t*u + b * h*o*p*x - b * h*o*s*u - b * i*k*r*y + b * i*k*t*w
              + b * i*m*p*y - b * i*m*t*u - b * i*o*p*w + b * i*o*r*u + b *
              j*k*r*x - b * j*k*s*w - b * j*m*p*x + b * j*m*s*u + b * j*n*p*w - b
              * j*n*r*u + c * f*l*s*y - c * f*l*t*x - c * f*n*q*y + c * f*n*t*v +
              c * f*o*q*x - c * f*o*s*v - c * g*k*s*y + c * g*k*t*x + c * g*n*p*y
              - c * g*n*t*u - c * g*o*p*x + c * g*o*s*u + c * i*k*q*y - c *
              i*k*t*v - c * i*l*p*y + c * i*l*t*u + c * i*o*p*v - c * i*o*q*u - c
              * j*k*q*x + c * j*k*s*v + c * j*l*p*x - c * j*l*s*u - c * j*n*p*v +
              c * j*n*q*u - d * f*l*r*y + d * f*l*t*w + d * f*m*q*y - d * f*m*t*v
              - d * f*o*q*w + d * f*o*r*v + d * g*k*r*y - d * g*k*t*w - d *
              g*m*p*y + d * g*m*t*u + d * g*o*p*w - d * g*o*r*u - d * h*k*q*y + d
              * h*k*t*v + d * h*l*p*y - d * h*l*t*u - d * h*o*p*v + d * h*o*q*u +
              d * j*k*q*w - d * j*k*r*v - d * j*l*p*w + d * j*l*r*u + d * j*m*p*v
              - d * j*m*q*u + e * f*l*r*x - e * f*l*s*w - e * f*m*q*x + e *
              f*m*s*v + e * f*n*q*w - e * f*n*r*v - e * g*k*r*x + e * g*k*s*w + e
              * g*m*p*x - e * g*m*s*u - e * g*n*p*w + e * g*n*r*u + e * h*k*q*x -
              e * h*k*s*v - e * h*l*p*x + e * h*l*s*u + e * h*n*p*v - e * h*n*q*u
              - e * i*k*q*w + e * i*k*r*v + e * i*l*p*w - e * i*l*r*u - e *
              i*m*p*v + e * i*m*q*u;
        }
        else {
            double ans = 0;
            double mu = 1;
            matrix redu(this->row_num - 1, this->col_num - 1);
            for (int j = 0; j != this->row_num - 1; ++j) {
                for (int k = 0; k != this->col_num - 1; ++k) {
                    redu.mat[j][k] = this->mat[j + 1][k + 1];
```

```
            }
        }
        ans += this->mat[0][0] * redu.det();
        for (int i = 1; i != this->col_num; ++i) {
            for (int j = 0; j != this->row_num - 1; ++j) {
                redu.mat[j][i - 1] = this->mat[j + 1][i - 1];
            }
            for (int h = 0; h != i; ++h) {
                mu *= -1.;
            }
            ans += this->mat[0][i] * redu.det() * mu;
            mu = 1;
        }
        redu.kill();
        return ans;
    }
}

matrix matrix::cofactor()
{
    matrix temp(this->row_num - 1, this->col_num - 1);
    matrix ans(this->row_num, this->col_num);
    for (int r = 0; r != this->row_num; ++r) {
        for (int c = 0; c != this->col_num; ++c) {
            for (int sr = 0; sr != this->row_num - 1; ++sr) {
                for (int sc = 0; sc != this->col_num - 1; ++sc) {
                    if (sr < r && sc < c) {
                        temp.mat[sr][sc] = this->mat[sr][sc];
                    }
                    else if (sr < r && sc >= c) {
                        temp.mat[sr][sc] = this->mat[sr][sc + 1];
                    }
                    else if (sr >= r && sc < c) {
                        temp.mat[sr][sc] = this->mat[sr + 1][sc];
                    }
                    else if (sr >= r && sc >= c) {
                        temp.mat[sr][sc] = this->mat[sr + 1][sc + 1];
                    }
                }
            }
            ans.mat[r][c] = temp.det() * pow(-1., r + c);
        }
    }
    return ans;
}

matrix matrix::inverse()
{
    if (this->row_num != this->col_num) {
        std::cerr << "[error][matrix -1]: must be a square matrix" << std::endl;
        matrix ans;
        return ans;
    }
    else if (this->det() == 0) {
        std::cerr << "[error][matrix -1]: det=0, the inverse doesn't exist" <<
```

```
            std::endl;
        matrix ans;
        return ans;
    }
    else if (this->row_num == 1 && this->col_num == 1) {
        matrix ans(1, 1);
        ans.assign(0, 0, 1.0 / this->Cell(0, 0));
        return ans;
    }
    else {
        matrix ans(*this);
        double k = 1. / ans.det();
        return k * ans.cofactor().transpose();
    }
}

void matrix::assign(int r, int c, double a) {
    this->mat[r][c] = a;
}

double matrix::Cell(int r, int c) {
    return this->mat[r][c];
}

void matrix::Merge(const matrix& m, int a, int b) {
    if (m.row_num > matrix::row_num || m.col_num > matrix::col_num) {
        std::cout << "[error][matrix merge]: dimension of the matrix to be merged exceed." << std::endl;
    }
    else {
        for (int i = 0; i != m.row_num; i++) {
            for (int j = 0; j != m.col_num; j++) {
                if ((a + i) >= 0 && (a + i) < matrix::row_num && (b + j) >= 0 && (b + j) < matrix::col_num) {
                    matrix::mat[a + i][b + j] = matrix::mat[a + i][b + j] + m.mat[i][j];
                }
            }
        }
    }
}

void matrix::Zero() {
    for (int i = 0; i != this->get_row(); i++) {
        for (int j = 0; j != this->get_col(); j++) {
            this->mat[i][j] = 0;
        }
    }
}

//insert rows and columns with 0s between the origional rows and columns
matrix matrix::InsertZeroRowColumn() {
    matrix ans(this->row_num * 2 - 1, this->col_num * 2 - 1);
    for (int i = 0; i != this->row_num; i++) {
        for (int j = 0; j != this->col_num; j++) {
            ans.assign(i * 2, j * 2, this->Cell(i, j));
```

```
        }
    }
    return ans;
}

// return eigen vector of this SINGULAR matrix
matrix matrix::EigenVector() {
    matrix rm(this->row_num - 1, this->col_num - 1);
    for (int i = 0; i != rm.row_num; i++) {
        for (int j = 0; j != rm.col_num; j++) {
            rm.assign(i, j, this->Cell(i + 1, j + 1));
        }
    }
    matrix rr(rm.row_num, 1);
    for (int i = 0; i != rr.row_num; i++) {
        rr.assign(i, 0, -1.0*this->Cell(i + 1, 0));
    }
    matrix ans(this->row_num, 1);
    ans.Zero();
    if (ans.row_num > 0) {
        ans.assign(0, 0, 1.0);
    }
    ans.Merge(rm.inverse()*rr, 1, 0);
    double length = ans.Cell(ans.row_num - 2, 0);
    for (int i = 0; i != ans.row_num; i++) {
        ans.assign(i, 0, ans.Cell(i, 0) / length);
    }
    return ans;
}

// return eigen vector of this SINGULAR matrix
matrix matrix::EigenVector(int n, double value) {
    matrix rm(this->row_num - 1, this->col_num - 1);
    for (int i = 0; i != rm.row_num; i++) {
        for (int j = 0; j != rm.col_num; j++) {
            rm.assign(i, j, this->Cell(i + 1, j + 1));
        }
    }
    matrix rr(rm.row_num, 1);
    for (int i = 0; i != rr.row_num; i++) {
        rr.assign(i, 0, -1.0*this->Cell(i + 1, 0));
    }
    matrix ans(this->row_num, 1);
    ans.Zero();
    if (ans.row_num > 0) {
        ans.assign(0, 0, 1.0);
    }
    ans.Merge(rm.inverse()*rr, 1, 0);
    double length = ans.Cell(n, 0);
    for (int i = 0; i != ans.row_num; i++) {
        ans.assign(i, 0, ans.Cell(i, 0) / length * value);
    }
    return ans;
}
```

Appendix B

OpenSees tcl script of
32-story Single DM model

```
# z - axis horizontal, y - axis vertival
# unit: kN m

wipe all;
model BasicBuilder -ndm 3;
set alpha1 0.06875
set alpha2 0.6875
set kt1 0.001
set kt2 2.43047e+07
set kc1 7.07045e+06
set kc2 785606
set kd1 0.001
set kd2 2.43047e+06
set BRBPostYieldStiffnessRatio1 1
set BRBPostYieldStiffnessRatio2 0.01
set m 225
set h 128
set lt 16
set BRB1_length 1
set BRB2_length 1
set column1_length 7.8
set column2_length 78.2
set numModes 6
set dampingratio 0.02
set BRB_udy1 0.00251817
set BRB_udy2 0.00511691
node 0 0 0 0
fix 0 1 1 1 1 1 1
node 1 0 1 0
fix 1 1 1 0 0 1 1
node 2 0 2 0
fix 2 1 1 0 0 1 1
node 3 0 3 0
fix 3 1 1 0 0 1 1
node 4 0 4 0
fix 4 1 1 0 0 1 1
node 5 0 5 0
fix 5 1 1 0 0 1 1
node 6 0 6 0
fix 6 1 1 0 0 1 1
node 7 0 7 0
fix 7 1 1 0 0 1 1
node 8 0 8 0
fix 8 1 1 0 0 1 1
node 129 0 8.8 0
fix 129 1 1 0 0 1 1
node 9 0 9 0
fix 9 1 1 0 0 1 1
node 10 0 10 0
fix 10 1 1 0 0 1 1
node 11 0 11 0
fix 11 1 1 0 0 1 1
node 12 0 12 0
fix 12 1 1 0 0 1 1
node 13 0 13 0
fix 13 1 1 0 0 1 1
node 14 0 14 0
fix 14 1 1 0 0 1 1
```

```
node 15 0 15 0
fix 15 1 1 0 0 1 1
node 16 0 16 0
fix 16 1 1 0 0 1 1
node 17 0 17 0
fix 17 1 1 0 0 1 1
node 18 0 18 0
fix 18 1 1 0 0 1 1
node 19 0 19 0
fix 19 1 1 0 0 1 1
node 20 0 20 0
fix 20 1 1 0 0 1 1
node 21 0 21 0
fix 21 1 1 0 0 1 1
node 22 0 22 0
fix 22 1 1 0 0 1 1
node 23 0 23 0
fix 23 1 1 0 0 1 1
node 24 0 24 0
fix 24 1 1 0 0 1 1
node 25 0 25 0
fix 25 1 1 0 0 1 1
node 26 0 26 0
fix 26 1 1 0 0 1 1
node 27 0 27 0
fix 27 1 1 0 0 1 1
node 28 0 28 0
fix 28 1 1 0 0 1 1
node 29 0 29 0
fix 29 1 1 0 0 1 1
node 30 0 30 0
fix 30 1 1 0 0 1 1
node 31 0 31 0
fix 31 1 1 0 0 1 1
node 32 0 32 0
fix 32 1 1 0 0 1 1
node 33 0 33 0
fix 33 1 1 0 0 1 1
node 34 0 34 0
fix 34 1 1 0 0 1 1
node 35 0 35 0
fix 35 1 1 0 0 1 1
node 36 0 36 0
fix 36 1 1 0 0 1 1
node 37 0 37 0
fix 37 1 1 0 0 1 1
node 38 0 38 0
fix 38 1 1 0 0 1 1
node 39 0 39 0
fix 39 1 1 0 0 1 1
node 40 0 40 0
fix 40 1 1 0 0 1 1
node 41 0 41 0
fix 41 1 1 0 0 1 1
node 42 0 42 0
fix 42 1 1 0 0 1 1
node 43 0 43 0
fix 43 1 1 0 0 1 1
```

```
node 44 0 44 0
fix 44 1 1 0 0 1 1
node 45 0 45 0
fix 45 1 1 0 0 1 1
node 46 0 46 0
fix 46 1 1 0 0 1 1
node 47 0 47 0
fix 47 1 1 0 0 1 1
node 48 0 48 0
fix 48 1 1 0 0 1 1
node 49 0 49 0
fix 49 1 1 0 0 1 1
node 50 0 50 0
fix 50 1 1 0 0 1 1
node 51 0 51 0
fix 51 1 1 0 0 1 1
node 52 0 52 0
fix 52 1 1 0 0 1 1
node 53 0 53 0
fix 53 1 1 0 0 1 1
node 54 0 54 0
fix 54 1 1 0 0 1 1
node 55 0 55 0
fix 55 1 1 0 0 1 1
node 56 0 56 0
fix 56 1 1 0 0 1 1
node 57 0 57 0
fix 57 1 1 0 0 1 1
node 58 0 58 0
fix 58 1 1 0 0 1 1
node 59 0 59 0
fix 59 1 1 0 0 1 1
node 60 0 60 0
fix 60 1 1 0 0 1 1
node 61 0 61 0
fix 61 1 1 0 0 1 1
node 62 0 62 0
fix 62 1 1 0 0 1 1
node 63 0 63 0
fix 63 1 1 0 0 1 1
node 64 0 64 0
fix 64 1 1 0 0 1 1
node 65 0 65 0
fix 65 1 1 0 0 1 1
node 66 0 66 0
fix 66 1 1 0 0 1 1
node 67 0 67 0
fix 67 1 1 0 0 1 1
node 68 0 68 0
fix 68 1 1 0 0 1 1
node 69 0 69 0
fix 69 1 1 0 0 1 1
node 70 0 70 0
fix 70 1 1 0 0 1 1
node 71 0 71 0
fix 71 1 1 0 0 1 1
node 72 0 72 0
fix 72 1 1 0 0 1 1
```

```
node 73 0 73 0
fix 73 1 1 0 0 1 1
node 74 0 74 0
fix 74 1 1 0 0 1 1
node 75 0 75 0
fix 75 1 1 0 0 1 1
node 76 0 76 0
fix 76 1 1 0 0 1 1
node 77 0 77 0
fix 77 1 1 0 0 1 1
node 78 0 78 0
fix 78 1 1 0 0 1 1
node 79 0 79 0
fix 79 1 1 0 0 1 1
node 80 0 80 0
fix 80 1 1 0 0 1 1
node 81 0 81 0
fix 81 1 1 0 0 1 1
node 82 0 82 0
fix 82 1 1 0 0 1 1
node 83 0 83 0
fix 83 1 1 0 0 1 1
node 84 0 84 0
fix 84 1 1 0 0 1 1
node 85 0 85 0
fix 85 1 1 0 0 1 1
node 86 0 86 0
fix 86 1 1 0 0 1 1
node 87 0 87 0
fix 87 1 1 0 0 1 1
node 88 0 88 0
fix 88 1 1 0 0 1 1
node 89 0 89 0
fix 89 1 1 0 0 1 1
node 90 0 90 0
fix 90 1 1 0 0 1 1
node 91 0 91 0
fix 91 1 1 0 0 1 1
node 92 0 92 0
fix 92 1 1 0 0 1 1
node 93 0 93 0
fix 93 1 1 0 0 1 1
node 94 0 94 0
fix 94 1 1 0 0 1 1
node 95 0 95 0
fix 95 1 1 0 0 1 1
node 96 0 96 0
fix 96 1 1 0 0 1 1
node 97 0 97 0
fix 97 1 1 0 0 1 1
node 98 0 98 0
fix 98 1 1 0 0 1 1
node 99 0 99 0
fix 99 1 1 0 0 1 1
node 100 0 100 0
fix 100 1 1 0 0 1 1
node 101 0 101 0
fix 101 1 1 0 0 1 1
```

```
node 102 0 102 0
fix 102 1 1 0 0 1 1
node 103 0 103 0
fix 103 1 1 0 0 1 1
node 104 0 104 0
fix 104 1 1 0 0 1 1
node 105 0 105 0
fix 105 1 1 0 0 1 1
node 106 0 106 0
fix 106 1 1 0 0 1 1
node 107 0 107 0
fix 107 1 1 0 0 1 1
node 108 0 108 0
fix 108 1 1 0 0 1 1
node 109 0 109 0
fix 109 1 1 0 0 1 1
node 110 0 110 0
fix 110 1 1 0 0 1 1
node 111 0 111 0
fix 111 1 1 0 0 1 1
node 112 0 112 0
fix 112 1 1 0 0 1 1
node 113 0 113 0
fix 113 1 1 0 0 1 1
node 114 0 114 0
fix 114 1 1 0 0 1 1
node 115 0 115 0
fix 115 1 1 0 0 1 1
node 116 0 116 0
fix 116 1 1 0 0 1 1
node 117 0 117 0
fix 117 1 1 0 0 1 1
node 118 0 118 0
fix 118 1 1 0 0 1 1
node 119 0 119 0
fix 119 1 1 0 0 1 1
node 120 0 120 0
fix 120 1 1 0 0 1 1
node 121 0 121 0
fix 121 1 1 0 0 1 1
node 122 0 122 0
fix 122 1 1 0 0 1 1
node 123 0 123 0
fix 123 1 1 0 0 1 1
node 124 0 124 0
fix 124 1 1 0 0 1 1
node 125 0 125 0
fix 125 1 1 0 0 1 1
node 126 0 126 0
fix 126 1 1 0 0 1 1
node 127 0 127 0
fix 127 1 1 0 0 1 1
node 128 0 128 0
fix 128 1 1 0 0 1 1
mass 0 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 1 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 2 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 3 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
```

```
mass 4 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 5 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 6 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 7 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 8 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 9 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 10 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 11 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 12 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 13 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 14 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 15 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 16 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 17 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 18 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 19 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 20 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 21 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 22 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 23 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 24 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 25 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 26 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 27 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 28 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 29 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 30 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 31 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 32 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 33 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 34 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 35 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 36 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 37 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 38 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 39 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 40 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 41 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 42 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 43 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 44 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 45 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 46 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 47 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 48 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 49 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 50 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 51 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 52 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 53 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 54 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 55 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 56 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 57 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 58 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 59 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 60 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 61 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
```

```
mass 62 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 63 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 64 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 65 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 66 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 67 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 68 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 69 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 70 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 71 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 72 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 73 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 74 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 75 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 76 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 77 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 78 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 79 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 80 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 81 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 82 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 83 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 84 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 85 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 86 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 87 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 88 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 89 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 90 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 91 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 92 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 93 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 94 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 95 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 96 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 97 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 98 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 99 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 100 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 101 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 102 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 103 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 104 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 105 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 106 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 107 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 108 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 109 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 110 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 111 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 112 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 113 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 114 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 115 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 116 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 117 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 118 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 119 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
```

```
mass 120 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 121 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 122 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 123 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 124 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 125 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 126 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 127 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
mass 128 1.00E-09 1.00E-09 $m 1.00E-09 1.00E-09 1.00E-09
node 1000 0 0 -16
node 2000 0 0 16
fix 1000 1 1 1 1 1 1
fix 2000 1 1 1 1 1 1
node 1001 0 7.8 -16
fix 1001 1 0 1 0 1 1
node 2001 0 7.8 16
fix 2001 1 0 1 0 1 1
node 1002 0 8.8 -16
fix 1002 1 0 1 0 1 1
node 2002 0 8.8 16
fix 2002 1 0 1 0 1 1
node 1003 0 87 -16
fix 1003 1 0 1 0 1 1
node 2003 0 87 16
fix 2003 1 0 1 0 1 1
node 1004 0 88 -16
fix 1004 1 0 1 0 1 1
node 2004 0 88 16
fix 2004 1 0 1 0 1 1
geomTransf Linear 1 1 0 0
# core structure
set E 2e+08
set nu 0.3
set SmallA 1e-9
set G 2e7
set J 2e7
set Iout1 6.82667e-09
set Iout2 165.92
set ColumnMat 1
set BRBmat1 2
set BRBmat2 3
set BRB_A1 5e-12
set BRB_Fy1 503634
set BRB_A2 0.0121523
set BRB_Fy2 1.02338e+06
set Column_A1 0.3111
set Column_A2 0.3111
uniaxialMaterial Elastic $ColumnMat $E
uniaxialMaterial Steel01 $BRBmat1 $BRB_Fy1 $E $BRBPostYieldStiffnessRatio1
uniaxialMaterial Steel01 $BRBmat2 $BRB_Fy2 $E $BRBPostYieldStiffnessRatio2
set element_core_1 1
set element_core_2 2
set element_core_3 3
set element_core_4 4
set element_core_5 5
set element_core_6 6
set element_core_7 7
set element_core_8 8
```

```
set element_core_9 9
set element_core_10 10
set element_core_11 11
set element_core_12 12
set element_core_13 13
set element_core_14 14
set element_core_15 15
set element_core_16 16
set element_core_17 17
set element_core_18 18
set element_core_19 19
set element_core_20 20
set element_core_21 21
set element_core_22 22
set element_core_23 23
set element_core_24 24
set element_core_25 25
set element_core_26 26
set element_core_27 27
set element_core_28 28
set element_core_29 29
set element_core_30 30
set element_core_31 31
set element_core_32 32
set element_core_33 33
set element_core_34 34
set element_core_35 35
set element_core_36 36
set element_core_37 37
set element_core_38 38
set element_core_39 39
set element_core_40 40
set element_core_41 41
set element_core_42 42
set element_core_43 43
set element_core_44 44
set element_core_45 45
set element_core_46 46
set element_core_47 47
set element_core_48 48
set element_core_49 49
set element_core_50 50
set element_core_51 51
set element_core_52 52
set element_core_53 53
set element_core_54 54
set element_core_55 55
set element_core_56 56
set element_core_57 57
set element_core_58 58
set element_core_59 59
set element_core_60 60
set element_core_61 61
set element_core_62 62
set element_core_63 63
set element_core_64 64
set element_core_65 65
set element_core_66 66
```

```
set element_core_67 67
set element_core_68 68
set element_core_69 69
set element_core_70 70
set element_core_71 71
set element_core_72 72
set element_core_73 73
set element_core_74 74
set element_core_75 75
set element_core_76 76
set element_core_77 77
set element_core_78 78
set element_core_79 79
set element_core_80 80
set element_core_81 81
set element_core_82 82
set element_core_83 83
set element_core_84 84
set element_core_85 85
set element_core_86 86
set element_core_87 87
set element_core_88 88
set element_core_89 89
set element_core_90 90
set element_core_91 91
set element_core_92 92
set element_core_93 93
set element_core_94 94
set element_core_95 95
set element_core_96 96
set element_core_97 97
set element_core_98 98
set element_core_99 99
set element_core_100 100
set element_core_101 101
set element_core_102 102
set element_core_103 103
set element_core_104 104
set element_core_105 105
set element_core_106 106
set element_core_107 107
set element_core_108 108
set element_core_109 109
set element_core_110 110
set element_core_111 111
set element_core_112 112
set element_core_113 113
set element_core_114 114
set element_core_115 115
set element_core_116 116
set element_core_117 117
set element_core_118 118
set element_core_119 119
set element_core_120 120
set element_core_121 121
set element_core_122 122
set element_core_123 123
set element_core_124 124
```

```
set element_core_125 125
set element_core_126 126
set element_core_127 127
set element_core_128 128
set element_core_129 129
element elasticBeamColumn $element_core_1 0 1 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_2 1 2 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_3 2 3 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_4 3 4 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_5 4 5 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_6 5 6 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_7 6 7 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_8 7 8 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_9 8 129 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_10 129 9 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_11 9 10 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_12 10 11 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_13 11 12 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_14 12 13 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_15 13 14 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_16 14 15 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_17 15 16 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_18 16 17 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_19 17 18 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_20 18 19 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_21 19 20 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_22 20 21 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_23 21 22 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_24 22 23 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_25 23 24 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_26 24 25 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_27 25 26 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_28 26 27 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_29 27 28 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_30 28 29 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_31 29 30 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_32 30 31 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_33 31 32 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_34 32 33 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_35 33 34 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_36 34 35 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_37 35 36 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_38 36 37 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_39 37 38 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_40 38 39 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_41 39 40 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_42 40 41 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_43 41 42 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_44 42 43 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_45 43 44 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_46 44 45 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_47 45 46 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_48 46 47 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_49 47 48 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_50 48 49 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_51 49 50 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_52 50 51 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_53 51 52 $SmallA $E $G $J 80 80 1
```

```
element elasticBeamColumn $element_core_54 52 53 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_55 53 54 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_56 54 55 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_57 55 56 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_58 56 57 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_59 57 58 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_60 58 59 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_61 59 60 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_62 60 61 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_63 61 62 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_64 62 63 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_65 63 64 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_66 64 65 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_67 65 66 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_68 66 67 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_69 67 68 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_70 68 69 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_71 69 70 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_72 70 71 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_73 71 72 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_74 72 73 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_75 73 74 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_76 74 75 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_77 75 76 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_78 76 77 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_79 77 78 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_80 78 79 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_81 79 80 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_82 80 81 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_83 81 82 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_84 82 83 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_85 83 84 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_86 84 85 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_87 85 86 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_88 86 87 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_89 87 88 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_90 88 89 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_91 89 90 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_92 90 91 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_93 91 92 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_94 92 93 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_95 93 94 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_96 94 95 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_97 95 96 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_98 96 97 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_99 97 98 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_100 98 99 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_101 99 100 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_102 100 101 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_103 101 102 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_104 102 103 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_105 103 104 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_106 104 105 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_107 105 106 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_108 106 107 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_109 107 108 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_110 108 109 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_111 109 110 $SmallA $E $G $J 80 80 1
```

```
element elasticBeamColumn $element_core_112 110 111 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_113 111 112 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_114 112 113 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_115 113 114 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_116 114 115 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_117 115 116 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_118 116 117 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_119 117 118 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_120 118 119 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_121 119 120 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_122 120 121 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_123 121 122 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_124 122 123 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_125 123 124 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_126 124 125 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_127 125 126 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_128 126 127 $SmallA $E $G $J 80 80 1
element elasticBeamColumn $element_core_129 127 128 $SmallA $E $G $J 80 80 1
set element_columnL1 50001
set element_columnR1 60001
set element_BRBL1 30001
set element_BRBR1 40001
set element_outriggerL1 10001
set element_outriggerR1 20001
element elasticBeamColumn $element_columnL1 1000 1001 $Column_A1 $E $G $J $SmallA $SmallA 1
element elasticBeamColumn $element_columnR1 2000 2001 $Column_A1 $E $G $J $SmallA $SmallA 1
element truss $element_BRBL1 1001 1002 $BRB_A1 $BRBmat1
element truss $element_BRBR1 2001 2002 $BRB_A1 $BRBmat1
element elasticBeamColumn $element_outriggerL1 1002 129 $SmallA $E $G $J 6.82667e-09 6.82667e-
09 1
element elasticBeamColumn $element_outriggerR1 2002 129 $SmallA $E $G $J 6.82667e-09 6.82667e-
09 1
set element_columnL2 50002
set element_columnR2 60002
set element_BRBL2 30002
set element_BRBR2 40002
set element_outriggerL2 10002
set element_outriggerR2 20002
element elasticBeamColumn $element_columnL2 1001 1003 $Column_A2 $E $G $J $SmallA $SmallA 1
element elasticBeamColumn $element_columnR2 2001 2003 $Column_A2 $E $G $J $SmallA $SmallA 1
element truss $element_BRBL2 1003 1004 $BRB_A2 $BRBmat2
element truss $element_BRBR2 2003 2004 $BRB_A2 $BRBmat2
element elasticBeamColumn $element_outriggerL2 1004 88 $SmallA $E $G $J 165.92 165.92 1
element elasticBeamColumn $element_outriggerR2 2004 88 $SmallA $E $G $J 165.92 165.92 1
#modal analysis
set fileperiod [open OpenseesPeriod.txt w]
set ind 0
set lambda [eigen $numModes]
set omega {}
set f {}
set T {}
set pi 3.141592654
foreach lam $lambda {
    lappend omega [expr sqrt($lam)]
    lappend f [expr sqrt($lam)/(2*$pi)]
    lappend T [expr (2*$pi)/sqrt($lam)]
}
puts "modal analysis done"
```

```
puts "vibration period (sec): "
foreach t $T {
puts "  $t"
puts $fileperiod "$t"
}
record
#dynamic analysis
#rayleigh $alphaM $betaK $betaKinit $betaKcomm
set w1 [expr 2 * $pi / [lindex $T 0]]
set w2 [expr 2 * $pi / [lindex $T 1]]
set alphaM [expr $dampingratio*2.0*$w1*$w2 / ($w1 + $w2)]
set betaK [expr 2.0*$dampingratio / ($w1 + $w2)]
rayleigh $alphaM $betaK 0.0 0.0
puts "rayleigh damping set"
puts "  damping ratio = $dampingratio"
puts "  alphaM = $alphaM"
puts "  betaK = $betaK"
#reading acceleration history
set accelSeries "Series -dt 0.01 -filePath OpenseesEQ.txt -factor 1.0"
pattern UniformExcitation 1  3 -accel $accelSeries;
recorder Node -file test8LateralDisp.txt -node 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108
109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 -dof 3 disp
recorder Node -file test8Rotation.txt -node 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81
82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109
110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 -dof 4 disp
recorder Node -file test8RoofDisp.txt -node 128 -dof 3 disp
recorder Node -file test8LateralVelocity.txt -node 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79
80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107
108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 -dof 3 vel
recorder Node -file test8LateralAcceleration.txt -node 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46
47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77
78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106
107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 -dof 3
accel
recorder Node -file test8RoofAcc.txt -node 128 -dof 3 accel
recorder Node -file test8BaseShear.txt -node 0 1000 2000 -dof 3 reaction
recorder Node -file test8BaseOTmoment.txt -node 0 -dof 4 reaction
recorder Element -file test8_BRBdeform1.txt -ele $element_BRBL1 $element_BRBR1 deformations
recorder Element -file test8_BRBforce1.txt -ele $element_BRBL1 $element_BRBR1 axialForce
recorder Element -file test8_Column1.txt -ele $element_columnL1 $element_columnR1 localForce
recorder Element -file test8_BRBdeform2.txt -ele $element_BRBL2 $element_BRBR2 deformations
recorder Element -file test8_BRBforce2.txt -ele $element_BRBL2 $element_BRBR2 axialForce
recorder Element -file test8_Column2.txt -ele $element_columnL2 $element_columnR2 localForce
source LibAnalysisDynamicParameters.tcl
set TmaxAnalysis 120
set DtAnalysis 0.01
set Tol 1e-3;
source DynamicAnalysis.tcl
record
puts "dynamic analysis done"
```

Appendix C

OpenSees tcl script of
32-story Single MBM model

```
# 96-story example model (MBM)
# vertical axis: y
# horizontal axis: z
# out-of-plane direction: x
# unit:kN m

proc pause {{message "Hit Enter to continue ==> "}} {
    puts -nonewline $message
    flush stdout
    gets stdin
}

wipe all;
model BasicBuilder -ndm 2 -ndf 3
set numModes 11

#source DisplayModel2D.tcl;
#source DisplayPlane.tcl;

set m 450
set E 200000000
set G 200000000
set J 200000000
set SmallA 0.000000001
set FloorBeamA 1e9
set dampingratio 0.02

set CoreColumnI 0.7872
set CoreColumnA 1.44
set CoreBeamI 0.09497
set CoreBeamA 0.176
set CoreBraceA 1.2
set CoreBraceI 0.000000000001
set PerimeterColA 0.3111
set PerimeterColI 0.04378
################################################
set OutriggerTrussA 18
set BRBPostYieldStiffnessRatio 0.01
################################################
set BRB_A2 0.048609398
set BRB_Fy2 258000

#set BRB_Fy2 1e9

#material
set ElasticMat 1
set BRBmat1 2
set BRBmat2 3
uniaxialMaterial Elastic $ElasticMat $E
uniaxialMaterial Steel01 $BRBmat2 $BRB_Fy2 $E $BRBPostYieldStiffnessRatio

# node
node 0 0 0
node 1 5 0
node 2 -5 0
node 3 21 0
node 4 -21 0
node 10 0 4
```

```
node 11 5 4
node 12 -5 4
node 13 21 4
node 14 -21 4
node 20 0 8
node 21 5 8
node 22 -5 8
node 23 21 8
node 24 -21 8
node 30 0 12
node 31 5 12
node 32 -5 12
node 33 21 12
node 34 -21 12
node 40 0 16
node 41 5 16
node 42 -5 16
node 43 21 16
node 44 -21 16
node 50 0 20
node 51 5 20
node 52 -5 20
node 53 21 20
node 54 -21 20
node 60 0 24
node 61 5 24
node 62 -5 24
node 63 21 24
node 64 -21 24
node 70 0 28
node 71 5 28
node 72 -5 28
node 73 21 28
node 74 -21 28
node 80 0 32
node 81 5 32
node 82 -5 32
node 83 21 32
node 84 -21 32
node 90 0 36
node 91 5 36
node 92 -5 36
node 93 21 36
node 94 -21 36
node 100 0 40
node 101 5 40
node 102 -5 40
node 103 21 40
node 104 -21 40
node 110 0 44
node 111 5 44
node 112 -5 44
node 113 21 44
node 114 -21 44
node 120 0 48
node 121 5 48
node 122 -5 48
node 123 21 48
```

```
node 124 -21 48
node 130 0 52
node 131 5 52
node 132 -5 52
node 133 21 52
node 134 -21 52
node 140 0 56
node 141 5 56
node 142 -5 56
node 143 21 56
node 144 -21 56
node 150 0 60
node 151 5 60
node 152 -5 60
node 153 21 60
node 154 -21 60
node 160 0 64
node 161 5 64
node 162 -5 64
node 163 21 64
node 164 -21 64
node 170 0 68
node 171 5 68
node 172 -5 68
node 173 21 68
node 174 -21 68
node 180 0 72
node 181 5 72
node 182 -5 72
node 183 21 72
node 184 -21 72
node 190 0 76
node 191 5 76
node 192 -5 76
node 193 21 76
node 194 -21 76
node 200 0 80
node 201 5 80
node 202 -5 80
node 203 21 80
node 204 -21 80
node 210 0 84
node 211 5 84
node 212 -5 84
node 213 21 84
node 214 -21 84
node 220 0 88
node 221 5 88
node 222 -5 88
node 223 21 88
node 224 -21 88
node 230 0 92
node 231 5 92
node 232 -5 92
#node 233 21 92
#node 234 -21 92
node 240 0 96
node 241 5 96
```

```
node 242 -5 96
node 243 21 96
node 244 -21 96
node 250 0 100
node 251 5 100
node 252 -5 100
node 253 21 100
node 254 -21 100
node 260 0 104
node 261 5 104
node 262 -5 104
node 263 21 104
node 264 -21 104
node 270 0 108
node 271 5 108
node 272 -5 108
node 273 21 108
node 274 -21 108
node 280 0 112
node 281 5 112
node 282 -5 112
node 283 21 112
node 284 -21 112
node 290 0 116
node 291 5 116
node 292 -5 116
node 293 21 116
node 294 -21 116
node 300 0 120
node 301 5 120
node 302 -5 120
node 303 21 120
node 304 -21 120
node 310 0 124
node 311 5 124
node 312 -5 124
node 313 21 124
node 314 -21 124
node 320 0 128
node 321 5 128
node 322 -5 128
node 323 21 128
node 324 -21 128

#fix
fix 0 1 1 1
fix 1 1 1 1
fix 2 1 1 1
fix 3 1 1 0
fix 4 1 1 0
fix 10 0 0 0
fix 11 0 0 0
fix 12 0 0 0
fix 13 0 0 0
fix 14 0 0 0
fix 20 0 0 0
fix 21 0 0 0
fix 22 0 0 0
```

```
fix 23 0 0 0
fix 24 0 0 0
fix 30 0 0 0
fix 31 0 0 0
fix 32 0 0 0
fix 33 0 0 0
fix 34 0 0 0
fix 40 0 0 0
fix 41 0 0 0
fix 42 0 0 0
fix 43 0 0 0
fix 44 0 0 0
fix 50 0 0 0
fix 51 0 0 0
fix 52 0 0 0
fix 53 0 0 0
fix 54 0 0 0
fix 60 0 0 0
fix 61 0 0 0
fix 62 0 0 0
fix 63 0 0 0
fix 64 0 0 0
fix 70 0 0 0
fix 71 0 0 0
fix 72 0 0 0
fix 73 0 0 0
fix 74 0 0 0
fix 80 0 0 0
fix 81 0 0 0
fix 82 0 0 0
fix 83 0 0 0
fix 84 0 0 0
fix 90 0 0 0
fix 91 0 0 0
fix 92 0 0 0
fix 93 0 0 0
fix 94 0 0 0
fix 100 0 0 0
fix 101 0 0 0
fix 102 0 0 0
fix 103 0 0 0
fix 104 0 0 0
fix 110 0 0 0
fix 111 0 0 0
fix 112 0 0 0
fix 113 0 0 0
fix 114 0 0 0
fix 120 0 0 0
fix 121 0 0 0
fix 122 0 0 0
fix 123 0 0 0
fix 124 0 0 0
fix 130 0 0 0
fix 131 0 0 0
fix 132 0 0 0
fix 133 0 0 0
fix 134 0 0 0
fix 140 0 0 0
```

```
fix 141 0 0 0
fix 142 0 0 0
fix 143 0 0 0
fix 144 0 0 0
fix 150 0 0 0
fix 151 0 0 0
fix 152 0 0 0
fix 153 0 0 0
fix 154 0 0 0
fix 160 0 0 0
fix 161 0 0 0
fix 162 0 0 0
fix 163 0 0 0
fix 164 0 0 0
fix 170 0 0 0
fix 171 0 0 0
fix 172 0 0 0
fix 173 0 0 0
fix 174 0 0 0
fix 180 0 0 0
fix 181 0 0 0
fix 182 0 0 0
fix 183 0 0 0
fix 184 0 0 0
fix 190 0 0 0
fix 191 0 0 0
fix 192 0 0 0
fix 193 0 0 0
fix 194 0 0 0
fix 200 0 0 0
fix 201 0 0 0
fix 202 0 0 0
fix 203 0 0 0
fix 204 0 0 0
fix 210 0 0 0
fix 211 0 0 0
fix 212 0 0 0
fix 213 0 0 0
fix 214 0 0 0
fix 220 0 0 0
fix 221 0 0 0
fix 222 0 0 0
fix 223 0 0 0
fix 224 0 0 0
fix 230 0 0 0
fix 231 0 0 0
fix 232 0 0 0
fix 233 0 0 0
fix 234 0 0 0
fix 240 0 0 0
fix 241 0 0 0
fix 242 0 0 0
fix 243 0 0 0
fix 244 0 0 0
fix 250 0 0 0
fix 251 0 0 0
fix 252 0 0 0
fix 253 0 0 0
```

```
fix 254 0 0 0
fix 260 0 0 0
fix 261 0 0 0
fix 262 0 0 0
fix 263 0 0 0
fix 264 0 0 0
fix 270 0 0 0
fix 271 0 0 0
fix 272 0 0 0
fix 273 0 0 0
fix 274 0 0 0
fix 280 0 0 0
fix 281 0 0 0
fix 282 0 0 0
fix 283 0 0 0
fix 284 0 0 0
fix 290 0 0 0
fix 291 0 0 0
fix 292 0 0 0
fix 293 0 0 0
fix 294 0 0 0
fix 300 0 0 0
fix 301 0 0 0
fix 302 0 0 0
fix 303 0 0 0
fix 304 0 0 0
fix 310 0 0 0
fix 311 0 0 0
fix 312 0 0 0
fix 313 0 0 0
fix 314 0 0 0
fix 320 0 0 0
fix 321 0 0 0
fix 322 0 0 0
fix 323 0 0 0
fix 324 0 0 0

#mass
mass 11 $m 0.000000001 0.000000001
mass 21 $m 0.000000001 0.000000001
mass 31 $m 0.000000001 0.000000001
mass 41 $m 0.000000001 0.000000001
mass 51 $m 0.000000001 0.000000001
mass 61 $m 0.000000001 0.000000001
mass 71 $m 0.000000001 0.000000001
mass 81 $m 0.000000001 0.000000001
mass 91 $m 0.000000001 0.000000001
mass 101 $m 0.000000001 0.000000001
mass 111 $m 0.000000001 0.000000001
mass 121 $m 0.000000001 0.000000001
mass 131 $m 0.000000001 0.000000001
mass 141 $m 0.000000001 0.000000001
mass 151 $m 0.000000001 0.000000001
mass 161 $m 0.000000001 0.000000001
mass 171 $m 0.000000001 0.000000001
mass 181 $m 0.000000001 0.000000001
mass 191 $m 0.000000001 0.000000001
mass 201 $m 0.000000001 0.000000001
```

```
mass 211 $m 0.000000001 0.000000001
mass 221 $m 0.000000001 0.000000001
mass 231 $m 0.000000001 0.000000001
mass 241 $m 0.000000001 0.000000001
mass 251 $m 0.000000001 0.000000001
mass 261 $m 0.000000001 0.000000001
mass 271 $m 0.000000001 0.000000001
mass 281 $m 0.000000001 0.000000001
mass 291 $m 0.000000001 0.000000001
mass 301 $m 0.000000001 0.000000001
mass 311 $m 0.000000001 0.000000001
mass 321 $m 0.000000001 0.000000001


mass 12 $m 0.000000001 0.000000001
mass 22 $m 0.000000001 0.000000001
mass 32 $m 0.000000001 0.000000001
mass 42 $m 0.000000001 0.000000001
mass 52 $m 0.000000001 0.000000001
mass 62 $m 0.000000001 0.000000001
mass 72 $m 0.000000001 0.000000001
mass 82 $m 0.000000001 0.000000001
mass 92 $m 0.000000001 0.000000001
mass 102 $m 0.000000001 0.000000001
mass 112 $m 0.000000001 0.000000001
mass 122 $m 0.000000001 0.000000001
mass 132 $m 0.000000001 0.000000001
mass 142 $m 0.000000001 0.000000001
mass 152 $m 0.000000001 0.000000001
mass 162 $m 0.000000001 0.000000001
mass 172 $m 0.000000001 0.000000001
mass 182 $m 0.000000001 0.000000001
mass 192 $m 0.000000001 0.000000001
mass 202 $m 0.000000001 0.000000001
mass 212 $m 0.000000001 0.000000001
mass 222 $m 0.000000001 0.000000001
mass 232 $m 0.000000001 0.000000001
mass 242 $m 0.000000001 0.000000001
mass 252 $m 0.000000001 0.000000001
mass 262 $m 0.000000001 0.000000001
mass 272 $m 0.000000001 0.000000001
mass 282 $m 0.000000001 0.000000001
mass 292 $m 0.000000001 0.000000001
mass 302 $m 0.000000001 0.000000001
mass 312 $m 0.000000001 0.000000001
mass 322 $m 0.000000001 0.000000001

geomTransf Linear 1

#element
element elasticBeamColumn 12 1 11 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 22 11 21 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 32 21 31 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 42 31 41 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 52 41 51 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 62 51 61 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 72 61 71 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 82 71 81 $CoreColumnA $E $CoreColumnI 1
```

```
element elasticBeamColumn 92 81 91 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 102 91 101 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 112 101 111 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 122 111 121 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 132 121 131 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 142 131 141 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 152 141 151 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 162 151 161 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 172 161 171 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 182 171 181 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 192 181 191 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 202 191 201 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 212 201 211 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 222 211 221 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 232 221 231 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 242 231 241 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 252 241 251 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 262 251 261 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 272 261 271 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 282 271 281 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 292 281 291 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 302 291 301 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 312 301 311 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 322 311 321 $CoreColumnA $E $CoreColumnI 1

element elasticBeamColumn 13 2 12 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 23 12 22 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 33 22 32 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 43 32 42 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 53 42 52 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 63 52 62 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 73 62 72 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 83 72 82 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 93 82 92 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 103 92 102 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 113 102 112 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 123 112 122 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 133 122 132 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 143 132 142 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 153 142 152 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 163 152 162 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 173 162 172 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 183 172 182 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 193 182 192 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 203 192 202 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 213 202 212 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 223 212 222 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 233 222 232 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 243 232 242 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 253 242 252 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 263 252 262 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 273 262 272 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 283 272 282 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 293 282 292 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 303 292 302 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 313 302 312 $CoreColumnA $E $CoreColumnI 1
element elasticBeamColumn 323 312 322 $CoreColumnA $E $CoreColumnI 1
```

```
element elasticBeamColumn 18 11 10 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 28 21 20 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 38 31 30 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 48 41 40 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 58 51 50 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 68 61 60 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 78 71 70 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 88 81 80 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 98 91 90 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 108 101 100 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 118 111 110 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 128 121 120 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 138 131 130 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 148 141 140 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 158 151 150 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 168 161 160 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 178 171 170 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 188 181 180 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 198 191 190 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 208 201 200 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 218 211 210 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 228 221 220 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 238 231 230 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 248 241 240 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 258 251 250 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 268 261 260 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 278 271 270 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 288 281 280 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 298 291 290 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 308 301 300 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 318 311 310 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 328 321 320 $CoreBeamA $E $CoreBeamI 1

element elasticBeamColumn 19 10 12 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 29 20 22 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 39 30 32 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 49 40 42 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 59 50 52 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 69 60 62 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 79 70 72 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 89 80 82 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 99 90 92 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 109 100 102 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 119 110 112 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 129 120 122 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 139 130 132 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 149 140 142 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 159 150 152 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 169 160 162 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 179 170 172 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 189 180 182 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 199 190 192 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 209 200 202 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 219 210 212 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 229 220 222 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 239 230 232 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 249 240 242 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 259 250 252 $CoreBeamA $E $CoreBeamI 1
```

```
element elasticBeamColumn 269 260 262 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 279 270 272 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 289 280 282 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 299 290 292 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 309 300 302 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 319 310 312 $CoreBeamA $E $CoreBeamI 1
element elasticBeamColumn 329 320 322 $CoreBeamA $E $CoreBeamI 1

element truss 10 1 10 $CoreBraceA $ElasticMat
element truss 20 10 21 $CoreBraceA $ElasticMat
element truss 30 21 30 $CoreBraceA $ElasticMat
element truss 40 30 41 $CoreBraceA $ElasticMat
element truss 50 41 50 $CoreBraceA $ElasticMat
element truss 60 50 61 $CoreBraceA $ElasticMat
element truss 70 61 70 $CoreBraceA $ElasticMat
element truss 80 70 81 $CoreBraceA $ElasticMat
element truss 90 81 90 $CoreBraceA $ElasticMat
element truss 100 90 101 $CoreBraceA $ElasticMat
element truss 110 101 110 $CoreBraceA $ElasticMat
element truss 120 110 121 $CoreBraceA $ElasticMat
element truss 130 121 130 $CoreBraceA $ElasticMat
element truss 140 130 141 $CoreBraceA $ElasticMat
element truss 150 141 150 $CoreBraceA $ElasticMat
element truss 160 150 161 $CoreBraceA $ElasticMat
element truss 170 161 170 $CoreBraceA $ElasticMat
element truss 180 170 181 $CoreBraceA $ElasticMat
element truss 190 181 190 $CoreBraceA $ElasticMat
element truss 200 190 201 $CoreBraceA $ElasticMat
element truss 210 201 210 $CoreBraceA $ElasticMat
element truss 220 210 221 $CoreBraceA $ElasticMat
element truss 230 221 230 $CoreBraceA $ElasticMat
element truss 240 230 241 $CoreBraceA $ElasticMat
element truss 250 241 250 $CoreBraceA $ElasticMat
element truss 260 250 261 $CoreBraceA $ElasticMat
element truss 270 261 270 $CoreBraceA $ElasticMat
element truss 280 270 281 $CoreBraceA $ElasticMat
element truss 290 281 290 $CoreBraceA $ElasticMat
element truss 300 290 301 $CoreBraceA $ElasticMat
element truss 310 301 310 $CoreBraceA $ElasticMat
element truss 320 310 321 $CoreBraceA $ElasticMat

element truss 11 2 10 $CoreBraceA $ElasticMat
element truss 21 10 22 $CoreBraceA $ElasticMat
element truss 31 22 30 $CoreBraceA $ElasticMat
element truss 41 30 42 $CoreBraceA $ElasticMat
element truss 51 42 50 $CoreBraceA $ElasticMat
element truss 61 50 62 $CoreBraceA $ElasticMat
element truss 71 62 70 $CoreBraceA $ElasticMat
element truss 81 70 82 $CoreBraceA $ElasticMat
element truss 91 82 90 $CoreBraceA $ElasticMat
element truss 101 90 102 $CoreBraceA $ElasticMat
element truss 111 102 110 $CoreBraceA $ElasticMat
element truss 121 110 122 $CoreBraceA $ElasticMat
element truss 131 122 130 $CoreBraceA $ElasticMat
element truss 141 130 142 $CoreBraceA $ElasticMat
element truss 151 142 150 $CoreBraceA $ElasticMat
element truss 161 150 162 $CoreBraceA $ElasticMat
element truss 171 162 170 $CoreBraceA $ElasticMat
```

```
element truss 181 170 182 $CoreBraceA $ElasticMat
element truss 191 182 190 $CoreBraceA $ElasticMat
element truss 201 190 202 $CoreBraceA $ElasticMat
element truss 211 202 210 $CoreBraceA $ElasticMat
element truss 221 210 222 $CoreBraceA $ElasticMat
element truss 231 222 230 $CoreBraceA $ElasticMat
element truss 241 230 242 $CoreBraceA $ElasticMat
element truss 251 242 250 $CoreBraceA $ElasticMat
element truss 261 250 262 $CoreBraceA $ElasticMat
element truss 271 262 270 $CoreBraceA $ElasticMat
element truss 281 270 282 $CoreBraceA $ElasticMat
element truss 291 282 290 $CoreBraceA $ElasticMat
element truss 301 290 302 $CoreBraceA $ElasticMat
element truss 311 302 310 $CoreBraceA $ElasticMat
element truss 321 310 322 $CoreBraceA $ElasticMat

element truss 16 13 11 $FloorBeamA $ElasticMat
element truss 26 23 21 $FloorBeamA $ElasticMat
element truss 36 33 31 $FloorBeamA $ElasticMat
element truss 46 43 41 $FloorBeamA $ElasticMat
element truss 56 53 51 $FloorBeamA $ElasticMat
element truss 66 63 61 $FloorBeamA $ElasticMat
element truss 76 73 71 $FloorBeamA $ElasticMat
element truss 86 83 81 $FloorBeamA $ElasticMat
element truss 96 93 91 $FloorBeamA $ElasticMat
element truss 106 103 101 $FloorBeamA $ElasticMat
element truss 116 113 111 $FloorBeamA $ElasticMat
element truss 126 123 121 $FloorBeamA $ElasticMat
element truss 136 133 131 $FloorBeamA $ElasticMat
element truss 146 143 141 $FloorBeamA $ElasticMat
element truss 156 153 151 $FloorBeamA $ElasticMat
element truss 166 163 161 $FloorBeamA $ElasticMat
element truss 176 173 171 $FloorBeamA $ElasticMat
element truss 186 183 181 $FloorBeamA $ElasticMat
element truss 196 193 191 $FloorBeamA $ElasticMat
element truss 206 203 201 $FloorBeamA $ElasticMat
element truss 216 213 211 $FloorBeamA $ElasticMat
#element truss 226 223 221 $FloorBeamA $ElasticMat
#element truss 236 233 231 $FloorBeamA $ElasticMat
element truss 246 243 241 $FloorBeamA $ElasticMat
element truss 256 253 251 $FloorBeamA $ElasticMat
element truss 266 263 261 $FloorBeamA $ElasticMat
element truss 276 273 271 $FloorBeamA $ElasticMat
element truss 286 283 281 $FloorBeamA $ElasticMat
element truss 296 293 291 $FloorBeamA $ElasticMat
element truss 306 303 301 $FloorBeamA $ElasticMat
element truss 316 313 311 $FloorBeamA $ElasticMat
element truss 326 323 321 $FloorBeamA $ElasticMat

element truss 17 12 14 $FloorBeamA $ElasticMat
element truss 27 22 24 $FloorBeamA $ElasticMat
element truss 37 32 34 $FloorBeamA $ElasticMat
element truss 47 42 44 $FloorBeamA $ElasticMat
element truss 57 52 54 $FloorBeamA $ElasticMat
element truss 67 62 64 $FloorBeamA $ElasticMat
element truss 77 72 74 $FloorBeamA $ElasticMat
element truss 87 82 84 $FloorBeamA $ElasticMat
element truss 97 92 94 $FloorBeamA $ElasticMat
```

```
element truss 107 102 104 $FloorBeamA $ElasticMat
element truss 117 112 114 $FloorBeamA $ElasticMat
element truss 127 122 124 $FloorBeamA $ElasticMat
element truss 137 132 134 $FloorBeamA $ElasticMat
element truss 147 142 144 $FloorBeamA $ElasticMat
element truss 157 152 154 $FloorBeamA $ElasticMat
element truss 167 162 164 $FloorBeamA $ElasticMat
element truss 177 172 174 $FloorBeamA $ElasticMat
element truss 187 182 184 $FloorBeamA $ElasticMat
element truss 197 192 194 $FloorBeamA $ElasticMat
element truss 207 202 204 $FloorBeamA $ElasticMat
element truss 217 212 214 $FloorBeamA $ElasticMat
#element truss 227 222 224 $FloorBeamA $ElasticMat
#element truss 237 232 234 $FloorBeamA $ElasticMat
element truss 247 242 244 $FloorBeamA $ElasticMat
element truss 257 252 254 $FloorBeamA $ElasticMat
element truss 267 262 264 $FloorBeamA $ElasticMat
element truss 277 272 274 $FloorBeamA $ElasticMat
element truss 287 282 284 $FloorBeamA $ElasticMat
element truss 297 292 294 $FloorBeamA $ElasticMat
element truss 307 302 304 $FloorBeamA $ElasticMat
element truss 317 312 314 $FloorBeamA $ElasticMat
element truss 327 322 324 $FloorBeamA $ElasticMat

element elasticBeamColumn 14 3 13 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 24 13 23 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 34 23 33 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 44 33 43 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 54 43 53 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 64 53 63 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 74 63 73 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 84 73 83 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 94 83 93 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 104 93 103 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 114 103 113 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 124 113 123 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 134 123 133 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 144 133 143 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 154 143 153 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 164 153 163 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 174 163 173 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 184 173 183 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 194 183 193 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 204 193 203 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 214 203 213 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 224 213 223 $PerimeterColA $E $PerimeterColI 1

#element elasticBeamColumn 234 223 233 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 244 223 243 $PerimeterColA $E $PerimeterColI 1

element elasticBeamColumn 254 243 253 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 264 253 263 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 274 263 273 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 284 273 283 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 294 283 293 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 304 293 303 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 314 303 313 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 324 313 323 $PerimeterColA $E $PerimeterColI 1
```

```
element elasticBeamColumn 15 4 14 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 25 14 24 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 35 24 34 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 45 34 44 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 55 44 54 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 65 54 64 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 75 64 74 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 85 74 84 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 95 84 94 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 105 94 104 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 115 104 114 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 125 114 124 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 135 124 134 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 145 134 144 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 155 144 154 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 165 154 164 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 175 164 174 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 185 174 184 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 195 184 194 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 205 194 204 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 215 204 214 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 225 214 224 $PerimeterColA $E $PerimeterColI 1

#element elasticBeamColumn 235 224 234 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 245 224 244 $PerimeterColA $E $PerimeterColI 1

element elasticBeamColumn 255 244 254 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 265 254 264 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 275 264 274 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 285 274 284 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 295 284 294 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 305 294 304 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 315 304 314 $PerimeterColA $E $PerimeterColI 1
element elasticBeamColumn 325 314 324 $PerimeterColA $E $PerimeterColI 1

#outrigger truss node
node 30000 21 92
node 30001 17 92
node 30002 13 92
node 30003 9 92
node 30004 17 88
node 30005 13 88
node 30006 9 88
node 40000 -21 92
node 40001 -17 92
node 40002 -13 92
node 40003 -9 92
node 40004 -17 88
node 40005 -13 88
node 40006 -9 88

#outrigger truss node fix
fix 30000 0 0 1
fix 30001 0 0 1
fix 30002 0 0 1
fix 30003 0 0 1
fix 30004 0 0 1
```

```
fix 30005 0 0 1
fix 30006 0 0 1
fix 40000 0 0 1
fix 40001 0 0 1
fix 40002 0 0 1
fix 40003 0 0 1
fix 40004 0 0 1
fix 40005 0 0 1
fix 40006 0 0 1

#outrigger truss member
element truss 30001 30000 30001 $OutriggerTrussA $ElasticMat
element truss 30002 30001 30002 $OutriggerTrussA $ElasticMat
element truss 30003 30002 30003 $OutriggerTrussA $ElasticMat
element truss 30004 30003 231 $OutriggerTrussA $ElasticMat
element truss 30005 30004 30005 $OutriggerTrussA $ElasticMat
element truss 30006 30005 30006 $OutriggerTrussA $ElasticMat
element truss 30007 30006 221 $OutriggerTrussA $ElasticMat
element truss 30008 30000 30004 $OutriggerTrussA $ElasticMat
element truss 30009 30004 30001 $OutriggerTrussA $ElasticMat
element truss 30010 30001 30005 $OutriggerTrussA $ElasticMat
element truss 30011 30005 30002 $OutriggerTrussA $ElasticMat
element truss 30012 30002 30006 $OutriggerTrussA $ElasticMat
element truss 30013 30006 30003 $OutriggerTrussA $ElasticMat
element truss 30014 30003 221 $OutriggerTrussA $ElasticMat
element truss 40001 40000 40001 $OutriggerTrussA $ElasticMat
element truss 40002 40001 40002 $OutriggerTrussA $ElasticMat
element truss 40003 40002 40003 $OutriggerTrussA $ElasticMat
element truss 40004 40003 232 $OutriggerTrussA $ElasticMat
element truss 40005 40004 40005 $OutriggerTrussA $ElasticMat
element truss 40006 40005 40006 $OutriggerTrussA $ElasticMat
element truss 40007 40006 222 $OutriggerTrussA $ElasticMat
element truss 40008 40000 40004 $OutriggerTrussA $ElasticMat
element truss 40009 40004 40001 $OutriggerTrussA $ElasticMat
element truss 40010 40001 40005 $OutriggerTrussA $ElasticMat
element truss 40011 40005 40002 $OutriggerTrussA $ElasticMat
element truss 40012 40002 40006 $OutriggerTrussA $ElasticMat
element truss 40013 40006 40003 $OutriggerTrussA $ElasticMat
element truss 40014 40003 222 $OutriggerTrussA $ElasticMat

#BRB truss member
element truss 30000 30000 223 $BRB_A2 $BRBmat2
element truss 40000 40000 224 $BRB_A2 $BRBmat2

# display the model with the node numbers
#DisplayModel2D NodeNumbers

#pause

#recorder
recorder Node -file ModeShape1.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
1"

recorder Node -file ModeShape2.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
```

```
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
2"

recorder Node -file ModeShape3.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
3"

recorder Node -file ModeShape4.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
4"

recorder Node -file ModeShape5.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
5"

recorder Node -file ModeShape6.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
6"

recorder Node -file ModeShape7.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
7"

recorder Node -file ModeShape8.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
8"

recorder Node -file ModeShape9.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
9"

recorder Node -file ModeShape10.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
10"

recorder Node -file ModeShape11.txt -node 1 11 21 31 41 51 61 71 81 91 \
101 111 121 131 141 151 161 171 \
181 191 201 211 221 231 241 251 261 271 281 291 301 311 321 -dof 1 "eigen
11"

#modal analysis
set fileperiod [open MBM_core_OpenseesPeriod.txt w]
set ind 0
set lambda [eigen $numModes]
set omega {}
set f {}
set T {}
set pi 3.141592654
foreach lam $lambda {
    lappend omega [expr sqrt($lam)]
```

```
    lappend f [expr sqrt($lam)/(2*$pi)]
    lappend T [expr (2*$pi)/sqrt($lam)]
}
puts "modal analysis done"
puts "vibration period (sec): "
foreach t $T {
puts "  $t"
puts $fileperiod "$t"
}

record

puts "modal analysis done"

#dynamic analysis
set w1 [expr 2 * $pi / [lindex $T 0]]
set w2 [expr 2 * $pi / [lindex $T 1]]
set alphaM [expr $dampingratio*2.0*$w1*$w2 / ($w1 + $w2)]
set betaK [expr 2.0*$dampingratio / ($w1 + $w2)]
rayleigh $alphaM $betaK 0.0 0.0
puts ""
puts "rayleigh damping set"
puts "  damping ratio = $dampingratio"
puts "  alphaM = $alphaM"
puts "  betaK = $betaK"
#
#set ViewScale 5;
#DisplayModel2D DeformedShape $ViewScale ;
#
recorder Node -file test8LateralDisp.txt -node 10 20 30 40 50 60 70 80 90
100 110 120 130 140 150 160 170 180 190 200 210 220 230 240 250 260 270 280
290 300 310 320 -dof 1 disp
recorder Node -file test8LateralVelo.txt -node 10 20 30 40 50 60 70 80 90
100 110 120 130 140 150 160 170 180 190 200 210 220 230 240 250 260 270 280
290 300 310 320 -dof 1 vel
recorder Node -file test8RoofDisp.txt -node 320 -dof 1 disp
recorder Node -file test8RoofAcc.txt -node 320 -dof 1 accel
recorder Node -file test8BaseShear.txt -node 1 2 3 4 -dof 1 reaction
recorder Node -file test8BaseAxialForce.txt -node 1 2 3 4 -dof 2 reaction
recorder Node -file test8BaseOTmoment.txt -node 1 2 3 4 -dof 3 reaction
recorder Element -file test8_Column1.txt -ele 14 15 localForce
recorder Element -file test8_Column2.txt -ele 244 245 localForce
recorder Element -file test8_BRBdeform2.txt -ele 30000 40000 deformations
recorder Element -file test8_BRBforce2.txt -ele 30000 40000 axialForce

#reading acceleration history
#8. EQ_BCJL2.txt   1.19 0.01
set accelSeries "Series -dt 0.01 -filePath EQ_BCJL2.txt -factor 1.14371"

pattern UniformExcitation 1  1 -accel $accelSeries;

source LibAnalysisDynamicParameters.tcl
set TmaxAnalysis 120
set DtAnalysis 0.01
set Tol 1e-3;
source DynamicAnalysis.tcl
record
puts ""
```

```
puts "dynamic analysis done"

#pause;
```

DYNAMICANALYSIS.TCL

```
set Nsteps [expr int($TmaxAnalysis/$DtAnalysis)];
set ok [analyze $Nsteps $DtAnalysis];
returns ok=0 if analysis was successful

if {$ok != 0} {
    # change some analysis parameters to achieve convergence
    # performance is slower inside this loop
    #    Time-controlled analysis
    set ok 0;
    set controlTime [getTime];
    while {$controlTime < $TmaxAnalysis && $ok == 0} {
        set controlTime [getTime]
        set ok [analyze 1 $DtAnalysis]
        if {$ok != 0} {
            puts "Trying Newton with Initial Tangent .."
            test NormDispIncr   $Tol 1000  0
            algorithm Newton -initial
            set ok [analyze 1 $DtAnalysis]
            test $testTypeDynamic $TolDynamic $maxNumIterDynamic  0 0
            algorithm $algorithmTypeDynamic
        }
        if {$ok != 0} {
            puts "Trying Broyden .."
            algorithm Broyden 8
            set ok [analyze 1 $DtAnalysis]
            algorithm $algorithmTypeDynamic
        }
        if {$ok != 0} {
            puts "Trying NewtonWithLineSearch .."
            algorithm NewtonLineSearch .8
            set ok [analyze 1 $DtAnalysis]
            algorithm $algorithmTypeDynamic
        }
    }
};      # end if ok !0

puts "Ground Motion Done. End Time: [getTime]"
set file [open OpenseesDynamicResult.txt w]

if {$ok != 0} {
    puts "Dynamic analysis did not converge"
    puts $file "fail"
}
if {$ok == 0} {
    puts $file "success"
}
close $file
```