Artificial Intelligence Approach for Seismic Control of Structures

DISSERTATION

Zur Erlangung des akademischen Grades eines
Doktor-Ingenieur
an der Fakultät Bauingenieurwesen
der Bauhaus-Universität Weimar

vorgelegt von
Hamid Radmard Rahmani
Weimar, June 2019

Mentor:
Prof. Dr.-Ing. habil. Carsten Könke, Bauhaus-Universität Weimar

Gutachter:
Prof. Dr.-Ing. Christian Koch, Bauhaus-Universität Weimar
Dr. M.A. Marco Wiering, University of Groningen, Netherlands

Disputation am 24. Februar 2020

Dedicated to

my daughter,
Mantreh

and

the soul of my father,
Ahmed

Abstract

In the first part of this research, the utilization of tuned mass dampers in the vibration control of tall buildings during earthquake excitations is studied. The main issues such as optimizing the parameters of the dampers and studying the effects of frequency content of the target earthquakes are addressed.

The non-dominated sorting genetic algorithm method is improved by upgrading generic operators, and is utilized to develop a framework for determining the optimum placement and parameters of dampers in tall buildings. A case study is presented in which the optimal placement and properties of dampers are determined for a model of a tall building under different earthquake excitations through computer simulations.

In the second part, a novel framework for the brain learning-based intelligent seismic control of smart structures is developed. In this approach, a deep neural network learns how to improve structural responses during earthquake excitations using feedback control.

Reinforcement learning method is improved and utilized to develop a framework for training the deep neural network as an intelligent controller. The efficiency of the developed framework is examined through two case studies including a single-degree-of-freedom system and a high-rise building under different earthquake excitation records.

The results show that the controller gradually develops an optimum control policy to reduce the vibrations of a structure under an earthquake excitation through a cyclical process of actions and observations.

It is shown that the controller efficiently improves the structural responses under new earthquake excitations for which it was not trained. Moreover, it is shown that the controller has a stable performance under uncertainties.

Contents

1 Introduction **13**

1.1 Introduction . . . 13

1.2 Thesis Outline . . . 16

2 Literature Review **18**

2.1 Introduction . . . 18

2.2 Seismic Control of Tall Buildings . . . 18

2.2.1 Tuned Mass Dampers . . . 19

2.2.2 Multiple Tuned Mass Dampers . . . 20

2.3 Active Control of Structures . . . 21

2.3.1 Active Tuned Mass Dampers . . . 22

2.3.2 Active Tendon Systems . . . 23

2.4 Semi-Active Control Systems . . . 23

2.4.1 Magnetorheological Fluid Dampers . . . 23

2.4.2 Semi-active Stiffness Dampers . . . 24

2.5 Advances in control Algorithms . . . 24

2.5.1 Introduction . . . 24

2.6 Research Objectives . . . 32

3 Seismic Control of Tall Buildings using Tuned Mass Dampers **34**

3.1 Introduction . . . 34

3.2 Fast and Elitist Multi-Objective Genetic Algorithm . . . 34

3.2.1 Introduction . . . 34

3.2.2 Elements of Genetic Algorithm . . . 34

3.2.3 Fast Elitist Multi-Objective Genetic Algorithm . . . 35

3.2.4 Repair Module . . . 37

3.2.5 Selection . . . 37

3.2.6 Optimization Variables . . . 38

3.2.7 Encoding . . . 39

3.2.8 Genetic Operators . . . 40

3.2.9 Fitness Function . . . 42

3.3 Framework . . . 43
3.3.1 Introduction . . . 43
3.3.2 Basic Algorithm . . . 43
3.4 Case Study . . . 44
3.4.1 Introduction . . . 44
3.4.2 Mathematical Model of Building . . . 45
3.4.3 Ground Motion Selection . . . 46
3.4.4 TMD Parameters . . . 49
3.4.5 Sensitivity Analysis of GA Parameters . . . 50
3.5 Results . . . 51
3.6 Discussion . . . 60
3.7 Conclusion . . . 65

4 Reinforcement Learning **67**
4.1 Introduction . . . 67
4.2 Machine Learning Methods . . . 67
4.3 Elements of Reinforcement Learning . . . 68
4.3.1 Markov Decision Process . . . 69
4.3.2 Markov Reward Process . . . 69
4.3.3 Bellman Expectation Equation . . . 70
4.3.4 Bellman Optimality Equation . . . 71
4.4 Dynamic Programming . . . 71
4.4.1 Policy Evaluation by State Iteration . . . 71
4.4.2 Policy Improvement . . . 72
4.4.3 Policy Iteration . . . 72
4.4.4 Summary . . . 72
4.5 Model Free Methods . . . 72
4.5.1 Monte-Carlo Policy Evaluation . . . 72
4.5.2 Temporal Difference Learning . . . 73
4.5.3 Q-Learning . . . 73
4.6 Exploration and Exploitation . . . 74
4.7 Summary . . . 75

5 Intelligent Controller **76**
5.1 Introduction . . . 76
5.2 MDP in Structural Control . . . 76
5.2.1 State . . . 76
5.2.2 Action . . . 77
5.2.3 Rewards . . . 77
5.2.3.1 Reward Function . . . 78

5.2.4 Learning Algorithm . . . 78
5.2.5 Enhanced Mini-Batch Learning . . . 79
5.2.5.1 Key-states selector function . . . 79
5.2.5.2 Reflexive $\gamma-$ Function . . . 79
5.3 Framework . . . 80
5.3.1 Analytical Model . . . 81
5.3.2 Neural Networks . . . 81
5.3.3 Learning Rule . . . 83
5.3.4 Testing Module . . . 83
5.3.5 Simulation Environment . . . 83

6 Case Studies **84**
6.1 Introduction . . . 84
6.2 Case study I - Single Degree of Freedom System . . . 84
6.2.1 Introduction . . . 84
6.2.2 System Properties . . . 84
6.2.3 Dynamics of Environment . . . 85
6.2.4 States . . . 86
6.2.5 Actions . . . 87
6.2.6 Reward Function . . . 87
6.2.7 Earthquake Excitations . . . 88
6.2.8 Learning by Mini-Batch Learning Method . . . 88
6.2.9 Improved Mini-Batch Learning . . . 89
6.2.10 Testing Phase . . . 92
6.2.11 Environmental Uncertainties . . . 95
6.3 Case Study 2 - High-Rise Building . . . 96
6.3.1 Introduction . . . 96
6.3.2 Benchmark Building . . . 96
6.3.3 Environment Dynamics . . . 98
6.3.4 Earthquake Excitations . . . 99
6.3.5 Actions . . . 99
6.3.6 Reward Function . . . 99
6.3.7 Learning by Improved Q-Learning . . . 100
6.3.8 Training Phase . . . 100
6.3.9 Testing Phase . . . 102
6.3.10 Environmental Uncertainties . . . 105

7 Conclusion **107**
7.1 Summary of Achievements . . . 107
7.2 Outlook . . . 108

Bibliography **109**

A Developed Algorithms **125**

A.1 Adapted Crossover Algorithm 125

A.2 Adapted Mutation Algorithm 126

A.3 Framework Basic Algorithm 127

A.4 Key States Algorithm 127

List of Figures

1.1 Collapse of tall buildings during the Taiwan earthquake in 2018 [1] . . 13

2.1 Tuned mass damper 19
2.2 Taipei 101's tuned mass damper [1] 20
2.3 Tallest completed buildings with dampers [2] 21
2.4 Schematic diagram of an ATMD [3] 22
2.5 Schematic diagram of an active tendon controller [4] 23

3.1 Non-dominated sorting of solutions [5] 36
3.2 Illustration of crowding distance [5] 36
3.3 Repair of an infeasible solution [5] 37
3.4 Roulette wheel selection 38
3.5 Building equipped by TMDs 39
3.6 Binary coding the TMD's properties 40
3.7 A schematic chromosome representation 41
3.8 Crossover types 42
3.9 Five mode shapes of the 76-story building 44
3.10 Earthquake acceleration records 46
3.10 Earthquake acceleration records obtained from the Pacific Earthquake Engineering Research (PEER) Center [6] 47
3.11 Response spectrum for a. unscaled and b. scaled earthquake records 49
3.12 Trend of objective J_1 during the generations 51
3.13 Uncontrolled/controlled responses- Bam earthquake 53
3.14 Uncontrolled/controlled responses- Elcentro earthquake 54
3.15 Uncontrolled/controlled responses- Kobe earthquake 55
3.16 Uncontrolled/controlled responses- Landers earthquake 56
3.17 Uncontrolled/controlled responses- Manjil earthquake 57
3.18 Uncontrolled/controlled responses- Northridge earthquake 58
3.19 Uncontrolled/controlled responses- SanFernando earthquake 59
3.20 Three stories with maximum modal displacements in first five natural modes 60
3.21 Fourier transform of excitations 61

3.21 Fourier transform of excitations 62
3.21 Fourier transform of excitations 63
3.21 Fourier transform of excitations 64

5.1 Process of creating reflexive γ-function 80
5.2 The developed framework for training an intelligent controller 82

6.1 The simulation environment 85
6.2 Schematic of single degree of freedom system 85
6.3 Landers earthquake's acceleration record 88
6.4 Average reward values during the learning phase 89
6.5 Controlled and uncontrolled responses of the frame to the Landers earthquake when the intelligent controller was trained using the original learning algorithm. 90
6.6 Improvement of the average rewards during the learning process using the enhanced mini-batch learning method. 91
6.7 Controlled and uncontrolled responses of the frame when the controller has been trained by the enhanced method. 91
6.8 Four earthquake acceleration records which were considered to test the performance of the controller 93
6.9 Uncontrolled and controlled responses of the frame to the test earthquake excitations which are new to the intelligent controller 94
6.10 Figure 1- N-S moment resisting frame of twenty story benchmark building[7] 97
6.11 Three mode shapes of the 20-story benchmark building 98
6.12 Improvement of the average rewards during the learning process using the improved method. 100
6.13 Controlled and uncontrolled displacement responses under Landers earthquake 101
6.14 Controlled and uncontrolled Story drifts under Landers earthquake . 101
6.15 Uncontrolled and controlled displacement responses 103
6.16 Uncontrolled and controlled Drifts 104
6.17 Typical transfer functions for the high-rise building under different earthquake excitations 105

List of Tables

3.2 Original earthquakes' specifications (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (g)) 48

3.3 Scaled earthquakes' specifications (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (g)) 48

3.1 Parameters of the design response spectrum 48

3.4 Parameter variation domain for TMDs 49

3.5 Crossover and Mutation variations 50

3.6 NSGA of Crossover and Mutation variations 50

3.7 NSGA II algorithm initial parameters 51

5.1 Magnitude of the agent's action 77

6.1 Control force range and load-steps 87

6.2 Utilized learning parameters 89

6.3 Controlled and uncontrolled responses of the frame to the Landers earthquake excitations when the controller has been trained by original and enhanced methods (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (m/s^2)). 92

6.4 Responses to the earthquake excitations (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (m/s^2)). 95

6.5 Performance of the controller in reducing displacement responses under environmental uncertainties (Dis. = Displacement (m)) 96

6.6 Actuator force range and the number of divisions used for discretization 99

6.7 Utilized Learning parameters 100

6.8 Comparing the controlled and uncontrolled displacement responses to Landers earthquake 102

6.9 Responses to the earthquake excitations (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (m/s^2)). 104

6.10 Performance of the controller in reducing displacement responses under environmental uncertainties (Dis. = Displacement (m)) 106

List of Algorithms

A.1 Crossover operator 126
A.2 Mutation operator 126
A.3 Basic algorithm of framework 127
A.4 Add key states to the mini learning batch. 128

Nomenclature

u_i^C	Controlled displacement response of story i
α	step-size parameter
β	Frequency ratio of the TMD to the structure
$\ddot{U}$	Acceleration vector
$\ddot{u}_{g,t}$	Ground acceleration in time-step t
$\ddot{u}_g$	Ground acceleration
Δt_e	Effective time-range in determing reward values
$\dot{\mathbf{U}}$	Velocity matrix
$\dot{U}$	Velocity vector
γ	Discount factor
Γ_n	See Equation (3.17)
$\mathbb{P}\left[S_{t+1} \mid S_t\right]$	Probability of transition to state s_{t+1} given the state s_t
$\mathbf{\Gamma}$	Ground acceleration guiding matrix
$\mathbf{C}$	Damping matrix
$\mathbf{K}$	Stiffness matrix
$\mathbf{M}$	Mass matrix
$\mathbf{P}$	Load guiding matrix
$\mathbf{U}$	Displacement matrix
$\mathbf{u}(t)$	Displacement response of the building
$\mathbf{u}_n(t)$	mode nth displacement response
$\mathcal{P}_{\mathrm{ss}}^{\mathrm{a}}$,	Probability of getting into state s' by taking action ain state s
$\mathcal{R}_{\mathrm{ss}}^{\mathrm{a}}$,	Reward obtained from ss' transition
$\mathcal{S}$	Set of states

ω_n	Natural frequency (undamped) (rad/sec)
ϕ_{jn}	Modal displacement for mass n in jth mode
ϕ_n	nth natural vibration mode
$\pi(a/s)$	Probability of taking action a in state s under policy π
π^*	Optimal policy
ψ	Damping ratio of the TMD to the structure
Δt_d	Time-delay betwen action and reward
ζ_n	damping ratio for nth mode
A	Acceleration response vector of structure and TMds
A	Set of actions
a	Action
a_i^C	Cntrolled acceleration response of story i
a_i^{UC}	Uncontrolled acceleration response of story i
a_M	Maxming uncontrolled acceleration response
A_t	Taken action action at the time t
a_t	Acceleration in time-step t
C	Damping matrix
C_{st}	Damping matrix of the structure
c_{st}	Damping of the structure
C_t	Damping mtrix of the TMD
c_t	Damping of the TMD
f	Frequecy
f_t	Actuation force at time-step t
G_t	Sum of the future discounted rewards in t
i	Story number
J_i	Objective function i
K	Stiffness matrix
K_{st}	Stiffness matrix of the structure
K_t	Stiffness matrix of the TMD

M	Mass matrix
m_0	Mass ratio of the TMD to the structure
m_j	Modal mass for the jth mode
M_{st}	Mass matrix of the structure
m_{st}	Mass of the structure
M_t	Mass matrix of the TMD
m_t	Mass of the TMD
N	Number of stories
P_{ac}	Penalty value for unit force of actuator
P_t	inertial forces
$q^*(s)$	Value of taking action a in state s under optimal policy
q^π(s,a)	Action value for action a under policy π in state s
$q_n(t)$	nth modal coordinate
$q_n(t)$	modal coordinate for nth mode
$R^{(n)}$	n-step look ahead reward function
R_M	Maximum reward vale
s	State
S_t	State at time-step t
U	Displacement response vector of structure and TMds
u_i^{UC}	Uncontrolled displacement response of story i
u_M	Maxming uncontrolled displacement response
u_t	Displacement in time-step t
V	Velocity response vector of structure and TMds
$v^*(s)$	Value of state s under optimal policy
$v^\pi(s)$	Value of state s under policy π
v_i^C	Controlled velocity response of story i
v_i^{UC}	Uncontrolled velocity response of story i
v_M	Maxming uncontrolled velocity response
v_{st}	Frequency of the structure

v_t	Frequency of the TMD
v_t	Velocity in time-step t
ϵ	exploration and exploitation adjustment parameter
$\mathrm{Q}(s, a)$	Vaue of the action a in the state s
$\mathrm{Q}^*(s, a)$	Maximum expected return for action a in state s

Chapter 1

Introduction

1.1 Introduction

Earthquakes are natural hazards responsible for damaging structures, taking many lives, and causing financial and social losses every year. Recent earthquake events show that, despite advances in seismic codes and control systems, even buildings in large cities of developed countries can be very vulnerable to earthquakes, which implies the necessity for developments in this area (see Figure 1.1).

The seismic control of structures is a challenging task in structural engineering. On one hand, earthquakes have a stochastic nature and wide frequency content; on the other hand, there are many uncertainties associated with modeling and seismically evaluating structures. The real behavior of the materials, the exact magnitude and distribution of the loads, and boundary conditions are some examples of such uncertainties in structural modeling. These issues are even more significant in important civil structures such as tall buildings, in which case the absence of a particular performance level during the earthquakes may result in noticeable social, economical, or political loss.

Figure 1.1: Collapse of tall buildings during the Taiwan earthquake in 2018[1]

A modern answer to these issues is the implementation of structural control systems, first introduced by Yao in 1972[8]. These control systems increase the structures' load-bearing capacity to withstand external excitations. These systems are broadly categorized as passive, active, semi-active, and hybrid systems.

From a historical point of view, passive control systems such as base isolations and tuned mass dampers (TMDs) were the first of these systems to be implemented. Passive control systems have been heavily researched and are already utilized in many countries[9, 10, 11, 12]. The advantage of these systems is that they do not require a power supply or additional hardware or software to work.

Passive controllers such as TMDs have been widely utilized for the vibration control of tall buildings. Generally, the seismic control of tall buildings is a challenging task due to their complex seismic behavior and a higher degree of uncertainty compared with mid- and low-rise buildings.

Because a single TMD can be tuned based on a single frequency, it can be subjected to miss-tuning, which reduces their performance during external excitations. To solve this issue, the use of multiple tuned mass dampers (MTMDs) was introduced; this involves attaching multiple TMDs to a structure, usually at the top level, and tuning each one to a particular frequency so that the control system can cover a wider frequency range than a single TMD system.

In tall buildings, due to the possibility of noticeable participation of the higher modes in the total seismic response of the structure, the performance of the MTMD system may be further improved by distributing the TMDs over the height of the building. However, identifying an optimal placement for the TMDs for the seismic control of tall buildings is a challenging task that is still not well studied in the literature. As a contribution in this regard, this research studies the optimum placement and properties of the TMDs to propose distributed MTMDs as a seismic controller system for tall buildings.

The main shortcoming of passive controllers is that their performance is very sensitive to their design parameters. Besides, they can not adapt their behavior to changes in their environment. For example, they can not distinguish between wind and earthquake excitations. As a solution, active and semi-active controllers were developed to overcome these issues. Utilizing these systems in civil structures has created a new generation of structures called *smart structures* that can determine their environment via sensors and adjust the controller behavior accordingly during natural hazards.

The essential component of a smart structure is its control algorithm, which determines the behavior of the controller system during the external excitations. Developing of an appropriate control algorithm has been a focal research point in the last decades[13, 14, 15, 16, 17, 18]. Various approaches have been adopted in the literature for developing optimal control algorithms. These approaches can be divided into two main categories:

1. classical approaches: in which some mathematical functions are utilized to determine the optimal control signals. linear-quadratic regulator (LQR), LQG, H_2, and H_∞ algorithms are some examples of classical control algorithms.

2. Intelligent approaches: in which computational intelligence methods are uti-

lized to determine the control forces. Examples of such controller types include fuzzy logic controllers (FLC), neural-fuzzy controllers, and genetic-fuzzy controllers.

Intelligent approaches are more advanced than the classical approaches due to their ability to handle uncertainties and changes in the environment, as discussed in Section 2.5.

Developing a robust algorithm for an intelligent controller is a challenging task. The current approaches for developing intelligent control algorithms for smart structures can be classified into the following methods:

- Developing a control policy based on classical control methods, and using computational intelligence methods to develop an intelligent algorithm that switches between defined policies based on the input data.
- Developing a control policy and adjusting the controller parameters based on the inputs.
- Fitting a control policy to an environment including a dynamic system and external excitations.

Concerning the first two approaches—both classical approaches—one of the main issues with developing optimal control policies is that they need some predefined policies to switch between or adjust them based on the input. In the latest family of approaches, the complicated task of developing the control policy would be undertaken by the learning algorithms.

The main issue with the latter approach is that the policy can be subjected to overfitting to the train data. This means that despite the controller performing well under the training data, it shows poor performance under test data. To improve the performance of such controllers to counteract overfitting, some researchers used neural networks (NN) to develop intelligent controllers.

The literature shows that, particularly under earthquake excitations, despite the capabilities of the NN in generalization, the performance of the developed controller is still very sensitive to the training data-set. This issue is related to the way that the NNs are trained. In the studied literature, the NNs were directly trained using a training data-set with the common back-propagation algorithm Levenberg-Marquardt. In machine learning, this type of learning is called *supervised learning*. In this method, the goal of the learning algorithm is to maximize its performance, which is evaluated by a cost function. As a result, the final performance of a well trained NN would be good if the test data-set is similar to the training data-set; conversely, performance would be reduced by distancing the test data-set and the training data-set. In seismic control problems, because each earthquake has its characteristics, in practice it is not possible to find an excitation record that guarantees that future earthquakes will similarly impact a particular building. Moreover, the trained NN is also trained based on an assumed set of structural characteristics, which are not constant and change during the life-cycle of the structure.

In this regard, the initial idea of this research developed based on this question: How can we develop an optimal control policy based on existing earthquake records

and an imperfect mathematical model of a dynamic system in a way that uses most of the existing data to protect structures from future earthquakes? Answering this question motivated the author to study the advances in *artificial intelligence,* which is widely used in computer science and robotics.

Artificial Intelligence (AI) refers to intelligence demonstrated by machines, in contrast to the natural intelligence displayed by humans and other animals. Artificial intelligence research focuses on "intelligent agents" that can perceive their environments and take actions to achieve a particular goal or learn to do a task through flexible adaptation [19, 20].

Artificial Intelligence is widely used in many industries such as education, health, commerce, transport, and robotics. The main reason that AI attracts so much research is related to its robustness, applicability, and flexibility in solving complex problems that were not attainable by computers in the past. Examples of such problems include reasoning, knowledge development, planning, communication, and perception [21, 22]. Another area that is highly influenced by AI is autonomous systems such as autonomous cars and helicopters[23]. The way that these systems learn to do a task is comparable to the process of human learning. As an example, the autonomous helicopter developed by Stanford University was able to learn how to move and control the movements of an inverted helicopter—a very complex task—directly by taking actions and observing the result [24]. The successful uses of AI in developing self-organized systems suggest that it has the potential to be used in structural seismic control problems as studied in this research.

This research aims to use an AI method inspired by the human brain called *reinforcement learning* (RL) to train a neural network that generates control signals to improve the vibrations of the structure under external excitations. The difference between this approach and previous approaches is that in this method, the controller interacts with the structural model by applying control forces, observing the results, and determining the training data-set based on those observations and the defined problem goal, i.e., improving the structural responses. As a result, during the training episodes, instead of focusing on improving the structural responses alone, the focus of the algorithm is to develop a control policy that improves the structural responses.

1.2 Thesis Outline

The structure of this research is guided by the following objectives, which form the outline of the thesis as follows:

Chapter 2 studies the literature relative to this research. Studies about the uses of TMDs in the vibration control of structures are presented with a focus on their uses in the vibration control of tall buildings. Then, some recent studies on active and semi-active control systems are presented and the literature surrounding different control algorithms is studied. Finally, the studied literature is summarized and the research objectives are defined.

Chapter 3 takes a computational intelligence approach to study the uses of the TMDs in the seismic control of tall buildings. This is the first objective of this research. In this chapter, initial theories about the genetic algorithm and NSGA-II methods are

presented. Then, a framework that automatizes studying the optimum placement and parameters of TMDs in tall buildings is proposed. Finally, the framework is examined on a 76-story benchmark as a case study and the results are presented and discussed, and the conclusions are drawn.

Chapter 4 Studies reinforcement learning as the basis of the second objective of this research, that is, developing an intelligent control system. In this chapter, the characteristics of an RL problem and a definition of a Markov decision process (MDP) are studied and different methods for solving an MDP are compared to find an appropriate method for developing the intelligent framework.

In chapter 5, first, the structural control problem is mapped to an RL problem. Then, the resulting issues of such a mapping process are addressed by improving the method and performing the required adaptations. After that, the studied theories and the developed methods are applied to develop an intelligent framework. After that, the framework is examined in two case studies, to train a deep NN as an intelligent controller of the structure. Finally, the results are discussed and the conclusions are drawn.

Chapter 2

Literature Review

2.1 Introduction

In general, structural control systems are divided into passive, active, semi-active, and hybrid control systems. Passive control systems work without power supply. They utilize various techniques such as isolating the structure from external excitations, developing forces to response to the structural motions, or dissipating the strain energy to improve the structural responses under the external excitations[25]. Some examples of passive control systems are tuned mass dampers (TMD), base isolators, viscous dampers, and frictional dampers. Among these techniques, TMDs are very common in the research and application of structural control problems and are also used in the first part of this research, which relates to the vibration control of tall buildings. The studied literature related to the use of TMDs in structural control problems is presented in this chapter.

On the other hand, active and semi-active control systems can adapt their performance to their environment, but they require a power supply and additional instruments such as computers or electronic boards. The essential component of such a control system—which is also the focus of this thesis—is the *control algorithm,* which determines the behavior of the control system under external excitation. In the following sections, this chapter will first present an overview of the semi-active and active control systems and then outline the recent advances in control algorithms.

2.2 Seismic Control of Tall Buildings

Given the modern development plans that large cities must develop to answer the needs of their fast-growing populations, it is anticipated that the buildings will continue to become taller and more expensive [26]. Therefore, the area of investigating solutions to protect tall buildings during natural disasters has gained much attention. Figure 2.3 schematically demonstrates the control systems used in some of the tallest buildings in the world. As has been shown, TMDs are effectively utilized in the vibration control of tall buildings.

2.2.1 Tuned Mass Dampers

A Tuned Mass Damper (TMD) is a mechanical energy dissipating system consisting of a mass, a spring, and a damper (see Figure 2.1) that can be tuned to a particular frequency. Generally, the TMDs are tuned relative to a particular structural response frequency so that when the structure vibrates in that frequency, the TMD will resonate out of phase and produce forces that work against the structural motion. According to the literature, the first concept of TMDs was applied by Frahm in 1909[27] to mitigate the rolling motion of a ship. Since that time, TMDs were widely utilized in the vibration control of dynamic systems as well as civil engineering structures such as bridges [28, 29, 30, 31, 32] and buildings [33, 34, 35, 36, 37, 38, 39, 40, 41]. A well-known example of such an application is Taipei 101, a 500-meter high building that uses a pendulum as a TMD, suspended from the 92nd floor and reaching the 87th floor. This building is designed to withstand typhoons and strong earthquakes (see Figure 2.2). The studies show that using a single TMD at the top level of a tall building can effectively reduce wind-induced motions [16, 42]. This is because generally, the structures response to the wind excitation based on the first structural mode, in which the top levels of the building have maximum modal responses compared with the other stories. Therefore, placing a single TMD on the top level with a tuning frequency closer to the fundamental structural frequency can efficiently reduce the structural responses.

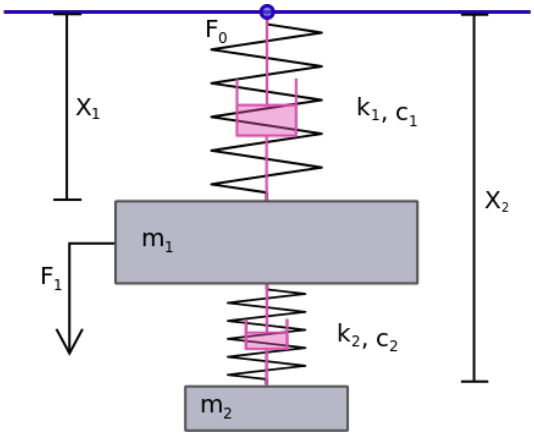


Figure 2.1: Tuned mass damper

Unlike the simplicity of its utilization, the design process of a TMD for seismic control is a challenging task because its performance is very sensitive to its tuning frequency. Generally, the design parameters of a TMD refer to its damping, tuning frequency, mass, and the corresponding values of the structures. In some studies, the optimum parameters for the TMDs were determined by developing the governing mathematical equations and minimizing the structural responses to the external excitations [31, 43, 37, 44, 45, 46, 47, 48, 49]. As an alternative, some researchers utilized computational intelligence methods to study the optimum parameters of the TMDs. Arfiadi et al. [35] used a hybrid-coding genetic algorithm method to study the optimum properties and the location of a TMD for a 10-story building under earthquake excitations. Pourzeinali et al. [50] utilized a non-dominated genetic

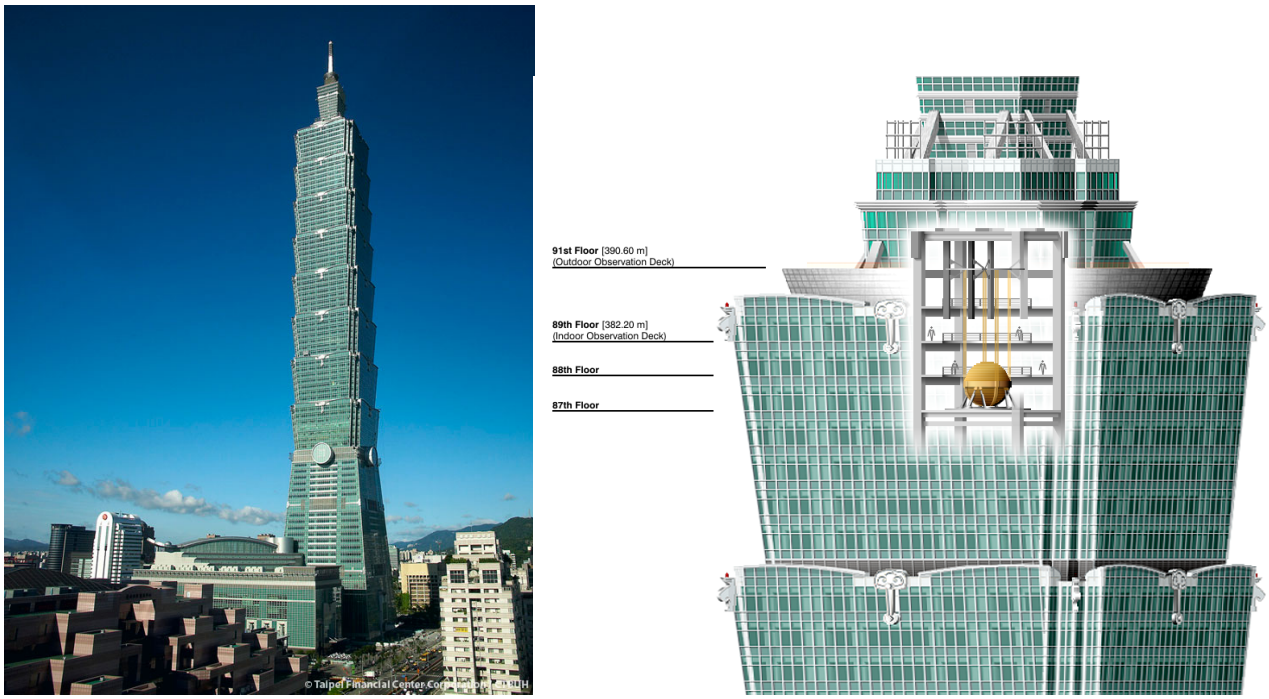


Figure 2.2: Taipei 101's tuned mass damper [1]

algorithm to perform multi-objective optimization to study the design parameters of a single TMD in a high-rise building under earthquake excitations.

2.2.2 Multiple Tuned Mass Dampers

As the TMDs can be tuned to a particular frequency, they are effective in a narrow band around the tuned frequency, which limits their applications in seismic control. To overcome this limitation, multiple tuned mass damper (MTMD) systems were studied[51, 36, 30, 52, 53, 54]. Multiple tuned mass damper systems cover a wider frequency range than single TMD systems. To find the optimum design parameters for the MTMD system, different methods have been utilized by researchers.

Li and Zhu [55] proposed a novel optimum criterion to optimize the properties of double TMDs for structures under ground acceleration. Li and Qu [54] considered the structure as a single degree of freedom (SDOF) system, which was equipped with MTMDs, and studied the optimum parameters of the TMDs. Chen and Wu [56] proposed a step-by-step procedure for the optimal placement of MTMDs in buildings. They examined their proposed method on a numerical model of a six-story building. Li and Qu [57] utilized a multi-objective optimization method to study the optimum parameters of MTMDs. Tharwat [58] utilized partial floor loads as MTMD systems to improve the responses of multi-story buildings.

However, in these studies on the seismic control of tall buildings, either a single TMD is utilized or the MTMDs were placed on the top level of the building. Therefore, there is still no comprehensive research on the utilization of TMDs for the seismic control of tall buildings. This research aims to identify the optimal placement and parameters of TMDs in the seismic control of tall buildings, without considering any

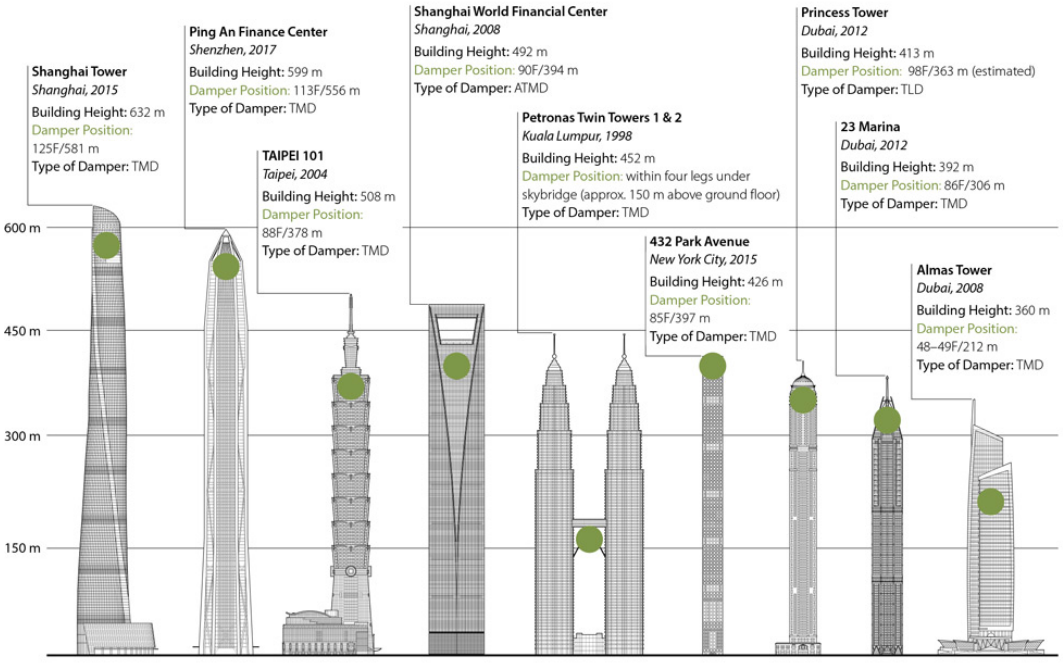


Figure 2.3: Tallest completed buildings with dampers [2]

limitations of the number or placement of the TMDs.

2.3 Active Control of Structures

One of the main shortcomings of a passive control system is that the system cannot be adjusted or changed easily or without incurring significant expenses. However, it is very likely that the assumed environmental or structural parameters will change over time due to issues such as: (a) degradations in the structural element's strength or stiffness due to damage or environmental effects; (b) changes in structural loads; or (c) changes in the serviceability of the structure, which affects the seismic performance demands. In addition, due to the stochastic nature of earthquakes, providing some type of adaptation by correlating the control system's behavior to the external excitation increases the safety level of the system.

Active and semi-active control systems overcome these issues by detecting their environment through sensors and adjusting their behavior accordingly. Depending on whether the structural responses are considered as feedback when calculating the control forces or not, the active control systems are categorized as either *closed-loop* or *open-loop* controllers. Generally, these controllers consist of three main parts including sensors, actuators, and the control algorithm. Sensors provide data about the environment to the control algorithm by measuring external excitations or structural responses. Actuators translate the control signals received form the control algorithm to external forces on the structure. There are several types of actuators, namely hydraulic, pneumatic, electromagnetic, piezoelectric, or motor-driven ball-screw actuators [59]. The control algorithm determines the behavior of the controller by mapping the input data from the sensors to the optimal control commands.

Of the different types of active controllers, active tuned mass dampers and active tendons are more common in structural control problems.

2.3.1 Active Tuned Mass Dampers

In the use of active tuned mass dampers (ATMDs), an actuator is placed between the structure and a mass and applies dynamic forces to the structure during external excitations, as shown schematically in Figure 2.4.

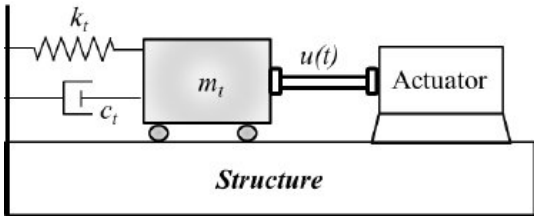


Figure 2.4: Schematic diagram of an ATMD [3]

Wu and Yang [60] studied the utilization of different control algorithms including LQG, H∞, and continuous sliding mode control (CSMC) for the active control of vibrations of the 310-meter Nanjing TV transmission tower in China. They concluded that, when designed appropriately, the LQG, H∞, and CSMC control strategies are suitable for full-scale implementations of the active mass driver.

Li et al. [14] studied the performance of theH_2 algorithm in the vibration control of offshore platforms. Saleh and Adeli [61] studied the utilization of multiple ATMDs in the vibration control of a multi-story building, which was subjected to blast loads. They showed that a multi-ATMD system can effectively improve structural responses.

Yamamoto et al. [62] investigated the utilization of ATMD systems with a hydraulic actuator in the vibration control of high-rise buildings. They examined their method on various buildings with a height range of 58 to approximately 190 meters, between 11 and 34 stories high. They also verified the results by performing forced vibration tests. In the verification tests, they utilized the ATMD systems to apply lateral forces to the structures as external excitations. They concluded that the ATMDs effectively reduced vibrations in the studied structures.

An important development in this area was made by Ikeda et al. [63], who studied the utilization of two ATMDs to control a 10-story building in Tokyo. In this study, the researchers utilized a linear-quadratic regulator (LQR) algorithm to control the lateral displacement and torsional motions of the building. The target building had already experienced one earthquake and wind loads and showed a 26% and 11% reduction in lateral and torsional motions, respectively, during an earthquake and a 33% reduction in displacement response peaks under wind loads. Abdollahirad et al. [64] combined three different algorithms including a discrete wavelet transform (DWT), a particle swarm optimization (PSO), and anLQR to develop a hybrid algorithm for the active control of structures. They examined their method on a 10-story structure subject to several ground motions, which resulted in better performance than LQR controllers alone. In order to control the strokes of the auxiliary mass in

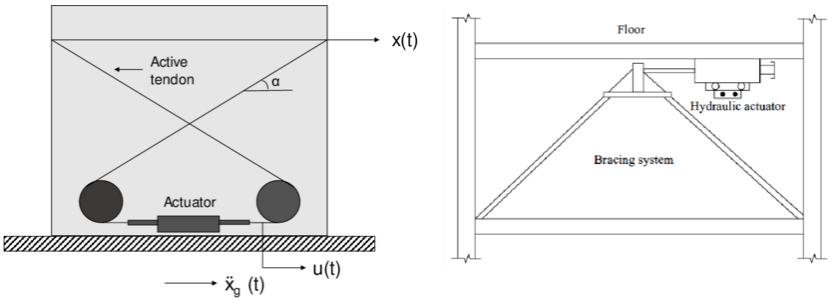


Figure 2.5: Schematic diagram of an active tendon controller [4]

active controllers, Iba et al. [65] utilized a neural oscillator and position controller to improve the performance of the previous ATMDs.

2.3.2 Active Tendon Systems

An active tendon system involves positioning prestressed cable braces between two stories of a building and an actuator that adjusts the levels of tension forces in the cables when the control forces act on the structure (see Figure 2.5). The literature shows that active tendon systems can be effectively used for the vibration control of bridges [66, 67, 68, 69]. Rodellar et al. [70] proposed a tendon control system to mitigate earthquake and wind vibrations in a long cable-stayed pedestrian bridge with a length spanning 142.5 meters. They utilized a Lyapunov-based controller to design the active control system. The bridge, located in Taft, California, was subjected to an earthquake in 1952. The results showed that the active tendon system significantly improved the response of the bridge.

In another study, Preumont et al. [71]compared different control strategies for controlling the vibration of suspension bridges equipped with active stay cables. They investigated several configurations where the active cables connected the pylon to the deck, or the deck to the catenary, to control the vibrations in an existing footbridge.

Nazarimofrad and Zahrai [72]studied the soil-structure interaction of a building equipped with an active tendon controller. They utilized an LQR algorithm to develop the controller and examined it on a 10-story building.

2.4 Semi-Active Control Systems

The difference between semi-active and active systems is that semi-active controllers require much lower energy to work so that, in practice, they can operate on a battery. The two main types of semi-active systems are studied in this section.

2.4.1 Magnetorheological Fluid Dampers

In a magnetorheological (MR) damper, the fluid in the piston-cylinder system produces damping forces; the current of the fluid can be controlled by the control algorithm in real-time. In the absence of power, the MR damper can act as a passive damper by mitigating the vibrations.

Jung et al. [73] studied the utilization of MR dampers in the vibration control of cable-stayed bridges subjected to earthquakes. They modeled an American Society Of Civil Engineers (ASCE) benchmark cable-stayed problem, which was developed based on the Cape Girardeau bridge measuring 633 meters long with two cable-stayed towers. They utilized 24 MR fluid dampers, each with the capacity to bear 1000 of force. The dampers were placed in four different locations on the bridge and a clipped-optimal and H_2/LQG algorithms were considered as control algorithms. In another study, Moon et al. [29] compared sliding mode controllers (SMCs) and LQG algorithms in the vibration control of the benchmark Cape Girardeau cable-stayed bridge. They concluded that the SMC algorithm is more effective for an MR system.

Xu et al. [74] studied the performance of MR dampers in controlling earthquake-induced vibrations in buildings with a podium structure. They studied a 12-story steel frame building with a surrounding three-story podium structure under several scaled earthquake records.

2.4.2 Semi-active Stiffness Dampers

A semi-active stiffness damper (SASD) consists of a fluid-filled cylinder, a piston, and a motor-controlled valve. The controller damping coefficient of the system works by controlling the valve opening, which affects the flow of the viscous fluid.

Fukukita et al. [11] studied the effectiveness of the SASD systems with an LQG controller compared with the viscous damping wall system (walls that consist of two plates with a viscous fluid filling the void between them). They examined both controller systems on a two-dimensional, 20-story benchmark model under the 1940 El Centro, 1968 Hachinohe, and 1994 Northridge earthquake excitations. They concluded that the passive viscous damping walls showed a better reduction of the peak acceleration and drift by 8% and 24%, respectively, compared with the SASD system.

2.5 Advances in control Algorithms

2.5.1 Introduction

The control algorithm is the main part of an active control system because it is responsible for determining the behavior of the controller during external excitations. In general, control algorithms are divided into open-loop, feedback, feedforward, and hybrid controls, which are a combination of the feedback and feedforward controls [75].

In open-loop control, the controller is independent of the "process output", which is the process variable that is being controlled [76]. In closed-loop control, some features of the feedback would be considered as additional inputs for the controller, and as result, a certain degree of adaptation would be achieved [77].

In feedforward control, the output of the controller, but not the system, would be utilized as additional inputs to the controller. Therefore, unlike the closed-loop control, no direct adaptation to the response of the system is considered in the control algorithm [78]. Although the closed-loop control provides more adjustability and are more commonly used in practice than feedforward controls [79], there are

several reasons why feedforward controls would be used instead. For example, when the system response can not be measured easily or when the closed-loop control causes instabilities [80].

A. Classical Control

Classical control algorithms are developed based on a proportional-integral-derivative (PID) controller. A PID controller is a feedback controller that continuously calculates an error value as the difference between a desired setpoint (SP) and a measured process value (PV) and applies a correction control force based on the proportional, integral, and derivative terms, denoted as P, I, and D, respectively [81]. Although the PID controller does not guarantee stability and optimal control, it is widely utilized in the industry by adjusting the tuning parameters based on characteristics of each problem [82].

A1. Feedback Control

Classical feedback controllers have been widely utilized in small- and large-scale systems. For small-scale systems, Dhanalakshm [83] developed a fuzzy PID controller to perform feedback control of a self-sensing Shape Memory Alloy actuated system. In another study, Moore and Moheimani [84] used the velocity feedback to develop self-sensing actuation on microelectromechanical systems (MEMS) nano-positioner.

For large-scale systems, Balas [85] developed a vibration suppressant for large space structures (LSS) using direct velocity feedback control. Large space structures are extremely flexible because of their large size, low rigidity, and low natural damping. Nelson et al. [86] investigated the feedback control of a tied-arch steel bridge in the United States. They considered both active and semi-active strategies. They concluded that the proposed active and semi-active control systems can significantly reduce the response at all critical resonant frequencies. Li et al. [87] proposed a wireless sensing and vibration control system in which positive position feedback controls were used for active vibration control.

A2. Combined Feedback and Feedforward Control

Rodriguez et al. [88] utilized feedback and feedforward controllers to develop a flatness-based active vibration controller for piezoelectric actuators. They concluded that such a strategy improves the damping of the entire operation's frequency range without the instability issues derived from high feedback gains. Li et al. [89] utilized a proportional differential controller and an acceleration feedforward controller to perform multi-mode vibration control for a stiffened plate. A stiffened plate consists of a deep frame, a long truss, and rivets. The long-term vibration of a stiffened plate may lead to fatigue crack, component life reduction, and structural damage. Thus, the vibration control of a stiffened plate has significant applications in industrial settings.

B. Optimal Control

With optimal control, the goal of the control algorithm is to maximize or minimize an objective function that controls the optimum performance [77].

Optimal control is based on a control law that minimizes or maximizes an objective function subject to constraints. The goal is to obtain optimal performance specifications for a given purpose (Ogata 2010). This is usually done offline (Vrabie et al., 2013 [79]).

B1. Classical Optimal Feedback Control

In optimal feedback control, the system dynamics is described by a set of linear differential equations and the cost function is defined by a quadratic function; the control problem is called a linear–quadratic (LQ) problem [90]. Several studies have been carried out on optimal control of smart structures. One of the main solutions for such control problems is provided by the linear–quadratic regulator (LQR) [91], In LQR control, the settings of a (regulating) controller governing either a machine or process (like an airplane or chemical reactor) are defined by a mathematical algorithm that minimizes a cost function with weighting factors. These weighting factors are often defined as a sum of the deviations of key measurements, like altitude or process temperature, from their desired values. The algorithm thus finds those settings that minimize deviations [92].

Ikeda et al. [63] utilized an LQR control algorithm to control the lateral displacement and also torsional motions of a 10-story building in Tokyo using two ATMDs. This building had already experienced one earthquake and several high-winds and showed a 26% reduction in lateral displacements and an 11% reduction in torsional motions during an earthquake, and a 33% reduction in displacement response peaks under wind loads. In another study, Deshmukh and Chandiramani [16] used an LQR algorithm to control the wind-induced motions of a benchmark building equipped with a semi-active controller with a variable stiffness tuned mass damper.

One of the most fundamental optimal control problems is linear–quadratic–Gaussian (LQG), which deals with a linear system driven with additive white Gaussian noise [93]. Under such an assumption, an optimal controller can be derived by a completion-of-squares argument, which is known as an LQG controller and is a combination of a Kalman filter and an LQR controller [94].

Fukukita et al. [11]studied the effectiveness of semi-active stiffness damper (SASD) systems with an LQG controller compared with the effectiveness of viscous damping walls (walls that consist of two plates with a viscous fluid filling the void between them). An SASD consists of a fluid-filled cylinder, a piston, and a motor-controlled valve. The controller damping coefficient of the system works by controlling the valve opening which affects the flow of the viscous fluid. They examined both systems on a two-dimensional, 20-story benchmark model under various earthquake excitations. They concluded that the passive viscous damping walls showed a better reduction of the peak acceleration and drift than the SASD system.

B2. Stochastic Control

Stochastic control is a subfield of control theory that deals with the existence of uncertainty either in observations or in the noise that drives the evolution of the system [95]. Nayyar et al. [96] studied a general model of a decentralized stochastic control problem in which multiple controllers share part of their information—a process known as a partial history sharing information structure. In this study, they first

established a structural property of optimal control strategies and then provided a dynamic programming decomposition of the problem of finding optimal control strategies. As an example, they used two subsystems and controllers in which the decentralized control strategy was reformulated into a centralized control problem by considering the shared information between controllers. Lin et al. [97] addressed the persistent monitoring problem in two-dimensional mission spaces where the objective was to control the trajectories of multiple cooperating agents to minimize an uncertainty metric. They incorporated a stochastic comparison algorithm for deriving global optimal elliptical trajectories.

B3. Model Predictive Control

Model predictive control (MPC) algorithms use a model to estimate the future evolution of a dynamic process to optimize the control signals to minimize or maximize an objective function. The MPC scheme has been commonly used in chemical, automotive, and aerospace industries [98, 99]. Mei et al. [100] utilized the MPC scheme in relation to acceleration feedback in the structural control under earthquake excitations. In this study, they utilized the Kalman-Bucy filter to estimate the states of systems in two case studies including a single-story and a three-story building, in which active tendon control devices were used. Koerber and King [101] utilized the MPC system to control the vibrations in wind turbines. Chandan and Alleyne [102] utilized centralized and decentralized MPCs to regulate the temperature of a nine-zone, three-story square building, and an eleven-zone circular building.

Riverso et al. [103] utilized decentralized MPCs to develop a control scheme for a linear system structured into physically coupled subsystems. This was done to guarantee asymptotic stability and to ensure the constrains satisfied the system inputs and states. They applied the developed control scheme to frequency controllers in power networks.

In another study, Liu et al. [104] studied the performance of a decentralized control system with a general LQG-type index involving both systems and inputs. In that research, they proposed an efficient iterative approach for the evaluation of decentralized steady-state Kalman filter gains. They examined the developed control scheme on two examples including a reactor–separator chemical process.

Palomo et al. [105] developed an MPC based on Laguererre functions for the vibration control of a three-story structure. They used a hybrid control including a passive controller as a tuned mass damper (TMD) located over the third story and a predictive vibration control scheme. The building was subjected to a variety of base excitations with a wide frequency range including the resonance frequencies. They concluded that the developed controller effectively controlled the vibrations in all of the considered excitations.

C. Robust Control

Robust control is an approach used for scenarios that explicitly deal with uncertainties. The goal of robust control is to achieve robust performance and stability in the presence of bounded modeling errors [106]. In contrast to adaptive control, in robust control, the policy is static [107].

C1. H_2 and H_∞ Control

Generally, H_2 and H_∞ controllers are used to synthesize controllers to achieve stabilization with guaranteed performance. In these approaches, the control problem would be mapped to a mathematical optimization problem so that the controller algorithm would be the optimal answer to such an optimization problem. Initial studies on H_2 and H_∞ control were conducted by Doyle et al. [108].

Wu and Yang [60] compared different control schemes including LQG, H∞, and continuous sliding mode control (CSMC) to control the vibrations of a 310-meter Nanjing TV transmission tower in China. They concluded that when designed appropriately, the LQG, H∞, and CSMC control strategies are suitable for full-scale implementations in an active mass driver.

Li et al. [14] developed a control scheme for offshore platforms using H_2 control. They showed that the ATMD system, which was developed using anH_2 algorithm, is more effective than a single TMD system.

Jung et al. [73] utilized clipped-optimal and H_2/LQG algorithms to control the vibrations of cable-stayed bridges subject to earthquakes. They modeled a benchmark cable-stayed problem which was developed based on the Cape Girardeau bridge measuring 633 meters long with two cable-stayed towers. They used 24 MR fluid dampers, each with a force-bearing capacity of 1000 kN, which were placed in four different locations on the bridge. The results of their analysis under various earthquake excitations showed a minimum 69% reduction in all structural responses.

Wang [109] utilized a dynamic output feedback controller that minimized the H_∞ norm of the closed-loop system to present a decentralized approach to the vibration control of large-scale civil structures. He compared the performance of the developed algorithm with a time-delayed decentralized control algorithm, which was developed based on the LQR criteria through numerical simulation of a five-story building under earthquake excitations.

Fallah and Taghikhany [110] utilized H_2/LQG control as a robust-optimum algorithm to develop a centralized and decentralized controller for a cable-stayed bridge. They compared the structural responses in both approaches under different earthquake records considering the time-delay in data transmission. They showed that in large-scale bridges, a decentralized solution performs better than other strategies.

C2. Sliding Mode Control

Sliding mode control (SMC) is a common control method for nonlinear systems in which the dynamics of the system can be altered by a control signal that forces the system to "slide" along a cross-section of the system's normal behavior.

Moon et al. [29] utilized SMC and the LQG formulation for vibration control of a cable-stayed bridge under seismic excitations. They evaluated the robustness of the SMC-based semi-active control system using magnetorheological (MR) dampers.

Wang and Adeli [111] proposed a time-varying gain function in the SMC. They developed two tuning algorithms for reducing the sliding gain function for nonlinear structures.

Wang and Adeli [112] proposed a filtered SMC approach for the vibration control of a 76-story benchmark building equipped with an active tuned mass damper (ATMD)

under wind loads. They showed that this approach can eliminate the high-frequency part of the control force, which results in less excitation of the structure, sensors, actuators, and dampers, and consequently better performance than the unfiltered SMC approach.

Soleymani et al. [113] proposed a modified SMC for the vibration control of a high-rise building under earthquake excitations. They developed a two-loop SMC in conjunction with a dynamic state predictor, which was responsible for considering the model uncertainties and actuator delays to determine the control forces.

C3. Backstepping Control

Backstepping was developed by Kokotovic[114] in 1990. It is a method of stabilizing the controls for a special class of nonlinear dynamic systems where the existing states of a first-order model are used recursively along with a Lyapunov function to stabilize the steady motions of another second-order model [115].

Wang et al. [116] proposed a decentralized adaptive backstepping control for a class of nonlinear time-varying systems. In this research, they developed a bound estimation approach and two smooth functions. They examined the developed method on double inverted pendulums.

Breschpietri et al. [117] developed a backstepping algorithm with delay compensation. They showed that the predictor-based feedback controller can efficiently yield asymptotic convergence for a class of linear systems subjected to the input-dependent input delay. They examined the algorithm on the fuel system of a gasoline engine equipped with indirect injection subjected to time-delay and varying bounded input.

Fan et al.[118] proposed an adaptive failure compensation using a back-stepping approach, which ensures the boundedness in the probability of all the closed-loop signals in the presence of stochastic actuator failures.

D. Intelligent Control

In many real control problems, the controller algorithm deals with a complex system with a high degree of uncertainty. In traditional control algorithms, the performance of the algorithm is dependent on the accuracy of the considered model and its dynamics, which is not possible in many real control problems.

Broadly speaking, intelligent control (IC) underlines what are called "soft computing" techniques to integrate computational process, reasoning, and decision making along with levels of precision or uncertainty in the available data, measurements, and design parameters. Therefore, IC is more realistic and is more useful in complex control problems with high degrees of uncertainty. The first concept of IC was introduced by Fu (as discussed by Housner et al.[25]) to improve the applicability of automatic control systems. Since that time, IC systems have been utilized in different areas.

The goal of IC is to develop an autonomous system that can operate in an unstructured and uncertain environment independently of human interaction [119]. Intelligent control uses various artificial intelligence computing approaches like neural networks, Bayesian probability, fuzzy logic, machine learning, evolutionary computation, and genetic algorithms, as well as the combination of these methods to create hybrid systems such as neuro-fuzzy [120] or genetic-fuzzy [121] controllers.

In the control of smart structures, fuzzy logic control (FLC) has attracted more research in the area of IC for the following reasons:

- Fuzzy logic, which is the basis of the FLC, is a very effective technique for dealing with uncertainties; therefore, FLC systems can handle uncertainties in the model of the structure and the measurements are more easily drawn than in the classical control theory.
- FLC can easily map nonlinear input-output relations.
- The whole fuzzy controller can be implemented on a fuzzy logic chip or in a dSPACE hardware chip, which guarantees faster processing, resulting in less computational time delay [122].

The main issue with fuzzy controllers is selecting the fuzzy parameters, which largely controls the performance of the controller. In this regard, various methods have been developed to achieve optimal parameters for FLC [123, 124, 125, 126] using online and offline methods and implemented in the vibration control of smart structures [15, 127, 128, 129].

In the recent years, computational intelligence methods such as neural networks [130, 131], evolutionary computing [132, 133], and machine learning methods [134, 135] were utilized to develop intelligent controllers.

D1. Adaptive Filters and Wavelet-based Control

Although intelligent controllers can handle structural and environmental nonlinearities and uncertainties to generate optimal control commands,no controller system can maintain optimal performance if the structural parameters or the excitations, for which it is optimized, change. Adaptive filters help the controllers in this regard, by providing some degree of adaptability to the structural and excitation parameters. Generally, adaptive controllers are classified into direct and indirect adaptive systems[136]. In, direct adaptive control, an error would be calculated based on the difference between the responses of the systems and the desired values and would be the basis for adjustments made by the controller. In the indirect method, the parameters would be estimated online and the controller parameters would be calculated as a control design problem based on the estimated parameters of the dynamics of the system.

Kim et al. [137] developed a hybrid feedback-least mean square (LMS) adaptive controller, which can be integrated by feedback control algorithms such as LQR or LQG to form an x-LMS algorithm. In another study, Caiyun [138] proposed an adaptive F-L algorithm based on Lyapunov's stability theory and demonstrated that the proposed algorithm is more effective compared with the FXLMS algorithm using gradient theory in a nonlinear active noise control system.

As an improvement to adaptive-based controllers, wavelet has been utilized to develop wavelet-based controllers. A wavelet is a wave-like oscillation with an amplitude that begins at zero, increases, and then decreases back to the zero. Wavelet has been shown to be very usable in structural control problems. For example, wavelets are used to extract information from an unknown signal, such as an earthquake signal

[139]. Adeli et al. [140, 78] introduced the wavelet transform as a filtering scheme for control problems by developing a hybrid feedback-LMS algorithm for the robust control of civil structures. Amini et al. [141] developed a wavelet-based adaptive pole assignment method for the vibration control of structures. In another study, Wang et al. [142] developed an adaptive control algorithm for the vibration control of large structures under dynamic loading by integrating a self-constructing wavelet neural network and an adaptive fuzzy sliding mode control approach.

D2. Evolutionary Computing

Evolutionary computing methods refer to a family of nature-inspired methods in which the possible solutions form the *population* which is then subjected to some nature-inspired operators to create a new generation. The new generation is always fitter or at least on par with the previous generation. Therefore, the generations would converge to the global extremes of the problem. The main advantage of these algorithms compared with the classical methods is that they are less sensitive to the initial values and also less likely to become trapped in local extremums.

Among the evolutionary methods, the genetic algorithm (GA) [143] is a more common approach to solving the structural control problem [144]. In most of the studied literature, the genetic algorithm was effectively used to obtain the optimum fuzzy parameters in fuzzy based controllers [145, 126, 146, 147].

Some other evolutionary methods that were used in structural control include particle swarm [148, 149, 150, 151, 39], artificial bee colony [152, 51, 153, 154], and ant colony [155]. Game theory-based algorithms are another type of evolutionary computing-based algorithm.

D3. Neural Controllers

As an alternative, neural controllers have been investigated in a few studies [156, 157, 158] in which the neural networks are utilized to generate the control commands.

In these studies, a backpropagation (BP) algorithm is utilized for offline training the neural network to generate the control commands. The BP algorithm requires target data, which are the desired control commands to train the neural network so that it produces the same commands with an acceptable error on the training data. The trained neural network would then be tested on the untrained data. Therefore, in the previous studies, neural networks were not more than a highly nonlinear function which estimates the control commands given the external excitation and based on a dictated control policy.

Madan [159] utilized a non-supervised learning method to train a neural controller to improve the responses of the structure to earthquake excitations. He implemented a modified counter-propagation neural network (CPN) and trained the controller using earthquake excitations data. The main advantage of his controller over previously developed controllers was that it was not required to define optimum control commands or optimum responses to train the controller because unsupervised learning occurred. However, this controller was developed based on the data clustering by the CPN. Although the controller was trained to reduce vibrations under the training data, no controlling policy was developed by the controller for the test data, and so

the algorithm worked by switching between some trained patterns. Because earthquake excitations are stochastic, in a real-life scenario, an earthquake doesn't follow the pattern of previous earthquakes and, also, during earthquake excitations, there is no time for a retraining process to occur.

2.6 Research Objectives

Considering the studied literature, the objectives of this dissertation are summarized as follows:

a. Passive control of tall buildings using TMDs

In the studied literature relative to TMDs, either the parameters of the TMDs have mainly been studied under wind-induced vibrations or a single TMD has been studied under earthquake excitation; currently, studying the optimum parameters of TMDs under earthquake excitation, without limiting the number and location of the TMDs, is still a challenging task, because of the stochastic nature of earthquakes and the complex seismic behavior of such buildings, which mandate extensive and thorough studies.

Additionally, in contrast to wind loads, during an earthquake, the higher modes may have more noticeable participation in the total response of tall buildings. This is mainly because of the (1) low frequency of the higher modes in these structures, compared to low- and mid-rise buildings, and (2) the wide frequency content of the earthquakes, which may activate the multiple modes in such buildings. Therefore, only controlling the lower modes by placing TMDs on the top levels would not necessarily optimally control the motions in these buildings during earthquakes. This research addresses these issues by studying the optimum placement and properties of TMDs in tall buildings that are subjected to earthquake excitation.

b. Intelligent control of smart structures

In the previous research on active control, the control algorithm either followed a dictated policy, switched between predefined policies, or adjusted and adapted some predefined control policies based on external excitations or structural parameters.

As a development in the area of intelligent control, this dissertation proposed an intelligent framework that creates an intelligent control system as a trained deep neural network through an automatic process. As a result, the complicated process of developing an intelligent controller algorithm would be simplified to preparing input for the framework including:

(1) a mathematical model of the dynamic system; and

(2) the desired external excitations.

Furthermore, the complicated task of developing the controller would be undertaken by the framework using artificial intelligence techniques. The method of developing the framework is a machine learning method called reinforcement learning (RL), which is inspired by the learning mechanism of the human brain. This method has been successfully implemented in creating self-organized systems and has solved some

challenging real-world problems. Autonomous helicopters[160], robotics [161, 162], automatic traffic signal control [163], chess mastery [164], and AlfaGo [165], which defeated the world's number one Go player, are some examples of the accomplishments of RL.

In this method, an *agent* learns to improve its *policy* to increase the chances of obtaining more *rewards* by interacting with an *environment*. Compared to the studied literature, the RL method helps the framework to develop a control policy based on certain experienced states. This would reduce the sensitivity of the controller to the inputs. As discussed in the Chapter 5, the resultant controller shows a stable performance despite changes in the inputs including structural parameters and external excitations. This ability is more important in the case of earthquakes because of their stochastic nature. For this reason, this thesis has considered different earthquake records to examine the developed intelligent framework.

Chapter 3

Seismic Control of Tall Buildings using Tuned Mass Dampers

3.1 Introduction

This chapter addresses the utilization of TMDs for vibration control of tall buildings during earthquake excitations. The objective is to develop a framework to find the optimum arrangement and parameters for TMDs which maximizes the performance of the system in terms of reducing the structural responses. To solve the resultant multi-objective optimization problem, an improved revision of the *Non-dominated Sorting Genetic Algorithm* (NSGA II [166]) is developed and applied to the problem. As a case study, a 76-story benchmark building, subjected to seven scaled earthquake excitations is considered.

3.2 Fast and Elitist Multi-Objective Genetic Algorithm

3.2.1 Introduction

Genetic Algorithm (GA) is a nature-inspired method within a larger family of methods called *evolutionary optimization methods*. Genetic algorithms rely on bio-inspired operators, such as mutation, crossover, and selection, to search for the solution space and find an optimum solution regarding the fitness function [167].

3.2.2 Elements of Genetic Algorithm

In GA method, a *population* of the solutions, called *individuals*, to an optimization problem is evolved toward better solutions. Each candidate solution has a chromosome consisting of an encoded set of variables.

The evolution usually starts from a random population and proceeds as an iterative process of performing genetic operations on the individuals to produce new members, while selecting the elite members (fittest individuals) to form new generations. The new generation of solutions is then used in the next iteration of the algorithm. Generally, the termination criteria involve either reaching a maximum number of generations or the desired fitness value[168].

- **Encoding**

In GA, the solutions are represented by encoding the variables. One of the commonly used encoding techniques is binary encoding in which the variables are coded to strings of 0s and 1s to form the chromosomes.

- **Selection**

During each successive generation, a portion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions are typically more likely to be selected.

- **Fitness**

The fitness function quantifies the quality of the represented solutions. In multi-objective problems, the fitness function includes some sub-functions, each related to a particular objective.

- **Genetic operators**

These include crossover and mutation. In the crossover operation, the algorithm exchanges some of one parent's genes with those of the other, and in the mutation, it changes some genes of one parent [169, 170, 171].

3.2.3 Fast Elitist Multi-Objective Genetic Algorithm

In this research, the fast elitist multi-objective (NSGA-II) method [166] is improved to investigate the optimum arrangement and properties of TMDs in a tall building. In this method, the initial population is randomly generated, as in a normal GA procedure, and then the algorithm sorts the population with respect to the *non-domination rank* and the *crowding distance*.

1. **Non-domination rank**

In general, X dominates Y if X is no worse than Y in all the objectives and if X is better than Y in at least one objective. In the next step, the non-dominant set in the population is selected as the first front. The second front contains the sets that are only dominated by the first front sets. This procedure continues until all the members in the population have been categorized into different fronts. The fronts are then sorted from the first to the last. Figure 3.1 shows an illustration of solutions belonging to different ranks [5].

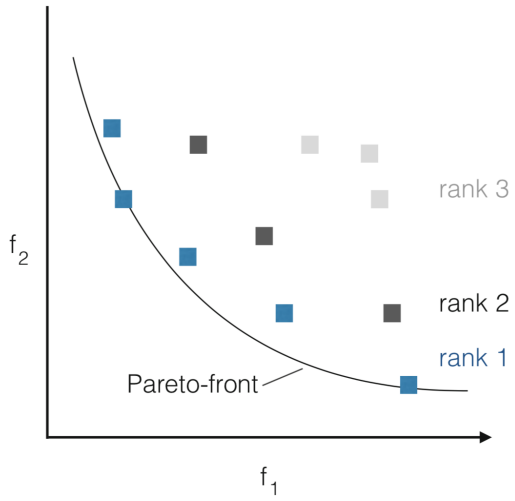


Figure 3.1: Non-dominated sorting of solutions[5]

2. **Crowding distance**

Among the non-dominated solutions or a union of the first ranks of non-dominated solutions, NSGA-II seeks a broad coverage. This is achieved with the crowding distance, which is the Manhattan distance between the left and right neighboring solutions for two objectives, as shown in Figure 3.2.

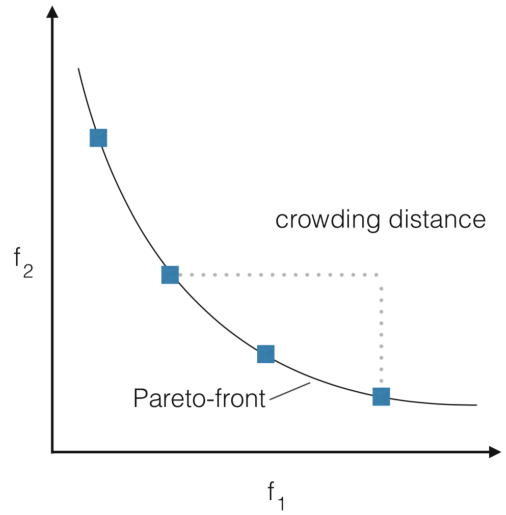


Figure 3.2: Illustration of crowding distance[5]

Consequently, each solution that cannot be dominated by other solutions and has a larger crowding distance than the others will obtain the first rank and so on.

3.2.4 Repair Module

The repair algorithm maps a solution in the infeasible region into a close solution in the feasible region. Figure 3.3 schematically shows the repair approach for a solution space with an infeasible solution and two solutions in the feasible region. In this research, an infeasible solution includes the out-of-limit properties for TMDs. The developed repair function calculates the shortest distance of the TMD properties (m_0, β, ψ) between the infeasible solution and feasible solution and corrects the chromosome with respect to the calculated distance.

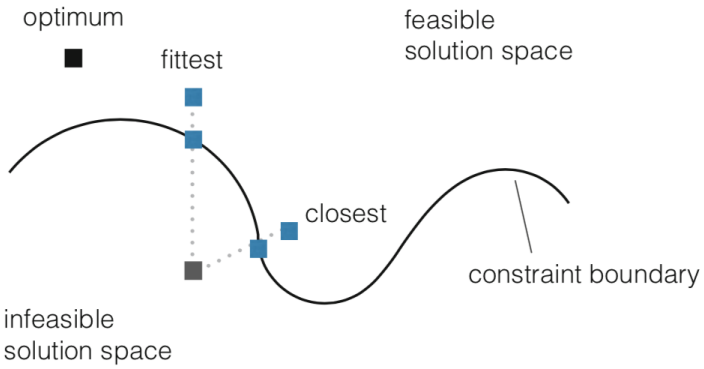


Figure 3.3: Repair of an infeasible solution[5]

3.2.5 Selection

The objective of selection is to choose the fitter individuals in the population to create off-springs for the next generation and then place them in a group commonly known as the mating pool. The mating pool is then subjected to further genetic operations that result in advancing the population to the next generation and hopefully closer to the optimal solution. As it is schematically illustrated in Figure 3.4, the principle of roulette selection is a linear search through a roulette wheel with the slots in the wheel weighted in proportion to the individual's fitness values. All the chromosomes (individuals) in the population are placed on the roulette wheel according to their fitness value [172].

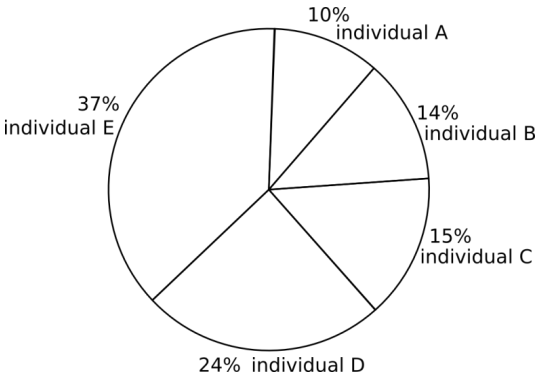


Figure 3.4: Roulette wheel selection

3.2.6 Optimization Variables

The optimization variables for multiple distributed tuned mass dampers system include:

1. Number of TMDs
2. Position of the TMDs→ story number
3. TMDs' properties→ m_0, β, ψ

These parameters are schematically shown in Figure 3.5.

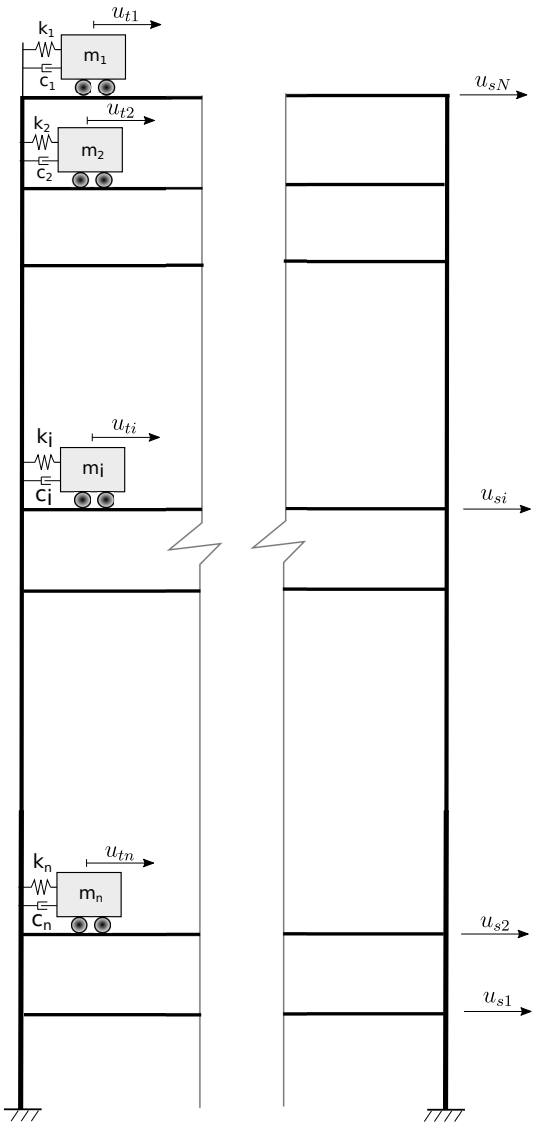


Figure 3.5: Building equipped by TMDs

3.2.7 Encoding

In this research, binary coding has been considered for coding TMD parameters into a binary string with a constant number of genes, as is shown in Figure 3.6. The parameters of a TMD is defined as follows:

$$m_0 = \frac{m_t}{m_{st}},\ \beta = \frac{\nu_t}{\nu_{st}},\ \xi = \frac{c_t}{c_{st}} \tag{3.1}$$

where the t and st indexes correspond to the TMD and the structure respectively. The chromosomes are then created by combining all genes for each solution. As a result, each chromosome contains the coded data of all TMDs in the building. Using this definition, the position of each TMD is presented by the position of the related set of genes in the chromosome. The schematic representation of the chromosomes are shown in Figure 3.7.

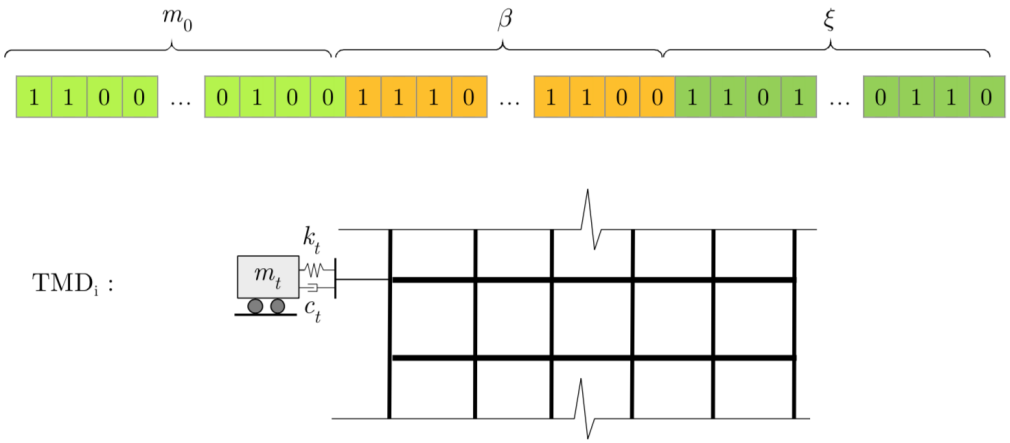


Figure 3.6: Binary coding the TMD's properties

3.2.8 Genetic Operators

a. Crossover function

The crossover operator exchanges random parts of two chromosomes to create two new off-springs. An appropriate strategy for selecting split points and the length of the transferred genes depends on the problem characteristics and affects the performance of the algorithm and the quality of the final results.

In this study, different variants have been tried to develop an appropriate crossover function. Examples of variants that have been commonly utilized in other researches but are not appropriate for this research are discussed as follows:

1. Single/k-point crossover - random points in the whole chromosome:

In this crossover type, one or k points in the parents' chromosome are selected randomly and the new off-springs will be then created by splitting and combining the parent chromosomes at the selected points. This process resulted in producing too many meaningless and low-quality off-springs which reduces the performance of the algorithm dramatically. Meaningless off-springs in this research mean TMDs without one or more than one properties (e.g. zero mass or stiffness).

2. Single/k-point crossover - random points in TMD genes:

Preventing the production of meaningless off-springs, the crossover function was improved in this study so that only a complete set of genes related to a particular parameter of a TMD in each parent could be selected for performing a k-point crossover. Although the chances of creating meaningless off-springs were noticeably reduced, the performance of the algorithm was still not acceptable. The results showed that the efficiency of the operator in improving the results was not acceptable, as following this process, all the genes within the considered range for a TMD would be subjected to the same operations regardless of the genes' positions.

As an example, the genes related to the stiffness of a TMD in a parent were exchanged with those related to the damping properties in another parent, which is not logical.

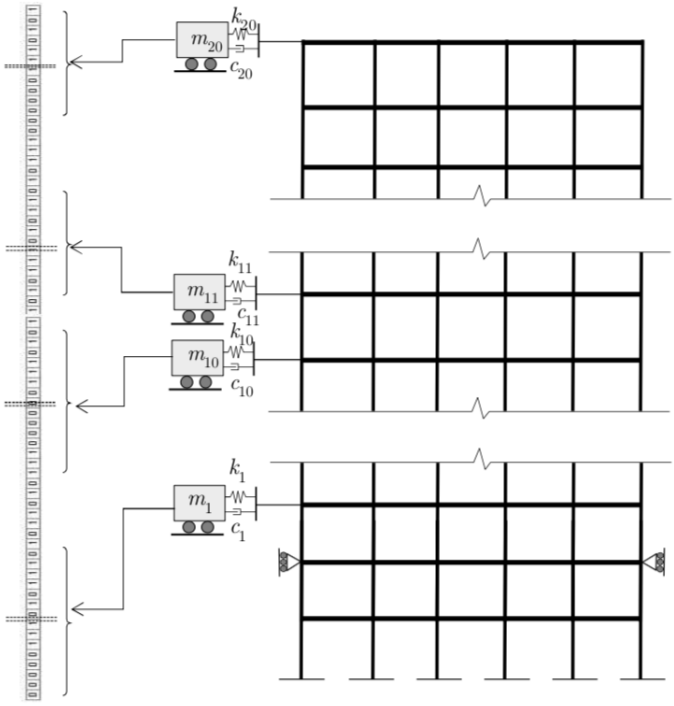


Figure 3.7: A schematic chromosome representation

As a result, despite its improvements compared to the first form, the second crossover-type leads to a very low convergence rate due to the production of low-quality offsprings. In addition, one possible shortcut for reaching an optimum solution was missed; this step involves attaching the TMD of one parent to a story in another parent.

Addressing these issues, a two-variation crossover function is developed (see Appendix A- Algorithm A.1). These two crossover variations are schematically demonstrated in Figure 3.8. As it is shown, in the first variant, the algorithm, exchanges the parameters in one or multiple TMDs in a parent with the corresponding values in another parent. In the second variation, the algorithm changes a TMD in a parent with a TMD in another parent.

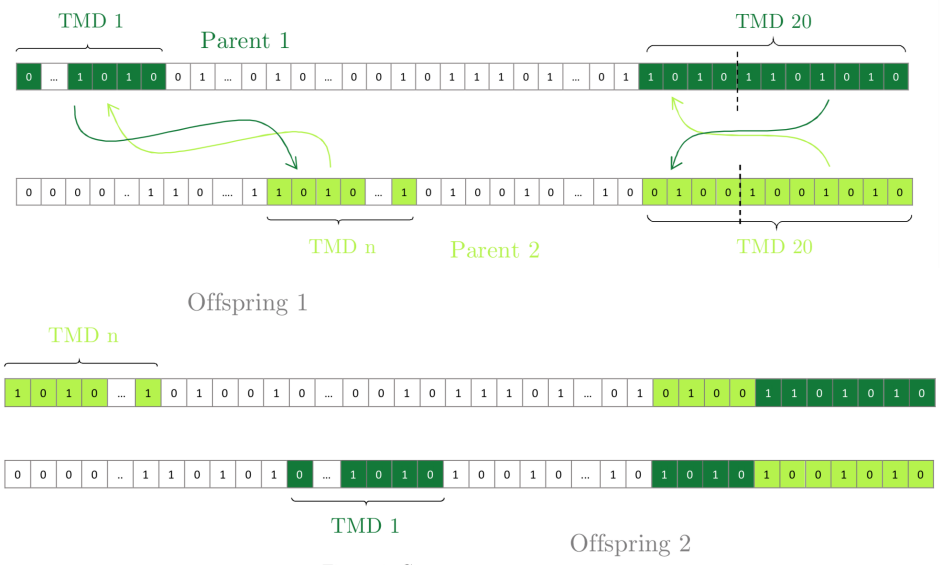


Figure 3.8: Crossover types

b.Mutation function

In the genetic algorithm, the mutation operator randomly changes one or multiple genes of a parent to produce new off-springs. Generally, in the binary-coded chromosome, the following function is utilized to change the genes:

$$BinaryMutation(gen) = \begin{cases} 1 & if\ gen\ value = 0 \\ 0 & if\ gen\ value = 1 \end{cases}$$

In GA problems, the mutation function helps the algorithm to explore the solution space more broadly and prevents it from sticking to local extremums. By studying different variants it was understood that developing the mutation function without considering the characteristics of the problem would result in producing meaningless off-springs. A two-variant mutation function was developed, which acted on the (1) genes related to TMD properties and (2) the group of genes related to the location of the TMDs (see Appendix A- Algorithm A.2).

3.2.9 Fitness Function

During the GA procedure, each solution comprises an arrangement of TMDs with different properties. In order to evaluate an individual solution, three objective functions are defined to shape the fitness function. The objective functions determine the maximum controlled to uncontrolled ratios of displacement, velocity, and acceleration responses:

$$J_1 = max. \left(\frac{u_i^C}{u_i^{UC}} \right)_{i=1,\dots,N} \tag{3.2}$$

$$J_2 = max. \left(\frac{v_i^C}{v_i^{UC}} \right)_{i=1,\dots,N} \tag{3.3}$$

$$J_3 = max. \left(\frac{a_i^C}{a_i^{UC}} \right)_{i=1,...,N} \tag{3.4}$$

where:

i:	Story number
N:	Number of stories in the tall building
u_i^{UC}:	Uncontrolled displacement response of story i
u_i^C:	Controlled displacement response of story i
v_i^{UC}:	Uncontrolled velocity response of story i
v_i^C:	Controlled velocity response of story i
a_i^{UC}:	Uncontrolled acceleration response of story i
a_i^C:	Controlled acceleration response of story i

3.3 Framework

3.3.1 Introduction

Based on the discussed methods, a framework is developed to study the optimal placement and properties of TMDs in a tall building. In this regard, the computer codes were developed in *Matlab* software. The inputs to the framework including :

1. Structural parameters: mass, stiffness, and damping matrices.
2. TMD parameters: mass, damping, and stiffness limits which shall be defined based on vendor documents.
3. Earthquake acceleration records.

The outputs of the framework include the optimal placement and properties of the TMDs and the evaluation results.

3.3.2 Basic Algorithm

The basic algorithm of the framework includes a major loop in which the generations are evolved. In the first iteration, the algorithm generates random chromosomes to form the initial population. after that, in each iteration, the algorithm adds new chromosomes by performing genetic operations on the current members and passes them to the *analysis* module to determine the controlled responses of the structure, equipped with the TMD system to the earthquake excitations. Next, the *fitness* module determines the fitness values for each chromosome and passes the population to the NSGA-II module which utilizes a refined history of the TMD arrangements and corresponding fitness values, to develop the paretos and sort the chromosomes

upon that. The pseudo-code of the program is presented in Appendix A- Algorithm A.3.

In order to improve the performance of the developed code and reduce the computation time, some advanced computer programming techniques, such as parallel computing, were utilized to produce the multiple populations in parallel.

3.4 Case Study

3.4.1 Introduction

As a case study, a 76-story, 306-meter tall concreted building is considered which was developed by Yang et al. [173]. The total mass of the building is 153,000 tonnes. The total volume of the building is 510,000 m^3, resulting in a mass density of 300 kg per cubic meter. The building is slender with a height to width ratio of $306.1/42 = 7.3$. The building has been designed to resist the wind loads. Considering its aspect ratio, it is a wind sensitive building. In the mathematical model of the building, the building is modeled as a vertical cantilever beam. A finite element model of the structure is created by considering the portion of the building between two adjacent floors as a classical beam element of a uniform thickness, leading to 76 translational and 76 rotational degrees of freedom. Then, all the 76 rotational DOFs have been removed by the static condensation. This results in 76 DOF, representing the displacement at floor levels in lateral direction. The first five natural frequencies are 0.16, 1.992, 3.79, and 6.35 Hz as schematically are shown in Figure 3.9. The proportional (76×76) damping matrix for the building with 76 lateral DOF is calculated using Rayleigh's approach and by considering a 1% damping ratio for the first five modes.

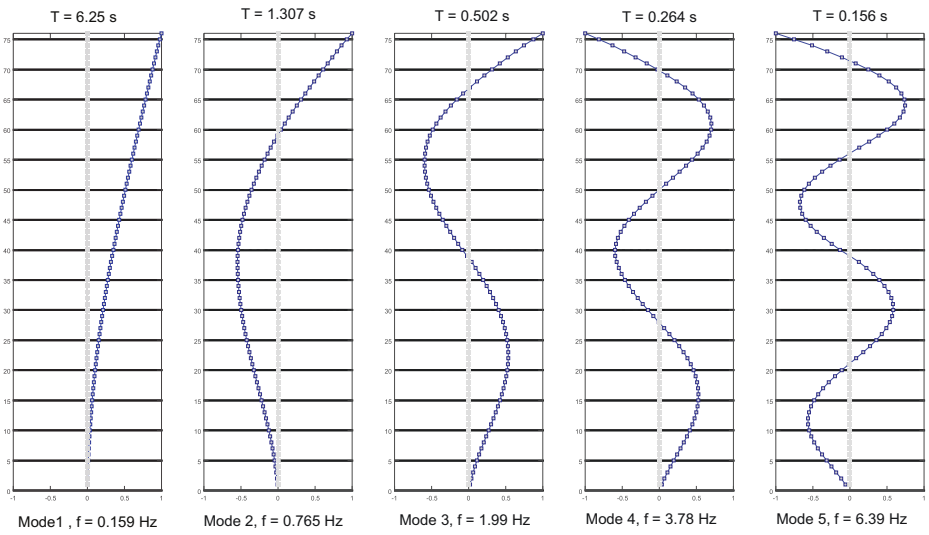


Figure 3.9: Five mode shapes of the 76-story building

3.4.2 Mathematical Model of Building

The governing equation of the motion for the tall building under earthquake excitation is as follow:

$$[M]\left\{\ddot{U}\right\} + [C]\left\{\dot{U}\right\} + [K]\left\{\dot{U}\right\} = \{P_t\} \tag{3.5}$$

where M, K and C represent mass, stiffness and damping matrices of the structure and the TMDs:

$$[M] = [M_{st}] + [M_t] \tag{3.6}$$

$$[C] = [C_{st}] + [C_t] \tag{3.7}$$

$$[K] = [K_{st}] + [K_t] \tag{3.8}$$

Indexes st and t indicate the DOFs of the building and the TMDs, respectively.

The external load vector, P_t in Eq. 3.5 comprises inertial forces due to ground acceleration as follows:

$$\{P_t\} = -\ddot{u}_g\,[M]\,\{1_t\} \tag{3.9}$$

where $\{1_t\}_{(N+n)\times 1} = [1\ 1\ \cdots\ 1]^T$ and the term $\ddot{u}_g$ represents the ground accelerations.

The structural responses, including displacement, velocity, and acceleration matrices are defined by:

$$\{U\} = \{u_{st1}, u_{st2}, \ldots, u_{stN}, u_{t1}, u_{t2}, \ldots, u_{tn}\} \tag{3.10}$$

$$\{V\} = \{v_{st1}, v_{st2}, \ldots, v_{stN}, v_{t1}, v_{t2}, \ldots, v_{tn}\} \tag{3.11}$$

$$\{A\} = \{a_{st1}, a_{st2}, \ldots, a_{stN}, a_{t1}, a_{t2}, \ldots, a_{tn}\} \tag{3.12}$$

In these equations, N and n represent the number of DOFs for the building and the TMDs respectively. Therefore, the dimensions of the M, K, and C matrices are $(N+n)\times(N+n)$.

The design parameters of a TMD include its damping, tuning frequency, and mass. Generally, the ratios of these parameters to the corresponding values of the building are considered as design parameters:

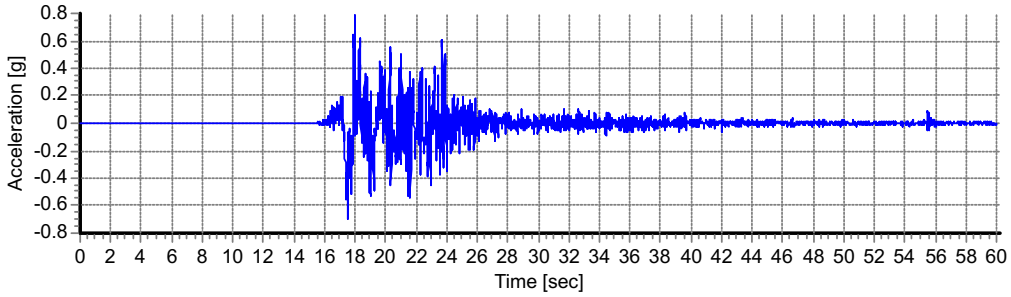
$$m_0 = \frac{m_t}{m_{st}}, \beta = \frac{\nu_t}{\nu_{st}}, \psi = \frac{c_t}{c_{st}} \tag{3.13}$$

Where m_0, β, ψ refer to mass, frequency, and damping ratios.

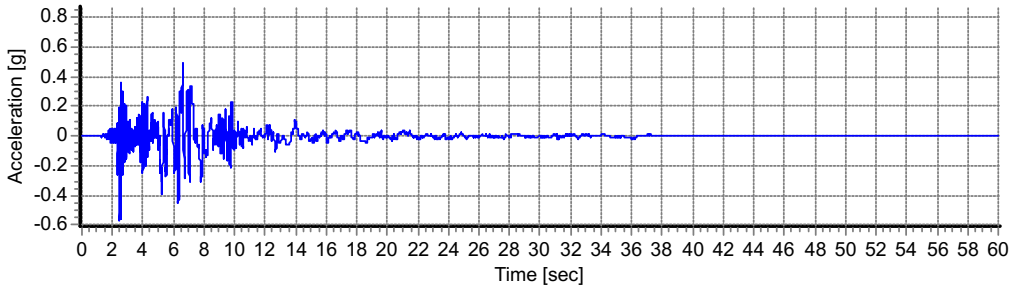
In order to solve the equations of the motion, Newmark's β method is utilized. Average acceleration method is considered by setting $\gamma = \frac{1}{2}$ and $\beta = \frac{1}{4}$ in the relevant formulation [174].

3.4.3 Ground Motion Selection

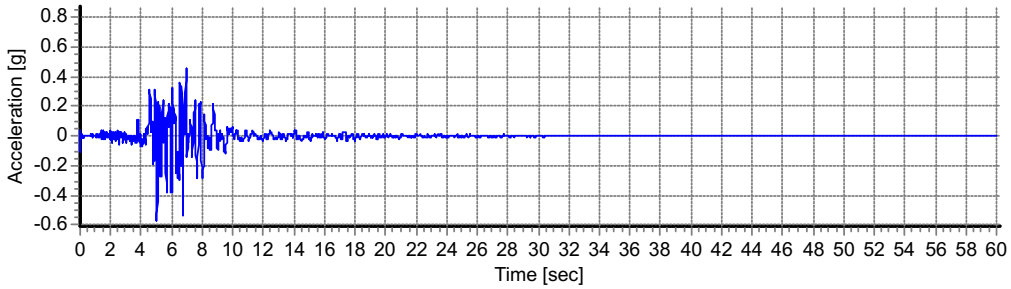
The ground motions are selected and scaled using an intensity-based assessment procedure according to ASCE/SEI 07-10 [175]. In this regard, seven earthquakes acceleration records were selected from the pacific earthquake engineering research (PEER) center, NGA strong motion database [6] and then scaled using a design response spectrum (see Figure 3.10). The seismic parameters and the considered design response spectrum are shown in Table 3.1.



(a) Bam, 2003- Iran

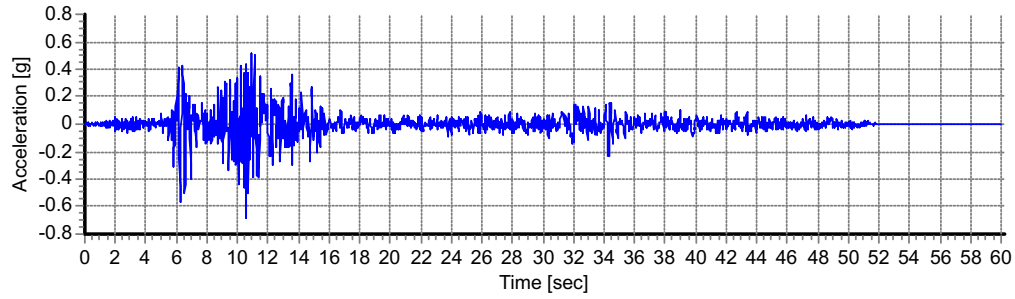


(b) Elcentro,1940- USA

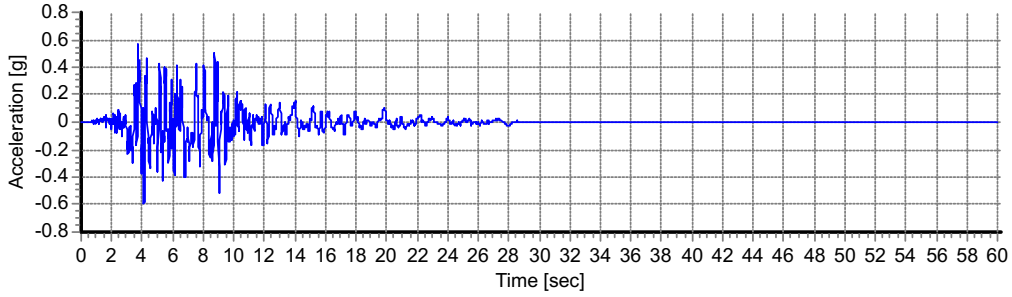


(c) Kobe, 1995- Japan

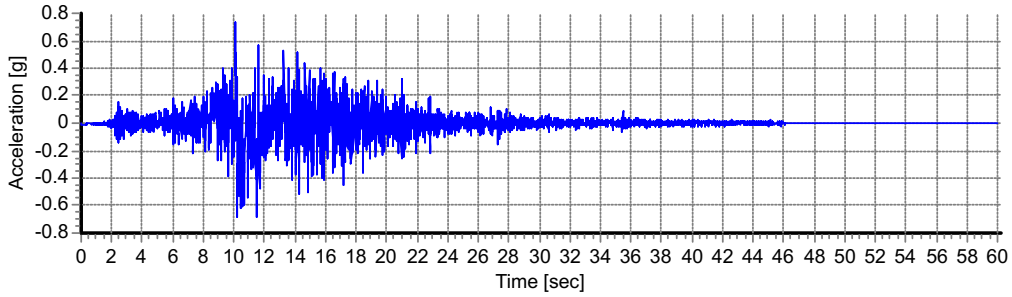
Figure 3.10: Earthquake acceleration records



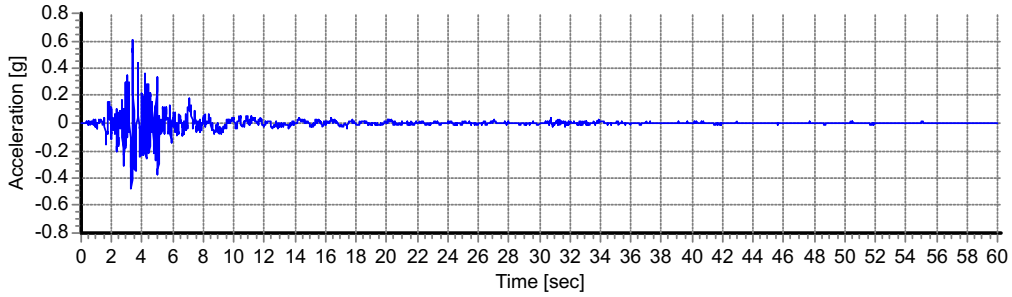
(c) Manjil, 2002, Iran



(d) Northridge, 1971- USA



(e) Landers, 1992-USA



(f) SanFernando, 1994-USA

Figure 3.10: Earthquake acceleration records obtained from the Pacific Earthquake Engineering Research (PEER) Center [6]

Accelerogram	Max. Acc.	Max. Vel.	Max. Dis.	Effective Design Acc.	Predominan Period (sec)	Significant Duration (sec)
1- Bam	0.80	124.12	33.94	0.69	0.20	8.00
2- Elcentro	0.44	67.01	27.89	0.30	0.06	11.46
3- Kobe	0.31	30.80	7.47	0.28	0.42	6.20
4- Manjil	0.51	42.45	14.87	0.47	0.16	28.66
5- Northridge	0.45	60.14	21.89	0.45	0.42	10.62
6-Landers	0.72	133.40	113.92	0.52	0.08	13.15
7- SanFernando	0.22	21.71	15.91	0.20	0.00	13.15

Table 3.2: Original earthquakes' specifications (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (g))

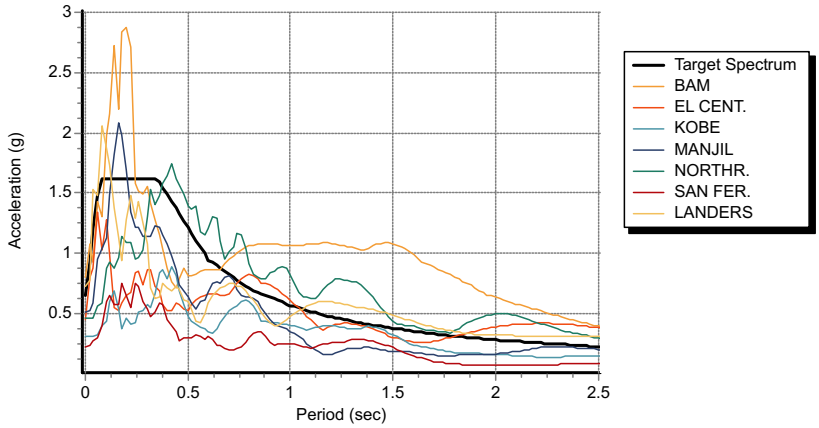
Accelerogram	Max. Acc.	Max. Vel.	Max. Dis.	Effective Design Acc.	Predominan Period (sec)	Significant Duration (sec)
1- Bam	0.78	128.55	33.95	0.64	0.20	8.24
2- Elcentro	0.57	75.76	28.36	0.48	0.32	9.16
3- Kobe	0.56	38.83	15.68	0.56	0.36	4.16
4- Manjil	0.68	42.06	15.02	0.55	0.08	28.26
5- Northridge	0.59	64.57	22.16	0.58	0.40	10.40
6-Landers	0.73	141.02	113.78	0.56	0.08	12.92
7- SanFernando	0.60	23.71	15.88	0.61	0.10	6.03

Table 3.3: Scaled earthquakes' specifications (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (g))

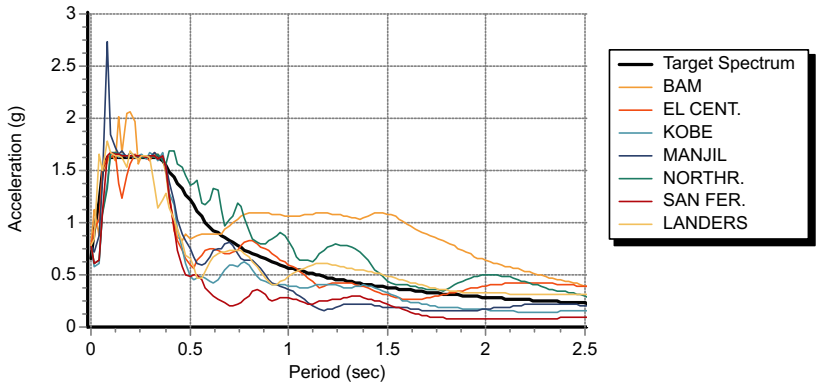
Site class	PGA	S_s	S_1	F_a	F_v	S_{MS}	S_{M1}	S_{DS}	S_{D1}
B	0.919	2.431 g	0.852 g	1	1	2.431 g	0.852 g	1.621 g	0.568 g

Table 3.1: Parameters of the design response spectrum

The specifications of the non-scaled and scaled selected earthquakes are presented in Tables 3.2 and 3.3. The spectrum of the non-scaled and scaled excitations are shown in Figure 3.11.



(a)



(b)

Figure 3.11: Response spectrum for a. unscaled and b. scaled earthquake records

3.4.4 TMD Parameters

The variation domain for m_0, β, and ψ are considered in an applicable range (see Table 3.4). The total mass ratio of TMDs, $m_t = \sum_{i=1}^{n} m_{0i}$, is limited to 3%, which is equal to the considered limit for each TMD. Therefore, the algorithm can either spend the allowable masses in a single TMD or divide it among multiple TMDs and distribute them over the height of the building.

Parameter	min. value	max. value
m_t	-	3%
m_0	0.2%	3%
β	0.8	1.3
ψ	5	40

Table 3.4: Parameter variation domain for TMDs

3.4.5 Sensitivity Analysis of GA Parameters

As the parameters of GA are highly dependent on the characteristics of each particular problem [169], a sensitivity analysis of the parameters was performed, and the optimum values were studied. As is shown in Table 3.5, during the sensitivity analysis, the crossover and mutation probabilities were iterated, and the objectives were compared for El Centro earthquake excitation.

The results then were sorted using the NSGA-II sorting algorithm and the pareto fronts were obtained as shown in Table 3.6. The optimum values for crossover and mutation probabilities were obtained as 0.7 and 0.2 respectively. The utilization of the obtained values for the GA parameters resulted in improving the quality of the solutions as well as the performance of the algorithm.

Variation No.	Crossover	Mutation	J1	J2	J3
1	0.6	0.1	0.845	0.917	0.941
2		0.2	0.860	0.926	0.950
3		0.3	0.855	0.924	0.948
4		0.4	0.839	0.915	0.942
5	**0.7**	0.1	0.862	0.925	0.946
6		**0.2**	**0.835**	**0.913**	**0.939**
7		0.3	0.861	0.925	0.947
8		0.4	0.846	0.919	0.944
9	0.8	0.1	0.869	0.929	0.950
10		0.2	0.876	0.932	0.952
11		0.3	0.842	0.915	0.941
12		0.4	0.852	0.922	0.947
13	0.9	0.1	0.871	0.930	0.951
14		0.2	0.866	0.927	0.947
15		0.3	0.874	0.932	0.953
16		0.4	0.839	0.913	0.939

Table 3.5: Crossover and Mutation variations

Pareto Front	Variation No.	Variation No.
1	**6**	-
2	16	-
3	4	11
4	1	-
5	8	-
6	5	12
7	3	7
8	2	14
9	9	-
10	13	-
11	10	15

Table 3.6: NSGA of Crossover and Mutation variations

As a result, the parameters of the NSGA-II algorithm were considered as they are shown in Table 3.7. The sufficiency of 500 generations as the limit for the number of generations was then evaluated, as shown in Figure 3.12.

Number of Generations	Population size	Crossover probability	Mutation probability
500	100	0.7	0.2

Table 3.7: NSGA II algorithm initial parameters

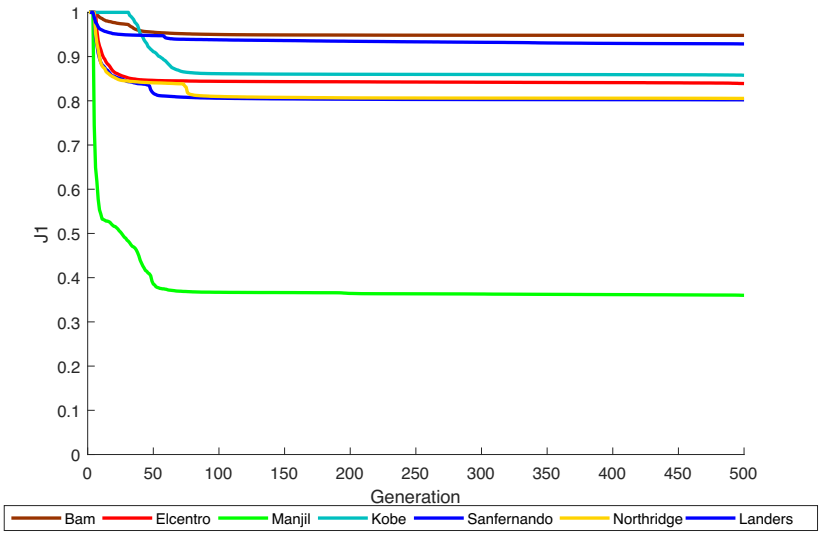


Figure 3.12: Trend of objective J_1 during the generations

3.5 Results

After preparing the inputs including the structural parameters as well as earthquake excitation records, the framework has been utilized for determining the optimum arrangement and parameters of TMDs.

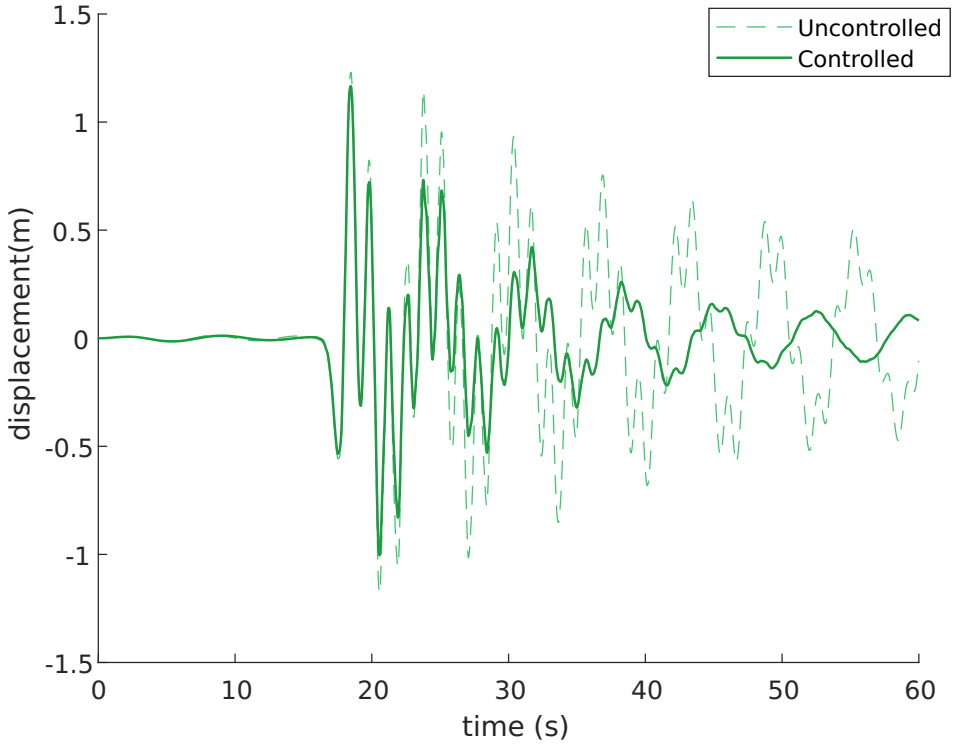
The results are presented in Figures 3.13- 3.19. As it is shown, the optimum number of TMDs is more than one for some of the excitations. The maximum number of TMDs is obtained as three for the Bam and Manjil excitations and the TMDs are placed in the 76, 75, and 64th levels in both cases. The tuning frequency of the TMDs was close to the fundamental frequency of the building for the top two TMDs and about 1.24 times the fundamental frequency for the TMD in the 64th story. The damping ratio of the TMDs placed on the top two stories was close to the maximum allowed value, while it was about 14 for the TMD in the 64th story.

It was observed that the controlled displacement responses of the building improved substantially by about 65% under the Manjil earthquake excitation. On the contrary, objective J_1 for Bam earthquake had a value of about 0.95, which translated to about 5% improvement in reducing maximum displacements, compared to the uncontrolled response. This low objective value was also obtained under the Landers earthquake, with a value of about 0.93 for J_1, implying 7% improvement in reducing

the displacement responses. However, the displacement responses show substantial improvements in damping further oscillations compared to uncontrolled buildings for both cases.

For Landers, Northridge, and San Fernando earthquakes, the TMDs were placed on 76 and 74th levels. Most of the allowed mass was dedicated to the TMD on the roof levels. The optimum tuning frequency of the TMDs was close to the fundamental frequency of the building. The optimum damping values of the TMDs were obtained between 36.39 to 39.78, which were close to the maximum considered damping ratio.

For the El Centro and Kobe earthquakes, the optimum results were obtained by placing a single TMD system on the roof. In both cases, all the allowed mass was dedicated to the TMD. Under these excitations, the frequency ratios of the TMDs were obtained as 1.06 and 1.05 which indicates a tuning frequency close to the fundamental frequency of the building.



(a) Roof displacement response

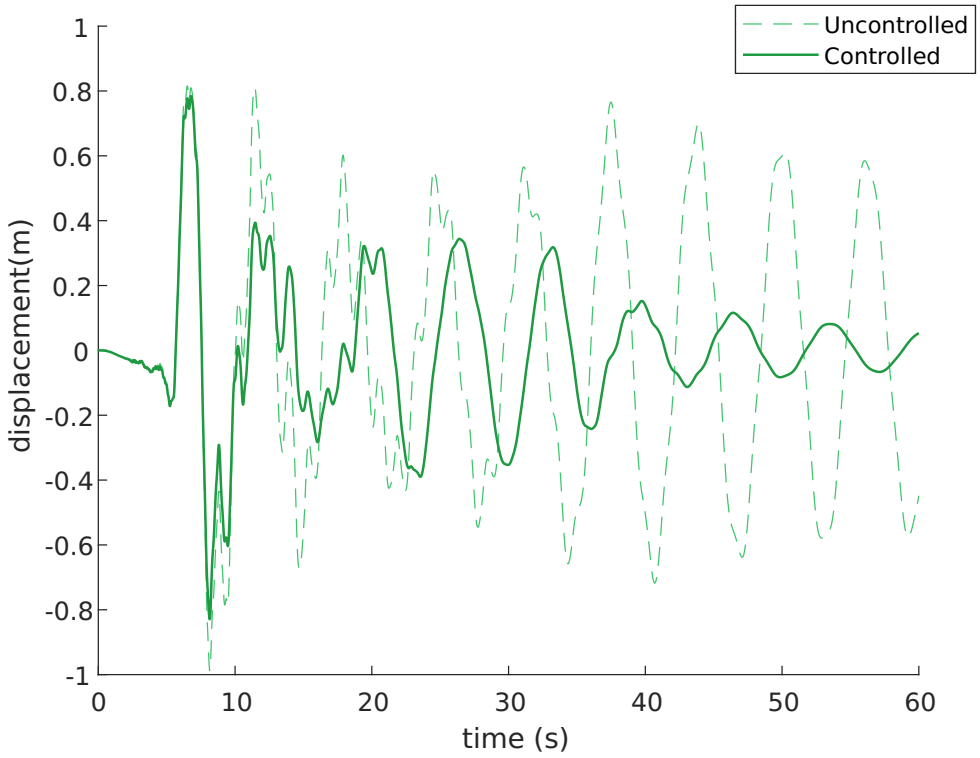
Story	m_0	β	ψ
76	1,43	0.98	39,53
75	0.93	1.01	39,46
64	0,64	1.24	18.37

(b) TMDs specifications

J1	J2	J3
0,95	0,88	0,83

(c) Objective values

Figure 3.13: Uncontrolled/controlled responses- Bam earthquake



(a) Roof displacement response

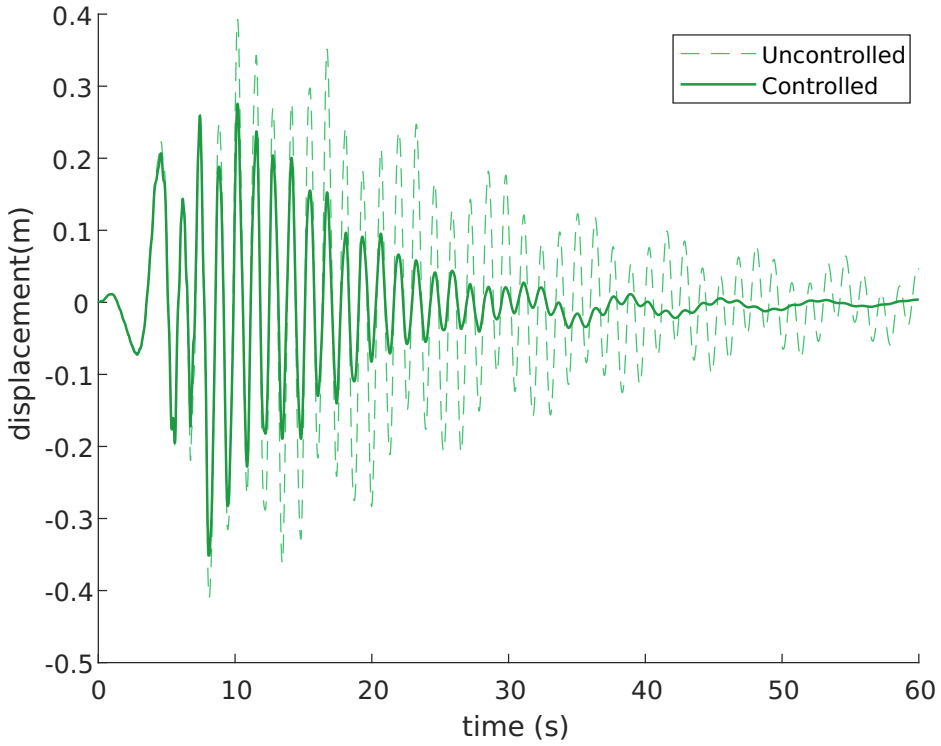
Story	m_0	β	ψ
76	2,99	1,06	39,53

(b) TMDs specifications

J1	J2	J3
0,84	0,91	0,93

(c) Objective values

Figure 3.14: Uncontrolled/controlled responses- Elcentro earthquake



(a) Roof displacement response

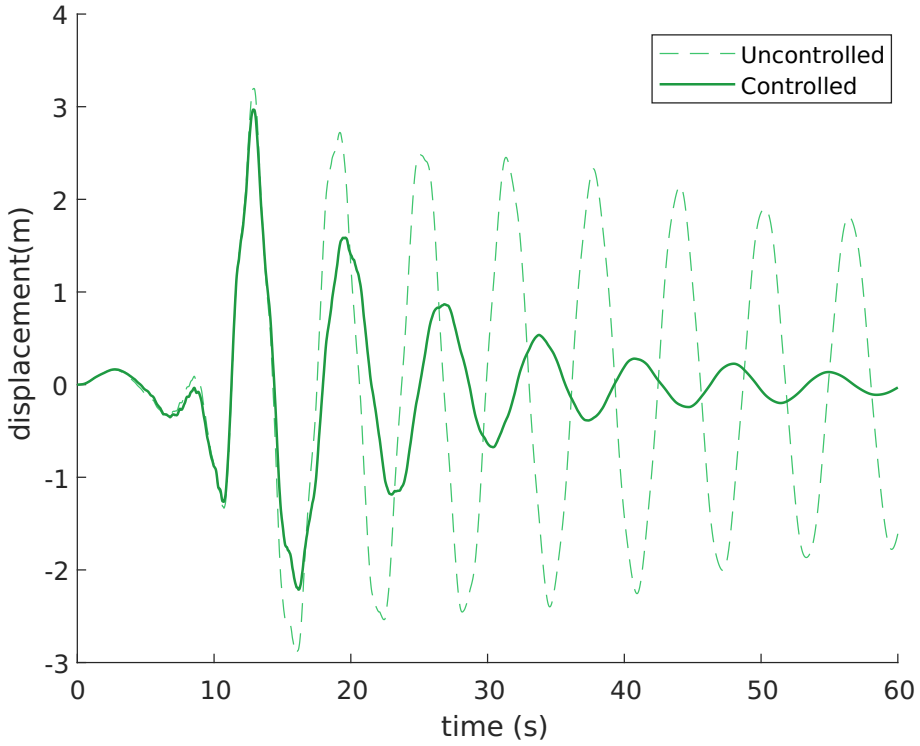
Story	m_0	β	ψ
76	3,00	1,05	39,78

(b) TMDs specifications

J1	J2	J3
0,86	0,90	0,97

(c) Objective values

Figure 3.15: Uncontrolled/controlled responses- Kobe earthquake



(a) Roof displacement response

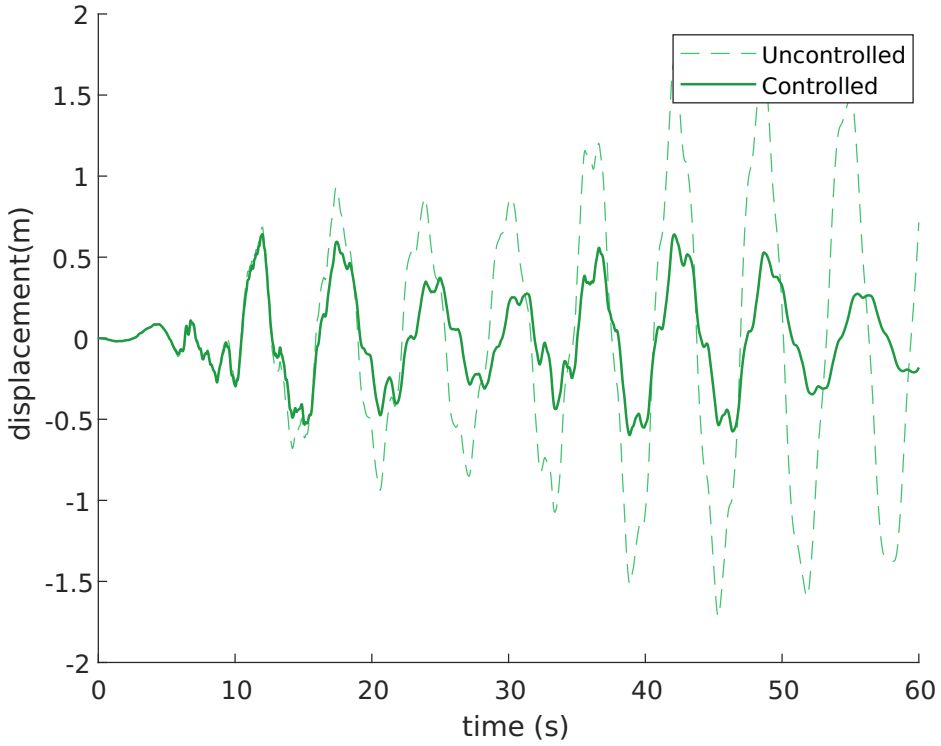
Story	m_0	β	ψ
76	2,01	1,01	37,07
74	1,00	1,01	37,13

(b) TMDs specifications

J1	J2	J3
0,93	0,89	0,98

(c) Objective values

Figure 3.16: Uncontrolled/controlled responses- Landers earthquake



(a) Roof displacement response

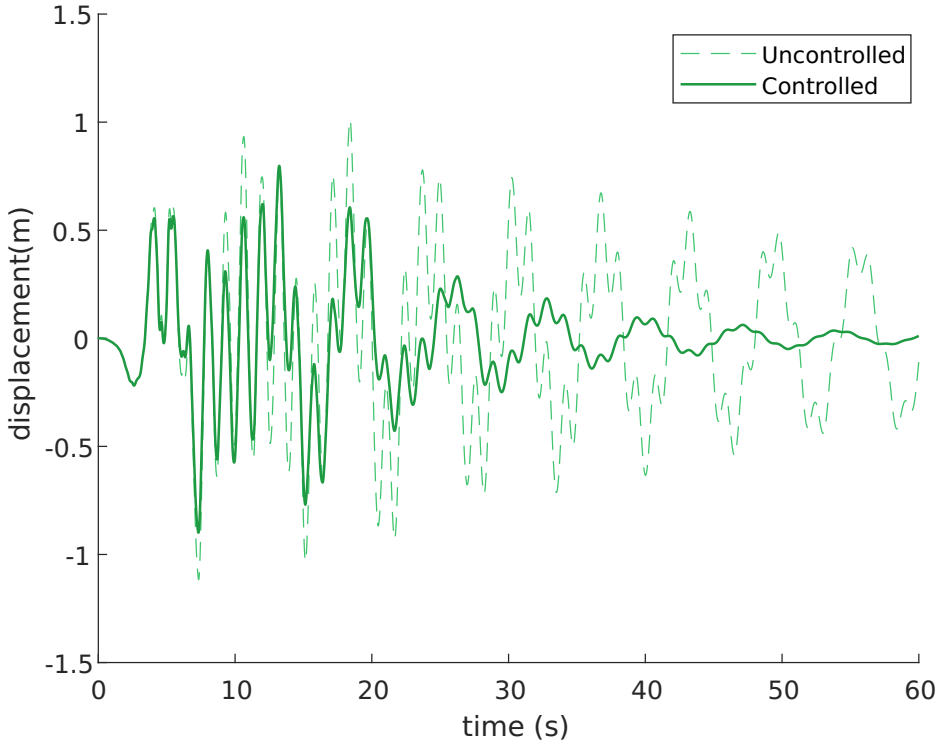
Story	m_0	β	ψ
76	1,50	1,02	35,12
75	0.95	1,01	31,34
64	0,55	1.21	14,92

(b) TMDs specifications

J1	J2	J3
0,36	0,57	0,95

(c) Objective values

Figure 3.17: Uncontrolled/controlled responses- Manjil earthquake



(a) Roof displacement response

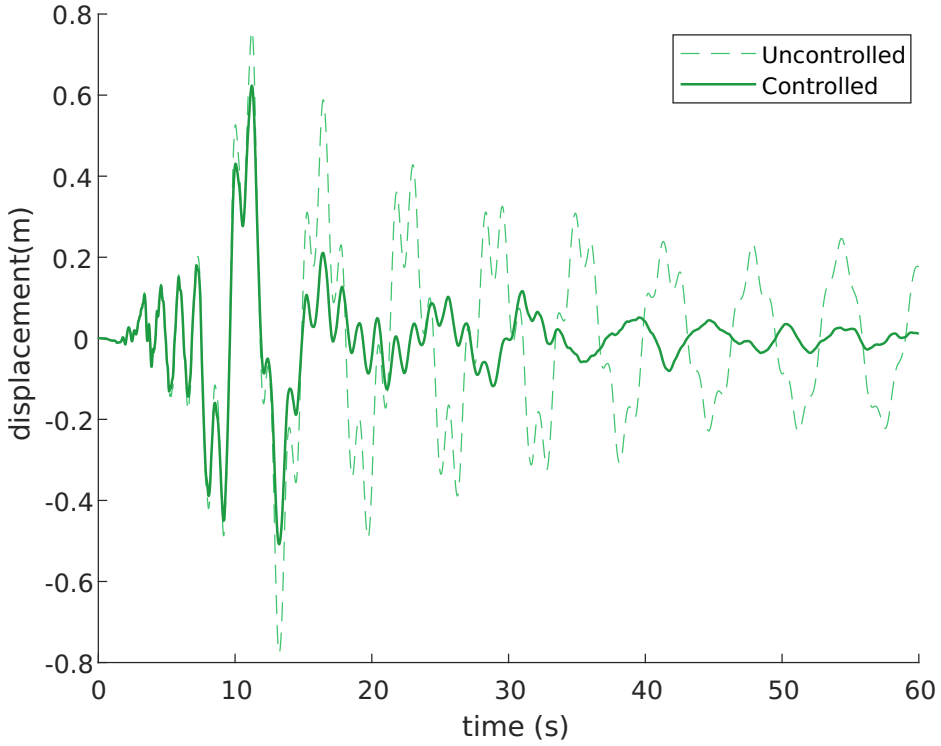
Story	m_0	β	ψ
76	2,26	1,04	37,05
74	0,75	1,04	36,98

(b) TMDs specifications

J1	J2	J3
0.81	0.84	0.95

(c) Objective values

Figure 3.18: Uncontrolled/controlled responses- Northridge earthquake



(a) Roof displacement response

Story	m_0	β	ψ
76	2,28	1,03	36,27
74	0,72	1,03	36,39

(b) TMDs specifications

J1	J2	J3
0.80	0.84	0.97

(c) Objective values

Figure 3.19: Uncontrolled/controlled responses- SanFernando earthquake

3.6 Discussion

As mentioned in the results section, the optimum target stories for placing the TMDs was included the top two levels for all earthquake excitations, and some other levels such as 74th and 64th levels for some earthquakes. To understand the reasons behind such optimum arrangement, the buildings' mode shapes are again presented in Figure 3.20 and the stories with maximum displacements are marked. As shown, for the first three modes, the top three levels have the maximum displacements. For the 4th mode, the roof and the 75th and 61st levels, and for the 5th mode, the roof, and the 75th and 64th levels were the stories with maximum modal displacements.

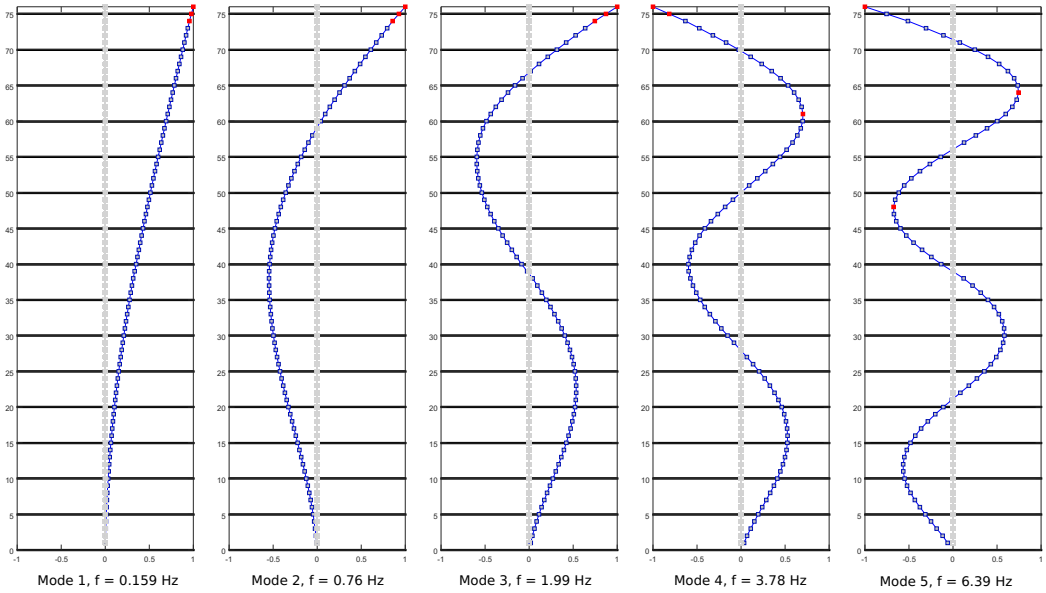
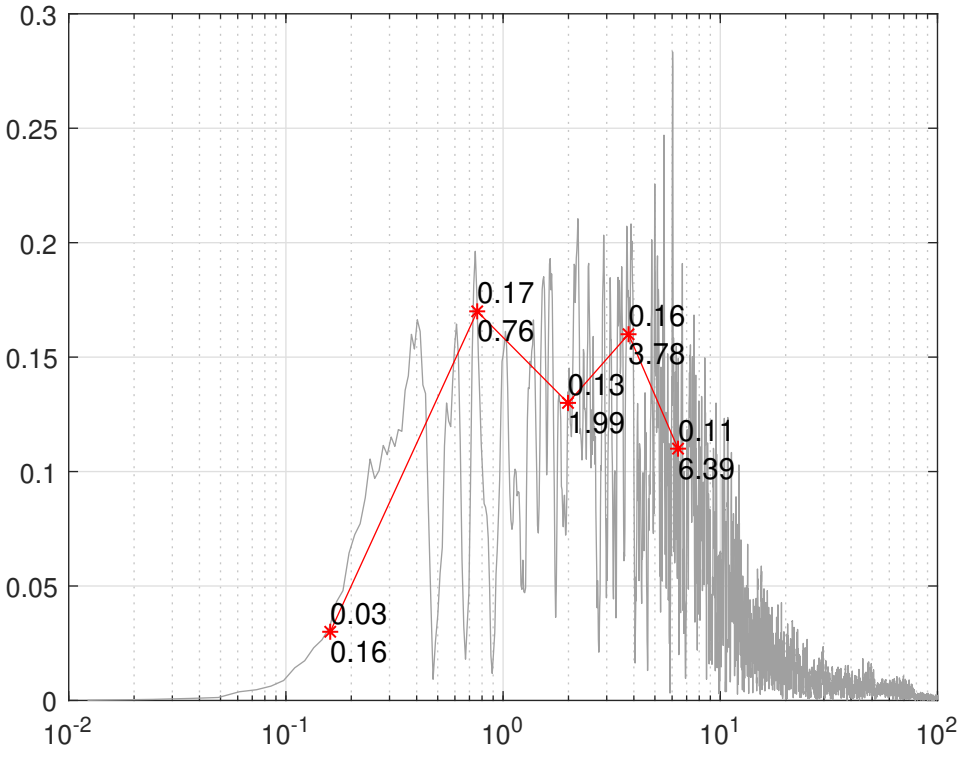


Figure 3.20: Three stories with maximum modal displacements in first five natural modes

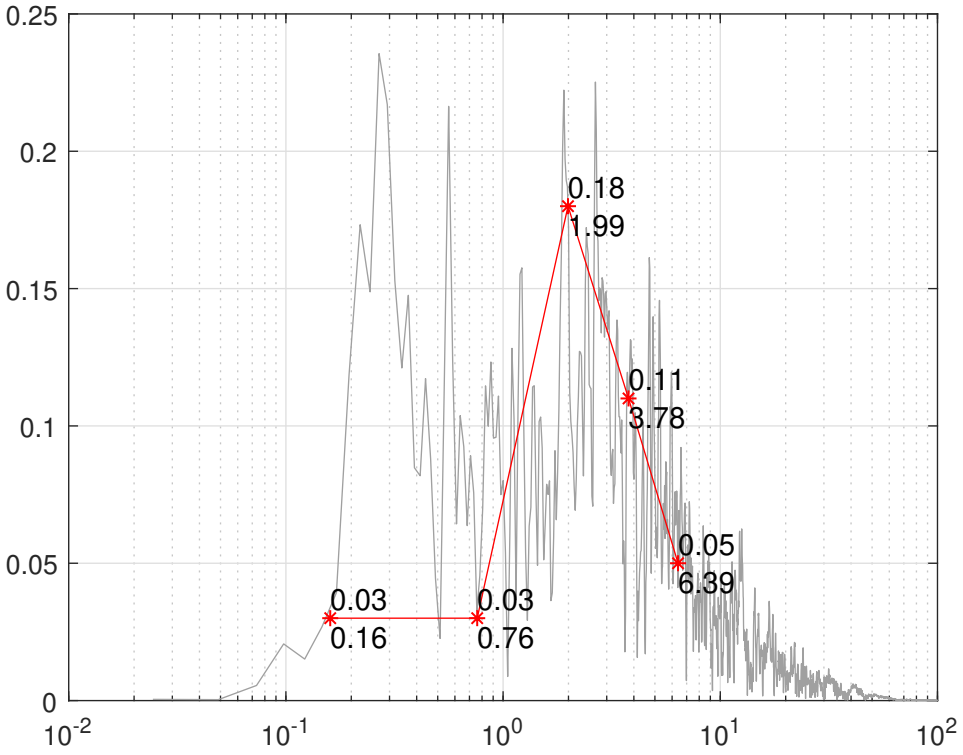
Consequently, it can be concluded that placing a TMD on the top stories would improve the modal displacements in all five modes which agrees with the optimization results.

On the other hand, for some earthquakes, TMDs were placed in the lower stories, which implies that the optimum placement of the TMDs may also be related to some excitation properties. For this reason, Fourier transformation was performed for each earthquake's excitation records, and the amplitudes related to building's mode frequencies were obtained as shown in Figure 3.21.

As shown here, unlike other earthquakes, for the Bam and Manjil earthquakes, the amplitudes of the excitation in the 4th and 5th modes are more than those for the lower modes. As a result, although these higher modes have lower mass participation factors, their participation in the total response of the earthquake is increased by higher excitation amplitudes.

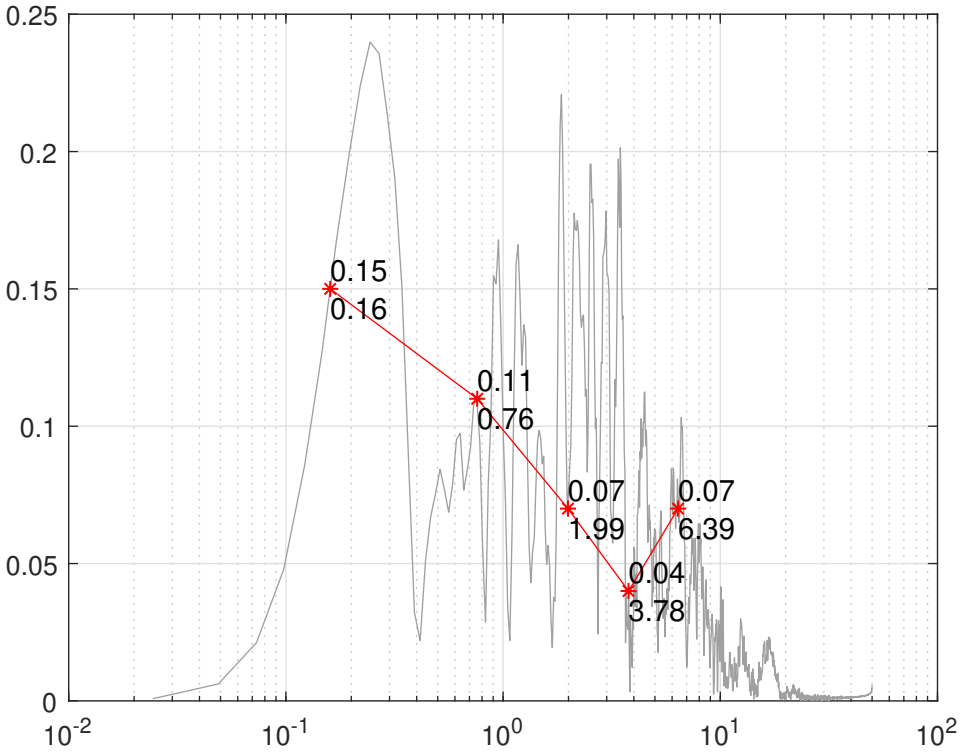


(a) Bam

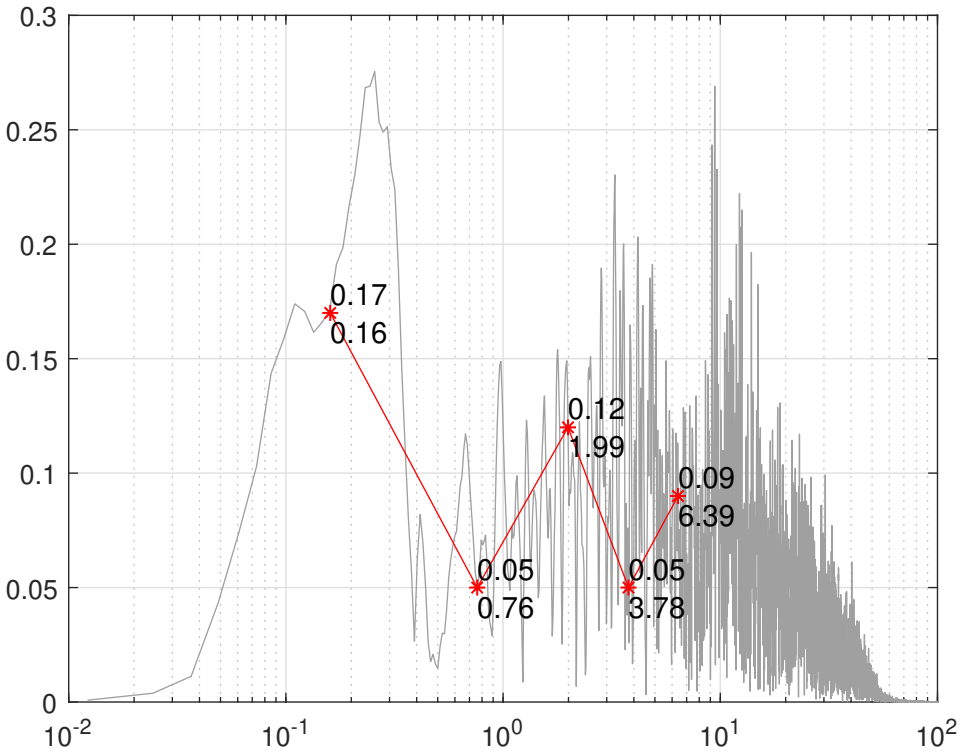


(b) Elcentro

Figure 3.21: Fourier transform of excitations

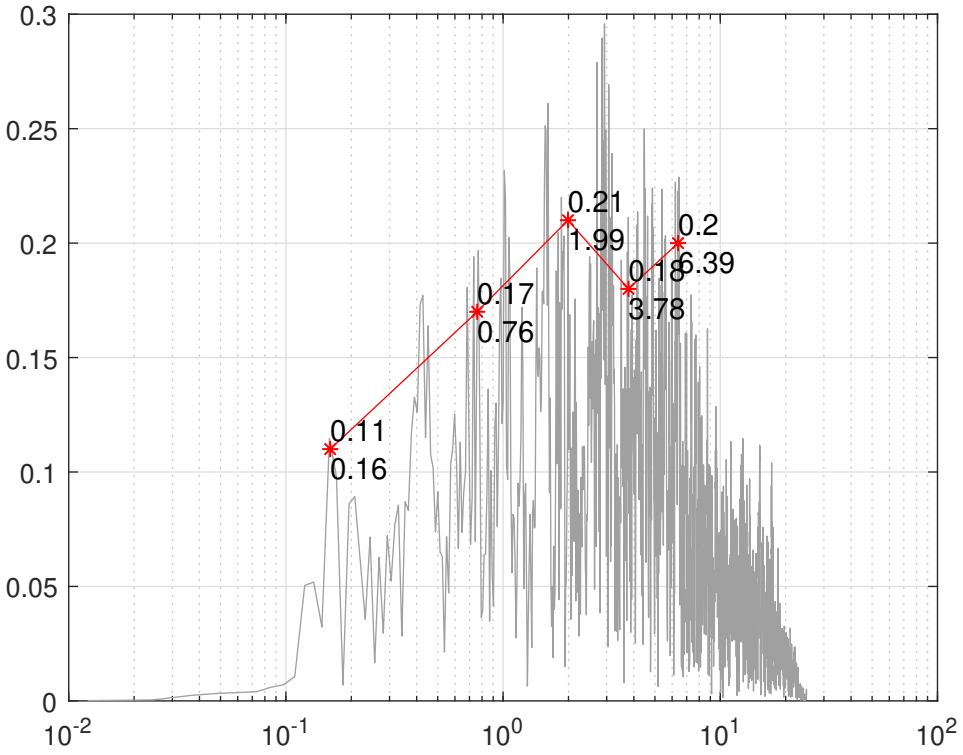


(d) Kobe

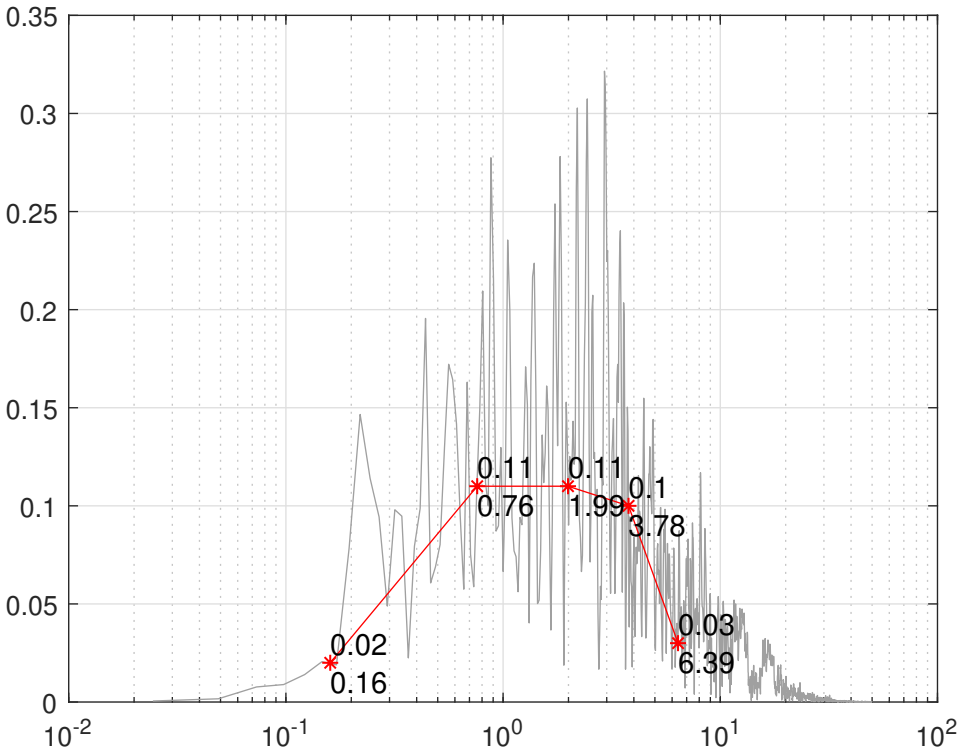


(e) Landers

Figure 3.21: Fourier transform of excitations

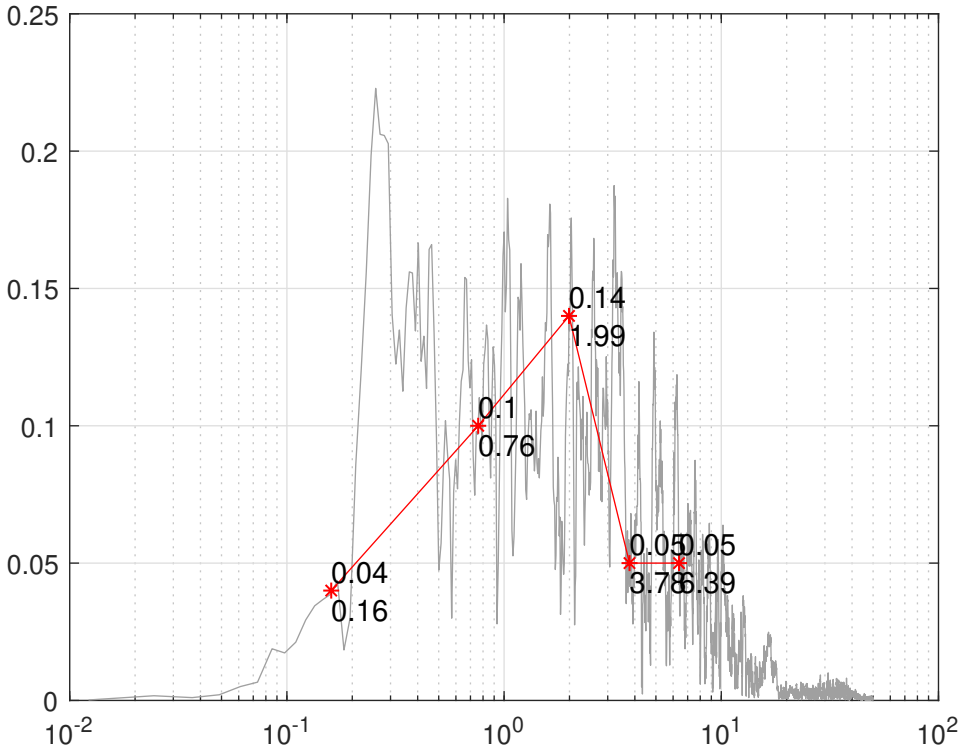


(f) Manjil



(g) Northridge

Figure 3.21: Fourier transform of excitations



(h) San Fernando

Figure 3.21: Fourier transform of excitations

To theoretically investigate the obtained results, the displacement response of the building under the ground motion is presented in Equation 3.14 as sum of the modal nodal displacements:

$$\mathbf{u}(t) = \sum_{n=1}^{N} \mathbf{u}_n(t) \tag{3.14}$$

where $\mathbf{u}_n$ represents the mode nth displacements. The contribution of the nth mode to the nodal displacement $\mathbf{u}$(t) is

$$\mathbf{u_n}(t) = \phi_n q_n(t) \tag{3.15}$$

where q_n refers to the modal coordinate which can be calculated from following equation:

$$\ddot{q_n} + 2\zeta_n \omega_n \dot{q}_n + \omega_n^2 q_n = -\Gamma_n \ddot{u}_g(t) \tag{3.16}$$

Γ_n is the modal participation factor of the nth mode and is the degree to which the nth mode participates in the total response. The modal participation factor can be calculated based on the modal displacements and masses as follows:

$$\Gamma_n = \frac{\{\phi_n\}[M]\{1\}}{\{\phi_n\}^T[M]\{\phi_n\}} = \frac{\sum_{j=1}^{N} m_j \phi_{jn}}{\sum_{j=1}^{N} m_j \phi_{jn}^2} \tag{3.17}$$

Equation 3.16 is related to a Single Degree of Freedom (SDOF) system with frequency and damping corresponding to the nth mode.

As shown here, in each mode n, the nodal displacement $\mathbf{u}_n(\mathrm{t})$ has a direct relationship with the modal displacements, ϕ_n, which means that the stories with maximum modal displacements would have greater participation in the building's modal response. In addition, it is obvious that the modal responses in the nth mode is also related to the frequency content of the earthquake excitation, $\ddot{u}_g$, which means that a larger acceleration amplitude at that mode's frequency would result in lager modal responses.

Therefore, the participation of a particular story in the total response of the building would be more than the other stories if:

1. The story has maximum displacement in the modes with the larger modal participation factor.
2. The story has maximum displacement in the nth mode with the lower participation factor, but the ground motion has a larger Fourier transformation amplitude in the nth mode's frequency, f_n.

These derivations validate the possibility of placing the TMDs in stories other than the top stories which agree with the obtained results.

3.7 Conclusion

Based on the obtained results and the following conclusions are drawn:

1. Under earthquakes excitations with noticeable amplitude at structure's higher mode frequencies, Compared to a single TMD on the roof level, a distributed multiple- TMDs system is more efficient in improving structural responses with the same mass.
2. The optimum placement of the TMDs include:

- The stories with maximum modal displacements in the lower structural modes.
- When earthquake has a high amplitude in the higher modes of the structure, the stories corresponding to the maximum modal displacement in that modes.

3. The optimum parameters for the TMDs that control the vibrations in the lower modes include the maximum allowed damping ratio. This indicates that increasing the damping ratio would improve the performance of such TMDs.

4. The results show that the performance of the TMDs is not good in reducing the initial maximums in displacement responses compared to their reduction of the later maximums after some initial oscillations.

5. Even in the cases with immediate maximum displacement responses, the multiple distributed TMD system significantly improves the damping.

Chapter 4

Reinforcement Learning

4.1 Introduction

The goal of the second part of this research is to develop a framework for intelligent control of smart structures. In this control scheme, the computer develops the optimum structural control policy through an automated process. In this regard, this chapter reviews the basic machine learning theories, utilized in developing the framework. First, an overview of the machine learning methods is presented and then, reinforcement learning theories are studied. Readers seeking more detail are encouraged to read fundamental books on the topic [176, 177].

4.2 Machine Learning Methods

The machine learning is a collection of the methods and statistical model trough which, the computer learns how to progressively improve its performance in doing a specific task [178]. The machine learning methods can be categorized into three types:

1. Supervised learning

In supervised learning, the algorithm builds a mathematical model that correlates inputs and some desired outputs. These methods are widely utilized in classification and regression problems [19].

2. Unsupervised learning

In unsupervised learning, the algorithm inputs data and finds structures in the data. The related methods are suitable for clustering and grouping problems. A central application for unsupervised-learning methods is the density estimations of the data in statistics [179].

3. Reinforcement learning

Reinforcement learning (RL) is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some

notion of cumulative reward. These methods are appropriate for problems in which the agent can affect the environment. In RL, the environment is typically formulated as a Markov decision process (MDP). The agent senses the state of the environment to some extent, takes actions that affect the environment, and evaluates the taken action by observing the resultant reward. The agent shall have a goal or goals related to the state of the environment [176]. During last years, several researches have been done on advancing the methods in RL [180, 181, 182, 183, 162, 163, 184].

Considering the characteristics of the structural control problem this thesis employs RL methods to formulate a framework for developing an intelligent controller system. In this section, the theories of RL have been presented and then the problem mapping, from a structural control problem to a machine learning problem, is expressed. Then the method is developed based on the characteristics of the problem. Finally, the case studies are presented and the conclusions are drawn.

4.3 Elements of Reinforcement Learning

The key elements of an RL problem including:

1. **Agent**

In RL the goal of an *agent* is to learn to do a task so that that it maximizes the sum of some *reward* value. This is done by interacting with the *environmen*t and affecting it, observing the feedback, and improving the policy.

The two main types of RL methods including model-based and model-free methods. In model-based methods, the agent has information about the environment dynamics which means that the agent knows how the environment works. In model-free methods, the agent knows nothing about the environment dynamics. In a structural control problem, the control algorithm plays the agent role and the actions are the control signals and it is considered that the controller has no information about the structural dynamics which is more realistic.

2. **Environment**

The *Environment* is a dynamic system that can be affected by the agent. From the agent's point of view, anything out of the agent is a part of the environment. The environment can be observable or partially observable to the agent. The dynamics of the environment determine its changes due to the agent's actions. In this research, a mathematical model of the structure is a part of the environment.

3. **Reward**

The *reward* is a numerical signal which defines the goal in the reinforcement learning problem and the *reward function* determines how much the taken actions were good or bad. As an example, in this research, as reducing the displacement responses of the system is a goal for the controller, obtaining zero structural responses by the agent (intelligent controller) would result in getting the maximum reward.

4. **State**

The *state* is an abstraction of the environment. The definition of the state is a critical task in RL problems as the agent learns to takes actions based on the current state. As an example, the current structural responses and the external excitations are considered as part of the state as discussed in Section 5.2.1.

5. **Policy**

The *policy* determines how the agent's behavior. Therefore, the policy function maps the states into actions. In this research, the policy function determines the optimum control signals, given the current state.

4.3.1 Markov Decision Process

Markov decision processes (MDPs) formally describes the task in an RL problem. Generally, An MDP is defined by $\{\mathcal{S}, \mathcal{A}, \mathcal{P}^{\mathrm{a}}_{\mathrm{ss}'}, \mathcal{R}^{\mathrm{a}}_{\mathrm{ss}'}, \gamma\}$ where $\mathcal{S}$, is the set of states, $\mathcal{A}$ is a set of actions, $\mathcal{P}^{\mathrm{a}}_{\mathrm{ss}'}$,is the probability of getting into state s' by taking action a in the state s, $\mathcal{R}^{\mathrm{a}}_{\mathrm{ss}'}$ is the corresponding reward, and $\gamma \in [0, 1]$ is a discount factor which adjusts the participation of the future reward in determining the current reward. In MDPs, the current state includes all the required information for determining the next action. It means that "the future is independent of the past given the present". In the formulation

A state S_t is Markov if and only if:

$$\mathbb{P}^{\mathrm{a}}\left[S_{t+1} \mid S_t\right] = \mathbb{P}^{\mathrm{a}}\left[S_{t+1} \mid S_1, S_2, \ldots, S_t\right] \tag{4.1}$$

The dynamics in MDPs would be represented by the *transition probability* matrix which correlates state s to a successor s',

$$P^{a}_{ss'} = \mathbb{P}^{\mathtt{a}}\left[S_{t+1} = s' \mid S_t = s\right] \tag{4.2}$$

In *model-free* methods such as Q-learning, such a probability matrix is not available to the agent.

The policy function, π, represents the probability of taking an action a in state s,

$$\pi(a/s) = \mathbb{P}[A_t{=}\mathrm{a}|\ S_t{=}\mathrm{s}] \tag{4.3}$$

4.3.2 Markov Reward Process

For a particular policy, π, in an MDP, the sequence of rewards and states forms Markov reward process in which $\mathcal{P}^{\pi}$ and $\mathcal{R}^{\pi}$ represents probability and reward matrices,

$$\mathcal{P}^{\pi}_{\mathrm{s,s}'} = \sum_{\mathrm{a}\epsilon\mathcal{A}} \pi\left(\mathrm{a} \mid \mathrm{s}\right) \mathcal{P}^{\mathrm{a}}_{\mathrm{ss}'} \tag{4.4}$$

$$\mathcal{R}^{\pi}_{s} = \sum_{\mathrm{a}\epsilon A} \pi(\mathrm{a} \mid \mathrm{s}) \mathcal{R}^{\mathrm{a}}_{s} \tag{4.5}$$

The *return,* G_t, is the sum of the discounted rewards in future starting from the time-step t,

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ... = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{4.6}$$

The state value-function $v_\pi(s)$ of an MDP is the expected return, starting state s, and following policy π,

$$v^\pi(s) = \mathbb{E}_\pi[G_t \mid S_t = s] = \mathbb{E}_\pi[R_{t+1} + \gamma v^\pi(S_{t+1}) \mid S_t = s] \tag{4.7}$$

For a particular policy π, the value of the action a in the state s is called *action-value* and would be represented by q^π(s,a),

$$q^\pi(\mathrm{s,a}) = \mathbb{E}_\pi[R_{t+1} + \gamma q^\pi(S_{t+1}, \pi(s_{t+1})) \mid S_t = s] \tag{4.8}$$

4.3.3 Bellman Expectation Equation

The Bellman equation correlates the state value function, the immediate reward and the state we end up together.

$$v(s) = \mathbb{E}[R_{t+1} + \gamma v(S_{t+1}) \mid S_t = S] \tag{4.9}$$

Using $\mathcal{R}_s$ and can be written as:

$$v(s) = \mathcal{R}_s + \gamma \sum_{s' \epsilon S} \mathcal{P}_{\mathrm{ss}'} \mathrm{v(s')} \tag{4.10}$$

Bellman equation can be expressed as:

$$v = \mathcal{R} + \gamma \mathcal{P} v \tag{4.11}$$

Considering the policy π, the *Bellman expectation equation* which correlates the value of the state and the value of the successor can be expressed as:

$$v^\pi(s) = \sum_{\mathrm{a} \epsilon A} \pi(\mathrm{a} \mid \mathrm{s}) \left(\mathcal{R}_s^{\mathrm{a}} + \gamma \sum_{s' \epsilon S} \mathcal{P}_{\mathrm{ss}'}^{\mathrm{a}} v^\pi(s') \right) \tag{4.12}$$

Similarly, the action-value function for policy π can be expressed as follows:

$$q^\pi(\mathrm{s,a}) = \mathcal{R}_s^{\mathrm{a}} + \gamma \sum_{s' \epsilon S} \mathcal{P}_{\mathrm{ss}'}^{\mathrm{a}} \sum_{\mathrm{a}' \epsilon A} \pi(\mathrm{a}' \mid \mathrm{s}') q^\pi(s', \mathrm{a}') \tag{4.13}$$

The Bellman expectation equation for a particular π can be expressed using MRP:

$$v^\pi = \mathcal{R}^\pi + \gamma \mathcal{P}^\pi v^\pi \tag{4.14}$$

There is a direct solution for Bellman equation in MDPs which can not be easily solved for complex or large problems:

$$v^\pi = \mathcal{R}^\pi (1 - \gamma \mathcal{P}^\pi) \tag{4.15}$$

4.3.4 Bellman Optimality Equation

As mentioned, the state- and action-values are related to the considered policy. Generally in RL, we look for an optimum policy π^* which maximizes the total return. Following such optimal policy, the optimal state-value function and action-value function can be derived which would have the maximum values over all policies.

$$v^*(s) = \max_{\pi} v^{\pi}(s) \tag{4.16}$$

$$q^*(\text{s,a}) = \max_{\pi} q^{\pi}(\text{s,a}) \tag{4.17}$$

After defining the optimum policy, the question is: *how to find such optimum policy?* The methods for determining the optimum policy are discussed in further sections.

4.4 Dynamic Programming

Dynamic programming refers to the algorithms which improve the policy π in an MDP so that it will gradually converge to an optimum policy π^*. Generally, there are two types of MDP problems in which dynamic programming can be effectively utilized:

1. *Prediction problems:* In prediction problems, the input include an MDP {S, $\mathcal{A}$,$\mathcal{P}$,$\mathcal{R}$,γ} and the policy π and the outputs are the value function v_π.

2. *Control problems:* In control problems, the input includes an MDP {S, $\mathcal{A}$,$\mathcal{P}$,$\mathcal{R}$,γ} and the outputs include the optimal value function v_π and the optimal policy π.

Considering the characteristics of the structural control problem, the formulation of the dynamic programming for control problems is studied in next sections.

4.4.1 Policy Evaluation by State Iteration

Evaluating a policy in an MDP means to predict the maximum values that can be extracted by following the policy π. In dynamic programming, we iterate the Bellman expectation backup for all states to evaluate the policy π.:

$$V(s) = \max_{a \in \mathcal{A}} \sum_{s' \in S} \mathcal{P}^{\mathtt{a}}_{ss'} \left[\mathcal{R}^{\mathtt{a}}_{ss'} + V(s') \right] \tag{4.18}$$

Iterating this process many times results in converging the value function to the optimal values [185].

4.4.2 Policy Improvement

After calculating the action values in each state using equation 4.13, if for all $s \in S$,

$$q^{\pi}(s, \pi'(s)) \geq v^{\pi}(s) \tag{4.19}$$

Then the policy π' would be as good as, or better than π.

Therefore, the improvement in policy can be obtained by selecting an action in each state which seems better according to Equation 4.19. In other word, in each state, we consider policy π' :

$$\pi'(\mathrm{s}) = \arg\max_{a} q^{\pi}(\mathrm{s,a}) \tag{4.20}$$

4.4.3 Policy Iteration

By improving the policy π to policy π' as discussed in 4.20, the corresponding value functions $v^{\pi'}$ can be calculated and again the policy π' can be improved regarding to $v^{\pi'}$. By iterating such process the policy and value function will converge to the optimal policy and value functions,

$\pi_0 \xrightarrow{E} v^{\pi_0} \xrightarrow{I} \pi_1 \xrightarrow{E} v^{\pi_1} \xrightarrow{I} \pi_2 \xrightarrow{E} \cdots \xrightarrow{I} \pi^* \xrightarrow{I} v^*$

Where E denotes a policy evaluation and I denote a policy improvement.

4.4.4 Summary

Dynamic programming can solve a finite MDP by iterating some computations for *policy evaluation* and *policy improvement*, which leads to calculating the optimum value function and policy.

The main advantage of dynamic programming over the direct method is the low computational costs and the guarantee of the convergence. On the other hand, dynamic programming methods need complete knowledge about the environment. Such methods called model-based methods which are not applicable in all MDPs. In the next section, *model-free methods* are studied that are more appropriate for solving MDPs related to structural control problems.

4.5 Model Free Methods

In the model-free methods, the dynamics of the environment is not known to the agent so the learning algorithm relies on the agent's observations. In this section, Monte-Carlo evaluation, temporal difference, and Q-learning methods as main model-free methods are studied.

4.5.1 Monte-Carlo Policy Evaluation

Monte-Carlo policy evaluation is a model-free method for policy evaluation in MDPs. It determines the value function V by averaging the returns from the scenarios starting from the state s and continues to the end of episode[186].

As the returns can have high variation, this approach will result in poor estimations in some problems. In addition, in some problems such as closed-loop problems, it is not always possible to reset the state of the system to some particular state. In such conditions, some additional bias would be required to be applied.

Addressing these issues, some methods such as every-visit Monte-Carlo are developed. In this method, the value function in each state s, is defined as the average value of all the visits:

$$V(s) = \frac{1}{N(s)} \sum_{i=1}^{N(s)} R^i(s) \tag{4.21}$$

in which i is the visit counts and the R^i is the return, related to the visit i, and the $N(s)$ counts the total number of the visits. Although Monte-Carlo methods can be effectively utilized in solving MRPs, it requires a fixed policy. Therefore, the optimum policy can be studied by iterating different policies.

4.5.2 Temporal Difference Learning

In the temporal difference (TD) learning, the agent updates $v(s)$ from any transaction (s, a, s', r) without estimating the possible actions and next states in each state. In such an approach, each iteration includes two steps:

step1- Taking an action in state s which transfers the agent to state s' and then gathering the rewards.

$$v_a^\pi(s) = \mathcal{R}_s^{\mathrm{a}} + \gamma v^\pi(s') \tag{4.22}$$

step2- Update the value of state s:

$$V^\pi(s) = (1 - \alpha)V^\pi(s) + \alpha v_a^\pi(s) \tag{4.23}$$

where α is a small number, for example, 0.1, to consider more portion of current state value comparing to the previous experiences.

4.5.3 Q-Learning

In Q-Learning, instead of iterating existing policies, the optimum policy would be built by an agent through taking actions and observation cycles. Therefore, unlike the Monte-Carlo method, we do not need to iterate some pre-defined policies. This method is more appropriate for the structural control problems as we want to avoid any pre-definition of the policy.

In the Q-learning method, the goal of the learning algorithm is to determine the maximum possible return for all possible actions in each state.

$$Q^*(s, a) = \max_\pi \mathbb{E}\left[r_t + \gamma r_{t+1} + \gamma^2 r_{t+1} + \ldots + \gamma^{n-1} r_n \mid s_t = s,\, a_t = a,\, \pi\right] \tag{4.24}$$

As it is shown, following the policy π, the maximum return Q^* for the given state s and the action a, is the sum of reward r, discounted by the factor of γ in each time-step t until end of the simulation (time-step n).
Bellman equation defines the optimum action-value for each action a in the state s as :

$$Q^*(s,a) = \mathbb{E}_{s'}\left[r + \gamma \underset{a'}{max} Q^*(s',a') \mid s,a\right] \tag{4.25}$$

In this equation, the optimum action-value for each action in current state s, is defined as the sum of the immediate reward r and the maximum action-value of all the possible actions in the next time-step. As a result, the optimum action-values can be determined through an iterative process of estimating the optimum action values, determining errors and calculating new values.

$$Q_{i+1}(s,a) = r + \gamma \underset{a'}{max}\, Q(s',a') \tag{4.26}$$

therefore, if $i \to \infty$, Q converges Q^*. In this thesis, a deep neural network (NN) would be trained to estimate the optimum action-values. As NN is good in generalization, this technic would help the agent to estimate the Q-values for new states, for which it was not trained.
Having the optimum action-values, the optimum policy π^* would be to take the actions with maximum action-values in all states. In practice but, the learning process using the Q-net is a challenging task as it can be subjected to instabilities and divergences caused by the nonlinear nature of the neural networks [187]. Some causes of such instabilities are related to the correlations present in the sequence of observations, data distribution and the correlations between the current action-values and the target values. Moreover, As a small update to the Q values may significantly change the policy. Addressing these issues, Volodymyr Mnih et al. [188] introduced a new variant of Q-learning called *mini-batch learning* by doing two main improvements:

1. They introduced a biologically inspired experience replay that randomizes over the data and as a result, it improves the method by breaking the correlations in the training states sequence.

2. They used an iterative update that adjusts the action-values towards target values which will only periodically updated, which breaks the correlation between inputs and outputs of the Q-net during the learning process.

Achieving the last goal, in parallel to Q-net, they utilized a separate neural network to estimate the optimum action-values, as target values for training the Q-net, during the learning process. This net was defined as a clone of the Q-net and was updated periodically [188].

4.6 Exploration and Exploitation

In Q-learning, the *exploration* and *exploitation* is controlled by a policy called $\epsilon - greedy$ policy in which the balance between exploration and exploitation is adjusted

by ϵ which has a value less than a unit. The agent then takes the random actions with the probability of ϵ.

In this research, a variable $\epsilon - greedy$ policy is considered during the training phase so that initially, ϵ has a maximum value of 0.9 but it will gradually be reduced to a value of 0.2. It is because in the initial steps the Q-values are not stabilized and the agent needs more exploration. As the training phase proceeds and the Q-values converge to the Q* values, the agent needs to do more exploitation to stabilizing the Q-values.

4.7 Summary

In this chapter, the basic theories in RL were presented, the definition of MDP was studied, and different methods for solving an MDP were compared. Considering the characteristics of the structural control problem, the Q-learning method was selected to develop an intelligent framework.

Chapter 5

Intelligent Controller

5.1 Introduction

As discussed, in order to develop an intelligent controller, we have to develop an MDP first. In this chapter, the components of the intelligent control systems are defined in the way that they shape the body of an MDP. It is shown that solving the resultant MDP requires adaptation and improvements of the current RL methods. After that, based on the developed methods and discussed theories a framework for training a neural network as an intelligent controller is developed in MATLAB and Simulink and finally, two case studies are presented and the conclusions are drawn.

5.2 MDP in Structural Control

The intelligent controller comprises a deep neural network called Q-net that directly learns by interacting with the environment in terms of applying the control forces and observing the building responses. Based on the obtained data from the environment, the learning algorithm determines the reward and correlates that to the taken actions in each state. The goal of the learning algorithm is to maximize the sum of the rewards by improving the policy. In order to map a structural control problem to an MDP, we shall define the elements of the reinforcement learning problem including state, the action, and the reward based on the characteristics of a structural control problem.

5.2.1 State

As defined in Equation 4.1, to build an Markov state in a structural control problem, each state shall include the required data for determining the actions for the next steps.

In this regard, the ground accelerations as well as the structural responses including acceleration, velocity, and displacement responses are included in the each state S_t. The displacements in the last three time steps are included in the state vector as based on the experience, it helps the algorithm r to understand the direction of the motion which is important specially when reaching the maximums in the oscillations.

$$\{u_t, u_{t-1}, u_{t-2}, v_t, a_t, \ddot{u}_{g,t}\} \epsilon S_t \tag{5.1}$$

where:

u_t : displacement in time t

v_t : velocity in time t

a_t : acceleration in time t

$\ddot{u}_{g,t}$: ground acceleration in time t

5.2.2 Action

The output of the intelligent controller is the control signals that will be sent to the actuator. The actuator then transforms the signals to the external forces and applies them to the structure in a real-time manner. The control forces are defined in an applicable range and are divided to a number of force-steps as shown in Table 5.1. The action-value is a number in a range of $[1, n]$.

min. f (N)	max. f (N)	number of actions	magnitude of each action(N)
f_1	f_n	n	$\frac{f_n - f_1}{n-1}$

Table 5.1: Magnitude of the agent's action

5.2.3 Rewards

In a structural control problem, the reward value, r_t, evaluates the performance of the intelligent controller in the time-step t. As after applying a control force f_t to the structures, the maximum effect of the taken action happens after a time-delay Δt_d, the reward function shall consider some steps to look ahead and sum the rewards.

As a result, the reward of the taken action, a_t, is defined as the sum of the obtained rewards in the range of $[t, t + \Delta t_e]$. Considering a small value for Δt, values the immediate rewards and larger value participate more time-steps in the reward calculation for the taken action at time-step t.

In the Q-learning formulation, the rewards in the future are discounted by a γ factor:

$$R^{(n)} = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \ldots + \gamma^{n-1} r_{t+n} \tag{5.2}$$

As the reward value depends on the γ and the number of the considered steps n, investigating an appropriate value for these parameters plays important role in total behavior of the learning algorithm. This research argues that the characteristics of the problem shall be considered in determining these parameters. In this regard, this research improved the current formulation for the reward function by introducing an enhanced mini-batch learning method in which enters some characteristics of the environment is entered in the learning formulation.

5.2.3.1 Reward Function

In the structural control problem, the maximum reward value is considered for the situation that structure has no response. In addition, as a larger actuation force requires more power, a penalty is also considered for that.

The resultant reward function is defined as a sum of the discussed partial rewards,

$$R_t = R_M - \left[(\frac{u_t}{u_M}) + (\frac{v_t}{v_M}) + (\frac{a_t}{a_M}) + f_t \times p_{ac} \right] \tag{5.3}$$

In which:

u_t: Displacement response at time-step t

u_M: Maximum uncontrolled displacement responses

v_t: Velocity response at time-step t

v_M: Maximum uncontrolled velocity responses

a_t: Acceleration response at time-step t

a_M: Maximum uncontrolled acceleration responses

f_t: The actuation force at time-step t

P_{ac}: Penalty value for actuator unit force

R_M : The maximum reward

5.2.4 Learning Algorithm

The intelligent controller comprises a deep neural network called Q-net directly learns to reduce the responses of a dynamic system by interacting with the environment through applying the control forces and observing the responses. The goal of the learning algorithm is to develop a policy to maximize the sum of the future rewards by taking more appropriate actions in each state.

The Q-net estimates the action-values which were defined in equation 4.26:

$$Q_{i+1}(s, a) = r + \gamma \underset{a'}{max} Q(s', a')$$

when, $i \to \infty$, the Q will converge Q^*.

As an intrinsic property of neural networks, the Q-net also helps the controller in terms of generalization. Therefore, as it will be shown in the case studies, a trained Q-net can estimate the optimum action-values for a state s_ψ which is new to the Q-net.

Having the optimum action-values, the optimum policy π^* would be obtained by following a greedy policy which means that the agent takes the actions with the maximum action-value.

5.2.5 Enhanced Mini-Batch Learning

After utilizing mini-batch learning method to train the Q-net during more than 10000 learning episodes, some issues are observed:

1. Although the trained controller was able to significantly reduce the average response of the system under the earthquake excitations, it was not good enough in reducing the peaks of the responses. It is obvious that in structural control problems, reducing the peak of the response is very important in concern of stability and serviceability.

2. At the end of the responses, a residual shifting from the origin was seen in the displacement responses which were caused by control forces during the free vibration phase. It indicates that the optimum action-values in the states with a very low reward value were not determined properly. However, such shifting from origin shall be prevented as is not acceptable in practice.

These issues are mainly related to the characteristics of an structural control problem which are not considered in the learning method formulations.

Addressing these issues, an *enhanced mini-batch learning* method is developed in which the characteristics of the environment are considered in the learning method. This has been done by redefining the reward estimation formulation in Q-learning and also adding a key-state selector function as presented in the next sections.

5.2.5.1 Key-states selector function

This function improves the learning process by adding a random selection of key states to the learning batch which itself is a collection of the random states. Randomization helps the algorithm by breaking relationships between the input data. The function considers a *state* as a *key-state* if:

- It has a very low immediate reward value. Such state is related to the maximums in the responses.

- It has a very high immediate reward value. This criteria targets the states in which the difference of the uncontrolled and controlled responses is higher than other states which happens in low oscillation amplitudes.

The pseudo-code of the developed module is presented in Appendix A- Algorithm A.4.

5.2.5.2 Reflexive $\gamma-$ Function

The idea behind developing the reflexive $\gamma-$ Function is to participate some characteristics of the environment in estimation of the future reward. In this method, the agent takes an action as his first interaction with the environment and observes the response of the environment to that action. Based on the normalized environment's response, the agent then builds an γ-function which reflects the influence of the actions on the environment.

As illustrated in the Figure 5.1, in structural control problem, this function is built upon the building's response to an impulse load . In this regard, the agent normalizes the absolute response function and connects the peak responses, in a time-range that covers the maximum response, to build the reflexive γ-function.

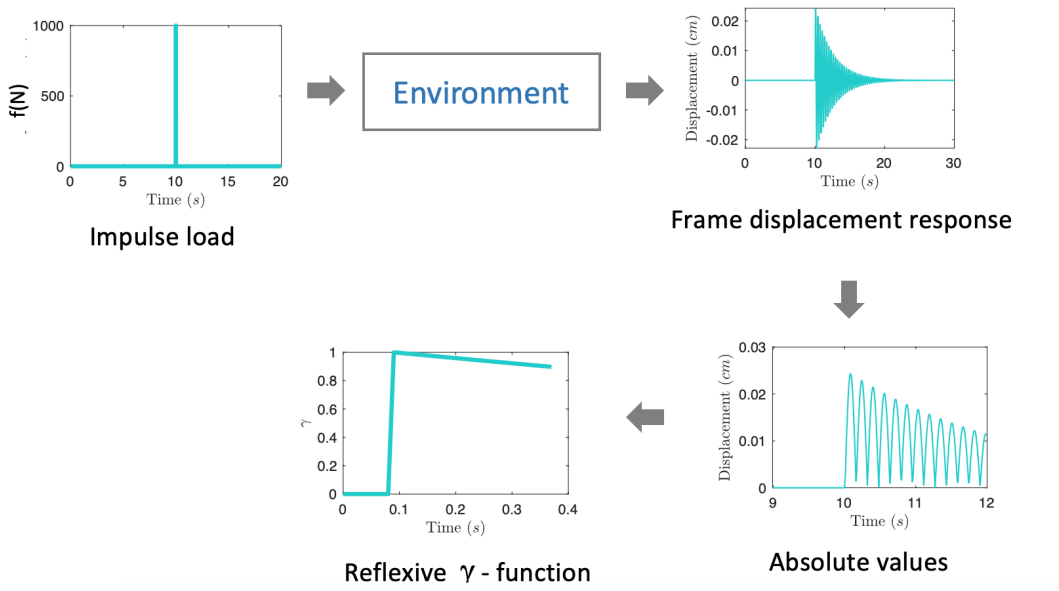


Figure 5.1: Process of creating reflexive γ-function

Having the γ-function, the agent determines the reward value for action a in the state s, by multiplying the immediate rewards to the corresponding γ- values in the time range $[t_0, t_0 + \Delta t_\gamma]$ and adding the maximum Q-value of the state s'.

$$Q_{i+1}(s,a) = \sum_{j=0}^{n} (\gamma_j r_j) + \gamma_{n+1} \max_{a'} Q(s', a') \tag{5.4}$$

where the γ is obtained from reflexive γ-function

r_j: immediate reward at the time t_0

n: number of considered time-steps in reflexive γ-function.

t_0 : The time when agent is experiencing the state s

Δt_γ time duration of reflexive γ-function

s': successor state at the time $t_0 + \Delta t_\gamma$

5.3 Framework

By using the discussed theories and the developed methods, a framework is developed to train a deep neural network for vibration control of a dynamic system. The inputs of the framework are the parameters of the structure including mass, damping, and stiffness matrices, and the external excitations. In the agent side, the architecture of the deep neural networks and the parameters of the training algorithm including the size of experience-memory, batch-size, and the stopping criteria shall be defined.

As it is illustrated in Figure 5.2, the main elements of the framework including:

a. Analytical model

b. Neural Networks

c. Learning module

d. Testing module

e. Simulation environment

The following sections describes the elements and their function in the framework.

5.3.1 Analytical Model

The analytical model includes three main functions:

1. state-space model
2. State generator
3. Reward generator

The state-space model determines the responses of the environment to an external excitation as well as the control forces. The state generator generates the state vector in each time-step based on the structural responses and the external excitations. The reward generator determines the immediate reward. The outputs of the analytical model will be used by the learning module.

5.3.2 Neural Networks

The intelligent controller consists of a deep NN called *Q-net* which interacts with an analytical model in terms of inputing the current state and sending the control signals. The output of the net is the vector of the Q-values. In addition to Q-net, a secondary stabilizer neural network is also developed to improve the performance of the learning module as discussed in Section 4.

In the literature still there is no concrete method for designing the architecture of the neural networks in machine learning problems and it is generally be done through an try and error process. In this research, the parameters of the net such as the number hidden layers, the number of the neurons, and the training algorithm are considered based on a try and error process. The optimum values for the parameters are expressed for each case study in the following sections.

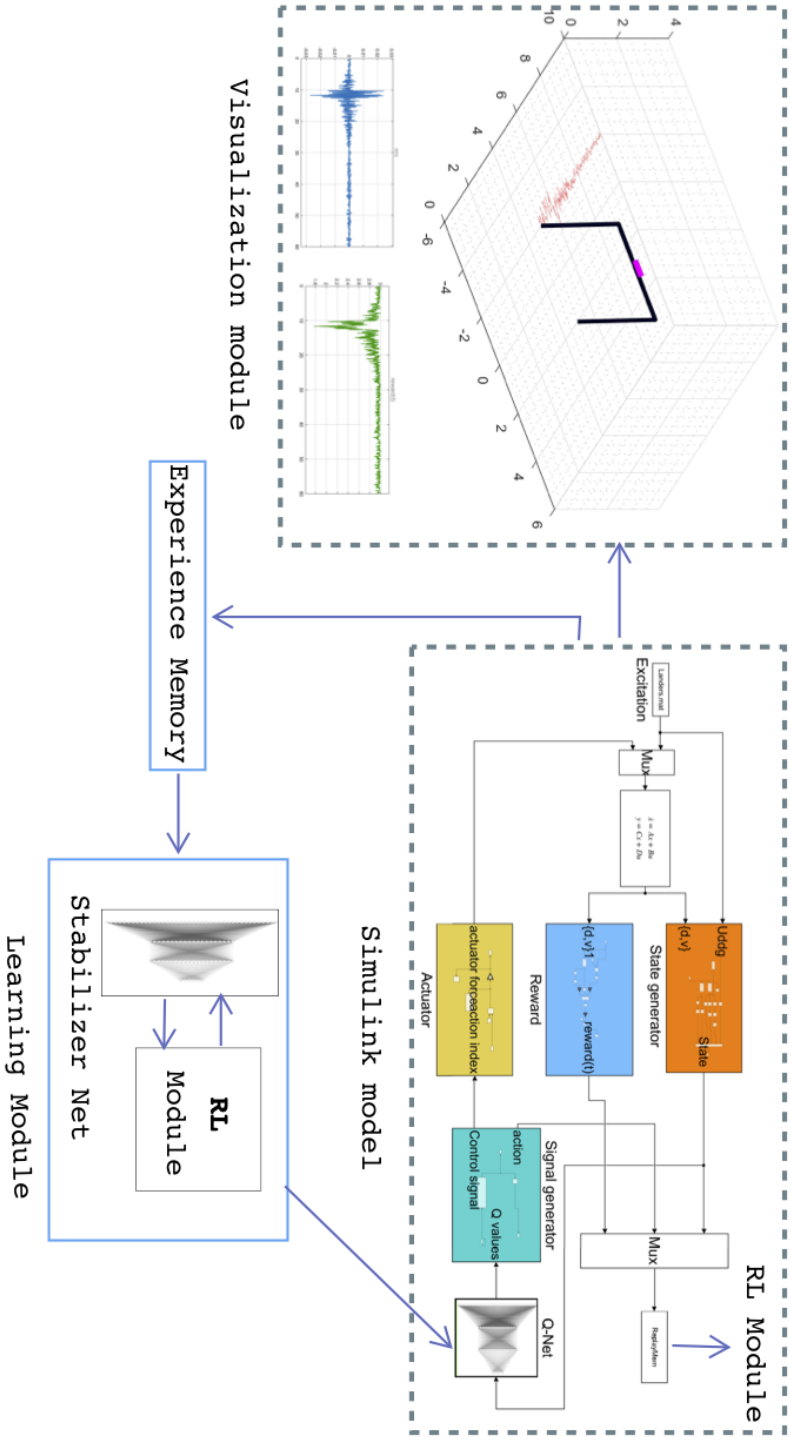


Figure 5.2: The developed framework for training an intelligent controller

5.3.3 Learning Rule

At the beginning of the training, the controller can not control the vibrations of the buildings as still it doesn't know the correct action-values in any state. During a large number of training episodes, the controller leans how to estimate the correct action-values. In general, the goal of the learning module is to improve the performance of the controller in terms of reducing the structural responses. The module interacts with the analytical model and the deep NN in terms of receiving the rewards and state vectors, and updating the weights and biases of the Q-net. In each call, this module trains the Q-net using a training dataset which is a batch of the random data from experience reply dataset as discussed mini-batch learning method. This module trains a stabilizer neural network and each 50 times of training the stabilizer-net, it updates the Q-net based on the weights and biases of the stabilizer-net. As discussed in the previous sections, this process results in converging the Q-values to the optimum values.

5.3.4 Testing Module

The testing module examines the performance of the intelligent controller under the earthquake records that are new to the controller's algorithm. This module interacts with the analytical model to perform the analysis and visualize the results.

5.3.5 Simulation Environment

In this research, a simulation environment is developed in Matlab software which graphically simulates the frame in training or testing episodes and shows some useful real-tine data such as earthquake excitation, reward values, control forces in a real-time manner. This helps us to trace the performance of the learning algorithm and the controller visually.

Chapter 6

Case Studies

6.1 Introduction

In this chapter, two case studies are presented. The goals is to:

1. Examine the effectiveness of the framework in developing an intelligent controller using the RL approach.
2. Studying the performance of the learning algorithm to train a deep neural network to reduce responses during an earthquake.
3. Comparing the performance of the developed method in RL, comparing to the original method.
4. Studying the performance of the trained controller under test excitations which are new to the algorithm.

The first case study is a single degree of freedom (SDOF) system and the second case study is a 20-story high-rise building which are considered as two extremes to evaluate the performance of the developed methods.

6.2 Case study I - Single Degree of Freedom System

6.2.1 Introduction

The first case study is a moment frame that is modeled as a single degree of freedom (SDOF) system. The simplicity of the dynamic behavior of an SDOF system facilitates the interpretation of the outputs and evaluation of the intelligent controller systems.

6.2.2 System Properties

The moment frame is modeled as an SDOF system as schematically is shown in Figure 6.2. The mass is considered as 2000 kg, the stiffness of the spring is 7.9×10^6

[N/s], and the damping is 250×10^3 N.s/m. As a result, the natural frequency ω, and the period T of the system are:

$$\omega = \sqrt{\frac{k}{m}} = \sqrt{\frac{7.9e6}{2000}} = 62.84 \, \frac{1}{s} \tag{6.1}$$

$$T = \frac{2\pi}{\omega} = 0.1 \; s \tag{6.2}$$

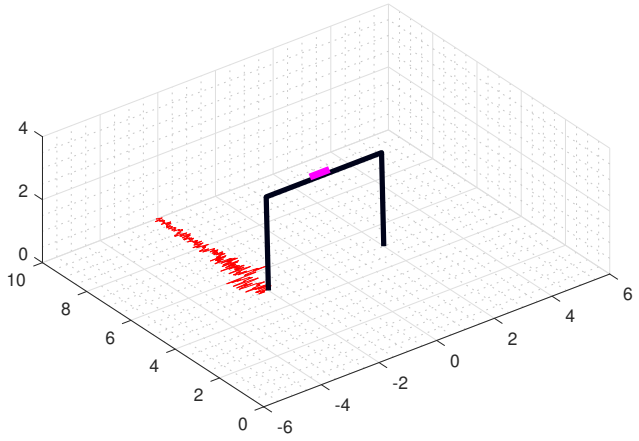


Figure 6.1: The simulation environment

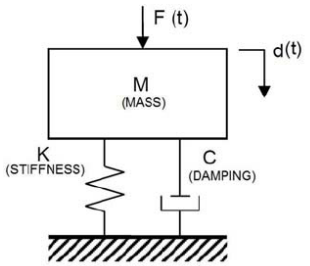


Figure 6.2: Schematic of single degree of freedom system

6.2.3 Dynamics of Environment

The Equation of the motion for the structure under earthquake excitations and the control forces is as follow:

$$m\ddot{u} + c\dot{u} + ku = -m\ddot{x}_g + f \tag{6.3}$$

in which m, c, k are the mass, damping and the stiffness matrices, $\ddot{x}_g$ is the ground acceleration as an external excitation, and f is the control force. u, $\dot{u}$, and $\ddot{u}$ are displacement, velocity, and acceleration vectors respectively.

By defining the state vector x as:

$$x = \{\mathrm{u},\ \dot{\mathrm{u}}\}^T \tag{6.4}$$

The state-space representation of the system:

$$\dot{x} = Ax + Ff + G\ddot{x}_g \tag{6.5}$$

$$y_m = C_m x \tag{6.6}$$

Considering $v = \{\ddot{x}_g,\ f\}$, Equation 6.5 can be written as:

$$\dot{x} = Ax + Bv \tag{6.7}$$

in which:

$$\begin{aligned} A &= \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -3947.8 & -125.66 \end{bmatrix} \\ B &= \begin{bmatrix} 0 & 0 \\ -1 & \frac{1}{m} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -1 & 5 \times 10^{-4} \end{bmatrix} \\ C_m &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned}$$

6.2.4 States

A proper definition of the state is an important task in RL problems. In order to build an MDP, we have defined the state vector S_t is as follow:

$$\{u_t, u_{t-1}, u_{t-2}, v_t, a_t, \ddot{u}_{g,t}\}\, \epsilon\, S_t \tag{6.8}$$

where:

u_t : displacement at the time t

v_t : velocity at the time t

a_t : acceleration at the time t

$\ddot{u}_{g,t}$: ground acceleration at the time t

This vector includes the structural responses as well as the ground acceleration that include required data for determining the current situation and estimate the future response. Considering the displacement responses in the previous time steps also helps the controller to determine the direction and the phase of the response.

6.2.5 Actions

In the structural control problems, the agent's actions are the control forces. In this regard, the control forces are limited to the applicable values with an absolute maximum value of $4000N$ (See Table 6.1). The force range is then divided to 40 load-steps to form 40 possible actions in each state. As a result, the action-value is a number in a range of $[1, 40]$.

min. $f(N)$	max. $f(N)$	number of load-steps	magnitude of each load step (N)
- 4000	+ 4000	40	100

Table 6.1: Control force range and load-steps

6.2.6 Reward Function

As it is discussed in the previous chapter, the reward function evaluates the behavior of the agent regarding the problem's objectives. In this regard, a multi-objective reward function is defined which adds four partial rewards to determine the reward value:

A. Displacement response

The first partial reward function reflects the performance of the controller in terms of reducing the displacement responses:

$$R_{1,t} = 1 - \frac{\mid u_t \mid}{u_{max}}$$

in which, u_t is the displacement value of the frame at the time t and u_{max} is the maximum uncontrolled displacement response.

B. Velocity response

This partial reward evaluates the velocity response of the frame:

$$R_{2,t} = 1 - \frac{\mid v_t \mid}{v_{max}}$$

in which v_t is the velocity response of the frame at the time t and v_{max} is the maximum uncontrolled velocity response.

C. Acceleration response

The performance of the controller in term of reducing the acceleration responses of the frame is evaluated by $R_{3,t}$:

$$R_{3,t} = 1 - \frac{\mid a_t \mid}{a_{max}}$$

in which a_t is the acceleration response of the frame at the time t and a_{max} is the maximum uncontrolled acceleration response.

D. Actuator force

The goal of the forth partial reward is to evaluate the required energy by applying a penalty value equal to 0.005 to the actuator force in each time-step:

$R_{4,t} = f_t \times P_a$

in which:

f_t = Actuation force in the time $t(N)$

P_a =Penalty value for unit actuator force (= 0.005)

By combining the four partial rewards, the reward value R, at the time t will be built:

$$R_t = R_{1,t} + R_{2,t} + R_{3,t} + R_{4,t}$$

6.2.7 Earthquake Excitations

In order to train the controller, Landers earthquake record is considered. The 1992 Lander earthquake occurred in June 28 with an epicenter near the town of Landers, California. The shock had a moment magnitude of 7.3 and a maximum. The acceleration record of the earthquake is obtained from the NGA strong motion database[6] (See Figure 6.3).

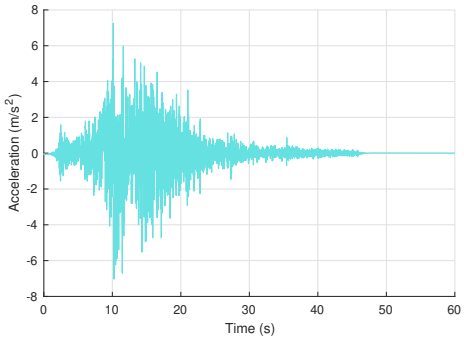


Figure 6.3: Landers earthquake's acceleration record

6.2.8 Learning by Mini-Batch Learning Method

Initially, the controller was trained using the mini-batch learning method [188]. In this method, after recording enough states into the experience reply, the target Q-values will be determined by the learning module which will be then utilized to train the Q-net. The utilized train data-sets include some random mini-batches. The parameters that controls the behavior of the learning algorithm are shown in Table 6.2.

In the beginning of the training phase, the controller had no idea about controlling the moment frame under the earthquake excitation as still it was not trained to evaluate the Q-values. Consequently, the controlled responses of the frame were even worse than the uncontrolled ones. Based on the discussed theories in the previous

Number of episodes	Size of experience reply	Number of states per episode	sensor sampling rate (Hz)	Mini-batch size
1000	60000	6000	100	50

Table 6.2: Utilized learning parameters

chapter, the learning algorithm gradually trained the Q-net to determine the optimum Q-values which resulted in improving the performance of the controller. As it is shown in Figure 6.4, during the training phase, the average reward has improved from about 2.41 to a maximum value of 2.78.

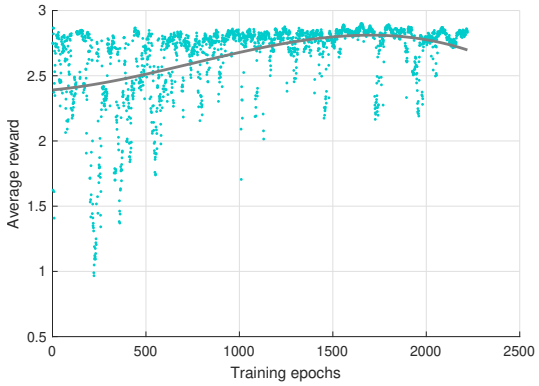


Figure 6.4: Average reward values during the learning phase

The algorithm stopped the training after 2200 episodes as no further improvements were achieved. The performance of the trained controller in terms of reducing the displacement, velocity, acceleration responses of the frame to the Landers earthquake are shown in Figure 6.5.

The results show that although the controller has improved the structural responses, it's performance is not so good in reducing the peak responses. In this regard, with respect to reducing the peak responses, the controller has shown a low performance in reducing peak displacement and velocity responses with the values of 7.1%, and 8.7%, respectively. The controller has reduced the peak acceleration response by 26.7% which indicates a better performance comparing to the displacement and the velocity as presented in Table 6.3.

Besides the issues with the performance, two discussed issues with the original learning algorithm including the low performance in reducing the peak responses and residual shift from origin were seen in the results.

6.2.9 Improved Mini-Batch Learning

Addressing the issues with the mini-batch learning method, this study has developed an enhanced mini-batch learning method which improves the performance of the original method in structural control problems as expressed previously in Section 5.2.5.

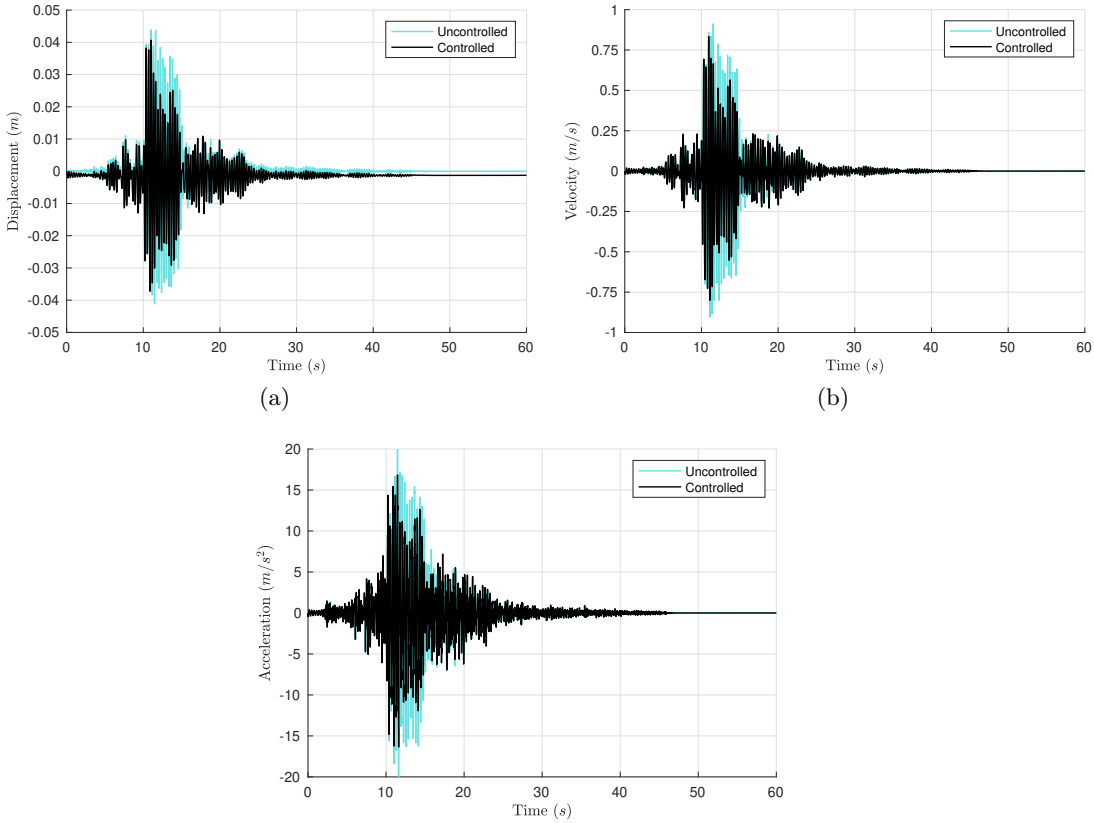


Figure 6.5: Controlled and uncontrolled responses of the frame to the Landers earthquake when the intelligent controller was trained using the original learning algorithm.

To study the performance of the developed method, the controller is trained using the enhanced method during 11000 learning episodes. As it is shown in Figure 6.6, the train episodes is reached 11000 as the performance was improving and the stopping criteria was not met. The results show that the average reward is increased from about 2.61 to a value of 2.90 which indicates a better performance of the learning algorithm.

The uncontrolled and controller response of the frame under both methods are compared in Figure 6.7. The results indicate that the improved method has significantly upgraded the performance of the controller in terms of reducing the peak responses as well as the average response. In addition, the issue with the shifting from the origin, which was seen in the initial results, is solved.

As summarized in Table 6.3, the performance of the controller in terms of reducing the peak of uncontrolled to controlled displacement response is improve from 7.1% to 46%, similar improvement for the velocity has been seen as it is improved from 8.7% to 41%, and for accelerations, the performance is improved from 26.7% to 37.8%.

The obtained results indicate a significant improvement, achieved by using the enhanced mini-batch learning method. Moreover, the average root mean square(RMS)

of responses has been reduced significantly.

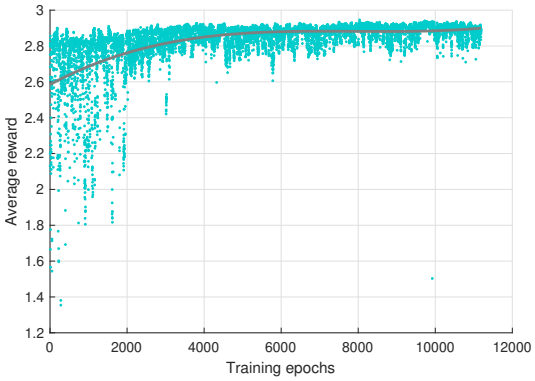
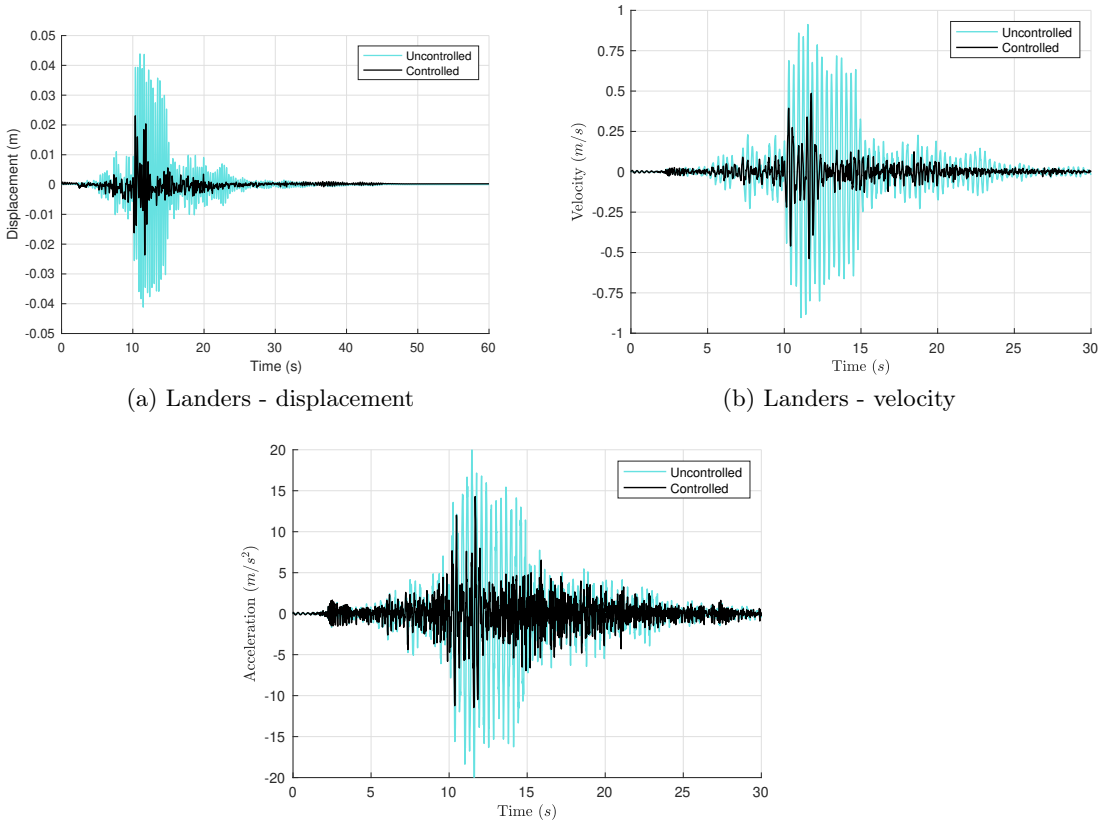


Figure 6.6: Improvement of the average rewards during the learning process using the enhanced mini-batch learning method.



(a) Landers - displacement

(b) Landers - velocity

(c) Landers - acceleration

Figure 6.7: Controlled and uncontrolled responses of the frame when the controller has been trained by the enhanced method.

Learning method		Uncontrolled	Controlled	Improvement
Original method	Peak Dis.	4.39	4.06	7.1%
	Peak Vel.	0.91	0.83	8.7%
	Peak Acc.	22.97	16.84	26.7 %
	RMS Dis.	0.0081	0.0068	15.5 %
	RMS Vel.	0.118	0.097	17.7 %
	RMS Acc.	3.378	2.816	16.6 %
Improved method	Peak Dis.	4.39	2.36	46.1 %
	Peak Vel.	0.91	0.54	41.0 %
	Peak Acc.	22.97	14.28	37.8 %
	RMS Dis.	0.0081	0.0017	78.8%
	RMS Vel.	0.118	0.0319	72.9%
	Peak Acc.	3.378	1.180	65.0

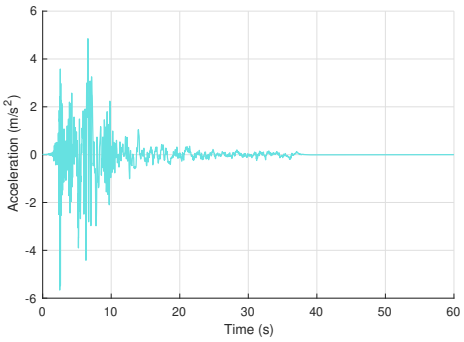
Table 6.3: Controlled and uncontrolled responses of the frame to the Landers earthquake excitations when the controller has been trained by original and enhanced methods (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (m/s^2)).

6.2.10 Testing Phase

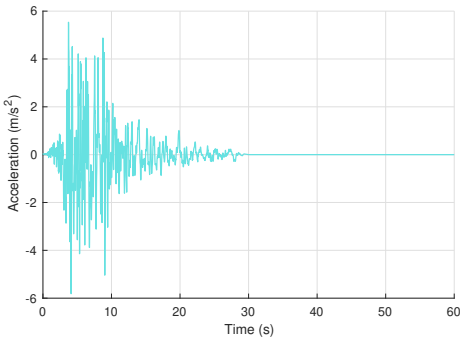
The performance stability of the controllers is an important issue. In this regard, the performance of the intelligent controller which was trained under Landers earthquake, has been tested under four earthquake records which were new to the controller. The acceleration records of these excitations are obtained from NGA strong motion database [6](see Figure 6.8). The obtained controlled and uncontrolled responses are presented in Figure 6.9.

As summarized in Table 6.4, in terms of reducing the displacement responses, the maximum achieved performance is 52.5% which is obtained for the Northridge earthquake and the minimum is a 40.9% for the Kobe earthquake. The performance of the controller in terms of reducing the velocity responses varies between 41.4% to 56.3% and the acceleration responses are improved between 37.3% to 50.1%.

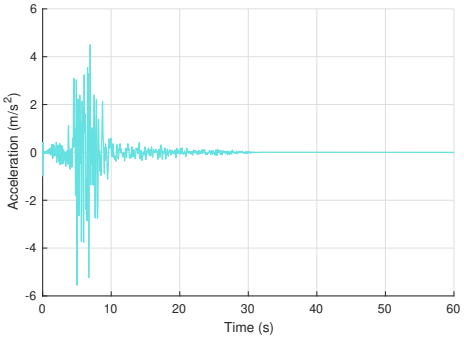
The average performance of the controller in terms of improving displacement responses under four new earthquakes is 47.1% which is comparable to the 46 % obtained for the Landers earthquake, the earthquake for which it was trained. The achieved average performance in terms of reducing velocity and accelerations are 49.2% and 43.4%, respectively, which are even higher than obtained values for the Landers earthquake in the training phase as 41.0 % and 37.8 %. In addition, the average RMS of displacement, velocity, and acceleration responses are significantly improved by 57.6 %, 67.9 %, and 46.6 %, respectively (Table 6.4). The results show that the controller has improved the frame responses in all new excitations for which it was not trained.



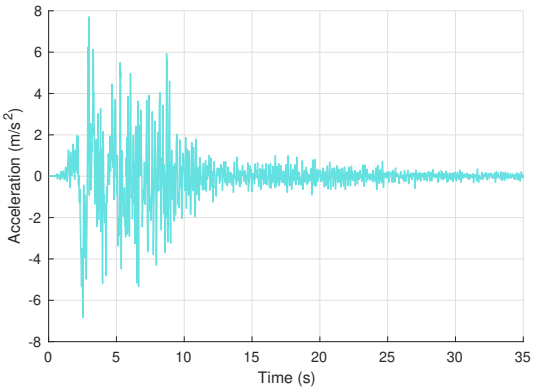
(a) El Centro, 1940- USA



(b) Northridge, 1971- USA

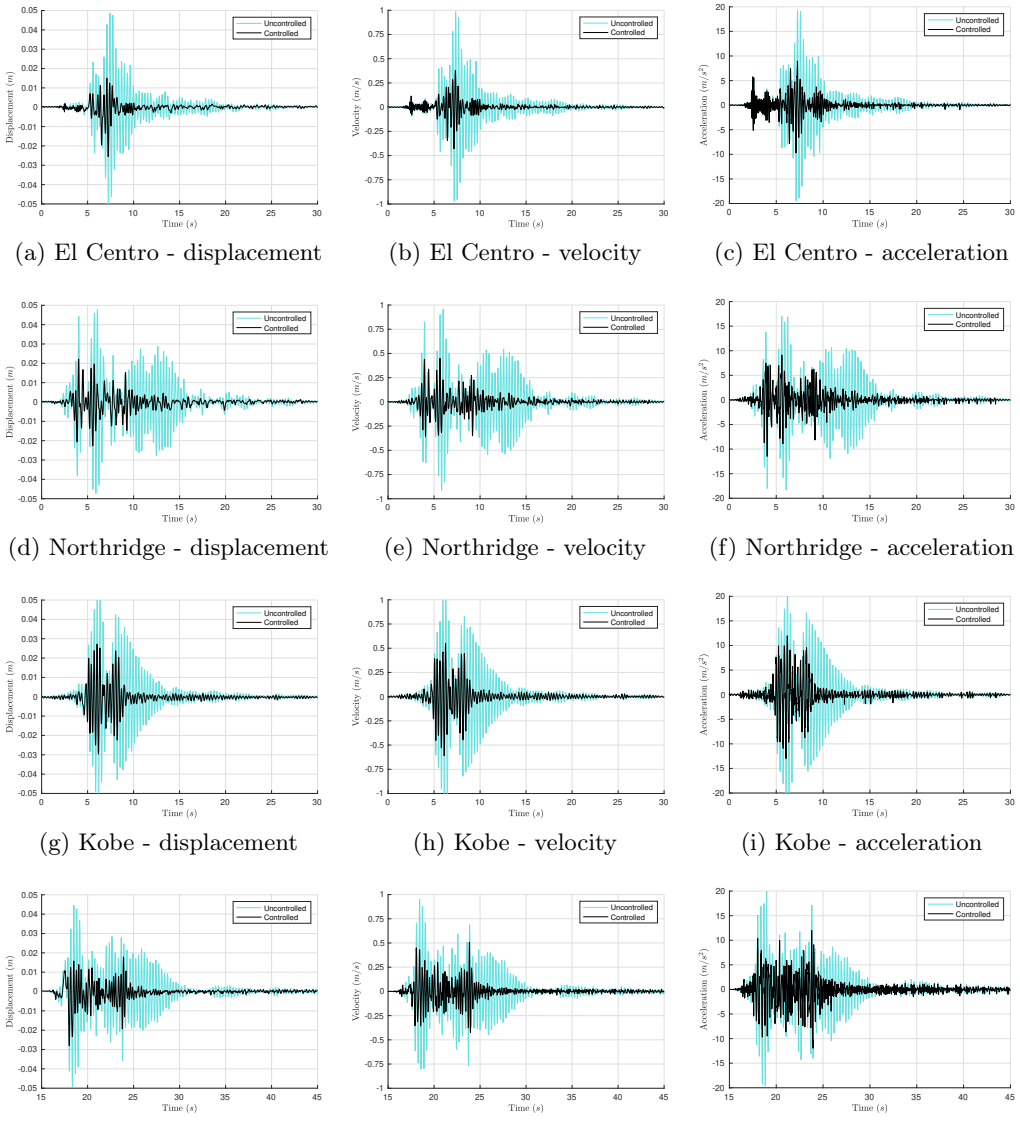


(c) Kobe, 1995- Japan



(d) Bam, 2003- Iran

Figure 6.8: Four earthquake acceleration records which were considered to test the performance of the controller



(a) El Centro - displacement (b) El Centro - velocity (c) El Centro - acceleration

(d) Northridge - displacement (e) Northridge - velocity (f) Northridge - acceleration

(g) Kobe - displacement (h) Kobe - velocity (i) Kobe - acceleration

(j) Bam - displacement (k) Bam - velocity (l) Bam - acceleration

Figure 6.9: Uncontrolled and controlled responses of the frame to the test earthquake excitations which are new to the intelligent controller .

Earthquakes		Uncontrolled	Controlled	Improvement
El Centro	Dis.	5.35	2.56	52.1 %
	Vel.	0.98	0.43	56.3 %
	Acc.	19.46	9.71	50.1 %
	RMS Dis.	0.0045	0.0010	77.9 %
	RMS Vel.	0.1178	0.0368	713 %
	RMS Acc.	2.0802	0.7190	65.4 %
Northridge	Dis.	4.77	2.27	52.5 %
	Vel.	0.95	0.45	52.9 %
	Acc.	18.28	11.46	37.3 %
	RMS Dis.	0.0067	0.0030	55.6 %
	RMS Vel.	0.1721	0.0399	76.8 %
	RMS Acc.	2.3172	1.4011	39.5 %
Kobe	Dis.	5.91	3.49	40.9 %
	Vel.	1.15	0.68	41.4 %
	Acc.	24.24	13.30	45.1 %
	RMS Dis.	0.0080	0.0049	38.3 %
	RMS Vel.	0.1498	0.0560	62.6 %
	RMS Acc.	2.8114	2.0754	26.1 %
Bam	Dis.	4.92	2.80	43.1 %
	Vel.	0.95	0.50	46.5 %
	Acc.	20.44	11.99	41.3 %
	RMS Dis.	0.0083	0.0044	46.4 %
	RMS Ve.	0.1304	0.0559	57.1 %
	RMS Acc.	3.1137	1.9390	37.7 %
Average response	Dis.	5.24	2.78	47.1 %
	Vel.	1.01	0.52	49.2 %
	Acc.	20.61	11.62	43.4 %
	RMS Dis.	0.0030	0.0071	57.6 %
	RMS Vels.	0.0441	0.1376	67.9 %
	RMS Acc.	1.4629	2.7403	46.6 %

Table 6.4: Responses to the earthquake excitations (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (m/s^2)).

6.2.11 Environmental Uncertainties

In the real-world situation, the structural parameters will not remain constant during the structure's life-cycle. Therefore, having a stable performance under the environmental uncertainties is an important issue in structural control systems. To study the performance of the algorithms under environmental uncertainties, the stiffness matrix is multiplied to uncertainty factors which varies between 5 % and 40 %. The goal of the controller is to reduce the structural responses under the El Centro earthquake. Note that the active control systems are not sensitive to the uncertainties in the damping matrix [189]).

As it is presented in Table 6.5, the obtained performance varies between 37.2% to

60.3% with an average value of 51.8%, which is close to 52.1% and obtained for the original structure. The results imply that the intelligent controller has showed a stable performance under the environmental uncertainties.

Uncertainty	Uncontrolled Dis.	Controlled Dis.	Improvement (%)
-5 %	0.0535	0.02559	52.1
+5 %	0.0537	0.02291	57.3
-10 %	0.0521	0.02699	48.2
+10 %	0.0493	0.01970	60.0
-15 %	0.0518	0.03016	41.8
+15 %	0.0411	0.01932	53.1
-20 %	0.0558	0.03507	37.2
+20 %	0.0401	0.01778	55.6
-30 %	0.0649	0.03682	43.2
+30 %	0.0429	0.01766	58.8
-40 %	0.0728	0.03350	54.0
+40 %	0.0442	0.01756	60.3
	Average		51.8

Table 6.5: Performance of the controller in reducing displacement responses under environmental uncertainties (Dis. = Displacement (m))

6.3 Case Study 2 - High-Rise Building

6.3.1 Introduction

In the second case study, the same framework is utilized to train a neural network for improving the responses of a high-rise building, subjected to different earthquake excitations. The utilized benchmark building is developed by Spencer et al. [7] for seismic control problems.

6.3.2 Benchmark Building

The Los Angeles twenty-story building is 30.48 m by 36.58 m in the plan, and 80.77 m in elevation which was designed under the code specifications for the Los Angeles city in California. The bays are 6.10 m on center, in both directions, with five bays in the north-south (N-S) direction and six bays in the east-west (E-W) direction. The building's lateral load resisting system is comprised of steel perimeter moment-resisting frames (MRFs).

The seismic mass of the first level is 5.32×10^5 kg, for the second level is 5.65×10^5 kg, for the third level to 20th level is 5.51×10^5 kg, and for the roof is 5.83×10^5 kg. The seismic mass of the entire structure is 1.16×10^7 kg.

The benchmark model is developed based on an in-plane (2-D) analysis of one-half of the entire structure. The considered frame is one of the N-S MRFs (the short direction of the building) as it is shown in Figure 6.10.

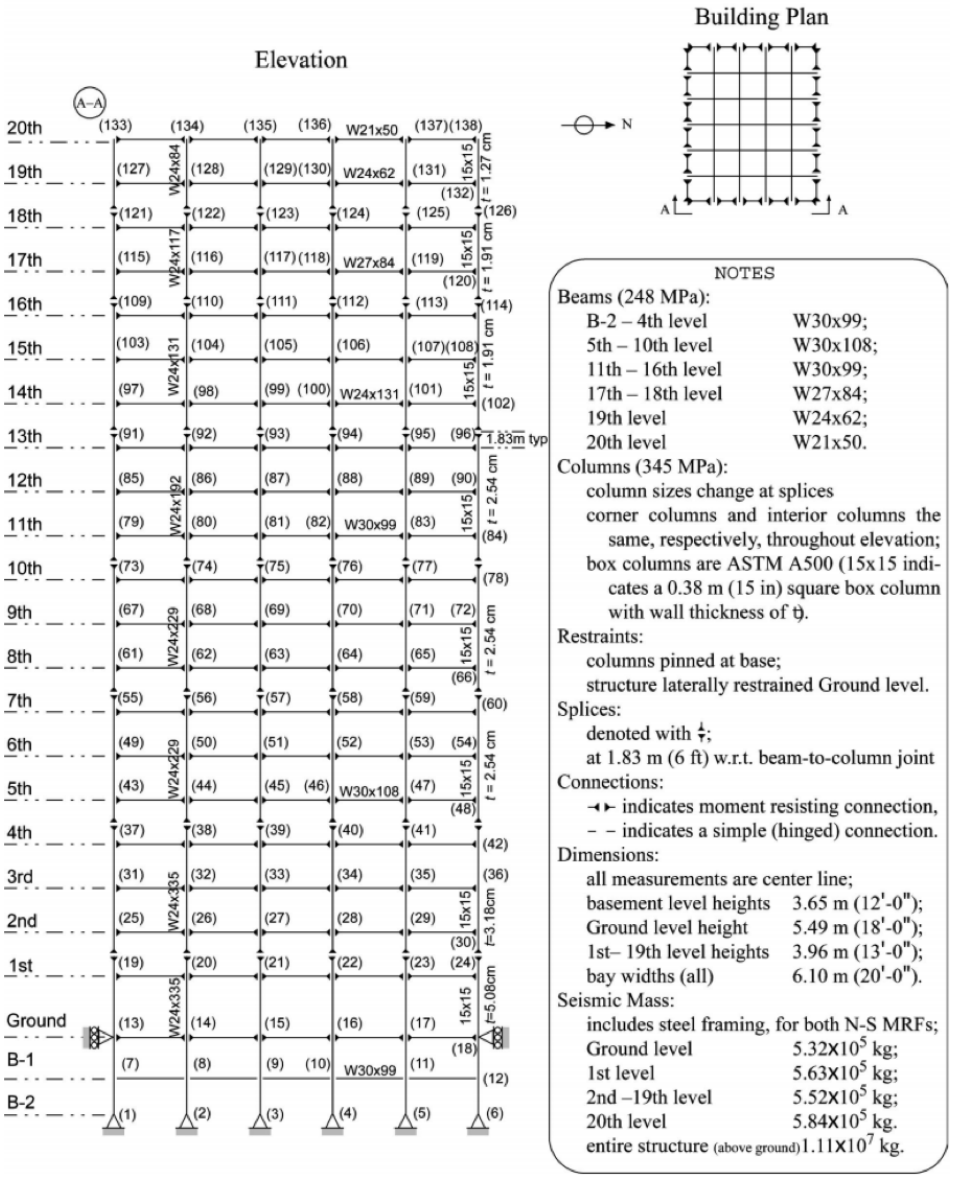
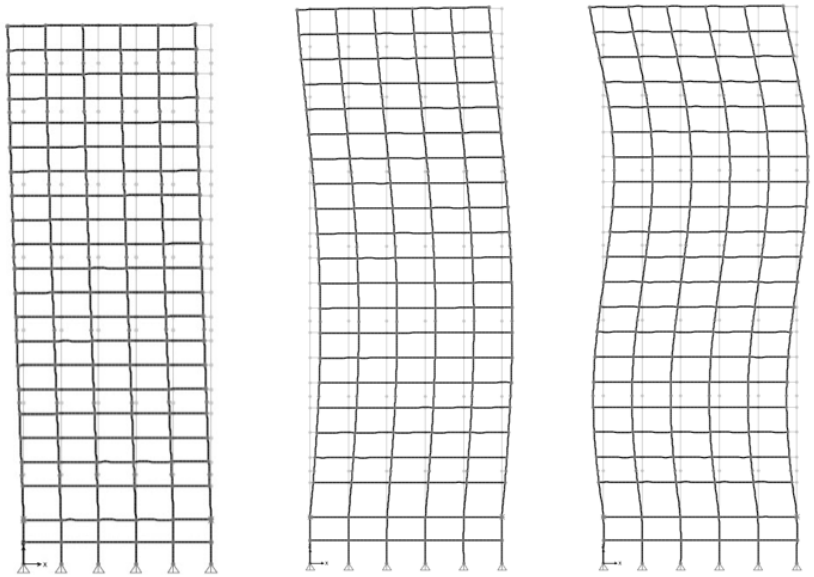


Figure 6.10: Figure 1- N-S moment resisting frame of twenty story benchmark building[7]

The seismic mass of a single N-S MRF, without the mass of the steel framing, for the first level is 2.54×10^5 kg, for the second level is 2.70×10^5 kg, for the third level to 20th level is 2.63×10^5 kg, and for the roof is 2.79×10^5 kg.

Each node in the developed 2D model of N-S MRF has three degrees-of-freedom (DOFs) including horizontal, vertical displacements and rotation. The DOFs are partitioned into active and dependent (slave) DOFs. the active horizontal DOFs, as well as the vertical DOFs for levels 1–21 located on the second and fifth column lines, are chosen to be active. All other vertical DOFs, including the vertical DOFs at splice locations, and all rotational DOFs are assumed dependent and condensed out. After condensations and applying Guyan reduction, the final model has 106

DOFs. The total seismic mass of the N-S MRF is 5.80×10^6 kg. The first three mode shapes are shown in Figure 6.11.



(a) Mode 1 (0.26 Hz) (b) Mode 2 (0.75 Hz) (c) Mode 3 (1.30 Hz)

Figure 6.11: Three mode shapes of the 20-story benchmark building

6.3.3 Environment Dynamics

The governing equation of the motion for the high-rise building under earthquake excitations and the control forces can be expressed as follows:

$$\mathbf{M\ddot{U}} + \mathbf{C\dot{U}} + \mathbf{KU} = -\mathbf{M\Gamma}\ddot{x} + \mathbf{P}f \tag{6.9}$$

The state vector is considered as:

$$x = \left\{\mathbf{U}, \dot{\mathbf{U}}\right\}^T$$

$$\dot{x} = Ax + Ff + G\ddot{x}_g$$

$$y_m = C_m x$$

The state-space parameters then would be defined as follow:

$$\dot{x} = Ax + Bv$$

in which:

$$
\begin{aligned}
A &= \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\frac{\mathbf{K}}{\mathbf{M}} & -\frac{\mathbf{C}}{\mathbf{M}} \end{bmatrix} \\
B &= \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -\mathbf{I} & \mathbf{M}^{-1} \end{bmatrix} \\
C_m &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}
\end{aligned}
$$

where $\mathbf{M}$, $\mathbf{K}$ and $\mathbf{C}$ represent the mass, the stiffness, and the damping matrices. $\mathbf{\Gamma}$ is a vector of zeros and ones which defines the loading of the ground acceleration to the structure, and $\mathbf{P}$ is a vector that defines how the control force would be produced by the controller, enters the structure, and $\ddot{x}_g$ represents the ground accelerations.

6.3.4 Earthquake Excitations

In order to learn the controller, Landers earthquake record is considered which was also utilized in the first case study (see Figure 6.3).

6.3.5 Actions

In the high-rise building, the control force is limited to the absolute maximum value of $10000N$ (See Table 6.6). The force range is then discretized to 80 load steps . As a result, the action-value is a number in a range of $[1, 80]$.

min. f (N)	max. f (N)	number of force steps	magnitude of each force step (N)
-100000	+100000	80	2500

Table 6.6: Actuator force range and the number of divisions used for discretization

6.3.6 Reward Function

Similar to the SDOF system, the reward function is defined as sum of partial rewards as follows:

$$R_t = R_M - \left[(\frac{u_t}{u_M}) + (\frac{v_t}{v_M}) + (\frac{a_t}{a_M}) + f_t \times p_{ac} \right]$$

Where:

u_t: Displacement response at time-step t

u_M: Maximum uncontrolled displacement responses

v_t: Velocity response at time-step t

v_M: Maximum uncontrolled velocity responses

a_t: Acceleration response at time-step t

a_M: Maximum uncontrolled acceleration responses

f_t: The actuation force at time-step t

P_{ac}: Penalty value for actuator unit force

R_M : The maximum reward

6.3.7 Learning by Improved Q-Learning

The controller was trained by the framework using the developed enhanced mini-batch learning method to reduce the vibrations of the building under the Landers earthquake excitations. The learning parameters are considered as shown in Table 6.7.

Number of episodes	Size of experience reply	Number of states per episode	sensor sampling rate (Hz)	Mini-batch size
1000	60000	6000	100	50

Table 6.7: Utilized Learning parameters

6.3.8 Training Phase

During the training phase, the controller experienced 1000 episodes. The performance of the algorithm is shown in Figure 6.12. After training phase, the maximum performance of the controller in terms of improving the displacement responses as well as the drifts are examined under the Landers earthquake as shown in Figures 3.13 and 6.14.

As it is summarized in Table 6.8, the results show that the controlled has successfully improved the peak displacement by 23.9%. The RMS displacement responses is also significantly improved by 61.24%. The achieved results indicates that the framework is able to train the neural network to improve the seismic responses of single story to high-rise buildings.

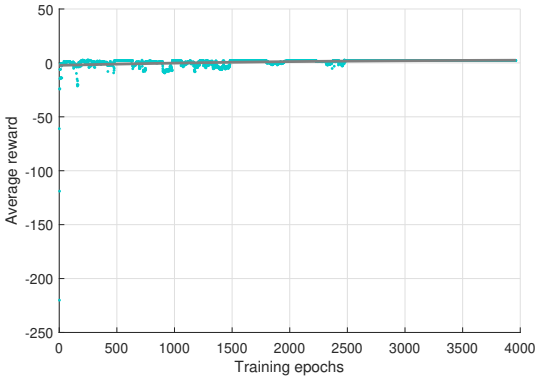


Figure 6.12: Improvement of the average rewards during the learning process using the improved method.

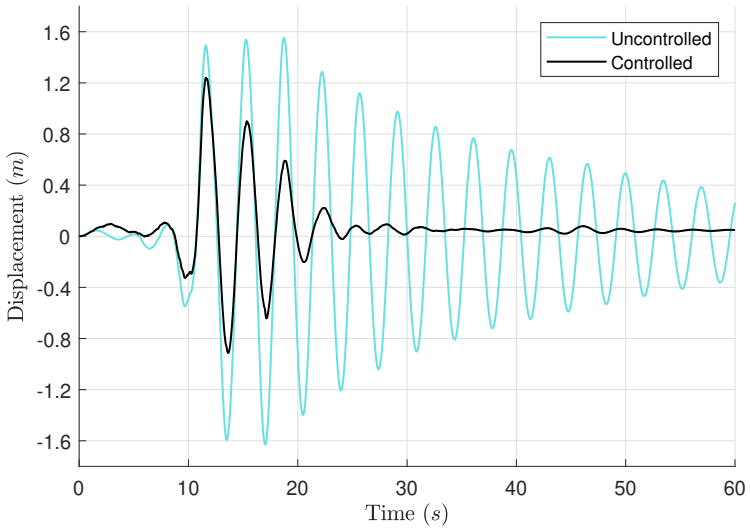


Figure 6.13: Controlled and uncontrolled displacement responses under Landers earthquake

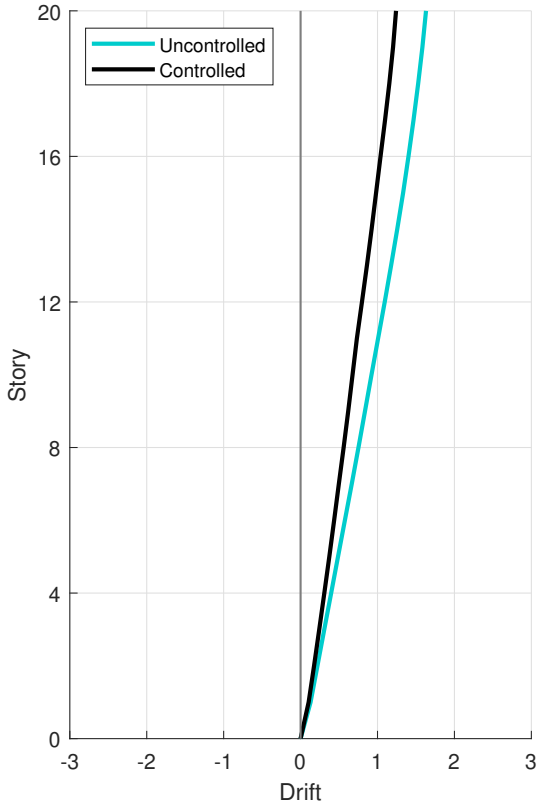


Figure 6.14: Controlled and uncontrolled Story drifts under Landers earthquake

	Uncontrolled	Controlled	Improvement (%)
Peak (cm)	163.06	124.08	23.90
RMS	0.62	0.24	61.24

Table 6.8: Comparing the controlled and uncontrolled displacement responses to Landers earthquake

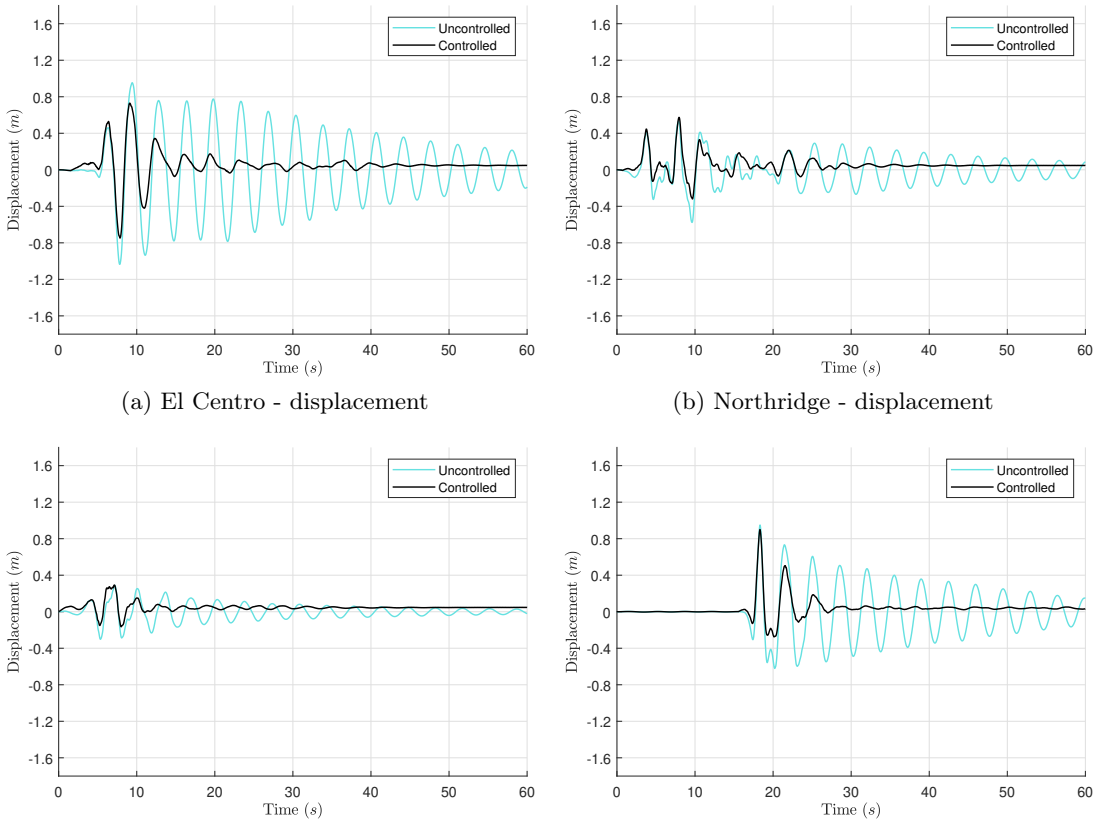
6.3.9 Testing Phase

The efficiency of the controller then was examined under four earthquake excitations as presented in Figure 6.8.

The results show that despite the diversity of the excitations, the controller was able to improve the structural responses in all cases as shown in Figure 6.15.

In terms of reducing the peak of displacement response, 9.16% improvement is obtained in average. The maximum performance is seen under El Centro earthquake with a value of 27.8% and the worst performance is obtained for the Northridge earthquake as 0.36% (See Table 6.9). Considering the displacement responses, it is seen that the damping of the system has been improved significantly in all cases.

The RMS displacement responses are also improved by the controller with an average value of 41.39%. The controlled and uncontrolled drifts are shown in Figure 3.13. As it is shown the drift of the stories is reduced under all earthquake excitations. The typical transfer functions for the high-rise building under different base excitations are compared in Figure 6.17.



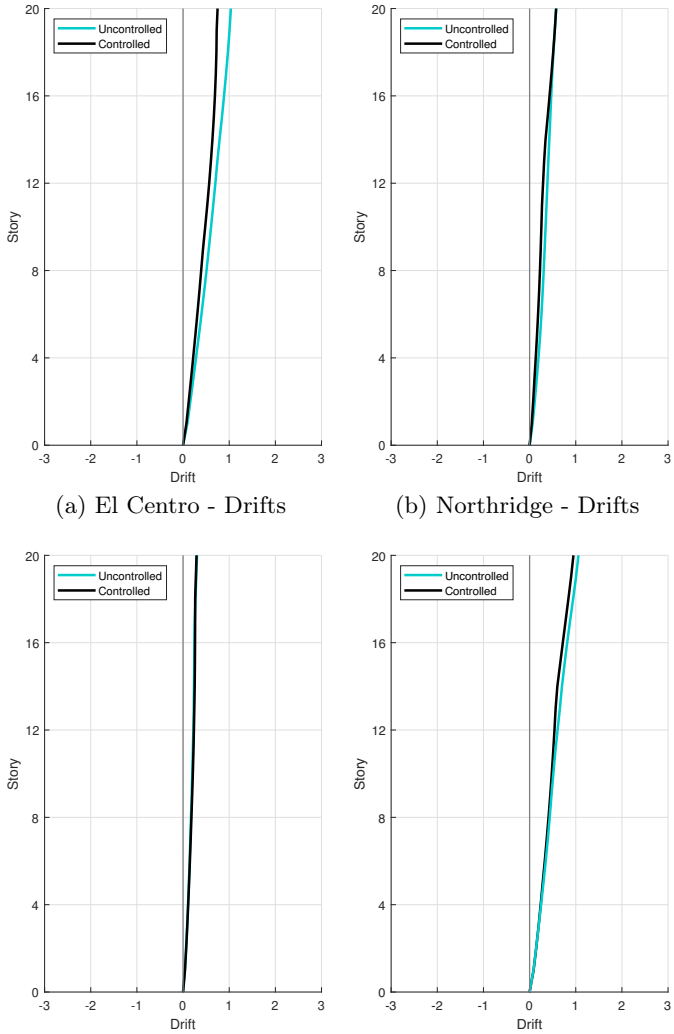
(a) El Centro - displacement

(b) Northridge - displacement

(c) Kobe - displacement

(d) Bam - displacement

Figure 6.15: Uncontrolled and controlled displacement responses .



(a) El Centro - Drifts

(b) Northridge - Drifts

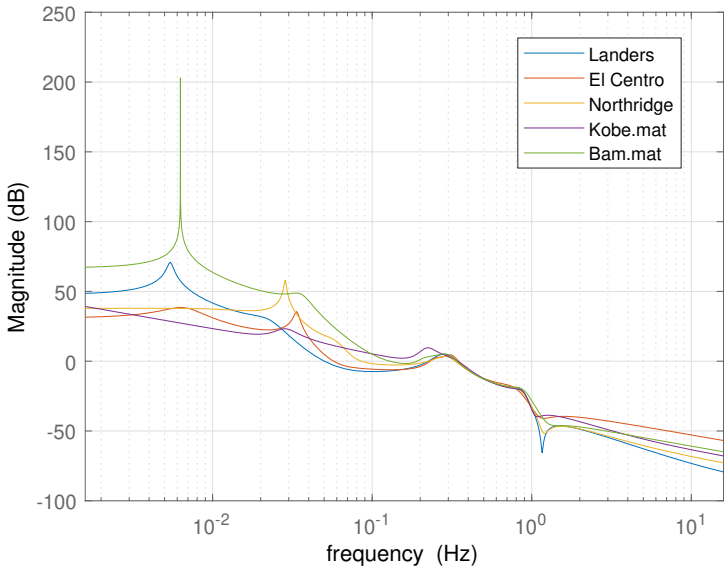
(c) Kobe - Drifts

(d) Bam - Drifts

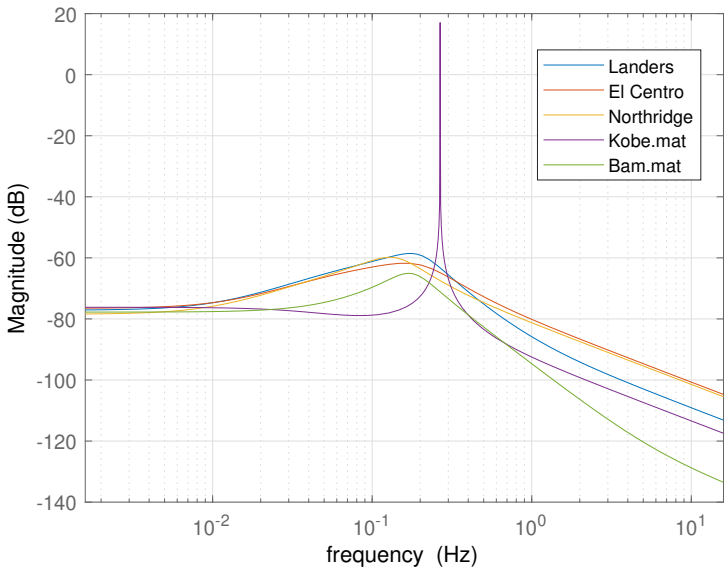
Figure 6.16: Uncontrolled and controlled Drifts

Earthquakes		Uncontrolled	Controlled	Improvement
El Centro	Peak	103.62	74.78	27.83
	RMS	0.40	0.17	58.28
Northridge	Peak	57.65	57.44	0.36
	RMS	0.16	0.11	31.58
Kobe	Peak	30.19	29.22	3.24
	RMS	0.09	0.07	22.88
Bam	Peak	94.96	90.00	5.22
	RMS	0.26	0.12	52.82
Average response	Peak	71.61	62.86	9.16
	RMS	0.23	0.12	41.39

Table 6.9: Responses to the earthquake excitations (Dis. = Displacement (cm), Vel. = Velocity (m/s), Acc. = Acceleration (m/s^2)).



(a) Ground excitation to roof horizontal displacement



(b) Horizontal force at roof to roof horizontal displacement

Figure 6.17: Typical transfer functions for the high-rise building under different earthquake excitations

6.3.10 Environmental Uncertainties

The performance of the controller is examined under environmental uncertainties. In this regard, the stiffness matrix is multiplied to an uncertainty factor and the peak displacement responses in uncontrolled and controlled are compared as are presented

in Table 6.10.

As it is shown, the performance varies between 18% to 30.36% with an average value of 24.65 % which is close to the value of 27.83 % which were obtained for the original structure under the El Centro earthquake.

Uncertainty	Uncontrolled Dis.	Controlled Dis.	Improvement (%)
-5 %	102.86	75.93	26.18
+5 %	103.96	76.24	26.67
-10 %	101.74	78.44	22.91
+10 %	103.87	76.04	26.79
-15 %	100.31	79.04	21.21
+15 %	103.34	75.32	27.11
-20 %	101.00	80.82	19.99
+20 %	102.65	72.27	29.59
-30 %	104.00	85.28	18.00
+30 %	100.54	70.02	30.36
-40 %	105.28	82.84	21.31
+40 %	97.26	72.23	25.73
	Average		24.65

Table 6.10: Performance of the controller in reducing displacement responses under environmental uncertainties (Dis. = Displacement (m))

Chapter 7

Conclusion

7.1 Summary of Achievements

This dissertation studied the applications of artificial intelligence in structural control problems. The achievements are expressed as follows:

I Vibration Control of Tall Buildings using TMDs

The first part of this dissertation took a deeper look into the utilization of the TMDs in vibration control of tall buildings during earthquake excitations. Although the utilization of TMDs has been extensively studied for decades, their optimal utilization for seismic control of the tall buildings is a challenging task that requires excessive research. As a development in this area, this dissertation has developed a framework to search the optimum arrangement and parameters of the TMDs in tall buildings under earthquake excitation. Based on the obtained results, the following conclusions are drawn:

1. The developed framework has successfully determined the optimal placement and parameters of the TMDs in a tall building given the structural parameters, earthquake excitation, and the assumed limitations for the parameters of the TMDs.

2. Comparing to a single TMD on the roof level, a distributed multi-TMD system has shown a better performance in improving the structural responses with the same amount of mass under earthquake excitations that have noticeable amplitude in the natural frequencies of the structure at the higher modes; a scenario which is likely to be happened within the lifetime of a tall building.

3. The optimum placements of TMDs include:

- In the stories with maximum modal displacements at the lower structural modes.

- In the stories with maximum modal displacements at the modes with the frequencies that the earthquake excitation has also a noticeable amplitude.

4. The optimum parameters of the TMDs that control the vibrations in the lower modes include the maximum allowed damping ratio. This indicates that increasing the damping ratio may improve the performance of the control system.

5. The results show that the TMDs are not good at reducing the initial maximums in the displacement responses compared to the maximums that occurs after some oscillations.

6. Even for the cases with an immediate maximum displacement response, the multiple distributed TMD system has significantly improved the total damping.

II Brain Learning-Based Seismic Control of Structures

In the second part, this research has introduced a novel brain learning-based method for seismic control of the smart structures. In this method, a deep neural network learns how to mitigate the vibrations of a dynamic system subjected to the earthquake excitations. The issues with the current RL methods in solving such structural control problem have been addressed and the method has been improved.

Two case studies were presented and performance of the developed intelligent controller was examined under different earthquake excitations. Moreover, the performance of the controller under environmental uncertainties was studied. Considering the obtained results and discussions in the previous sections, the following conclusions have been drawn:

1. The developed framework has successfully trained a deep neural network to significantly improve the responses of a dynamic system to the earthquake excitations.

2. The developed intelligent controller retains its performance even under earthquake excitations for which it was not trained.

3. The results of the sensitivity analysis show that the controller has a stable performance under environmental uncertainties, which implies the applicability potential of the proposed controlling approach in the real situations.

4. Training the controller by the original mini-batch learning method in RL results in poor performance of the controller in reducing the peak displacement and velocity responses. In addition, a shifting from the origin has been observed in the displacement responses of the frame.

5. The proposed improved mini-batch learning method solves the addressed issues with the original method and significantly improves the performance of the controller. The improved method can be applied to the similar problems that suffer such issues.

7.2 Outlook

In this research, artificial intelligence methods were applied to advance the current methods of seismic control of structures and reinforcement learning. In the first part,

a framework for determining the optimum placement and parameters of the TMDs was introduced and the second part was devoted to introducing a brain learning-based framework for the seismic control of smart structures. The following potential improvements to the current methods are suggested:

1. Combining different types of dampers to improve the seismic behavior of the tall buildings.
2. Considering the earthquakes and wind loads in studying the optimum placement and parameters of the control system in tall buildings.
3. Calibrating the developed framework by performing experimental tests on small scales models.
4. Performing experimental tests on the developed intelligent controller system.
5. Considering nonlinearities effects on the performance of the controller systems.

Bibliography

[1] Wikimedia Commons. File:taipei 101 tuned mass damper he.png — wikimedia commons, the free media repository, 2015. [Online; accessed 9-April-2020].

[2] Tallest Buildings with Dampers. http://www.ctbuh.org/Publications/CTBUHJournal/InNumbers/TallestBuildingswithDampers/tabid/7962/language/en-US/Default.aspx, file = /Users/hamid/Zotero/storage/PVYHLRPF/Default.html.

[3] Ruban Sugumar and T. K. Datta. Stochastic Control of Building frame Using Active Tuned Mass Damper. *dimension*, 7:1.

[4] Naveed Anwar, Thaung Aung, and Fawad Najam. Smart systems for structural response control - an overview. 05 2016.

[5] Oliver Kramer. *Genetic Algorithm Essentials*. Springer Publishing Company, Incorporated, 1st edition, 2017.

[6] PEER Ground Motion Database - PEER Center. https://ngawest2.berkeley.edu/.

[7] B. F. Spencer Jr, Richard E. Christenson, and Shirley J. Dyke. Next generation benchmark control problem for seismically excited buildings. In *Proceedings of the Second World Conference on Structural Control*, volume 2, pages 1135–1360. Kyoto Japan, 1998.

[8] J. T. Yao. CONCEPT OF STRUCTURAL CONTROL. *Journal of the Structural Division*, 98(st 7), July 1972.

[9] B. F. Spencer and M. K. Sain. Controlling buildings: A new frontier in feedback. *IEEE Control Systems Magazine*, 17(6):19–35, December 1997.

[10] T. T. Soong and B. F. Spencer. Supplemental energy dissipation: State-of-the-art and state-of-the-practice. *Engineering Structures*, 24(3):243–259, March 2002.

[11] Fukukita Akira, Saito Tomoo, and Shiba Keiji. Control Effect for 20-Story Benchmark Building Using Passive or Semiactive Device. *Journal of Engineering Mechanics*, 130(4):430–436, April 2004.

[12] Chi-Chang Lin, Jin-Min Ueng, and Teng-Ching Huang. Seismic response reduction of irregular buildings using passive tuned mass dampers. *Engineering Structures*, 22(5):513–524, 2000.

[13] Jabbari Faryar, Schmitendorf W. E., and Yang J. N. H∞ Control for Seismic-Excited Buildings with Acceleration Feedback. *Journal of Engineering Mechanics*, 121(9):994–1002, September 1995.

[14] Hua Jun Li, Sau-Lon James Hu, and Christopher Jakubiak. H2 active vibration control for offshore platform subjected to wave loading. *Journal of Sound and Vibration*, 263(4):709–724, June 2003.

[15] C. Oktay Azeloglu, Ahmet Sagirli, and Ayse Edincliler. Vibration mitigation of nonlinear crane system against earthquake excitations with the self-tuning fuzzy logic PID controller. *Nonlinear Dynamics*, 84(4):1915–1928, June 2016.

[16] S. N. Deshmukh and N. K. Chandiramani. LQR Control of Wind Excited Benchmark Building Using Variable Stiffness Tuned Mass Damper. *Shock and Vibration*, 2014:1–12, 2014.

[17] N. R. Fisco and H. Adeli. Smart structures: Part II — Hybrid control systems and control strategies. *Scientia Iranica*, 18(3):285–295, 2011.

[18] N. R. Fisco and H. Adeli. Smart structures: Part I—Active and semi-active control. *Scientia Iranica*, 18(3):275–284, 2011.

[19] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, Limited, August 2009.

[20] David Lynton Poole, Alan K. Mackworth, and Randy Goebel. *Computational Intelligence: A Logical Approach*, volume 1. Oxford University Press New York, 1998.

[21] High-level perception, representation, and analogy: A critique of artificial intelligence methodology: Journal of Experimental & Theoretical Artificial Intelligence: Vol 4, No 3.

[22] Artificial Intelligence in Medicine | NEJM. https://www.nejm.org/doi/pdf/10.1056/NEJM198703123161109.

[23] Monika Hengstler, Ellen Enkel, and Selina Duelli. Applied artificial intelligence and trust—The case of autonomous vehicles and medical assistance devices. *Technological Forecasting and Social Change*, 105:105–120, April 2016.

[24] Andrew Y. Ng, Adam Coates, Mark Diel, Varun Ganapathi, Jamie Schulte, Ben Tse, Eric Berger, and Eric Liang. Autonomous Inverted Helicopter Flight via Reinforcement Learning. In Marcelo H. Ang and Oussama Khatib, editors, *Experimental Robotics IX*, Springer Tracts in Advanced Robotics, pages 363–372. Springer Berlin Heidelberg, 2006.

[25] Housner G. W., Bergman L. A., Caughey T. K., Chassiakos A. G., Claus R. O., Masri S. F., Skelton R. E., Soong T. T., Spencer B. F., and Yao J. T. P. Structural Control: Past, Present, and Future. *Journal of Engineering Mechanics*, 123(9):897–971, September 1997.

[26] Mir M. Ali and Kyoung Sun Moon. Structural Developments in Tall Buildings: Current Trends and Future Prospects. *Architectural Science Review*, 50(3):205–223, September 2007.

[27] Hermann Frahm. Device for damping vibrations of bodies, April 1911.

[28] Quan Li, Jiansheng Fan, Jianguo Nie, Quanwang Li, and Yu Chen. Crowd-induced random vibration of footbridge and vibration control using multiple tuned mass dampers. *Journal of Sound and Vibration*, 329(19):4068–4092, September 2010.

[29] Moon Seok J., Bergman Lawrence A., and Voulgaris Petros G. Sliding Mode Control of Cable-Stayed Bridge Subjected to Seismic Excitation. *Journal of Engineering Mechanics*, 129(1):71–78, January 2003.

[30] Nam Hoang, Yozo Fujino, and Pennung Warnitchai. Optimal tuned mass damper for seismic applications and practical design formulas. *Engineering Structures*, 30(3):707–715, March 2008.

[31] A. Casalotti, A. Arena, and W. Lacarbonara. Mitigation of post-flutter oscillations in suspension bridges by hysteretic tuned mass dampers. *Engineering Structures*, 69:62–71, June 2014.

[32] JF Wang, CC Lin, and BL Chen. Vibration suppression for high-speed railway bridges using tuned mass dampers. *International Journal of Solids and Structures*, 40(2):465–491, 2003.

[33] José Luis Almazán, Gilda Espinoza, and Juan Jesús Aguirre. Torsional balance of asymmetric structures by means of tuned mass dampers. *Engineering Structures*, 42:308–328, September 2012.

[34] Genda Chen and Jingning Wu. Experimental study on multiple tuned mass dampers to reduce seismic responses of a three-storey building structure. *Earthquake Engineering & Structural Dynamics*, 32(5):793–810, April 2003.

[35] Y Arfiadi and MNS Hadi. Optimum placement and properties of tuned mass dampers using hybrid genetic algorithms. *Iran University of Science & Technology*, 1(1):167–187, 2011.

[36] Said Elias and Vasant Matsagar. Distributed Multiple Tuned Mass Dampers for Wind Vibration Response Control of High-Rise Building. *Journal of Engineering*, 2014:1–11, 2014.

[37] Yozo Fujino and Masato Abe. Design formulas for tuned mass dampers based on a perturbation technique. *Earthquake Engineering & Structural Dynamics*, 22(10):833–854, 1993.

[38] K.C.S. Kwok and B. Samali. Performance of tuned mass dampers under wind loads. *Engineering Structures*, 17(9):655–667, November 1995.

[39] A. Y. T. Leung and Haijun Zhang. Particle swarm optimization of tuned mass dampers. *Engineering Structures*, 31(3):715–728, 2009.

[40] Chunxiang Li. Performance of multiple tuned mass dampers for attenuating undesirable oscillations of structures under the ground acceleration. *Earthquake Engineering & Structural Dynamics*, 29(9):1405–1421, September 2000.

[41] Chien-Liang Lee, Yung-Tsang Chen, Lap-Loi Chung, and Yen-Po Wang. Optimal design theories and applications of tuned mass dampers. *Engineering Structures*, 28(1):43–53, 2006.

[42] CC Chang and WL Qu. Unified dynamic absorber design formulas for wind-induced vibration control of tall buildings. *The structural design of tall buildings*, 7(2):147–166, 1998.

[43] A.S. Joshi and R.S. Jangid. OPTIMUM PARAMETERS OF MULTIPLE TUNED MASS DAMPERS FOR BASE-EXCITED DAMPED SYSTEMS. *Journal of Sound and Vibration*, 202(5):657–667, May 1997.

[44] Chi-Chang Lin, Lyan-Ywan Lu, Ging-Long Lin, and Ting-Wei Yang. Vibration control of seismic structures using semi-active friction multiple tuned mass dampers. *Engineering Structures*, 32(10):3404–3417, 2010.

[45] AG Thompson. Optimum tuning and damping of a dynamic vibration absorber applied to a force excited and damped primary system. *Journal of Sound and Vibration*, 77(3):403–415, 1981.

[46] Hsiang-Chuan Tsai and Guan-Cheng Lin. Optimum tuned-mass dampers for minimizing steady-state response of support-excited and damped systems. *Earthquake engineering & structural dynamics*, 22(11):957–973, 1993.

[47] Jer-Fu Wang and Chi-Chang Lin. Seismic performance of multiple tuned mass dampers for soil–irregular building interaction systems. *International Journal of Solids and Structures*, 42(20):5536–5554, October 2005.

[48] GB Warburton and EO Ayorinde. Optimum absorber parameters for simple systems. *Earthquake Engineering & Structural Dynamics*, 8(3):197–217, 1980.

[49] HJ Lin, SW Zhang, KD Sun, Q He, JR Lark, and WF Williams. Precise and efficient computation of complex structures with TMD devices. *Journal of sound and vibration*, 223(5):693–701, 1999.

[50] S Pourzeynali, S Salimi, and H Eimani Kalesar. Robust multi-objective optimization design of TMD control device to reduce tall building responses against earthquake excitations using genetic algorithms. *Scientia Iranica*, 20(2):207–221, 2013.

[51] Ali Bozer and Şaban S. Özsarıyıldız. Free parameter search of multiple tuned mass dampers by using artificial bee colony algorithm. *Structural Control and Health Monitoring*, 25(2):e2066, 2018.

[52] M. Jokic, M. Stegic, and M. Butkovic. Reduced-order multiple tuned mass damper optimization: A bounded real lemma for descriptor systems approach. *Journal of Sound and Vibration*, 330(22):5259–5268, 2011.

[53] Alka Y. Pisal and R. S. Jangid. Seismic response of multi-story structure with multiple tuned mass friction dampers. *International Journal of Advanced Structural Engineering (IJASE)*, 6(1):1–13, 2014.

[54] Hong-Nan Li and Xiang-Lei Ni. Optimization of non-uniformly distributed multiple tuned mass damper. *Journal of Sound and Vibration*, 308(1):80–97, 2007.

[55] Chunxiang Li and Bilei Zhu. Estimating double tuned mass dampers for structures under ground acceleration using a novel optimum criterion. *Journal of Sound and Vibration*, 298(1-2):280–297, 2006.

[56] Genda Chen and Jingning Wu. Optimal Placement of Multiple Tune Mass Dampers for Seismic Structures. *Journal of Structural Engineering*, 127(9):1054–1062, 2001.

[57] Chunxiang Li and Weilian Qu. Optimum properties of multiple tuned mass dampers for reduction of translational and torsional response of structures subject to ground acceleration. *Engineering Structures*, 28(4):472–494, 2006.

[58] Tharwat A. Sakr. Vibration control of buildings by using partial floor loads as multiple tuned mass dampers. *HBRC Journal*.

[59] M. P. Singh, E. E. Matheu, and L. E. Suarez. Active and Semi-Active Control of Structures Under Seismic Excitation. *Earthquake Engineering & Structural Dynamics*, 26(2):193–213.

[60] Wu J. C. and Yang J. N. Active Control of Transmission Tower under Stochastic Wind. *Journal of Structural Engineering*, 124(11):1302–1312, November 1998.

[61] A. Saleh and H. Adeli. Optimal control of adaptive building structures under blast loading. *Mechatronics*, 8(8):821–844, December 1998.

[62] Masashi Yamamoto, Satoru Aizawa, Masahiko Higashino, and Kotaro Toyama. Practical applications of active mass dampers with hydraulic actuator. *Earthquake Engineering & Structural Dynamics*, 30(11):1697–1717.

[63] Yoshiki Ikeda, Katsuyasu Sasaki, Mitsuo Sakamoto, and Takuji Kobori. Active mass driver system as the first application of active structural control. *Earthquake Engineering & Structural Dynamics*, 30(11):1575–1595.

[64] Mahdi Abdollahirad, Arcan Yanik, and Unal Aldemir. Wavelet PSO-Based LQR Algorithm for Optimal Structural Control Using Active Tuned Mass Dampers. In *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V008T13A002–V008T13A002. American Society of Mechanical Engineers, August 2015.

[65] D. Iba, J. Hongu, T. Sasaki, S. Shima, M. Nakamura, and I. Moriwaki. Active mass damper system for high-rise buildings using neural oscillator and position controller: Sinusoidally varying desired displacement of auxiliary mass

intended for reduction of maximum control force. In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2017*, volume 10168, page 101683F. International Society for Optics and Photonics, April 2017.

[66] Y. Achkire and A. Preumont. Active Tendon Control of Cable-Stayed Bridges. *Earthquake Engineering & Structural Dynamics*, 25(6):585–597, June 1996.

[67] Khaldoon Bani-Hani and Jamshid Ghaboussi. Neural networks for structural control of a benchmark problem, active tendon system. *Earthquake Engineering & Structural Dynamics*, 27(11):1225–1245, November 1998.

[68] Frédéric Bossens and André Preumont. Active tendon control of cable-stayed bridges: A large-scale demonstration. *Earthquake Engineering & Structural Dynamics*, 30(7):961–979, July 2001.

[69] Pennung Warnitchai, Yozo Fujino, Benito M. Pacheco, and Remi Agret. An experimental study on active tendon control of cable-stayed bridges. *Earthquake Engineering & Structural Dynamics*, 22(2):93–111, February 1993.

[70] J. Rodellar, V. Mañosa, and C. Monroy. An active tendon control scheme for cable-stayed bridges with model uncertainties and seismic excitation. *Journal of Structural Control*, 9(1):75–94.

[71] Andre Preumont, Matteo Voltan, Andrea Sangiovanni, Bilal Mokrani, and David Alaluf. Active tendon control of suspension bridges. *Smart Structures and Systems*, 18(1):31–52, July 2016.

[72] Ebrahim Nazarimofrad and Seyed Mehdi Zahrai. Seismic control of irregular multistory buildings using active tendons considering soil–structure interaction effect. *Soil Dynamics and Earthquake Engineering*, 89:100–115, 2016.

[73] Jung Hyung-Jo, Spencer Billie F., and Lee In-Won. Control of Seismically Excited Cable-Stayed Bridge Employing Magnetorheological Fluid Dampers. *Journal of Structural Engineering*, 129(7):873–883, July 2003.

[74] Xu Y. L., Chen J., Ng C. L., and Qu W. L. Semiactive Seismic Response Control of Buildings with Podium Structure. *Journal of Structural Engineering*, 131(6):890–899, June 2005.

[75] Soong T. T. and Spencer B. F. Active Structural Control: Theory and Practice. *Journal of Engineering Mechanics*, 118(6):1282–1285, June 1992.

[76] Gene F. Franklin, J. David Powell, Abbas Emami-Naeini, and J. David Powell. *Feedback Control of Dynamic Systems*, volume 3. Addison-Wesley Reading, MA, 1994.

[77] Katsuhiko Ogata. *Modern Control Engineering*. Prentice Hall, 2010.

[78] Hojjat Adeli and Hongjin Kim. Wavelet-Based Vibration Control of Smart Buildings and Bridges. https://www.crcpress.com/Wavelet-Based-Vibration-Control-of-Smart-Buildings-and-Bridges/Adeli-Kim/p/book/9781420089233.

[79] Draguna Vrabie, Kyriakos G. Vamvoudakis, and Frank L. Lewis. *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*. INST OF ELECTRICAL ENGINE, London, October 2012.

[80] Vahid Hassani, Tegoeh Tjahjowidodo, and Thanh Nho Do. A survey on hysteresis modeling, identification and control. *Mechanical Systems and Signal Processing*, 49:209–233, December 2014.

[81] Richard C. Dorf and Robert H. Bishop. *Modern Control Systems*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 9th edition, 2000.

[82] Richard C. Dorf and Andrew Kusiak. Handbook of Design, Manufacturing and Automation. https://www.wiley.com/en-us/Handbook+of+Design%2C+Manufacturing+and+Automation-p-9780471552185.

[83] K. Dhanalakshmi. Differential resistance feedback control of a self-sensing shape memory alloy actuated system. *ISA transactions*, 53(2):289–297, 2014.

[84] Steven Ian Moore and SO Reza Moheimani. Vibration control with MEMS electrostatic drives: A self-sensing approach. *IEEE Transactions on Control Systems Technology*, 23(3):1237–1244, 2015.

[85] Mark J. Balas. Direct velocity feedback control of large space structures. *Journal of Guidance, Control, and Dynamics*, 2(3):252–253, 1979.

[86] Garrett Nelson, Rajesh Rajamani, Andrew Gastineau, Steven F. Wojtkiewicz, and Arturo E. Schultz. Bridge life extension using semiactive vibration control. *IEEE/ASME Transactions on Mechatronics*, 20(1):207–216, 2015.

[87] Peng Li, Luyu Li, Gangbing Song, and Yan Yu. Wireless sensing and vibration control with increased redundancy and robustness design. *IEEE transactions on cybernetics*, 44(11):2076–2087, 2014.

[88] J. M. Rodriguez-Fortun, J. Orus, J. Alfonso, Francisco Buil Gimeno, and Jose A. Castellanos. Flatness-based active vibration control for piezoelectric actuators. *IEEE/ASME Transactions on Mechatronics*, 18(1):221–229, 2013.

[89] Shengquan Li, Juan Li, and Yueping Mo. Piezoelectric multimode vibration control for stiffened plate using ADRC-based acceleration compensation. *IEEE Transactions on Industrial Electronics*, 61(12):6892–6902, 2014.

[90] Huibert Kwakernaak and Raphael Sivan. *Linear Optimal Control Systems*, volume 1. Wiley-interscience New York, 1972.

[91] H. Adeli and A. Saleh. Optimal control of adaptive/smart bridge structures. *Journal of Structural Engineering*, 123(2):218–226, 1997.

[92] Eduardo D. Sontag. *Mathematical Control Theory: Deterministic Finite Dimensional Systems*, volume 6. Springer Science & Business Media, 2013.

[93] Karl J. \AAström. *Introduction to Stochastic Control Theory*. Courier Corporation, 2012.

[94] Dan Simon. *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. John Wiley & Sons, 2006.

[95] Hongzhe Dai, Hao Zhang, and Wei Wang. A multiwavelet neural network-based response surface method for structural reliability analysis. *Computer-Aided Civil and Infrastructure Engineering*, 30(2):151–162, 2015.

[96] Ashutosh Nayyar, Aditya Mahajan, and Demosthenis Teneketzis. Decentralized stochastic control with partial history sharing: A common information approach. *IEEE Transactions on Automatic Control*, 58(7):1644–1658, 2013.

[97] Xuchao Lin and Christos G. Cassandras. An optimal control approach to the multi-agent persistent monitoring problem in two-dimensional spaces. *IEEE Transactions on Automatic Control*, 60(6):1659–1664, 2015.

[98] James Blake Rawlings and David Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Pub. Madison, Wisconsin, 2009.

[99] S. Joe Qin and Thomas A. Badgwell. A survey of industrial model predictive control technology. *Control engineering practice*, 11(7):733–764, 2003.

[100] Gang Mei, Ahsan Kareem, and Jeffrey C. Kantor. Model predictive control of structures under earthquakes using acceleration feedback. *Journal of engineering Mechanics*, 128(5):574–585, 2002.

[101] Arne Koerber and Rudibert King. Combined feedback–feedforward control of wind turbines using state-constrained model predictive control. *IEEE Transactions on Control Systems Technology*, 21(4):1117–1128, 2013.

[102] Vikas Chandan and Andrew Alleyne. Optimal partitioning for the decentralized thermal control of buildings. *IEEE Transactions on Control Systems Technology*, 21(5):1756–1770, 2013.

[103] Stefano Riverso, Marcello Farina, and Giancarlo Ferrari-Trecate. Plug-and-play decentralized model predictive control for linear systems. *IEEE Transactions on Automatic Control*, 58(10):2608–2614, 2013.

[104] Su Liu, Jinfeng Liu, Yiping Feng, and Gang Rong. Performance assessment of decentralized control systems: An iterative approach. *Control Engineering Practice*, 22:252–263, 2014.

[105] Josue Enríquez-Zárate1 Guillermo Valencia-Palomo, Francisco-Ronay López-Estrada, Gerardo Silva-Navarro, and José Antonio Hoyo-Montaño. Efficient predictive vibration control of a building-like structure. 2019.

[106] William S. Levine. *Control System Fundamentals*. CRC press, 2019.

[107] Jürgen Ackermann. *Robuste Regelung: Analyse Und Entwurf von Linearen Regelungssystemen Mit Unsicheren Physikalischen Parametern*. Springer-Verlag, 2013.

[108] John C. Doyle, Keith Glover, Pramod P. Khargonekar, and Bruce A. Francis. State-space solutions to standard H/sub 2/and H/sub infinity/control problems. *IEEE Transactions on Automatic control*, 34(8):831–847, 1989.

[109] Yang Wang. Time-delayed dynamic output feedback $\mathscr{H}$ïž∞ controller design for civil structures: A decentralized approach through homotopic transformation. *Structural Control and Health Monitoring*, 18(2):121–139, 2011.

[110] Arash Yeganeh Fallah and Touraj Taghikhany. Time-delayed decentralized H2/LQG controller for cable-stayed bridge under seismic loading. *Structural Control and Health Monitoring*, 20(3):354–372, 2013.

[111] Nengmou Wang and Hojjat Adeli. Algorithms for chattering reduction in system control. *Journal of the Franklin Institute*, 349(8):2687–2703, 2012.

[112] Nengmou Wang and Hojjat Adeli. Robust vibration control of wind-excited highrise building structures. *Journal of Civil Engineering and Management*, 21(8):967–976, 2015.

[113] Mehdi Soleymani, Amir Hossein Abolmasoumi, Hasanali Bahrami, Arash Khalatbari-S, Elham Khoshbin, and Sirus Sayahi. Modified sliding mode control of a seismic active mass damper system considering model uncertainties and input time delay. *Journal of Vibration and Control*, 24(6):1051–1064, 2018.

[114] Petar V. Kokotovic. The joy of feedback: Nonlinear and adaptive. *IEEE Control Systems Magazine*, 12(3):7–17, 1992.

[115] Rodolphe Sepulchre, Mrdjan Jankovic, and Petar V. Kokotovic. *Constructive Nonlinear Control*. Springer Science & Business Media, 2012.

[116] Chenliang Wang and Yan Lin. Decentralized adaptive tracking control for a class of interconnected nonlinear time-varying systems. *Automatica*, 54:16–24, 2015.

[117] Delphine Bresch-Pietri, Jonathan Chauvin, and Nicolas Petit. Prediction-based stabilization of linear systems subject to input-dependent input delay of integral-type. *IEEE Transactions on Automatic Control*, 59(9):2385–2399, 2014.

[118] Huijin Fan, Bing Liu, Yindong Shen, and Wei Wang. Adaptive failure compensation control for uncertain systems with stochastic actuator failures. *IEEE Transactions on Automatic Control*, 59(3):808–814, 2014.

[119] Alexander S. Poznyak, Edgar N. Sanchez, and Wen Yu. *Differential Neural Networks for Robust Nonlinear Control: Identification, State Estimation and Trajectory Tracking*. World Scientific, 2001.

[120] Jyh-Shing Roger Jang, Chuen-Tsai Sun, and Eiji Mizutani. Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. *IEEE Transactions on automatic control*, 42(10):1482–1484, 1997.

[121] Mo Jamshidi and Ali Zilouchian. *Intelligent Control Systems Using Soft Computing Methodologies*. CRC press, 2001.

[122] F. Ali Sk. Semi-active control of earthquake induced vibrations in structures using MR dampers: Algorithm development, experimental verification and benchmark applications. *Bangalore, India*, 2008.

[123] A. S. Ahlawat and A. Ramaswamy. Multiobjective optimal structural vibration control using fuzzy logic control system. *Journal of Structural Engineering*, 127(11):1330–1337, 2001.

[124] A. S. Ahlawat and A. Ramaswamy. Multiobjective optimal absorber system for torsionally coupled seismically excited structures. *Engineering Structures*, 25(7):941–950, 2003.

[125] A. S. Ahlawat and A. Ramaswamy. Multiobjective optimal fuzzy logic controller driven active and hybrid control systems for seismically excited nonlinear buildings. *Journal of engineering mechanics*, 130(4):416–423, 2004.

[126] Hyun-Su Kim and Paul N Roschke. Design of fuzzy logic controller for smart base isolation system using genetic algorithm. *Engineering Structures*, 28(1):84–96, 2006.

[127] Pelayo Quirós, Pedro Alonso, Irene Díaz, and Susana Montes. On the use of fuzzy partitions to protect data. *Integrated Computer-Aided Engineering*, 21(4):355–366, 2014.

[128] Bijan Samali and Mohammed Al-Dawod. Performance of a five-storey benchmark model using an active tuned mass damper and a fuzzy controller. *Engineering Structures*, 25(13):1597–1610, 2003.

[129] Mehdi Soleymani and Masoud Khodadadi. Adaptive fuzzy controller for active tuned mass damper of a benchmark tall building subjected to seismic and wind loads. *The Structural Design of Tall and Special Buildings*, 23(10):781–800, 2014.

[130] Jeremie Cabessa and Hava T Siegelmann. The super-Turing computational power of plastic recurrent neural networks. *International journal of neural systems*, 24(08):1450029, 2014.

[131] Francesco Donnarumma, Roberto Prevete, Fabian Chersi, and Giovanni Pezzulo. A programmer–interpreter neural network architecture for prefrontal cognitive control. *International journal of neural systems*, 25(06):1550017, 2015.

[132] María del Mar Martínez Ballesteros. Evolutionary algorithms to discover quantitative association rules. 2011.

[133] Jixiang Cheng, Gexiang Zhang, Fabio Caraffini, and Ferrante Neri. Multicriteria adaptive differential evolution for global numerical optimization. *Integrated Computer-Aided Engineering*, 22(2):103–107, 2015.

[134] Di You, Carlos Fabian Benitez-Quiroz, and Aleix M Martinez. Multiobjective optimization for model selection in kernel methods in regression. *IEEE transactions on neural networks and learning systems*, 25(10):1879–1893, 2014.

[135] Pablo Mesejo, Oscar Ibánez, Enrique Fernández-Blanco, Francisco Cedrón, Alejandro Pazos, and Ana B Porto-Pazos. Artificial neuron–glia networks learning approach based on cooperative coevolution. *International journal of neural systems*, 25(04):1550012, 2015.

[136] Gang Tao. *Adaptive Control Design and Analysis*, volume 37. John Wiley & Sons, 2003.

[137] Hongjin Kim and Hojjat Adeli. Hybrid feedback-least mean square algorithm for structural control. *Journal of Structural Engineering*, 130(1):120–127, 2004.

[138] Wu Caiyun. Active noise control based on adaptive bilinear FL algorithm. In *2012 International Conference on Computer Distributed Control and Intelligent Environmental Monitoring*, pages 89–92. IEEE, 2012.

[139] Ziqin Zhou and Hojjat Adeli. Time-frequency signal analysis of earthquake records using Mexican hat wavelets. *Computer-Aided Civil and Infrastructure Engineering*, 18(5):379–389, 2003.

[140] Hojjat Adeli and Hongjin Kim. *Wavelet-Based Vibration Control of Smart Buildings and Bridges*. CRC Press Taylor & Francis, Boca Raton, Florida, 2009.

[141] Fereidoun Amini and Masoud Zabihi Samani. A wavelet-based adaptive pole assignment method for structural control. *Computer-Aided Civil and Infrastructure Engineering*, 29(6):464–477, 2014.

[142] Nengmou Wang and Hojjat Adeli. Self-constructing wavelet neural network algorithm for nonlinear control of large structures. *Engineering Applications of Artificial Intelligence*, 41:249–258, 2015.

[143] John H Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. U Michigan Press, 1975.

[144] Peter J. Fleming and Robin C. Purshouse. Evolutionary algorithms in control systems engineering: A survey. *Control engineering practice*, 10(11):1223–1241, 2002.

[145] C. S. Chang and S. S. Sim. Optimising train movements through coast control using genetic algorithms. *IEE Proceedings-Electric Power Applications*, 144(1):65–73, 1997.

[146] Jinwoo Kim, Yoonkeon Moon, and Bernard P. Zeigler. Designing fuzzy net controllers using genetic algorithms. *IEEE control systems Magazine*, 15(3):66–72, 1995.

[147] S. Pourzeynali, H. H. Lavasani, and A. H. Modarayi. Active control of high rise building structures using fuzzy logic and genetic algorithms. *Engineering Structures*, 29(3):346–357, 2007.

[148] Andrew YT Leung, Haijun Zhang, C. C. Cheng, and Y. Y. Lee. Particle swarm optimization of TMD by non-stationary base excitation during earthquake. *Earthquake Engineering & Structural Dynamics*, 37(9):1223–1246, 2008.

[149] Magdalene Marinaki, Yannis Marinakis, and Georgios E. Stavroulakis. Fuzzy control optimized by a multi-objective particle swarm optimization algorithm for vibration suppression of smart structures. *Structural and Multidisciplinary Optimization*, 43(1):29–42, 2011.

[150] Mohammad Najafzadeh. Neuro-fuzzy GMDH based particle swarm optimization for prediction of scour depth at downstream of grade control structures. *Engineering Science and Technology, an International Journal*, 18(1):42–51, 2015.

[151] Majid Zamani, Nasser Sadati, and Masoud Karimi Ghartemani. Design of an H∞ PID controller using particle swarm optimization. *International Journal of Control, Automation and Systems*, 7(2):273–280, 2009.

[152] Liang Bai, Yun-Wen Feng, Ning Li, and Xiao-Feng Xue. Optimal fuzzy iterative learning control based on artificial bee colony for vibration control of piezoelectric smart structures. *Journal of Vibroengineering*, 21(1), 2019.

[153] K. Naidu, H. Mokhlis, and AH Abu Bakar. Multiobjective optimization using weighted sum artificial bee colony algorithm for load frequency control. *International Journal of Electrical Power & Energy Systems*, 55:657–667, 2014.

[154] S. N. Omkar, J. Senthilnath, Rahul Khandelwal, G. Narayana Naik, and S. Gopalakrishnan. Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures. *Applied Soft Computing*, 11(1):489–499, 2011.

[155] Jagatheesan Kaliannan, Anand Baskaran, Nilanjan Dey, and Amira S. Ashour. Ant colony optimization algorithm based PID controller for LFC of single area power system with non-linearity and boiler dynamics. *World J. Model. Simul*, 12(1):3–14, 2016.

[156] Bani-Hani Khaldoon and Ghaboussi Jamshid. Nonlinear Structural Control Using Neural Networks. *Journal of Engineering Mechanics*, 124(3):319–327, March 1998.

[157] HM Chen, KH Tsai, GZ Qi, JCS Yang, and F Amini. Neural network for structure control. *Journal of Computing in Civil Engineering*, 9(2):168–176, 1995.

[158] Jamshid Ghaboussi and Abdolreza Joghataie. Active control of structures using neural networks. *Journal of Engineering Mechanics*, 1995.

[159] Alok Madan. Vibration control of building structures using self-organizing and self-learning neural networks. *Journal of sound and vibration*, 287(4):759–784, 2005.

[160] Andrew Y. Ng, Adam Coates, Mark Diel, Varun Ganapathi, Jamie Schulte, Ben Tse, Eric Berger, and Eric Liang. Autonomous Inverted Helicopter Flight via Reinforcement Learning. In Marcelo H. Ang and Oussama Khatib, editors, *Experimental Robotics IX*, Springer Tracts in Advanced Robotics, pages 363–372. Springer Berlin Heidelberg, 2006.

[161] Long-Ji Lin. Reinforcement Learning for Robots Using Neural Networks. Technical Report CMU-CS-93-103, CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, January 1993.

[162] Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, page 0278364913495721, 2013.

[163] Patrick Mannion, Jim Duggan, and Enda Howley. Parallel Reinforcement Learning for Traffic Signal Control. *Procedia Computer Science*, 52:956–961, 2015.

[164] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm. *arXiv:1712.01815 [cs]*, December 2017.

[165] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, October 2017.

[166] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002.

[167] Melanie Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, 1998.

[168] Darrell Whitley. A genetic algorithm tutorial. *Statistics and Computing*, 4(2):65–85, June 1994.

[169] Mitsuo Gen and Runwei Cheng. *Genetic Algorithms and Engineering Optimization*. John Wiley & Sons, 2000.

[170] Kazi Shah Nawaz Ripon, Sam Kwong, and K. F. Man. A real-coding jumping gene genetic algorithm (RJGGA) for multiobjective optimization. *Information Sciences*, 177(2):632–654, January 2007.

[171] Günther R. Raidl and Bryant A. Julstrom. A Weighted Coding in a Genetic Algorithm for the Degree-constrained Minimum Spanning Tree Problem. In *Proceedings of the 2000 ACM Symposium on Applied Computing - Volume 1*, SAC '00, pages 440–445, New York, NY, USA, 2000. ACM.

[172] A Comparative Analysis of Selection Schemes Used in Genetic Algorithms. *Foundations of Genetic Algorithms*, 1:69–93, January 1991.

[173] J. N. Yang, B. Samali, and J. Wu. Benchmark Problem for Response Control Wind excited Tall Buildings. *Journal of Engineering Mechanics*, 130(4):437–446, 2004.

[174] Nathan M. Newmark. A Method of Computation for Structural Dynamics. *Journal of the Engineering Mechanics Division*, 85(3):67–94, 1959.

[175] American Society of Civil Engineers. *Minimum Design Loads for Buildings and Other Structures: Third Printing*. American Society of Civil Engineers, October 2013.

[176] Andrew G. Barto Richard S. Sutton. *Reinforcement Learning, an Introduction*. The MIT Press, 1998.

[177] Marco Wiering and Martijn van Otterlo. *Reinforcement Learning: State-of-the-Art*. Springer Science & Business Media, March 2012.

[178] Christopher Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer-Verlag, New York, 2006.

[179] Md Rezaul Karim and Sridhar Alla. *Scala and Spark for Big Data Analytics: Explore the Concepts of Functional Programming, Data Streaming, and Machine Learning*. Packt Publishing Ltd, July 2017.

[180] Marco Wiering and Jürgen Schmidhuber. Efficient model-based exploration. In *Proceedings of the Sixth International Conference on Simulation of Adaptive Behavior: From Animals to Animats*, volume 6, pages 223–228, 1998.

[181] Martijn Van Otterlo and Marco Wiering. Reinforcement learning and markov decision processes. In *Reinforcement Learning*, pages 3–42. Springer, 2012.

[182] Marco Wiering and Jürgen Schmidhuber. HQ-learning. *Adaptive Behavior*, 6(2):219–246, 1997.

[183] B. Abdulhai and L. Kattan. Reinforcement learning: Introduction to theory and potential for transport application. *Can. J. Civ. Eng.*, 30:981–991, 2003.

[184] Ronald J. Williams. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. In *Reinforcement Learning*, The Springer International Series in Engineering and Computer Science, pages 5–32. Springer, Boston, MA, 1992.

[185] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 1. Athena scientific Belmont, MA, 1995.

[186] Richard S. Sutton and Andrew G. Barto. The MIT Press, 2012.

[187] J. N. Tsitsiklis and B. Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5):674–690, May 1997.

[188] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, February 2015.

[189] J. N. Yang and A. Akbarpour. Effect of system uncertainty on control of seismic-excited buildings. *Journal of Engineering Mechanics*, 116(2):462–478, 1990.

Appendix A

Developed Algorithms

A.1 Adapted Crossover Algorithm

The crossover function is developed to create new offsprings given the chromosomes of the parents. The algorithm uses two-variants:

- **Variant 1:** The operator acts on the parameters of the TMDs separately using the k-point crossover method, meaning that in each call, the crossover operator affects the stiffness, mass or damping of the parents by exchanging the related properties using the k-point crossover function. The produced off-springs have TMDs in the same locations of the parents, but with different parameters. This function improves the parameters of the TMDs regardless of their positions.
- **Variant 2:** This function acts on the corresponding gens to the location of TMDs in the parents. The resultant off-springs contain TMDs with the same parameters of their parents but in different locations of the building.

Algorithm A.1 Crossover operator

```
procedure CrossOver(individual1, individual2)
if rnd(1) < 0.5
   rndTmd1 = RandomSelect TMD in individual1
   rndTmd2 = RandomSelect TMD in individual2
   k = rndint(3)
   tmd1New = K-PointCrossover(rndTmd1)
   tmd2New = K-PointCrossover(rndTmd2)
   offspring1 = Rebuild(individual1, tmd1New)
   offspring2 = Rebuild(individual2, tmd2New)
else
   rndTmd1 = randomselect TMD in individual1
   rndTmd2 = RandomSelect TMD in individual2
   tmd1New = rndTmd2
   tmd2New = rndTmd1
   offspring1 = Rebuild(individual1, tmd1New)
   offspring2 = Rebuild(individual2, tmd2New)
end if
return (offspring1, offspring2)
end procedure
```

A.2 Adapted Mutation Algorithm

The two-variant mutation function is developed to affect the (1) genes related to TMD parameters and (2) set of genes which are related to the location of the TMDs.

Algorithm A.2 Mutation operator

```
procedure Mutation(individual)
if rnd(1) < 0.5
   rndTmd = RandomSelect TMD in individual
   newStory = rndint(76)
   tmdNew = MoveTmd(rndTmd, newStory)
   offspring = Rebuild(individual, tmdNew)
else
   rndTmd = RandomSelect TMD in individual
   argetGenCount = rndint(10)
   for i=1 to targetGenCount
     targetGen = RandomSelectGen(rndTmd)
     mutTmd = BinaryMutation(rndTmd, targetGen)
   end for
   offspring = Rebuild(individual, tmdNew)
end if
return (offspring)
end procedure
```

A.3 Framework Basic Algorithm

The basic algorithm of the framework includes a major loop in which the generations are evolved. In the first iteration, the algorithm generates random chromosomes to form an initial population. In each iteration, the algorithm adds new chromosomes by performing genetic operations on the current members and pass them to the *analyze* module to determine the controlled responses of the structure. After that, the *fitness* module, determines the fitness values for each chromosome and pass the population to NSGA-II module which utilizes a refined history of the TMD arrangements and corresponding fitness values, to develop the paretos and sort the chromosomes upon that.

Algorithm A.3 Basic algorithm of framework

```
population = InitializePopulation()
repeat
  repeat
    {parent1, parent2} = MakeSelection (population)
    if rnd(1) ≥ crossoverProbability
      {offspring1, offspring2} = Crossover(parent1, parent2)
      Repair(offsprint1, offspring2)
      ComputeFitness(offsprint1, offspring2)
    end if
    if rnd(1) ≥ mutationProbability
      {offspring1, offspring2} = Mutation(parent1, parent2)
      Repair(offsprint1, offspring2)
      ComputeFitness(offsprint1, offspring2)
    end if
  until size(population) ≤ M
  tempPopulation = population
  newPopulation = [ ]
  repeat
    p = ParetoFront(tempPopulation)
    ps = CrowdingDistance(p)
    newPopulation=newPopulation + ps
    tempPopulation = tempPopulation - ps
  until size(tempPopulation) > 2
  population = newPopulation
until generationNumber ≤ N
```

A.4 Key States Algorithm

This algorithms adds a random selection of key states to the training batch in each learning, which itself is a collection of the randomly selected states.

The key states are defined as:

- States with a very low immediate reward values. These states are related to the maximum responses so it worth to try different actions in such states.

- States with a high reward value which targets the states in which the difference of the uncontrolled and controlled responses is higher than other states. Generally, this happens when the amplitude of the oscillation is very low.

Algorithm A.4 Add key states to the mini learning batch.

require LearningBatch,Controlled Reponses,Uncontrolled Responses
ensure LearningBatchKeyStatesAdded
function AddKeyStates (LearningBatch,CR,UCR)
SortedBatch= SORT(LearningBatch, Rewards)
TargetSt1 = RANDOMSELECT (SortedBatch, 10, 100)
Performance = UCR - CR
CurrentLearningBatch = ADD(LearningBatch, Performance)
SortedBatch= SORT(LearningBatch, Performance)
TargetSt2 = RANDOMSELECT (SortedBatch, 10, 100)$
BatchAddedKeyStates = ADD(LearningBatch, TargetSt1)
BatchAddedKeyStates = ADD(LearningBatch, TargetSt2)
return BatchAddedKeyStates
end function