PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Ge Ou

Entitled
Robust Hybrid Simultion with Improved Fidelity: Theory, Methdology and Implementation

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Shirley J. Dyke
Chair

Patricia Davies

Arun Prakash

Julio A. Ramirez

Billie F. Spencer

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): Shirley J. Dyke

Approved by: Dulcy M. Abraham 7/23/2016

Head of the Departmental Graduate Program Date

ROBUST HYBRID SIMULATION WITH IMPROVED FIDELITY: THEORY, METHODOLOGY, AND IMPLEMENTATION

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Ge Ou

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2016

Purdue University

West Lafayette, Indiana

To my grandfather Hongchun Liu.

ACKNOWLEDGMENTS

The financial support of the National Science Foundation, partially under grants CMMI 1011534, CMMI 0927178, CNS 1136075, and Purdue University Bilsland Fellowship, are greatly appreciated. Support for the development of the experimental facility (novel CIRST instrument) was made possible through the NSF MRI Program under Grant CNS 1028668 and the School of Mechanical Engineering at Purdue. Also, I am grateful of having the privilege working in the Network of Earthquake Engineering Simulation (NEES) program, this experience has further facilitated my passion in academic career.

I would like to express my sincere thanks to my research advisor, Professor Shirley J. Dyke, for her extensive guidance and support to initiate and improve this work. I remember she used to revise all my writings over and over again with great patience even on her tight schedule. Her immense knowledge, commitment, modesty, and continuous encouragement has inspired me beyond academic work.

I sincerely appreciate the efforts of my committee members, Professor Patricia Davies, Professor Arun Prakash, Professor Julio Ramirez, and Professor B. F. Spencer, Jr., for their suggestions and constructive questions which enriched my research from various perspectives, and also for their help and support for my academic career.

Some research in this dissertation is supported under NEES US-China collaboration project. Through this project I have had the opportunity to work with Professor Bin Wu and his group in the Harbin Institute of Technology, China. I would like to thank him for his support on conducting experiments there. Also, some of the ideas cannot be implemented without the help of Ge Yang, for modifying the HyTest software.

The help from my fellow graduate students in the Intelligent Infrastructural System Laboratory (IISL) are dearly appreciated. Special acknowledgement to Dr. Xiuyu Gao, who has led me to the area of control and real time hybrid simulation; Professor Wei Song, who is continuously providing me suggestions on my research problems and tips on preparing future success; and Dr. Zhuoxiong Sun, not only he has helped me with various experimental setups and testings, but our fruitful and intensive discussion, and the lifelong friendship are my very best memories during this journey.

Most importantly, heartfelt thanks to my parents Jinping Ou and Liwei Cheng, from whom I have learned passion and conscientiousness, which has eventually led to this work. Special gratitude is owed to my fiance Bing Han, for his undoubted faith in me and unconditional supports for all my dreams.

This dissertation is dedicated to my grandfather Hongchun Liu.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

ABSTRACT

Ou, Ge PhD, Purdue University, August 2016. Robust Hybrid Simulation with Improved Fidelity: Theory, Methodology, and Implementation. Major Professor: Shirley J. Dyke.

Civil engineers of today have been charged with the task of providing resilient and sustainable infrastructure designs, and to use those for establishing resilient communities. To achieve this mission, improved designs, new materials, and efficient retrofit strategies are being introduced around the world. Before many of these techniques are used in the real world, efficient methods are needed to evaluate the performance of those innovations through high fidelity experimentation.

Hybrid simulation is an integrated, numerical-experimental method that combines the benefits of simulation with component-level experiments. In hybrid simulation, the structural components which are difficult to model are constructed physically (named the experimental substructure) while the rest of the structure is computationally modeled in a simulation (named the numerical substructure). During hybrid simulation, the boundary condition information between the numerical substructure and experimental substructure is exchanged at each numerical substructure integration step.

The objective of this dissertation is to advance the state of the art in hybrid simulation. First, a robust platform for hybrid simulation running in real time is developed that considers the complex interactions between various components of the physical-computational system. Next, to improve the fidelity of hybrid simulations that contain numerical elements that are similar to the physical specimen, online system identification is integrated into hybrid simulation. The improvement of fidelity through hybrid simulation with model updating is illustrated through the model updating performance as well as a global assessment by comparing to the shake table test results.

1. INTRODUCTION

Extreme natural hazards, such as earthquakes, tsunamis, tornadoes, and hurricanes are great threats to civilian safety and security. The total casualties from the 2005 Katrina Hurricane, 2008 Wenchuan Earthquake (Magnitude 8.0), 2010 Haiti Earthquake (Magnitude 7.0), 2011 Japan Earthquake (Magnitude 9.0) and Tsunami, and 2105 Nepal Earthquake (Magnitude 8.0) have reached over 400,000, and the economic losses are well over $500 billion. One critical mission for todays civil engineers is to provide resilient and sustainable design approaches for mitigating these hazards, with the goal of eventually forming resilient communities.

To achieve such a mission, novel structural systems such as rocking walls, self centering frames, and different composite shear wall systems are being introduced; new materials including FRPs, high strength steel and high strength concrete are being manufactured; and new retrofit concepts are being designed to prepare our aged infrastructure to meet engineering needs according to design guidelines [1]- [8]. The behavior of these new structural designs must also be investigated and assessed before construction through either experimental or analytical means. Because structural responses under extreme events are often hard to predict with complete accuracy using numerical models, experimental methods such as quasi-static testing and shake table testing have been developed and used extensively.

Quasi-static testing is used to understand the nonlinear behavior of an element under a cyclic loading protocol, which often provides the basis for codes and guidelines on element designs. However, global (structural) level response cannot be obtained simultaneously during quasi static testing and thus requires further numerical analysis. Shake table testing provides a direct assessment of structural designs while considering realistic seismic loading. In a shake table testing, the entire structure is mounted on a shake table while the table reproduces a ground motion either according to a design standard or a previously earthquake record. It is considered to be the most realistic testing method for evaluating the global performance of a structure. However, the application of shake table testing is limited by the cost and the available facility for large scale structures. There are a limited number of shake tables worldwide able to test full scale structures. Most notably, these include the E-Defense (1200 ton vertical payload) in Miki City, Japan and the LHPOST (2000 ton vertical payload) in San Diego, CA, United States, as well as some new laboratories coming online in China in the near future. Testing using other shake table facilities will require compensation for similitude and scaling effects. Furthermore, some structures are simply too large to test on even full scale shake tables.

As an effective alternative to investigate performance of an infrastructure with increased fidelity, hybrid simulation was proposed in 1969 by Hakuno et al. [9] which is also known as online testing, pseudo-dynamic testing, and substructure testing. In a hybrid simulation, a structure is decomposed into a numerical substructure and an experimental substructure, some terminologies are later elaborated in chapter 2. The interface between the two substructures can be any physical loading system, normally a hydraulic actuator, can also be a shake table, an electric motor, etc. Boundary conditions between the substructures are realized through such loading systems. For the common form of hybrid simulation with hydraulic actuators, the hardware setup is similar as a quasi-static testing. Instead of using a predetermined loading path as in quasi-static testing, in a hybrid simulation, the physical specimen interacts with the numerical model which loading path is the element response calculated under realistic excitation in the specific structure. Therefore, hybrid simulation can provide observations on both local behavior (element level) for component capacity analysis and global behavior (structural level) for structural performance assessment [10].

The first hybrid simulation test of a single degree freedom system was conducted using an analog computer to solve the equations of motion and an electromagnetic actuator to impose the boundary condition. In the mid-1970s, the present form of hybrid simulation was established utilizing digital computers for discrete system

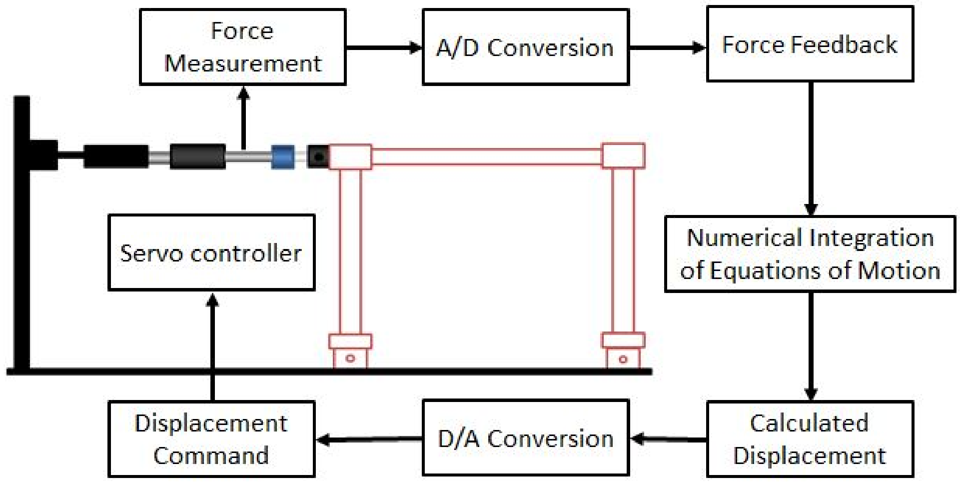


Figure 1.1.: Basic Procedure for Implementing Hybrid Simulation

analysis and hydraulic actuators for boundary condition compatibility execution [11]. A schematic implementation of hybrid simulation is illustrated in Fig. 1.1. From the 1970s to the 1990s, efforts were made on developing precise displacement control algorithms for interface devices to minimize error propagation due to systematic undershoot and overshoot in loading steps [12], [13], [14].

Most of the research efforts toward establishing hybrid simulation methods were dedicated to the simulation of the numerical substructure, and integration stability and accuracy were examined. Different integration algorithms including explicit, implicit, or the mixture of the two, have been applied for hybrid simulation. Explicit methods are conditionally stable, which means the affordable time interval can be very small according to the highest natural mode in a structure [15]. This requirement is always a concern for implementing hybrid simulation. Alternatively, unconditionally stable implicit algorithms require substep iterations. These substep iterations, if implemented physically on the experimental substructure, can cause overshoots which do not exist in the true structural response and purely induced by the computational iterations [16]. These overshoots in the physical loading path can affect the hysteretic behavior of the experimental substructure. Advances in the integration algorithms are further made to improve computational accuracy with minimum or no iterations, such as α operator splitting method, predictor-corrector method, Rosenbrock-based method, etc. [17] - [22].

1.1 Challenges in Real Time Hybrid Simulation

In early applications of hybrid simulation, studies focused on experimental substructures consisting of steel bracings, joints, and frames, concrete and masonry buildings, bridge piers, etc. [23]- [28] which do not exhibit significant rate dependent effects. Therefore, all rate dependent effects (inertia or viscous) were considered to be represented in the numerical substructure and the loading of the actuators needed to realize the boundary conditions could be performed slowly. With a variety of promising energy dissipation devices (magnetorheological and viscous dampers) being developed for structural dynamic response mitigation, these devices were considered to be very good examples of experimental substructures in hybrid simulation due to their distinguishing role in an infrastructure. However, in order to accurately investigate the behavior of these rate dependent energy dissipation devices, hybrid simulation needed to be implemented at a real time scale. Based on these requirements, a new branch of hybrid simulation was established, known as real time hybrid simulation (RTHS). RTHS has introduced several new challenges due to the need for real time execution and implementation, in addition to all of the existing challenges associated with hybrid simulation.

One important challenge of RTHS is the implementation of boundary conditions by the interface devices. In RTHS, the loading in the experimental substructure is no longer executed at an extended time scale, and the sampling rate needed for each loading step can be as fast as several kHz. Therefore, possible time lags due to actuator dynamics or due to delays resulting from hardware communication cannot be ignored. Time delays and lags are equivalent to negative damping in RTHS and can further cause instability in the test [29]. Several compensation algorithms and actuator control algorithms have been developed for RTHS, including the polynomial extrapolation [30], [31], various time delay and adaptive time delay compensation methods [32], [33], [34], [35], [36] and upper bound delay compensation algorithm [37]. Additionally, modern control algorithms have been introduced into interface device control in RTHS. First a model-based control approach was proposed by Carrion and Spencer [38], [39], and its later modification included optimized Linear Quadratic Regulator (LQR) in the feedback for uncertainties is presented [40], [41]. Robust control algorithms such as H_∞ control algorithm were considered to achieve a trade-off between the control accuracy and the systematic and random errors, including modeling errors, measurement noises and disturbances [42], [43], [44].

As in hybrid simulation, the stability of the integration scheme used for the numerical substructure represents another challenge. With the small time intervals (often 1msec or less) available in RTHS, explicit integration methods are normally adopted. Some explicit algorithms have been proved to be unconditionally stable and unaffected by the highest natural frequency of the structure [21], [45], [46], [47]. Predictor-corrector based numerical integration algorithms that help reducing delays in RTHS due to the transfer system dynamics [48]. Some implicit integration algorithms have also been investigated by researchers including an equivalent force control for solving nonlinear equations of motion [49]. HHT-α spiting method with fixed number of substep iteration [50] provides stable experimental results for RTHS. However, the aforementioned HHT-α method requires numerical-experimental information exchange at each substep, where the substep displacement commands is calculated based on the measured restoring force from the previous substep [51]. This substep numerical-experimental information exchange normally is not supported by a real time computing platform.

The interactions present in the closed loop arrangement in RTHS introduces additional challenges as well. Due to the nature of the hybrid simulation concept, the numerical and experimental substructures are linked in sequential order and the response of the experimental substructure cannot be obtained instantaneously. In

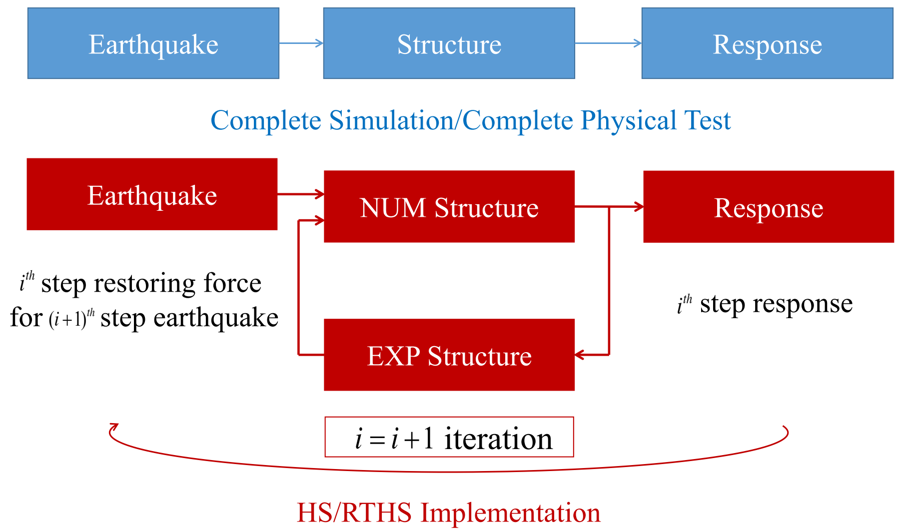


Figure 1.2.: Schematic Drawing for Traditional Simulation/Shake Table Testing and HS/RTHS

hybrid simulation, several small sub-steps displacement incremental are implemented to achieve the $i+1^{th}$ step displacement, therefore the steady state force response of the $i+1^{th}$ step is available. However, the implementation of RTHS on an real-time operating system (RTOS) platform is normally executed at one single fixed rate, and no sub-step data acquisition is allowed. In RTHS, the restoring force is measured at the beginning of the RTHS $i+1$ step which uses the response of the i^{th} step displacement. For example, to compute response at the first time step, the restoring force is set to be zero because no input has yet been sent to the experimental substructure. However, the true restoring force should be $R(x_1, \dot{x}_1)$. Consequently, a unit delay, as shown in Fig. 1.2, exists in the experimental force measurement. This delay is normally considered as computational delay or communication delay [52]. Once the transfer system lag is compensated for using a controller to accurately enforce the boundary conditions with almost zero step time delay, such computational delay can affect the stability of an integration algorithm, further leading to constraints on the integration step size. This effect is most pronounced for stiff, lightly-damped structures that have relatively high natural frequencies associated with the first few dominant modes. Due to the growing interest in understanding and improving the fidelity of RTHS results, there is a need to use larger, more sophisticated numerical models within RTHS [53]. Such high-fidelity models often take more time to run than the conventional RTHS execution time interval of 1 msec, which creates a need for exploring RTHS at lower execution rates and consequently higher computational delays.

1.2 Hybrid Simulation with Model Updating

In early hybrid simulation applications, it was only necessary to include one or two nonlinear components in the experimental substructure [54]. Due to the advances later in the new system and element designs such as post-tensioned rocking connections and shear walls, etc. [55] , those new elements may be spatially distributed within a single structural system. Lab capacity (number of actuators, lab space) and budget may limit the number of physical components in one hybrid simulation, leaving a larger portion of their counterparts in the numerical substructure. In these applications, hybrid simulation fidelity is more dominated by modeling accuracy of those counterparts rather than the responses of the physical components.

Recent research has expanded the hybrid simulation concept by incorporating model updating methods, with the goal of improving the accuracy of the numerical model. In hybrid simulation with model updating, the model parameters of the tested specimen are identified using the measured response of the physical component. Further, those parameters are updated to its counterparts in the numerical substructure. The identified parameters can be of a phenomenological (macro) model which may not have physical meaning, which requires the identical geometry and constitution between numerical and physical counterparts, or a constitutive model in which the parameters describe the material property.

Phenomenological models in the Bouc-Wen family have been adopted widely among recent hybrid simulation with model updating applications [56], [57], [58], [59], [60]. Compared to phenomenological model, a constitutive model provides a deeper understanding on structural component behaviors such as component level of damage, serviceability, and prediction of failure. Furthermore, the tested components and its counterpart in numerical simulation do not need to be exactly identical. Hazem el al. [62] first proposed a hybrid simulation framework where the finite element software ZeusNL is combined with model updating algorithms such as genetic algorithm or neural network to identify the parameter of constitutive bilinear steel model and nonlinear concrete model. This framework is validated through numerical examples and offline experimental data [63].

1.3 Motivation of the Dissertation

Hybrid simulation, especially real time hybrid simulation is an integrated feedback system with complex components, including physical specimen, numerical integration algorithm, interface device, the control algorithms, and D/A and A/D conversion. Generally, in the previous development of RTHS techniques, each component is investigated in an isolated form. The interactions between noise and other uncertainties in control, interface, and specimen, are not generally discussed. A robust hybrid simulation platform considering not only the performance of each component but the performance of the entire closed loop system is desired.

To improve test fidelity, hybrid simulation with model updating is introduced when components similar to the physical specimen, and thus, unknown, exist in the numerical substructure [56]. However, the investigation of the limitations of such concept has not been performed. Online model updating algorithms require knowledge of the excitation to the physical substructure as well as its response to identify the model parameters. This excitation normally takes the form of a structural response which is already filtered by the structure itself and likely contains limited frequency information, especially on examining the dynamic system where specimen response is rate dependent. In hybrid simulation, the identification information is more related to amplitude where the loading does not contain frequency content with low speed execution. In RTHS, the information maybe related to both amplitude and frequency. Other possible limitations relate to the varying level of complexity of the nonlinear models to be identified. Clearly, the performance of the chosen model updating algorithm with respect to such challenges should be carefully examined prior to implementing the test.

Even though hybrid simulation with model updating concept has been successfully applied and validated through some past examples, those verification studies mainly focused on local (model level) estimation accuracy [58], [59], [60]. The evidence supporting hybrid simulation fidelity was related to parameters convergence during the testing. However, global response comparison has not thus far been convincingly demonstrated. In the past literature, those global responses are compared with a baseline numerical simulation, which may not be accurate. Also, the performance of model updating on both phenomenological and constitutive models of the same specimens have not been investigated.

1.4 Organization of the Dissertation

The remainder of this dissertation is as follows: Chapter 2 introduces the terminology and implementation procedure for hybrid simulation, real time hybrid simulation, and hybrid simulation with model updating.

Chapter 3-7 are dedicated to the theory and methodology to establish robust real time hybrid simulation technique. Chapter 3 discusses the model of interface dynamics and the procedure of their identification. The identified model is further validated with experimental data. Chapter 4 describes several current interface control algorithms. The accuracy and robustness of those algorithms considering the effect of noise, disturbance, and modeling errors is systematically investigated. Continuing from the conclusions in chapter 4, chapter 5 formulates a robust integrated actuator control algorithm which aims at improving robustness of the control algorithm without sacrificing accuracy when high noise to signal ratio exists. This algorithm is implemented on two different physical setups with different loading capacities. Chapter 6 reviews the available numerical integration algorithms in real time hybrid simulation. An associated stability analysis is conducted considering a unit time delay due to the sequential loading feedback loop. Based on the observations, a modified Runge-Kutta integration algorithm is proposed in Chapter 7. This integration algorithm explicitly incorporates the existence of the unit delay. Both the theoretical derivation and experimental validation are presented.

Chapter 8-11 discuss the methodology, implementation and performance analysis for hybrid simulation with model updating. Chapter 8 provides a summary of several available phenomenological and constitutive steel models. In Chapter 9, the formulation of hybrid simulation with model updating is presented. The model updating accuracy analysis is carried out to examine the feasibility of using such model updating algorithm under hybrid simulation, demonstrated through a simulation example. Finally in chapter 10, two hybrid simulation with model updating platforms are established, where both phenomenological and constitutive models are updated. The implementation procedure and model updating performance for both cases are discussed in Chapter 10. To further analyze the improvement in hybrid simulation fidelity with model updating algorithm incorporated, responses from hybrid simulation experiments are compared to shake table test results in Chapter 11.

Finally, in Chapter 12 a summary of this dissertation is presented along with some potential future research suggestions.

2. HYBRID SIMULATION FORMULATION

In hybrid simulation, one structure is decomposed into two substructures as in Fig. 2.1: the experimental substructure and the numerical substructure. In the experimental substructure, one or several structural components which are hard to model can be physically investigated. The rest of the structure, which is well understood, is modeled in the numerical substructure. The loading path of the physical specimen is the element response calculated under realistic excitation in the specific structure. Therefore, hybrid simulation can provide observations on both local behavior (element level) for component capacity anlaysis and global behavior (structural level) for structural performance assessment [10].

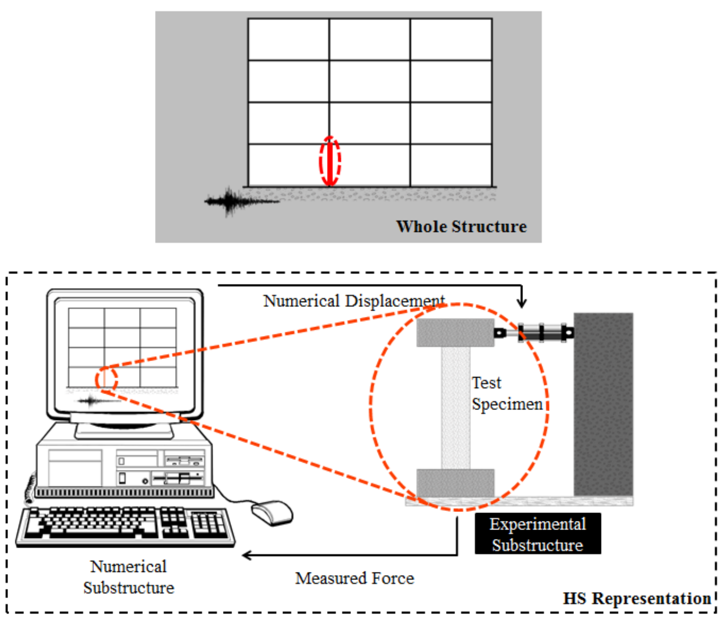


Figure 2.1.: Whole Structure Model Representation in Hybrid Simulation [64]

2.1 Hybrid Simulation

The implementation of hybrid simulation can take different forms depends on the available hardware and software in a specific laboratory. In the basic concept of hybrid simulation, at least one computational component and one physical component are needed [64]. Compared with the entire structure being modeled through solving the equation of motion, in hybrid simulation, part of the numerical model is substituted with the components to be tested in the laboratory. The numerical boundary condition obtained at each integration step is physically implemented and measured using interface devices and data acquisition systems. In this chapter, to simplify the explanation of hybrid simulation concept, it is assumed there is only one numerical component and one physical component.

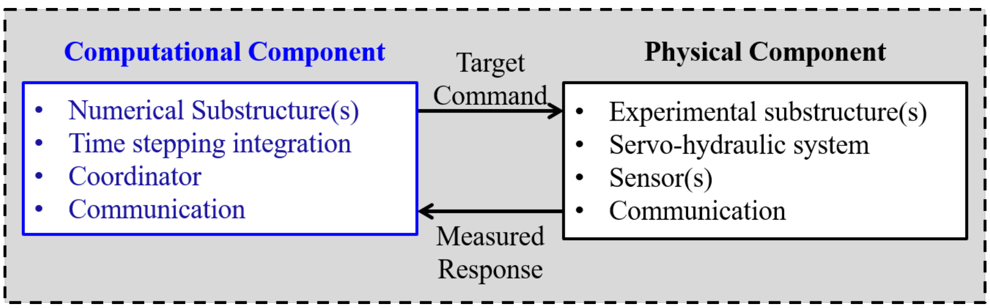


Figure 2.2.: Basic Components in Hybrid Simulation [64]

Fig. 2.2 shows a configuration of the basic components in a hybrid simulation. In a hybrid simulation, the computational component sends a target command to the physical component and the physical component returns the measured response. The target command can be boundary condition displacement, force, or the combination of the two, and the measured response can be either specimen restoring force or displacement [65]. Based on the different target command and the interface device control goal, there are displacement controlled hybrid simulation, force controlled hybrid simulation, and mixed-controlled hybrid simulation. In this dissertation, only displacement controlled hybrid simulation is discussed where the target command is also known as desired displacement, the measured response is typically the restoring force of the specimen. The interface device is represented by a hydraulic actuator, which could also be a shake table or electric motor as well.

2.1.1 Fundamental Theory and Procedure

The problem statement posed in hybrid simulation is to determine the seismic response of a structural model composed of experimental and numerical substructures. To this end, the complete structural model is idealized as a discrete parameter system with a finite number of degrees of freedom. The time-discretized equation of motion of the master model (combined experimental and numerical substructures) is expressed as:

$$M\ddot{x}_i + C\dot{x}_i + R(x_i, \dot{x}_i) = F_i \tag{2.1}$$

in which, M and C, are mass and damping matrices of the master model in hybrid simulation, respectively; F_i is the external force vector; $\ddot{x}_i$ and $\dot{x}_i$ are velocity and acceleration vectors, respectively; $R(x_i, \dot{x}_i)$ is the restoring force vector that consists of the measured force in the experimental substructures R^E and the simulated force in the numerical substructure R^N; and the subscript i denotes responses at the i^{th} integration step. If the numerical substructure is linear elastic, R can be expressed as Kx_i where K is the stiffness matrix of the numerical substructure and x_i is the displacement vector. Note that the experimental restoring force vector may include strain rate-dependent, damping, or inertial forces, depending on the characteristics of the experimental substructure.

Similar to any numerical simulation, time-stepping integration procedures are applied to solve the equations of motion in Eq. 2.1. A variety of integration methods have been proposed for hybrid simulation, with explicit methods being the simplest to implement. Implicit methods, though often preferred for numerical simulation of structural responses, are challenging to implement for hybrid simulation due to the required iterations.

In a typical hybrid simulation, typically follows the steps below:

- For time step $i+1$, calculate the response based integration algorithm, if explicit algorithm is used, then:

$$x_{i+1} = x_i + \Delta \dot{x}_i + \frac{1}{2}\Delta t^2 \ddot{x}_i \tag{2.2}$$

- Impose displacement x_{i+1} to test specimen and measure experimental measured force $R^E(x_{i+1}, \dot{x}_{i+1})$.
- Calculate numerical response $\dot{x}_i, \ddot{x}_i$ use integration scheme given f_{i+} and R^E_{i+1}.
- Go to time step $i+2$.

Fig. 2.3 shows a diagram of this procedure and the flow of a hybrid simulation framework. The processes at each step includes solution of the equations of motion; execution of the target displacement; measurement of the restoring force; and communication. It should be noted that while these processes are essential in most hybrid simulations, there are many different ways to configure these tasks depending on hybrid simulation software, laboratory equipment, and numerical models, etc. In particular, configurations for conventional hybrid simulation and real-time hybrid simulation can be slightly different due to constraints and requirements.

2.1.2 Numerical Components

The numerical components in a hybrid simulation are similar as in a numerical analysis of a dynamic system. A structural model (here named the numerical substructure) and a time stepping integration algorithm are needed. The difference is the numerical model does not only implement the time stepping integration with numerical data but is also sending and receiving data from the physical testing. This normally requires a coordinator to associate with different communication protocol. Some standard packages (UI-SIMCOR [66], OpenFresco [67], and HyTest [69]) have

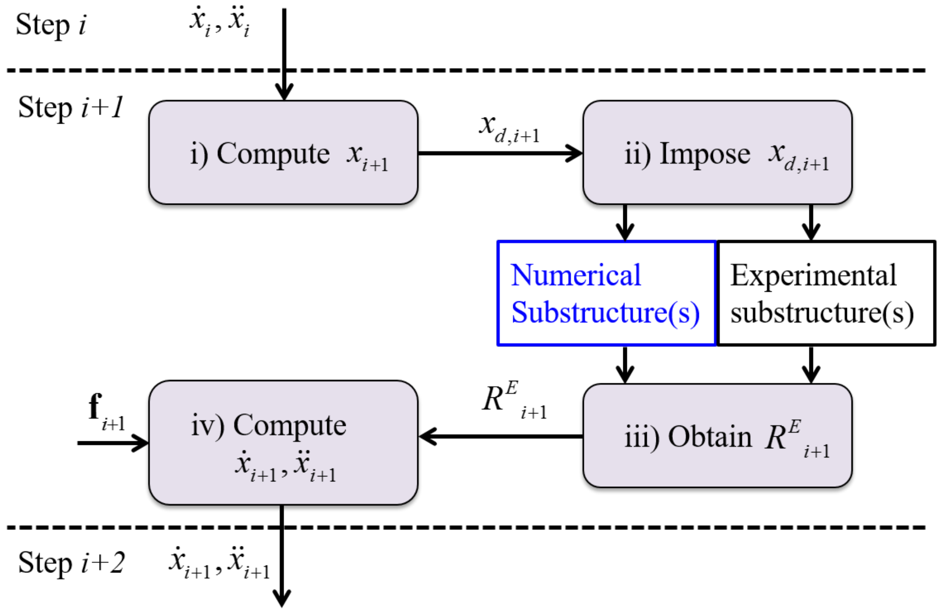


Figure 2.3.: Time Stepping Procedure in a Typical Hybrid Simulation

been developed to accomplish such tasks. Fig. 2.4 shows two typical configurations of the computational components in hybrid simulation.

Configuration I has a designated coordinator to define a hybrid simulation. In this configuration, the time-stepping integration algorithm and communications are handled by the coordinator. The numerical and experimental substructures are outsourced to finite element program and a physical laboratory, respectively. Advantages of this configuration are: i) all of the substructures are equally handled by the coordinator (no differences between the numerical and experimental substructures at the coordinator) and; ii) it is relatively easy to define multiple/different components to represent substructures in hybrid simulation. However, a disadvantage of this configuration is that all of the target displacements and restoring forces have to be transferred via network at each time step. Thus, if the sizes of the models become large, communication and duration of the simulation can become problematic.

In Configuration II, there is no designated coordinator, but finite element software serves as the master simulation that handles communication with the physical laboratory. In this configuration, the time-stepping integration algorithm and the numerical substructures are processed in the master simulation, and the experimental substructures are treated as outsourced elements. The advantage of this configuration is that it does not require network communication of data in the numerical substructures. Thus, it is suitable for a hybrid simulation with a large number of degrees-of-freedom. However, the disadvantage is that capabilities and functionalities are limited by the finite element software package adopted as the master simulation. OpenSees uses this configuration to conduct hybrid simulation [68].

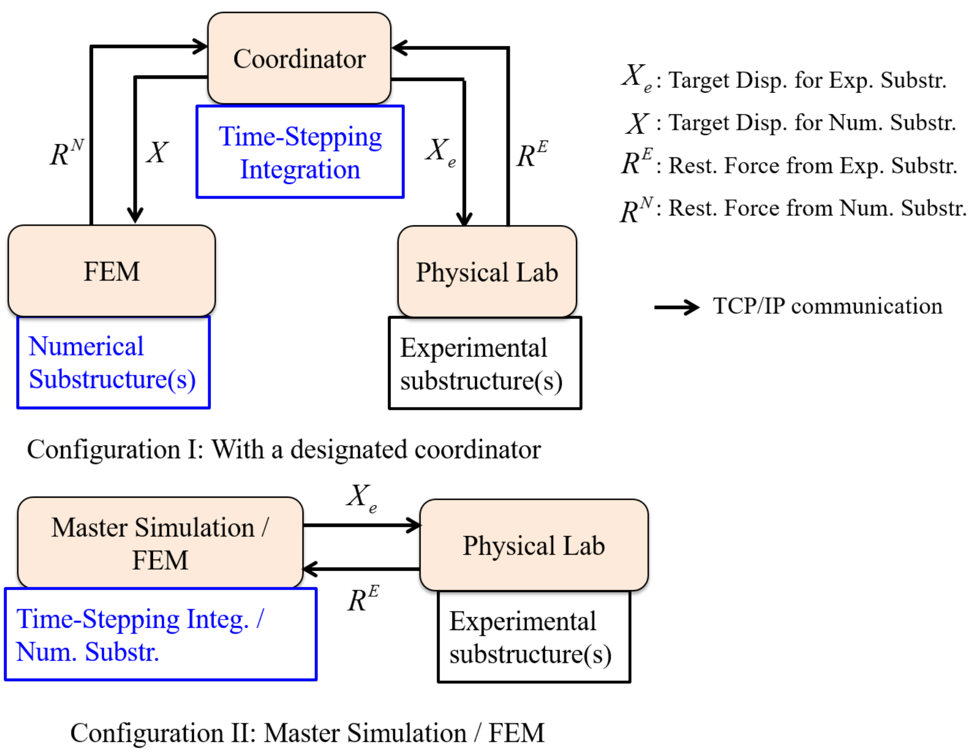


Figure 2.4.: Example Configurations of the Computational Components in Hybrid Simulation

2.1.3 Physical Components

The physical components in a hybrid simulation are similar to those used in a quasi static (cyclic loading) testing. Fig. 2.5 shows a schematic of the essential physical components in hybrid simulation. A hydraulic actuator is needed for implementing the target displacement. The physical specimen to be examined is attached to the hydraulic actuator. Sensors such as linear variable differential transducers (LVDTs) and load cells are needed for measuring the executed displacement (measured displacement) and restoring force.

Difference in the test configuration may be present due to the choices made in regard to the hydraulic actuator controller. The enclosed section in the dashed line represents the servo-hydraulic feedback control system that consists of the hydraulic actuator, hydraulic controller and sensors. Actuators are used to impose the target displacements to the physical specimen and sensors are needed for measurements. In most laboratories, such hydraulic actuator controller is embedded in commercial software provided by the hydraulic actuator manufacturers, including a hydraulic controller, hydraulic power supply, service manifold, etc. Because of the requirement for accurate boundary condition compatibility in hybrid simulation, some specific hydraulic control algorithms are developed. These controller can be implemented as a outer loop in addition to the existing controller. More details of the inner loop and outer loop controller are discussed in chapter 3, 4, 5 of this dissertation.

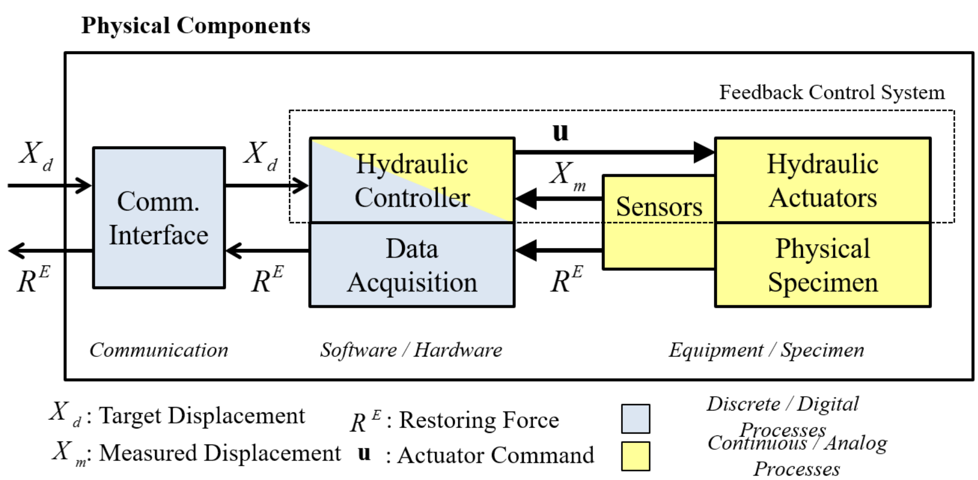


Figure 2.5.: Configuration of the Physical Components in Hybrid Simulation

2.2 Real Time Hybrid Simulation (RTHS)

With the increased applications involving dissipation devices in structural dynamic response mitigation, there has been a growing interest in real time hybrid simulation (RTHS) where the hybrid simulation is implemented at a real time scale. The benefits of RTHS are similar as in hybrid simulation.

However, the main difference between hybrid simulation and RTHS is that in RTHS the test is intended to be executed in real-time. This means that if the integration time step (i.e. sampling of the simulation, time discretization) is 1 ms, all of the processes in each step in Fig. 2.3) would be completed within 1 ms. To meet such a demanding constraint, there are certain system requirements in RTHS. Furthermore, computational and physical components have to be configured together, accounting for specifications of each other; unlike the conventional hybrid simulation, computational and physical components cannot be configured independently [64].

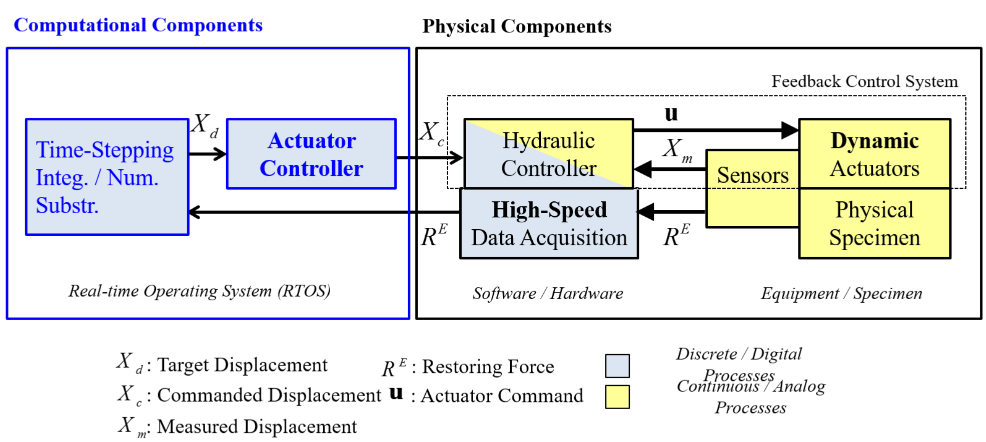


Figure 2.6.: an RTHS Configuration Example

Fig. 2.6 shows an example configuration of RTHS. In this configuration, the time-stepping integration algorithm and the numerical substructure are combined together as a computational component. Note that the layout of the components in RTHS can be slightly different (one may say data acquisition should be in the computational components).

On the physical side, in order to achieve the target displacements in real-time, RTHS requires dynamic actuators that are capable of producing required velocity. This requirement may require additional investigation of the hydraulic system such as flow rate of hydraulic power supply, capacities of accumulator, and rated-flow of the servo valve, etc. The other requirement in the physical components in RTHS is high-speed data acquisition system that allows sampling of sensor data faster than the sampling of the simulation .

One of the critical components in RTHS is the actuator controller. In RTHS, the loading in the experimental substructure is no longer executed at an extended time scale, and sampling rate for each loading step can be as fast as several kHzs. Therefore, the possible time lag and delay due to the inherent actuator dynamics and hardware communication cannot be ignored. To successfully perform RTHS, the actuator controller is often essential to compensate for adverse effects of actuator dynamics and delays. The actuator controller is usually designed to generate the command displacements from the target displacements such that the measured displacements are close to the target displacements. On the computational side, the time-stepping integration, analysis of the numerical substructure, and input /output (I/O) between the physical components (I/O) have to be realized in real-time. In most RTHS, the time-stepping integration and the numerical substructures are integrated into a single program to increase computational efficiency and eliminate unnecessary communication time, as in Fig. 2.7. In hybrid simulation, the execution of the $i+1^{th}$ step displacement is implemented through several small substep increments, the steady state response of the specimen restoring force at the $i+1^{th}$ step is available at the substeps. However, the implementation of RTHS is on an RTOS platform which is normally executed at one single fixed rate, no sub-step data acquisition is allowed. In RTHS, the restoring force is measured at the beginning of the RTHS $i+1$ step which is the response of the i^{th} step displacement. This unit delay, or also known as the computational and communication delay, can affect the accuracy and even the stability of RTHS. This effect is elaborated in chapter 6.

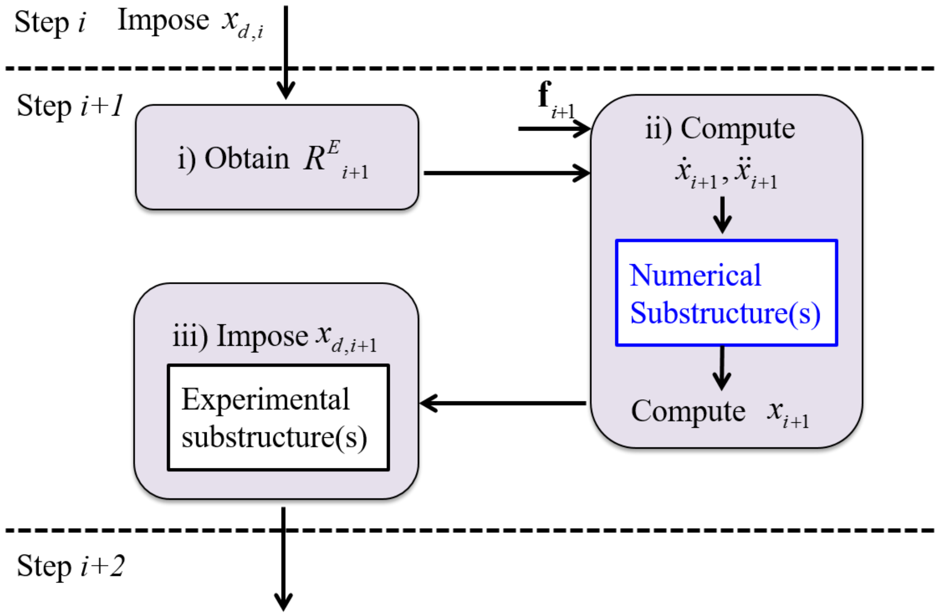


Figure 2.7.: Time Stepping Procedure in an RTHS

2.3 Hybrid Simulation with Model Updating (HSMU)

Hybrid simulation with model updating (HSMU) is an extension to hybrid simulation. In the early hybrid simulation applications, it was only possible to include one or two nonlinear components in the experimental substructure. Later, with advances such as new structural systems and component design RTHS enables their efficient investigation. Those target components may be spatially distributed in an infrastructure. The laboratory capacity (number of actuators, lab space) and budget may limit the number of physical components in one hybrid simulation, leaving a larger portion of their counterparts in the numerical substructure. In these applications, hybrid simulation fidelity is often dominated by the modeling accuracy of those counterparts rather than the responses of the physical components. Model updating methods have been integrated into hybrid simulation, with the goal of improving the accuracy of the numerical model. In HSMU, the model parameters of the tested specimen are identified using the measured response of the physical component.

The equation of motion of a dynamic system in Eq. 2.1 can be written into hybrid simulation form as:

$$M^N\ddot{x}^N + C^N\dot{x}^N + F^E(x^E, \dot{x}^E) + R^N(x^N, \dot{x}^N, \theta_R) = F \tag{2.3}$$

$$M^E\ddot{x}^E + C^E\dot{x}^E + R^E(x^E, \dot{x}^E) = F^E(x^E, \dot{x}^E) \tag{2.4}$$

where the superscript $(\ )^N$ and $(\ )^E$ denote the portions of the reference structure included in the numerical and experimental substructures respectively, $M = M^E + M^N$, $C = C^E + C^N$. F^E denotes the measured force in the experimental substructure. $R(x, \dot{x}, \theta_R))$ is the restoring force provided by the nonlinear components. With model updating:

$$\tilde{\theta}_R = \Psi(R^E, x^E, \dot{x}^E, \theta_\Psi) \tag{2.5}$$

where Ψ indicates the model updating is performed in real-time, θ_Ψ is the parameter being updated through the chosen model updating algorithm, The parameter set of the numerical model of the physical specimen $\tilde{\theta}_R$ can be recursively identified through minimizing the cost function associated with the model updating algorithm. For the components whose composition are similar to the physical specimen, their restoring force can be modified into: $R^N = R(x^N, \dot{x}^N, \tilde{\theta}_R)$. Therefore, Eq. 2.3 can be modified into:

$$M^N\ddot{x}^N + C^N\dot{x}^N + F^E(x^E, \dot{x}^E) + R^N(x^N, \dot{x}^N, \tilde{\theta}_R) = F \tag{2.6}$$

The physical components in HSMU are the same as they are in the hybrid simulation. In the computational components, an additional model updating algorithm is needed. In some cases, the model updating algorithm is coded in a different programming environment, thus a coordinator is needed to implement the communication. The expended configuration example of HSMU is shown in Fig. 2.8.

Because of the additional model updating step, the implementation of the time stepping in HSMU is modified, shown in Fig. 2.9. In each step, the nonlinear model

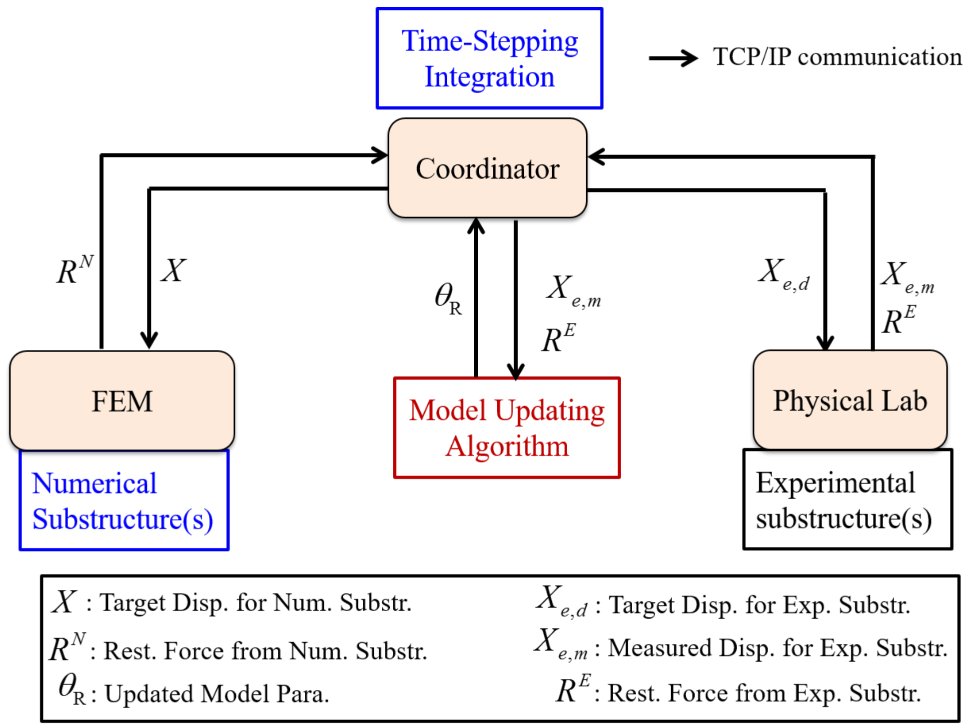


Figure 2.8.: Example Configurations of the Computational Components in HSMU

parameter describes the physical specimen behavior is identified, this parameter set $\theta_{R,i}$ is updated to the numerical substructure. In this case, the quality of identified parameter $\theta_{R,i}$ dominates the fidelity of the hybrid simulation test. There have been arising interests on the discussion of the choices of different nonlinear models, different model updating algorithms, the accuracy and convergence of model updating results, and the evaluation of the fidelity of HSMU. Some of these open tasks are investigated in chapter 8 - 11 of this dissertation.

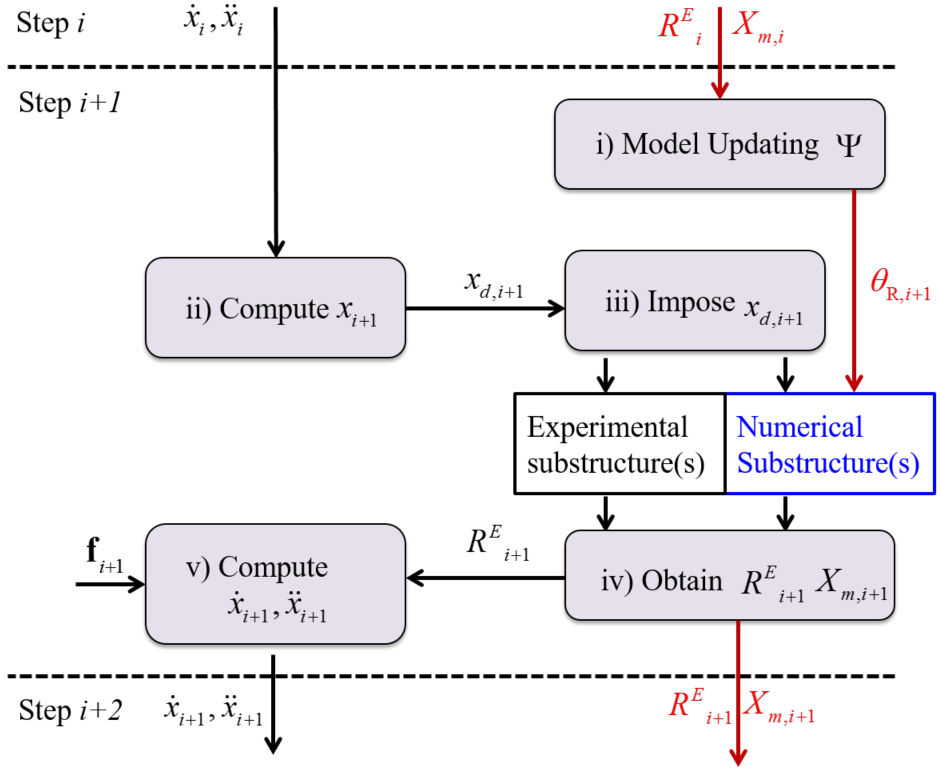


Figure 2.9.: Time Stepping Procedure in an HSMU

2.4 Conclusion

This chapter presents an overview of hybrid simulation formulation, components, and implementation procedure. The basic concept and key benefit are very similar for hybrid simulation and RTHS. However, due to the different implementation constraints in RTHS, the challenge is different. Also the expansion of hybrid simulation incorporating model updating is introduced, some of the open tasks associated with HSMU should be addressed.

3. ACTUATOR IDENTIFICATION

RTHS is an accepted technique used to conduct efficient and high fidelity lab testing. One of the challenges to acquire reliable RTHS results is to achieve synchronization of boundary conditions between the computational and physical substructure [42]. A reliable, practical and easy-to-use system identification method to find a parameterized servo-hydraulic system model would be quite useful for controller design and simulation.

Various servo-hydraulic system models have been considered in the literature. Experiments have shown that a linearized model is valid under low frequency and small amplitude applications [70]. Zhao [71] proposed a servo-hydraulic actuator system identification procedure for effective force testing based on a white-box identification method. Because online measurement of each individual parameter in the hydraulic loop is generally infeasible, they were determined based on manufacturer specifications or educated estimations. Alternatively, a black-box method was introduced by Jelali and Kroll [70]. The system transfer function is established based solely on the measured input and output data sets. Thus, the parameters obtained from a black box identification method do not have physical significance. A grey-box identification method offers a compromise between these two extremes by establishing a parameterized model of the system but leaves specific parameter values as unknowns. This method is widely used in most servo-hydraulic system modeling process in structural testing [72], [73]. The system transfer functions are obtained and an optimization technique is employed to optimize system characteristic parameters for a particular experiment setup. However, in these studies, the parameters were identified for a single test setup. Each time a new setup is established, the identification procedure needs to be repeated. Therefore, a servo-hydraulic actuator model applicable to a wider range of specimens is needed.

Herein, a general method is presented to identify the characteristics of the servo-hydraulic actuator within a certain operation range to assist controller design and RTHS simulation. This model will be applicable for modeling the same actuator with different physical specimens and re-identification is no longer needed. To extract unknown parameters in the servo-hydraulic system, the system identification problem evolves to a global optimization over the continuous space corresponding to specimen variation. An objective function is selected to find the optimal value of each parameter and reflect the physical characteristics of the system. Genetic algorithms (GA) is applied to efficiently obtain the optimal value. The proposed method is demonstrated to be effective for the purpose of RTHS controller design and simulation and might be applied to other scenarios where the linearized system model is valid.

3.1 Linear Actuator Model

Servo-hydraulic system is an arrangement of individual components connected to achieve hydraulic power transfer. The basic structure of a servo-hydraulic system consists of a hydraulic power supply, control elements (valves, sensors, etc.) and actuating elements (cylinders, etc.) [70]. Since the force/pressure output demand of a servo-hydraulic actuator is negligible when compared with the power supply in the lab, only the dynamic responses of control and actuating elements is considered here. The basic structure of the system is shown below in Fig. 3.1. Commonly, an internal proportional feedback control scheme is applied to track the actuator performance by controlling the servo-valve to regulate hydraulic flow. The hydraulic flow enables the actuator to move the target structure as well as sending out real-time displacement measurement from an LVDT, completing the internal controller feedback loop.

The dashed line in Fig. 3.1 indicates the natural velocity feedback inherent in the system. The coupled dynamics prevent researchers from completely separating the dynamic response of the structure and the actuator to model them as independent components in series [74]. The mathematical model of each component in the servo-

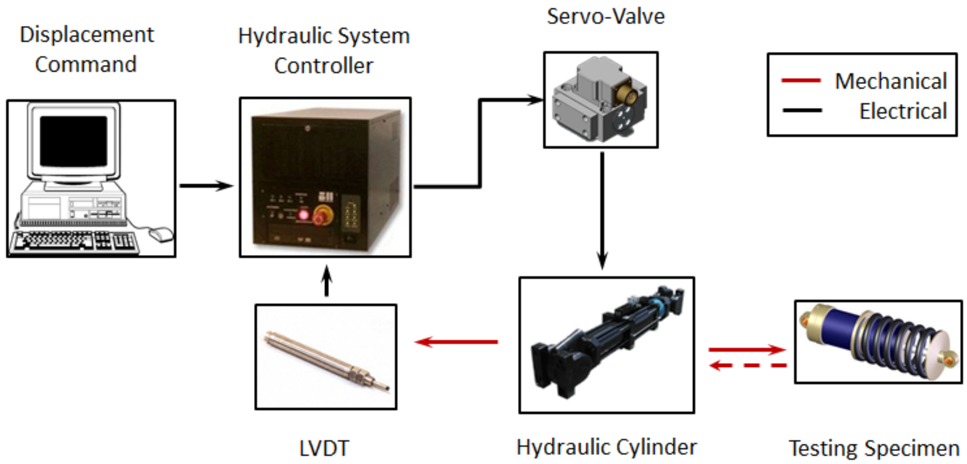


Figure 3.1.: Block Diagram of Servo-hydraulic System

hydraulic system has been well-documented in previous literature [71], [74], [75]. The modeling focuses on four stages of the system: servo-valve and its controller, flow dynamic, actuator, testing structure and its coupling with the servo-hydraulic system.

The governing equations of the servo-hydraulic system are described in following sections [71], [74].

3.1.1 Servo-valve Dynamics

Servo-valve dynamics can be simplified into first order transfer function, assuming inherent time delay τ in the servo-valve. The spool position to valve input is assumed to be linear dependent with ratio K_v as

$$\tau\dot{\tilde{x}}_v + \tilde{x}_v = K_v v_i \tag{3.1}$$

$$G_v(s) = \frac{x_v(s)}{v_i(s)} = \frac{K_v}{\tau s + 1} \tag{3.2}$$

where K_v is the flow gain and τ is the equivalent time constant of the servo-valve. $\tilde{x}_v$ is the normalized spool position $\frac{x_v}{Max(x_v)}$. v_i is the valve control signal transmitted from the hydraulic system internal proportional controller: $v_i = K_p(X_c - X_m)$, X_c is the command signal to actuator and X_m is measured signal from LVDT feedback with proportional gain of K_p.

3.1.2 Hydraulic Cylinder Dynamics

The dynamics in the hydraulic cylinder are linearized based on assuming: a) leakage flow is neglected when valve orifices are closed; b) the square root relationship between orifice flow Q_L and spool position x_v, is put into an approximated linearzed form as load pressure P_L is negligible compared to the pressure supply P_s.

$$Q_L = K_a\dot{P}_L + C_l P_L + A\dot{X}_m \tag{3.3}$$

$$Q_L = K_v x_v \sqrt{1 - \frac{x_v P_L}{|x_v| P_s}} \tag{3.4}$$

where $K_a = \frac{V_t}{4\beta_e}$, V_t is the volume of the actuator cylinder and β_e is the effective bulk modulus of the fluid. C_l is the leakage coefficient of piston from on chamber to another. A is the piston area. X_m is measured displacement from LVDT. The $A\dot{X}_m$ indicated the natural velocity feedback in actuator and is causing control-structure interaction. P_L and Q_L are the load pressure and flow rate respectively. $K_v$is called the flow gain of the servo-valve and P_s is the hydraulic pressure supply. Then the equation can be linearized as $Q_L = K_v x_v$.

3.1.3 Specimen Dynamics

The actuator force generated by $P_L A$ is applied to the test structure, and the response of the structure is dominated by the equation of motion:

$$F = P_L A = m\ddot{x} + c\dot{x} + kx \tag{3.5}$$

$$G_x(s) = \frac{x(s)}{P_L(s)} = \frac{A}{ms^2 + cs + k} \tag{3.6}$$

where the mass m and damping c is the sum of specimen and actuator mass, damping respectively. Since the actuator stroke is considered as rigid, therefore, the stiffness k is assumed to be contributed by the specimen attached to the actuator. The overall system described by the block diagram in Matlab Simulink is shown in Fig. 3.2. Parameters involved in identifying the linear model is listed in Table 3.1.

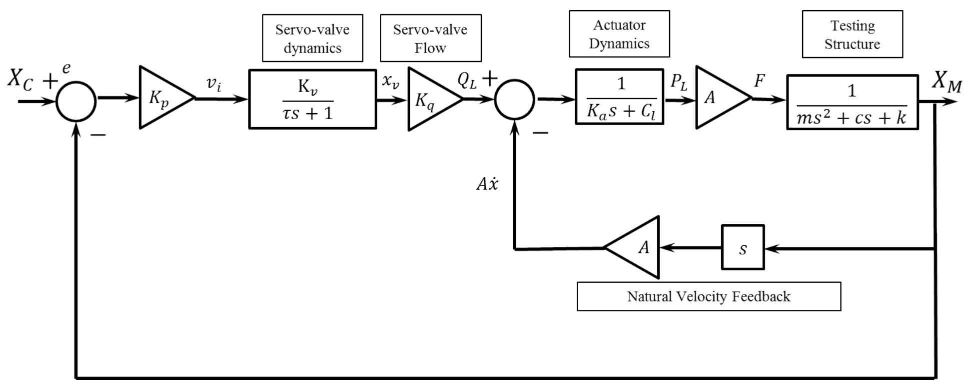


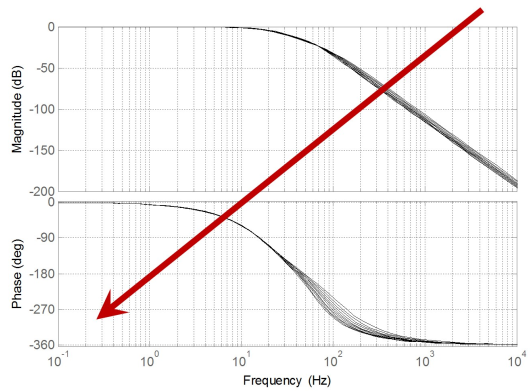
Figure 3.2.: Block Diagram of Servo-hydraulic System

Table 3.1.: Servo-hydraulic system parameters for system identification

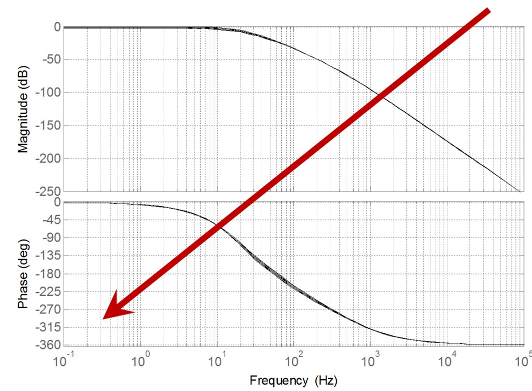
k_q	Valve Flow Gain	K_v	Valve Pressure Gain
τ	Servo-valve Time Delay Constant	C_l	Piston leakage coefficient
A	Piston Area	V_t	Fluid Volume
K_p	Internal Controller Proportional Gain	β_e	Effective Bulk Modulus
m	system and piston mass	c	system and actuator damping

3.1.4 Transfer Function Sensitivity

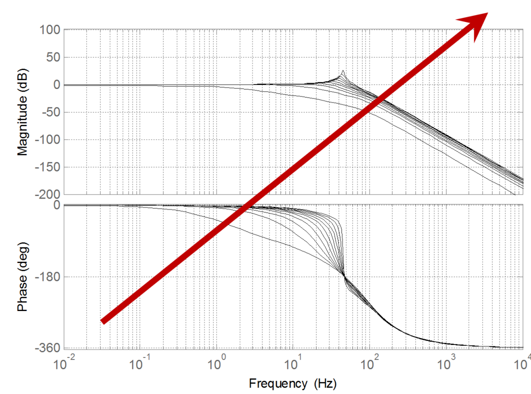
To understand this linearized system model, the influence of specimen parameters on the system transfer function is investigated. Specimen mass m_L, stiffness k, and internal controller gain K_p are the three variables to consider. Thus, they were each varied over a range to understand their influence on the transfer function.



(a) Mass Variation



(b) Stiffness Variation



(c) Controller Gain

Figure 3.3.: System Transfer Function Sensitivity to System Variables (Directions of Arrows Indicate Variable Increasing)

Fig. 3.3 shows the sensitivity of the system transfer function with specimen mass, stiffness and internal controller gain in numerical simulations. In each of the three cases the parameter (m, k and K_p) varied between 10% to 1000% of the original value. The controller gain variation clearly has the most significant effect on the shape of the transfer function. Specimen mass and stiffness variation do not change the transfer function extensively, especially in terms of phase lag. However, it is worth noticing that innately, the system transfer function does yield a unit gain when the stiffness is not zero, which is due to the $C_l k$ in the denominator. This phenomenon will not affect the functionality of the model in most cases since $C_l k$ is usually relatively small compared to the numerator. But when the specimen stiffness is large (e.g., a stiff frame) the transfer function needs to be modified to satisfy the static condition. A general procedure to adjust the transfer function for large stiffness specimen case is not within the scope of this chapter.

3.2 Genetic Algorithm

Genetic algorithms (GA) is an optimization technique using the concepts of natural evolution and the survival of the fittest. Compared to the straightforward greedy searching criterion which requires high computing power, GA has two intrinsic advantages. First, it searches many peaks in the population in parallel, and exchange information within the peaks to search broadly [75], increasing the possibility to converge to the global optimum. Thus, the selection of initial population range is not critical. On the other hand, the algorithms starts with an initial population randomly distributed over all dimensions of the search space which considerably reduces the time required in searching. The advantages of GA make it an appropriate choice in servo-hydraulic actuator identification where there are a large number of unknowns.

GA has five basic operations: initialization, selection, crossover and mutation. The structure of this algorithm is shown in Fig. 3.4.

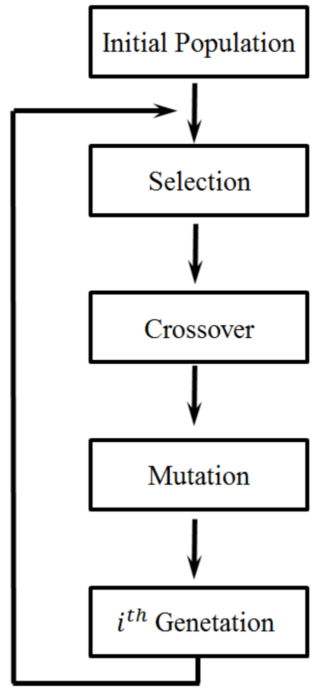


Figure 3.4.: General Genetic Algorithms (GA) Processing Structure

To start evolution, the first generation (i.e., parents) is generated and distributed randomly over the boundaries within which they are defined. The population size defines the number of first generation vectors, or can also be understood as the number of chromosomes. Increasing the population usually yields a better result [71], [76], but having too large of a population decreases computational efficiency [77]. The initial population for a particular element is generated by choosing n sample values using the rule

$$x_i = z_i(x_{\max} - x_{\min}) + x_{\min} \tag{3.7}$$

where a random variable $z_i[0, 1]$, x_{max} is the upper bound for x_i, and x_{min} is the lower bound. They are then clustered together to form an initial population in $n \times m$ matrix form where n is the population size and m is the number of unknown parameters.

The fitness function defines the rules to rank the optimized search. In the system identification case, a better value from the fitness function indicates a better fitting of the parameter-based model for the sampled data under various test specimen and input signal configurations. Based on the fitness of each vector, they are re-arranged in the population matrix with the fittest vector at the top. A selection ratio (ζ) is predefined for the algorithm to indicate the portion of population left after a generation of natural selection. The second generation would then have a population of $n \times (1 - \zeta)$ with the least fit $n \times \zeta$ vectors removed from the matrix, or dying out from the population.

Similar to the natural evolution process, the vectors that survive the natural selection process mate with each other to exchange gene information. The organized matrix generated in the last stage is randomly re-arranged in the column to form new vectors. This process is visualized in Fig. 3.5 using a simple 56 matrix example. Ψ is the crossover operator.

$$\text{Old Vector 1} \rightarrow \quad \Psi \begin{bmatrix} \boxed{a_1 \;\; b_1 \;\; c_1 \;\; d_1 \;\; e_1 \;\; f_1} \\ a_2 \;\; b_2 \;\; c_2 \;\; d_2 \;\; e_2 \;\; f_2 \\ a_3 \;\; b_3 \;\; c_3 \;\; d_3 \;\; e_3 \;\; f_3 \\ a_4 \;\; b_4 \;\; c_4 \;\; d_4 \;\; e_4 \;\; f_4 \\ a_5 \;\; b_5 \;\; c_5 \;\; d_5 \;\; e_5 \;\; f_5 \end{bmatrix} \triangleq \begin{bmatrix} \boxed{a_3 \;\; b_4 \;\; c_5 \;\; d_1 \;\; e_2 \;\; f_3} \\ a_1 \;\; b_2 \;\; c_1 \;\; d_3 \;\; e_4 \;\; f_5 \\ a_5 \;\; b_5 \;\; c_4 \;\; d_5 \;\; e_1 \;\; f_2 \\ a_4 \;\; b_1 \;\; c_2 \;\; d_2 \;\; e_3 \;\; f_4 \\ a_2 \;\; b_3 \;\; c_1 \;\; d_4 \;\; e_5 \;\; f_1 \end{bmatrix} \quad \leftarrow \text{New Vector 1}$$

Figure 3.5.: General Crossover Process for GA

$$\Omega \begin{bmatrix} a_3 & b_4 & c_5 & d_1 & e_2 & f_3 \\ a_1 & b_2 & c_1 & d_3 & e_4 & f_5 \\ a_5 & b_5 & c_4 & d_5 & e_1 & f_2 \\ a_4 & b_1 & c_2 & d_2 & e_3 & f_4 \\ a_2 & b_3 & c_1 & d_4 & e_5 & f_1 \end{bmatrix} \triangleq \underbrace{\begin{bmatrix} a_3 & b_4 & c_5 & d_1 & e_2 & f_3 \\ a_1 & b_2 & c_1 & d_3 & e_4 & f_5 \\ a_5 & b_5 & c_4 & d_5 & e_1 & f_2 \\ a_4 & b_1 & c_2 & d_2 & e_3 & f_4 \\ a_2 & b_3 & c_1 & d_4 & e_5 & f_1 \end{bmatrix}}_{\text{Original Matrix}} + \underbrace{\begin{bmatrix} 0 & 0 & M_c & 0 & 0 & 0 \\ M_a & 0 & 0 & 0 & M_e & 0 \\ 0 & 0 & 0 & 0 & 0 & M_f \\ 0 & M_b & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & M_d & 0 & 0 \end{bmatrix}}_{\text{Mutation Matrix}}$$

Figure 3.6.: General Mutation Process for GA

The matrix after crossover lays the foundation of the next generation vectors. From the biological perspective, mutations occur in the process of mating. In GA, mutations are also simulated to search for possible global optimum outside of the current population. A mutation rate is defined for GA so that after a new generation of population is generated in the form of matrix, Fig. 3.6 shows the mutation process for the example matrix considered previously. Ω is the mutation operator. A mutation rate μ is defined for GA so that after a new generation of population is generated in the matrix, $\mu \times n$ of the elements in each column is substituted by a random number within the assigned boundaries. The exact positions in each column to insert these numbers are chosen randomly. In this identification procedure, μ is selected to be 0.2.

After crossover and mutation are complete, the next generation of population is generated. This process will continue iterating until the stop criterion is met. The stop criterion can either be set such that the population is less than a threshold, or the fitness for a particular vector is larger than a particular value. The last vector left or the arithmetic average of each element in the last few vectors is considered as the GA optimization results. If the last few vectors still have a large standard deviation, another round of GA with narrower boundaries may be required, or it can be combined with an exhaustive searching method using greedy criterion to finalize the parameters.

To utilize GA for servo-hydraulic actuator identification, each of the unknown system parameters in Table 3.1 is assigned an estimated range. The initial population of each parameter is then generated randomly within the range. After selecting the appropriate fitness function and stop criteria, selection, crossover and mutation are conducted to yield a set of system parameters that best reproduce the response recorded from the experiments.

3.3 Experimental Implementation

Experiments are conducted in the Intelligent Infrastructure Systems Laboratory at Purdue University. A Shore Western 910D actuator with 6.0 inch (152 mm) stroke and 1.1 kip (4,893 N) force output is used as target actuator. Actuators of similar size are widely used in RTHS to drive the physical specimens, usually magnetorheological (MR) dampers [40], [73]. The actuator has an internal LVDT (G.L. Collins, LMT-711P34) to measure displacement. It is controlled by a Schenck-Pegasus 162M servo-valve rated for 15 GPM (3.41 m^3/hr) at 3000psi (20,684 kPa). This servo-valve has a nominal operating range of 0 to 60 Hz. The system in the lab is shown in Fig. 3.8. A high performance Speedgoat (Speedgoat GmbH, 2011) real time kernel with Core i5 3.6GHz processor is configured as the real-time target machine. The desired displacement command is generated from a Matlab host computer and compiled onto the target machine to send real-time command to the Shore Western SC6000 hydraulic system controller. All measurements are carried out at a sampling frequency of 4096Hz. The time domain data is then converted to the frequency domain using the fast Fourier transform (FFT).

The physical specimens for system identification are a series of spring-mass sets. These specimens are well-understood and can thus be helpful to separate the information of test specimen from the actuator system. A connector is attached to the actuator rod to make it compatible with various loading configurations. The structural configurations shown in Fig. 3.7 are applied in our system identification procedure.

For each test specimen, three proportional gain and displacement amplitude values are used in an effort to derive a model applicable to a wider range of interest. The proportional gains and amplitudes chosen are: K_p=5000,7000,10000; Amplitude=0.02,0.05,0.1 in. To fully excite the system dynamics across the frequency of interest, a band-limited white noise (BLWN) with a bandwidth of 0-80 Hz is chosen as the excitation for system ID.

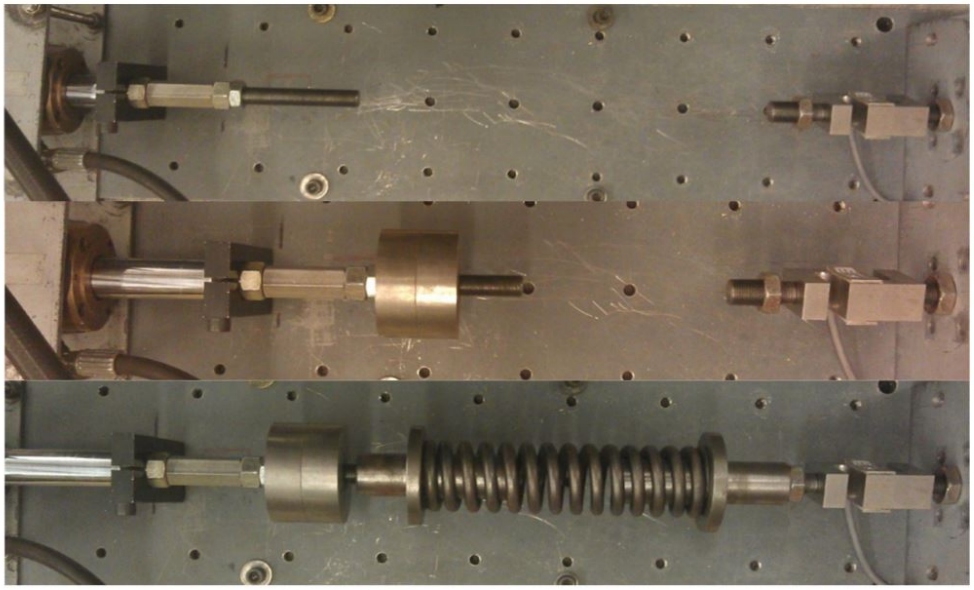


Figure 3.7.: Different Specimens Used in Servo-hydraulic System Identification

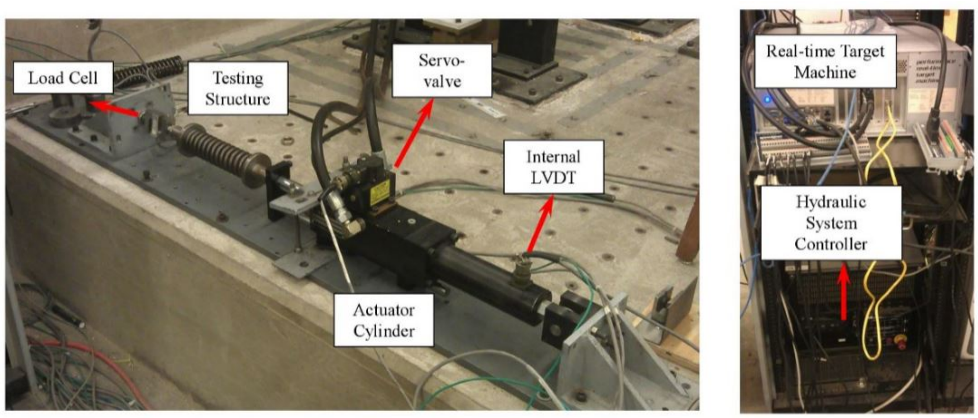


Figure 3.8.: Actuator and Experimental Setup for Servo-hydraulic System Identification

3.3.1 Identification Results

The initial population size in GA is 200. The boundary conditions for each parameter are listed in Table 3.2. Mean Square (RMS) error between the measured displacement of the actuator and the calculated displacement based on the purposed model was chosen to construct the fitness function.

Table 3.2.: Boundary Values for Unknown Parameters

Parameter	Z_0	m_0	K_a	C_l	T_v	c
Unit	$m^3/(s \cdot kPa)$	kg	m^3/kPa	$m^3/(skPa)$	s	$kN - s/m$
Minimum	1E-5	1	2E-12	2E-9	0.001	0.8
Maximum	2E-5	10	2E-10	2E-8	0.006	5

The fitness of the model with each test case is defined as

$$F(i,j) = \frac{\sigma(x_{m,j} - x_{i,j})}{\sigma(x_{m,j})} \tag{3.8}$$

where $F(i,j)$indicate the fitness of vector $i$for test case j, $x_{m,j}$ is the measured output for test case $j$and $x_{i,j}$is the calculated output for test case $j$based on the model predicted by parameter vector i. σ indicates standard deviation. The fitness for parameter vector i is than calculated by $F(i) = \sum_{j=1}^{N} F(i,j)$ where N is the number of transfer functions we have generated from the experiment. The selection rate and mutation rate in our study are 0.8 and 0.2, respectively. Reproduction stops when the population size reduces to 5.

The idea behind GA is to extract the fittest information from the initial population. Fig. 3.10 shows the evolution of GA for two specific system characteristic values plotting over 12 generations: the Z_0 value defined before and the viscous damping in the actuator c. As can be observed through the scatter, the initially randomly distributed data within the boundaries converges to a smaller region of the domain as the data go through the natural selection process. This trend can also be visualized by the cost function value defined by the Eq. 3.8. As population generation increases, this value decreases consistently, reaching a stable but low value when the algorithm meets the tolerance. The identified parameters of the system transfer function are tabulated in Table 3.3.

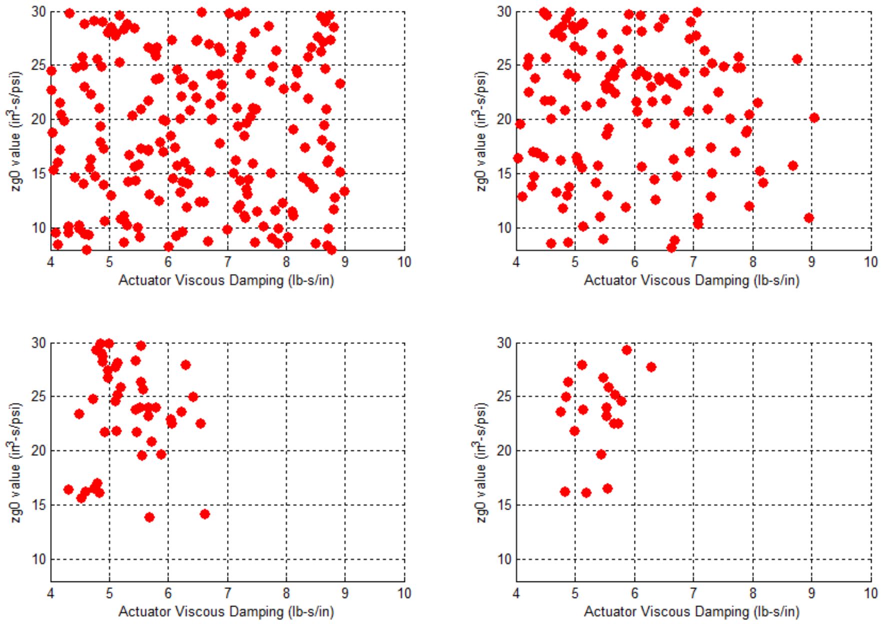


Figure 3.9.: Evolution of Actuator Damping and z_0 Value

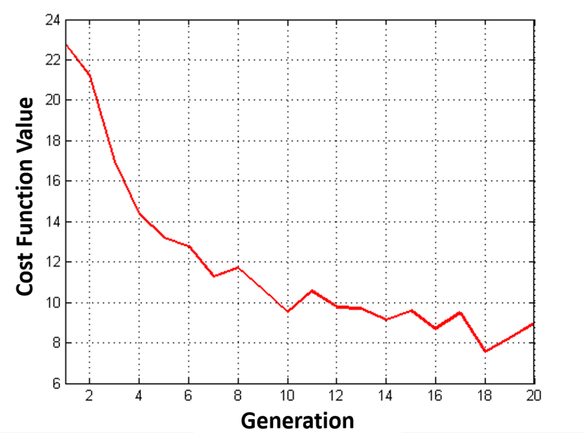


Figure 3.10.: Fitting Function Value Decreasing Trend

Based on the identified K_a value, and a rough estimation of the actuator cylinder volume using piston area and stroke, the effective bulk modulus of the fluid is estimated as

$$\beta_e = \frac{1}{4} \times \frac{V_{chamber}}{K_a} = \frac{6 \times 10^{-4} m^3}{4 \times 1.39 \times 10^{-10} m^3/kPa} = 1.1 \times 10^6 kPa \qquad (3.9)$$

Table 3.3.: Identified system parameters

	Parameter	Value	Unit
Z_0	$K_{vp}K_cK_qA$	1.48E-05	$m^3/(s \cdot kPa)$
v	Valve time delay	0.0036	s
C_l	Piston leakage coefficient	1E-8	$m^3/(s \cdot kPa)$
c	Effective actuator damping	3.2	$kN-s/m$
m_0	Actuator initial mass	4.25	kg
K_a	$V_t/4\beta_e$	1.39E-10	m^3/kPa

where $V_{chamber}$ is an estimation of the chamber volume based on bore diameter and chamber length. A reference value of bulk modulus for hydraulic oil at the lab temperature is approximately $1.5 \times 10^6 kPa$ [78] indicating a high fidelity in the identified parameters.

3.3.2 Experimental Verification

To verify this method a series of similar tests with the internal controller proportional gain K_p set differently as 6. This choice is made because, according to our study on the influence of each parameter on the transfer function in Section 2.1.5, K_p can most significantly change the system transfer function and K_p=6000 cases are not part of the fitness function in GA.

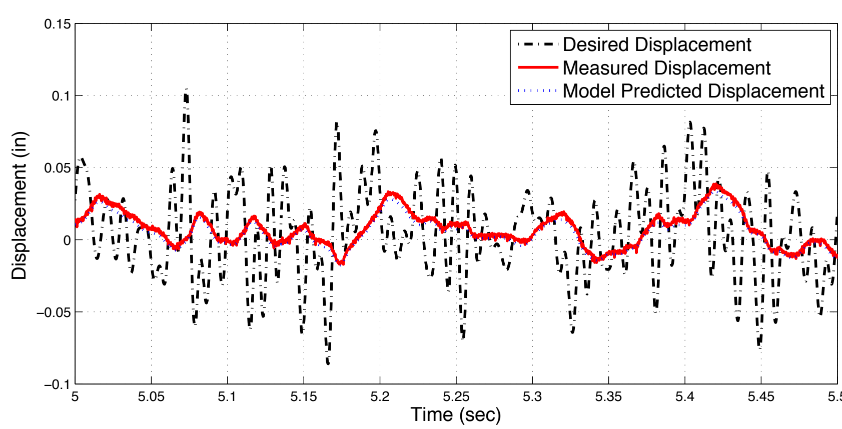
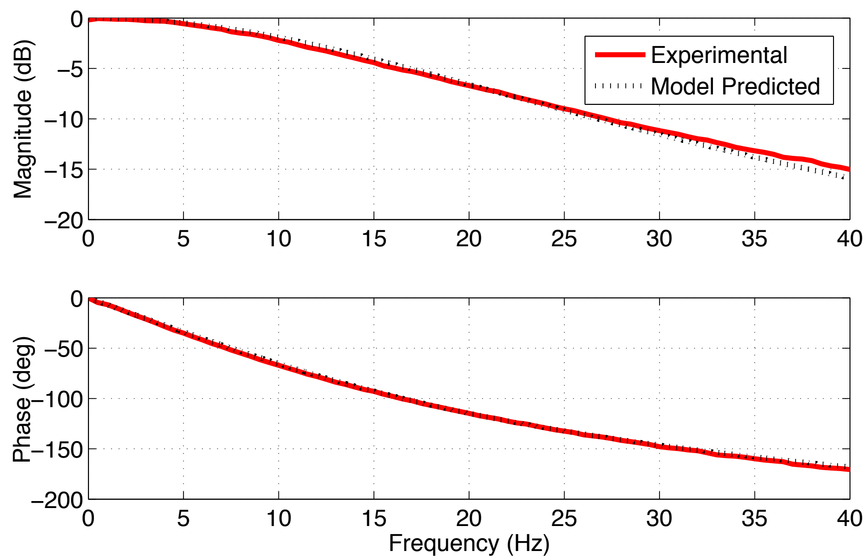
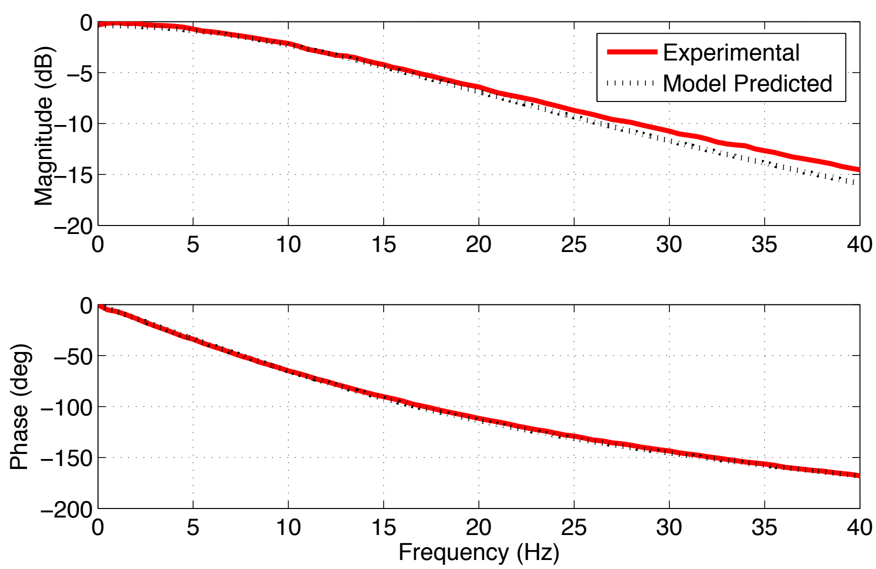


Figure 3.11.: Time Domain Verification of the Identified Model



(a) No Structure Attached, K_p=6000, RMS Amplitude=0.02 in



(b) Mass=0.98 kg, Stiffness=67.6 kN/m, K_p=6000, RMS amplitude=0.1 in

Figure 3.12.: Frequency Domain Verification Results for Servo-hydraulic System Identification

Figs. 3.11 and 3.12 show the time domain and frequency domain verification of the servo-hydraulic actuator model under BLWN excitation in experiments. From the time domain perspective, the model accurately predicts the system amplitude and time lag for a BLWN input signal. This conclusion is further confirmed by the transfer function, where perfect tracking is observed up to 30 Hz. When the frequency goes higher than 30 Hz the modeled amplitude starts to deviate from the experiment results, while the modeled phase still matches perfectly. In general, since the frequencies higher than 20 Hz are not typically of interest in earthquake engineering [79], the model generated by this procedure would be sufficient to help the modeling and control design of any similar servo-hydraulic actuator involved in RTHS.

3.4 Conclusion

This chapter presents a high fidelity and efficient general procedure to identify servo-hydraulic actuators involved in RTHS and similar experimental applications. The system transfer function is derived from a fourth-order, component-based model of the servo-hydraulic system. Known parameters from the design of the actuator and the physical test setup are employed in the model. The remaining parameters in the experimental transfer function are identified using a series of white noise inputs with varying physical components. Genetic algorithms is used to optimize system characteristic parameters in this transfer function. Various test specimens are attached to the actuator, and the dynamics in each case is compared with the mathematical model to generate a fitness function in GA. The proposed approach is found to be highly efficient and have fast convergence for this application and the results of the parametric identification is demonstrated to be effective. The resulting model is evaluated using different specimen cases and yields a good match between experimental results and predicted responses.

4. ACTUATOR CONTROL ALGORITHMS AND PERFORMANCE

Real time hybrid simulation (RTHS) has tthe additional benefit of enabling investigating rate dependent components in the experimental substructure. Some challenges of RTHS have been stated in earlier chapters, including accurate boundary condition tracking, integration stability and experimental-numerical interaction in a closed loop form. In 1996, Horiuchi *et al* [29] first drew the conclusion that time lag in RTHS, modeled as a pure time delay, is equivalent to adding negative damping. When the time delay is significant, instability may occur in the closed loop RTHS. In response to this issue, many researchers have sought to compensate for the time delay. First, Horiuchi *et al* used polynomial extrapolation based on displacement to predict and compensate the delay effect [87]. Later, Horiuchi and Konno linearly extrapolated the acceleration instead, to improve stability [30]. Darby *et al* introduced the first online estimation of time delay, and compensated for the updated time delay is compensated in real time [32]. Ahmadizadeh *et al* improved time delay estimation based on reference and measured signal slope [33]. Nguyen and Dorka also developed phase lag compensation with online system identification, where a black box with recursive estimation is used in online updating for system delay [34]. Wallace extended the work of Horiuchi and Darby, updating the polynomial coefficient by a least squares algorithm in real time to compensate for actuator dynamics [31]. Recently, Wu *et al* proposed an upper bound delay compensation algorithm [37].

Another important branch in the actuator control literature is the model based control approach, which was first proposed by Carrion and Spence [38], [39]. They introduced a data based actuator system identification and inverted the identified plant model directly by adding a low pass filter to ensure the system strictly proper. Phillips and Spencer modified FF compensation to eliminate of low pass filter and using a time domain representation to replace the improper inverse of actuator plant, one linear quadratic regulator (LQR) is added as optimized feedback regulation for uncertainties [40]. Combining the model based approach with delay/lag compensation, Chen and Ricles assumed the actuator plant can be modeled as a first order transfer function, and compensated with an inverse controller that uses a single estimation parameter representing the time delay/lag in actuator [35]. They later modified their inverse control algorithm with an adaptive feature to update the delay estimation parameter in real time [36]. In 2011, Gao *et al* proposed a robust actuator controller for RTHS based on H_∞ theory which, for the first time, considered noise, disturbance and model uncertainty effects in actuator control [42], [43], [44]. This H_∞ controller is designed by trading off tracking performance and control robustness, giving the user the power to meet the testing needs.

All the aforementioned control algorithms require model and parameter identification of the actuators with different level of complexity. However, the effect of modeling error on the control algorithm performance has not been studied. Meanwhile, for a dynamic system control, noise and disturbance is considering as another source affecting control performance. Thus, it is necessary to investigate the noise effect on different control algorithms.

This chapter includes a literature review and parametric study on existing actuator control algorithms. Three commonly used control algorithms in RTHS are studied, including the inverse control algorithm proposed by Chen and Ricles [35], which is a feedforward control algorithm; the feedforward and feedback algorithm (FFFB) by Phillips and Spencer [41], and the H_∞ control algorithm by Gao and Dyke [44]. First, the control parameters are defined in each algorithm. Simulations are carried out to understand how the design of these control parameters can affect the tracking performance. In the end, an experimental example is utilized to investigate how the tracking performance is affected according to modeling error and system noise.

4.1 Actuator Control Algorithms in RTHS

In this section, three actuator control algorithms are presented with design parameters indicated. In the inverse control algorithm, the system is modeled as a first order time delay system with constant parameter α; in the feedforward and feedback control algorithm, a feed forward loop is designed in time domain as a direct inverse of a third order plant model, and the linear quadratic regulator (LQR) feedback loop is implemented by appropriate weighting matrices Q_{LQR} and R_{LQR}; and, in the H_∞ control algorithm, the goal is to design a desired open loop shaping function $G_d(s)$.

4.1.1 Inverse Control Algorithm Formulation

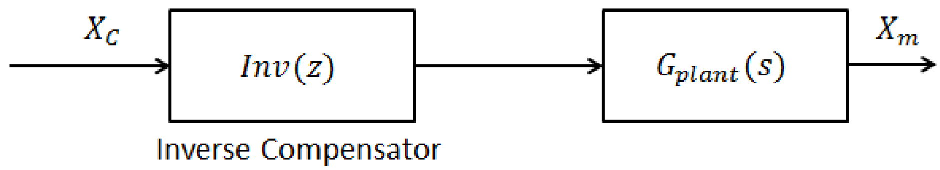


Figure 4.1.: Block Diagram of the Inverse Control Algorithm

Chen and Ricles proposed the inverse control algorithm for actuator time delay, shown in Fig. 4.1. The time delay $t_d = \alpha \cdot \delta t$ can be calculated by:

$$G(s) = \frac{X_m(s)}{X_c(s)} \tag{4.1}$$

$$t_d = \frac{-\angle G(\omega)}{\omega} \tag{4.2}$$

where $G(s)$ is the transfer function between measured signal $X_m(s)$ and command signal $X_c(s)$, time delay is represented by phase lag of transfer function $G(s)$ at circular frequency ω.

Assuming the response of actuator can be linearly interpolated at the i^{th} step, then measured signal at $i+1^{th}$ step is expressed as following:

$$X_m{}^{i+1} = X_m{}^i + \frac{1}{\alpha} \cdot (X_c{}^{i+1} - X_m{}^i) \tag{4.3}$$

Applying discrete z transformation,

$$G_d(z) = \frac{X_m(z)}{X_c(z)} = \frac{z}{\alpha \cdot z - (\alpha - 1)} \tag{4.4}$$

To compensate time delay, use the inverse of $G_d(z)$ as a feedforward block, the inverse controller $G_c(z)$ is derived to be

$$G_c(z) = \frac{1}{G_d(z)} = \frac{X_c(z)}{X_m(z)} = \frac{\alpha \cdot z - (\alpha - 1)}{z} \tag{4.5}$$

In the inverse control algorithm, the only parameter that affects control performance is α. Later, the sensitivity of the change in α to control performance is discussed.

4.1.2 Feedforward-Feedback Control Formulation

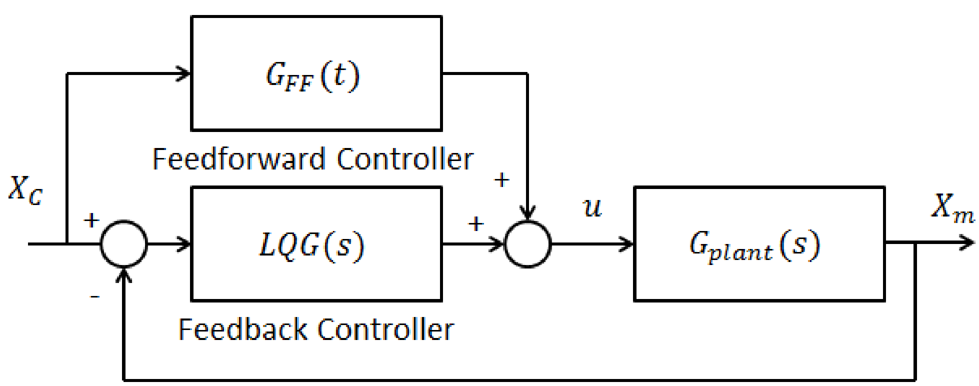


Figure 4.2.: Block Diagram of Feedforward and Feedback Controller Components

Carrion and Spencer proposed a feedforward-feedback control algorithm for real-time servo-hydraulic system [39]. The feedforward controller is formulated as the inverse of an identified actuator-specimen plant $G(s)$ and a low pass filter to stabilize the system. Phillips and Spencer reformulated the feedforward - feedback algorithm to a regulation problem, which includes a linear quadratic Gaussian (LQG) based regulator in the feedback loop to reduce residual tracking error after feedforward control. The previous low pass filter is eliminated in the reformulation to get rid of the unwanted dynamics induced by the low pass filter. The schemetic drawing of feedforward and feedback controller is as as Fig. 4.2.

Feedforward Controller

The plant model $G(s)$ is identified based on a open loop test (with no additional controller applied) with physical specimen attached to the hydraulic system. A typical third order servo-actuator transfer function is presented to describe transfer function between measured displacement and command displacement, as

$$G(s) = \frac{X_m(s)}{X_c s} = \frac{K_n}{(s-a)(s-b)(s-c)} \tag{4.6}$$

where, K_n is a constant zero polynomial coefficient and a, b, c are the poles of the plant. To directly compensate servo actuator dynamics, the inverse of $G(s)$ can be written in frequency domain as:

$$G_{FF}(s) = \frac{1}{G(s)} = \frac{(s-a)(s-b)(s-c)}{K} = a_0 + a_1 s + a_2 s^2 + a_3 s^3 \tag{4.7}$$

where a_0, a_1, a_2, a_3 are pole polynomial coefficients expanded from Eq. 4.6. In the time domain, Eq. 4.7 can be written as:

$$G_{FF}(s) = a_0 x + a_1 \dot{x}_c(t) + a_2 \ddot{x}_c(t) + a_3 \dddot{x}_c(t) \tag{4.8}$$

In general, the equations of motion are solved at the i^{th} time step and the $i+1^{th}$ displacement is imposed to the physical specimen. In discrete form, Eq. 4.8 can be written as:

$$G_{FF}(t) = a_0 x_c^{i+1} + a_1 \dot{x}_c^{i+1} + a_2 \ddot{x}_c^{i+1} + a_3 \dddot{x}_c^{i+1} \tag{4.9}$$

where the desired acceleration is assumed to be linearly extrapolated over one time step:

$$\ddot{x}_c^{i+1} = 2\ddot{x}_c^{i} - \ddot{x}_c^{i-1} \tag{4.10}$$

thus, $\dot{x}_c^{i+1}$ and $\dddot{x}_c^{i+1}$ can be written as $\dot{x}_c^{i+1} = \dot{x}_c^{i} + \frac{\Delta t}{2}(\ddot{x}_c^{i} + \ddot{x}_c^{i+1})$ and $\dddot{x}_c^{i+1} = \frac{1}{\Delta t}(\ddot{x}_c^{i} - \ddot{x}_c^{i-1})$.

Feedback Controller

Because the feedforward control part is determined by the estimation of the plant model which implies that modeling error might affect the performance. The use of LQG control in the feedback is aimed to reduce the tracking error with respect to modeling error and disturbances.

The system is written in state space form as:

$$\dot{\tilde{z}} = A\tilde{z} + Bu_{FB} + Ew_f \tag{4.11}$$

$$\tilde{y} = C\tilde{z} + v_f \tag{4.12}$$

where disturbance w_f and measurement noise v_f is introduced. Only output $\tilde{y} = X_m - X_c$ which is the error between the measured signal and the command signal can be observed all the time. Disturbances in the system are assumed to be a filtered Gaussian white noise after a second-order filter, the peak, bandwidth and roll off of the disturbance can be defined individually.

$$\dot{z}_f = Az_f + E_f w \tag{4.13}$$

$$w_f = C_f z_f \tag{4.14}$$

The augmented system can be written as: $z_a = \begin{bmatrix} z_f \\ \tilde{z} \end{bmatrix}$. and can be estimated by Kalman filter, with estimated state $\hat{z}_a$ and observer gain matrix L_{Kal}

$$\dot{\hat{z}}_a = A\hat{z}_a + Bu_{FB} + L_K(\tilde{y} - C\hat{z}_a) \tag{4.15}$$

Thus, the feedback control command u_{FB} can be obtained using an LQR design assuming full state feedback as:

$$J_{LQR} = \int [\tilde{y}^T Q_{LQR} \tilde{y} + u_{FB}^T R_{LQR} u_{FB}] dt \tag{4.16}$$

$$u_{FB} = -K_{LQR}\hat{z}_a \tag{4.17}$$

$$\dot{\hat{z}}_a = A\hat{z}_a - BK_{LQR}\hat{z}_a + L_{Kal}(\tilde{y} - C\hat{z}_a) \tag{4.18}$$

where K_{LQR} is the optimal state feedback gain matrix and J_{LQR} is the costs function minimized by LQR design, Q_{LQR} and R_{LQR} are the weighting matrices for output and system inputs respectively.

For the FFFB control algorithm, the design parameter is the weighting matrix Q_{LQR} and R_{LQR}.

4.1.3 H_∞ Control Algorithm Formulation

In 1989, Glover and McFarlane introduced a control design procedure that applies loop-shaping methods for exploring performance/robust stability trade offs [81]. For a system represented by a transfer function $G(s)$ and controller $K(s)$ in Fig. 4.3 the system sensitivity function is defined in Eq. 4.19 to quantify disturbance and noise.

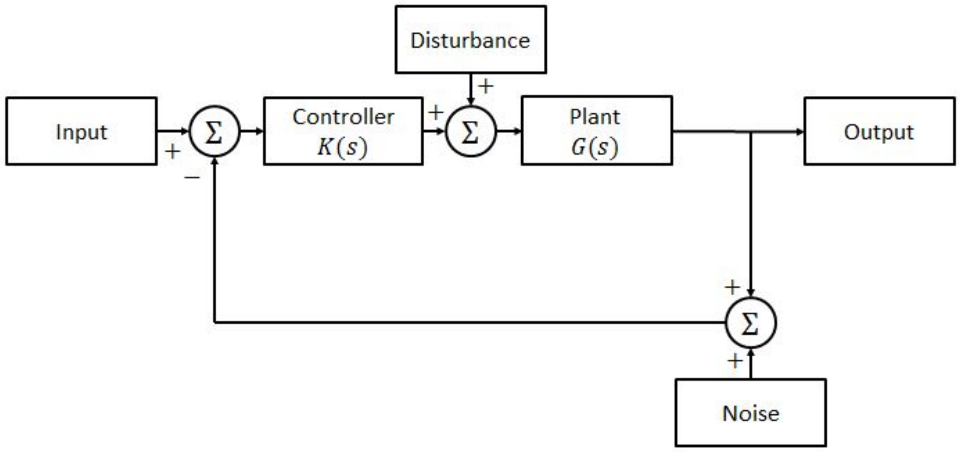


Figure 4.3.: Block Diagram of a Typical Feedback Control System

For tracking performance measurement, the complementary sensitivity function (also known as the output-input transfer function) is defined as Eq. 4.20.

$$S_0(s) = (1 + G(s)K(s))^{-1} \tag{4.19}$$

$$T_0(s) = 1 - S_0(s) = GK \cdot (1 + G(s)K(s))^{-1} \tag{4.20}$$

Consider a weighting function or a loop shaping function, $G_d(s)$ as

$$G_d(s) = \frac{T_0(s)}{S_0(s)} = \frac{G(s)K(s) \cdot (1 + G(s)K(s))^{-1}}{(1 + G(s)K(s))^{-1}} = G(s)K(s) \tag{4.21}$$

In the desired signal tracking frequency, T_0 should be close to unity, indicating good tracking performance. Meanwhile, S_0 is close to 0 based on Eq. 4.19, therefore, G_d is with its lower bound limit. In the high frequency range, where model uncertainties and noise effect are dominant, $S_0(s)$ needs to be large and $T_0(s)$ should be minimized, this indicates the feedback noise does not affect the control command signal. Thus, $G_d(s)$ is with upper bound limit. The upper bound, lower bound, and the loop shaping function satisfying such criterion are shown in following Figure 4.4.

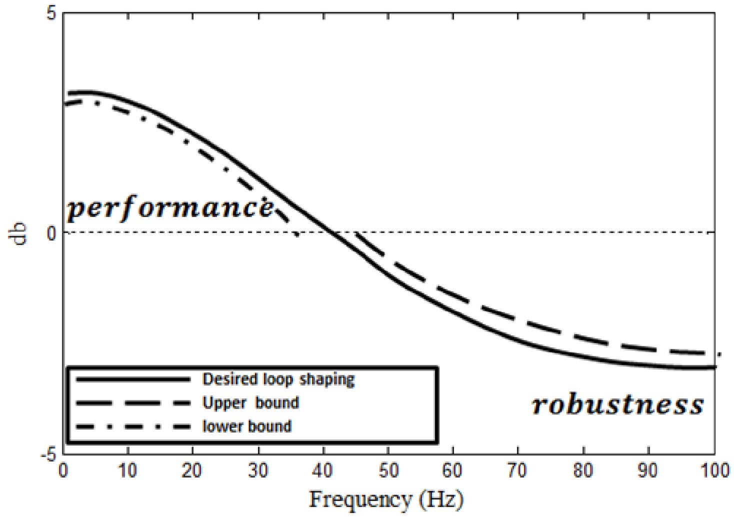


Figure 4.4.: Upper Bound and Lower Bound for Loop Shaping Function

H_∞ methods allows for optimization of a transfer function $F(s)$, with its maximum singular value over the entire frequency range, $s = j\omega$, j is the imaginary unit. The goal of the H_∞ control method is to design a controller $K(s)$ by following rule, where $\lambda = 1$ indicates the best fit.

$$||F(s)||_\infty = sup\overline{\sigma}(F(s)) = \sqrt{\lambda_{max} F(s) * F(s)} \tag{4.22}$$

$$\overline{\sigma}(G(s)K(s)) \geq \frac{1}{\lambda}\overline{\sigma}(G_d(s)) \tag{4.23}$$

$$\overline{\sigma}(G(s)K(s)) \leq \lambda\overline{\sigma}(G_d(s)) \tag{4.24}$$

In H_∞ control algorithm design, the effective design of the controller is based on the choice of the loop shaping function G_d. The parameters in designing the loop shaping function G_d is discussed in the numerical study.

4.2 Numerical Study on Actuator Control Design and Performance

In this section, the design procedures for three different control algorithms are presented. Control performance is defined with an RMS error indicator and time delay between desired and measured signal. The sensitivity of controller parameter to control performance and control effort (command output voltage) is discussed. Because the FFFB algorithm assumes linear extrapolation of acceleration, the plant model in this section is represented using a third order transfer function, identified from an experimental setup, as in Fig. 4.5. The plant transfer function is written:

$$G(s) = \frac{4.8507 \times 10^6}{s^3 + 353.3s^2 + 79264s + 4.8304 \times 10^6} \tag{4.25}$$

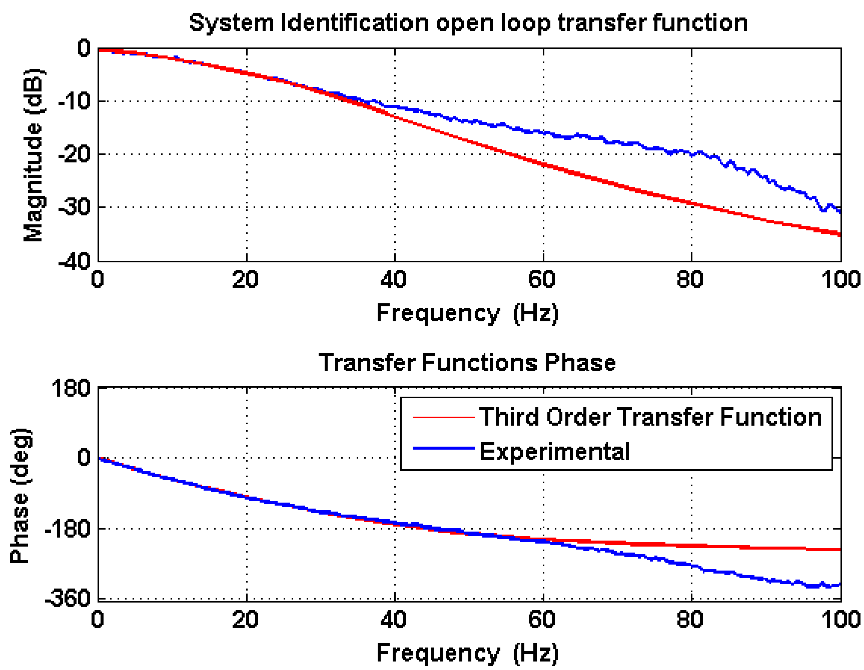


Figure 4.5.: Data Based Plant Frequency Response: Excited with 0-100 Hz BLWN Noise

The RMS error indicator used to determine the tracking performance in the time domain is defined as:

$$RMS_e = \frac{\sqrt{\Sigma(X_m - X_c)_i{}^2}}{\sqrt{\Sigma(X_c)_i{}^2}} \tag{4.26}$$

4.2.1 Parametric Study of Inverse Control Algorithm

In the inverse control algorithm, the only parameter that changes the performance is α which is the estimation for time delay. From Eq. 4.2, α can be estimated with given sampling frequency and time delay t_d. The estimated time delay t_d from experiment is 12.9 millisecond.

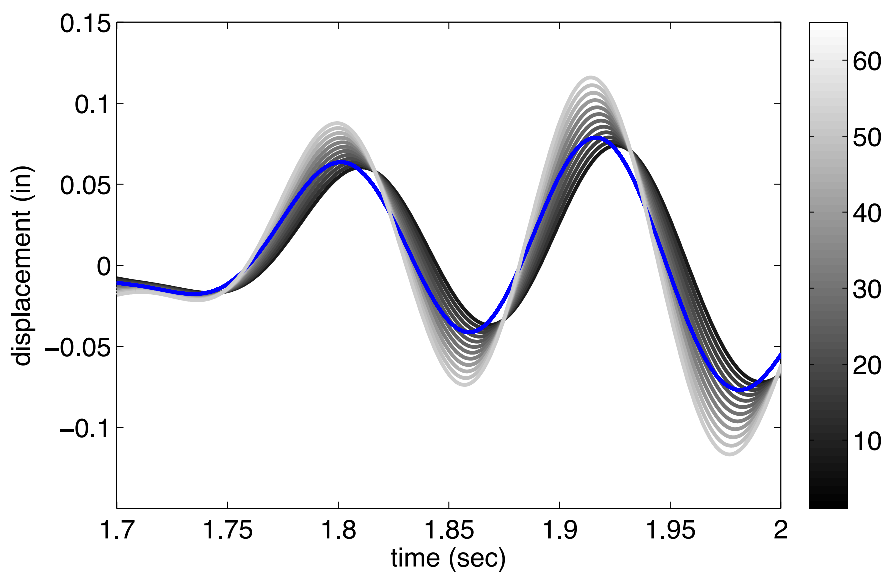


Figure 4.6.: Time Domain Signal Tracking Performance with Different Inverse Controller Designs

The design parameter α using estimated time delay changes from 7 milliseconds to 30 milliseconds (equivalent to $T_d = 14{-}60$ steps), at sampling frequency 2048 Hz. The time domain tracking performance is shown in Fig. 4.6 for a band limited white noise with frequency range of 0-10 Hz. The grey scale bar shows the corresponding

α value used in designing the controller. When the controller design is with a larger α value which is equivalent to assuming larger time delay in the system, the time delay is better compensated. However, when α value keeps increasing, the overshoot effect takes over the advantage of time delay compensation, which induces larger RMS error, as shown in Fig. 4.7.

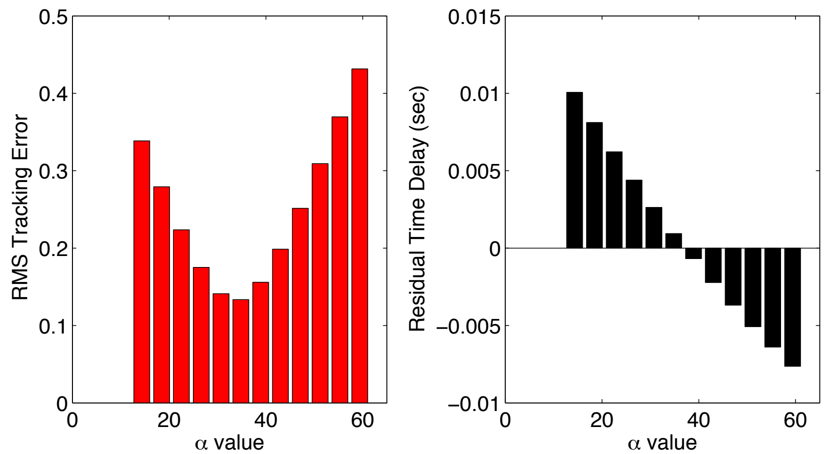


Figure 4.7.: RMS Error and Time Delay of Compensated System with Different Inverse Control Designs

4.2.2 Parametric Study of Feedforward and Feedback Control Algorithm

FFFB control algorithm performs an inversion of a third order transfer function and extrapolates to the time domain. The feedforward loop design depends on modeling accuracy, where a_0, a_1, a_2, a_3 value in Eq. 4.8 is derived from G in Eq. 4.25. Parameters in the feedback loop are flexibly selected based on the LQR algorithm. As in Eq. 4.16, the optimization weighting Q/R ratio determines the control gain J_{LQG}. In this section, Q/R ratio changes gradually between 1 to 4000, and the performance of FFFB algorithm is evaluated also in terms of the RMS error and the residual system time delay. Fig. 4.8 indicates that the larger the Q/R value, the better the control performance is. Also, the residual time delay is further reduced. With the FFFB controller, because the feedforward loop takes numerical differentiation of the measured signal and commanded signal, control output is sensitive for both signal amplitude and sampling frequency as in Fig. 4.9. The physical plant has a limitation on the input voltage, (commonly ± 10 volts), large input signal or running at high sampling frequency can be a potential problem for FFFB controller.

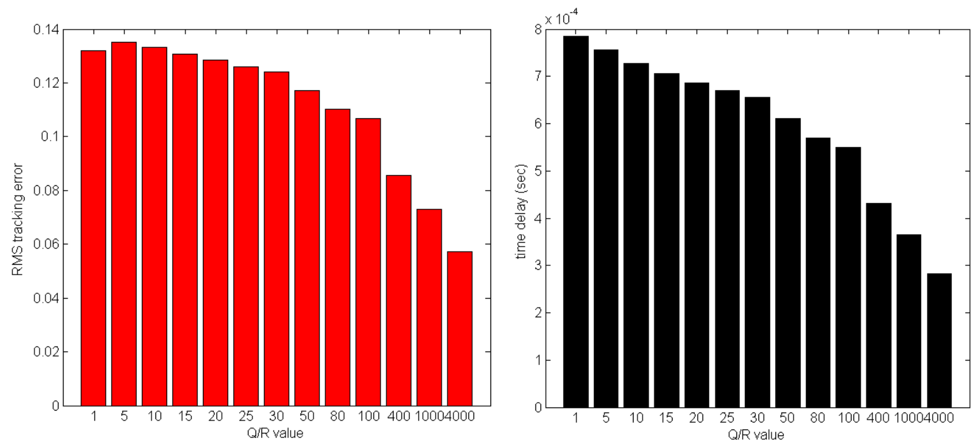


Figure 4.8.: RMS Error and Time Delay of Compensated System with Different FFFB Control Designs

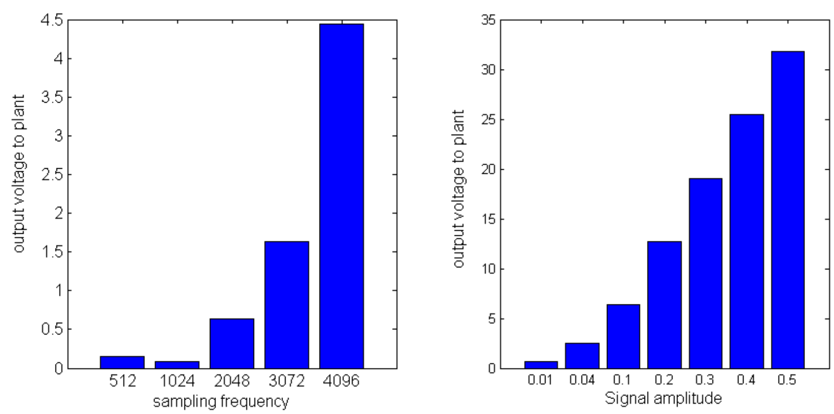


Figure 4.9.: Output Voltage of FFFB Controller under Different Sampling Frequencies and Signal Amplitudes

4.2.3 Parametric Study of H_∞ Control Algorithm

H_∞ control algorithm allows for a flexible trade off between controller robustness and performance. As mentioned earlier, the design of an H_∞ controller is equivalent to design to achieve a desired open loop transfer function G_d. The parameters in a shape function can vary broadly which makes control design less intuitive. In this section, only consider a second order transfer function for G_d as:

$$G_d(s) = \frac{G_{dc} \times ab}{(s+a)(s+b)} \tag{4.27}$$

where, $-a, -b$ are poles of the open loop system. The closed loop transfer function G_{CL} is:

$$G_{CL}(s) = \frac{G_{dc} \times ab}{(s+a)(s+b) + G_{dc} \times ab} \tag{4.28}$$

Rewriting Eq. 4.28 into complex form, its magnitude and phase can be expressed as:

$$G_{CL}(\omega) = \frac{G_{dc}[-\omega^2 + G_{dc} + ab - (a+b)j\omega]}{\omega^4 - (G_{dc} + 6ab + a^2 + b^2)\omega^2 + (G_{dc} + ab)^2} \tag{4.29}$$

$$|G_{CL}(\omega)| = \sqrt{\frac{G_{dc}[-\omega^2 + G_{dc} + ab - (a+b)j\omega]}{\omega^4 - (G_{dc} + 6ab + a^2 + b^2)\omega^2 + (G_{dc} + ab)^2}} \tag{4.30}$$

$$\theta_{CL}(\omega) = \arctan \frac{(a+b)\omega}{-\omega^2 + G_{dc} + ab} \tag{4.31}$$

Ensuring the closed loop transfer function will have a small phase lag (equivalent to time delay) requires a larger a, b values and larger constant gain G_{dc}. Solving Eq. 4.31 yields many parameter sets for a, b, G_{dc} which satisfy a desired predetermined cross over frequency. Simulation results for the 2^{nd} order open loop transfer function G_d are studied with a, b, G_{dc} parameters indicated in Fig. 4.10. It shows the larger the G_{dc}, a, b value, the better the control performance is(less RMS error and smaller time delay).

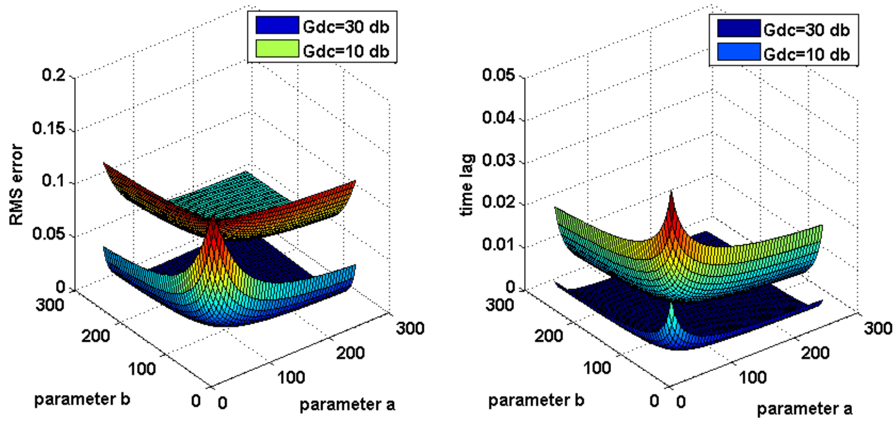


Figure 4.10.: RMS Error and Time Delay of Compensated System with Different H_∞ Control Designs

$-a, -b$ are open loop poles for the system, this finding can be expanded to a higher order open loop transfer function where

$$G_d(s) = \frac{G_{dc} \times a_1 a_2 a_3 ... a_n}{(s + a_1)(s + a_2)(s + a_3)...(s + a_n)} \tag{4.32}$$

However, during experimental testing, system noise, if existed, can be excited and amplified with high constant open loop gain G_{dc}. This effect is considered as robustness limitation and is discussed in detail in later sections.

In this section the controller parameters sensitivity are discussed, based on a simulation task, the key parameters of three commonly used actuator control algorithms are investigated. For the inverse control algorithm, when time delay associated parameter α increases, the residual time lag between measured plant output to command signal decreases. However, at large α, the tracking accuracy is reduced due to the signal overshoot. Based on the simulation, inverse controller works better for system with small delay. In FFFB controller, high Q/R ratio increases the tracking accuracy. This control algorithm is easy to design with third order transfer function information. However, it is found for tracking command at larger amplitudes, this algorithm generates a large output in voltage output reaches the limit of physical facility.

The H_∞ controller is the most flexible control algorithm among the three, the order of loop shaping function is configurable as well as the constant gain G_{dc} and loop shaping function pole locations. From simulation, it is found that the constant gain G_{dc} is the dominant parameter for control accuracy. However, a smaller pole value (more stable, further to the negative real axis), reduces the tracking accuracy (RMS error and larger delay). Again, that is the trade-off between accuracy and robustness for H_∞ controller design.

4.3 Experimental Study on Control Accuracy

In this section, an experimental study is carried out for illustrating the control performance of the same three control algorithms. The equipment and facility used for experimental investigation are located in the Intelligent Infrastructure System Lab (IISL). Due to its inherent rate dependent characteristic, the magneto-rheological (MR) damper is considered as a representative experimental substructure in RTHS. In the experimental setup, a small scale MR damper is attached to the same actuator used in chapter 3, the performance of linear control algorithms are investigated on such setup. An RD-8040-1 MR damper with a maximum force capacity of 450 lbf (2 kN) and a stroke of 2.2 inch (56 mm) manufactured by Lord Corporation is attached to the Shore Westsern 910D actuator. A Lord Wonder Box pulse-width modulator has been applied to control MR damper excitation input, the MR damper is considered in on mode with input of 3 Amps and off with input of 0 Amps.

As indicated in the previous chapter, the real-time control system is implemented using a Matlab xPC framework to ensure all physical components to meet the time constraints. A high performance Speedgoat/xPC real-time kernel is utilized as the target PC and it is used with a Core i5 3.6 GHz processor for intensive processing of complex model in real-time computation. Two 18-bit analog I/O boards with high accuracy are integrated into the digital control system which supports up to 32 input A/D channels and 8 output D/A channels at the same time. A Shore

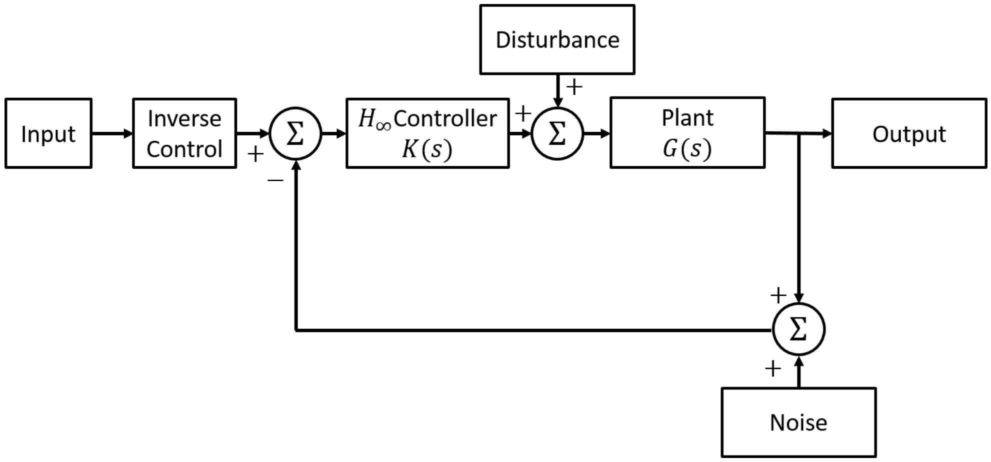


Figure 4.11.: Block Diagram of a combination of Inverse Controller and H_∞ Controller

Western SC6000 analog servo-controller is equipped in the testing to provide control of hydraulic actuators.

Four different control algorithms are used in the experimental investigation. In addition to the three discussed control algorithms, a new controller is proposed as a combination of the H_∞ controller and the inverse controller, the schematic block diagram in Fig. 4.11. Due to the effective improvement of inverse control on systems with small time delays, it is desired to include such a feedforward loop to eliminate any residual time lag after H_∞ when smaller open loop pole (a, b value as discussed earlier) is selected. This controller is later expanded to formulate the Robust Integrated Actuator Control (RIAC) algorithm, which is presented in detail in chapter 5.

Because all the control algorithms discussed in this chapter are based on linear system assumption, an approximated linearization of the MR damper - actuator nonlinear system is implemented. One common used approximation is performed as follows: 1) take transfer functions of the nonlinear system with band limited white noise (BLWN) displacement input by and measured displacement output, 2) run the transfer function identification with both MR damper off (0 amps current) and on (3 amps current) condition; 3) take the average of the approximated transfer function in step 2) to represent an averaged linear behavior among both conditions. The on

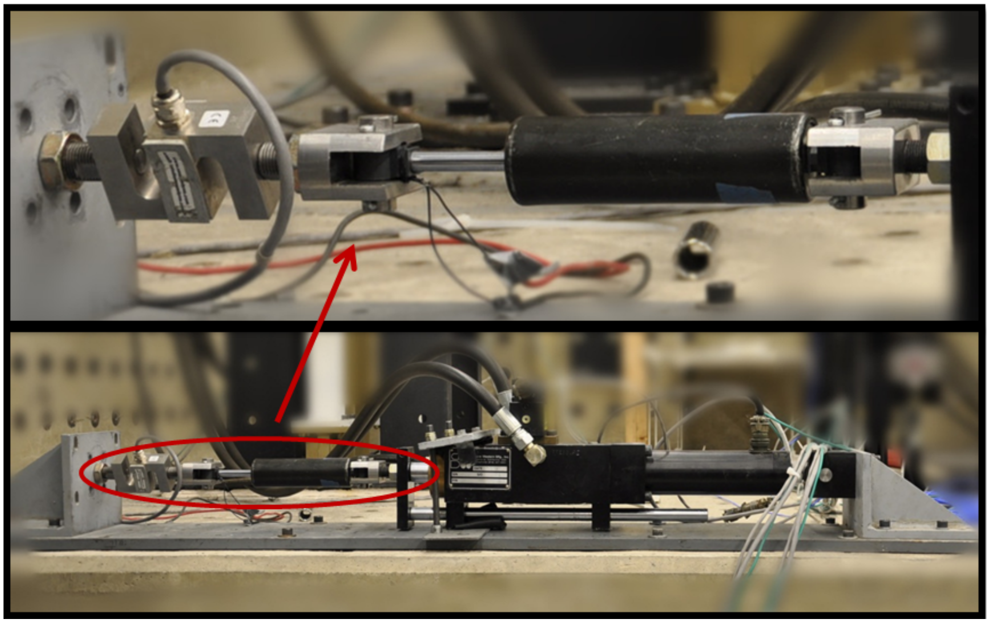


Figure 4.12.: Experimental Setup: MR damper Attached to Hydraulic Actuator

and off transfer functions with corresponding curve fitting results are shown in Fig. 4.13.

$$G_{off}(s) = \frac{1 \times 10^6}{s^3 + 287s^2 + 5 \times 10^4 s + 1 \times 10^6} \tag{4.33}$$

$$G_{on}(s) = \frac{7 \times 10^6}{s^3 + 467s^2 + 1.7 \times 10^5 s + 7 \times 10^6} \tag{4.34}$$

The averaged transfer function is as:

$$G_{avg}(s) = \frac{4 \times 10^6}{s^3 + 377s^2 + 1 \times 10^5 s + 4 \times 10^6} \tag{4.35}$$

Two input signals are utilized to investigate the control performance: one is a 0-20 Hz BLWN with an RMS amplitude of 0.03 inch, another is 0.5 Hz sinusoidal signal with amplitude of 0.3 inch. Parameters used for all controllers are listed in Table 4.1. For both tests, controller designs are based on the averaged transfer function in Equation 4.35. During the BLWN testing, MR damper input current was first set at

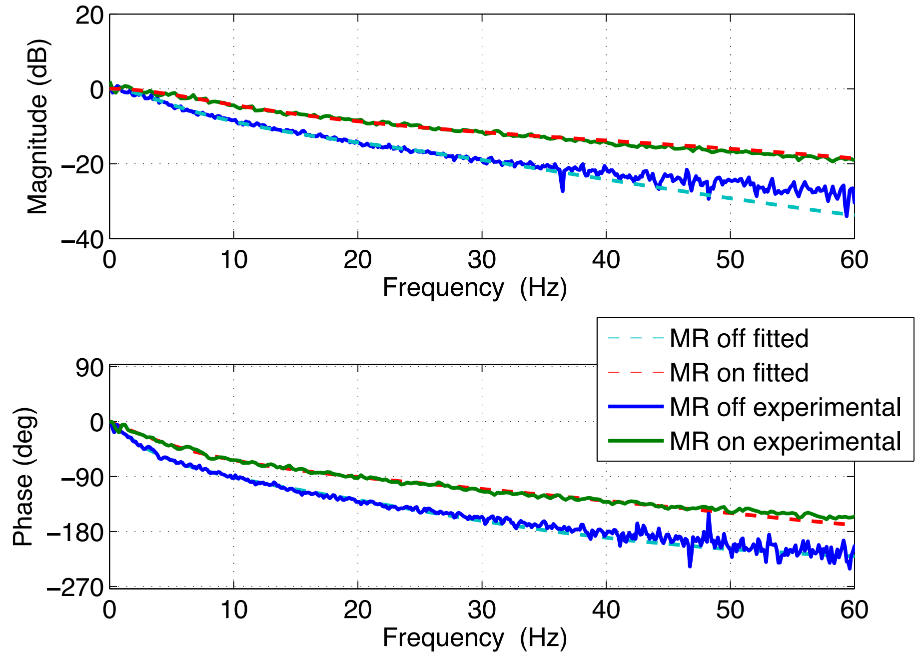


Figure 4.13.: Data Based Plant Frequency Response: Excited with 0-60 Hz BLWN Noise

0 Amps (off condition) and then 3 Amps (on condition). In sinusoidal signal testing, only MR damper off condition is considered.

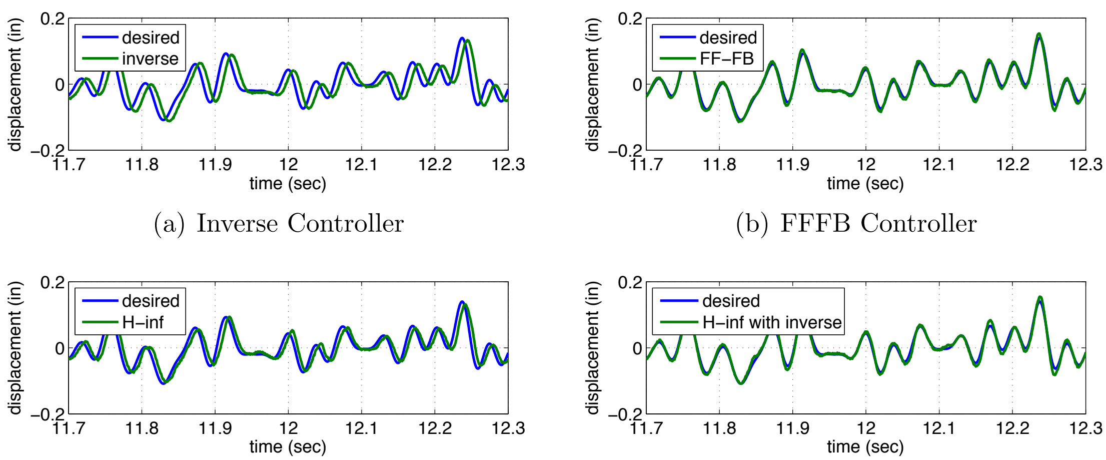
Table 4.1.: Design Parameters and Tracking Results for Different Control Algorithms

	Parmeter	RMS error MR Off, BLWN	RMS error MR On, BLWN	RMS error MR off, Sine
Inverse Controller	t_d=0.01 sec	0.77	0.79	0.026
Feedforward Feedback Controller	$Q/R = 400$	0.14	0.11	0.007
H_∞ Controller	$G_d(s) = \frac{1.883\times10^5}{s^2+600s+7500}$	0.49	0.54	0.05
H_∞ with Inverse Controller	$G_d(s) = \frac{1.883\times10^5}{s^2+600s+7500}$ T_{dH_∞}=0.006 sec	0.15	0.13	0.009

Table 4.2.: Testing matrix for system identification

Combination Case No.	m (lbf)	k (lb_f/in)	Proportional Gain
1	0	0	7000
2	1.77	0	7000
3	1.77	207	7000
4 (nominal case)	2	386	7000
5	4	0	7000
6	0	0	6000
7	1.77	0	6000
8	1.77	207	6000
9	2	386	6000
10	4	0	6000

The comparison results for different controllers are shown in Fig. 4.14 and 4.15. Quantified RMS errors and residual time delays are illustrated in Fig. 4.16. As observed from the time domain plots, the measured and desired displacements are closed to identical for FFFB and H_∞ with inverse controllers. Between on and off condition, time lags after inverse controller are not the same, which reveals the need of a more versatile control scheme for controlling such system. FFFB has slightly better performance for on condition, while the H_∞ with inverse controller has smaller RMS error for the off mode. In the frequency domain time lag plot, time lags for FF-FB controller shown to be flat and close to zero for both MR damper on and off modes. Residual delay is around 4-5 ms for H_∞ controller, and is well compensated by the additional inverse control algorithm in the modified H_∞ control design.



(a) Inverse Controller

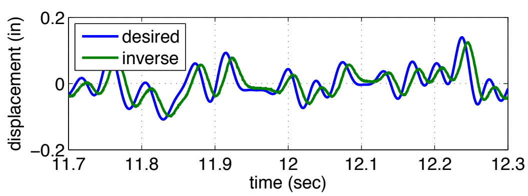
(b) FFFB Controller

(c) H_∞ Controller

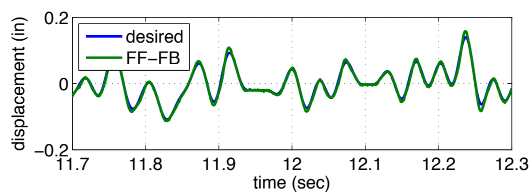
(d) H_∞ with Inverse Controller

Figure 4.14.: Time Domain Tracking Performance for Different Control Algorithms, MR damper Off Condition, Tracking of 0-20 Hz BLWN

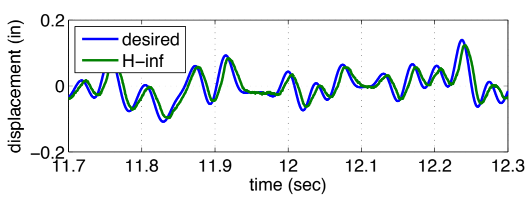
For sinusoidal signal, Fig. 4.17 shows quantified RMS errors and time domain tracking, all controllers have acceptable tracking accuracy. For FFFB control algorithm, and modified H_∞ control with incorporating inverse controller, RMS errors are both under 1%. Inverse controller is found to be more effective for a narrow band input.



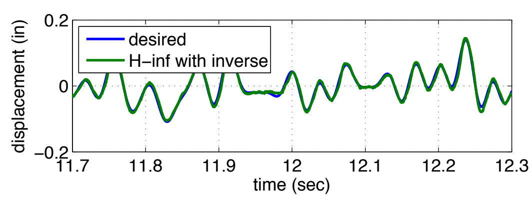
(a) Inverse Controller



(b) FFFB Controller



(c) H_∞ Controller



(d) H_∞ with Inverse Controller

Figure 4.15.: Time Domain Tracking Performance for Different Control Algorithms, MR damper On Condition, Tracking of 0-20 Hz BLWN

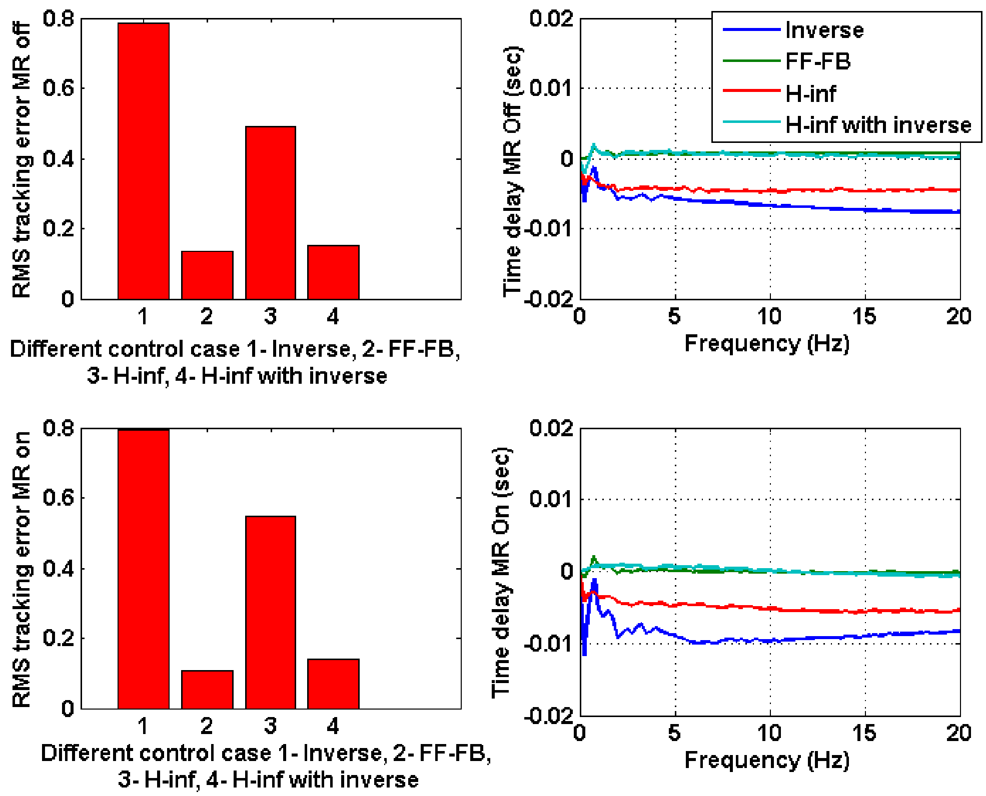


Figure 4.16.: RMS Error and Time Delay between Desired Signal and Measured Signal after Different Controllers, Tracking of 0-20 Hz BLWN, Upper Figure: MR Damper Off Condition, Lower Figure: MR Damper On Condition

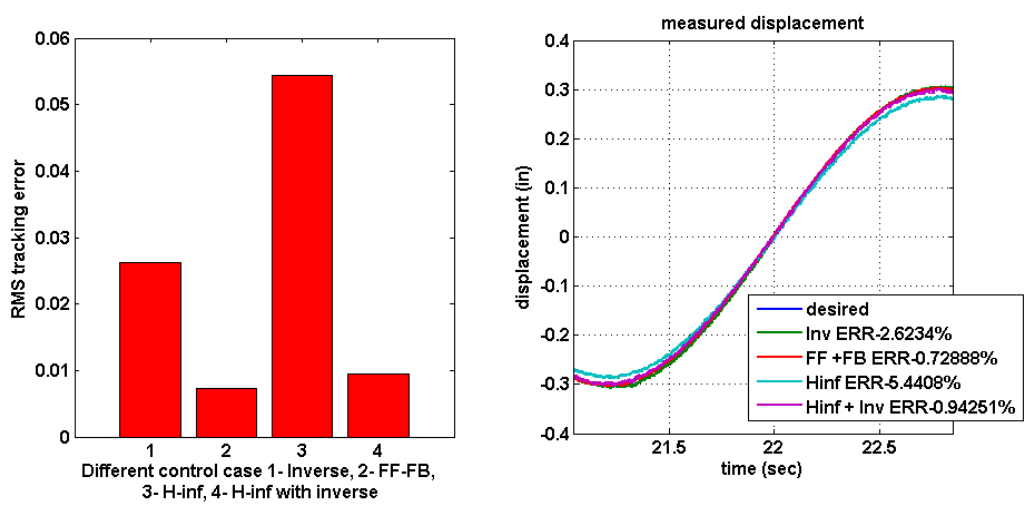


Figure 4.17.: RMS Error and Time Domain Tracking Performance under Different Controllers, MR Damper Off Condition, Tracking of 0.5 Hz Sine Wave

4.4 Control Algorithm Robustness Study

Previously, uncertainties have not been thoroughly studied in developing actuator control algorithms for RTHS. Those uncertainties include systematic uncertainties such as modeling errors, and the random uncertainties as the existence of noise. In this section, controller robustness according to the two types of uncertainties are investigated. Modeling error due to misidentification of specimen properties and modeling error due to system simplification (different transfer function order) are both considered.

4.4.1 Control Robustness to Modeling Error

As discussed in chapter 3, the plant transfer function is changing with specimen mass, stiffness and is changing significantly with different proportional gain. Thus, system misidentification error in this chapter is assumed more on the testing specimen (experimental substructure) rather than in the hydraulic actuator.

Using the example from chapter 3, the attached specimen spring has a stiffness $386lb/in$ and an added mass of $2lb$. The proportional gain of 7000 is defined as a nominal specimen in this section. The other test cases shown in Table 4.2 are considered to have modeling error.

Fig. 4.18 shows the system transfer functions corresponding to each specimen setup, the dash line is the nominal plant. Transfer function plots are sorted into two groups which is due to the proportional gain change from 7000 to 6000.

The control parameters are defined in Table 4.3, assuming the plant model is the nominal case, with plant transfer function:

$$G_R(s) = \frac{4.837 \times 10^9}{s^4 + 904.7s^3 + 4.9126 \times 10^5 s^2 + 7.9909 \times 10^7 s + 5.0123 \times 10^9} \tag{4.36}$$

The results in Fig. 4.19 and 4.20 demonstrate that control performance is not quite sensitive to specimen stiffness and mass change in this case, because the mass

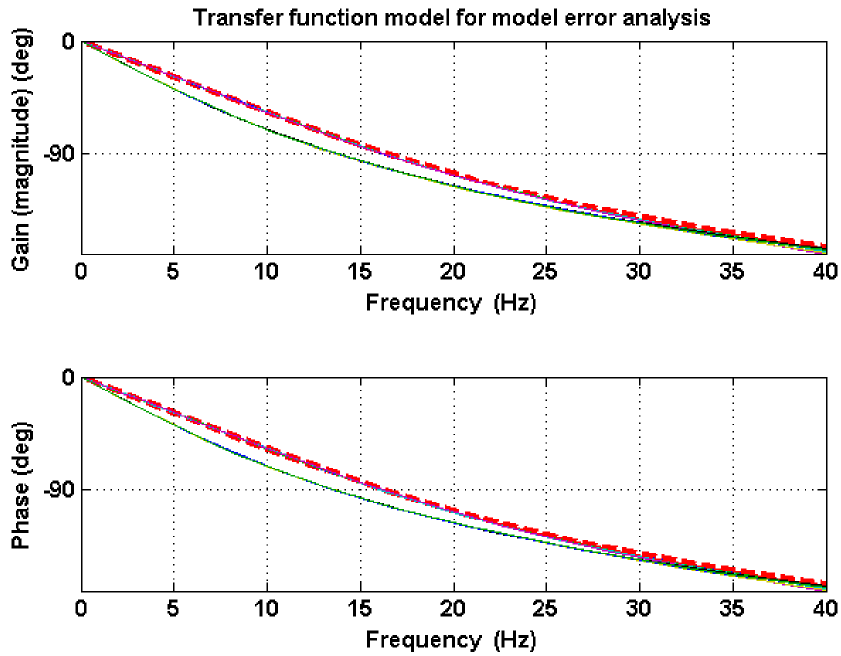
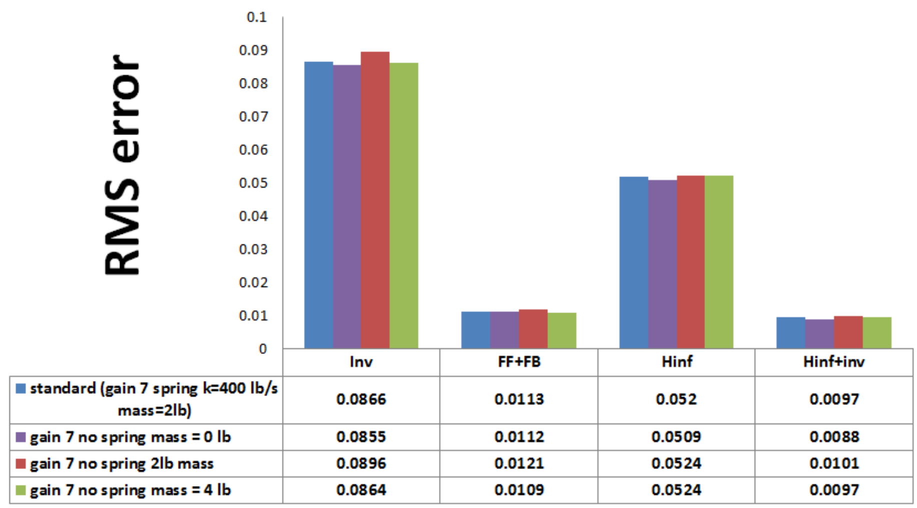


Figure 4.18.: Transfer Function for Systems with Different Parameters, Dash Line Shows the Nominal Model

Table 4.3.: Parameters and Compensation Results for Different Controllers in Nominal System

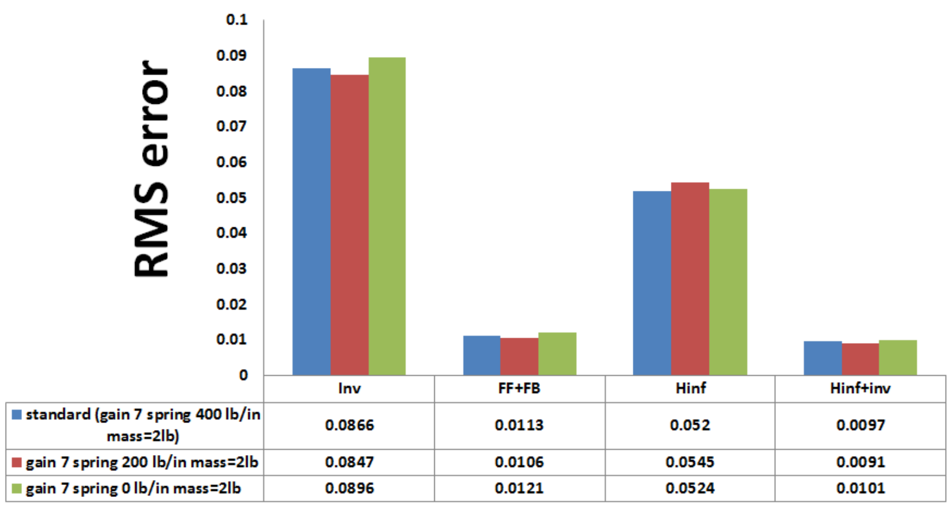
	Parmeter	RMS error
Inverse Compensator	t_d=0.015 sec	0.7392
Feedforward Feedback	$Q/R = 400$	0.0928
H_∞ compensation	$G_d(s) = \frac{4.2578\times10^4}{(s+20)(s+230+7647)}$	0.4752
H_∞ with inverse	$G_d(s) = \frac{4.2578\times10^4}{(s+20)(s+230+7647)}$ T_{dH_∞}=0.0055 sec	0.0790

and stiffness change is relatively small as compared to actuator capacity. This finding is also supported by Fig. 4.18, where only P gain change can noticeably affect the transfer function. From the comparison, both H_∞ with inverse and FFFB controller yield competitive performance with RMS error under 1%. However, in Fig. 4.21, where the P gain is reduced from 7000 to 6000, the original H_∞ control design and H_∞ with inverse controller are affected more, and for the H_∞ with inverse controller, error changes from 0.97% to 1.52%. However, the FFFB controller is not affected by P gain setting quite much which shows high robustness.



	Inv	FF+FB	Hinf	Hinf+inv
standard (gain 7 spring k=400 lb/s mass=2lb)	0.0866	0.0113	0.052	0.0097
gain 7 no spring mass = 0 lb	0.0855	0.0112	0.0509	0.0088
gain 7 no spring 2lb mass	0.0896	0.0121	0.0524	0.0101
gain 7 no spring mass = 4 lb	0.0864	0.0109	0.0524	0.0097

Figure 4.19.: Sensitivity of Tracking Performance for Different Control Algorithms According to Different Specimen Mass



	Inv	FF+FB	Hinf	Hinf+inv
standard (gain 7 spring 400 lb/in mass=2lb)	0.0866	0.0113	0.052	0.0097
gain 7 spring 200 lb/in mass=2lb	0.0847	0.0106	0.0545	0.0091
gain 7 spring 0 lb/in mass=2lb	0.0896	0.0121	0.0524	0.0101

Figure 4.20.: Sensitivity of Tracking Performance for Different Control Algorithms According to Different Specimen Stiffness

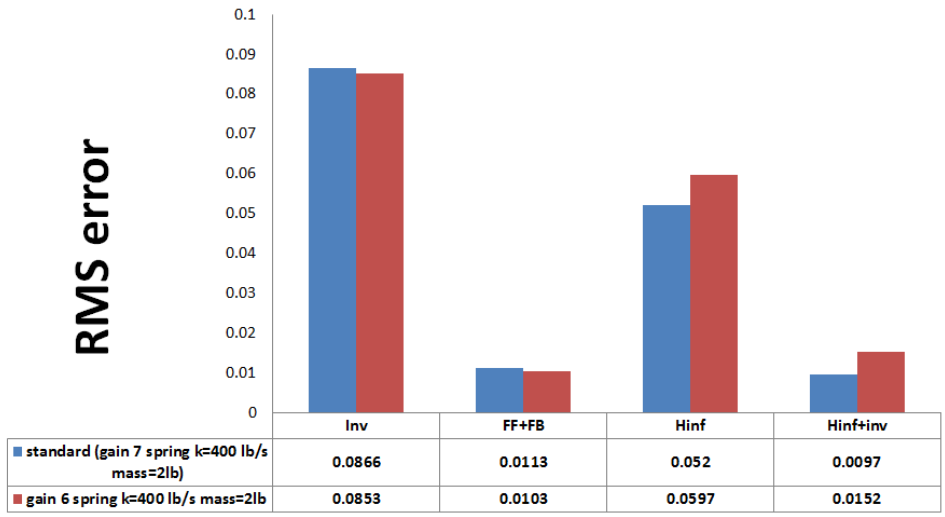


Figure 4.21.: Sensitivity of Tracking Performance for Different Control Algorithms According to Different P Gain

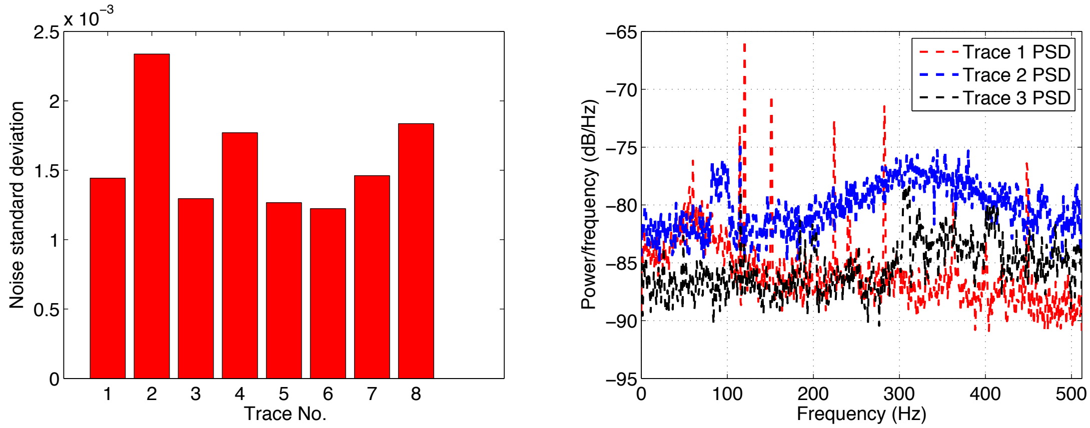
4.4.2 Control Robustness to Random Uncertainties

In this section, a experimental based simulation is conducted, where the influence of different levels of measurement noise are discussed. The controller parameters optimization and control performance under different signal/noise ratios are presented. The plant model in this section is the model identified from the experimental study.

The noise is measured at random testing time from experimental test. Fig. 4.22 demonstrates the power spectrum and standard deviation for different noise traces which is not stationary at all the time. Therefore, such uncertainty in the distribution requires control robustness regarding to noise and disturbance.

Controller aggressiveness, which is represented by the key design parameter of a controller that dominates the performance, in the inverse compensation, is the delay parameter α (sec); in the feed-forward and feed-back compensation, is the Q/R ratio in the LQG feedback loop and in H_∞ control is the static constant G_{dc} in Eq. 4.27.

To understand controller sensitivity to noise, the noise trace 2 (the most severe case with largest standard deviation) is added to plant feedback in the measurement. In simulation, a band limited white noise signal with bandwidth of 10 Hz is im-



(a) Standard Deviation (b) Power Spectrum Density

Figure 4.22.: Characteristics of Different Noise Traces

plemented as input signal, different signal/noise ratio is investigated while the key design parameters in each controller varies. The signal/noise ratio is defined as RMS amplitude of input signal to the noise standard deviation.

The controller sensitivity to noise is quantified by the level of noise in the measurement. First the measurement signal went through a low pass filter with pass band up to 30Hz and the cutoff frequency is set at 50Hz. Noise content in measurement is considered as the residue between filtered signal and unfiltered signal with signal synchronization. Fig. 4.23, 4.24, and 4.25 illustrate the noise content excited under different level of controller agressiveness for inverse compensator, Feedforward feedback compensator and H_∞ compensator respectively.

From inverse compensation, for smaller signal/noise ratio, noise content in measurement does not vary with design parameter α, however, when signal/noise ratio increases, the effect of α increase leads to the increase of noise content in output measurement. In FF-FB compensation, this trend can be observed through all signal/noise ratio cases and similarly to H_∞ compensation algorithm. In H_{infty} compensation in Fig. 4.25, when the aggressiveness of the compensator increases, the closed loop system may go unstable due to the severe amplification of noise in the measurement which indicates the high sensitivity to noise.

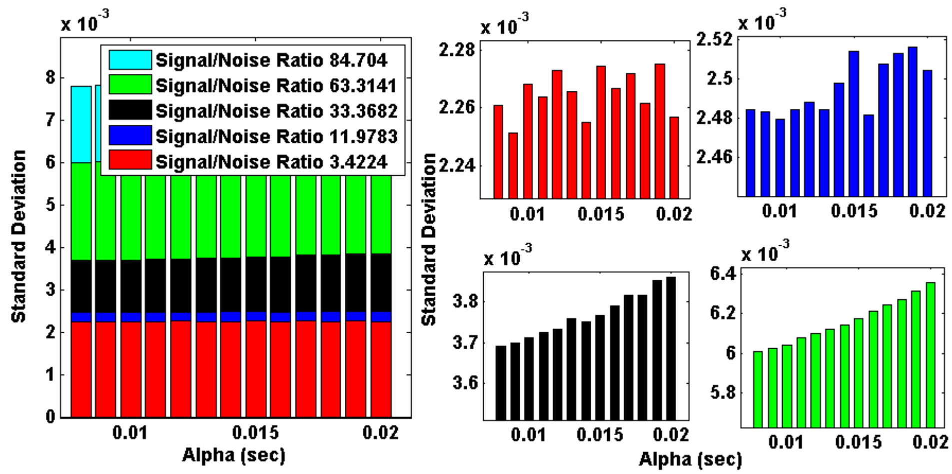


Figure 4.23.: Noise Amplification with Different Controller Aggressive Level, Inverse Controller

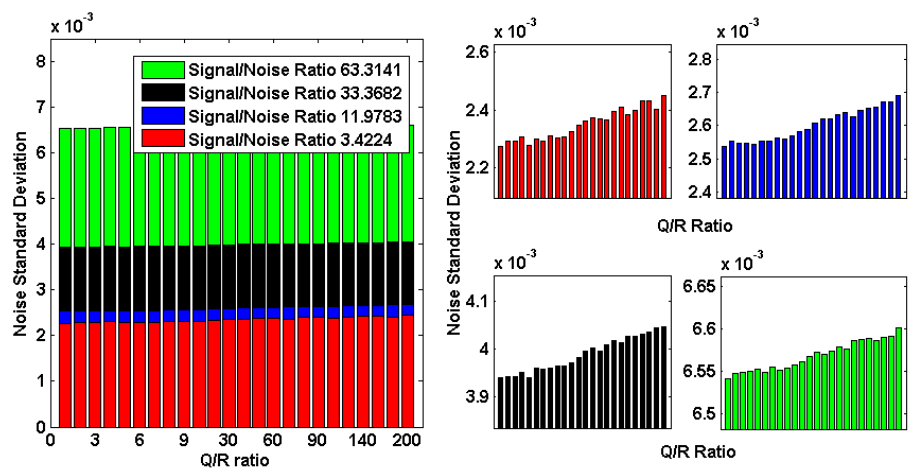


Figure 4.24.: Noise Amplification with Different Controller Aggressive Level, FFFB Controller

In tracking control, more aggressive controller has better performance such as smaller time delay, smaller RMS error between desired tracking signal and measured signal shown in Fig. 4.26. However, to further study the aggressiveness of the feedback controller design under noise environment, RMS error between desired and measured signal is calculated to study the compensation efficiency.

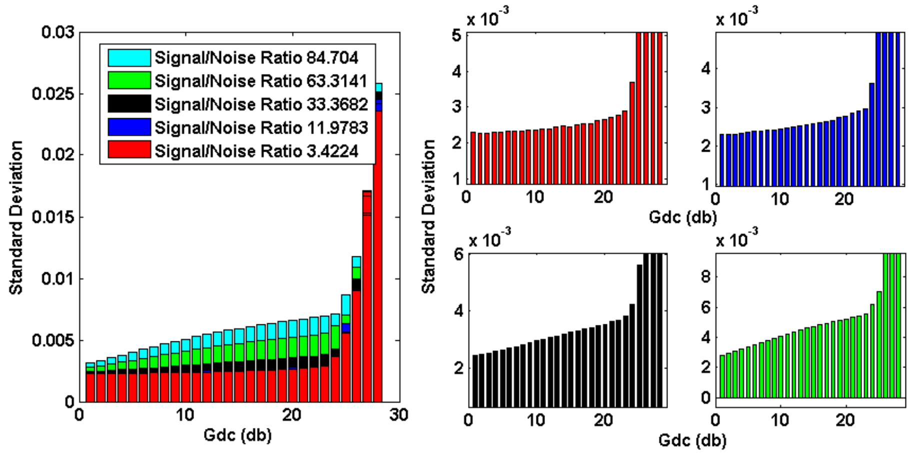


Figure 4.25.: Noise Amplification with Different Controller Aggressive Level, H_∞ Controller

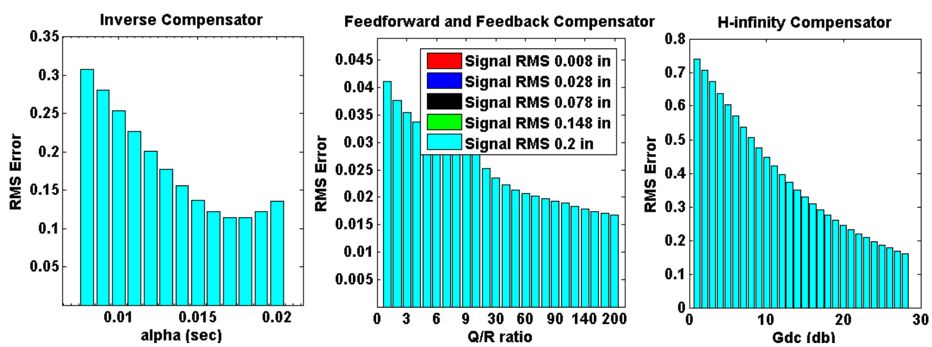


Figure 4.26.: Tracking RMS between Desired and Measured Signal with Different Controller Aggressive Levels, Left Figure: Inverse Controller, Middle Figure: FFFB Controller, Right Figure: H_∞ Controller

Simulation results indicate the optimization of inverse control algorithm is not affected by noise or the signal/noise level, trend for time delay constant α is the same as without noise. It is because that inverse compensation is a feed forward only compensation, noise in the feedback does not affect the design of controller.

For FFFB compensation in Fig. 4.28, noise is amplified by controller and the effect of noise amplification overcomes the improvement due to increase in controller aggressiveness. For smaller signal/noise ratio, the optimized Q/R ratio is 9, this limitation is increasing when signal/noise ratio increases.

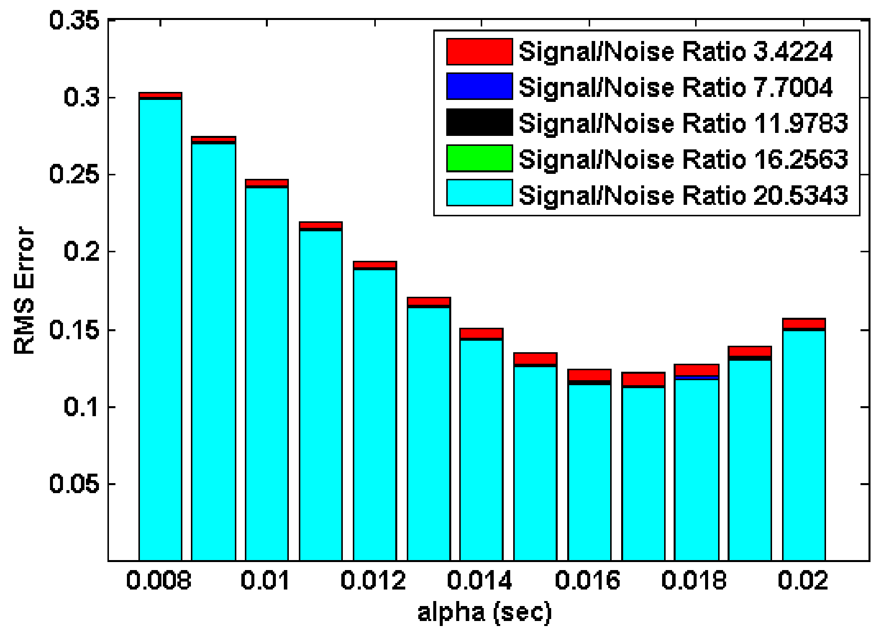


Figure 4.27.: Tracking RMS between Desired and Measured Signal under Different Controller Aggressive Level, Inverse Controller

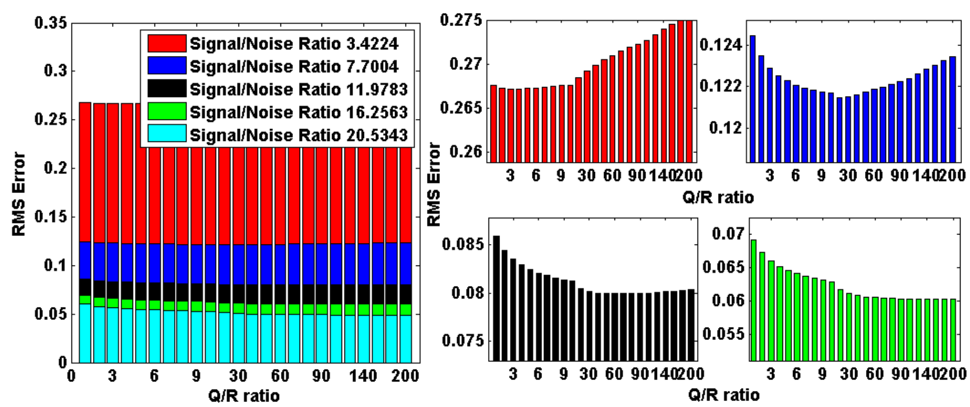


Figure 4.28.: Tracking RMS between Desired and Measured Signal with Different Controller Aggressive Levels, FFFB Controller

Such trend shows clearly in H_∞ controller design. As compare to Fig. 4.26 where control performance is constantly improved by increase static design gain G_{dc}, Fig. 4.29 indicates in a noisy environment, noise is largely amplified and even drives the system to unstable. Severe observation is found even when signal/noise ratio is small, demonstrating noise in the feedback can dominate the tracking performance.

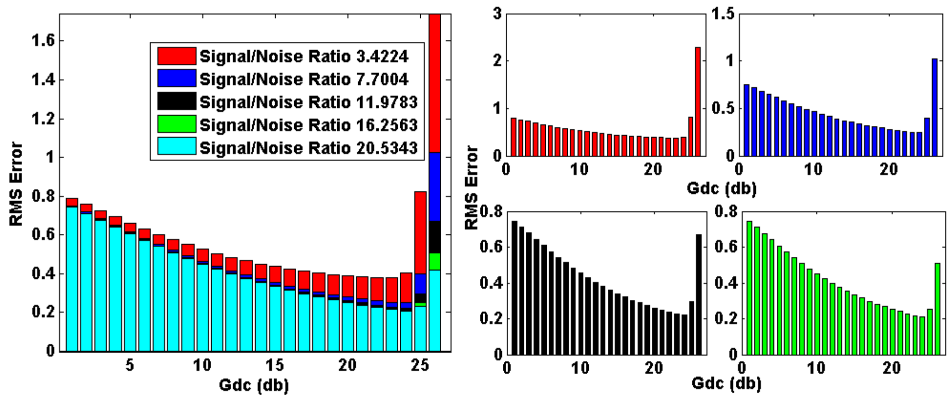


Figure 4.29.: Tracking RMS between Desired and Measured Signal with Different Controller Aggressive Levels, H_∞ Controller

4.5 Conclusion

This chapter presents three commonly used actuator control algorithms and discusses the controller performance to each control parameter. For the inverse control algorithm, when time delay associated parameter α increases, residual time lag between measured plant output to command signal decreases. However, at large α, the tracking accuracy is reduced due to the signal overshoot. Based on the simulation, inverse controller works better for system with small delay. Also from the experimental study, inverse controller is found to be more effective for a narrow band input.

In the FFFB controller, high Q/R ratio increases the tracking accuracy. This control algorithm is easy to design by giving a third order transfer function information. However, it is found for tracking command of larger amplitude, this algorithm generates a large output in voltage output reaches the limit of physical facility. Also the experimental study shows the FFFB controller is robust to plant modeling error.

The H_∞ controller is the most flexible control algorithm among the three, the order of loop shaping function is configurable as well as the constant gain G_{dc} and loop shaping function pole locations. From simulation, it is found that the constant gain G_{dc} is the dominant parameter for control accuracy. Larger pole value, increases the tracking accuracy, however, noise effect from measurement can be amplified. Again, that is the trade-off between accuracy and robustness for H_∞ controller design.

From the random uncertainty study, noise in measurement is always amplified by the increase of controller aggressiveness in feedback control algorithms. This observation illustrates that noise effect in the system can limit the performance of those controllers. The optimal design of such controller changes with different signal/noise ratio. Therefore, the noise content in the sensor measurement should be carefully investigated in designing a feedback control algorithm. This is further discussed in chapter 5.

5. ROBUST INTEGRATED ACTUATOR CONTROL ALGORITHM

For over two decades, hybrid simulation was always performed on an extended time scale, neglecting the effects of rate dependent behaviors [83], [84]. Several promising load rate dependent auxiliary devices have been developed recently. These developments, combined with recent innovations in embedded systems and real-time operating systems, have enabled the earthquake engineering community to embrace this new technology [85]. Therefore, Executing a hybrid simulation in real time scale is necessary and possible.

Even though it is technically possible to implement real time hybrid simulation (RTHS), challenges certainly exist. During RTHS, data acquisition, numerical integration and experimental application of the numerical response through a loading (transfer) system is restricted within a small time frame (0.5 to 1 ms). To execute the numerical computations and experimental actions simultaneously, high precision tracking control of the transfer system is required. However, tracking performance is normally limited by time delays and time lags introduced by communication, A/D D/A conversion, dynamics of control device, control structural interaction (CSI) [74] and noise in the feedback loop. Those limitations require individual examination and careful compensation.

In 2011, Gao *et al* proposed a robust actuator controller for RTHS based on H_∞ theory which, for the first time, considered noise, disturbance and model uncertainty effects in actuator control [42], [43], [44]. This H_∞ controller is designed by trading off tracking performance and control robustness, giving the user the power to meet the testing needs. However, as indicated in chapter 4, when the noise/signal ratio in the system is high within the frequency range of interest, the sensor noise may be amplified, possibly resulting in a deterioration in performance.

The robust integrated actuator control algorithm proposed herein integrates the most effective features of the methods discussed. To reduce the noise impact, the linear quadratic estimation (LQE) scheme is included. For greater flexibility and higher accuracy during testing, an additional feedforward block minimizes any small residual time delay/lag. Further, it is proved analytically that RIAC has the same stability characteristics as the H_∞ algorithm. A step-by-step design and implementation procedure for RIAC is suggested based on the needs of a particular user. RIAC is implemented on two experimental setups, and both simulation and experimental studies demonstrate the efficacy of the RIAC. RIAC is found to significantly reduce the impact of noise as compared to an H_∞ design, and to improve the phase response effectively. To further validate the RIAC method, RTHS is performed using a three story steel building as the numerical substructure and an MR damper as the physical specimen. Two very different experimental setups are used to demonstrate the versatility of RIAC. The results indicate that the RIAC is a very effective and accurate method for imposing boundary condition in RTHS.

5.1 RTHS Formulation

In RTHS, a structure system is divided into numerical and experimental substructures. The numerical substructure contains the well-understood components and leaves the hard-to-model components in the physical setup. An illustration of a hybrid simulation is shown in Fig. 5.1. A three story building equipped with damping device is separated into a 3 DOF shear model in the numerical portion and the damping device is physically tested in the lab by attaching it to an actuator.

The interaction between experimental substructure and numerical substructure through actuator control is shown in Fig. 5.2. Such interaction has different layers: i) an inner loop PID control is used in most manufacturer provided software that

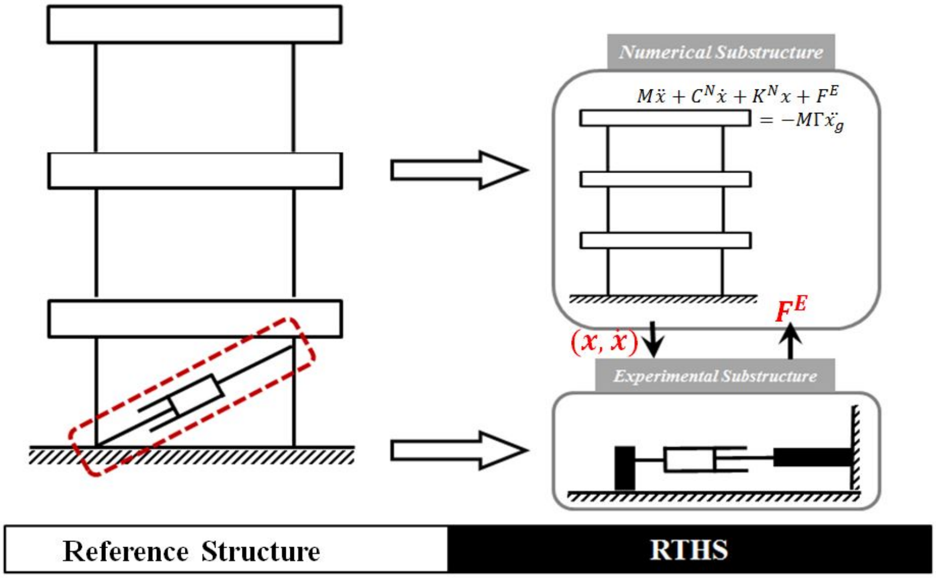


Figure 5.1.: RTHS System Concept

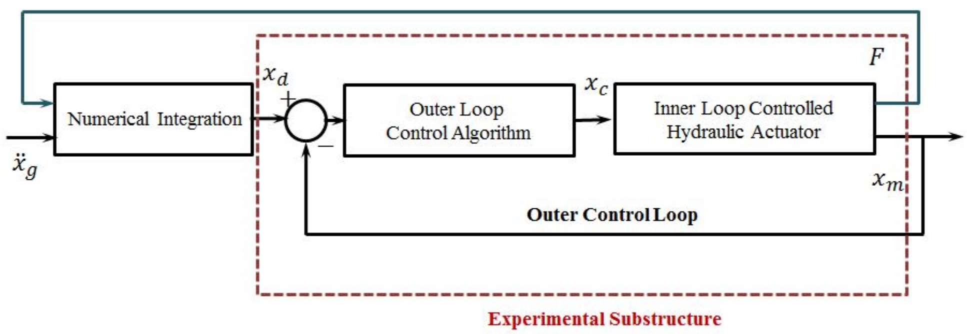


Figure 5.2.: Numerical-Experimental Interaction in RTHS

stabilized the actuator; ii) the outer loop control algorithm grantees desired response from numerical codes is implemented appropriately; and, iii) the force feedback loop between experimental substructure and numerical substructure.

The equation of motion for the numerical substructure is:

$$M^N\ddot{x} + C^N\dot{x} + K^N x + F^E(x, \dot{x}) = -M\Gamma\ddot{x}_g \tag{5.1}$$

where M^N, C^N, K^N indicates mass, damping and stiffness matrices in the numerical substructure. F^E is the measured force from the experimental substructure and $\ddot{x}_g$ is the earthquake acceleration record.

A typical RTHS implementation is as follows:

1. For initial time step, $i = 1$:
   Calculate initial numerical response from Eq 5.1 and get x_1, $\dot{x_1}$ given $x_{g,1}$ and ${F_1}^E = [0]$.
   Set $x_{d,1} = x_1$.

2. For time step i, $(i > 1)$:
   Impose desired response $x_{d,i-1}$ through outer loop control algorithm to transfer system (hydraulic actuator) with command signal $x_{c,i}$

3. Actuator executes command signal, achieves real response $x_{m,i}$ to attached specimen, then measure experimental restoring force ${F_i}^E$ due to actuator motion.

4. Calculate numerical response $x_i, \dot{x}_i$ using integration scheme with given $x_{g,i+1}$ and ${F_i}^E$
   Set $x_{d,i} = x_i$.

5. Set $i = i + 1$, go to step 2.

The accuracy between desired boundary condition $[x_d, \dot{x}_d]$ with the transfer system in step 2 and 3 dominates the fidelity of the RTHS. Here, the outer loop control algorithm is designed based on the assumption that the plant (the actuator model) contains both the hydraulic actuator and the internal PID loop together, and can be considered as a linear time invariant system.

5.2 Actuator Dynamic Model

As stated in chapter 3, the proportional gain controlled hydraulic actuator transfer function can be linearized between output displacement x_m and command x_d. The system can be represented in block diagram as shown in Fig. 5.3 [70]. Several important assumptions have been made in this mathematical model: (1) fluid properties are constant; (2) servo-valves are not saturated; (3) supply pressure is much greater than the load pressure; (4) friction force can be modeled as viscous damping; and (5) main stage spool opening is proportional to pilot stage flow. These assumptions are typically acceptable due to the relatively low frequency and small amplitude nature of RTHS.

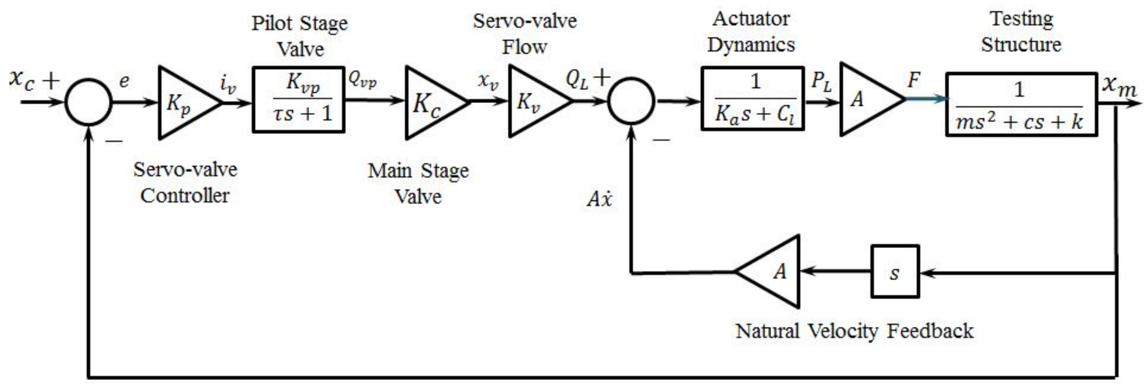


Figure 5.3.: Hydraulic Actuator with Inner Control Loop

Table 5.1.: Servo-Hydraulic System Parameters

k_v	Valve Flow Gain	K_c	Valve Pressure Gain
τ	Servo-valve Time Delay Constant	C_l	Piston leakage coefficient
A	Piston Area	V_t	Fluid Volume
K_p	Internal Controller Proportional Gain	β_e	Effective Bulk Modulus
K_{vp}	Pilot Stage Valve Flow Gain	m	System and Piston Mass
c	System and Actuator Damping	k	System and Actuator Stiffness

The linearized hydraulic system is derived as a fourth order transfer function directly from the block diagram Fig. 5.3:

$$G_{x_m,x_d} = \frac{X_m(s)}{X_d(s)} = \frac{Z_0}{P_1 s^4 + P_2 s^3 + P_3 s^2 + P_4 s + P_5} \tag{5.2}$$

where,

$$Z_0 = K_p K_{vp} K_c K_v A \tag{5.3}$$

$$K_a = V_t / 4\beta_e \tag{5.4}$$

$$P_1 = K_a \tau m \tag{5.5}$$

$$P_2 = K_a \tau c + \tau C_l m + K_a m \tag{5.6}$$

$$P_3 = K_a \tau k + \tau C_l c + K_a c + A^2 \tau + m C_l \tag{5.7}$$

$$P_4 = \tau C_l k + K_a k + A^2 + C_l c \tag{5.8}$$

$$P_5 = Z_0 K_p + C_l k \tag{5.9}$$

Parameters $Z_0, P_1, P_2, P_3, P_4, P_5$ can be identified through complex function curve fitting and will be fully addressed in Section 5.4.

5.2.1 Noise Sources in the Hydraulic System

As stated in chapter 4, noise in the actuator control system can limit the performance of control algorithms. Noise in the hydraulic actuation system normally comes from two sources. One is the mechanical and fluid vibration in hydraulic actuation components and the other is the electrical noise due to power sources, ground loops, etc. Table 5.2 listed all common sources of hydraulic actuation noise. More detailed information is discussed in [88] and [89]. Those noise sources are very hard to distinguish and defeat individually during testing. To achieve control robustness, impact of noise should always be carefully studied [90].

Table 5.2.: Noise Sources in Hydraulic Actuation System

Mechanical Vibration	**Fluid Vibration**	**Electrical Noise**
Structural impact and Friction	Hydraulic impact	Power line disturbances
Rotary imbalance	Hydraulic fluid pump	Externally conducted noise
Hydraulic valve and cylinder	Cavitation induced vibration	Transmitted noise
Pipeline and Tank resonance	Turbulent flow and vortex	Ground loops

5.3 RIAC Algorithm

The Robust Integrated Actuator Control strategy uses H_∞ optimization to design the core controller to meet the needs of the user. H_∞ allows for a trade-off between system performance and its robustness. To further improve control performance for RTHS applications, a Linear Quadratic Estimator (LQE) is used to reduce measurement uncertainty. In the final setup, one feed forward block is used in such a way that it does not affect the stability of the feedback system, but does enhance tracking performance. The entire system is shown in Fig. 5.4. The H_∞ block is designed in the continuous Laplace domain, and the LQE and feedforward blocks are designed in the discrete z domain.

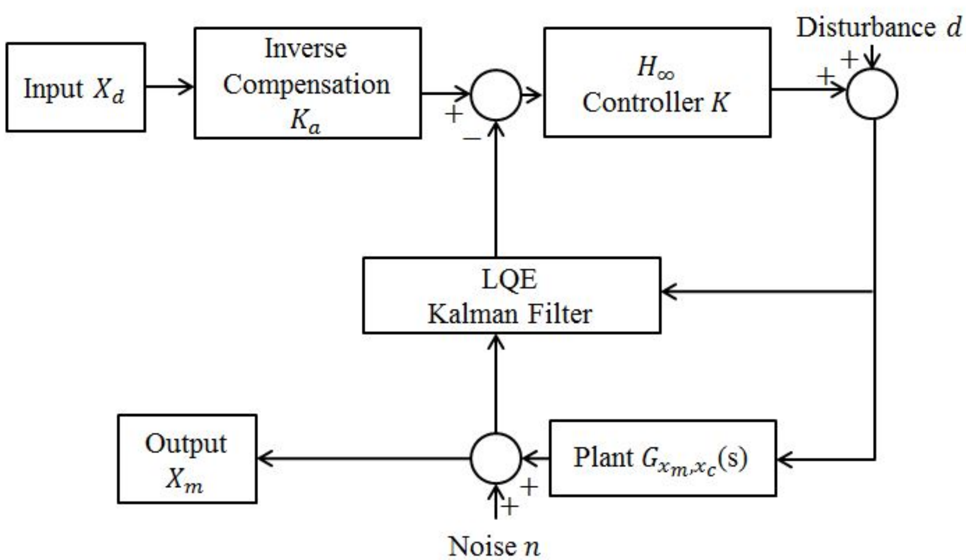


Figure 5.4.: Robusted Integrated Actuator Control (RIAC) Block Diagram

5.3.1 H_∞ontrol Algorithm

Gao *et al* [42], [43], [44] first introduced robust H_∞ control into actuator control for RTHS. The feedback block diagram with the H_∞ algorithm is shown in Fig. 5.5.

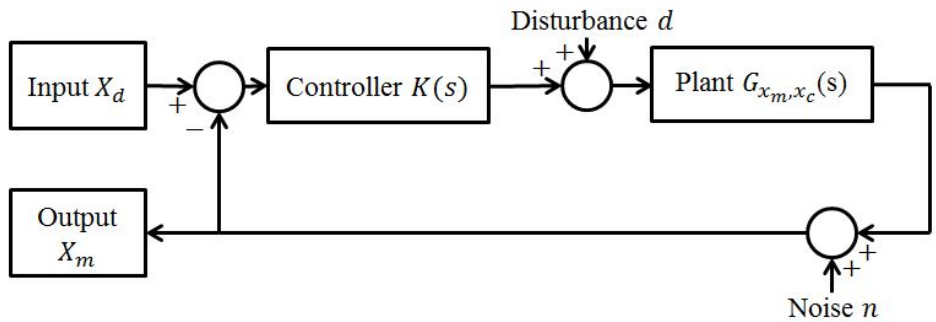


Figure 5.5.: H_∞ Feedback Control Block Diagram

For a typical feedback control system, the sensitivity function S_0 and complementary sensitivity function T_0 (same as I/O transfer function) are given as

$$S_0(s) = (1 + G_p(s)K(s))^{-1} \tag{5.10}$$

$$T_0(s) = 1 - S_0(s) = \frac{G_P(s)K(s)}{1 + G_P(s)K(s)} \tag{5.11}$$

$$x_m(t) = T_0(x_d(t) - n(t)) + S_0 G_P d(t) \tag{5.12}$$

where, $n(t)$ is system measurement noise and $d(t)$ is system processing disturbance. $s = j\omega$ indicates as equation in frequency domain (written in upper case), time domain functions in terms of t are written in lower case.

For the RTHS implementation, the input tracking signal $x_d(t)$ should be imposed on the actuator accurately. Thus, the I/O transfer function $T_0(s)$ should be close to unity over the relevant control frequency (low frequency), and close to 0 at high frequency where noise/disturbance signal is dominant. An desired open loop transfer function $W_0(s)$ should be designed according to such requirements. H_∞ algorithm is used to design the optimal controller $K(s)$ that assures the system open loop transfer function GK closely meets the target function $W_0(s)$.

$W_0(s)$ can be written in state space form $W_0(s) = ss[A_w, B_w, C_w, D_w]$. Controller $K(s)$ is defined as $K(s) = G_c G_d$, where direct inverse compensation $G_c(s)$ is designed as $G_c(s) = W_0(s)/G_{x_m,x_d}$. And $G_d(s)$ is obtained from H_∞ optimization.

$$\begin{aligned}(A_w - B_w P^{-1} D_w^* C_w)^* X + X(A_w - B_w P^{-1} D_w^* C_w)\\ -X B_w P^{-1} B_w^* X C_w^* (I - D_w^* P^{-1} D_w^*) C_w = 0\end{aligned} \tag{5.13}$$

$$\begin{aligned}(A_w - B_w T^{-1} D_w^* C_w) Z + Z(A_w - B_w T^{-1} D_w^* C_w)^*\\ -Z C_w T^{-1} C_w^* Z B_w (I - D_w^* T^{-1} D_w^*) B_w^* = 0\end{aligned} \tag{5.14}$$

$$P = I + D_w^* D_w, T = I + D_w D_w^* \tag{5.15}$$

Thus, X and Z can be obtained by solving two generalized algebraic Riccati equations, Eqs. 5.13 and 5.14, where $(^*)$ denotes the complex conjugate transpose of one matrix.

To obtain G_d, H_∞ optimization:

$$\left\| \begin{bmatrix} G_d(I - W_0 G_d)^{-1} \tilde{L}^{-1} \\ (I - W_0 G_d)^{-1} \tilde{L}^{-1} \end{bmatrix} \right\|_\infty \leq \gamma \tag{5.16}$$

$$\tilde{L} = T^{-1/2} + T^{-1/2} C_w (sI - A_w - U C_w)^{-1} U \tag{5.17}$$

where

$$U = -(Z C_w^* + B_w D_w^*) T^{-1} \tag{5.18}$$

The constructed controller G_d in state space form is:

$$\begin{bmatrix} A_{G_d} \\ B_{G_d} \\ C_{G_d} \\ D_{G_d} \end{bmatrix} = \begin{bmatrix} A_w - B_w V + \gamma^2 W_1^{*-1} Z C_w^* (C_w - D_w V) \\ \gamma^2 W_1^{*-1} Z C_w^* \\ B_w^* X \\ -D_w^* \end{bmatrix} \tag{5.19}$$

where, $W_1^* = I + (XZ - \gamma^2 I)$ and $V = P^{-1}(D_w^* C_w + B_w^* X)$.

The performance of H_∞ compensation largely depends the design of W_0. Fig. 5.6 shows three different H_∞ optimization results and their corresponding performance under $W_{0,A,B,C}$ for plant G, is described by:

$$G = \frac{1.422 \times 10^8}{s^4 + 282.875s^3 + 6.2817 \times 10^4 s^2 + 5.7227 \times 10^6 s + 1.4128 \times 10^8} \tag{5.20}$$

It is illustrated that design A is the most aggressive controller among the three and tracking can perform up to 200 Hz. However, high frequency noise attenuates only slightly even up to 1 kHz. Alternatively, design C is effective in depressing noise influence in tracking, but the acceptable tracking performance is only acceptable to 30 Hz. Thus, it is clear that there is a trade-off between performance and sensitivity. In a real world experiment, those limitations can make it impossible to perform a successful test.

5.3.2 Linear Quadratic Estimation

Control accuracy is compromised when noise content in the feedback measurement is high. This phenomenon is more difficult to tackle using H_∞ optimization when the noise frequency is at or close to the frequency of the control signal. To reduce noise in the actuator displacement measurement, the Linear Quadratic Estimator (LQE), also known as Kalman filter method is implemented to estimate the actuator displacement [91]. A discrete actuator system with processing disturbance $d(t)$ and measurement noise $n(t)$ is written in discrete state space form:

$$x_P(k+1) = A_d x_P(k) + B_d u(k) + d(k) \tag{5.21}$$

$$y_p(k+1) = C_d x_P(k) + n(k) \tag{5.22}$$

where, system state space matrices A_d, B_d, C_d, are converted to discrete state space from actuator model described in Eq. 5.2, $[x_p(k)]$ is the plant state vector and $y_p(k)$ is direct measured displacement from the actuator's internal LVDT.

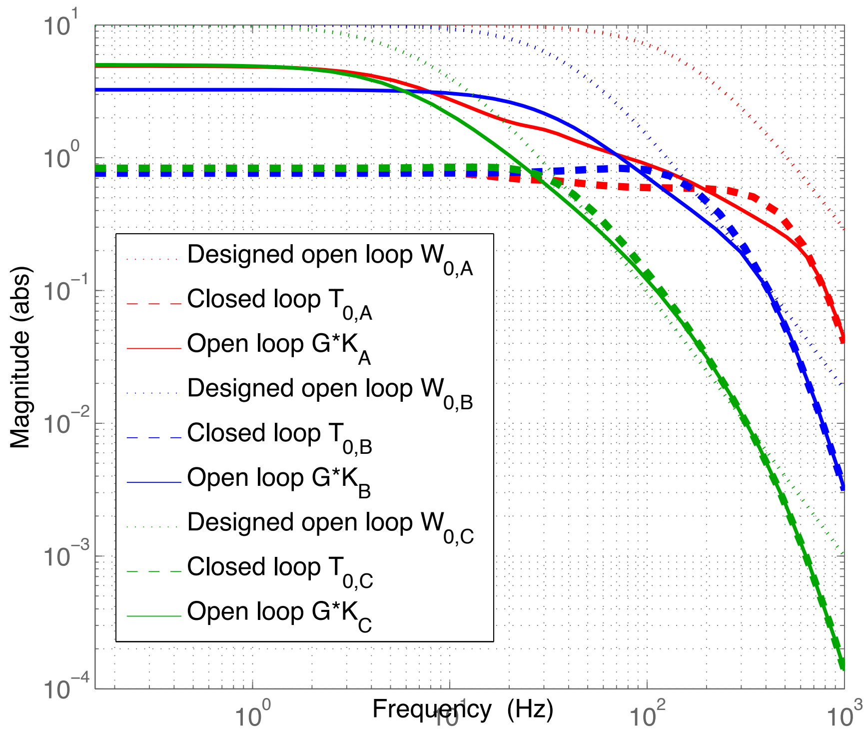


Figure 5.6.: H_∞ Control Performance under Different Open Loop W_0 Design

For each time step k, the Linear Quadratic Estimation (LQE), is formulated as:

(i) Time update:

$$\hat{x}_P^-(k+1) = A_d x_P(k) + B_d u(k) \tag{5.23}$$

$$P_K^-(k+1) = A_d P_K^-(k) A_d^T + Q \tag{5.24}$$

(ii) Measurement update:

$$K_k(k+1) = P_K^-(k+1)(C_d P_K^-(k+1)C^T + R)^{-1} \quad (5.25)$$

$$\hat{x}_P(k+1) = \hat{x}_P^-(k+1) + K_k(k+1)(y_P(k) - C_d\hat{x}_P^-(k+1)) \quad (5.26)$$

$$P_K(k+1) = (I - K_k(K+1)C_d)P_K^-(k+1) \quad (5.27)$$

$$\hat{y}_P(k+1) = C_d\hat{x}_P(k+1) \quad (5.28)$$

where P_K is the Kalman filter error covariance matrix, K_k is the Kalman filter gain, Q is the predefined processing disturbance matrix and R is the predefined measurement noise. Q/R ratio determines the estimator weights on plant output and system measurement.

(iii) Set $k = k + 1$, go to step (i).

5.3.3 Inverse Compensation

Because actuator delay/lag is critical for RTHS, when the small residual delay/lag is found in experimental implementation after H_∞ controller, an additional block dedicated to small time delay/lag is integrated in RIAC. The inverse compensation algorithm proposed by Chen [35], where the system delay/lag is assumed to be constant for the entire frequency range. The compensated system after H_∞ is modeled as a first order system:

$$G_a(z) = \frac{z}{\alpha \cdot z - (\alpha - 1)} \quad (5.29)$$

The open loop inverse compensation is the direct inverse of $G_d(z)$:

$$K_a(z) = \frac{\alpha \cdot z - (\alpha - 1)}{z} \quad (5.30)$$

5.3.4 Stability analysis

From Fig. 5.4, it is clear that the feedforward block does not interact with H_∞ feedback loop, and the system after loop shaping control can be written in a discrete transfer function as:

$$G_{ss} = K_a \times \frac{K(z)G_{x_m,x_c}(z)}{1 + K(z)G_{x_m,x_c}(z)} \tag{5.31}$$

$$= \frac{\alpha \cdot z - (\alpha - 1)}{z} \times \frac{K(z)G_{x_m,x_c}(z)}{1 + K(z)G_{x_m,x_c}(z)} \tag{5.32}$$

From Eq. 5.32, G_{ss} is determined by H_∞ feedback control design and is irrelevant to the α value in feed forward design. Since the H_∞ controller is designed based on fitting the desired open loop target function W_0 of the feedback system, closed loop stability is guaranteed from section 5.3.1. The pole locations for the RIAC control system Fig. 5.5 and the H_∞ control system Fig. 5.6 stay the same and are inside of the unit circle Fig. 5.7, indicating that RIAC maintains the stability characteristics of the H_∞ design. In this example, G_{ss} is designed $W_{0,A}$ in previous loop shaping illustration case and it is assumed that the estimator does not affect the system characteristics. Since this proof is general and irrelevant to actuator model G or W_0 choice, detail information of G and $W_{0,A}$ is not presented.

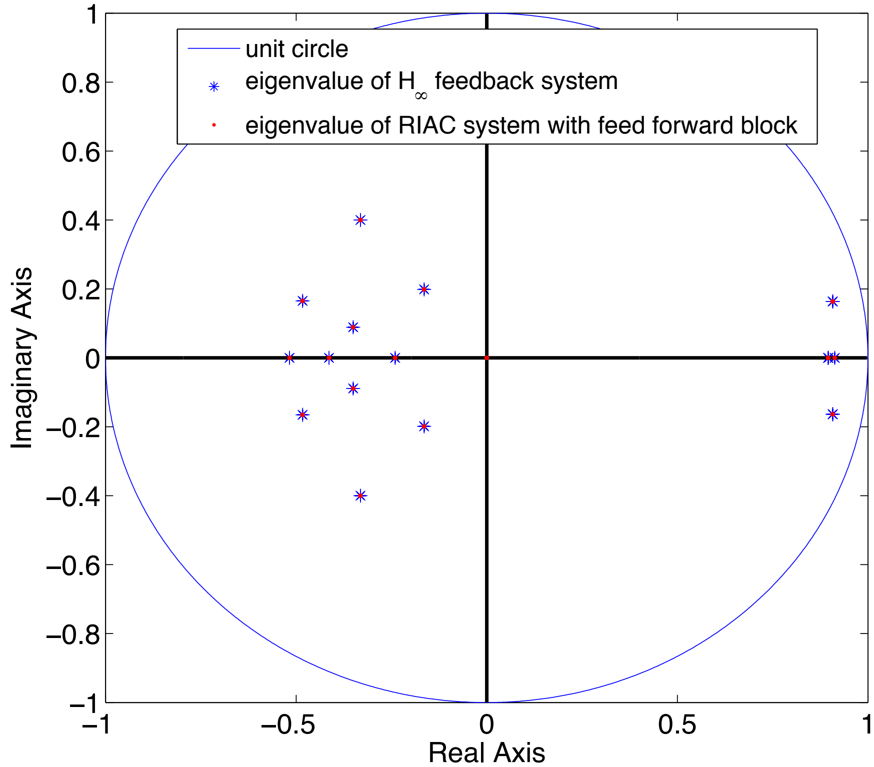


Figure 5.7.: Pole Positions for RIAC System and H_∞ System under the Same W_0

5.4 Experimental Verification

The design procedure of RIAC is divided into three main stages (Fig. 5.8): system identification, controller design and experimental tuning. Here two experimental setups are used to demonstrate the procedure and performance of the controller: Setup A. one large capacity but relatively slow actuator of 2500 kN, and Setup B. a small scale fast actuator. RTHS is performed using a three story steel building as the numerical substructure and an MR damper as the physical specimen. Since the capacity of the actuator in Setup A is significantly larger than the MR damper maximum force, the nonlinearity of the MR damper is negligible to the hydraulic system during testing. In Setup B, an MR damper nonlinearity should also be considered as part of the actuator dynamics during control.

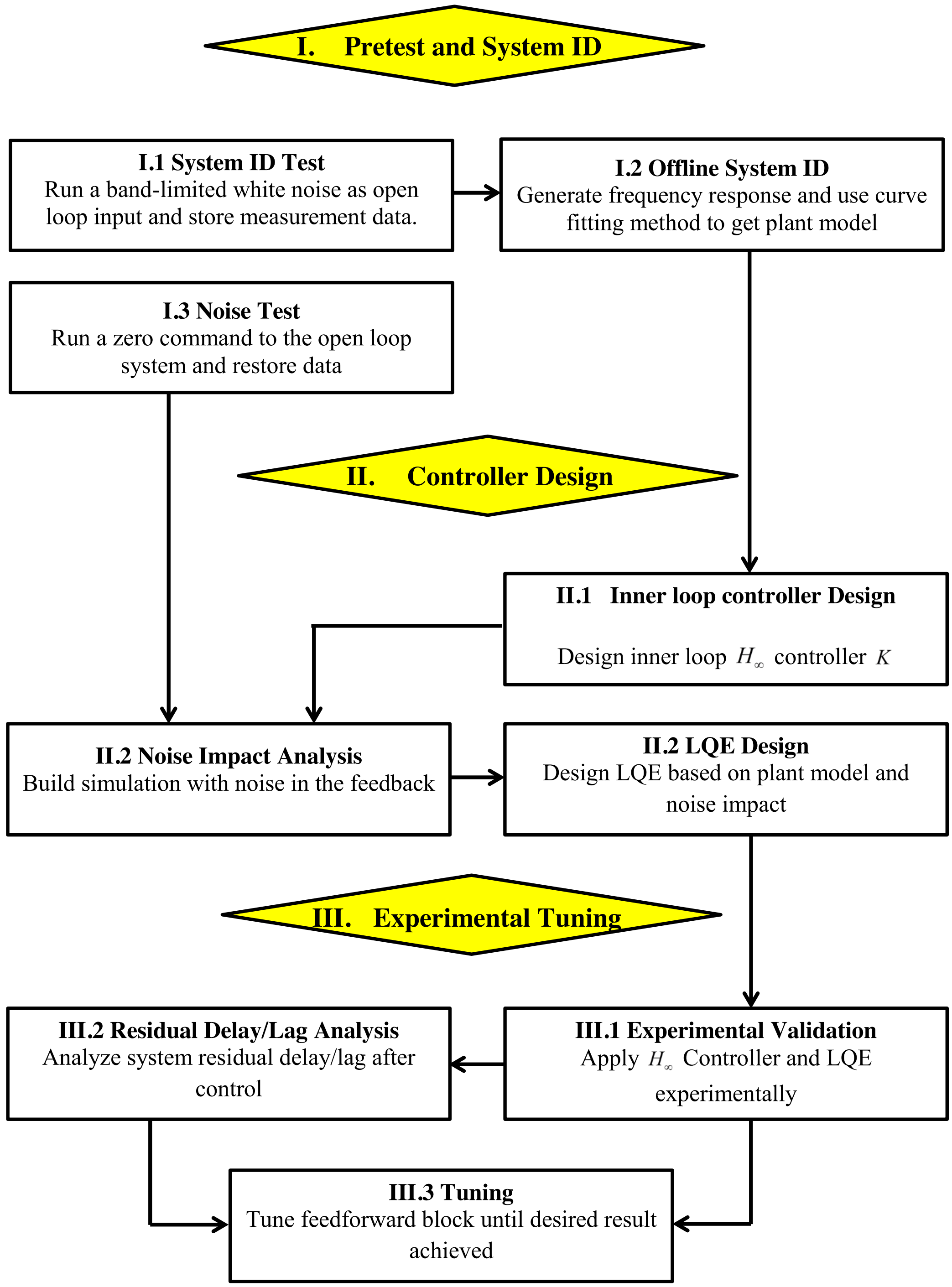


Figure 5.8.: Flow Chart for Implementing RIAC Design and Validation Test

5.4.1 Control Validation on Setup A

Test setup A, the loading system shown in Fig. 5.9 is located in School of Civil Engineering, Harbin Institute Technology (HIT), China and its maximum loading capacity is over 2500 kN. The loading system is constrained in the vertical direction and the MTS Flex GT (Model 793.00) system software controls the actuator through an inner PID loop shown in Fig. 5.3. It supports up to 8 servo-valves and the internal LVDT in each actuator can be measured at a maximum rate of 6000Hz using 16 bit resolution. The hydraulic system is also equipped with two accumulators which supply flow to reach larger the short term velocities, when needed. The saturation velocity limit in this setup is around 90 mm/s.

An outer loop control and external command is applied through MATLAB® compatible real time interface hardware dSpace 1104 (SN. 127174). This system is also used as the DAQ system which supports five A/D, D/A channels (one at 16 bit and four at 12 bit resolution) to be sampled simultaneously. The sampling frequency in this test is 1024 Hz.

The magneto-rheological (MR) damper is made by LORD company (RD-1055-3). The MR damper is operated with external excitation current 0.0 and 1.0 Amps for passive off and passive on condition with a maximum force capacity of 2.5 kN at 1 Amps.

In Setup A, the loading capacity of the actuator is very large compared to the MR damper maximum force. The hydraulic system is identified without MR damper attached using a 0-100Hz BLWN input signal. The time domain and frequency domain response is shown in Fig. 5.10 and 5.11, respectively. The fitted system frequency response is also found in Fig. 5.11, and the plant can be written as a fourth order transfer function given by:

$$G_{x_m,x_c,A} = \frac{1.5 \times 10^8}{s^4 + 281s^3 + 6.6 \times 10^4 s^2 + 6.0 \times 10^6 s + 1.5 \times 10^8} \tag{5.33}$$

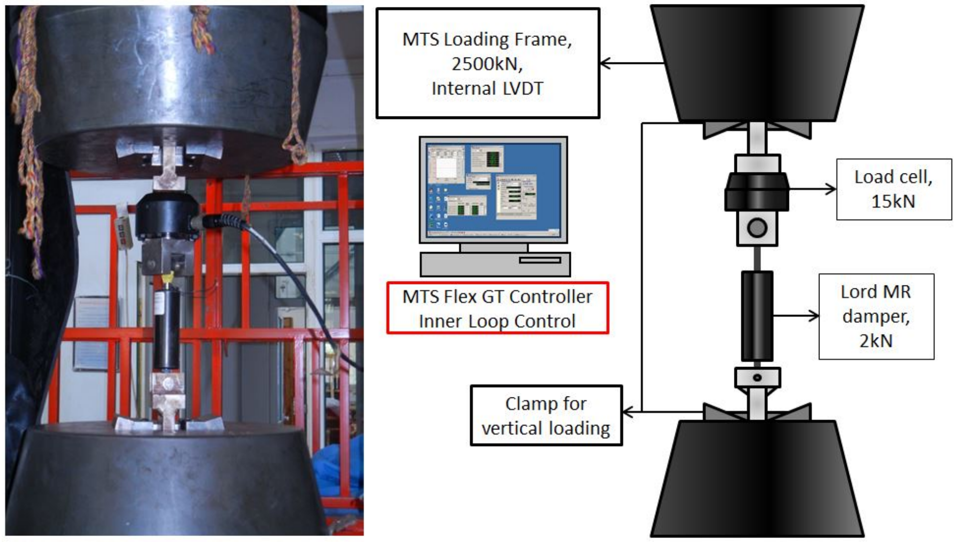


Figure 5.9.: Experimental Setup for a Large Scale MTS Loading Frame with MR Damper Attached

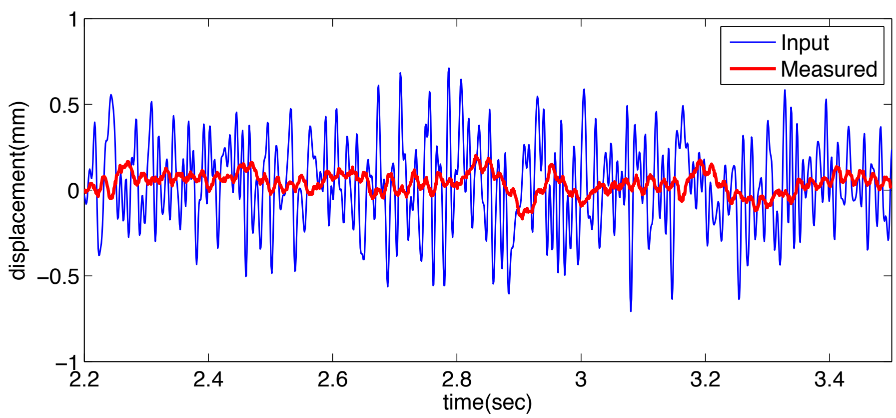
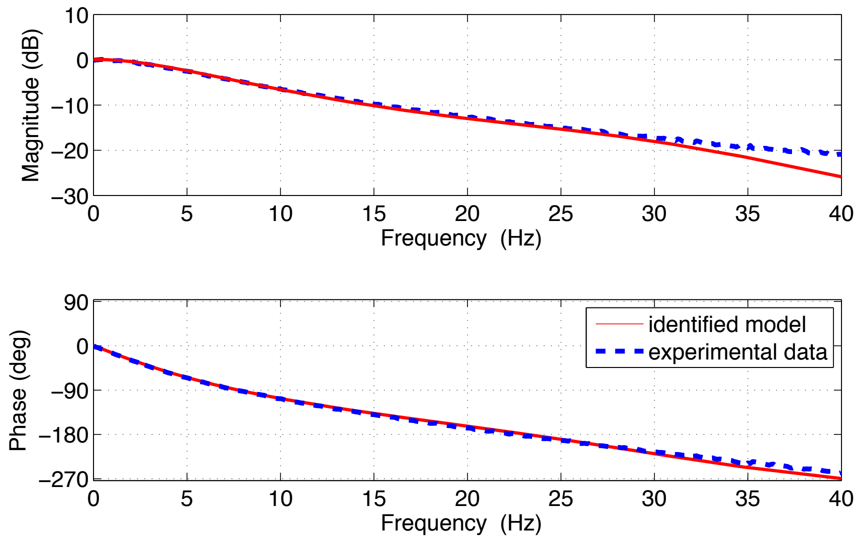


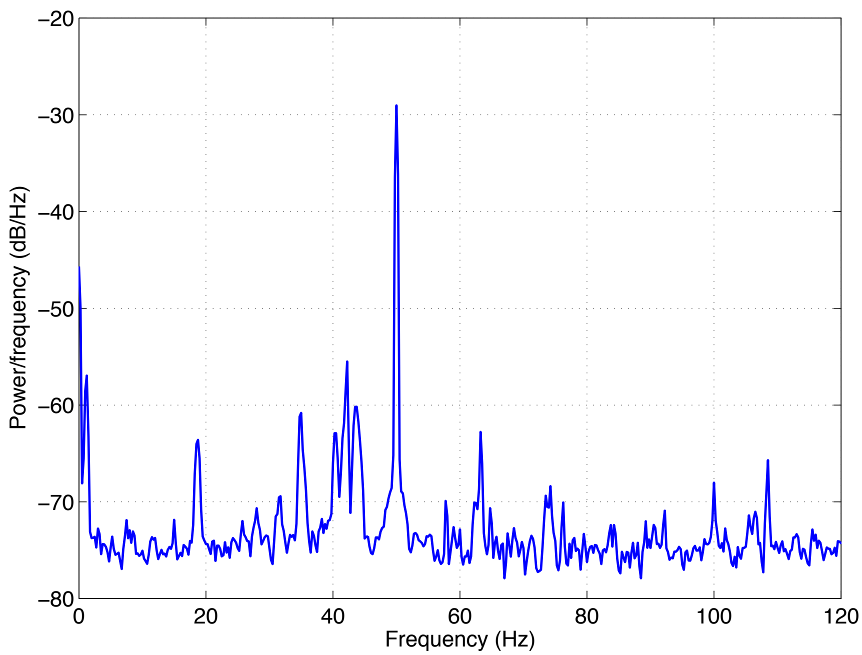
Figure 5.10.: Open Loop System Input and Output for System ID

Fig. 5.11 indicates noise power spectrum density in the hydraulic system. There is a significant peak around 50Hz which is the electric circuit frequency in China. For H_∞ design, W_0 is chosen as:

$$W_{0(A)} = \frac{1.1 \times 10^{17}}{s^6 + 5152s^5 + 9.8 \times 10^6 s^4 + 8.6 \times 10^9 s^3 + 3.6 \times 10^{12} s^2 + 6.9 \times 10^{14} s + 4.4 \times 10^{16}} \quad (5.34)$$



(a) Frequence Response and Identified Transfer Function of the Open Loop System



(b) Noise Power Spectrum Density in Hydraulic System

Figure 5.11.: Input-Output System Identification and Noise Analysis: System A

The desired open loop transfer functions W_0, open loop transfer function after control $G_{x_m \cdot x_c, A}K$ and H_∞ closed loop are illustrated in Fig. 5.12. The noise at 50 Hz is present in the desired tracking frequency (flat region in Fig. 5.12) range, thus, it is necessary to use LQE for noise mitigation in the feedback measurement. A comparison is made between two controllers in simulation considering noise in the feedback loop. Case 1 is the original H_∞ design and case 2 consider LQE (RIAC). In Fig. 5.13, for case 1, the noise 50 Hz is observed and well as another peak at 120 Hz is also amplified due to dynamics in the control gain K. However, both the 50 Hz and 120 Hz peaks have been greatly depressed using RIAC (case 2).

An additional small time delay of $T_d = 1.5$ millisecond is found during further tuning. The feed forward compensation parameter is defined as $\alpha = T_d f_s$, where f_s is sampling frequency during testing. A comparison between the RIAC algorithm and the original H_∞ design with the same desired open loop W_0 is shown in Fig. 5.14 using a BLWN with a bandwidth 25 Hz. Further experimental validation is shown in the same figure using bandwidth only up to 12 Hz which considers the hydraulic fluid velocity limitation. The comparison results show that the experimental result matches the RIAC simulation.

5.4.2 Control Validation on Setup B

The small scale loading frame shown in Fig. 5.15 is located in Intelligent Infrastructure System Laboratory (IISL), Purdue University. This hydraulic system has the maximum loading capacity of 10 kN (velocity limit). The actuator in the loading frame is equipped with an internal LVDT and is controlled by SC6000 controller as the inner PID loop. The external command is applied through a high performance Speedgoat/xPC (Speedgoat GmbH, 2011) real-time kernel. High-resolution, high accuracy 18-bit analog I/O boards are integrated into this digital control system that supports up to 32 differential simultaneous A/D channels and 8 D/A channels, with a minimum I/O latency of less than 5 μs for all channels.

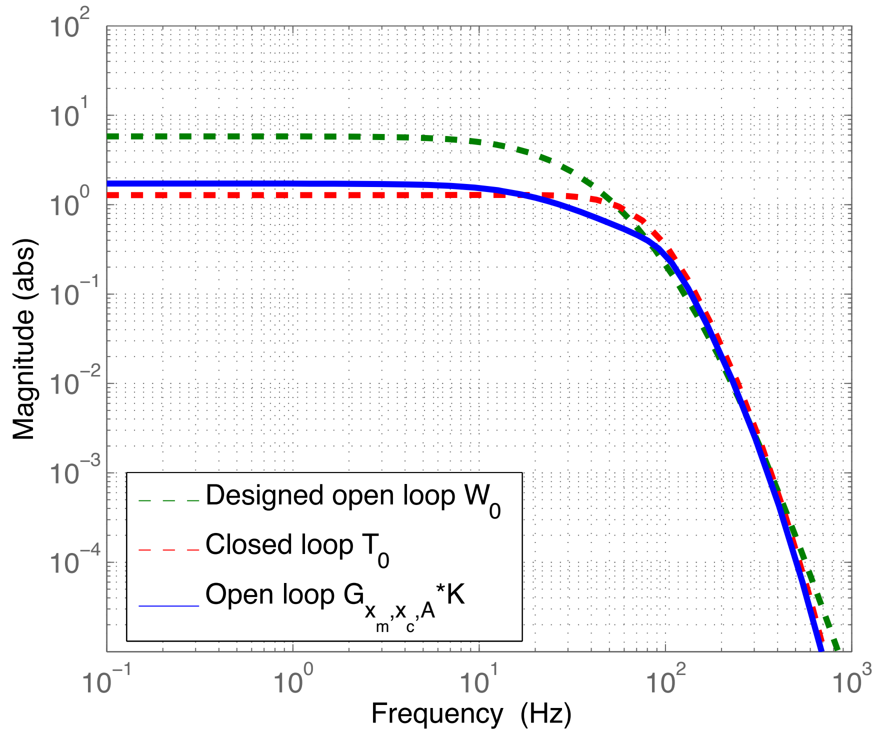


Figure 5.12.: H_∞ Feedback Loop Design, System A

The magneto-rheological (MR) damper is made by LORD company (RD-8041-1). Specified peak to peak damper force is greater than 2000N when subjected to a velocity of 1.97 in/sec (5 cm/sec) and 1 Amps current input. The MR damper is operated with external excitation current 0.0 and 1.0 Amps for passive off and passive on conditions. A LORD Wonder Box device provides closed-loop current control that operates as an interface device between DAQ output and MR damper. The output current with the Wonder Box will be 0.0 A when the control input is approximately 0.4-0.6 V, and is linearly proportional to the input voltage above.

In setup B, the loading capacity of the actuator is at the same order of magnitude as the MR damper maximum force. Thus, the MR damper operating condition affects the response and properties of the hydraulic system. The hydraulic system is identified with MR damper on/off condition [38], [39] using 0-100Hz BLWN and time domain

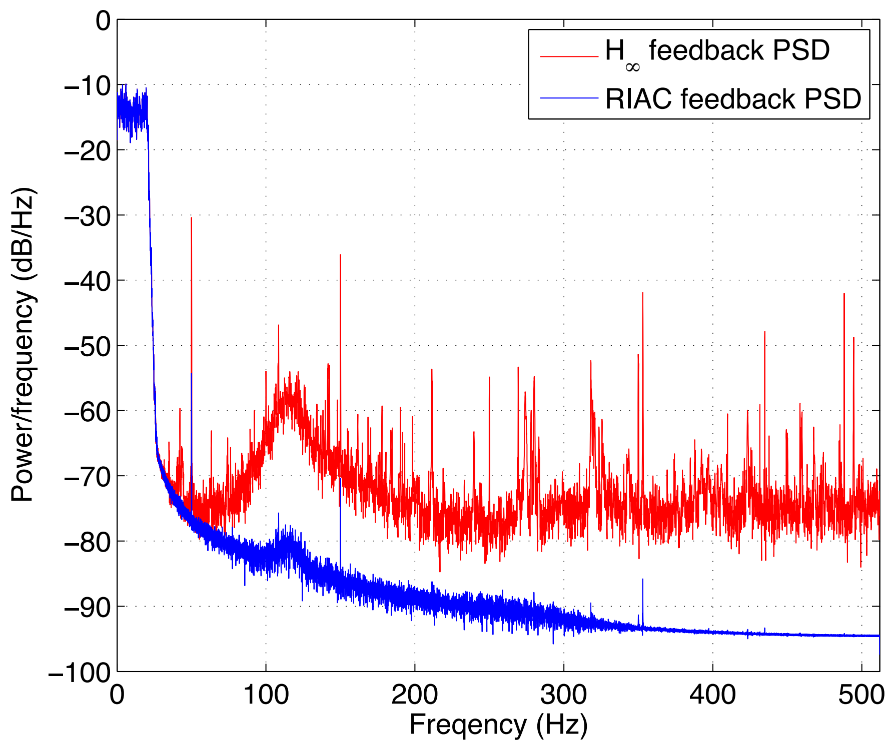


Figure 5.13.: Simulation Comparsion: System measurement PSD Due to Feedback Noise, System A

response is compared in Fig. 5.16. Similarly, the plant can be written in a fourth order transfer function as Eq. 5.35 - Eq. 5.37, comparison between experimental and estimation frequency response is in Fig. 5.15.

$$G_{x_m,x_c,OFF,b} = \frac{3.12 \times 10^9}{s^4 + 517.47s^3 + 3.008 \times 10^5 s^2 + 5.49 \times 10^7 s + 3.17 \times 10^9} \tag{5.35}$$

$$G_{x_m,x_c,ON,b} = \frac{4.70 \times 10^9}{s^4 + 639.55s^3 + 3.50 \times 10^5 s^2 + 7.51 \times 10^7 s + 4.79 \times 10^9} \tag{5.36}$$

$$G_{x_m,x_c,AVG,b} = \frac{3.91 \times 10^9}{s^4 + 578.51s^3 + 3.25 \times 10^5 s^2 + 6.50 \times 10^7 s + 3.98 \times 10^9} \tag{5.37}$$

And the loop shaping design for setup B is:

$$W_{0(B)} = \frac{1.13 \times 10^8}{s^3 + 250s^2 + 1.08 \times 10^5 s + 2 \times 10^6} \tag{5.38}$$

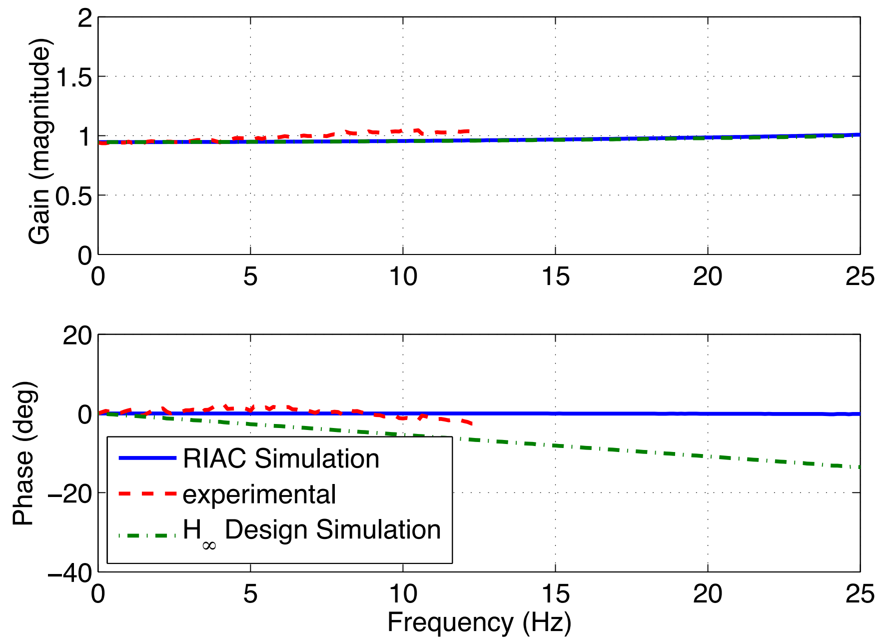


Figure 5.14.: Control Simulation and Experimental Performance, System A

Similarly, noise content is defined as displacement response (LVDT signal) measured with zero input to the inner loop. The actuator feedback signal is well grounded, and in this case there is no significant peak shown in noise power spectrum density plot in Fig. 5.17. Thus, the desired open loop W_0 is designed to be more aggressive as in Fig. 5.18. The efficacy of LQE in RIAC algorithm is demonstrated through a comparison study similar to that done with setup A where measured noise is added to the feedback loop for both H_∞ control case and RIAC control case. The power spectrum density of measured system outputs of both systems are compared in Fig. 5.19. Measurement noise impact is significantly depressed by RIAC same as found in setup A.

An additional time delay $T_d = 4$ msec is found during tuning. The feedforward compensation parameter is the same as defined before $\alpha = T_d f_s$. Fig. 5.20 demonstrates the comparison between simulation of a 30 Hz BLWN tracking using RIAC,

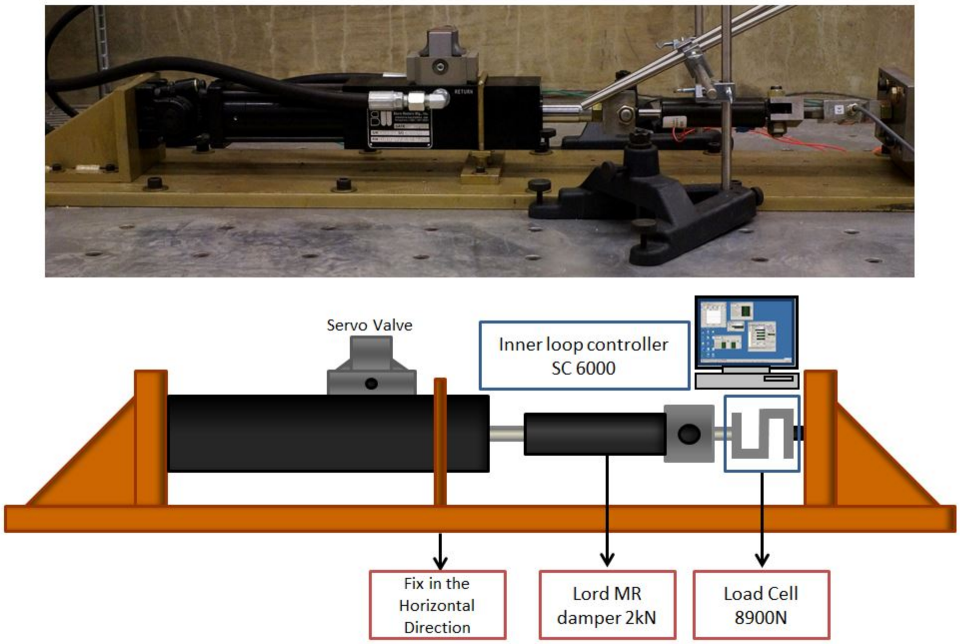


Figure 5.15.: Experimental Setup for a Small Scale Actuator with MR damper Attached, System B

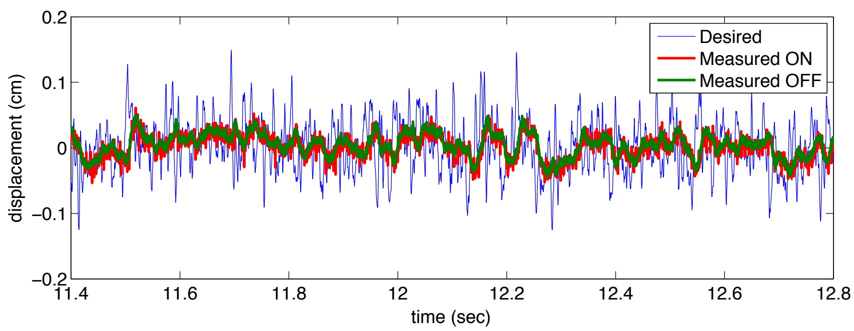
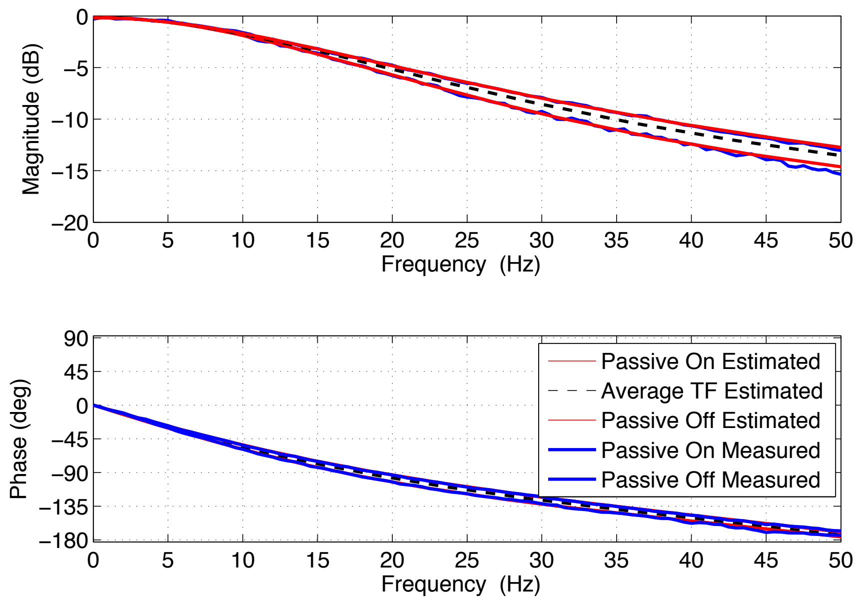
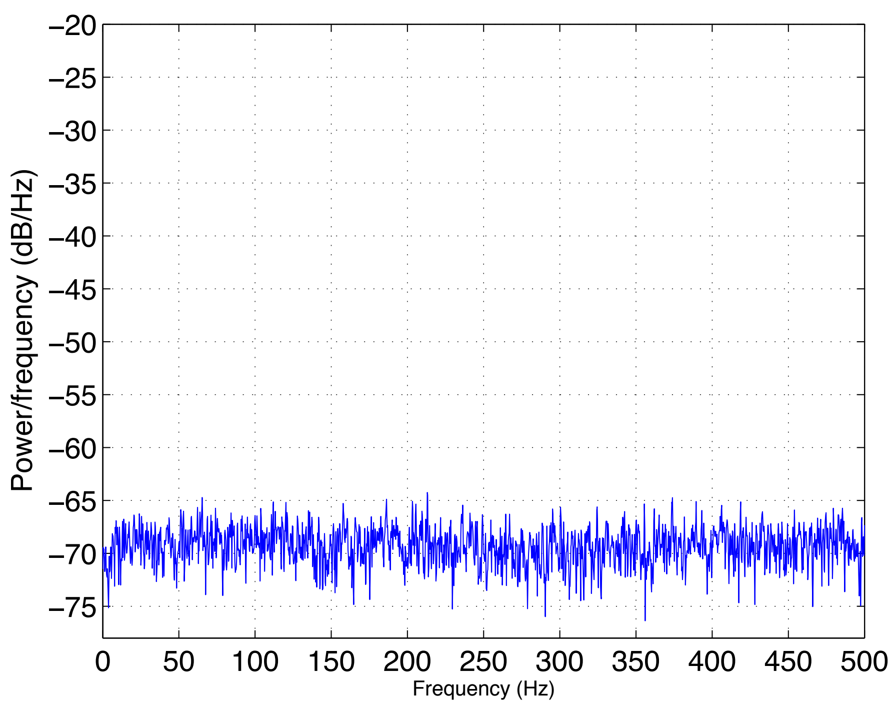


Figure 5.16.: Open Loop System Input and Output For MR damper On/Off condition, System B

H_∞ and an experimental test of 25 Hz BLWN. The feed forward block helps mitigate any residual delay in the original H_∞ design.



(a) Frequence Response and Identified Transfer Function of the Open Loop System



(b) Noise Power Spectrum Density in Hydraulic System

Figure 5.17.: Input-Output System Identification and Noise Analysis, System B

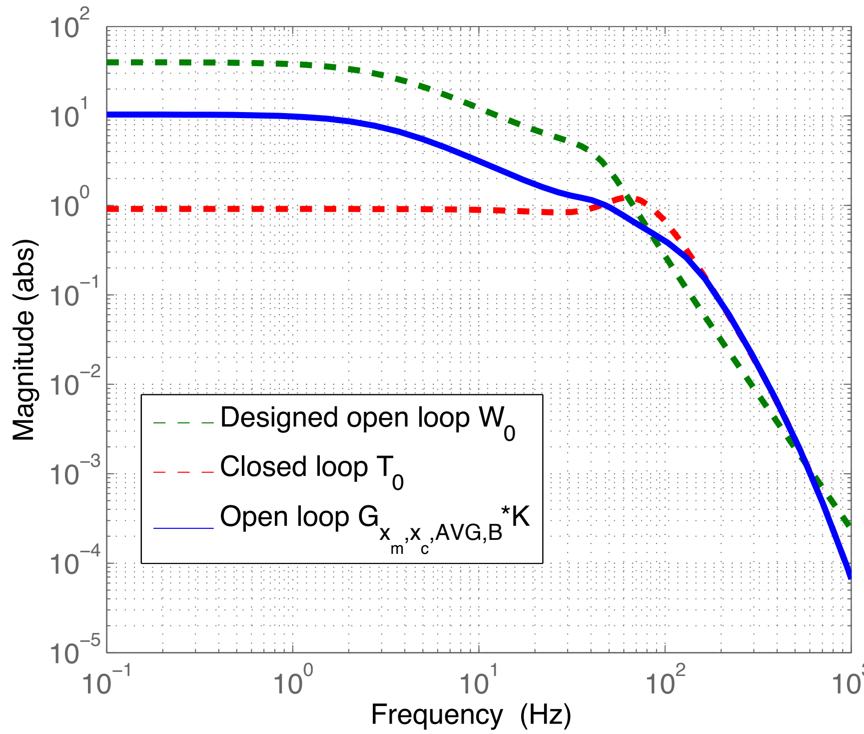


Figure 5.18.: H_∞ Feedback Loop Design, System B

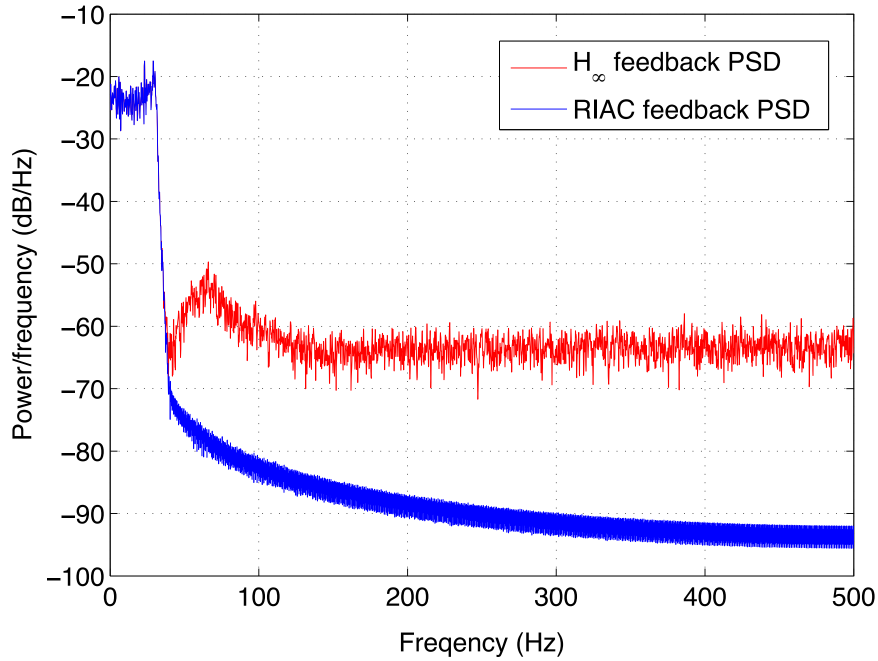


Figure 5.19.: Simulation Comparsion: System Measurement PSD Due to Feedback Noise, System B

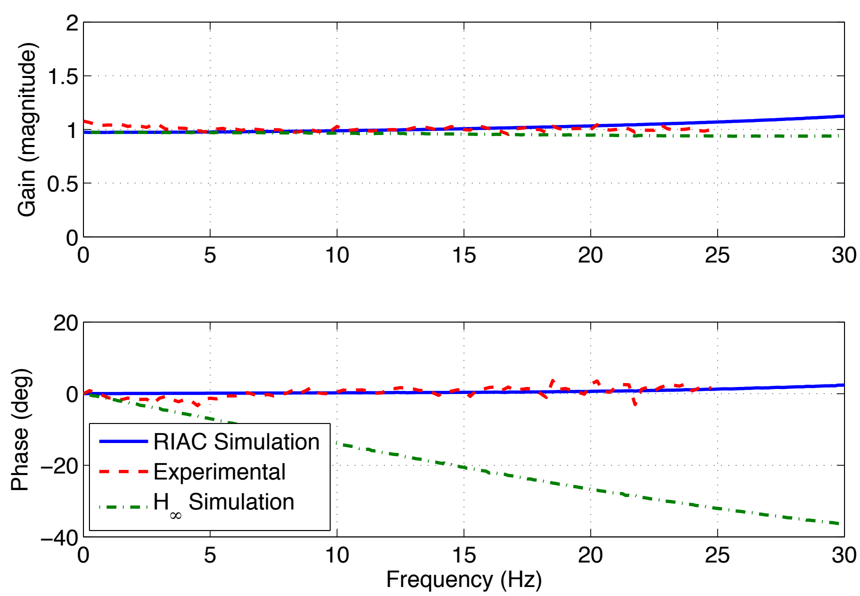


Figure 5.20.: Control Simulation and Experimental Performance, System B

5.5 Real Time Hybrid Simulation Results

The RTHS application studied for validating RIAC algorithm is shown in Fig. 5.1. The target structure is a three story steel frame equipped with an MR damper on the first floor for earthquake response mitigation [83]. The numerical component is the three story frame identified from a physical setup located in HIT and the MR damper is tested experimentally. The restoring force in Eq. 5.1 is the force produced by the MR damper. Three earthquake records are tested using RTHS.

The three story frame is lightly damped and the modes of the structure are at 2.89 Hz, 8.069 and 12.286 Hz, respectively and the experimentally identified mass, stiffness and damping matrices of the structure are:

$$\begin{aligned} M^N &= \begin{bmatrix} 419.5 & 4.4 & 2.2 \\ 4.4 & 364.5 & 10.0 \\ 2.2 & 10.0 & 319.4 \end{bmatrix} kg \ , \\ C^N &= \begin{bmatrix} 88.1 & -4.1 & -1.8 \\ -4.1 & 74.3 & -4.5 \\ -1.8 & -4.5 & 61.2 \end{bmatrix} N/(m/s) \ , \\ K^N &= \begin{bmatrix} 143.3 & -72.1 & 3.7 \\ -72.1 & 130.6 & -60.7 \\ 3.7 & -60.1 & 54.7 \end{bmatrix} \times 10^4 \ N/m \end{aligned} \tag{5.39}$$

In RTHS validation, three different earthquake records are implemented as the excitation including 0.5 scaled El-Centro earthquake, 0.35 scaled Kobe earthquake and full scale Morgan earthquake. MR damper is set as on and off mode for each

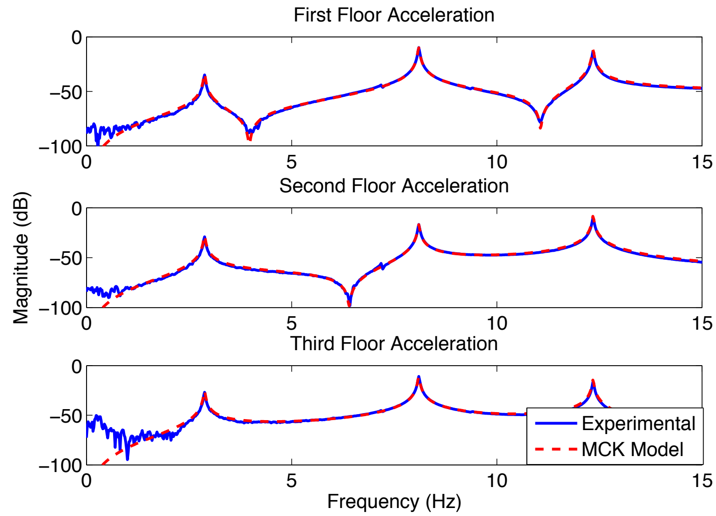


Figure 5.21.: Experimental and Identified Structural Frequency Response

test, respectively. To quantitatively analyze control performance of RIAC algorithm, four error indicators are considered here:

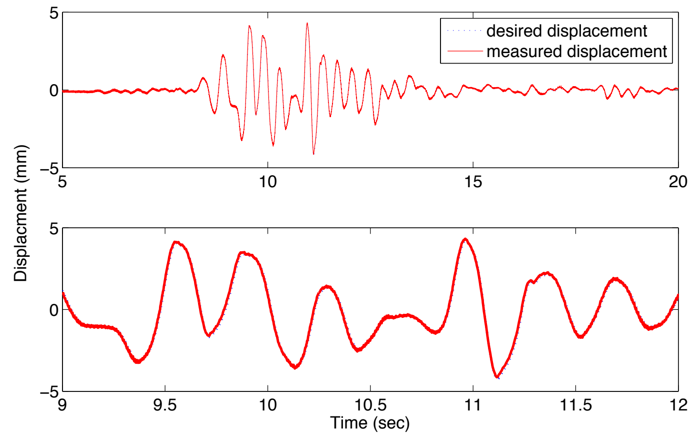
$$J_1 = \sqrt{\frac{\Sigma_n (D_m - D_d)^2}{n}} = RMS(D_e) \tag{5.40}$$

$$J_2 = \sqrt{\frac{\Sigma_n (D_m - D_d)^2}{n}} / \sqrt{\frac{\Sigma_n D_d^2}{n}} = RMS(D_e)/RMS(D_d) \tag{5.41}$$

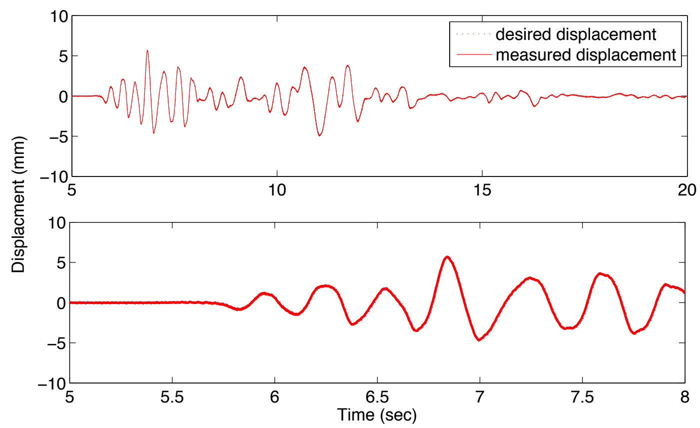
$$J_3 = \sqrt{\frac{\Sigma_n (D_m - D_d)^2}{n}} / max(D_d) = RMS(D_e)/max(D_d) \tag{5.42}$$

where, D_m is measured displacement, D_d is desired displacement and $D_e = D_m - D_d$ is the tracking error.

Table 5.3 listed quantitative errors for each cases under Eq. 5.40 - 5.42. The overall results are good during testing on experimental setup A. However, for MR damper passive off case, the desired motion of actuator exceeds its velocity limit at 90 mm/s. Therefore, passive on cases perform significantly better compared to passive off cases. Fig. 5.22 illustrated displacement tracking performance in passive on case in setup A, and also the displacement tracking and velocity saturation during passive off tests as in Fig. 5.23.



(a) 0.35 scale Kobe earthquake, passive on

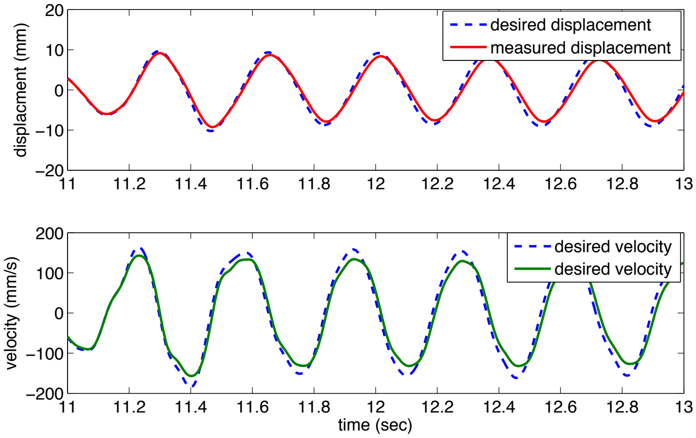


(b) Full scale Morgan earthquake, passive on

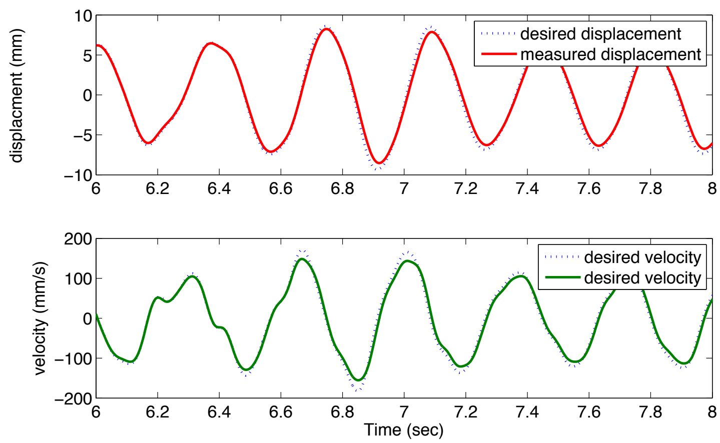
Figure 5.22.: RTHS results, System A

Table 5.3.: Error indices for all cases

	Setup A			Setup B		
Earthquake Record	J_1	J_2	J_3	J_1	J_2	J_3
El Centrol Passive On	0.0440	0.0306	0.0090	0.0361	0.0736	0.0181
El Centrol Passive Off	0.0634	0.0615	0.0093	0.0652	0.0705	0.0137
Morgan Passive On	0.043	0.0353	0.0075	0.0474	0.0732	0.0139
Morgan Passive Off	0.1163	0.0872	0.0135	0.0344	0.0228	0.0054
Kobe Passive On	0.0387	0.0330	0.0092	0.0389	0.0789	0.0159
Kobe Passive Off	0.3234	0.2048	0.0334	0.0434	0.0208	0.0064



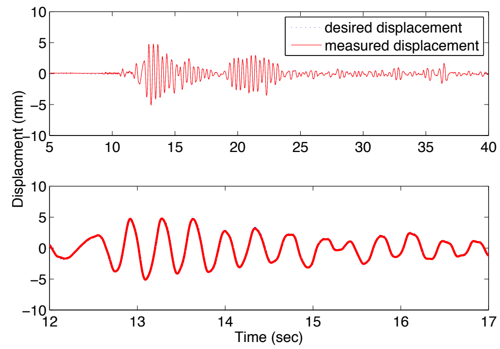
(a) 0.35 scale Kobe earthquake, passive off, tracking performance and velocity saturation



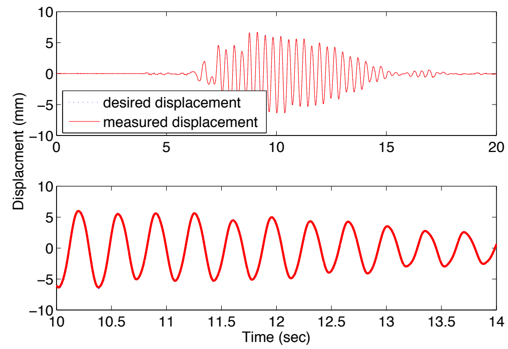
(b) Full scale Morgan earthquake, passive off, tracking performance and velocity saturation

Figure 5.23.: RTHS results for System A with Velocity Saturation

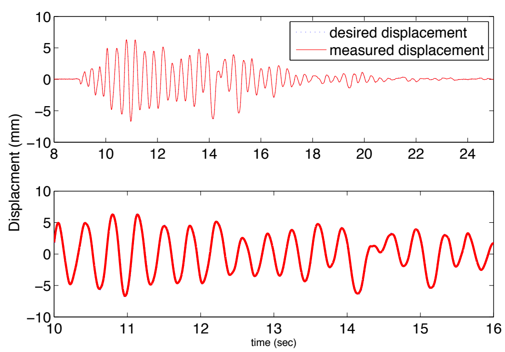
To demonstrate the RIAC controller performance on another setup, RTHS tests have been implemented on setup B shown in Fig. 5.24. Tracking results match well (Fig. 5.24(a) - 5.24(c)) for all cases, and tracking errors are showing in Fig. 5.25(a) - 5.25(d). Quantitative results listed in Table 5.3 illustrated the consistency between all six tests on setup B.



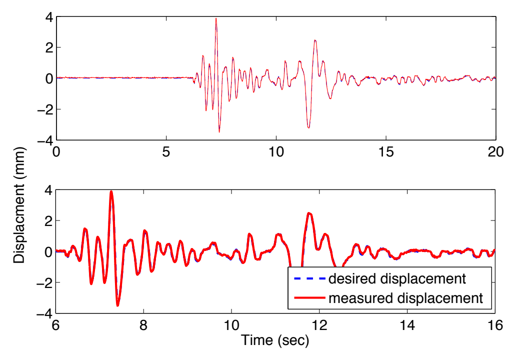
(a) 0.5 scale El-Centro earthquake, passive on



(b) 0.35 scale Kobe earthquake, passive off

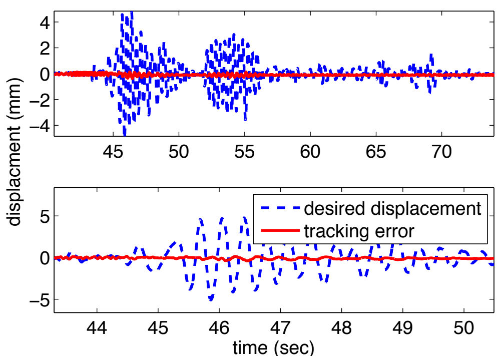


(c) Full scale Morgan earthquake, passive off

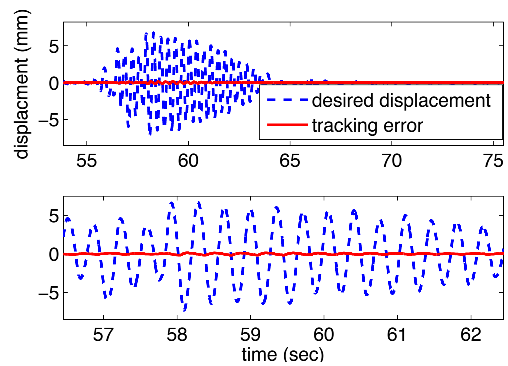


(d) Full scale Morgan earthquake, passive on

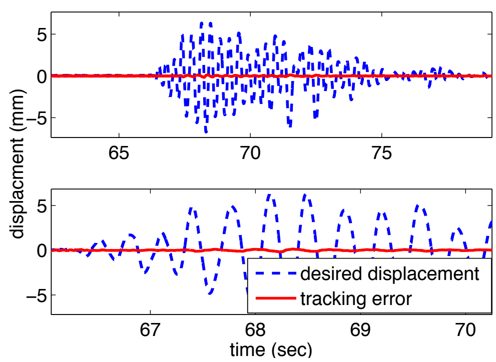
Figure 5.24.: RTHS results, System B



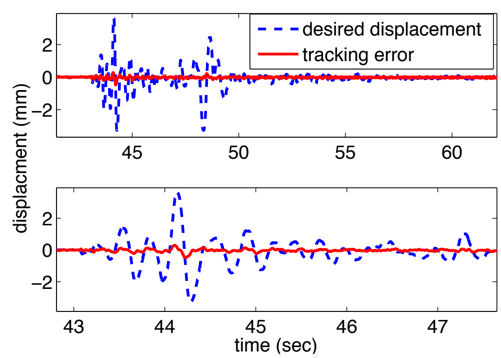
(a) 0.5 scale El-Centro earthquake, passive on, tracking error



(b) 0.35 scale Kobe earthquake, passive off, tracking error



(c) Full scale Morgan earthquake, passive off, tracking error



(d) Full scale Morgan earthquake, passive on, tracking error

Figure 5.25.: RTHS tracking errors, System B

5.6 Conclusions

The need to achieve accurate boundary condition synchronization is strongly linked to the success of RTHS test. Most of the recent research has focused on time delay compensation and hydraulic system dynamics. Model uncertainties and noise present in the hydraulic system has not been carefully considered in the design of the actuator controller previously. A new algorithm for actuator control is proposed in this chapter. By integrating the most effective feature to develop a flexible and versatile closed loop control system, the new robust integrated actuator control algorithm meets the needs of the RTHS user. The limitations of the original H_∞ design are overcome, while the robust stability is preserved. In both simulation and experimental results, the RIAC significantly reduced noise impact on the closed loop system, especially when the noise peak is in the desired control frequency range. RIAC enables the user to fully consider the system dynamics as well as the uncertainty (error or measurement noise) and still establish a design yielding highly accurate tracking. Test results discussed in the chapter indicated that RIAC is appropriate and effective for controlling both large and small, slow and fast systems and is very accurate and effective for RTHS. The tracking results achieved in both setups demonstrates feasibility and accuracy of RIAC.

6. INTEGRATION ALGORITHM IN RTHS

As indicated in chapter 2, in hybrid simulation, the execution of the $i+1^{th}$ step displacement is implemented through several small sub-step increments, and the steady state response of the specimen restoring force at the $i+1^{th}$ step is available at the final sub-step. However, the implementation of RTHS is on an RTOS platform which is normally executed at one single fixed rate, no sub-step data acquisition is allowed. Therefore, the response of the experimental substructure is not available instantaneously. In RTHS, the restoring force is measured at the beginning of the RTHS $i+1$ step which is the response of the i^{th} step displacement. For example, even for computing response at the first time step, the restoring force is zero since no input has yet been sent to the experimental substructure. However, the true restoring force should be $R(x_1, \dot{x}_1)$.

Consequently, a unit delay, as shown in Fig. 6.1, exists in the experimental force measurement. This delay is normally considered as a portion of the total computational delay [52]. Once the transfer system lag is compensated properly to enforce the boundary conditions accurately with almost zero delay, the effect of this computational delay dominates the stability and accuracy of RTHS. This effect is most pronounced for stiff, lightly-damped structures that may have relatively high natural frequencies associated with the first few dominant modes (potentially higher than the common RTHS execution rates of 1 - 4 kHz) where this delay can lead to instability of the RTHS closed loop. Even for stable RTHS, the presence of even small computational delays can have detrimental effect on the accuracy of the results. Further, as some researchers are focusing on the fidelity of RTHS results, there is a need to use larger, more sophisticated numerical models within RTHS. Such high-fidelity models often take more time to run than the conventional RTHS execution rate of 1024 Hz, creating a need for exploring RTHS at lower execution rates and consequently higher computational delays.

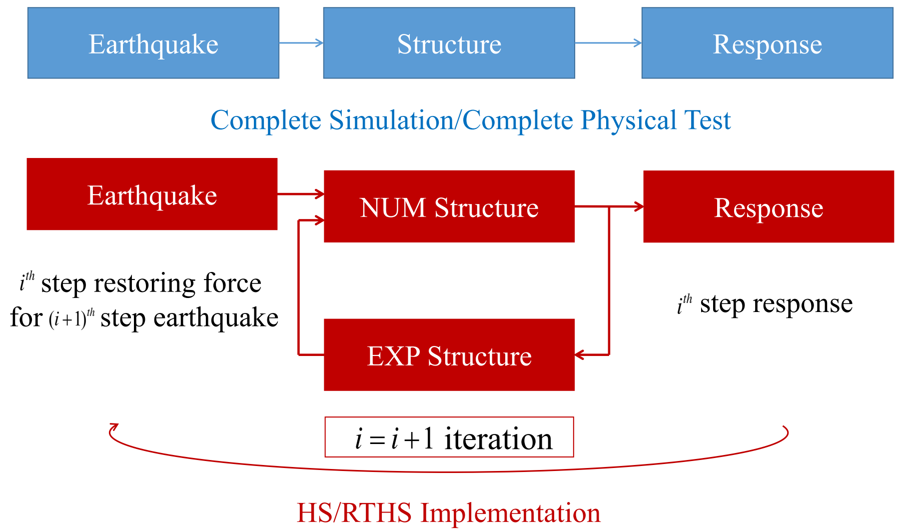


Figure 6.1.: Schematic Drawing for Traditional Simulation/Shake Table Test and RTHS

Many integration schemes have been used in RTHS, including both explicit and implicit methods. Due to the need for fast computation, most of the integration schemes developed for RTHS are explicit. Some explicit algorithms are unconditionally stable and are not affected by the highest natural frequency of the structure [21]; [45]; [46]; [36]. However, when the computational delay is considered, the stability condition for the integration scheme may be affected leading to constrains on the integration step size. Other methods, such as predictor-corrector methods are, also some of the numerical integration techniques used to solve ordinary differential equations. There are some predictor-corrector based numerical integration algorithms that help reduce delays in RTHS. However, they are more focused on the delay in the transfer system rather than the inherent delay in RTHS [48]. Some implicit integration algorithms that have been investigated by researchers include an equivalent force control for solving nonlinear equations of motion [49] and HHT-α method with fixed number of substep iteration [50] which provides stable experimental results for

RTHS. However, the aforementioned HHT-α method requires numerical-experimental information exchange at each substep, where the substep displacement commands is calculated based on the measured restoring force from the previous substep [51].

In this chapter, computational delay is evaluated analytically using different integration algorithms, including the Newmark-β algorithm, CR algorithm, a discrete state-space method and the conventional Runge-Kutta Method. The stability characteristics of RTHS closed loop using these integration schemes are studied here and compared to a no delay/pure simulation case. The results indicate that computational delay in RTHS affects the stability and accuracy of the test, and are also dependent on the partitioning between the experimental and numerical substructures.

6.1 Computational Delay in RTHS

In this section, computation delay is described and discussed in detail. Several integration algorithms developed for RTHS are listed. The effect of computational delay is analyzed by comparing the stability characteristics of the RTHS closed loop using different integration algorithms. Some assumptions are made for the analysis: 1) mass contains only in the numerical portion, experimental mass is ignored; 2) Transfer system (hydraulic actuator) that links the numerical and experimental substructures is considered to be perfectly controlled, desired displacement is assumed to be accurately imposed to experimental substructure; 3) Reference system is considered to be linear time invariant (LTI), including both numerical and experimental substructure; 4) Since multiple DOFs (MDOFs) case analysis can be decomposed into several decoupled single DOF cases, theoretical stability characteristics are only derived for SDOF systems.

Structural dynamics equation of motion:

$$M\ddot{x} + C\dot{x} + Kx = -M\Gamma\ddot{x}_g \tag{6.1}$$

where M, C, K are entire system mass, damping, stiffness matrices, and $\ddot{x}_g$ indicates earthquake excitation. We can write the equation of motion of an RTHS in the form:

$$M\ddot{x} + C^N\dot{x} + K^N x + F^E(x,\dot{x}) = -M\Gamma\ddot{x}_g \tag{6.2}$$

where the superscript $(\ )^N$ and $(\ )^E$ denote the portions in numerical and experimental substructures, and F^E denotes the measured force in experimental substructure.

RTHS typically follows the steps below:

For initial time step, $i = 1$

Step 1. Calculate initial numerical response x_1, given $x_{g,1}$ and $F_1 = 0$.

For time step, $i > 1$

Step 2. Impose displacement x_{i-1}, calculated from previous time step $(i-1)$ to test specimen and measure experimental measured force $F^E(x_{i-1}, \dot{x}_{i-1})$.

Step 3. Calculate numerical response $(x_i, \dot{x}_i)$use integration scheme given $x_{g,i}$ and F^E_{i-1}.

Step 4. Set $i = i + 1$, go to step 2.

In step 2, the desired displacement x_{i-1} should be accurately imposed using the transfer system (hydraulic actuator). In this step, researchers normally assume a delay/lag resulting from actuator dynamics and compensate for it with extrapolation. However, even when perfect tracking in the transfer system is achieved, where the desired displacement x_{i-1} can be implemented simultaneously and force measurement F^E is available and accurate at time step i, the measured force obtained from step i is $F^E(x_{i-1}, \dot{x}_{i-1})$. Consider an LTI system, the discrete equation of motion for RTHS with perfect tracking can then be written:

$$M\ddot{x}_i + C^N\dot{x}_i + K^N x_i = -M\Gamma\ddot{x}_{g,i} - C^E\dot{x}_{i-1} - K^E x_{i-1} \tag{6.3}$$

It is obvious that $x_{i-1} \neq x_i, \dot{x}_{i-1} \neq \dot{x}_i$, thus, Eq. 6.3 is not automatically equivalent to Eq. 6.1.

6.2 Integration Stability with Computational Delay

To understand the differences introduced by the computational delay, different algorithms used for the numerical model are studied in the following sections. Mathematical presentations for both pure simulation or RTHS are derived individually for: Newmark-β (NB) method (central difference (CD) method and averaged acceleration (AA) method), Chen-Ricles (CR) method, Runge-Kutta (RK) method, discrete state space method (Tustin's) method.

6.2.1 Newmark-βethod

Newmark β algorithm is well known and discussed in many textbooks in structural dynamics, the brief derivation is presented. Consider the SDOF case in 6.1, the explicitly calculated displacement, velocity and acceleration are:

$$\dot{x}_i = \dot{x_{i-1}} + \Delta t[(1-\gamma)\ddot{x_{i-1}} + \gamma \ddot{x}_i] \tag{6.4}$$

$$x_i = x_{i-1} + \Delta t \dot{x_{i-1}} + (1-2\beta)/2\Delta t^2 \ddot{x_{i-1}} + \beta \Delta t^2 \ddot{x}_i \tag{6.5}$$

$$\ddot{x}_i = \frac{-Kx_{i-1} - C^* \dot{x_{i-1}} - M^* \ddot{x_{i-1}}}{\hat{M}} \tag{6.6}$$

where, $C^* = (K\Delta t + C)$, $M^* = [\Delta t C(1-\gamma) + K(1-2\beta)/2\Delta t^2$, $\hat{M} = M + \Delta t \gamma C + K\beta \Delta t^2$

Assuming no excitation force $\ddot{x}_{g,i} = 0$, displacement, velocity and acceleration between two adjacent time steps can be related in a recursive form:

$$X_i = A_{NB} X_{i-1} \tag{6.7}$$

where, A_{NB} is the amplification matrix, $X_i = [x_i, \dot{x}_i, \ddot{x}_i]^T$ is the system state.

$$A_{NB} = \begin{bmatrix} 1-\beta\Delta t^2 K/\hat{M} & \Delta t - \beta\Delta t^2 C^*/\hat{M} & (1-2\beta)\Delta t^2/2 - \beta\Delta t^2 M^*/\hat{M} \\ -K\gamma\Delta t/\hat{M} & 1-\gamma\Delta t C^*/\hat{M} & (1-\gamma)\Delta t - \gamma\Delta t M^*/\hat{M} \\ -K/\hat{M} & -C^*/\hat{M} & -M^*/\hat{M} \end{bmatrix} \tag{6.8}$$

Consider RTHS with unit delay, solving the finite difference equation using Newmark-β integration as in Eq. 6.3, the Eq.6.5 and 6.4 preserves but the acceleration calculation is slightly different

$$\ddot{x}_i = \frac{-Kx_{i-1} - C_R^* \dot{x_{i-1}} - M_R^* \ddot{x_{i-1}}}{\hat{M}_R} \tag{6.9}$$

where, $C_R^* = (K^N\Delta t + C)$, $M_R^* = [\Delta t C^N(1-\gamma) + K^N(1-2\beta)/2\Delta t^2$, $\hat{M}_R = M + \Delta t\gamma C^N + K^N\beta\Delta t^2$.

Based on Eq.6.4, 6.5, 6.9, same amplification matrix in RTHS can be written in a recursive form:

$$X_i = A_{NB,R} X_{i-1} \tag{6.10}$$

$$A_{NB,R} = \begin{bmatrix} 1-\beta\Delta t^2 K/\hat{M}_R & \Delta t - \beta\Delta t^2 C_R^*/\hat{M}_R & (1-2\beta)\Delta t^2/2 - \beta\Delta t^2 M_R^*/\hat{M}_R \\ -K\gamma\Delta t/\hat{M}_R & 1-\gamma\Delta t C_R^*/\hat{M} & (1-\gamma)\Delta t - \gamma\Delta t M_R^*/\hat{M}_R \\ -K/\hat{M}_R & -C_R^*/\hat{M}_R & -M_R^*/\hat{M}_R \end{bmatrix} \tag{6.11}$$

Central difference method

Newmark-β method is equivalent to Central Difference (CD) method when $\beta = 0$, $\gamma = 1/2$. The evaluation of RTHS integration stability is based on the comparison of spectral radii between Eq. 6.8, Eq. 6.11 with $\beta = 0$, $\gamma = 1/2$.

In Fig. 6.2 and 6.3, the x axis is the normalized integration time interval $\Omega = \omega \cdot dt$, with ω indicating the natural frequency of the SDOF structure. Results indicate that for the undamped case, due to the computational delay, any partitioning ratio

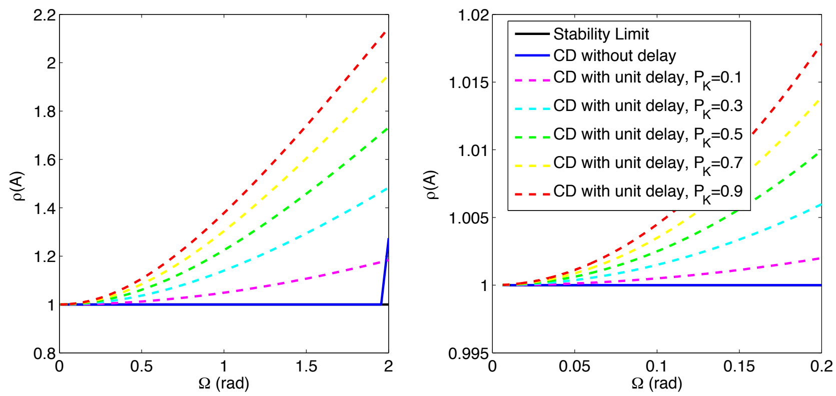


Figure 6.2.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Central Difference, $\zeta = 0$

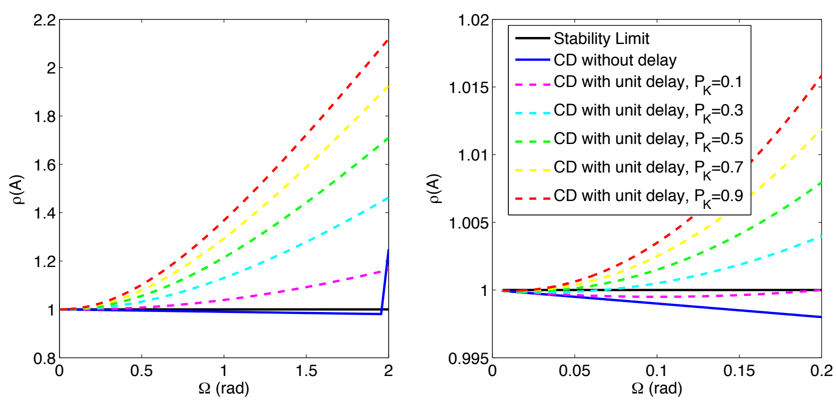


Figure 6.3.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Central Difference, $\zeta = 0.01$

results an unstable RTHS, showing as the maximum spectral radius larger than the unit value, this conclusion extended the conclusion in [87] which majorly focused on experimental delay/lag . Any type of delay/lag in RTHS system is equivalent to add negative damping to structure. For the undamped case, any negative damping makes RTHS loop unstable.

When the system is lightly damped, stability is preserved until $\Omega = 2$ for entire system integration. In RTHS, when computational delay occurs, the stability range for Ω is largely reduced. From the curves in Fig. 6.3, it shows that partition rate also changes the stability characteristics. Generally, the larger partitioning rate, the smaller tolerance for Ω.

Average acceleration method

Newmark-β method is proven to be stable when $2\beta \leq \gamma \leq 1/2$, Newmark-$\beta$ method is equivalent to averaged acceleration method for $\beta = 1/4$, $\gamma = 1/2$. Similarly, spectral radii are used as a stability criterion.

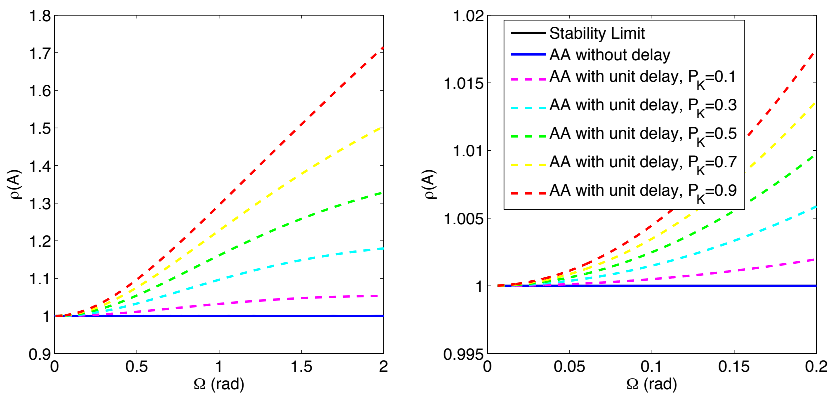


Figure 6.4.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Average Acceleration, $\zeta = 0$

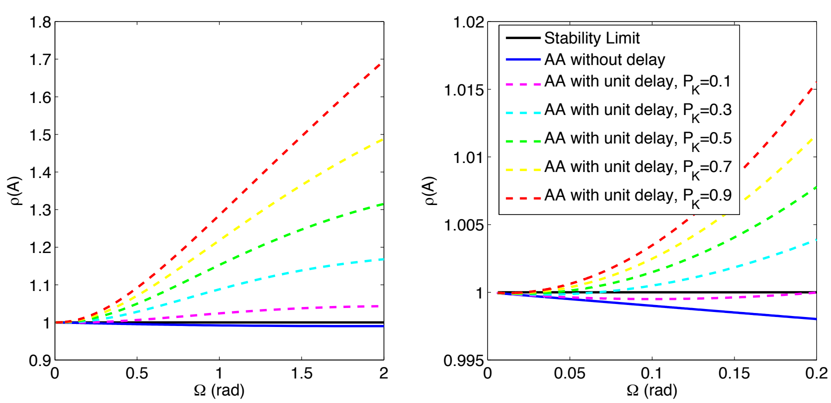


Figure 6.5.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Average Acceleration, $\zeta = 0.01$

The average acceleration method is unconditional stable for entire system integration. However, for RTHS case, it is unstable for no damping structure, similar as central difference method. For lightly damped structure, the stability range for Ω decreases largely as well.

6.2.2 CR method

Another direct integration algorithm developed after Chen and Ricles [36] is developed in discrete transfer function form under control theory.

The Laplace transformation of Eq. 6.1 is:

$$Ms^2X(s) + 2CsX(s) + KX(s) = -MX_g(s) \tag{6.12}$$

$$G(s) = \frac{X(s)}{X_g(s)} = \frac{1}{Ms^2 + Cs + K} = \frac{1}{(s^2 + 2\zeta\omega_n s + \omega_n^2)M} \tag{6.13}$$

where, $s = j\omega$ which is the variable in Laplace domain; $X(s)$ and $X_g(s)$ is Laplace transform of structural displacement $x(t)$ and earthquake excitation $\ddot{x}_g(t)$, $G(s)$ is the transfer function in frequency domain. Pole location for Eq. 6.13 is calculated:

$$p_{1,2} = -\zeta \cdot \omega_n \pm i\omega_n \cdot \sqrt{(1-\zeta^2)} \tag{6.14}$$

where $i = \sqrt{-1}$. Using Tustin's method for discretization:

$$z = e^{s\Delta t} = \frac{1 + s \cdot \Delta t/2}{1 - s \cdot \Delta t/2} \tag{6.15}$$

Pole locations for the discrete system using Tustin method are:

$$p_{z,1,2} = \frac{2 + (-\zeta \pm i\sqrt{1-\zeta^2})\omega_n \Delta t}{2 - (-\zeta \pm i\sqrt{1-\zeta^2})\omega_n \Delta t} \tag{6.16}$$

CR integration assumes:

$$\dot{x}_{i+1} = \dot{x}_i + \alpha_1 \cdot \Delta t \ddot{x}_i \tag{6.17}$$

$$x_{i+1} = x_i + \Delta t \cdot \dot{x}_i + \alpha_2 \cdot \Delta t^2 \ddot{x}_i \tag{6.18}$$

Substitute Eqs. 6.18, 6.17 into Eq. 6.16 and solve for α_1 and α_2:

$$\alpha_1 = \alpha_2 = \frac{4}{4 + 4\zeta\omega_n \Delta t + \omega_n^2 \Delta t^2} \tag{6.19}$$

Write Eqs. 6.17 and 6.18 into recursive form, assuming to external force ($x_{g,i} = 0$):

$$A_{CR} = \begin{bmatrix} 1 & \Delta t & \Delta t^2 \cdot \alpha \\ 0 & 1 & \Delta \alpha \\ -K/M & -(C + K\Delta t)/M & -(C \cdot \Delta t \cdot \alpha + K \cdot \Delta t^2 \cdot \alpha)/M \end{bmatrix} \tag{6.20}$$

Similarly, solving the equation for Eq. 6.3:

$$A_{CR,R} = \begin{bmatrix} 1 & \Delta t & \Delta t^2 \cdot \alpha \\ 0 & 1 & \Delta \alpha \\ -K/M^N & -(C + K^N \Delta t)/M^N & -(C^N \cdot \Delta t \cdot \alpha + K^N \cdot \Delta t^2 \cdot \alpha)/M \end{bmatrix} \tag{6.21}$$

Compare the spectral radii for ordinary simulation and RTHS case for stability analysis.

Similarly, the plots prove that CR method is unconditional stable for entire structure. However, when computational delay exist, RTHS stability characteristics changed significantly.

6.2.3 Discrete state space method

In control engineering, state space representation is a mathematical model of a physical system as a set of input, output and state variables related by first-order dif-

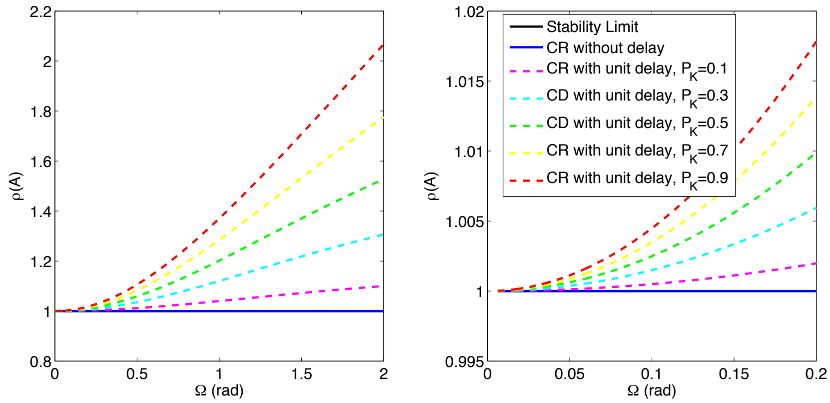


Figure 6.6.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, CR algorithm, $\zeta = 0$

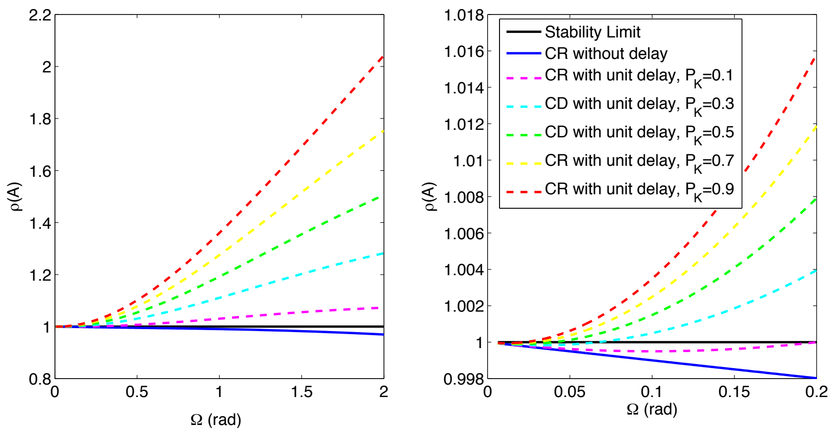


Figure 6.7.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, CR algorithm, $\zeta = 0.01$

ferential equations. For continuous linear time invariant (LTI) systems, the standard continuous state-space representation is given below:

$$\dot{x} = Ax + Bu \tag{6.22}$$

$$y = Cx + Du \tag{6.23}$$

where, x is the vector $(n \times 1)$ of state, u is the vector $(p \times 1)$ of input and y is the vector $(q \times 1)$ of output. A is the system matrix $(n \times n)$ (same as amplification matrix), B is the input matrix ($n \times p$), C is the output matrix $q \times n$, and D is the feedforward matrix $q \times p$.

Using state space representation for Eq. 6.1

$$\dot{X} = A_S X + B_S U \tag{6.24}$$

$$Y = C_S X + D_S U \tag{6.25}$$

where $U = \ddot{x}_g$,

$$A_S = \begin{bmatrix} 0 & 1 \\ -K/M & -C/M \end{bmatrix} B_S = \begin{bmatrix} 0 \\ -1 \end{bmatrix} X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$
$$C_S = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -K/M & -C/M \end{bmatrix} D_S = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} Y = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix}$$

Regarding to time step value Δt, the amplification matrix can be written into discrete form:

$$X(k+1) = e^{A_S \Delta t} X(k) + A_S^{-1}(e^{A_S \Delta t} - I) B_S u(k) \tag{6.26}$$

Using Tustin's method to discretize continuous state space representation:

$$e^{A_S \Delta t} = (I + \frac{1}{2} A_S \Delta t)(I - \frac{1}{2} A_S \Delta t)^{-1} \tag{6.27}$$

Eq. 6.27 is the discretized amplification matrix.

$$A_{DSS} = \begin{bmatrix} 1 & \frac{1}{2}\Delta t \\ -\frac{1}{2}\Delta t K/M & -\frac{1}{2}\Delta t C/M + 1 \end{bmatrix} \begin{bmatrix} 1 & -\frac{1}{2}\Delta t \\ \frac{1}{2}\Delta t K/M & \frac{1}{2}\Delta t C/M + 1 \end{bmatrix}^{-1} \tag{6.28}$$

$$B_{DSS} = A_S^{-1}(A_{DSS} - I) B_S \tag{6.29}$$

Use continuous state space representation for RTHS formulation as in Eq. 6.2:

$$\dot{X} = A_R X + B_R U_R \tag{6.30}$$

$$Y = C_R X + D_R U \tag{6.31}$$

where $U_R = \ddot{x}_g + F^E$, recall $F^E = K^E x + C^E \dot{x}$

$$A_R = \begin{bmatrix} 0 & 1 \\ -K^N/M & -C^N/M \end{bmatrix} B_R = \begin{bmatrix} 0 \\ -1 \end{bmatrix} X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$
$$C_R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -K^N/M & -C^N/M \end{bmatrix} D_R = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} Y = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix}$$

Consider experimental force for no earthquake excitation case, where $U_R = F^E$. The amplification matrix is:

$$A_{DSSR} = A^*_{DSS} + B^*_{DSS} \begin{bmatrix} -K^E/M & -C^E/M \end{bmatrix} \tag{6.32}$$

where,

$$A^*_{DSS} = \begin{bmatrix} 1 & \frac{1}{2}\Delta t \\ -\frac{1}{2}\Delta t K^N/M & -\frac{1}{2}\Delta t C^N/M + 1 \end{bmatrix} \begin{bmatrix} 1 & -\frac{1}{2}\Delta t \\ \frac{1}{2}\Delta t K^N/M & \frac{1}{2}\Delta t C^N/M + 1 \end{bmatrix}^{-1}$$
$$B^*_{DSS} = A_R^{-1}(A^*_{DSS} - I)B_R$$

The spectral radii is also used for stability analysis, the comparisons between simulation of the entire system and the RTHS systems are illustrated in Fig. 6.8 and 6.9.

6.2.4 Runge-Kutta method

The Runge-Kutta (RK) integration method is considered as a single-step method that evolves the solution from X_{i-1} to X_i, without requiring information from the

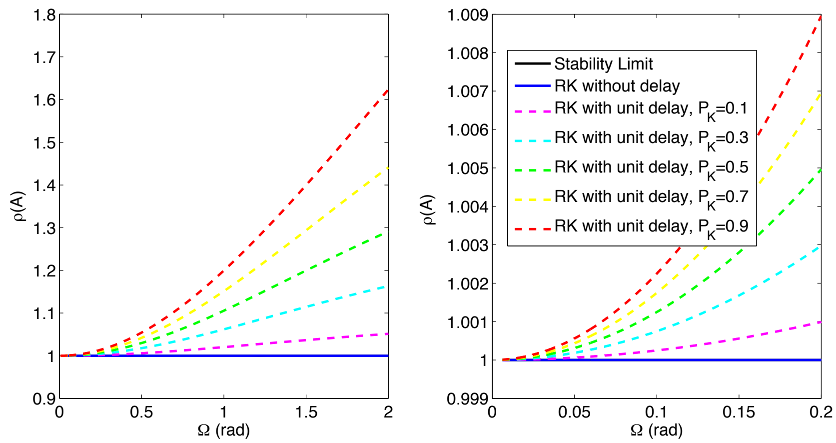


Figure 6.8.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Discrete Stat Space using Tustin's method, $\zeta = 0$

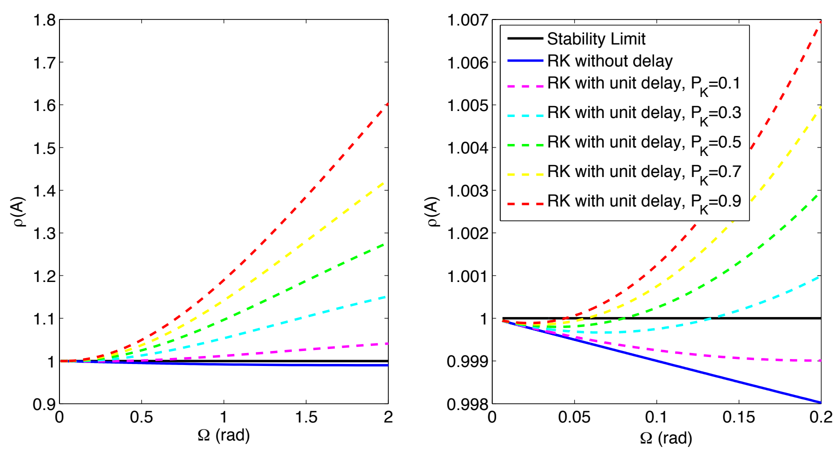


Figure 6.9.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Discrete Stat Space using Tustin's method, $\zeta = 0.01$

previous time step. The classic 4th order RK method uses intermediate stages with information and interpolation of excitation at $i-1$ and i time steps and preserves the 4th order accuracy. However, in RTHS, due to the nature that external experimental force from experimental substructure measurement F^E has unit time delay and F_i^E is not available, this computational delay should be considered in numerical integration.

$$
\begin{aligned}
\dot{y} &= h(y,t) \\
k_1 &= h(y_i, t_i) \\
k_2 &= h(y_i + \Delta t/2 k_1, t_i + \Delta t/2) \\
k_3 &= h(y_i + \Delta t/2 k_2, t_i + \Delta t/2) \\
k_4 &= h(y_i + \Delta t k_3, t_i + \Delta t) \\
y_{i+1} &= y_i + \Delta t(b_1 k_1 + b_2 k_2 + b_3 k_3 + b_4 k_4)
\end{aligned}
$$

Reconsider the state space form of reference structure in Eq. 6.24.

$$\dot{X} = A_S X + B_S U \tag{6.33}$$

$$Y = C_S X + D_S U \tag{6.34}$$

where $U = \ddot{x}_g$,

$$A_S = \begin{bmatrix} 0 & 1 \\ -K/M & -C/M \end{bmatrix} B_S = \begin{bmatrix} 0 \\ -1 \end{bmatrix} X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

$$
\begin{aligned}
\dot{X}_{(i+1),1} &= k_1 = A_S X_i + B_S U_i \\
\dot{X}_{(i+1),2} &= k_2 = A_S(X_i + \Delta t/2 k_1) + B_S(U_i + U_{i+1})/2 \\
\dot{X}_{(i+1),3} &= k_3 = A_S(X_i + \Delta t/2 k_2) + B_S(U_i + U_{i+1})/2 \\
\dot{X}_{(i+1),4} &= k_4 = A_S(X_i + \Delta t k_3) + B_S U_{i+1}
\end{aligned}
$$

$$X_{i+1} = X_i + \Delta t/6(k_1 + 2k_2 + 2k_3 + k_4) \tag{6.35}$$

Assuming no earthquake excitation ($U_i = 0$).

$$\dot{X_{(i+1),1}} = k_1 = A_S X_i \tag{6.36}$$

$$\dot{X_{(i+1),2}} = k_2 = A_S(X_i + \Delta t/2k_1) \tag{6.37}$$

$$\dot{X_{(i+1),3}} = k_3 = A_S(X_i + \Delta t/2k_2) \tag{6.38}$$

$$\dot{X_{(i+1),4}} = k_4 = A_S(X_i + \Delta t k_3) \tag{6.39}$$

$$X_{i+1} = X_i + \Delta t/6(k_1 + 2k_2 + 2k_3 + k_4) \tag{6.40}$$

The amplification matrix in RK is:

$$A_{RK} = I + \Delta t/6(A_{S,1} + A_{S,2} + A_{S,3} + A_{S,4}) \tag{6.41}$$

$$A_{S,1} = A_S \tag{6.42}$$

$$A_{S,2} = A_S + A_S \cdot \Delta t/2A_{S,1} \tag{6.43}$$

$$A_{S,3} = A_S + A_S \cdot \Delta t/2A_{S,2} \tag{6.44}$$

$$A_{S,4} = A_S + A_S \cdot \Delta t A_{S,3} \tag{6.45}$$

Consider RTHS with unit delay, solving Eq. 6.3 without earthquake excitation:

$$\dot{X_{(i+1),1}} = k_1 = A_S X_i + B_S U_{R,i}$$
$$\dot{X_{(i+1),2}} = k_2 = A_S(X_i + \Delta t/2k_1) + B_S(U_{R,i} + U_{R,i+1})/2$$
$$\dot{X_{(i+1),3}} = k_3 = A_S(X_i + \Delta t/2k_2) + B_S(U_{R,i} + U_{R,i+1})/2$$
$$\dot{X_{(i+1),4}} = k_4 = A_S(X_i + \Delta t k_3) + B_S U_{R,i+1}$$

The amplification matrix in RTHS is :

$$A_{RKR} = \begin{bmatrix} 0 & I \\ 0 & I \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ A_R & B_R \end{bmatrix} \tag{6.46}$$

where

$$A_R = I + \Delta t/6(A_{R,1} + A_{R,2} + A_{R,3} + A_{R,4})$$

$$A_{R,1} = \begin{bmatrix} 0 & 1 \\ -K^N/M & -C^N/M \end{bmatrix}, B_{R,1} = \begin{bmatrix} 0 & 1 \\ -K^E/M & -C^E/M \end{bmatrix}$$

$$A_{R,2} = A_{R,1} + \Delta t/2A_{R,1}^2 - B_{R,1}/2, B_{R,2} = \Delta t/2A_{R,1}B_{R,1} + B_{R,1}/2$$

$$A_{R,3} = A_{R,1} + \Delta t/2A_{R,2}A_{R,1} - B_{R,1}/2, B_{R,3} = \Delta t/2A_{R,1}B_{R,2} + B_{R,1}/2$$

$$A_{R,4} = A_{R,1} + \Delta tA_{R,3}A_{R,1} - B_{R,1}, B_{R,4} = \Delta tA_{R,1}B_{R,2}$$

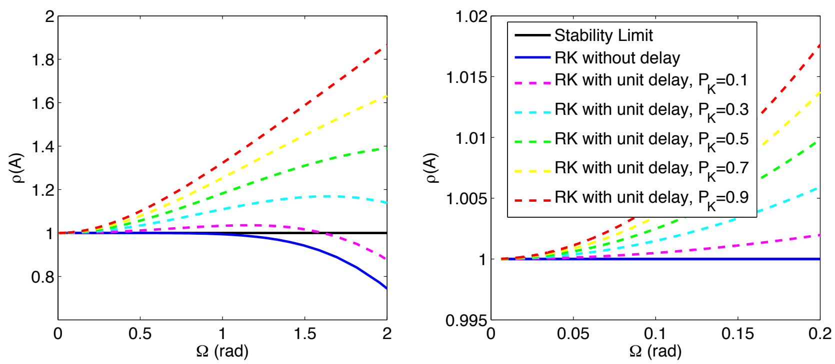


Figure 6.10.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Runge Kutta algorithm, $\zeta = 0$

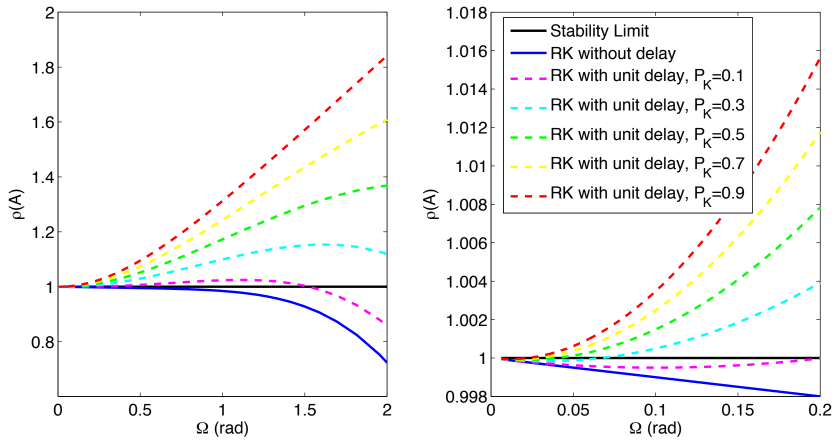


Figure 6.11.: Spectral Radii Comparison between Conventional Numerical Integration and RTHS, Runge Kutta algorithm, $\zeta = 0.01$

6.3 Conclusion

This chapter discussed several widely used integration method in structural dynamics, and compared their stability in the entire system simulation and in RTHS simulation. The results indicate computation delay in RTHS changes the stability characteristics in integration scheme. For an undamped structure, any partition rate results in unstable RTHS loop, and for lightly damped structure, the stable Ω range (equivalent to sampling intervals) is largely reduced to keep the closed loop stable. Also, larger partitioning rates has more restrictions on the selection of the sampling intervals. This chapter extended the conclusion made in [87], when transfer system control is accurate, computational delay in RTHS affects the loop stability.

7. MODIFIED RUNGE-KUTTA INTEGRATION ALGORITHM FOR RTHS

A modified Runge-Kutta (MRK) integration scheme is proposed in this chapter consisting of three computational stages, 1) a pseudo experimental response is calculated by solving the equation of motion using the force measured at both time steps i and $i-1$ and 2) a pseudo feedback force is predicted at the time step $i+1$ using pseudo response from stage 1; and 3), the corrected system response is calculated by solving the equation of motion again using the predicted feedback force for time step $i+1$ combined with the measured experimental force at time step i. This procedure for modifying this algorithm can be applied to other existing integration schemes by repeating stages 1 to 3. This MRK method can be would fall into the class of model-based predictor-corrector integration method modified specifically to compensate for the inherent unit delay in RTHS. The performance of the MRK algorithm, proposed here, is theoretically examined and is shown to have improved the stability and accuracy of RTHS compared to regular integration schemes. For illustrating the capabilities of the proposed algorithm, an example is considered using a lightly damped SDOF structure with different cases of partitioning of stiffness in experimental substructure considered. A moment resisting frame is used as the experimental substructure for experimental verification of the MRK algorithm, with a fixed stiffness partition ratio in the experimental substructure to illustrate the improved accuracy obtained using MRK.

7.1 Modified Runge-Kutta Integration Algorithm

The MRK integration algorithm is built on the classic Runge-Kutta method and it is aimed to minimize the stability issue brought by computational delay described in chapter 6. The mathematical formulation of MRK is derived in this chapter, and a theoretical analysis is performed by comparing stability characteristics of classic Runge Kutta algorithm in reference structure simulation and stability characteristics of MRK in RTHS. First the formulation is derived in single degree of freedom (SDOF) case and then is extended to multiple degrees of freedom (MDOF). It is concluded that, compared to results in chapter 6, MRK largely improves the stability performance of RTHS due to the computational delay.

Recall structural dynamics equation of motion and RTHS dynamics equation of motion:

$$M\ddot{x} + C\dot{x} + Kx = -M\Gamma\ddot{x}_g \tag{7.1}$$

$$M\ddot{x} + C^N\dot{x} + K^N x + \Lambda F^E(x, \dot{x}) = -M\Gamma\ddot{x}_g \tag{7.2}$$

where the superscript $(\ )^N$ and $(\ )^E$ denote the portions in numerical and experimental substructures, and F^E denotes the measured force in experimental substructure.

7.1.1 Single Degree of Freedom Case

Consider an RTHS test, also with assuming perfect tracking in experiment. A flow chart using regular Runge-Kutta integration in computational experiment is shown in Fig. 7.1. It is realized that the time step in the measured experimental force F^E_{i-1} and F^E_i used as external force in computation does not match excitation earthquake record $\ddot{x}_{g,i}$ and $\ddot{x}_{g,i+1}$. To solve this unit step delay in the measured force F_E, the pseudo measured force is predicted after first calculation loop using the calculated response x_{i+1}. Since this next time step response is calculated under delayed experimental force, this response is labeled as pseudo response and written in $\tilde{x}_{i+1}$.

This pseudo measured force is predicted using known specimen initial stiffness and damping:

$$\tilde{F}^E_{i+1} = K^E\tilde{x}_{i+1} + K^E\tilde{\dot{x}}_{i+1} \tag{7.3}$$

Together with measured experimental force at current step, F_i^E and $\tilde{F}_{i+1}^E$ used as external force in computation finally match excitation earthquake record $\ddot{x}_{g,i}$ and $\ddot{x}_{g,i+1}$. Then the second Runge-Kutta iteration are performed to calculate the exact next step response x_{i+1}. A detailed flow chart for using MRK in RTHS is listed in Fig. 7.2

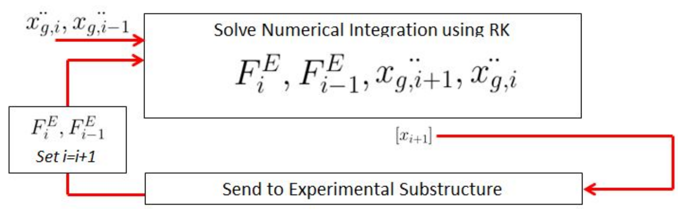


Figure 7.1.: Flow Chart for Conventional Runge-Kutta Integration in RTHS

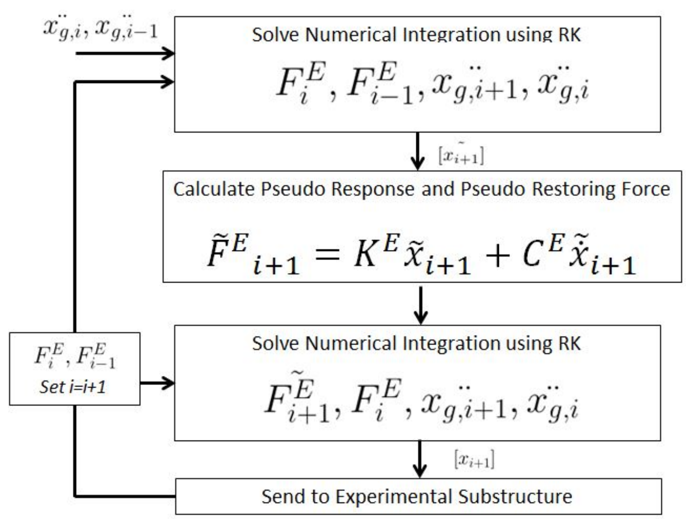


Figure 7.2.: Flow Chart of Modified Runge-Kutta Integration in RTHS

Recall the structural equation of motion solved using the conventional RK:

$$\dot{\tilde{X}}_{(i+1),1} = \tilde{k}_1 = A_S X_i + B_S U_i$$
$$\dot{\tilde{X}}_{(i+1),2} = \tilde{k}_2 = A_S(X_i + \Delta t/2k_1) + B_S(U_i + U_{i+1})/2$$
$$\dot{\tilde{X}}_{(i+1),3} = \tilde{k}_3 = A_S(X_i + \Delta t/2k_2) + B_S(U_i + U_{i+1})/2$$
$$\dot{\tilde{X}}_{(i+1),4} = \tilde{k}_4 = A_S(X_i + \Delta t k_3) + B_S U_{i+1}$$

$$\tilde{X}_{i+1} = X_i + \Delta t/6(\tilde{k}_1 + 2\tilde{k}_2 + 2\tilde{k}_3 + \tilde{k}_4) \tag{7.4}$$

The response of the structure is calculated using conventional Runge-Kutta, using two time step inputs U_i and U_{i+1}. Consider free response case, where earthquake excitation, $\ddot{x}_g = 0$. Then the RK integration depends on two time steps of measured force $U_i = F^E_{i-1}$, $U_{i+1} = F^E_i$. Thus, the exact response using the predicted pseudo force $\tilde{U}_{i+1} = \tilde{F}^E_{i+1}$ and measured force $\tilde{U}_i = F^E_i$ should be less affected by the accuracy of $\tilde{F}^E_{i+1}$.

$$\dot{X}_{(i+1),1} = k_1 = A_S X_i + B_S \tilde{U}_i$$
$$\dot{X}_{(i+1),2} = k_2 = A_S(X_i + \Delta t/2k_1) + B_S(\tilde{U}_i + \tilde{U}_{i+1})/2$$
$$\dot{X}_{(i+1),3} = k_3 = A_S(X_i + \Delta t/2k_2) + B_S(\tilde{U}_i + \tilde{U}_{i+1})/2$$
$$\dot{X}_{(i+1),4} = k_4 = A_S(X_i + \Delta t k_3) + B_S \tilde{U}_{i+1}$$

$$X_{i+1} = X_i + \Delta t/6(k_1 + 2k_2 + 2k_3 + k_4) \tag{7.5}$$

Write the MRK integration for RTHS in a recursive manner:

$$A_{MRKR} = \begin{bmatrix} 0 & I \\ A_R^2 - B_R & -A_R B_R \end{bmatrix} \tag{7.6}$$

where

$$A_R = I + \Delta t/6(A_{R,1} + A_{R,2} + A_{R,3} + A_{R,4})$$

$$B_R = \Delta t/6(B_{R,1} + B_{R,2} + B_{R,3} + B_{R,4})$$

$$A_{R,1} = \begin{bmatrix} 0 & 1 \\ -K^N/M & -C^N/M \end{bmatrix}, B_{R,1} = \begin{bmatrix} 0 & 1 \\ -K^E/M & -C^E/M \end{bmatrix}$$

$$A_{R,2} = A_{R,1} + \Delta t/2A_{R,1}^2 - B_{R,1}/2, B_{R,2} = \Delta t/2A_{R,1}B_{R,1} + B_{R,1}/2$$

$$A_{R,3} = A_{R,1} + \Delta t/2A_{R,2}A_{R,1} - B_{R,1}/2, B_{R,3} = \Delta t/2A_{R,1}B_{R,2} + B_{R,1}/2$$

$$A_{R,4} = A_{R,1} + \Delta tA_{R,3}A_{R,1} - B_{R,1}, B_{R,4} = \Delta tA_{R,1}B_{R,2}$$

Recall the amplification matrix in RK is:

$$A_{RK} = I + \Delta t/6(A_{S,1} + A_{S,2} + A_{S,3} + A_{S,4}) \tag{7.7}$$

$$A_{S,1} = A_S \tag{7.8}$$

$$A_{S,2} = A_S + A_S \cdot \Delta t/2A_{S,1} \tag{7.9}$$

$$A_{S,3} = A_S + A_S \cdot \Delta t/2A_{S,2} \tag{7.10}$$

$$A_{S,4} = A_S + A_S \cdot \Delta tA_{S,3} \tag{7.11}$$

where,

$$A_S = \begin{bmatrix} 0 & 1 \\ -K/M & -C/M \end{bmatrix}$$

The spectral radii for the MRK method are plotted in FIG. 7.3 and 7.4. It can be concluded that, compared to results in chapter 6, MRK largely improves the stability performance of RTHS in the presence of computational delay. This algorithm can be considered as a model-based predictor-corrector method, which compensates specifically the inherent unit delay in RTHS. However, the effect of model accuracy (K^E, C^E) and specimen nonlinear dynamics needs further analysis.

7.1.2 Multi-Degrees of Freedom Case

For MDOF cases, two different methods are considered. One is the direct numerical integration approach and the other one is a modal analysis method using modal transformations.

The state space form for MDOF reference systems:

$$\dot{X} = A_M X + B_M U \tag{7.12}$$

$$Y = C_M X + D_M U \tag{7.13}$$

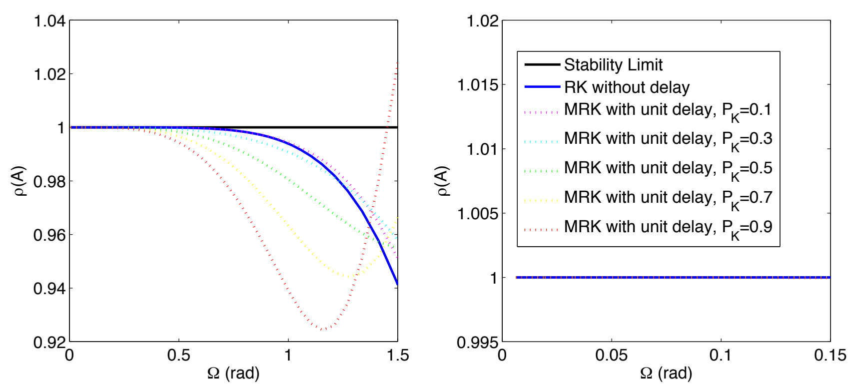


Figure 7.3.: Spectral Radii Comparison between Entire Simulation and RTHS, Modified Runge Kutta algorithm, $\zeta = 0$

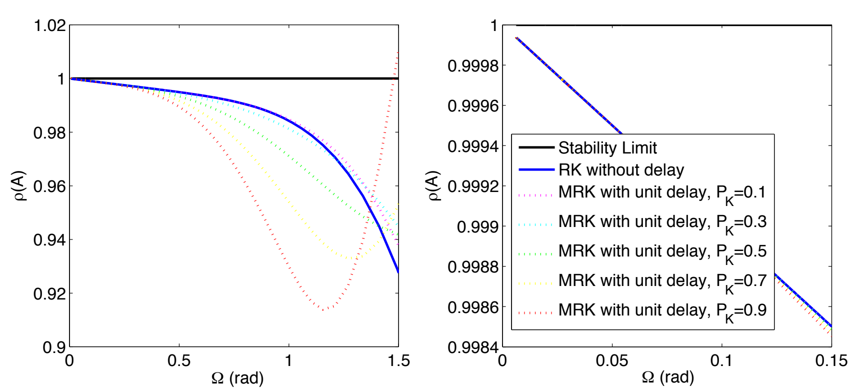


Figure 7.4.: Spectral Radii Comparison between Entire Simulation and RTHS, Modified Runge Kutta algorithm, $\zeta = 0.01$

where $U = \ddot{x}_g$,

$$A_M = \begin{bmatrix} 0 & 1 \\ -M^{-1}K & -M^{-1}C \end{bmatrix} B_M = \begin{bmatrix} 0 \\ -\Gamma \end{bmatrix} X = \begin{bmatrix} D \\ \dot{D} \end{bmatrix}$$

where, D is the vector for MDOF displacement and $\dot{D}$ is vector of MDOF system velocity, Γ is $N \times 1$ vector of one, N is the number of all freedom. M, C, K are global mass, damping, stiffness matrices for reference structure.

Use classic RK for numerical integration fir reference structure:

$$\begin{aligned}
\dot{X}_{(i+1),1} &= k_1 = A_M X_i + B_M U_i \\
\dot{X}_{(i+1),2} &= k_2 = A_M(X_i + \Delta t/2 k_1) + B_M(U_i + U_{i+1})/2 \\
\dot{X}_{(i+1),3} &= k_3 = A_M(X_i + \Delta t/2 k_2) + B_M(U_i + U_{i+1})/2 \\
\dot{X}_{(i+1),4} &= k_4 = A_M(X_i + \Delta t k_3) + B_M U_{i+1}
\end{aligned}$$

$$\dot{X}_{i+1} = X_i + \Delta t/6(k_1 + 2k_2 + 2k_3 + k_4) \tag{7.14}$$

Similar as SDOF case, the form using direct integration method is straight forward.

$$F^E = K^E D_J + K^E \dot{D}_J \tag{7.15}$$

where $J = j_1, j_2, ...j_n$ indicates the global degree freedom in reference structural matrices and C^E and K^E are local damping and stiffness matrices for experimental substructure in RTHS.

The state space form for MDOF RTHS systems:

$$\dot{X} = A_{M,R} X + B_{M,R} U \tag{7.16}$$

where $U = [\ddot{x}_g, F^E]$

$$A_{M,R} = \begin{bmatrix} 0 & 1 \\ -M^{-1}K^N & -M^{-1}C^N \end{bmatrix} B_{M,R} = \begin{bmatrix} 0 & 0 \\ -\Gamma & -M^{-1}\Lambda \end{bmatrix} X = \begin{bmatrix} D \\ \dot{D} \end{bmatrix}$$

where, C^N, K^N are global mass, damping, stiffness matrices for numerical substructure in RTHS, Λ is the Lagrange multiplier ($N \times 1$ vector) has the form of all zeros and only $\Lambda_{j_n} = 1$, j_n indicates the DOFs linked with experimental substructure.

For the first step, calculate pseudo response:

$$\begin{aligned}
\tilde{\dot{X}}_{(i+1),1} &= \tilde{k}_1 = A_{M,R}X_i + B_{M,R}U_i \\
\tilde{\dot{X}}_{(i+1),2} &= \tilde{k}_2 = A_{M,R}(X_i + \Delta t/2k_1) + B_{M,R}(U_i + U_{i+1})/2 \\
\tilde{\dot{X}}_{(i+1),3} &= \tilde{k}_3 = A_{M,R}(X_i + \Delta t/2k_2) + B_{M,R}(U_i + U_{i+1})/2 \\
\tilde{\dot{X}}_{(i+1),4} &= \tilde{k}_4 = A_{M,R}(X_i + \Delta t k_3) + B_{M,R}U_{i+1}
\end{aligned}$$

$$\tilde{X}_{i+1} = X_i + \Delta t/6(\tilde{k}_1 + 2\tilde{k}_2 + 2\tilde{k}_3 + \tilde{k}_4) \tag{7.17}$$

Calculate the pseudo experimental force:

$$\tilde{F}^E_{i+1} = K^E\tilde{D}_{i+1,J} + K^E\tilde{\dot{D}}_{i+1,J} \tag{7.18}$$

Calculate the exact response, with $\tilde{U}_i = [\ddot{x}_{g,i}, \tilde{F}^E_i]$:

$$\begin{aligned}
\dot{X}_{(i+1),1} &= k_1 = A_{M,R}X_i + B_{M,R}\tilde{U}_i \\
\dot{X}_{(i+1),2} &= k_2 = A_{M,R}(X_i + \Delta t/2k_1) + B_{M,R}(\tilde{U}_i + \tilde{U}_{i+1})/2 \\
\dot{X}_{(i+1),3} &= k_3 = A_{M,R}(X_i + \Delta t/2k_2) + B_{M,R}(\tilde{U}_i + \tilde{U}_{i+1})/2 \\
\dot{X}_{(i+1),4} &= k_4 = A_{M,R}(X_i + \Delta t k_3) + B_{M,R}\tilde{U}_{i+1}
\end{aligned}$$

$$X_{i+1} = X_i + \Delta t/6(k_1 + 2k_2 + 2k_3 + k_4) \tag{7.19}$$

In the reference structure, define a modal matrix Φ in which each column is an eigenvector of the matrix.

$$\Phi = \begin{bmatrix} \varphi_1 & \varphi_2 & \varphi_3 & ... & \varphi_N \end{bmatrix} \tag{7.20}$$

Consider free vibration response for MDOF system:

$$\begin{aligned} M\ddot{X} + KX &= 0 \\ X &= \varphi cos(\omega t - \theta) \\ \ddot{X} &= -\varphi\omega^2 \cdot cos(\omega t - \theta) \\ [-M\omega^2 + K]\varphi \cdot cos(\omega t - \theta) &= 0 \end{aligned}$$

$$\omega^2 M\varphi = K\varphi \tag{7.21}$$

where ω's are known as the natural frequencies of the system, for the i^{th} natural frequency:

$$[-\omega_{(i)}^2 M + K]\varphi_{(i)} = 0 \tag{7.22}$$

The solution $\varphi_{(i)}$ represents the i^{th} mode shape corresponding to the i^{th} natural frequency.

$$[M] = \Phi^T M\Phi \tag{7.23}$$

$$[K] = \Phi^T K\Phi \tag{7.24}$$

where $[M]$ is the modal mass matrix and $[K]$ is the modal stiffness matrix. Using modal transformation, assuming $X = \Phi q$, where q is the system response in modal coordinate.

$$M\ddot{X} + KX = -M\Gamma\ddot{x}_g \tag{7.25}$$

$$M\Phi\ddot{q} + K\Phi q = -M\Gamma\ddot{x}_g \tag{7.26}$$

$$\Phi^T M\Phi\ddot{q} + \Phi^T K\Phi q = -\Phi^T M\Gamma\ddot{x}_g \tag{7.27}$$

Substitute Eqs. 7.23 and 7.24 into Eq. 7.27, then the modal form for structural response is:

$$[M]\ddot{q} + [K]q = -\Phi^T M\Gamma\ddot{x}_g \tag{7.28}$$

It can be seen that $[M]$ and $[K]$ can be decoupled for each modal degree of freedom in to several SDOF systems, similarly, if structural damping matrix C is proportional to structural mass matrix and stiffness matrix M and K, such as $C = \alpha_M M + \alpha_K K$. Then the structure with damping can be written as:

$$[M]\ddot{q} + [C]\dot{q} + [K]q = -\Phi^T M\Gamma\ddot{x}_g \tag{7.29}$$

where $[C] = \Phi^T C\Phi$. Consider RTHS form for structural response:

$$M\ddot{x} + C^N\dot{x} + K^N x = -M\Gamma\ddot{x}_g - \Lambda F^E(x, \dot{x}) \tag{7.30}$$

Assume $X = \Phi q$. substitute into 7.30,

$$M\ddot{X} + KX = -M\Gamma\ddot{x}_g$$

$$M\Phi\ddot{q} + K\Phi q = -M\Gamma\ddot{x}_g$$

$$\Phi^T M\Phi\ddot{q} + \Phi^T C^N\Phi\dot{q} + \Phi^T K^N\Phi q = -\Phi^T M\Gamma\ddot{x}_g - \Phi^T\Lambda F^E(x, \dot{x})$$

$$[M]\ddot{q} + [C^N]\dot{q} + [K^N]q = -\Phi^T M\Gamma\ddot{x}_g - \Phi^T\Lambda F^E(x, \dot{x})$$

Each modal degree freedom can be viewed as a SDOF system and solved using a SDOF numerical integration method. The stability and accuracy of the MDOF system is determined by the highest natural frequency ω_N and the sampling interval Δt.

7.1.3 Robustness analysis of MRK in RTHS

The MRK algorithm utilizes a psuedo measured force from Eq. 7.3 using a high fidelity model of the experimental substructure, i.e. no modeling error in K^E, C^E. Therefore, the effect of modeling error in the experimental substructure and the effect of specimen nonlinearity on the performance of this method should be examined. Assuming that the true stiffness of the experimental substructure is K_t^E and modeling error only exists in the specimen stiffness (with estimated stiffness K_{est}^E), the equivalent stiffness can be written as $K = K_{est}^N + K_{est}^E$ and the amplification matrix for MRK with modeling error in RTHS can be derived as before.

As two illustrative examples, Fig. 7.5 and 7.6 represent the most critical cases of partitioning ratio 90% ($P_K = 0.9$) and 0% respectively, with 1% damping ratio in both systems. The investigated modeling errors are $\pm(10\%, 20\%, 30\%, 40\%)$ of the true specimen stiffness K_t^E. Results with stiffness over-estimation and under-estimation distribute evenly about the reference line (MRK with no error). It is observed that the under-estimation of specimen stiffness ($K_{est}^E < K_t^E$) potentially may destabilize the RTHS closed loop when there is no damping in the system. However, the results are still significantly better compared to other conventional integration schemes (c.f. Fig. 6.4). For lightly damped system, the stability of the RTHS is preserved and the performance is similar to MRK with no error. Inevitably, solution accuracy is also largely affected by modeling error. However, this may not be due to MRK itself, but because of the existence of modeling error in general.

The effect of specimen nonlinearity is also studied. Up to 80% stiffness degradation (equivalent system with $K = K_{est}^N + K_{est}^E$ with $K_t^E = 0.2K_{est}^E$) is analyzed for 90% partitioning ratio ($P_K = 0.9$). Fig. 7.7 shows the spectral radii for stiffness degrading

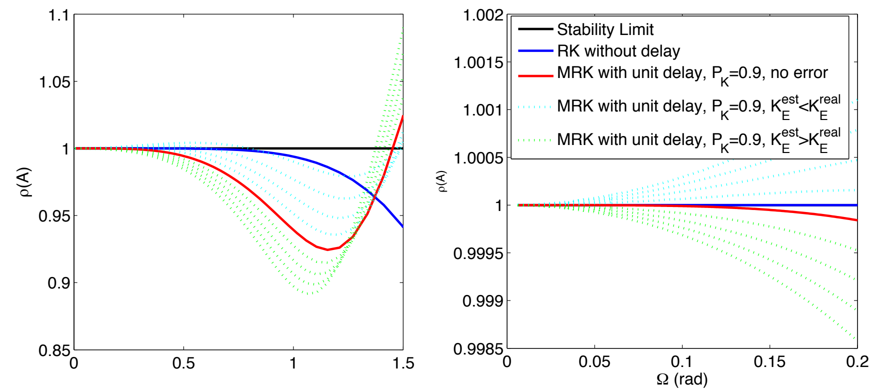


Figure 7.5.: Spectral Radii comparison for MRK with Modeling Error, Damping $\zeta = 0$

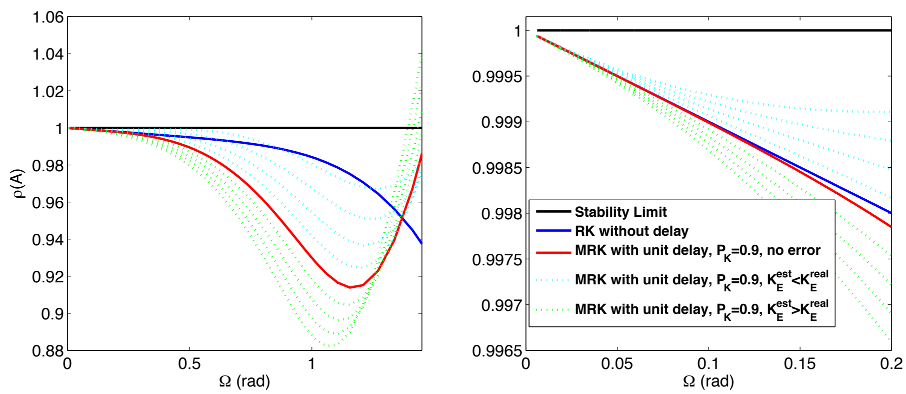


Figure 7.6.: Spectral Radii comparison for MRK with Modeling Error, Damping $\zeta = 0.01$

(stiffness over estimation), for a system with no damping. It is evident that the RTHS closed loop stability is still preserved even with 80% stiffness degradation.

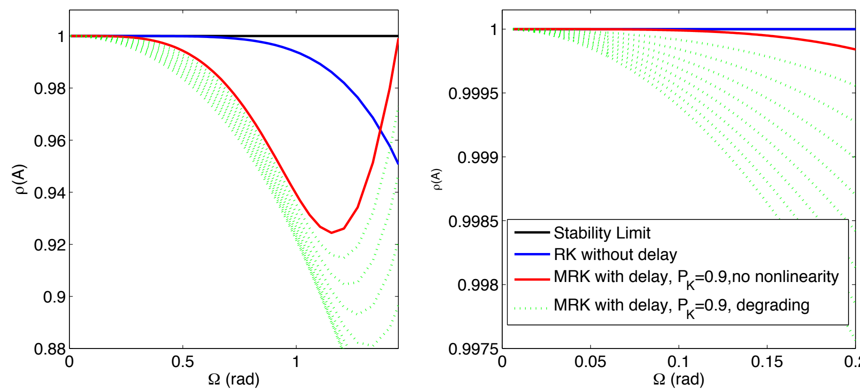


Figure 7.7.: Spectral Radii Comparison for MRK with Stiffness Degrading, Damping $\zeta = 0$.

7.2 Numerical Examples

This chapter provides several numerical examples to understand the performance of the MRK integration scheme in RTHS. Three cases are studied, including A) undamped SDOF system with stiffness partitioned into numerical and experimental substructures; B) lightly damped SDOF system with stiffness partitioned into numerical and experimental substructures; and C), one 9 DOF structure with a single DOF experimental substructure.

For the SDOF, the general numerical examples are simulated in MATLAB ® SIMULINK as shown in Fig. 7.8. In Numerical Sub block, different integration algorithm are evaluated. MRK performance are compared with traditional integration methods discussed in chapter 6, including: central difference method and average acceleration method using Newmark-β, CR method, discrete state space method and classical Runge-Kutta method. In the Experimental Sub block, since integration stability and accuracy may be affected by partitioning ratios in RTHS, different partition ratios are studied and simulated. The unit delay block is added after force generated in experimental substructure to break the algebraic loop and also to simulate the computational delay in real test scenario.

For the MDOF case, the effects of a) different sampling rates, b) different partitioning ratios, and c) different partitioning DOFs are analyzed to understand MRK integration performance in MDOF RTHS.

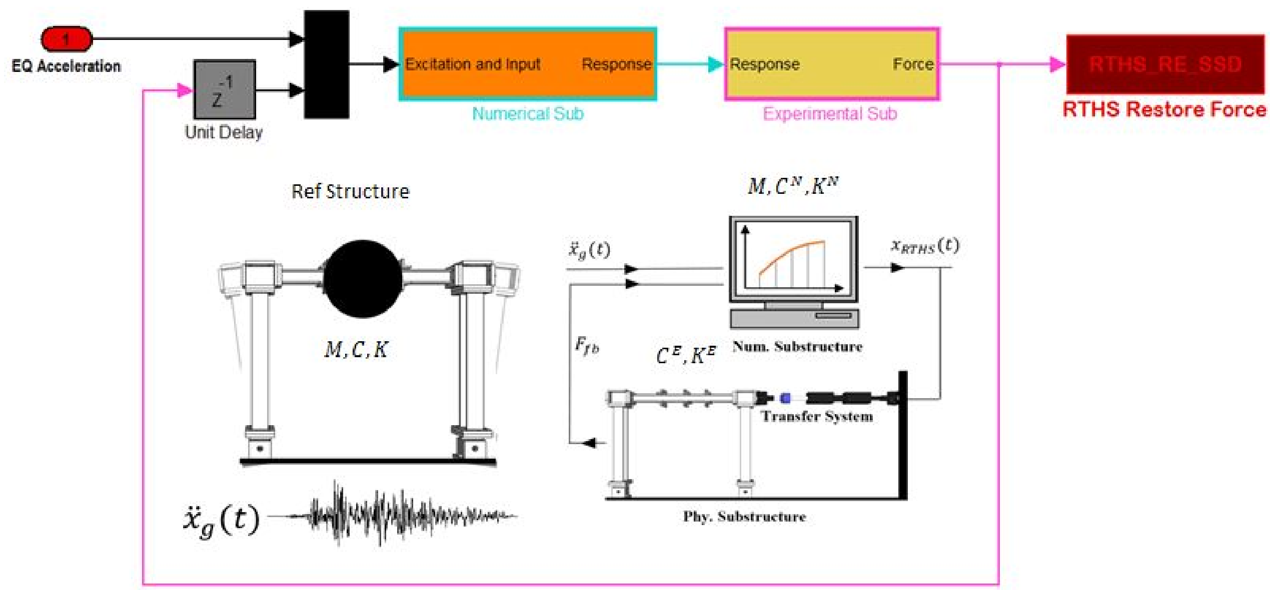


Figure 7.8.: Numerical Example: SDOF RTHS

7.2.1 Single DOF RTHS

Schematic diagram for SDOF RTHS simulation is shown in Fig. 7.8. The reference structure has $m = 155.6$ Kg, $k = 1.05 \times 10^5$ N/m, damping ratio is 0.00 (undamped structure) and 0.01 (lightly damped structure) same as discussed in chapter. 6. Sampling frequency are chosen at 1024 Hz and 512 Hz.

The accuracy of each integration algorithm is quantified using RMS error indicator. Introduce root mean square error (RMS) indicator:

$$E_{RMS} = \sqrt{\frac{\Sigma_l (D_{RTHS} - D_{Ref})^2}{l}} / \sqrt{\frac{\Sigma_l D_{Ref}^2}{l}} = RMS(D_e)/RMS(D_{Ref}) \qquad (7.31)$$

Performance for different integration algorithm in different RTHS cases are listed from Table 7.1   7.4, where 'UNS' in the table indicates this simulation is unstable.

Results from Table 7.1 matches the analytic plots (Fig. 6.2, 6.4, 6.6, 6.8, 6.10) in Chapter 6. For an undamped reference structure, the unit delay can destroy the stability of the entire RTHS.

For a lightly damped structure ($\zeta = 0.01$), when smaller partition ratio is chosen (0.1, 0.3), the stability is preserved, however, the accuracy is affected using other traditional algorithms. Sampling frequency can affect the accuracy if compare Table 7.2 and 7.4.

The MRK algorithm, in all scenario in these two numerical examples, matches well with reference solution. The accuracy is affected by partition rate, damping ratio and sampling rate as well. The worst performance among all scenarios are undamped reference structure with 90% stiffness in experimental substructure, sampled at 512 Hz. This finding is demonstrated earlier in Fig. 7.3 and 7.4.

Table 7.1.: RMS Error for All Integration Method in SDOF RTHS, 512 Hz, Damping $\zeta = 0$

Partition Rate	CD-NB	AA-NB	CR	DSS	RK	MRK
10 %	UNS	UNS	UNS	UNS	UNS	0.0121
30 %	UNS	UNS	UNS	UNS	UNS	0.0323
50 %	UNS	UNS	UNS	UNS	UNS	0.0530
70 %	UNS	UNS	UNS	UNS	UNS	0.0737
90 %	UNS	UNS	UNS	UNS	UNS	0.0944

Table 7.2.: RMS Error for All Integration Method in SDOF RTHS, 512 Hz, Damping $\zeta = 1\%$

Partition Rate	CD-NB	AA-NB	CR	DSS	RK	MRK
10 %	0.1838	0.1833	0.1950	0.1833	0.1833	0.0007
30 %	0.9485	0.9419	0.9920	0.9419	0.9506	0.0020
50 %	UNS	UNS	UNS	UNS	UNS	0.0032
70 %	UNS	UNS	UNS	UNS	UNS	0.0045
90 %	UNS	UNS	UNS	UNS	UNS	0.0057

Table 7.3.: RMS Error for All Integration Method in SDOF RTHS, 1024 Hz, Damping $\zeta = 0$

Partition Rate	CD-NB	AA-NB	CR	DSS	RK	MRK
10 %	UNS	UNS	UNS	UNS	UNS	0.0027
30 %	UNS	UNS	UNS	UNS	UNS	0.0080
50 %	UNS	UNS	UNS	UNS	UNS	0.0133
70 %	UNS	UNS	UNS	UNS	UNS	0.0185
90 %	UNS	UNS	UNS	UNS	UNS	0.0238

Table 7.4.: RMS Error for All Integration Method in SDOF RTHS, 1024 Hz, Damping $\zeta = 1\%$

Partition Rate	CD-NB	AA-NB	CR	DSS	RK	MRK
10 %	0.0848	0.0836	0.0837	0.0836	0.0835	0.0002
30 %	0.3069	0.3045	0.3049	0.3045	0.3049	0.0005
50 %	0.6645	0.6587	0.6603	0.6587	0.6600	0.0008
70 %	UNS	UNS	UNS	UNS	UNS	0.0011
90 %	UNS	UNS	UNS	UNS	UNS	0.0015

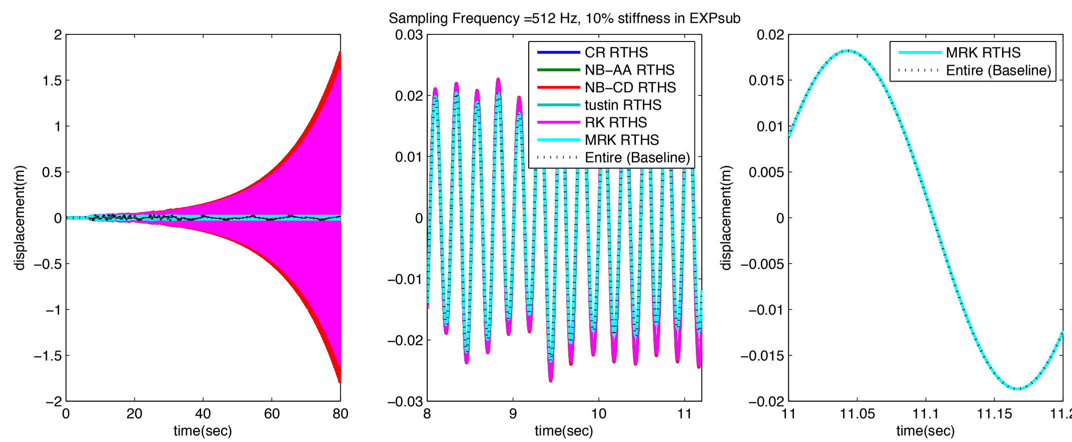


Figure 7.9.: SDOF Numerical RTHS, Partitioning Ratio=10%, Sampling 512 Hz, Undamped

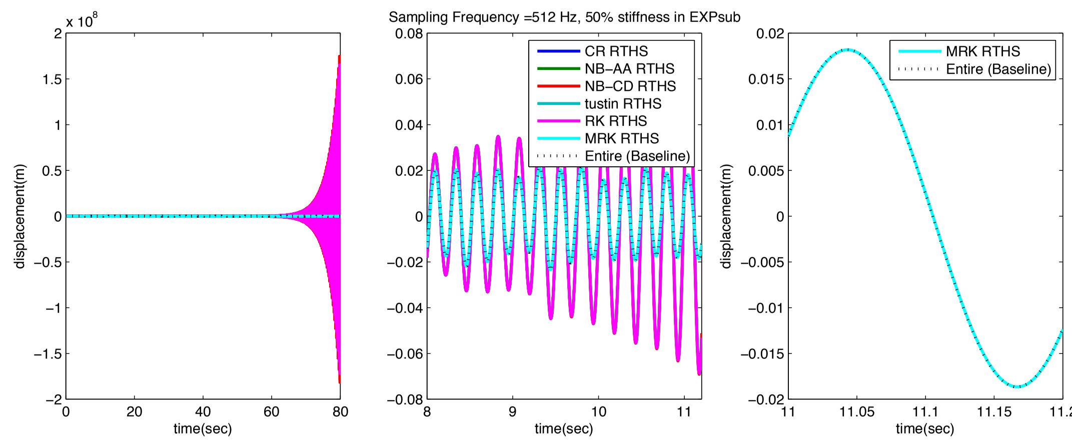


Figure 7.10.: SDOF Numerical RTHS, Partitioning Ratio=50%, Sampling 512 Hz, Undamped

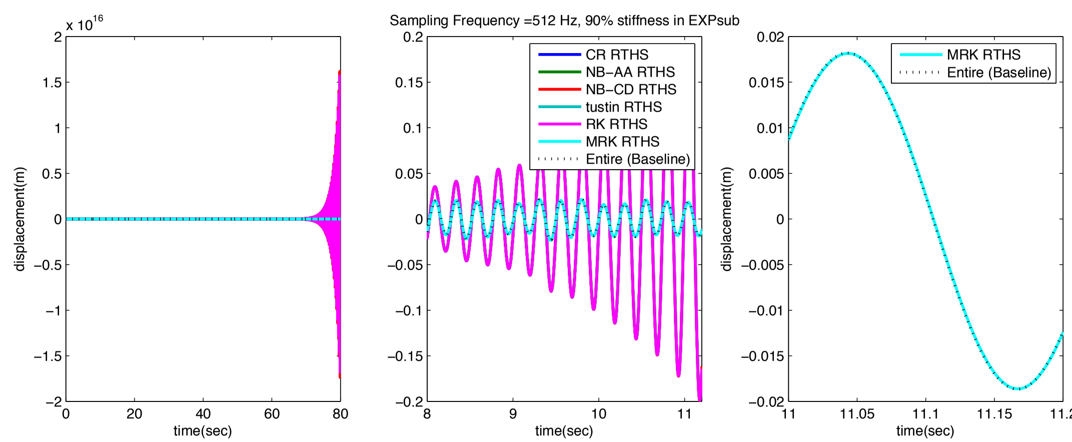


Figure 7.11.: SDOF Numerical RTHS, Partitioning Ratio=90%, Sampling 512 Hz, Undamped

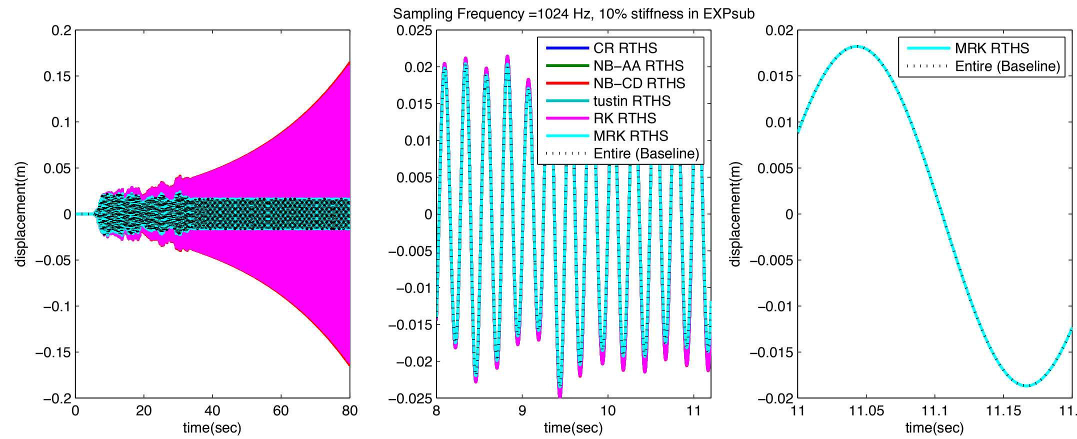


Figure 7.12.: SDOF Numerical RTHS, Partitioning Ratio=10%, Sampling 1024 Hz, Undamped

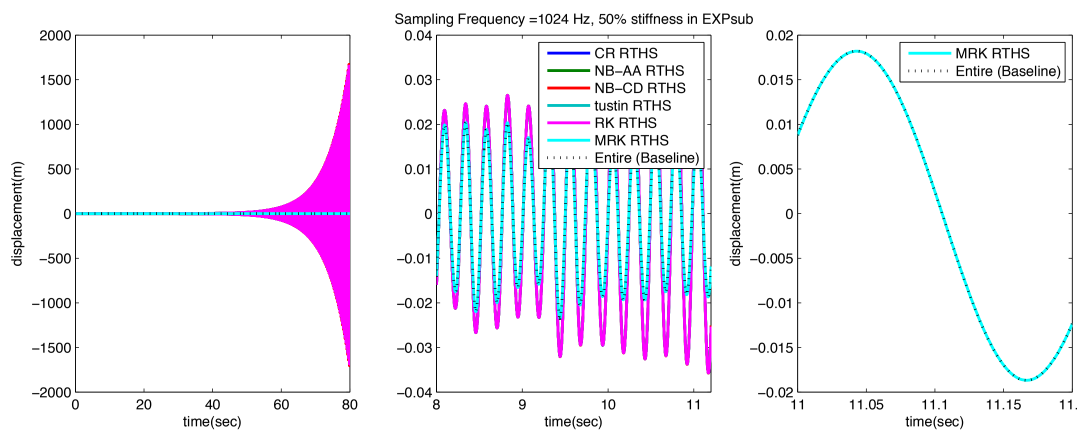


Figure 7.13.: SDOF Numerical RTHS, Partitioning Ratio=50%, Sampling 1024 Hz, Undamped

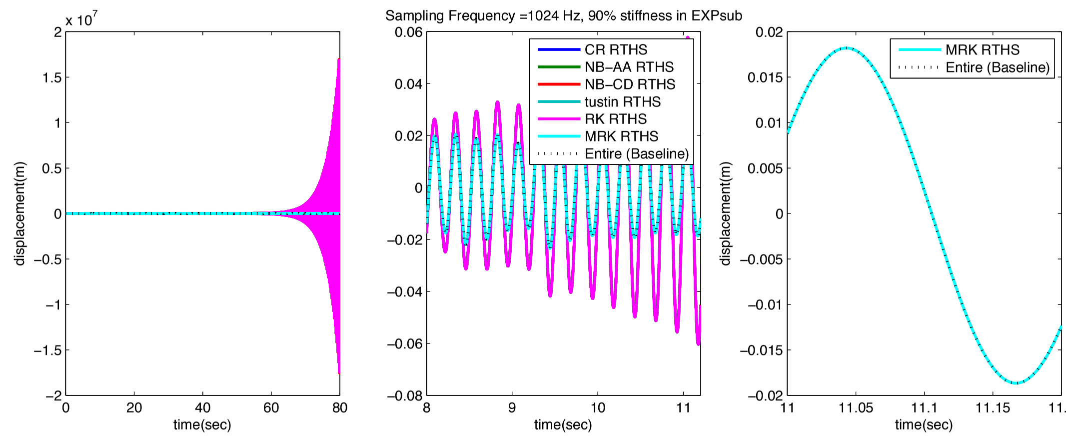


Figure 7.14.: SDOF Numerical RTHS, Partitioning Ratio=90%, Sampling 1024 Hz, Undamped

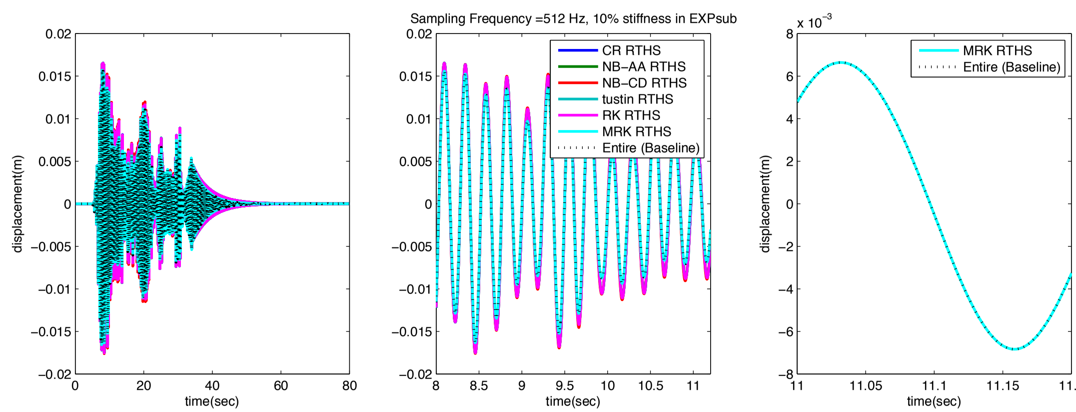


Figure 7.15.: SDOF Numerical RTHS, Partitioning Ratio=10%, Sampling 512 Hz, Damping ratio=0.01

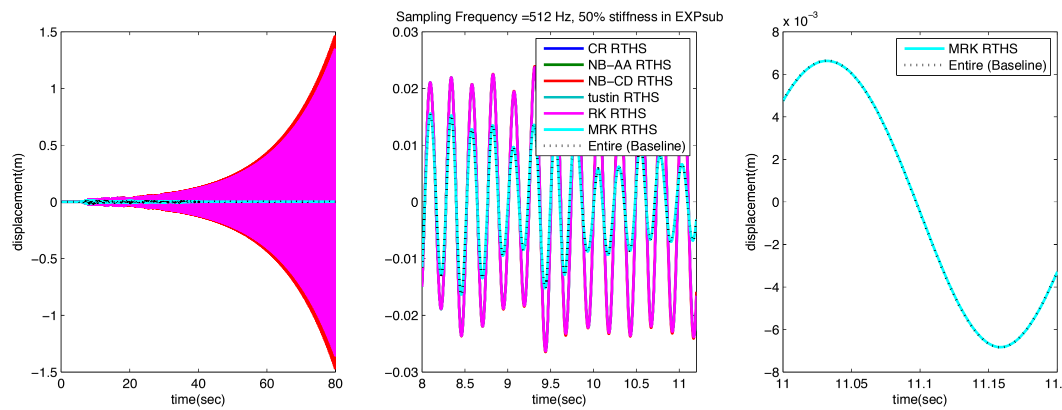


Figure 7.16.: SDOF Numerical RTHS, Partitioning Ratio=50%, Sampling 512 Hz, Damping Ratio=0.01

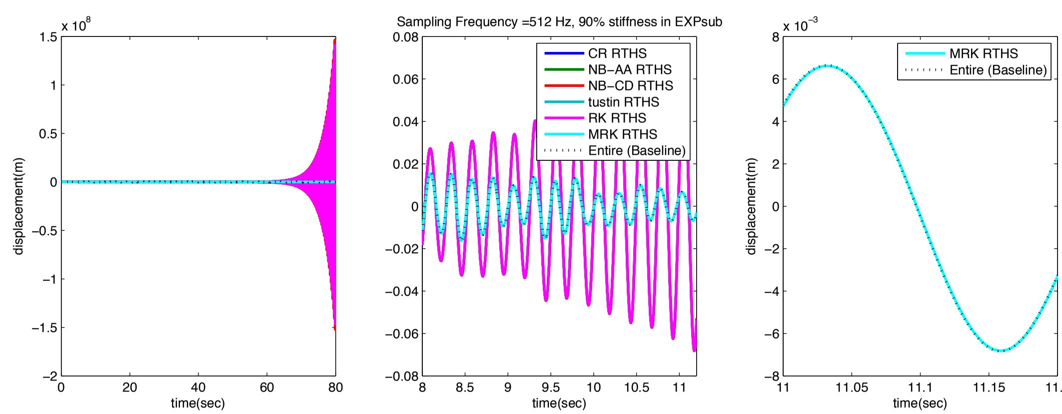


Figure 7.17.: SDOF Numerical RTHS, Partitioning Ratio=90%, Sampling 512 Hz, Damping Ratio=0.01

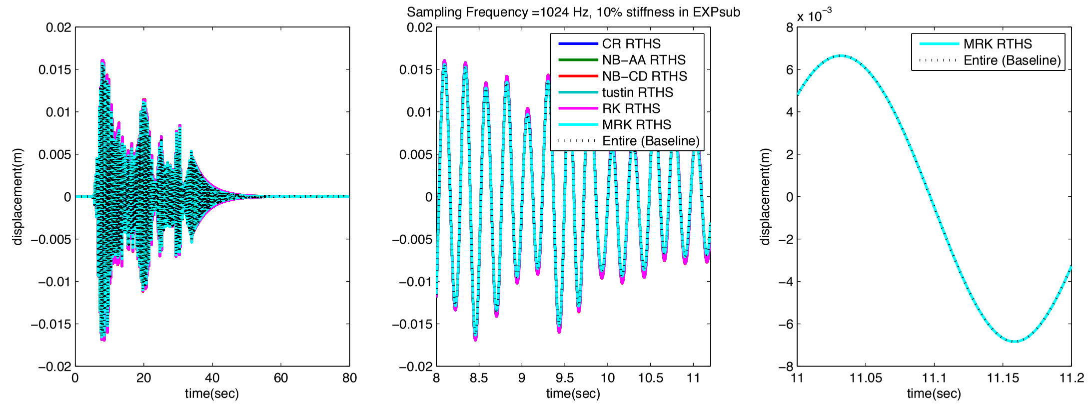


Figure 7.18.: SDOF Numerical RTHS, Partitioning Ratio=10%, Sampling 1024 Hz, Damping Ratio=0.01

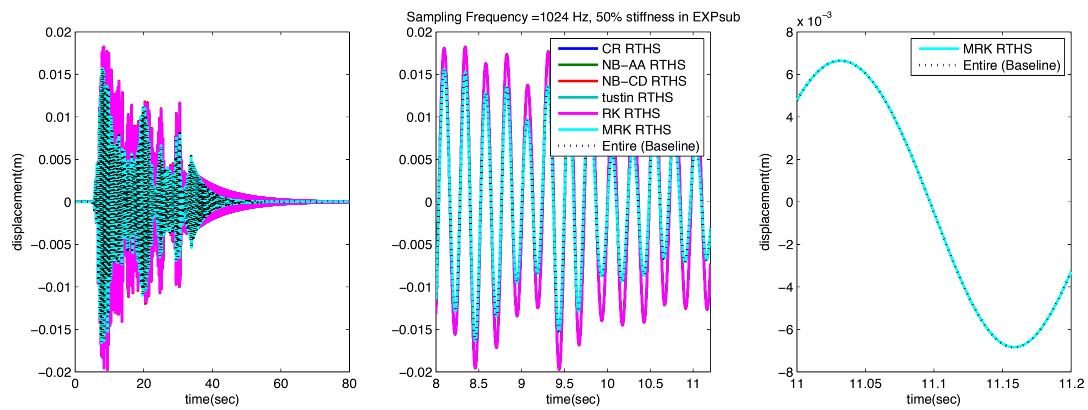


Figure 7.19.: SDOF SDOF RTHS numerical example: partition rate=50%, sampling 1024 Hz, damping ratio=0.01

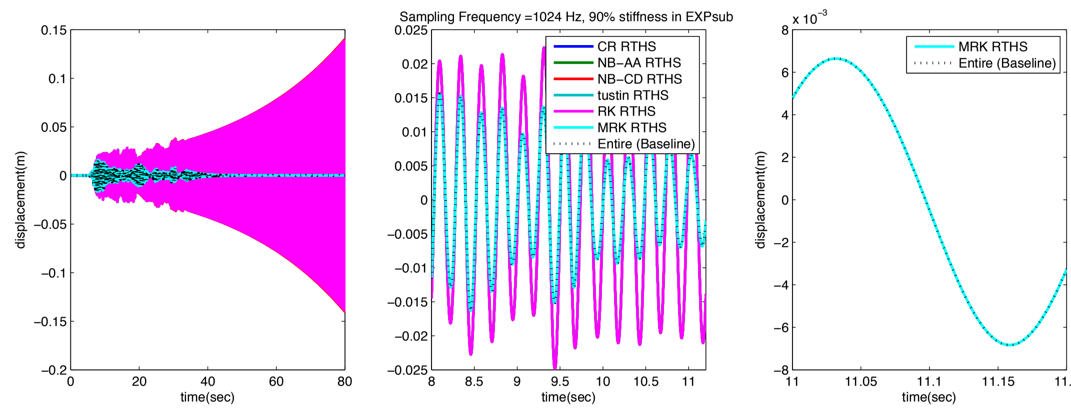


Figure 7.20.: SDOF Numerical RTHS, Partitioning Ratio=90%, Sampling 1024 Hz, Damping Ratio=0.01

7.2.2 Multiple DOFs RTHS

For MDOF example, the reference structure is the 9 story moment resisting frame picked from benchmark control problems for seismically excited nonlinear buildings [96]. The structure is 45.73 m by 45.73 m in plan (5 bays in both N-S and E-W directions) and 37.19 m in elevation. Detail information of this benchmark structure is described in Fig. 7.21. Originally, the finite element model of the benchmark structure has 198 DOFs, such model has been condensed into lumped mass shear model of 9 DOFs [97].

The condensed model has M, C, K matrices as following (units in kg, $N \cdot s/m$ and N/m):

$$M = \begin{bmatrix} 1010 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 989 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 989 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 989 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 989 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 989 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 989 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 989 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1070 \end{bmatrix} \times 10^3$$

$$C = \begin{bmatrix} 0.952 & -0.387 & -0.074 & -0.029 & -0.014 & -0.008 & -0.005 & -0.003 & -0.003 \\ -0.387 & 1.072 & -0.347 & -0.066 & -0.026 & -0.014 & -0.008 & -0.005 & -0.004 \\ -0.074 & -0.347 & 1.021 & -0.329 & -0.063 & -0.026 & -0.013 & -0.008 & -0.006 \\ -0.029 & -0.066 & -0.329 & 0.967 & -0.313 & -0.061 & -0.024 & -0.013 & -0.010 \\ -0.014 & -0.026 & -0.063 & -0.313 & 0.909 & -0.295 & -0.054 & -0.022 & -0.015 \\ -0.008 & -0.014 & -0.026 & -0.061 & -0.295 & 0.821 & -0.249 & -0.049 & -0.026 \\ -0.005 & -0.008 & -0.013 & -0.024 & -0.054 & -0.249 & 0.736 & -0.241 & -0.060 \\ -0.003 & -0.005 & -0.008 & -0.013 & -0.022 & -0.049 & -0.241 & 0.674 & -0.255 \\ -0.003 & -0.004 & -0.006 & -0.010 & -0.015 & -0.026 & -0.060 & -0.255 & 0.458 \end{bmatrix} \times 10^6$$

$$K = \begin{bmatrix} 6.225 & -4.35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -4.354 & 8.627 & -4.273 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -4.273 & 7.809 & -3.535 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -3.535 & 6.980 & -3.444 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -3.444 & 6.448 & -3.003 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -3.003 & 5.087 & -2.084 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2.084 & 3.912 & -1.827 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1.827 & 3.493 & -1.671 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.671 & 1.671 \end{bmatrix} \times 10^8$$

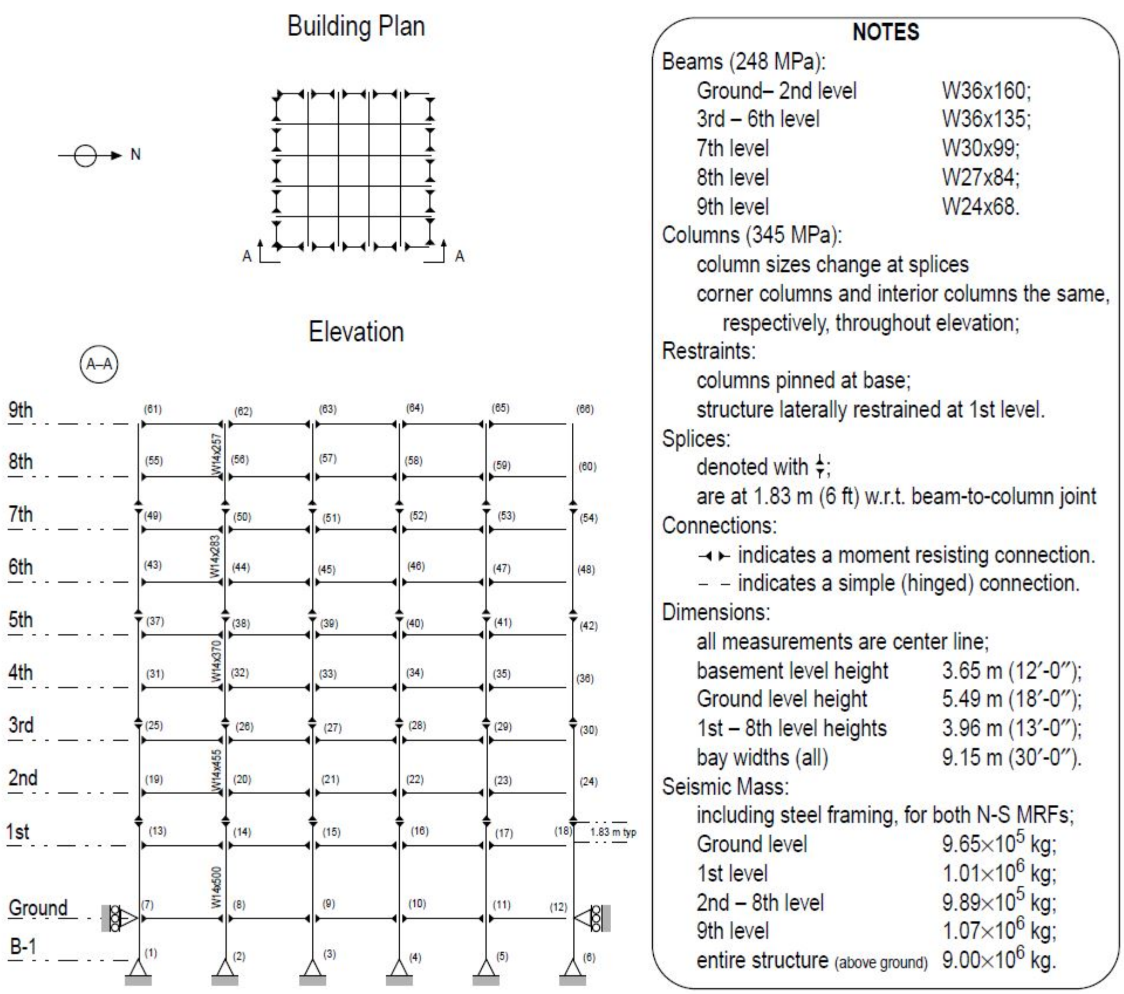


Figure 7.21.: 9 Story Reference Building in Benchmark Control, N-S MRF

The experimental substructure in MDOF case is chosen from one column to 6 columns (is equivalent to one bay) in the section plan (36 columns in each floor), this is equivalent to partition rate from 1/36 (2.78 %) to 1/6 (16.67 %). The RTHS numerical examples in this section evaluates MRK performance for different partition rates and partitioned at different floors. Performance for MRK in 9 DOF structure is listed in Table 7.5 and 7.6 for sampling rate 512 Hz and 1024 Hz, respectively. It is illustrated that for partition rate up to 16.67%, RMS error in MRK is smaller than 0.02% for sampling rate of 512 Hz and 0.01% in for sampling rate of 512 Hz. The effect of partition rate (1/36 to 1/6) does not occur significantly in this example.

Table 7.5.: RMS Error for MRK in 9DOF RTHS, 512 Hz

Partition DOF	Partition 1/36	Partition 1/18	Partition 1/12	Partition 1/9	Partition 5/36	Partition 1/6
Floor 1	0.0331%	0.0327%	0.0324%	0.0320%	0.0316%	0.0313%
Floor 2	0.0294%	0.0286%	0.0278%	0.0270%	0.0263%	0.0255%
Floor 3	0.0252%	0.0244%	0.0236%	0.0228%	0.0221%	0.0213%
Floor 4	0.0203%	0.0196%	0.0190%	0.0183%	0.0178%	0.0173%
Floor 5	0.0157%	0.0152%	0.0147%	0.0142%	0.0139%	0.0137%
Floor 6	0.0113%	0.0107%	0.0102%	0.0098%	0.0094%	0.0093%
Floor 7	0.0090%	0.0082%	0.0076%	0.0070%	0.0066%	0.0063%
Floor 8	0.0123%	0.0116%	0.0111%	0.0106%	0.0103%	0.0101%
Floor 9	0.0165%	0.0162%	0.0158%	0.0155%	0.0151%	0.0148%

Table 7.6.: RMS Error for MRK in 9DOF RTHS, 1024 Hz

Partition DOF	Partition 1/36	Partition 1/18	Partition 1/12	Partition 1/9	Partition 5/36	Partition 1/6
Floor 1	0.0083%	0.0082%	0.0081%	0.0080%	0.0079%	0.0078%
Floor 2	0.0073%	0.0071%	0.0069%	0.0068%	0.0066%	0.0064%
Floor 3	0.0063%	0.0061%	0.0059%	0.0057%	0.0055%	0.0053%
Floor 4	0.0051%	0.0049%	0.0047%	0.0046%	0.0044%	0.0043%
Floor 5	0.0039%	0.0038%	0.0037%	0.0036%	0.0035%	0.0034%
Floor 6	0.0028%	0.0027%	0.0025%	0.0024%	0.0023%	0.0023%
Floor 7	0.0022%	0.0021%	0.0019%	0.0017%	0.0016%	0.0015%
Floor 8	0.0031%	0.0029%	0.0028%	0.0027%	0.0026%	0.0025%
Floor 9	0.0041%	0.0040%	0.0040%	0.0039%	0.0038%	0.0037%

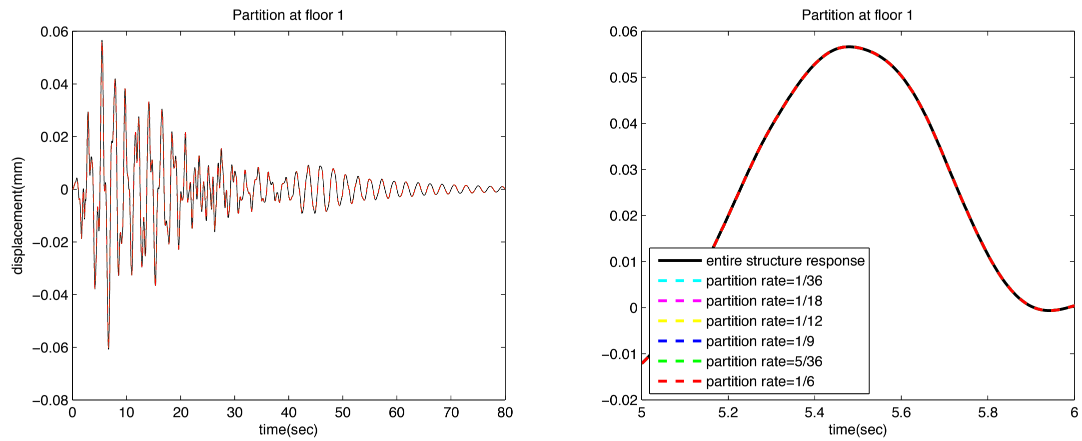


Figure 7.22.: MDOF Numerical RTHS: Partition at Floor 1, Sampling 512 Hz, Floor 1 Displacement

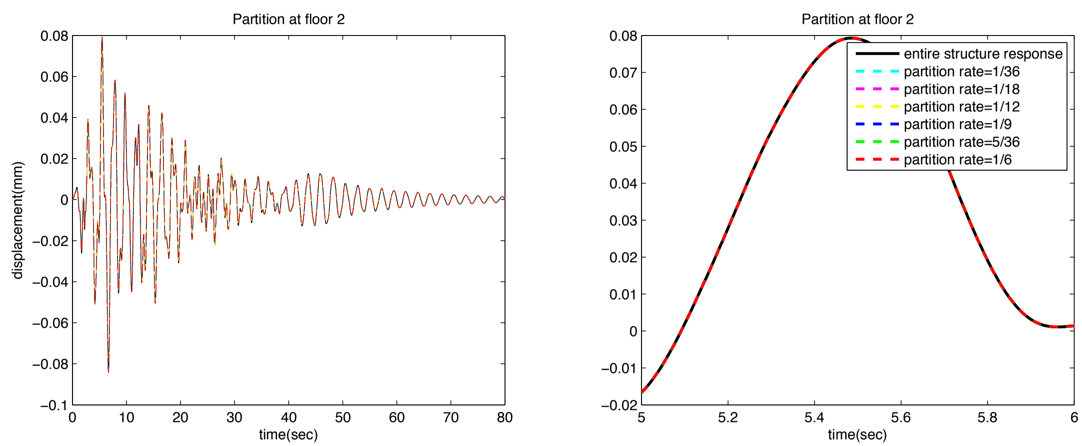


Figure 7.23.: MDOF Numerical RTHS: Partition at Floor 2, Sampling 512 Hz, Floor 2 Displacement

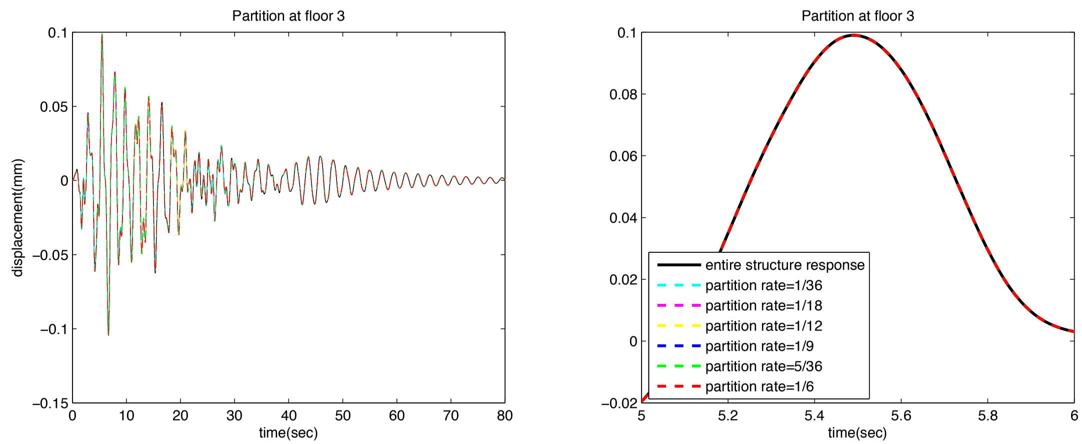


Figure 7.24.: MDOF Numerical RTHS: Partition at Floor 3, Sampling 512 Hz, Floor 3 Displacement

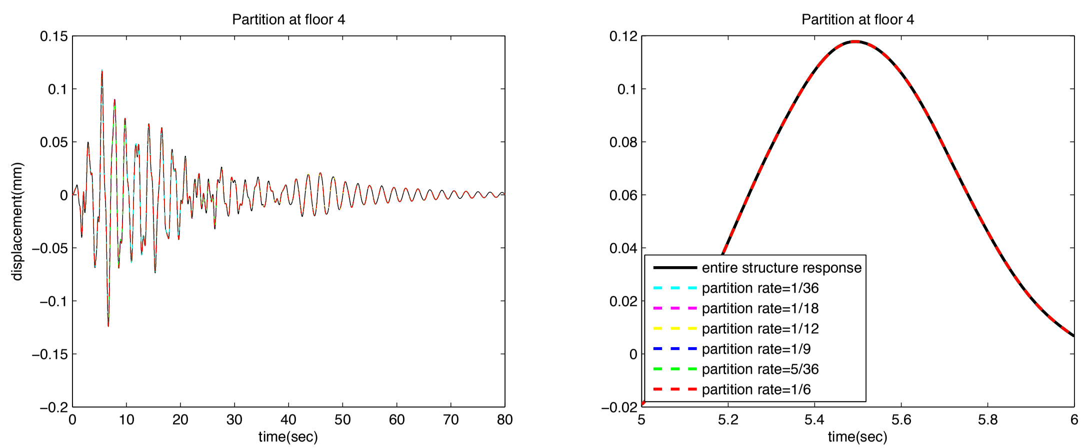


Figure 7.25.: MDOF Numerical RTHS: Partition at Floor 4, Sampling 512 Hz, Floor 4 Displacement

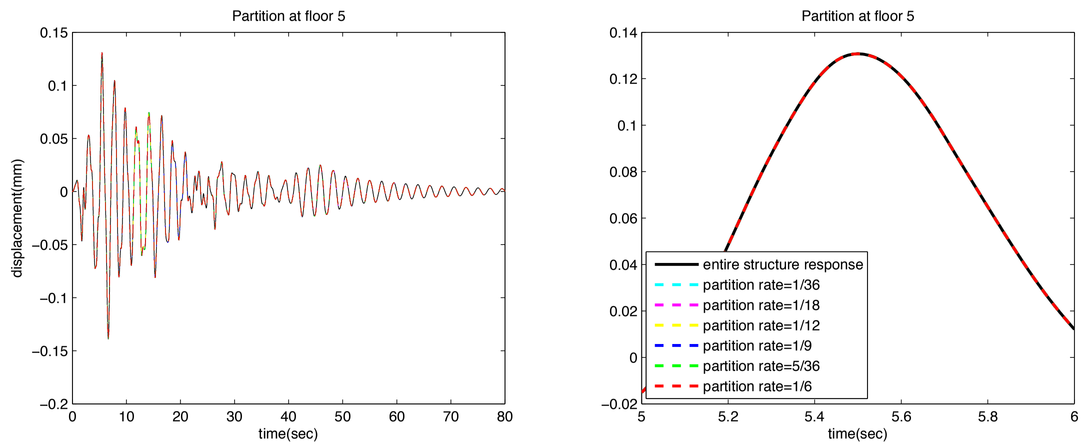


Figure 7.26.: MDOF Numerical RTHS: Partition at Floor 5, Sampling 512 Hz, Floor 5 Displacement

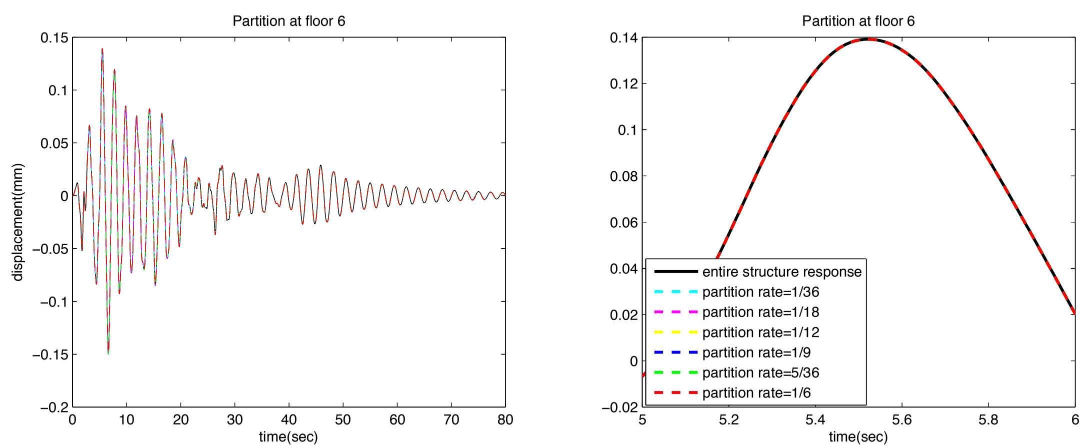


Figure 7.27.: MDOF Numerical RTHS: Partition at Floor 6, Sampling 512 Hz, Floor 6 Displacement

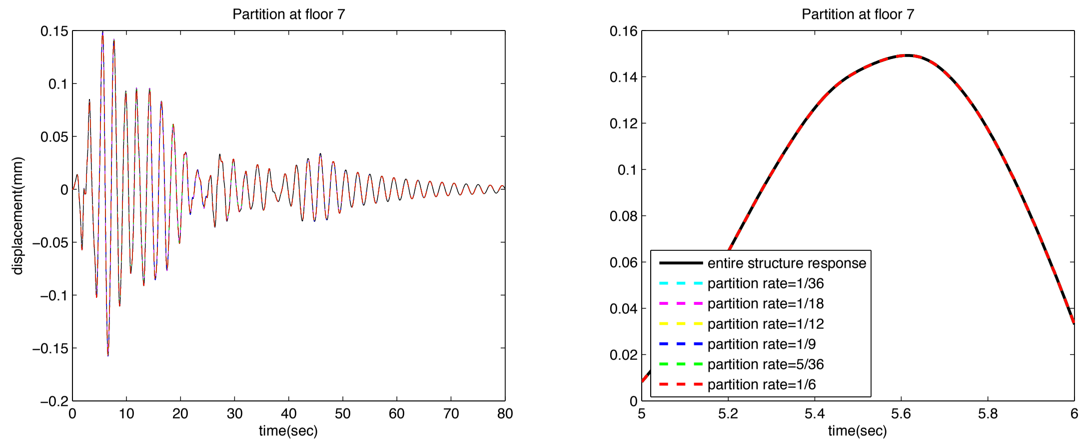


Figure 7.28.: MDOF Numerical RTHS: Partition at Floor 7, Sampling 512 Hz, Floor 7 Displacement

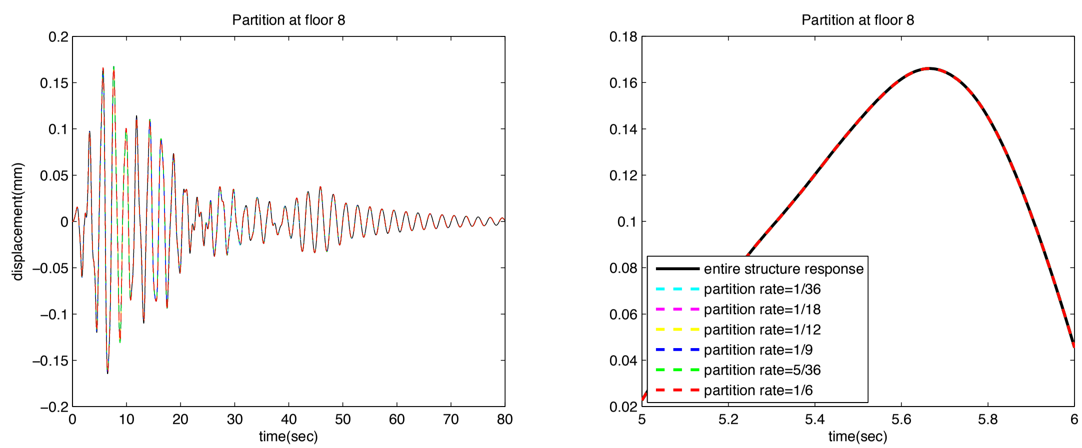


Figure 7.29.: MDOF Numerical RTHS: Partition at Floor 8, Sampling 512 Hz, Floor 8 Displacement

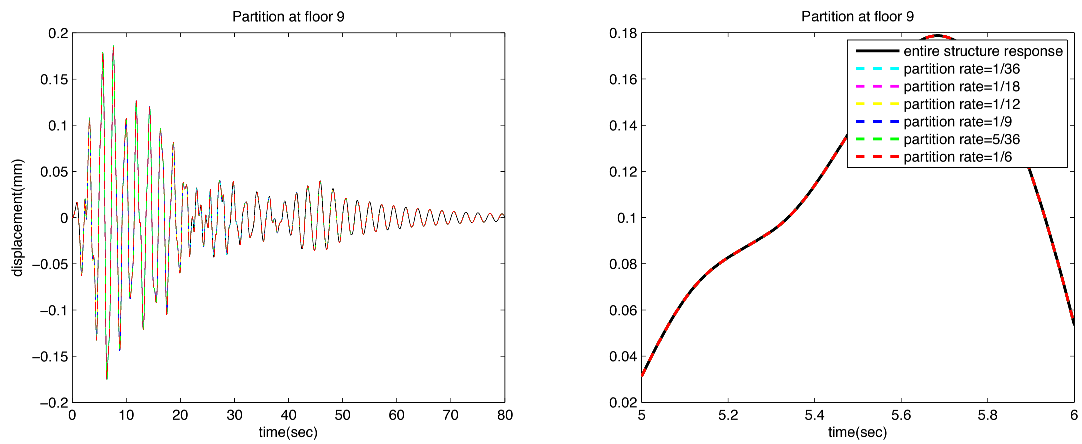


Figure 7.30.: MDOF Numerical RTHS: Partition at Floor 9, Sampling 512 Hz, Floor 9 Displacement

7.3 Experimental Validation

Actual experimental RTHS is also conducted to demonstrate the effectiveness of using modified Runge-Kutta integration algorithm over conventional integration. The experimental substructure chosen is a steel frame located in the Intelligent Infrastructure System Laboratory `https://engineering.purdue.edu/IISL/` located at Purdue University shown in Fig. 7.31.

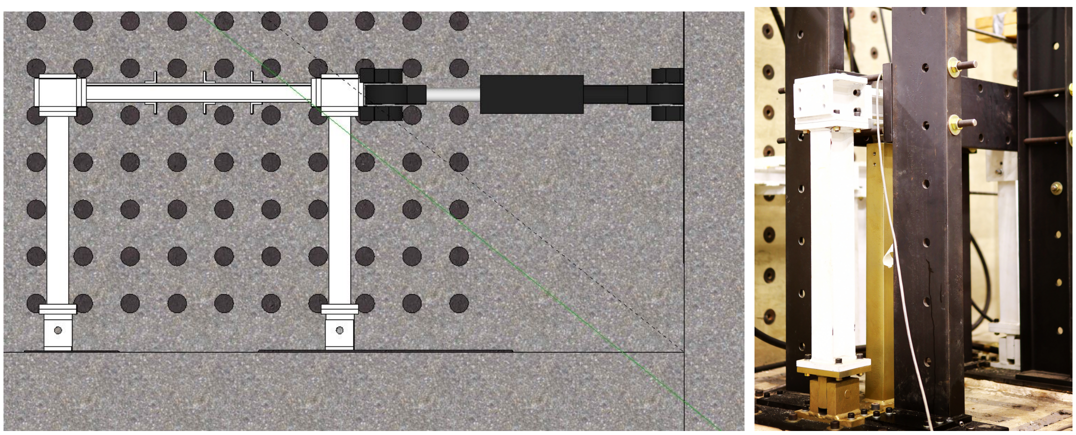


Figure 7.31.: Experimental Substructure Setup for RTHS Validation

The stiffness of the experimental substructure K^E is identified through a predefined displacement test, K^E is identified as 8.25 kips/in (1444.7 kN/m) as shown in Fig. 7.32. The actuator attached has a maximum force capacity of 2 kips (8.82k kN), which limited the maximum displacement response of the RTHS under 0.25 inch (6.35 mm). The structure's dynamic properties are identified using a hammer test, with results shown in Fig. 7.33. The structure is highly damped with damping ratio ζ_E of 5.54%. Note that experimental mass M^E of the structure is 80.3 lb (36.4 kg), which is negligible compared to M^N noted in Table 7.7.

Four different RTHS cases are designed with natural frequencies of 0.5 Hz, 1 Hz, 1.5 Hz and 2 Hz. Each is excited with both El-Centro and Kobe earthquakes, and the earthquake intensities in both cases were chosen in order to keep the response displacement under 0.2 inch. In each case, 2/3 of the structural stiffness is assumed

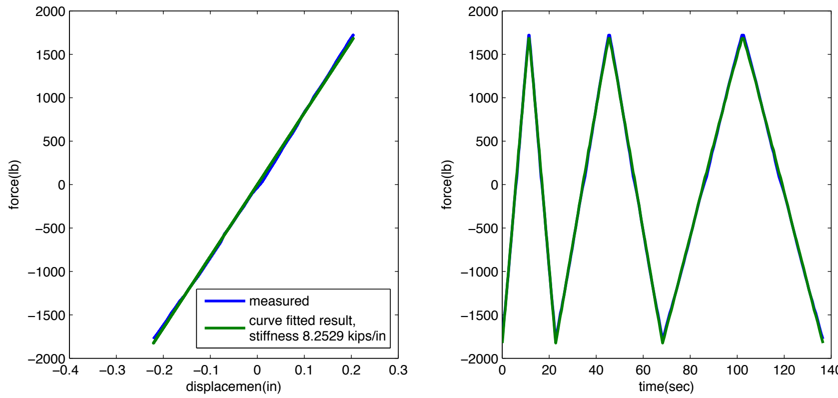


Figure 7.32.: Experimental Stiffness Identification for RTHS

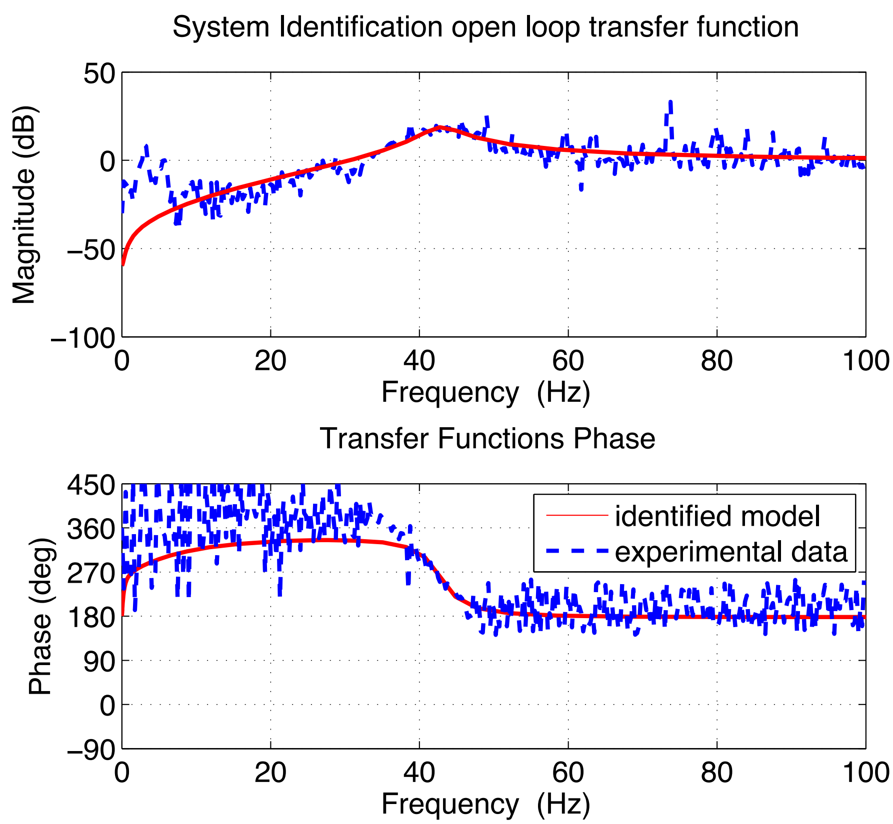


Figure 7.33.: Experimental Dynamic Property for RTHS

to be contributed by experimental substructure, and the damping ratio of reference structure in the three cases was set to 2%. Test parameter details are shown in Table 7.7, and in the following expressions:

$$K^N = 0.5K^E, P_K = 67\% \tag{7.32}$$

$$M^N = (K^E + K^N)/\omega^2 - M^E \tag{7.33}$$

$$C^N = 2\zeta\sqrt{KM} - C^E \tag{7.34}$$

$$K = K^N + K^E \tag{7.35}$$

$$C = C^N + C^E \tag{7.36}$$

$$M = (K^E + K^N)/\omega^2 \tag{7.37}$$

where ω is the natural frequency of the reference structure in rad/s. RTHS results are further compared to a simulation of the reference structure, with structural properties M, C, K. Sampling frequency is kept at 1024 Hz for RTHS and 2048 for reference structure simulation.

Table 7.7.: Partition Mass Damping and Stiffness for Experimental Validation

EQ Intensity Units	$f = \omega/2\pi$ Hz	M^N 10^3 Lb	C^N kips/in $\cdot$ s	K^N kips/in	M^E 10^3 Lb	C^E kips/in $\cdot$ s	K^E kips/in
3%	0.5	53	0.15	4.12	0.081	0.005	8.25
4%	1.0	13.3	0.073	4.12	0.081	0.005	8.25
5%	1.5	5.9	0.047	4.12	0.081	0.005	8.25
7%	2.0	3.3	0.033	4.12	0.081	0.005	8.25

In addition to inner PID control loop, the actuator is controlled using the robust integration actuator control (RIAC) algorithm as the external control loop [82]. The RIAC uses the H_∞ as feedback core controller, Kalman filter to minimize effect of noise on feedback measurement and a delay compensation block for online tuning. Tracking performance of band limited white noise signal of 10 Hz bandwidth using RIAC algorithm is shown in Fig. 7.34. The time delay of the actuator after RIAC is shown to be under 1 msec which indicates that the actuator lag is of a similar magnitude compared to the unit computation delay.

Three integration methods, Newmark-β (with $\gamma = 0.5$ and $\beta = 0.25$), conventional Runge-Kutta and the modified Runge-Kutta, are examined with this RTHS setup. The RTHS results from these cases are presented in Fig. 7.35 and 7.36. The CR method and the discrete state-space method yields very similar responses compared to the Newmark-β and the conventional RK method, and therefore were omitted from these plots. All these results are generated using the same actuator control algorithm and testing took place on the same day.

Table 7.8.: Experimental RTHS Error Comparison: El-Centro Earthquake

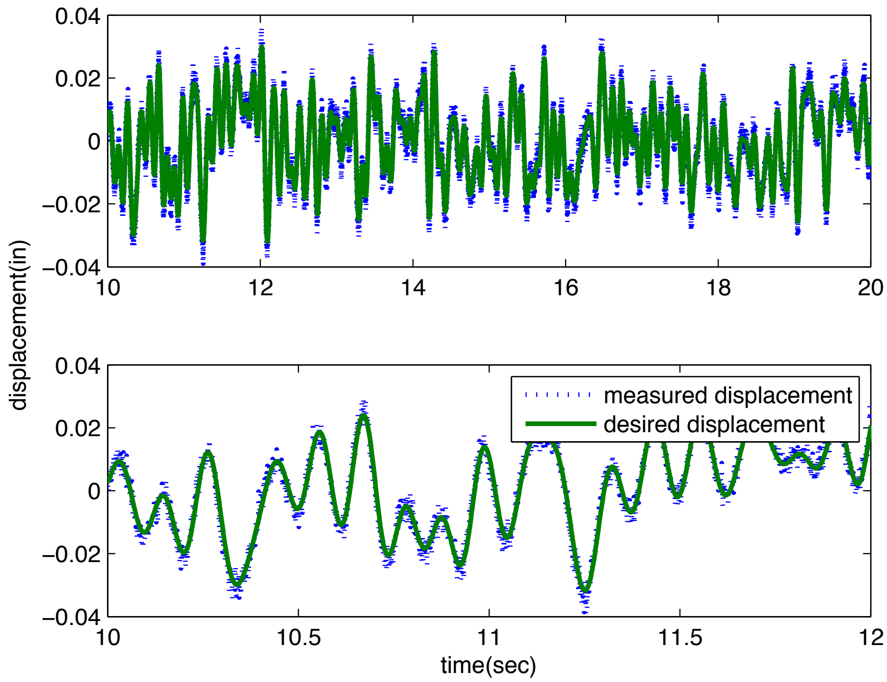
$f = \omega/2\pi$	Ω	Error using NB	Error using RK	Error using MRK
0.5 Hz	0.0031	7.24%	5.62%	3.16%
1.0 Hz	0.0061	4.5%	3.65%	1.76%
1.5 Hz	0.0092	12.67%	12.4%	4.48 %
2.0 Hz	0.0123	11.86%	12.91%	3.37%

Table 7.9.: Experimental RTHS Error Comparison: Kobe Earthquake

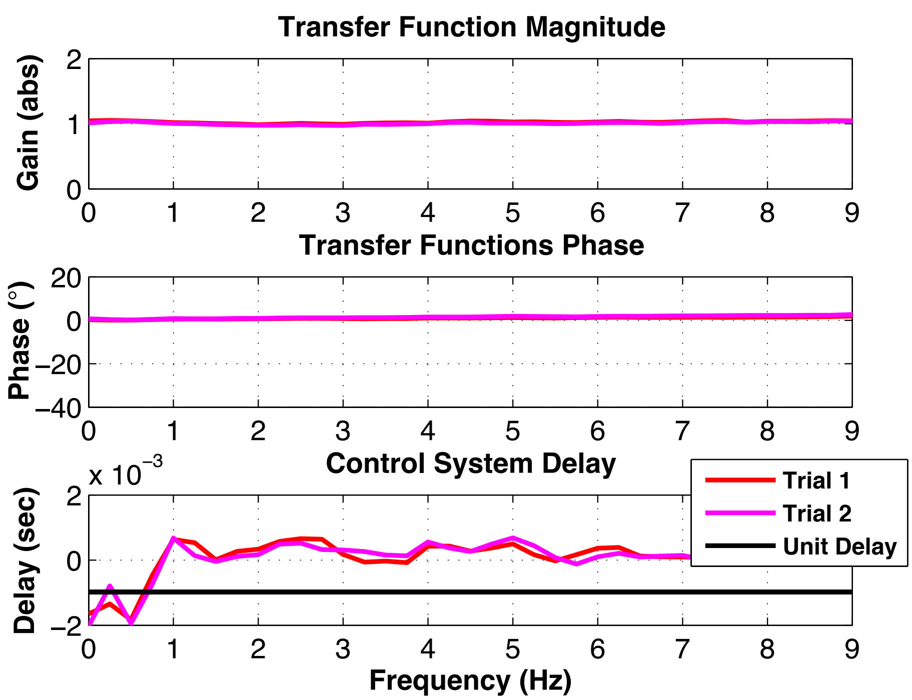
$f = \omega/2\pi$	Ω	Error using NB	Error using RK	Error using MRK
0.5 Hz	0.0031	7.65%	6.68%	3.97%
1.0 Hz	0.0061	6.39%	5.11%	2.42%
1.5 Hz	0.0092	12.27%	11.12%	4.11 %
2.0 Hz	0.0123	30.69%	28.28%	7.44%

It is clear from the responses in Fig. 7.35 and 7.36 that the proposed MRK method performs far better than any of the conventional integration schemes. Computed values of the RMS error are listed in Tables 7.8 and 7.9. In general the errors are lower for low natural frequencies of the reference structure and increase for higher natural frequencies. As demonstrated in the numerical examples, RTHS with RK, CR and NB integration algorithm yields unstable results with $P_K = 0.7$ at $\Omega = 0.025$, in the experimental validation. For the case where the reference structure has a natural frequency at 2 Hz, RTHS with conventional integration is only marginally stable and thus, it was decided not to continue increasing the natural frequency of the reference structure. However, for RTHS using MRK, the test is extended to reference structures with natural frequencies of 2.5 Hz ($\Omega = 0.0153$) and 3 Hz ($\Omega = 0.0184$) (shown in Figures 7.35(e) and 7.35(f)) without any concerns of instability.

Compared to simulated RTHS results, the errors in the actual RTHS are observed to be slightly higher because of the inexact enforcement of the boundary conditions and the underlying complexity of RTHS. Factors such as two feedback loops (force feedback in RTHS and displacement feedback in actuator control), noise and system uncertainties (such as actuator dynamics and test specimen variabilities) also affect the accuracy of RTHS results. Nevertheless, all the results indicate that the RTHS performance is affected negatively in the presence of a single-step computational delay and that performance is significantly improved by using the MRK algorithm.

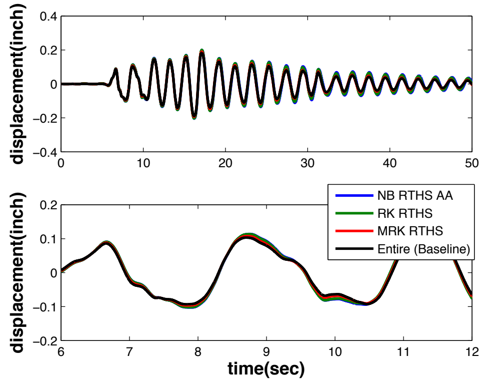


(a) Tracking time history using RIAC

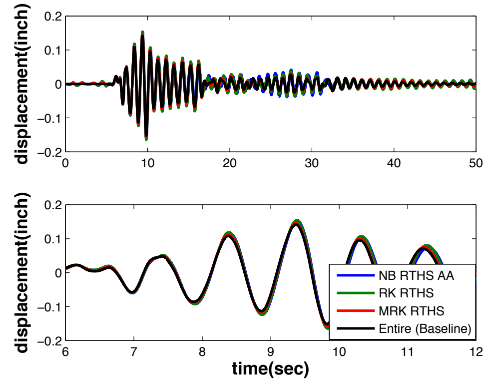


(b) Tracking transfer function using RIAC

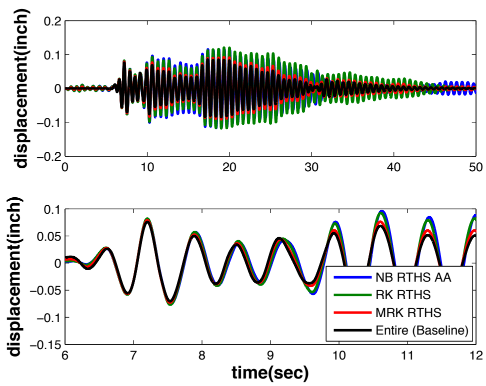
Figure 7.34.: Controller Tracking Performance for Experimental Substructure



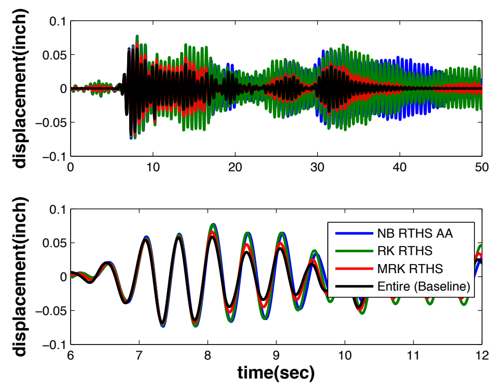
(a) RTHS Results: $f_n = 0.5$ Hz



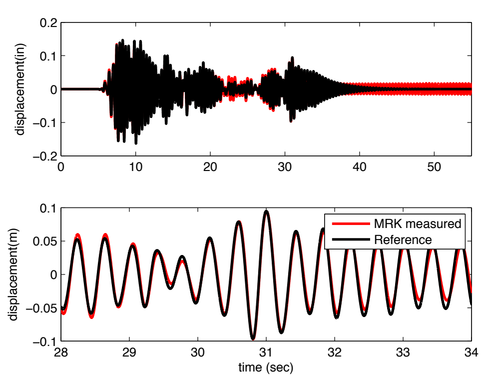
(b) RTHS Results: $f_n = 1$ Hz



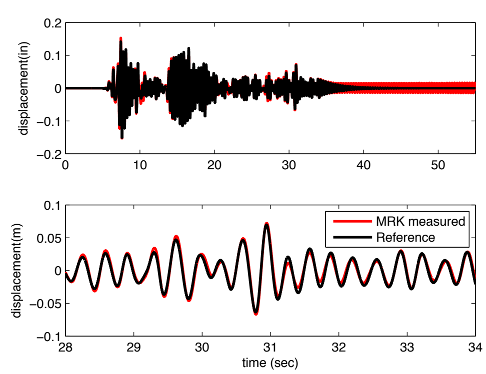
(c) RTHS Results: $f_n = 1.5$ Hz



(d) RTHS Results: $f_n = 2$ Hz

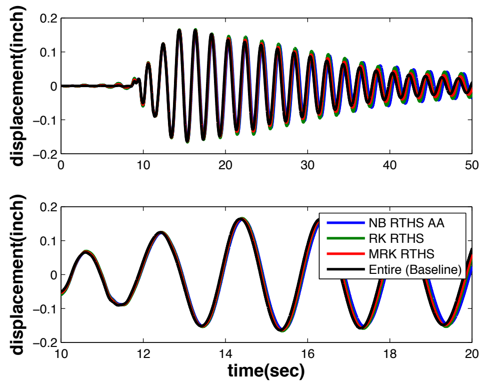


(e) RTHS Results (MRK): $f_n = 2.5$ Hz

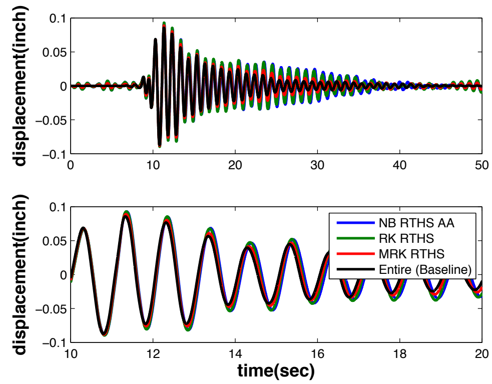


(f) RTHS Results (MRK): $f_n = 3$ Hz

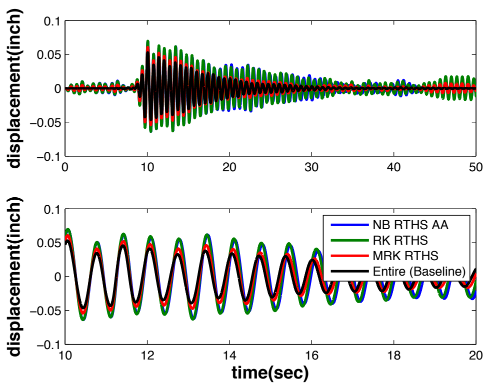
Figure 7.35.: Experimental RTHS: El-Centro Earthquake



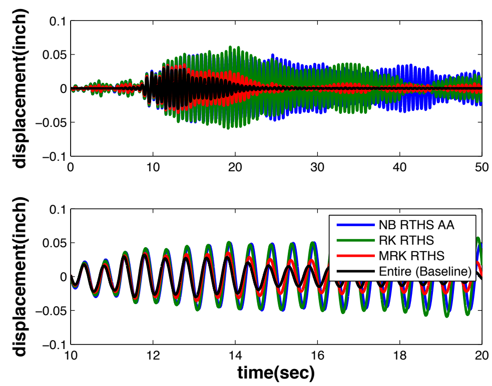
(a) RTHS Results: $f_n = 0.5$ Hz



(b) RTHS Results: $f_n = 1$ Hz



(c) RTHS Results: $f_n = 1.5$ Hz



(d) RTHS Results: $f_n = 2$ Hz

Figure 7.36.: Experimental RTHS: Kobe Earthquake

7.4 Conclusion

An inherent unit delay exists in the force measurement of the experimental substructure in RTHS due to the sequential order of communication between the numerical and experimental substructures and this may cause instability or performance degradation of the test. To explicitly evaluate the effect of this unit delay, different integration algorithms were employed in this study. Compared to pure simulation, both analytical and simulation results indicate that, the unit delay in the closed loop RTHS affects the stability and is highly dependent upon integration step size, structure natural frequency and structure partitioning ratio. Therefore, a modified Runge-Kutta integration algorithm is proposed to predict the feedback force measurement and minimize the effect of this inherent delay. The MRK integration includes three computation stages, 1) pseudo response calculation, 2) prediction of the measured force and 3) corrected response calculation. Results illustrate that the modified Runge-Kutta improves the performance of RTHS. Further, a robustness analysis, considering modeling error in the experimental substructure, demonstrates that only under-estimation of structure stiffness (specimen stiffening) may affect MRK stability for the undamped case. For lightly damped structures with high partitioning ratio, the MRK method is shown to be robust for up to 40% modeling error. Experimental RTHS is also implemented to verify the effectiveness of the modified Runge-Kutta integration algorithm over conventional integration algorithms. A moment resisting frame with a large stiffness is tested as the experimental substructure in RTHS. Results indicated that the modified Runge-Kutta algorithm improves the accuracy of the RTHS and extends the stability limit of the test. It may be noted that, this method can directly be applied to other existing integration algorithms by adapting the three computation stages to that method in a similar manner as shown here.

8. NONLINEAR STRUCTURAL MODELS

In hybrid simulation, one attractive benefit of is that only the critical component must be fabricated and tested, the remainder of the structure can be numerically modeled. Therefore it is intuitive to choose auxiliary devices (MR dampers, base isolation, etc.) as the experimental substructure, because the functionality of those devices (normally for vibration control purposes) is unique and quite distinguishable from the structural components. However, for evaluating structures in which a given component (column, bridge pier, structural connection) may be used in multiple instances in the structure, one might take the approach of using a limited number of the repeated components as the critical physical specimens and leave the rest in the simulation. Therefore, the modeling error of these similar components may contribute significantly to the global response, and affect the fidelity of a RTHS. Thus, during hybrid simulation, if the numerical model of those nonlinear components can be updated according to the measure response from the experimental substructure, the fidelity of the experiment can be improved.

To describe such nonlinear behavior and enable model updating in hybrid simulation, a proper nonlinear model should be selected. Commonly used nonlinear models describe steel hysteresis can be categorized into two groups, one is named phenomenological model, in which macro mechanical behavior (displacement-force hysteresis and energy dissipation) is captured and described through differential equations. However, the associated parameters may not have physical meaning, therefore information cannot be transmitted from one test specimen to another. In addition, the initial conditions for model parameters cannot be estimated without physical testing. Another group is the constitutive model which is associated with the constitutive relationship (strain-stress relationship) of structural materials. The initial parameters of these models can be estimated from material test, with higher accuracy expected.

To consider the capabilities and limitations for their use in hybrid simulation, both phenomenological models and constitutive models are investigated in this chapter.

8.1 Structural Phenomenological Model

One of the commonly used hysteretic phenomenological models is the Bouc-Wen model, which was first proposed by Bouc [98] and then modified by Wen [99]. Ever since, variations on the Bouc-Wen model have been introduced to capture more complex material hysteresis such as pinching and degrading [100], [101], [102]. Those models have been validated experimentally and shown to be applicable to represent inelastic steel nonlinearity several decades [103], [104]. Two Bouc-Wen models are considered, including its general form and an extended form.

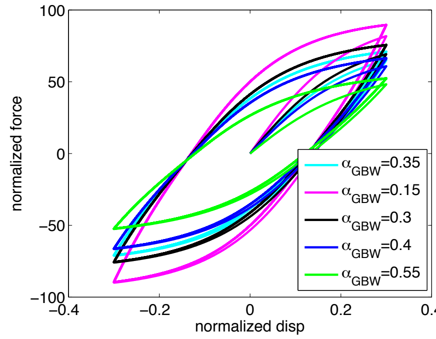
8.1.1 General Bouc-Wen Model

The classic Bouc-Wen model (denoted here as the General BW, or GBW in the manuscript) model stated in Eq. 8.1 and 8.2.

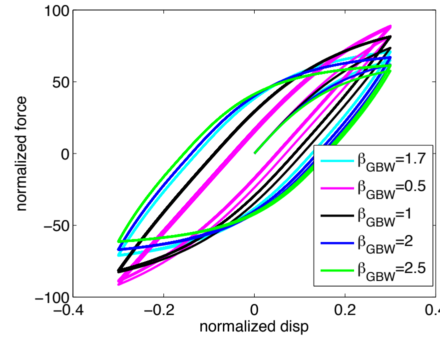
$$R_{GBW}(x^E, z) = \alpha_{GBW} k_{GBW} x^E + (1-\alpha) k_{GBW} z \tag{8.1}$$

$$\dot{z} = A_{GBW}\dot{x}^E - \beta_{GBW}|\dot{x}^E||z|^{n_{GBW}-1}z - \gamma_{GBW}\dot{x}^E|z|^{n_{GBW}} \tag{8.2}$$

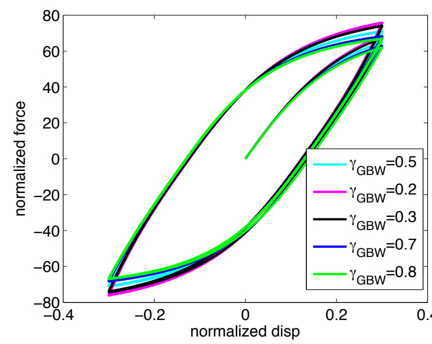
where k_{GBW} is the stiffness coefficient and $0 \leq \alpha_{GBW} \leq 1$ determines the level of nonlinearity, $\alpha_{GBW} = 1$ indicates the system is purely linear and $\alpha_{GBW} = 1$ indicates the system is purely hysteretic. A_{GBW}, β_{GBW}, n_{GBW}, γ_{GBW} govern the shape of the hysteresis loop. The hysteretic shape change with each parameter is illustrated in Fig. 8.1, and the results indicate that several parameters may contribute to the same shape change simultaneously. Thus, it is very hard to distinguish betwen the change resulting from each parameter. the simulation, the baseline parameter set is: $[\alpha_{GBW} = 0.35$; $\beta_{GBW} = 0.5$; $\gamma_{GBW} = 1.7$; $A_{GBW} = 15$, $K_{GBW} = 35$, $n_{GBW} = 1.7]$. In the simulation, only the indicated parameter value changes, the rest is kept to be the same as the baseline value.



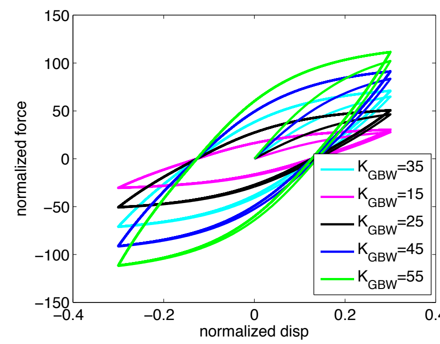
(a) α_{GBW} Change



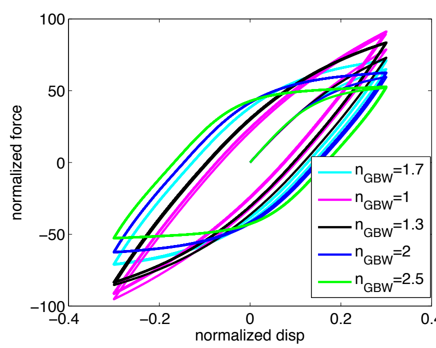
(b) β_{GBW} Change



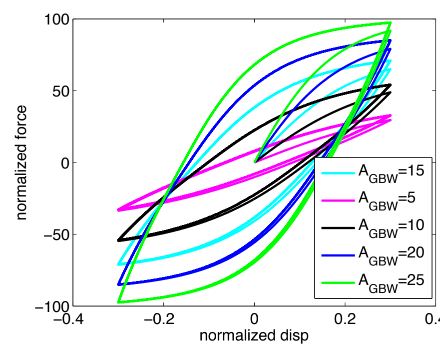
(c) γ_{GBW} Change



(d) K_{GBW} Change



(e) n_{GBW} Change



(f) A_{GBW} Change

Figure 8.1.: Hysteretic Shape Change to Each Parameter

8.1.2 Bouc-Wen-Baber-Noori model

In order to capture a greater variety of material behavior, an extended Bouc-Wen-Baber-Noori model [105], [100] model (denoted here as the Extended BW, or EBW) is considered, this model includes pinching and degradation effects of a structure component, represented in Eqs. 8.3 - 8.11.

$$R_{EBW}(x^E, z) = \alpha_{EBW} k_{EBW} x^E + (1 - \alpha) k_{EBW} z \tag{8.3}$$

α_{EBW} and k_{EBW} have similar definitions as α_{GBW} and k_{GBW} in the GBW.

$$\dot{z} = h(z)\{\frac{\dot{x}^E - \nu(\varepsilon)(\beta_{EBW}|\dot{x}^E||z|^{n_{EBW}-1}z + \gamma_{EBW}\dot{x}^E|z|^{n_{EBW}}}{\eta(\varepsilon)}\} \tag{8.4}$$

and, response duration and severity is measured by $\varepsilon(t)$ which is proportional to energy dissipation $E(t)$ in Eq. 8.5.

$$E(t) = \int (1 - \alpha_{EBW}) k_{EBW} z \dot{x}^E dt, \varepsilon(t) = \int z \dot{x}^E dt \tag{8.5}$$

$$\nu(\varepsilon) = 1 + \delta_{\nu B2}\varepsilon \tag{8.6}$$

$$\eta(\varepsilon) = 1 + \delta_{\eta B2}\varepsilon \tag{8.7}$$

where ν and η are degradation shape function, and $\delta_{\nu EBW}$, $\delta_{\eta EBW}$ are degradation parameters. To describe the pinching function, $h(z)$ is given by Eq. 8.8-8.11.

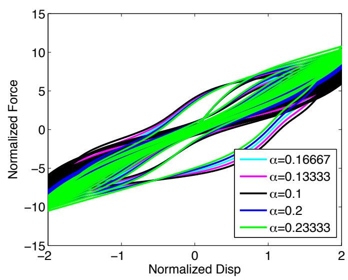
$$h(z) = 1 - \zeta_{1B2} e^{-[z \cdot sgn(\dot{x}^E) - q_{EBW} z_{x^E}]^2/\zeta_{2B2}^2} \tag{8.8}$$

$$\zeta_1(\varepsilon) = \zeta_{sB2}(1 - e^{-p_{EBW}\varepsilon}) \tag{8.9}$$

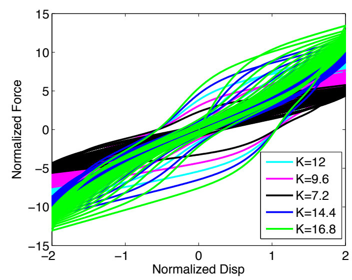
$$\zeta_2(\varepsilon) = (\Psi_{EBW} + \delta_{\Psi B2}\varepsilon)(\lambda_{EBW} + \zeta_1) \tag{8.10}$$

$$z_{x^E} = [\frac{1}{\nu(\varepsilon)(\beta_{EBW} + \gamma_{EBW})}]^{\frac{1}{n_{EBW}}} \tag{8.11}$$

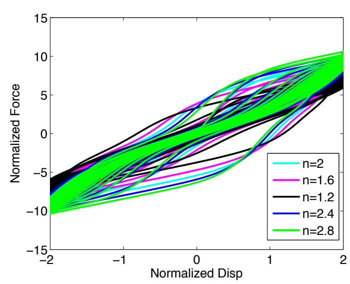
The parameters λEBW, ζ_{sEBW}, p_{EBW}, q_{EBW}, Ψ_{EBW}, and $\delta_{\Psi EBW}$ are involved in describing the pinching effect. p_{EBW} measures the initial drop of the slope, ζ_{sEBW} describe the total slip, Ψ_{EBW} is a parameter that contributes to the amount of pinching. $\delta_{\Psi EBW}$ specifies for the desired rate of pinching.



(a) α_{EBW} Change



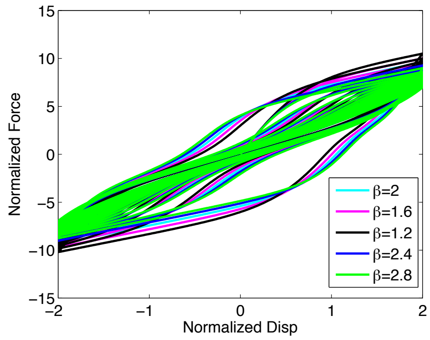
(b) k_{EBW} Change



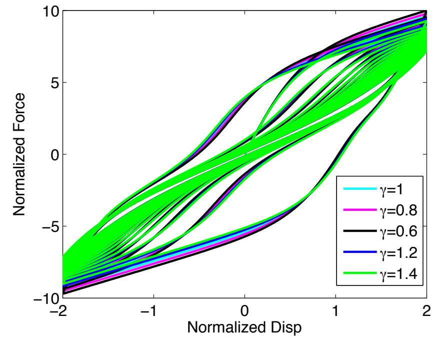
(c) n_{EBW} Change

Figure 8.2.: Hysteretic Shape Change to General Nonlinear Parameters

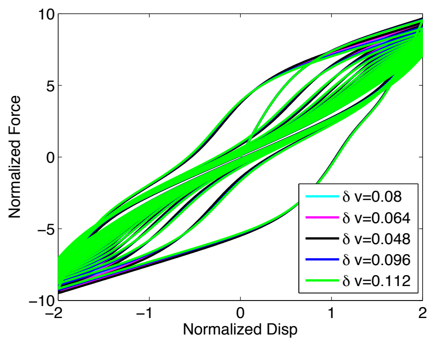
As in the analysis in the GBW case, the change of the hysteretic shape in EBW to each model parameter is investigated. Fig. 8.2 illustrates the changes in the hysteretic shape due to general nonlinear parameters α_{EBW}, k_{EBW}, and n_{EBW}. In Fig. 8.3, the degrading behavior dominated by β_{EBW}, γ_{EBW}, $\delta_{\nu EBW}$, and $\delta_{\eta EBW}$ is presented. in Fig. 8.4, the pinching shape change to the parameters associated is demonstrated.



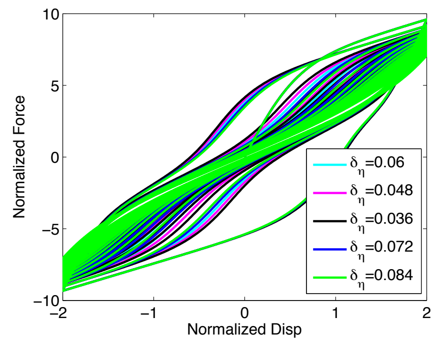
(a) β_{EBW} Change



(b) γ_{EBW} Change



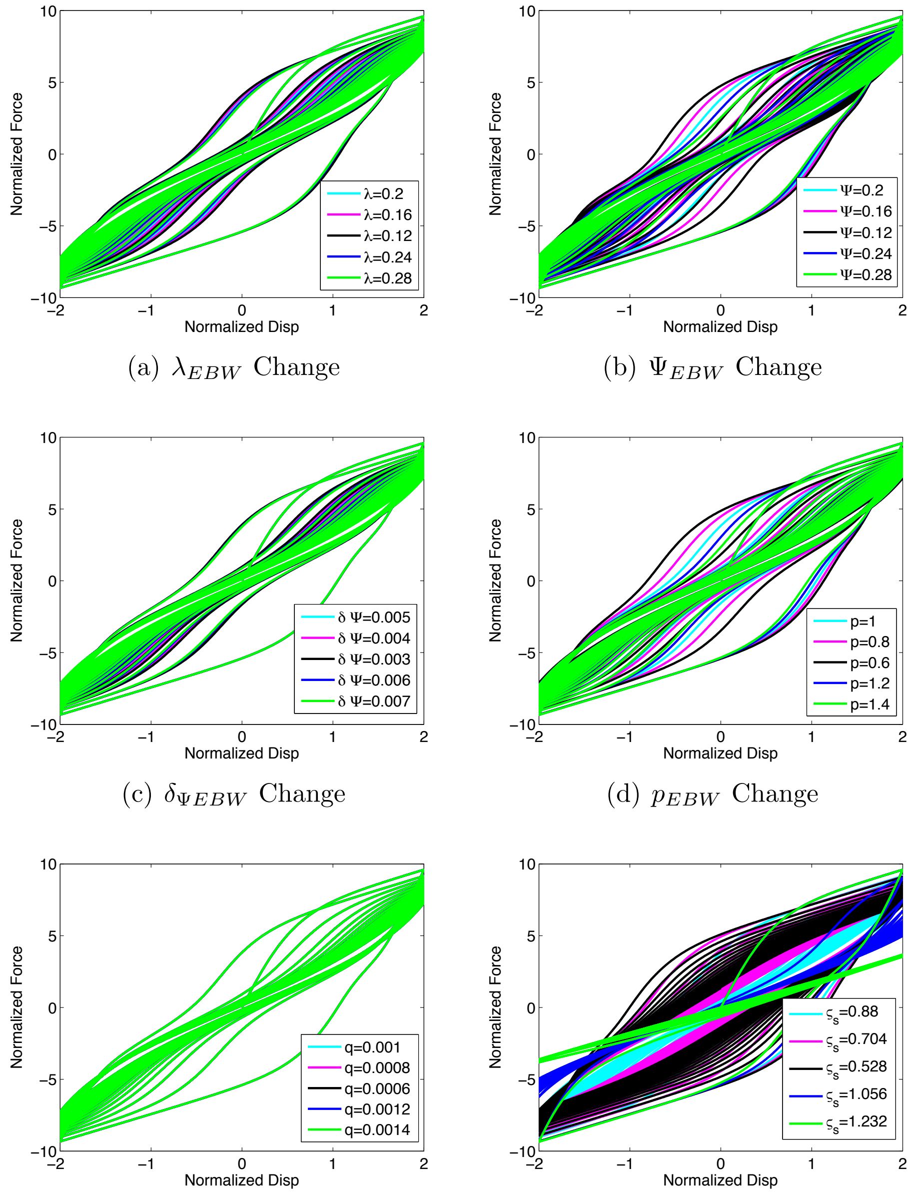
(c) $\delta_{\nu EBW}$ Change



(d) $\delta_{\eta EBW}$ Change

Figure 8.3.: Hysteretic Shape Change to Degrading Parameters

The simulation is built with a baseline parameter set with initial values: [$\alpha_{EBW} = 0.16$, $k_{EBW} = 12$, $\beta_{EBW} = 2$, $n_{EBW} = 2$, $\delta_{\eta EBW} = 0.06$, $\delta_{\nu EBW} = 0.08$, $q_{EBW} = 0.001$, $\gamma_{EBW} = 1$, $\zeta_{sEBW} = 0.88$, $p_{EBW} = 1$, $\Psi_{EBW} = 0.2$, $\delta_{\Psi EBW} = 0.005$, $\lambda_{EBW} = 0.2$]. In the simulation, only the indicated parameter value changes, the rest is kept to be the same as the baseline value. Each parameter varies up to $\pm 40\%$ of its baseline value, results indicate the sensitivity of each parameter is not the same. For model updating or system identification purpose, the convergence can be affected.



(a) λ_{EBW} Change

(b) Ψ_{EBW} Change

(c) $\delta_{\Psi EBW}$ Change

(d) p_{EBW} Change

(e) q_{EBW} Change

(f) ζ_{sEBW} Change

Figure 8.4.: Hysteretic Shape Change to Pinching Parameters

8.2 Structural Constitutive Model

Constitutive models integrate material properties in representing structural behavior. These models can provide information to assess component damage level, and structural serviceability and reliability. Therefore, it is tempting to study the feasibility of model updating algorithms with these constitutive models. In this section, to be compatible with the widely used hybrid simulation numerical solver platform OpenSees, two material models steel01 (bilinear model) and steel02 (Menegotto-Pinto model) are discussed.

8.2.1 Steel Bilinear Model

The most basic model for yielding steel is the bilinear model, as in Fig. 8.5. The strain-hardening coefficient b_s is the ratio of the post-yield tangent modulus betwee E^p and the initial elastic modulus E. If only considers the isotropic hardening, the simplified bilinear model can be described as:

$$E^p = b_s \cdot E \tag{8.12}$$

8.2.2 Steel Mengotto-Pinto Model

Menegotto and Pinto [106] established a nonlinear model to describe the stress-strain behavior of the reinforcing steel, later this model modified by Filippou et al. [107] which includes isotropic strain hardening. This Menegotto-Pinto can generate smooth hysteresis shape which can closer represent the experimental results than the bilinear curve. The empirical form of the model is:

$$\sigma^* = b \cdot \epsilon^* + \frac{(1-b) \cdot \epsilon^*}{1+\epsilon^R}^{1/R} \tag{8.13}$$

where, $\epsilon^* = \frac{\epsilon-\epsilon_{rev}}{\epsilon_0-\epsilon_{rev}}$, $\sigma^* = \frac{\sigma-\sigma_{rev}}{\sigma_0-\sigma_{rev}}$ Tangent modulus:

$$E^{tan} = \frac{\delta\sigma}{\delta\epsilon} = \frac{\sigma_0-\sigma_{rev}}{\epsilon_0-\epsilon_{rev}} \cdot \frac{\delta\sigma^*}{\delta\epsilon^*} \tag{8.14}$$

where: $\frac{\delta\sigma^*}{\delta\epsilon^*} = b + [\frac{1-b}{(1+\epsilon^{*R})^{1/R}}] \cdot [1 - \frac{\epsilon^{*R}}{1+\epsilon^{*R}}]$.

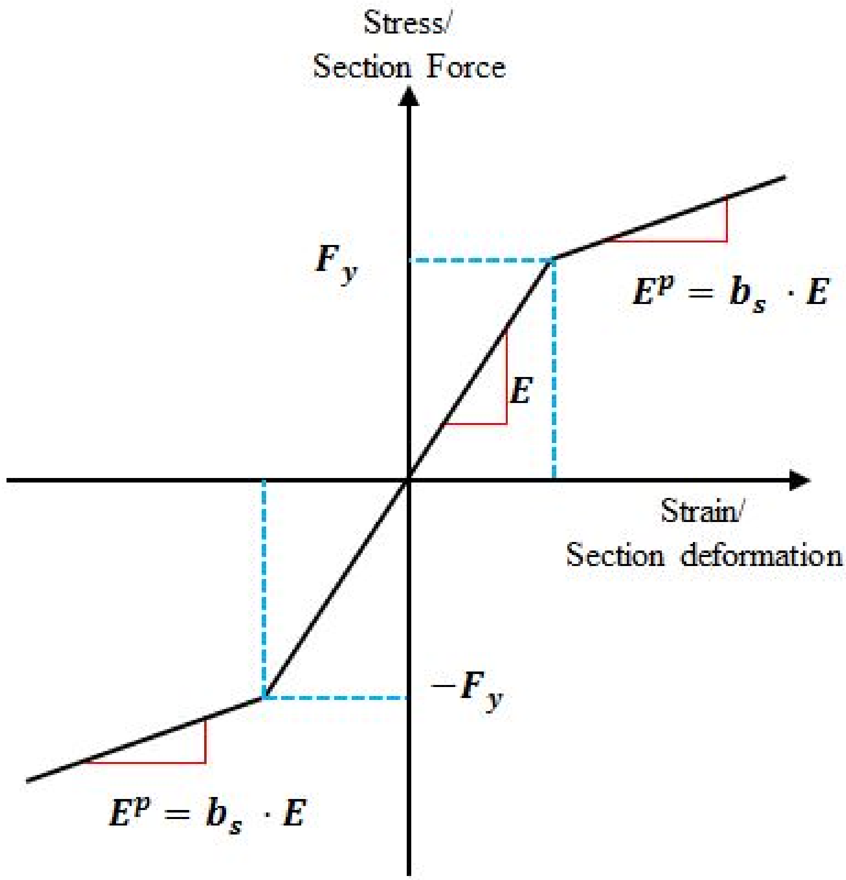


Figure 8.5.: Steel Bilinear Model

The hysteresis loop in the Megetto Pinto model is described by four asymptotes and the curve transition between them. The first two straight line asymptotes are represented with slope $E(a)$, indicating the initial material Young's Modulus. The origins of these asymptotes are at the strain reversal points (A and C in the Figure 8.6(a) and 8.6(b)), which stress and strain are σ_{rev} and ϵ_{rev}, these points are recorded during loading history. The other two asymptotes are with slope of E^{tan}, indicate the material stiffness after hardening. b descirbes the hardening ratio between $E(a)$ and E^{tan}. The intersection of the asympototes with different slopes (point B) are at the point with stress and strain of σ_0 and ϵ_0, where σ_0, ϵ_0,σ_{rev} and ϵ_{rev} are updated after each strain reversal.

The curve transition from two linear asymptote with different slopes is represented by curvature parameter R, this curved transition preserves a close representation of the Bauschinger effect.

In Fig. 8.6(b), R is dependent on the absolute strain difference between the current asymptote intersection point (point B) and the previous loading history (minimum or maximum strain reversal point) depending on whether the strain is decreasing or increasing, respectively.

For Steel02 in OpenSees, R is defined as:

$$R(\xi) = R_0(1 - \frac{cR_1 \cdot \xi}{cR_2 + \xi}) \tag{8.15}$$

where, R_0 is the value of the parameter R during first loading, and cR_1 and cR_2. ϵ can be expressed as:

$$\xi = |\frac{\epsilon^m - \epsilon_0}{\epsilon_y}| \tag{8.16}$$

where, ϵ^m is the strain at the previous maximum or minimum strain reversal point depending on whether the current strain is increasing or decreasing, respectively.

A shift of σ_0 and ϵ_0 is proposed to account for isotropic hardening, [107]:

a. If the incremental strain changes from positive to negative:

$$\Delta^N = 1 + a_1(\frac{\epsilon^{max} - \epsilon^{min}}{2 \cdot a_2 \cdot \epsilon_y})^{0.8} \tag{8.17}$$

$$\epsilon_0 = \frac{-\sigma_y \Delta^N + E^{tan} \epsilon_y \Delta^N - \sigma_{rev} + E \cdot \epsilon_{rev}}{E - E^{tan}} \tag{8.18}$$

$$\sigma_0 = -\sigma_y \cdot \Delta^N + E^{tan} \cdot (\epsilon_0 + \epsilon_y \cdot \Delta^N) \tag{8.19}$$

b. If the incremental strain changes from negative to positive:

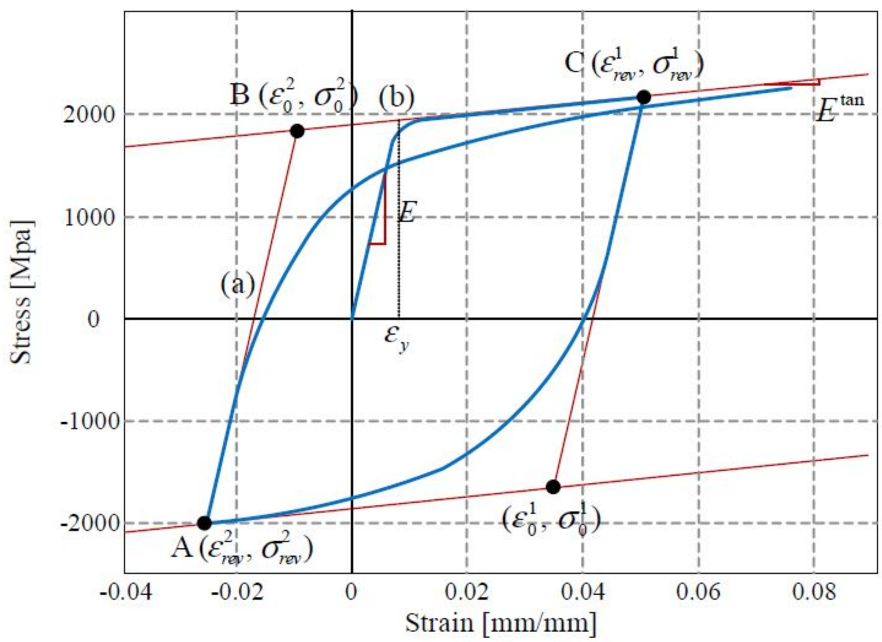
$$\Delta^P = 1 + a_3(\frac{\epsilon^{max} - \epsilon^{min}}{2 \cdot a_4 \cdot \epsilon_y})^{0.8} \tag{8.20}$$

$$\epsilon_0 = \frac{\sigma_y \Delta^P + E^{tan} \epsilon_y \Delta^P - \sigma_{rev} + E \cdot \epsilon_{rev}}{E - E^{tan}} \tag{8.21}$$

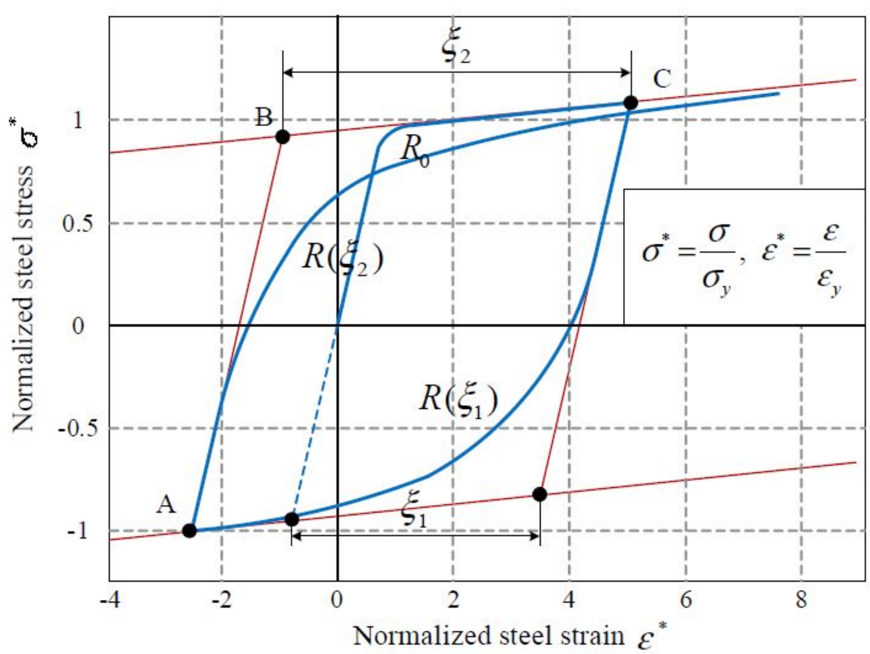
$$\sigma_0 = \sigma_y \cdot \Delta^P + E^{tan} \cdot (\epsilon_0 - \epsilon_y \cdot \Delta^P) \tag{8.22}$$

a_1, a_2, a_3, and a_4 are isotropic hardening parameters. From previous studies that a_1, a_2, a_3, and a_4 must be determined through curve fitting of the model with experimental results [94]. Default values are $a_1 = a_3 = 0$ and $a_2 = a_4 = 55$ in OpenSees, and ϵ_{max} and ϵ_{min} are the maximum and minimum strain at the reversal points.

Therefore, parameters for Menegotto-Pinto model are: E, b, ϵ_y, R_0, cR_1, cR_2, a_1, a_2, a_3, and a_4. A global simulation is built to understand the hysteretic shape change to each parameter, as in Fig. 8.7. The physical meaning of E, b, and ϵ to the hysteretic shape change is quite clear, and parameters R_0, cR_1, cR_2 parameters only associated to the hysteretic shape. Therefore, Menegotto-Pinto is an attractive model which represent steel Bauschinger effect more accurately (compare to the bilinear model), meanwhile, it can deliver a relative accurate initial guess of a component behavior (dominated by E, b and ϵ) as compared to phenomenological model. However, as stated earlier, the memory of the past loading history is stored as switch functions with flag indications, the implementation of online model updating or recursive updating method would be quite difficult.

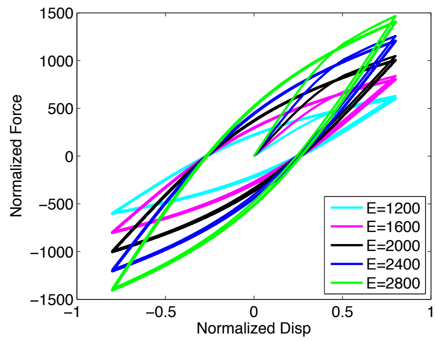


(a) Menegotto-Pinto Hysteresis

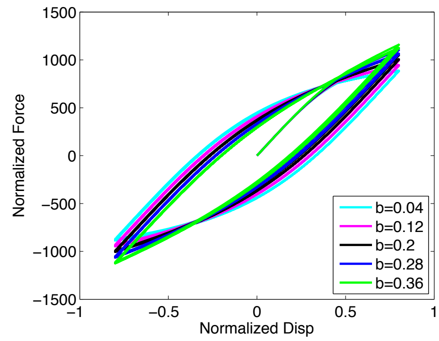


(b) Definition of Curvature Parameter

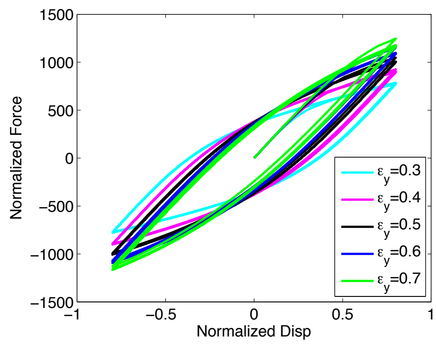
Figure 8.6.: Steel Menegotto-Pinto steel model [94]



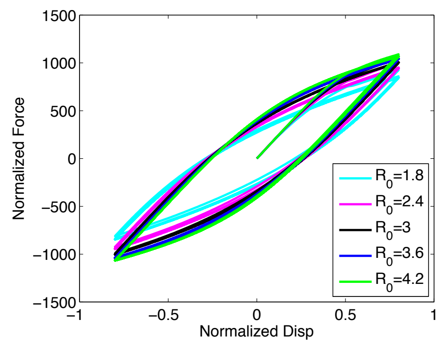
(a) E Change



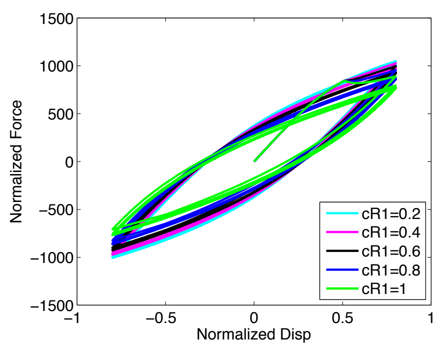
(b) b Change



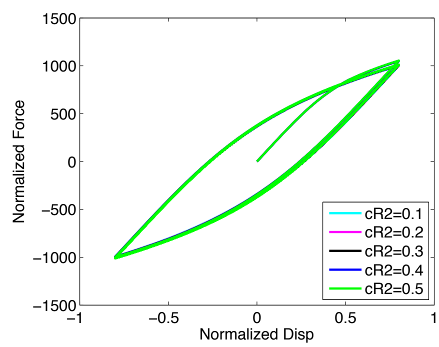
(c) ϵ_y Change



(d) R_0 Change



(e) cR_1 Change



(f) cR_2 Change

Figure 8.7.: Hysteretic Shape Change to Menegotto-Pinto Parameters

8.3 Conclusion

Several widely used nonlinear models are described and discussed in this section. Phenomenological models are capable of capturing the hysteretic shape of a structural component. One major drawback of these models is that the parameters do not have physical meaning, therefore, it is very difficult to start any model updating process with a reasonable initial guess. Also, knowledge of previous component models and test results cannot be transmitted to a new specimen. In addition, in Bouc-Wen models, the hysteretic shape is more sensitive to some parameters. This can affect model updating convergence.

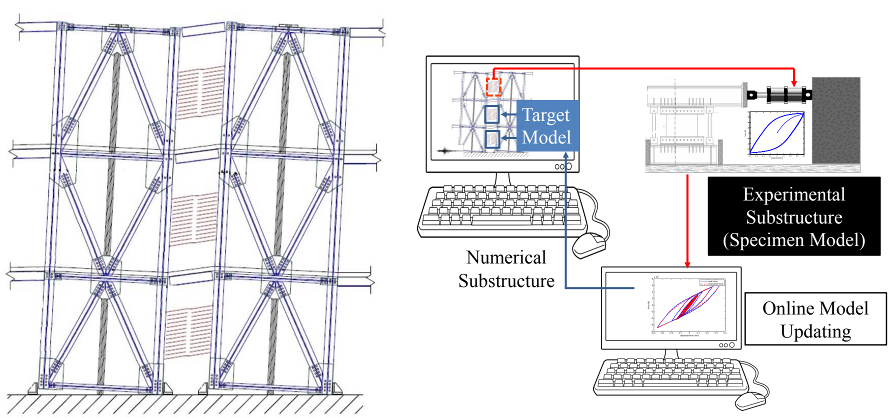
Steel constitutive models including the bilinear model and the Menegotto-Pinto model are discussed. Both models are dominated by material properties such as the Young's modulus, the yield strain, and the strain-hardening ratio. There are shape parameters in the Menegotto-Pinto model to capture the Bauchinger effect of a steel component. However, the Menegotto-Pinto model is associated with memory of loading history, this has been simplified with flag indications in computational implementation. Therefore, computational execution of the Menegotto-Pinto model is equivalent to piece-wise function, it is difficult to incorporate online (recursive) model updating algorithms.

9. HYBRID SIMULATION WITH MODEL UPDATING FORMULATION AND NUMERICAL ANALYSIS

As stated in chapter 8, generally, a given component (column, bridge pier, structural connection) may be used in multiple instances in one structure, in hybrid simulation, one might take the approach of using a limited number of the repeated components as the critical physical specimens and leave the rest in the numerical substructure. Rather than only exchanging information at the interface (displacement, acceleration or restoring force) as in the conventional hybrid simulation, in hybrid simulation with model updating (HSMU), model information can also be extracted from the response of the experimental substructure. Such model information can then be used to improve the representation of similar components in the numerical model.

Kwon *et al.* [56] first introduced the concept of representing an entire structure with several key physical components, and modifying their numerical models using the physical response in real-time. The numerical model used in simulation consisted of a collection of Bouc-Wen models with predetermined parameters. During HS, a weighting factor was identified for each Bouc-Wen model until the summation of their weighted responses matched the measured physical response. Thus, the accuracy of this approach highly depends on the chosen initial collection of Bouc-Wen models. In the subsequent two years, several techniques to apply model updating in HS have been developed, mostly in the unscented Kalman filter family. Those approaches include using the constrained unscented Kalman filter (CUKF) in RTHS [108], [109] and the unscented Kalman filter (UKF) algorithm in HS [58] and RTHS [110] to identify Bouc-Wen model parameters. Experimental results in the aforementioned work demonstrate the feasibility of model updating in hybrid simulation and the associated improvement in testing accuracy.

Although HSMU has been experimentally validated, an evaluation of the limitations of HSMU has not yet been performed. For HSMU, online model updating algorithms require knowledge of the excitation to the experimental substructure as well as its response to identify the model parameters. This excitation normally takes the form of a structural response which is already filtered (the structure itself is a filter) and likely contains limited frequency information, especially on examining the dynamic system where specimen response is rate dependent. In HS, the identification information is more related to amplitude where the loading does not contain frequency content with low speed execution. While in RTHS, the information maybe related to both amplitude and frequency. Other possible limitations relate to the varying level of complexity of the nonlinear models to be identified. In addition, parameter convergence in model updating can affect the behavior of other numerical components which receives model updating parameters in real time. Clearly, the performance of the chosen model updating algorithm with respect to such challenges should be carefully examined prior to implementing model updating in hybrid simulation.



(a) Rocking Frame with Connection Plates [112]

(b) Concept of HSMU

Figure 9.1.: Numerical Example

In this chapter, HSMU is validated and evaluated through a numerical study, representing a practical design shown in Fig. 9.1(a) from [111]. This case study is based on a NEES (Network for Earthquake Engineering Simulation) project [112], where

Deierlein *et al.* [113] proposed a self-centering frame concept with post-tensioned (PT) strands in two linear frames, linked with three energy dissipation shear plates. In the discussion of the RTHSMU simulation, we define the **specimen model** as a virtual nonlinear experimental substructure to be loaded based on the structural response in the RTHS, which in this case is simply the upper shear plate. The nonlinear behavior of the specimen model is represented using Bouc-Wen models with different levels of complexity. The other two plates with the same nominal design are included within the numerical substructure, each defined to be a **target model**. The initial parameters of the target models are unknown (selected parameters are far away from the specimen model) and are expected to be updated using information about the specimen model acquired during the hybrid simulation. The CUKF is used as the online nonlinear model identification method.

This HSMU simulation (Fig. 9.1(b)) includes: the numerical substructure consisting of frames with PT strands, and target model which is updated with real-time parameters; an experimental substructure which only consists of the specimen model; and the online model updating algorithm. The performance of HSMU simulation is first presented at the global level, where the self-centering frame behavior is compared to baseline simulation results without any modeling errors in either the target model or the specimen model, and then discussed thoroughly on the component level in terms of the model updating. The evaluation and validation of the model updating performance considers the following: richness of the input (broadband signal, structural response, sinusoidal signal, etc.), complexity of the target model, and selected tunable variables in the model updating algorithm.

9.1 Hybrid Simulation with Model Updating

Consider the equation of motion for a reference structural with nonlinear components in conventional simulation written as:

$$M\ddot{x} + C\dot{x} + Kx + R(x, \dot{x}, \theta_R) = -M\Gamma\ddot{x}_g \tag{9.1}$$

where M, C, K are the linear mass, damping, stiffness matrices of the reference structure, $R(x, \dot{x}, \theta_R))$ is the restoring force provided by the nonlinear components, θ_R are the parameters of the nonlinear components, x, $\dot{x}$ and $\ddot{x}$ are structural responses (displacement, velocity, acceleration), and $\ddot{x}_g$ denotes earthquake excitation.

We write the equations of motion for hybrid simulation in the form:

$$M^N \ddot{x}^N + C^N \dot{x}^N + K^N x^N + F^E(x^E, \dot{x}^E) + R^N(x^N, \dot{x}^N, \theta_R) = -M\Gamma \ddot{x}_g \tag{9.2}$$

$$M^E \ddot{x}^E + C^E \dot{x}^E + K^E x^E + R^E(x^E, \dot{x}^E) = F^E(x^E, \dot{x}^E) \tag{9.3}$$

where the superscript $(\ )^N$ and $(\ )^E$ denote the portions of the reference structure included in the numerical and experimental substructures respectively, $M = M^E + M^N$, $C = C^E + C^N$, $K = K^E + K^N$. F^E denotes the measured force in the experimental substructure. This representation of the reference structure. The fidelity of hybrid simulation is based on how accurately Eq. 9.2 and 9.3 represent the Eq. 9.1 when implemented.

To focus on the analysis of the impact of model updating, we assume boundary condition continuity is preserved ($x^E = x^N$ and $\dot{x}^E = \dot{x}^N$). Because it is relatively straightforward to identify the properties of a linear structure M^E, C^E, K^E prior to testing, the accuracy of the hybrid simulation depends mainly on the ratio of R^N/R and the modeling error in $R^N(x^N, \dot{x}^N, \theta_R)$. In many past hybrid simulation studies such as those with isolated dampers as the physical components, the nonlinear restoring force is dominated by those and $R^E >> R^N$. However, when the physical specimen is selected to include structural components that are used in multiple instances within a structure, the limited number of specimen selected for physical experimentation means that a significant portion of the nonlinear behavior resides in the numerical substructure ($R^N >> R^E$) and there maybe modeling errors present in R^N. Thus, the modified formulation of hybrid simulation which includes model updating is:

$$M^N\ddot{x}^N + C^N\dot{x}^N + K^N x^N + F^E(x^E, \dot{x}^E) + R^N(x^N, \dot{x}^N, \tilde{\theta}_R) = -M\Gamma\ddot{x}_g \tag{9.4}$$

$$M^E\ddot{x}^E + C^E\dot{x}^E + K^E x^E + R^E(x^E, \dot{x}^E) = F^E(x^E, \dot{x}^E) \tag{9.5}$$

$$\tilde{\theta}_R = \Psi(R^E, x^E, \dot{x}^E, \theta_\Psi) \tag{9.6}$$

where Ψ indicates the model updating is performed in real-time, θ_Ψ is the parameter being updated through the chosen model updating algorithm, $\tilde{\theta}_R$ is the recursively identified nonlinear model parameters that minimize the associated cost function. The numerical restoring force $R^N(x^N, \dot{x}^N, \tilde{\theta}_R)$ is adapting in real-time based on the physical responses.

9.2 Constrained Unscented Kalman Filter Algorithm

One popular class of system identification methods that can readily be implemented in real-time is the Kalman filter (KF) family. To minimize the expected mean square error of the cost function, the KF estimates the state vector with an optimal gain which can flexibly handle trade-offs between identification accuracy and robustness due to measurement noise and system uncertainties [114], [115]. The original KF algorithm works for linear systems, the extended Kalman Filter (EKF) [116] and the Unscented Kalman Filter (UKF) [117] were developed subsequently for nonlinear systems. With the EKF the estimation is optimized based on linear approximations of the nonlinear system through a Jacobian Matrix, while UKF uses the unscented transformation (UT). A large amount of effort has been dedicated to comparing the performance of EKF and UKF, with conclusions that vary based on the degree of nonlinearity in the selected model and the application of interest [115], [118], [119], [120], [121].

One significant benefit of using UKF is that no Jacobian matrix is needed which avoids the possibility of errors in the derivation. The UKF is extended from estimating the state vector to parameter identification, and has been applied to estimate Bouc-Wen model parameter sets [57] through numerical validation. The result indicate that the UKF can be used effectively for nonlinear structural identification, and further extensions consider experimental validation in [122]. This approach continues to be recognized as being effective for online HS applications [58], [110].

In the Bouc-Wen model series, certain model parameters have physical meaning, such as $\alpha_{B,i}$ which determines the severity of structure nonlinearity and has a range limit of $0 \leq \alpha_{B,i} \leq 1$, $k_{B,i}$ is the initial linear stiffness of the component which should be non-negative. Thus, the nonlinear system identification problem becomes a parameter estimation with interval constraints. The CUKF algorithm has been developed for state estimation with interval constraints, and has been experimentally validated in structural testing applications involving the Bouc-Wen model in RTHS [108], turbofan engines states estimation [123], and tire force, velocity etc. estimation for vehicles [124].

Consider a stochastic nonlinear discrete-time dynamic system:

$$\theta_k = F(\theta_{k-1}, u_{k-1}, k-1) + w_{k-1} \tag{9.7}$$

$$y_k = H(\theta_k, k) + v_k \tag{9.8}$$

where F and H are process and observation models. For a parameter estimation problem, θ_{k-1} is the system parameter vector. Assume for all $k \leq 1$, input u_k, measurement y_k , and the PDFs of $\rho(\theta_0)$, $\rho(w)$, $\rho(v)$ are known. Also, w and v are the process noise and measurement noise, with zero mean and known variances, represented by Q and R. θ_0 is the initial condition (guess) of the parameter estimation vector.

For the state and parameter estimation problem, the goal is to determine the set of θ_k that maximizes the profit function $J(\theta_k) = \rho(\theta k|(y_1...y_k))$. For a nonlinear system, the optimization of J cannot be represented by its mean $\hat{\theta}_{k|k}$ and covariance $P^{\theta|\theta}_{k|k}$ as in the linear KF. Thus, UT is used to best approximate the mean and covariance of y_k with its nonlinear transformation $y_k = H(\theta_k, k)$. The UT is based on projecting the current estimation set θ_k to an additional $2L$ sigma points (L is the number of parameter to be identified) Θ_k, the mean $\hat{\Theta}_k = \sum_0^{j=2L} \gamma_{j,k}\Theta_{j,k}$ with weighting factors γ_j ($j = 1..2L$), and $\sum_{j=1}^{L} \gamma_j = 1$. Unlike the UKF, CUKF projects the sigma points with interval constraints named the interval constraint unscented transformation (ICUT), assuming the given interval constraints $d_k \le \theta_k \le d_k$:

$$\begin{aligned}\Theta_k =& \hat{\theta}_k 1_{1\times 2L+1} + [0_{L\times 1}\omega_{1,k} col_1[(P^{x|x}_{k|k})^{1/2}] ... + \omega_{L,k} col_L[(P^{\theta|\theta}_{k|k})^{1/2}] \\ &- \omega_{L+1,k} col_{L+1}[(P^{\theta|\theta}_{k|k})^{1/2}] ... - \omega_{2L+1,k} col_{2L+1}[(P^{\theta|\theta}_{k|k})^{1/2}]]\end{aligned}$$

where, $(\cdot)^{1/2}$ denotes the Cholesky square root and λ_C is predefined ICUT parameter $\lambda_C > -L$. For $i = 1...2L$ and $j = 1...2L$, $\omega_{jj,k}$ =min($col_j(\Omega)$), Ω is defined:

$$\Omega(i,j) = \begin{cases} \sqrt{L+\lambda_C}, & \text{if } Ui,j = 0, \\ \min(\sqrt{L+\lambda_C}, \frac{e_{i,k}-\hat{x}_{i,k}}{U_{i,j}}), & \text{if } U_{i,j} > 0, \\ \min(\sqrt{L+\lambda_C}, \frac{d_{i,k}-\hat{x}_{i,k}}{U_{i,j}}), & \text{if } U_{i,j} < 0, \end{cases}$$

where $U = [(P^{\theta\theta}_k)^{1/2}, -(P^{\theta\theta}_k)^{1/2}]$, and the time varying weights $\gamma_k = [\gamma_{0,k}...\gamma_{2L+1,k}]$ given by:

$$\gamma_{0,k} = b_k, \gamma_{j,k} = a_k\omega_{i,j} + b_k \tag{9.9}$$

$$a_k = \frac{2\lambda_C - 1}{2(L+\lambda_C)(\sum_{j=1}^{L}\omega_{j,k} - (2L+1)\sqrt{L+\lambda_C})} \tag{9.10}$$

$$b_k = \frac{1}{2(L+\lambda_C)} - \frac{2\lambda_C - 1}{2\sqrt{L+\lambda_C}(\sum_{j=1}^{L}\omega_{j,k} - (2L+1)\sqrt{L+\lambda_C})} \tag{9.11}$$

The ICUT procedure above is then defined using:

$$[\gamma_k, \Theta_k] = \Phi_{ICUT}(\hat{\theta}_k, P^{xx}_k, d_k, e_k, L, \lambda_C) \tag{9.12}$$

Once we have defined the ICUT, the forecast step is given as:

$$[\gamma_{k-1}, \Theta_{k-1|k-1}] = \Phi_{ICUT}(\hat{\theta}_k, P_k^{\theta\theta}, d_k, e_k, L, \lambda_C) \tag{9.13}$$

$$\Theta_{j,k|k-1} = F(\Theta_{k-1|k-1}, u_{k-1}, k-1) + w_{k-1} \tag{9.14}$$

$$\hat{\theta}_{k|k-1} = \sum_{j=0}^{2L} \gamma_{j,k-1}\Theta_{j,k|k-1} \tag{9.15}$$

$$P_{k|k-1}^{\theta\theta} = \sum_{j=0}^{2L} \gamma_{j,k-1}[\Theta_{j,k|k-1} - \hat{\theta}_{k|k-1}][\Theta_{j,k|k-1} - \hat{\theta}_{k|k-1}]^T + Q_{k-1} \tag{9.16}$$

$$[\gamma_k, \Theta_{k|k-1}] = \Phi_{ICUT}(\hat{\theta}_{k|k-1}, P_{k|k-1}^{\theta\theta}, d_k, e_k, L, \lambda_C) \tag{9.17}$$

$$\mathcal{Y}_{j,k|k-1} = H(\Theta_{j,k|k-1}, k) \tag{9.18}$$

$$\hat{y}_{k|k-1} = \sum_{j=0}^{2L} \gamma_{j,k}\mathcal{Y}_{j,k|k-1} \tag{9.19}$$

$$P_{k|k-1}^{yy} = \sum_{j=0}^{2L} \gamma_{j,k}[\mathcal{Y}_{j,k|k-1} - \hat{y}_{k|k-1}][\mathcal{Y}_{j,k|k-1} - \hat{y}_{k|k-1}]^T + R_k \tag{9.20}$$

$$P_{k|k-1}^{\theta y} = \sum_{j=0}^{2L} \gamma_{j,k}[\Theta_{j,k|k-1} - \hat{\theta}_{k|k-1}][\mathcal{Y}_{j,k|k-1} - \hat{y}_{k|k-1}]^T \tag{9.21}$$

$P_{k|k-1}^{\theta\theta}$ is the forecast error covariance, $P_{k|k-1}^{yy}$ is the innovation covariance, $P_{k|k-1}^{\theta y}$ is the cross covariance. Similar to classic KF update, the estimate step (also known as data assimilation step) is defined:

$$K_k = P_{k|k-1}^{\theta y}(P_{k|k-1}^{yy})^{-1} \tag{9.22}$$

$$\hat{\theta}_{k|k} = K_k(y_k - \hat{y}_{k|k-1}) \tag{9.23}$$

$$P_{k|k}^{\theta\theta} = P_{k|k-1}^{\theta\theta} - K_k P_{k|k-1}^{yy} K_k^T \tag{9.24}$$

where K^k is the Kalman gain matrix and $P_{k|k}^{\theta\theta}$ is the data-assimilation error-covariance.

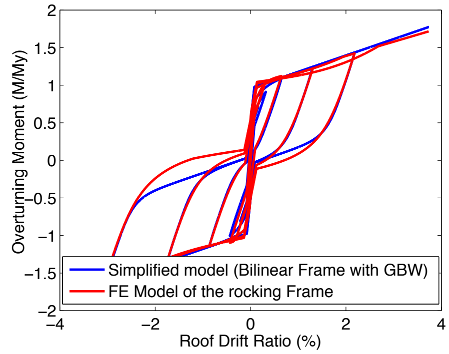
In summary, to apply the CUKF user-defined parameters include the initial guess of the parameter vector θ_0, the ICUT parameter λ_C, the process noise variance Q

and measurement noise variance R. The choices of those parameters may affect the accuracy of system identification as discussed subsequently.

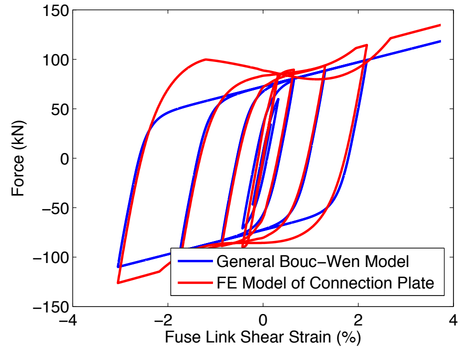
9.3 Numerical Example

The numerical example for demonstrating HSMU is a controlled, self-centering frame proposed by Deierlein *et al*, [111], [2], [113], [3] which is composed of two steel frames, vertical post tension strands in both frames, and the two frames are linked by three identical shear connection as shown in Fig. 9.1(a). The design concept is as follows: 1) the steel frames with post tension strands act as a two stage linear (bilinear) frame determined by the initial force in the strands and the top displacement; 2) the vertical post-tension strands provide the self-centering force; and, 3) the replaceable shear connection plates dissipate energy during dynamic loading. It is considered to be a very practical application of hybrid simulation analysis because the bilinear behavior of the frames with PT strands can be isolated in the numerical substructure. Meanwhile, model updating is beneficial since all the connection plates are identical and experience similar loading paths.

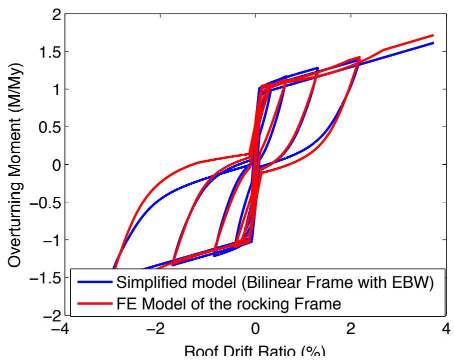
To demonstrate the HSMU approach, the structure is represented with a bilinear frame model and GBW/EBW models as presented in chapter 8. The simplified model responses are compared to the OpenSees finite element model available within the NEES data repository [112]. The resulting comparison is provided in Fig. 9.2, and indicates a slight difference in the hysteresis behavior (see Fig. 9.2(b) and 9.2(d)). At very large displacements the shear plate will deform out-of-plane and this type of hysteresis is not captured by the BW model family. Here the parameters for both the GBW and EBW models are determined through genetic algorithms by minimizing the error between the BW model response and finite element model response. The nominal values of each parameter are listed in Table 9.1 and 9.2. The simplified model with this nominal parameter set are assumed in the sequel as the reference solution in the following analysis which also defines the true parameters for the specimen model.



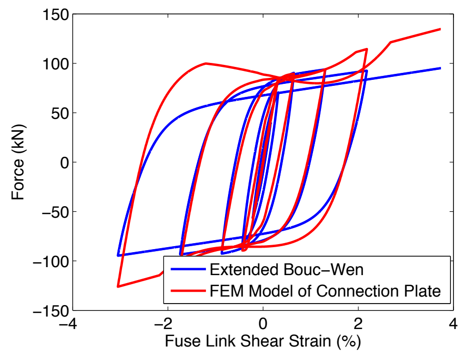
(a) Frame Response using GBW model



(b) Connection Plate Response using GBW model



(c) Frame Response using EBW model

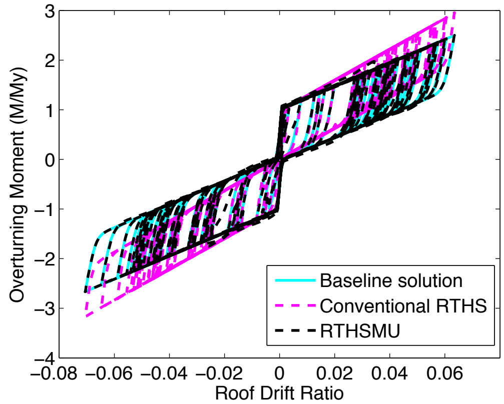


(d) Connection Plate Response using EBW model

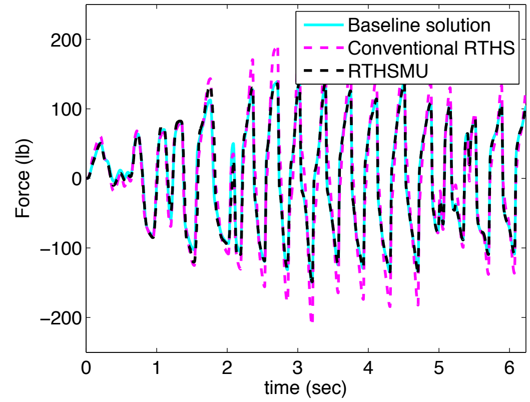
Figure 9.2.: Comparison between GBW/EBW Model and FEM Responses

HSMU efficiency is qualitatively demonstrated through a simulation of the case study following the concept described in Fig. 9.1(b). In this HSMU simulation the third story connection plate is assumed to be loaded as the experimental substructure in hybrid simulation, and also modeled using the specimen model parameter (model with nominal parameters) with its input and response available for system identification. The other two plates (known as the target models) have initial modeling errors as they are modeled using BW models with 50% error in each parameters. This model will adopt the estimated parameter sets generated by the model updating algorithm. The HSMU results are compared to both a) the baseline simulation result (**reference solution**) where no modeling error is assumed and b) a simulation representative of **conventional hybrid simulation**, where only the third story connection plate (specimen model) is accurately loaded and the two remaining plates are modeled numerically with erroneous parameters (50% error) and do not adopt the updated parameters.

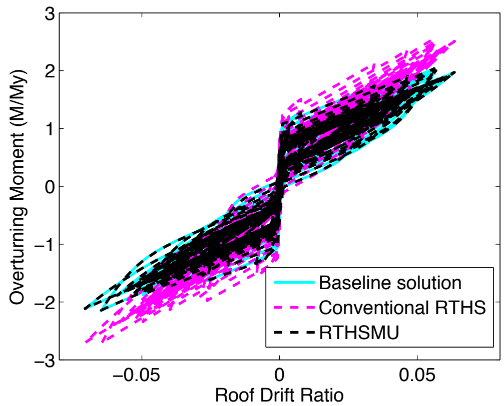
The entire self-centering system (two frames, PT strands and connection plates) is simulated using a scaled El-Centro earthquake excitation, and a comparison of the results is shown in Fig. 9.3. It is clear that significant error exists in the global response (top displacement-moment of the self-centering frame) using conventional hybrid simulation when there is modeling error in the target model. The HSMU response initially exhibits the same conditions as the conventional hybrid simulation, however, after several seconds it converges to the baseline solution as in Fig. 9.3(b). For the EBW model, the convergence is achieved even faster, inside of 1 second, with the initial model parameters placed 50% away from the true parameters. This result indicates the attractive benefit offered by HSMU which improves the hybrid simulation accuracy without the need for testing all instances of components with modeling uncertainties. This general finding has also been reported in [56], [108], [58], [110].



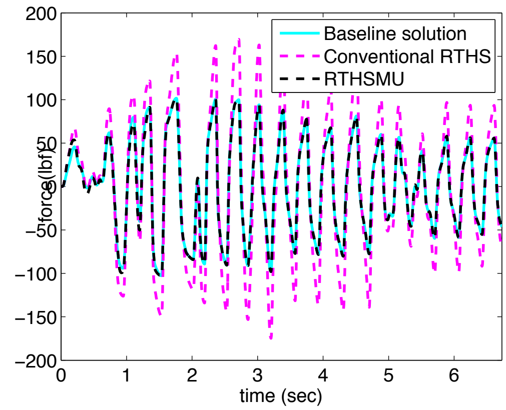
(a) Frame Response Comparison, GBW



(b) Response from 2^{nd} Story Connection Plate, GBW



(c) Frame Response Comparison, EBW



(d) Response from 2^{nd} Story Connection Plate, EBW

Figure 9.3.: Comparison between Baseline Solution, Conventional RTHS and RTHSMU

9.4 Model Sensitivity Study

The accuracy of the system identification will first depend on the sensitivity of the system response to each parameter in the model. As stated earlier, in Bouc-Wen models, hysteretic behavior has different sensitivity to different parameters, it would be difficult to obtain a good estimate of the global solution if each of the model parameters do not participate in the response. Therefore, a sensitivity analysis for model parameters are needed. Such an analysis is normally implemented numerically based on the variation of a given parameter set (base value) to determine the quantitative change in the output. This is known as a local sensitivity analysis when each parameter is considered in isolation. When all of the parameters vary simultaneously in a given range, and the ranking of each is evaluated, it is referred to as a global sensitivity analysis. Because the analysis is performed numerically, the result (sensitivity ranking) depends on the choice of the base value set. The base value sets chosen in this analysis are determined using the finite element model of the shear plate and is demonstrated in Fig. 9.2(b) and 9.2(d).

9.4.1 General Bouc-Wen Model Sensitivity Study

The predetermined base values of the GBW are in Table 9.1. In both local and global analyses, a driving input consisting of a 0.5 Hz sine wave is used and the hysteretic behavior of the BW model with the base value set is demonstrated in Fig. 9.4(a). In the local sensitivity analysis, each single parameter gradually changes within the range of [0.5 1.5] of its base value and the remaining parameters retain their base values. Fig. 9.4(b) demonstrates the variation in the system response as a result of the change in several parameters. The results are represented in terms of the associated RMS error, and the local sensitivity ranking can be calculated accordingly as shown in Table 9.1.

The RMS error is defined as:

$$E_{RMS} = \sqrt{\frac{\Sigma_i^n (R_V - R_{Base})^2}{n}} / \sqrt{\frac{\Sigma_i^n (R_{Base})^2}{n}} \tag{9.25}$$

In the global sensitivity analysis, all parameters vary simultaneously. Sobol indices [125] are used for evaluate parameter global sensitivity, as suggested in [103].

Consider an integralable function $f(x)$ defined in I^m which can be represented as:

$$f(x) = f_0 + \sum_{s=1}^{L} \sum_{i_1<...<i_s}^{L} f_{i_1...i_s}(x_{i_1}, ..., x_{i_s}) \tag{9.26}$$

$$f(x) = f_0 + \sum_{i} f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j) + ... + f_{12...n}(x_1, x_2, ..., x_L) \tag{9.27}$$

where, L is the number of parameters involved. The total summation is 2^L. Assume Eq.9.26 is the ANOVA representation of $f(x)$, also called decomposition into summands of different dimensions [126], the square integral of $f(x)$ is:

$$\int f^2(x)dx - f_0^2 = \sum_{s=1}^{L} \sum_{i_1<...<i_s}^{L} f_{i_1...i_s}^2 dx_{i_1 1}...dx_{i_s} \tag{9.28}$$

Assume constant $D = \int f^2(x)dx - f_0^2$, $D_{i_1...i_s} = \int f_{i_1...i_s}^2 dx_{i_1 1}...dx_{i_s}$ are called variances.

$$D = \sum_{s=1}^{L} \sum_{i_1<...<i_s}^{L} D_{i_1...i_s} \tag{9.29}$$

Then the global sensitivity index is defined as:

$$D = \sum_{s=1}^{L} \sum_{i_1<...<i_s}^{L} D_{i_1...i_s} \tag{9.30}$$

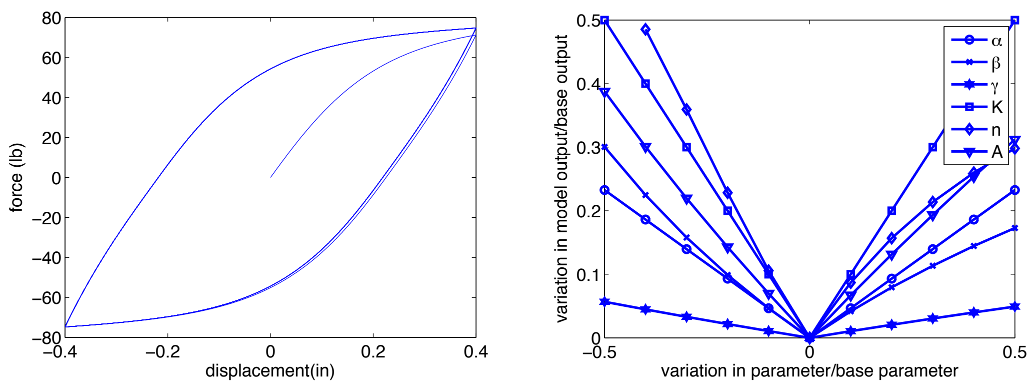
The integer s is often called the order or the dimension of the index. All the $s_{i_1...i_s}$ are non-negative and their summation equals one. For each sensitivity analysis, $2^L - 1$ indices are calculated and the effect of one single parameter S_i can be defined as:

$$S_j = 1 - \sum_{i_k \neq j} S_{i_1...i_s} \tag{9.31}$$

In the global sensitivity analysis, a Monte Carlo simulation of 4000 runs are implemented with each parameter selected randomly within [0.5 1.5] range of its base value, and the input signal remains the same as in the local sensitivity analysis. The result of S_j of each parameter in the GBW is listed in Table 9.1.

Table 9.1.: General Bouc-Wen Model Sensitivity Ranking

Parameter	Base Value	Local Ranking	S_i	Global Ranking
α_{GBW}	0.35	5	7.96 %	5
k_{GBW}	35	2	31.8%	1
A_{GBW}	15	3	10.39%	4
β_{GBW}	1.7	4	18.04%	3
n_{GBW}	1.7	1	28.31%	2
γ_{GBW}	0.5	6	2.45%	6



(a) Representative Hysteretic Behavior of GBW model

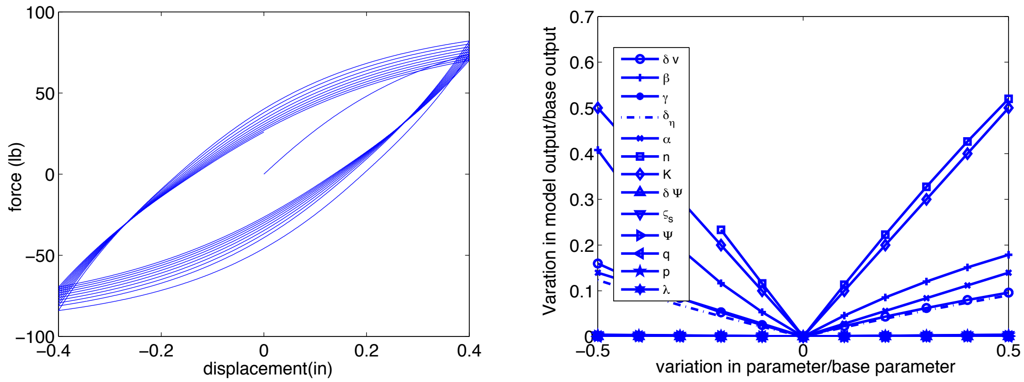
(b) Local Parameter Sensitivty of GBW model

Figure 9.4.: General Bouc-Wen Model Sensitivity

9.4.2 Extended Bouc-Wen Model Sensitivity Study

The local and global sensitivity analysis are similarly conducted using EBW model, an input of 0.5 Hz, and the hysteretic behavior of the base value set is demonstrated in Fig. 9.5(a). The predetermined base values of the extended Bouc-Wen model are shown in Table 9.2 and the resulting local sensitivity plot is shown in Fig. 9.5(b) with a variation of ±50% in each parameter. The results of the EBW model are summarized in Table 9.2. As stated in chapter 8, for phenomenological models, initial parameters can have large error compared to the true estimation, thus, assuming 50% error in each parameter is reasonable.

The sensitivity analysis of the GBW indicates that each parameter contributed more evenly to the global response, which indicates that a parameter identification



(a) Representative Hysteretic Behavior of Extended BW model

(b) Local Parameter Sensitivty of Extended BW model

Figure 9.5.: Extended Bouc-Wen Model Sensitivity

Table 9.2.: Extended Bouc-Wen Model Sensitivity Ranking

Parameter	Base Value	Local Ranking	S_i	Global Ranking
α_{EBW}	0.03	4	0.0941	4
k_{EBW}	291.67	2	0.39	1
β_{EBW}	4.98	3	0.14	3
n_{EBW}	1.41	1	0.18	2
$\delta_{\eta EBW}$	0.33	6	0.0283	7
$\delta_{\nu EBW}$	0.12	5	0.0421	5
q_{EBW}	0.011	9	0.0139	10
γ_{EBW}	0.089	7	0.0024	12
ζ_{sEBW}	0.13	8	0.0212	9
p_{EBW}	0.023	10	0.0226	8
Ψ_{EBW}	0.64	11	0.0031	11
$\delta_{\Psi EBW}$	0.016	12	0.0313	6
λ_{EBW}	0.18	13	0.001	13

performed using the GBW model will result in relatively more accurate parameters. With the EBW model, specific parameters, k_{EBW}, n_{EBW}, β_{EBW}, α_{EBW}, $\delta_{\nu EBW}$, $\delta_{\eta EBW}$ are shown to be more dominant in the hysteretic response, and thus accurate identification of the entire parameter set may be challenging in some cases.

9.5 Model Updating Performance in HSMU

The success of HSMU depends on the model updating performance. Besides typical factors to be considered in the model updating accuracy analysis, including parameter initial condition θ_0, model updating algorithm variable (Q, R and λ_c etc. for CUKF), other concerns specific to HSMU should be examined. One limitation of HSMU is the incomplete excitation that is important for system identification [127].

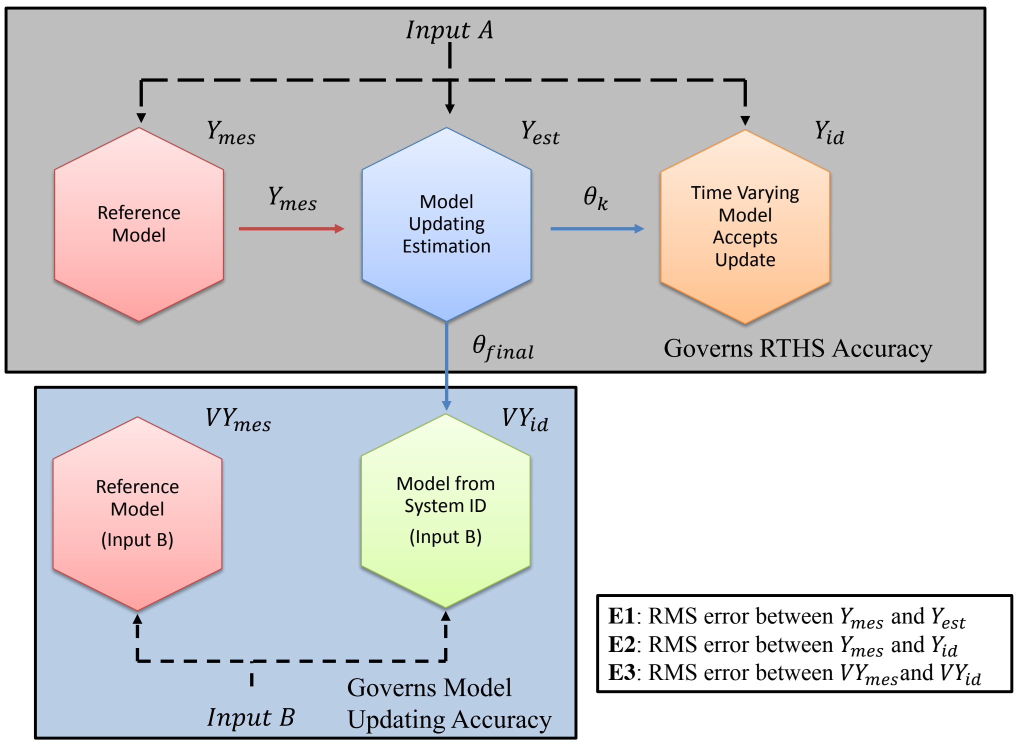


Figure 9.6.: Local Accuracy Measures for HSMU

9.5.1 Model Updating Accuracy with Different Inputs

In HSMU, the specimen to be identified is driven by the local structural response which is a combination of several sinusoidal signals with irregular amplitudes at the structures dominant modes. Meanwhile, the target model adopts structural parameters in real-time, and also generates the response of that model to a given input. Thus, it is important to evaluate HSMU feasibility through 1) understanding the effect of information richness (frequency and amplitude) on identification performance, 2) assessment of the time-varying model output accuracy that adopts the online estimation parameter set, and 3) validating the accuracy of HSMU using an alternate excitation. For hybrid simulation applications, the specimen is normally not rate dependent (frequency irrelevant), information richness is more related to amplitude.

Three levels of accuracy measures are proposed, as shown in Fig. 9.6, Error indicator 1 (E1) evaluates the **estimation output error** between CUKF estimated output (Y_{est}) and the reference model output (measurement Y_{mes}), E2 indicates the **model updating output error** between the time varying model response (Y_{id}) and the reference model output ($Y_m es$), which is the main indicator to judge the HSMU success. Both E1 and E2 are calculated after model updating algorithm runs 4 sec assuming steady state solution achieved. E3 is defined with an alternate input which is sent to A) the specimen model (with true parameter) and B) a new model with identified parameter (after model updating procedure), to compute the error between VY_{mes} and VY_{id}, representing the **system identification error**.

Three input signals are considered in the excitation completeness analysis (Table 9.3).

Table 9.3.: Simulation Cases for Different Inputs

Case No.	Excitation A	Excitation B
A	Structural response after earthquake	BLWN
B	0.1 Hz sine wave with amplitude 1 inch	BLWN
C	0.1 Hz sine wave with amplitude 0.1 inch	BLWN
D	0.1, 0.5, 3 Hz sine wave with amplitude 1 inch each	BLWN

General Bouc-Wen Model

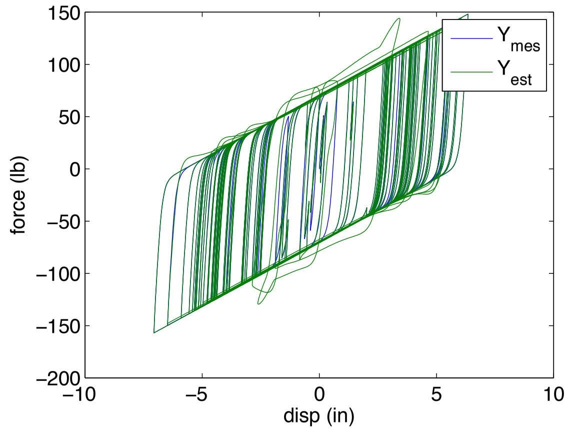
The nominal (true) parameter for the GBW case is given in Table 9.1. The erroneous initial condition used in the target model for model updating has 50% over-estimation in each value. The results shown in the excitation completeness analysis are computed using the best possible combination of Q, R, λ_C later in the chapter.

Table 9.4.: Error Indices for Excitation Completeness Analysis, GBW

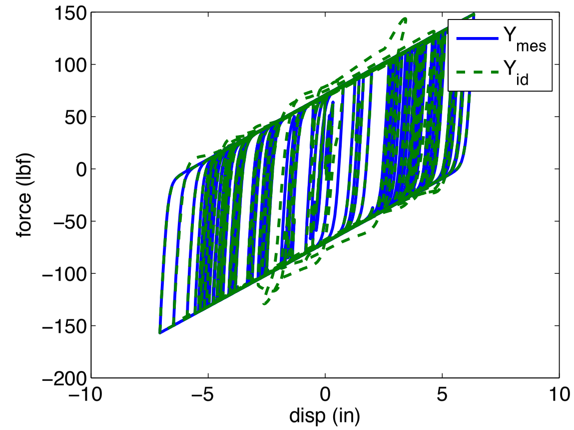
Cases	E_1	E_2	E_3
A	0.32%	0.57%	2.0 %
B	1.23%	1.23%	**9.33** %
C	2.01%	2.41%	**22.9** %
D	3.57%	3.55%	2.02%

The model updating results with GBW are shown in Fig. 9.7 - 9.10 and the quantitative indices are in Table 9.4. For both case A (BLWN) and D (structural response), all three errors are quite small. If the excitation information provides sufficient information in identifying component hysteresis, system estimation error E3 is reduced. However, for simulation with single sinusoidal input (case B and C), both E1 and E2 have small value which illustrates the CUKF algorithm is able to estimate a set of model parameters that satisfy the minimum error tolerance, and the time-varying target model using the identified parameter sets Y_{id} provides an accurate response as compared to Y_{mes}. However, the E3 error is much larger using an alternate validation signal (BLWN in these cases). Specially in case B, where the excitation peak displacement is much smaller (0.3 inch) compared to the validation signal peak of 8 inch, the system identification error E3 is the largest. With the incomplete excitation, the model updating algorithm generates a local optimal solution specific to the excitation, which cannot represent the component behavior in general. Thus, it is concluded that not all hybrid simulation cases are suitable for incorporating model updating, and some numerical simulation about model updating behavior in a specific hybrid simulation with a specific component model should be examined. This phenomenon is expected to be even more dominant in RTHS if the specimen model is rate dependent, such as the MR damper.

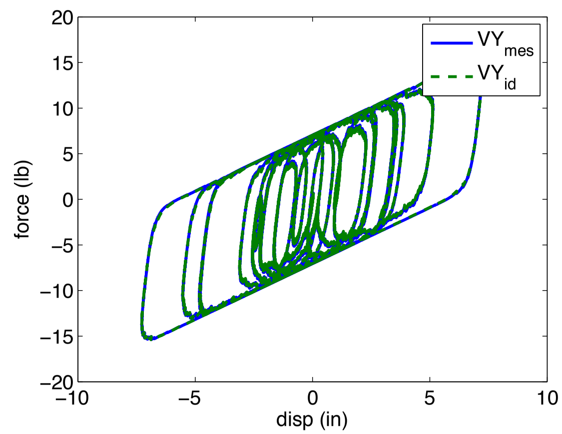
Fig. 9.11 shows the parameter convergence. Cases A and D are assumed to have settled to its global optimal solution indicated by E3. However, although the solution converges after 5 seconds, the final parameters are different from the nominal set. This result may be due to the known fact that the solution of Bouc-Wen model is not unique, as discussed in [128], [129]. Meanwhile, if the excitation signal is not complete, the parameter set is likely to oscillate periodically and not truly converge, as in case C. This is related to both the excitation signal incompleteness and also the choice of different model updating parameter, Q, R, λ_c, etc. Thus, a model updating parameter study is discussed later.



(a) E1-Case A

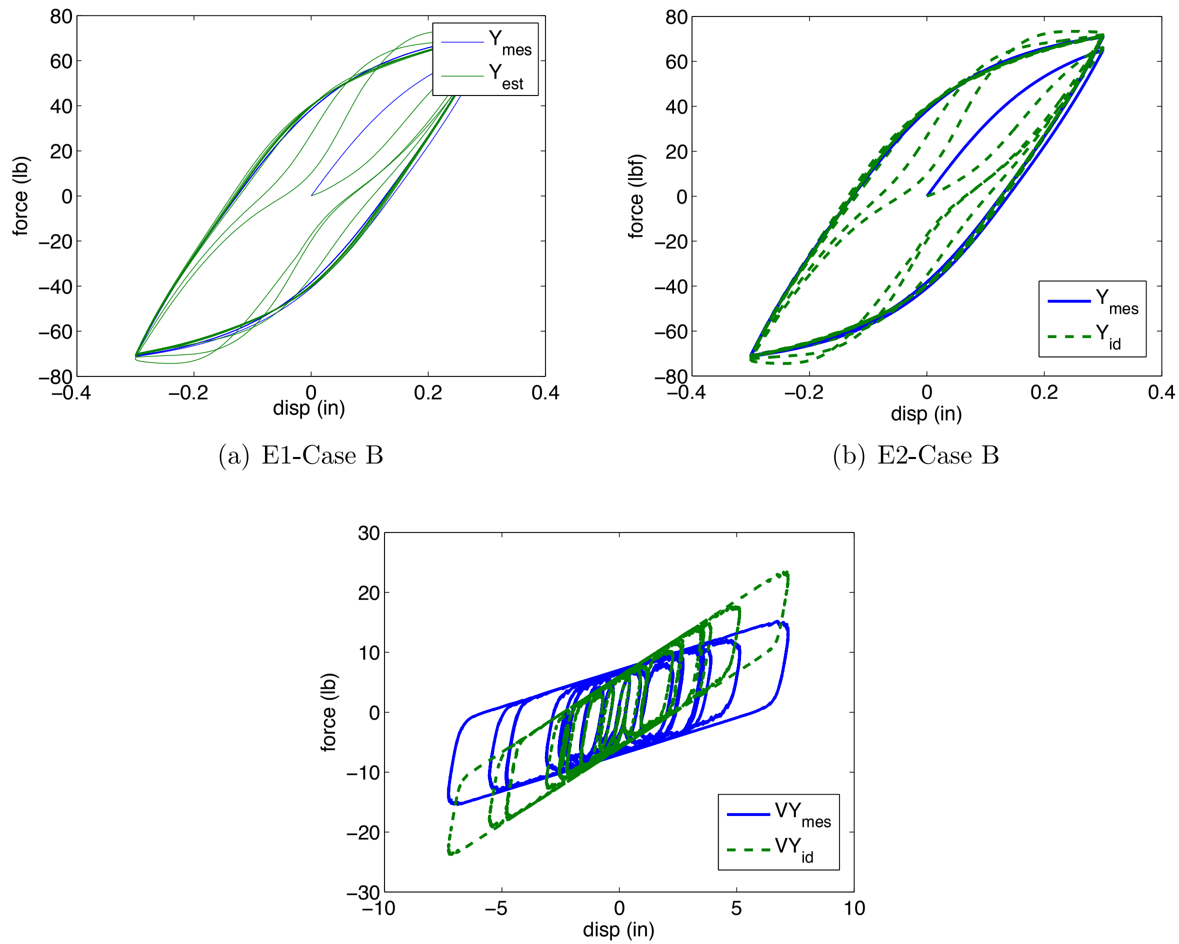


(b) E2-Case A



(c) E3-Case A

Figure 9.7.: Real-Time Model Updating Performance and Validation Using Various Excitations, GBW Model (Case A)

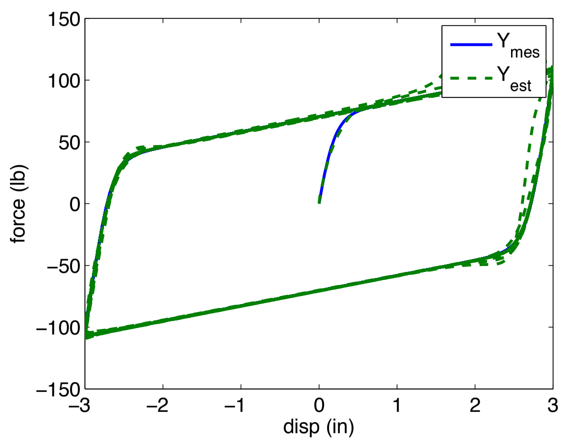


(a) E1-Case B

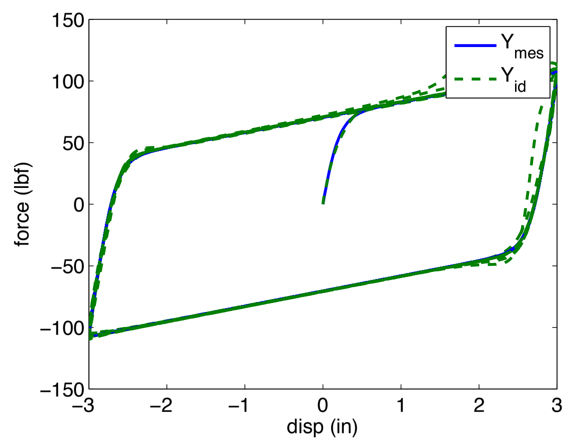
(b) E2-Case B

(c) E3-Case B

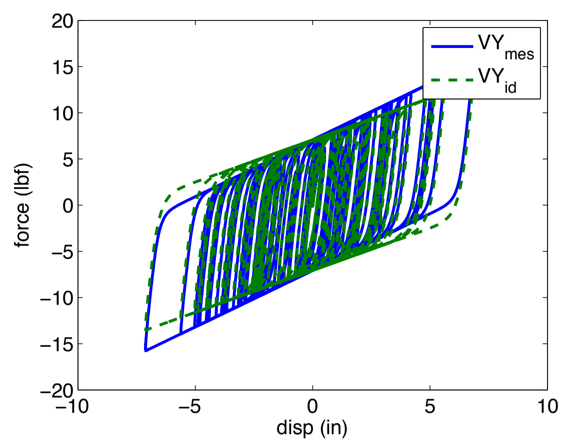
Figure 9.8.: Real-Time Model Updating Performance and Validation Using Various Excitations, GBW Model (Case B)



(a) E1-Case C

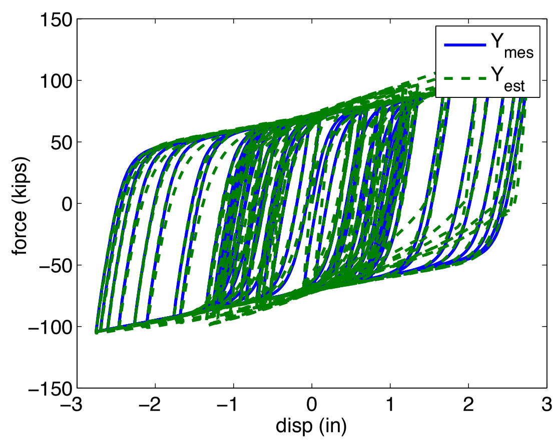


(b) E2-Case C

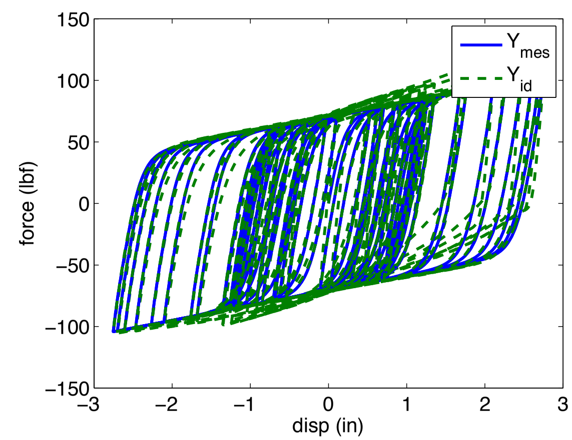


(c) E3-Case C

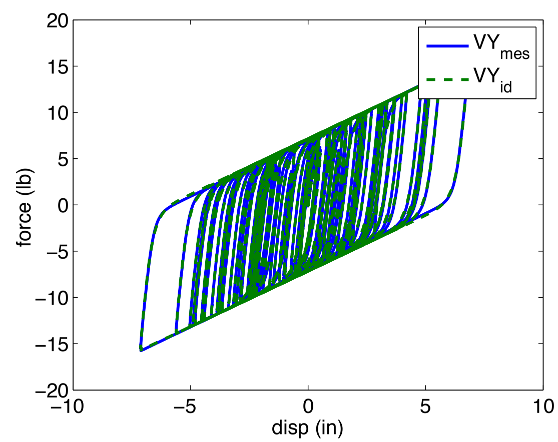
Figure 9.9.: Real-Time Model Updating Performance and Validation Using Various Excitations, GBW Model (Case C)



(a) E1-Case D

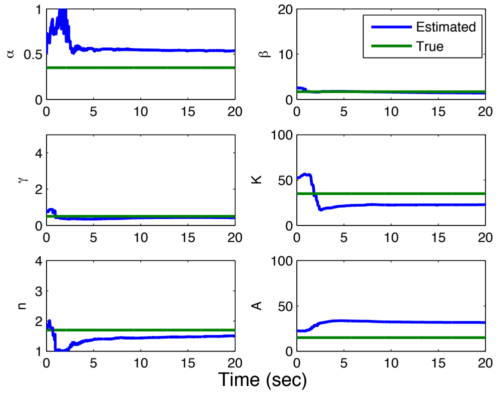


(b) E2-Case D

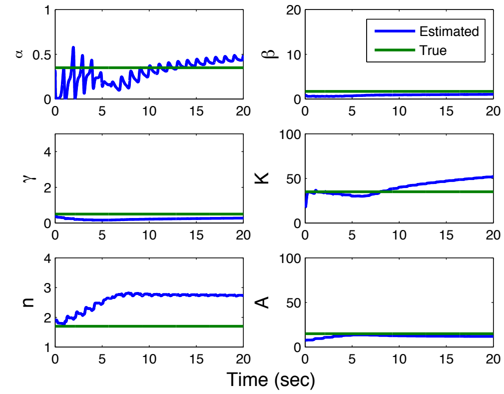


(c) E3-Case D

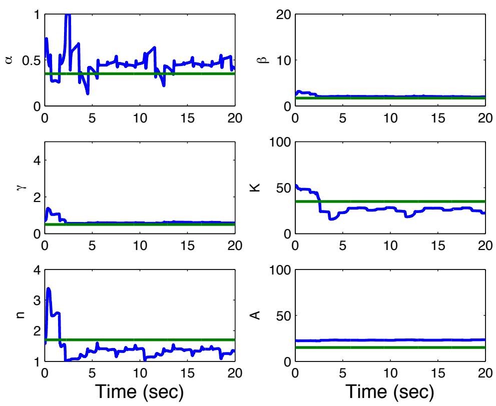
Figure 9.10.: Real-Time Model Updating Performance and Validation Using Various Excitations, GBW Model (Case D)



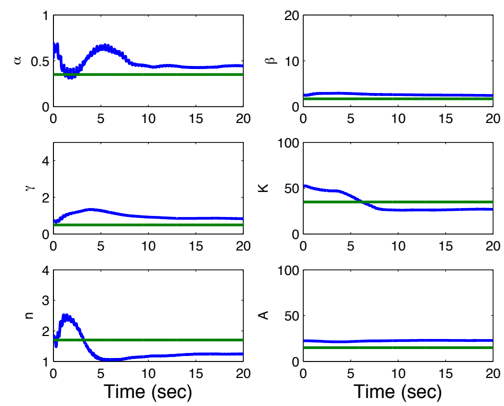
(a) Parameter Convergence - Case A



(b) Parameter Convergence - Case B



(c) Parameter Convergence - Case C



(d) Parameter Convergence - Case D

Figure 9.11.: Parameter Convergence for Excitation Completeness Analysis, General BW model

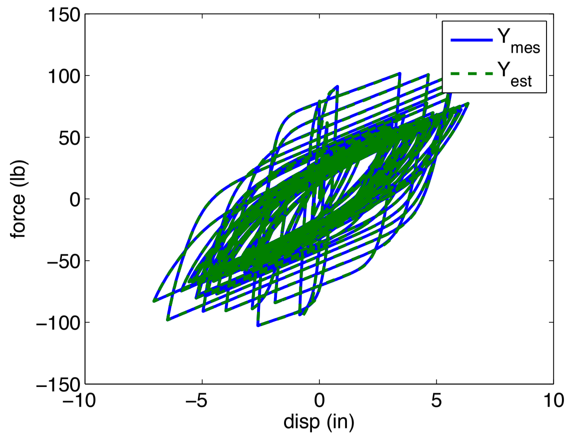
Extended Bouc-Wen Model

A similar study is performed using the EBW model and a 50% over estimation in each parameter is used. The performance of the model updating procedure for the EBW is shown in Fig. 9.12 - 9.15, and the quantitative indices are in Table 9.5. For the EBW model, only the first six parameters which have higher rank in the sensitivity analysis are considered in the parameter convergence analysis in Fig. 9.16.

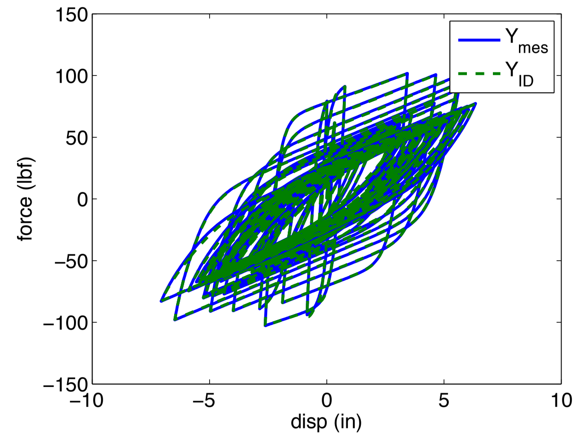
Table 9.5.: Error Indices for Excitation Completeness Analysis, EBW

Cases	E_1	E_2	E_3
A	0.29%	0.916%	6.64%
B	0.46763%	0.7632%	27.4%
C	0.32%	1.63%	7.4%
D	0.48%	0.57%	2.308%

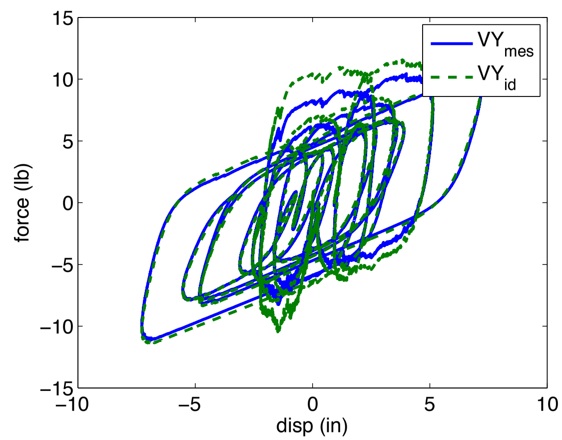
Similar conclusions can be made from EBW model analysis, results indicate estimated error indicator $E2$ and model updating output error index E_2 are small for all simulation cases. Larger system identification error $E3$ is observed for limited excitation signal B and C. When the excitation peak is much smaller compared to the validation signal peak, the estimation accuracy is low. Also, larger system identification error is observed in case A (6.64%) as compared to the one in GBW (2.0%). In this case, the identified parameter set does not converge to the nominal values. There are a couple of possible reasons for this. First, the EBW model with the same hysteresis behavior may be represented by several global solutions (i.e. the solution is not unique). Second, based on the sensitivity analysis, the contribution of the less sensitive parameter is less than 1% using the specific base value set. It is challenging to estimate those parameters that participate to a lesser extent in the response using any system identification algorithm, thus the true parameter set (true global solution) is very challenging to reach. However, even with limited input information and residual error in the parameter estimation, the system identification error (E3) here is still considered to be quite tolerable for most purposes.



(a) E1-Case A

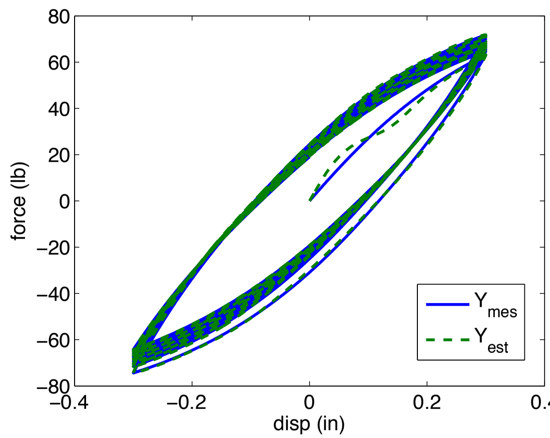


(b) E2-Case A

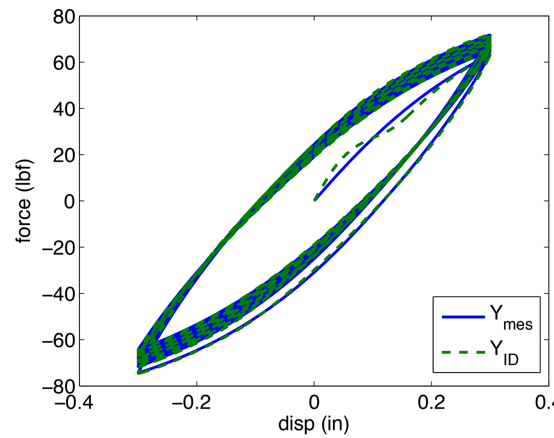


(c) E3-Case A

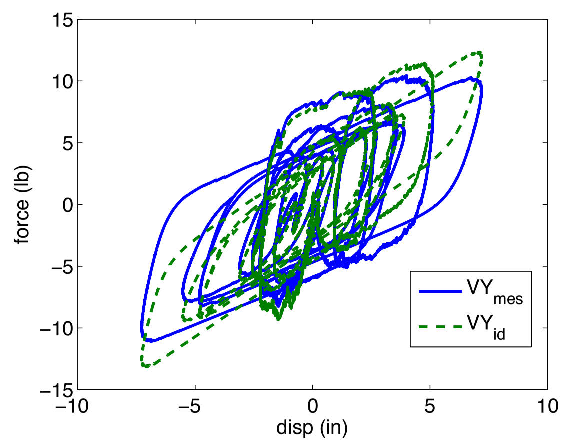
Figure 9.12.: Real-Time Model Updating Performance and Validation Using Various Excitations, EBW Model (Case A)



(a) E1-Case B

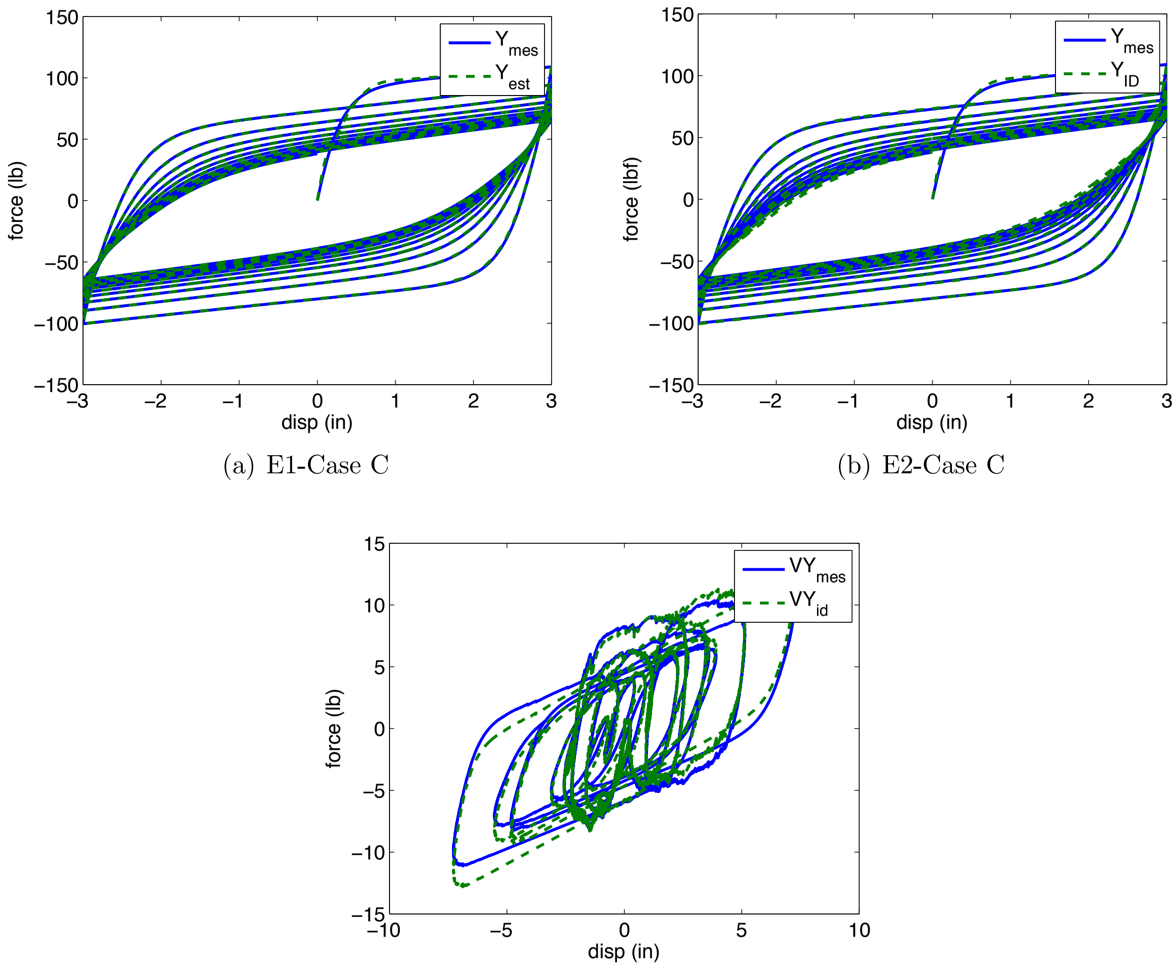


(b) E2-Case B



(c) E3-Case B

Figure 9.13.: Real-Time Model Updating Performance and Validation Using Various Excitations, EBW Model (Case B)

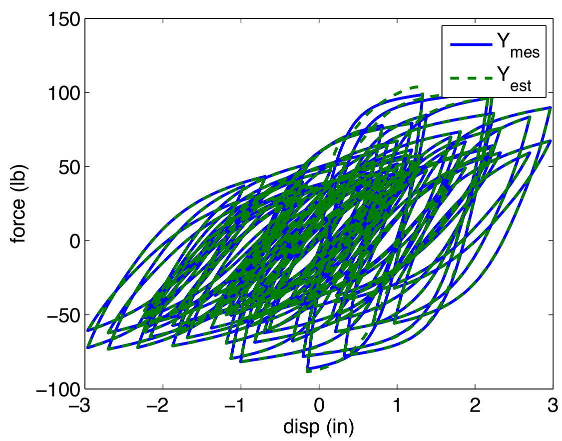


(a) E1-Case C

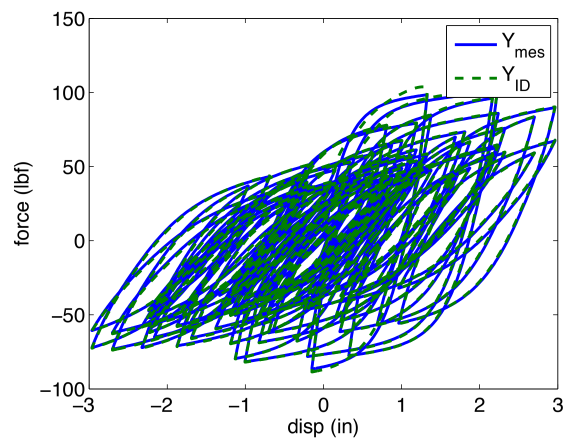
(b) E2-Case C

(c) E3-Case C

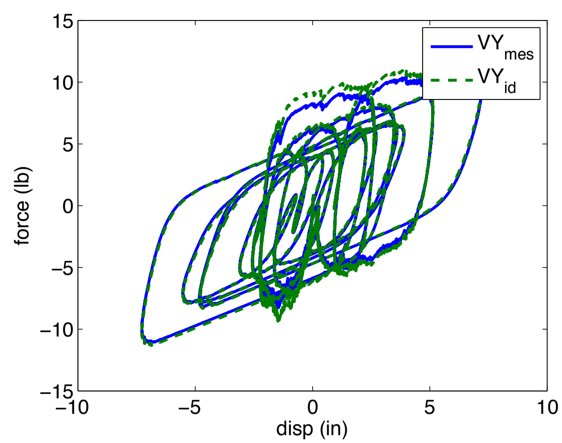
Figure 9.14.: Real-Time Model Updating Performance and Validation Using Various Excitations, EBW Model (Case C)



(a) E1-Case D

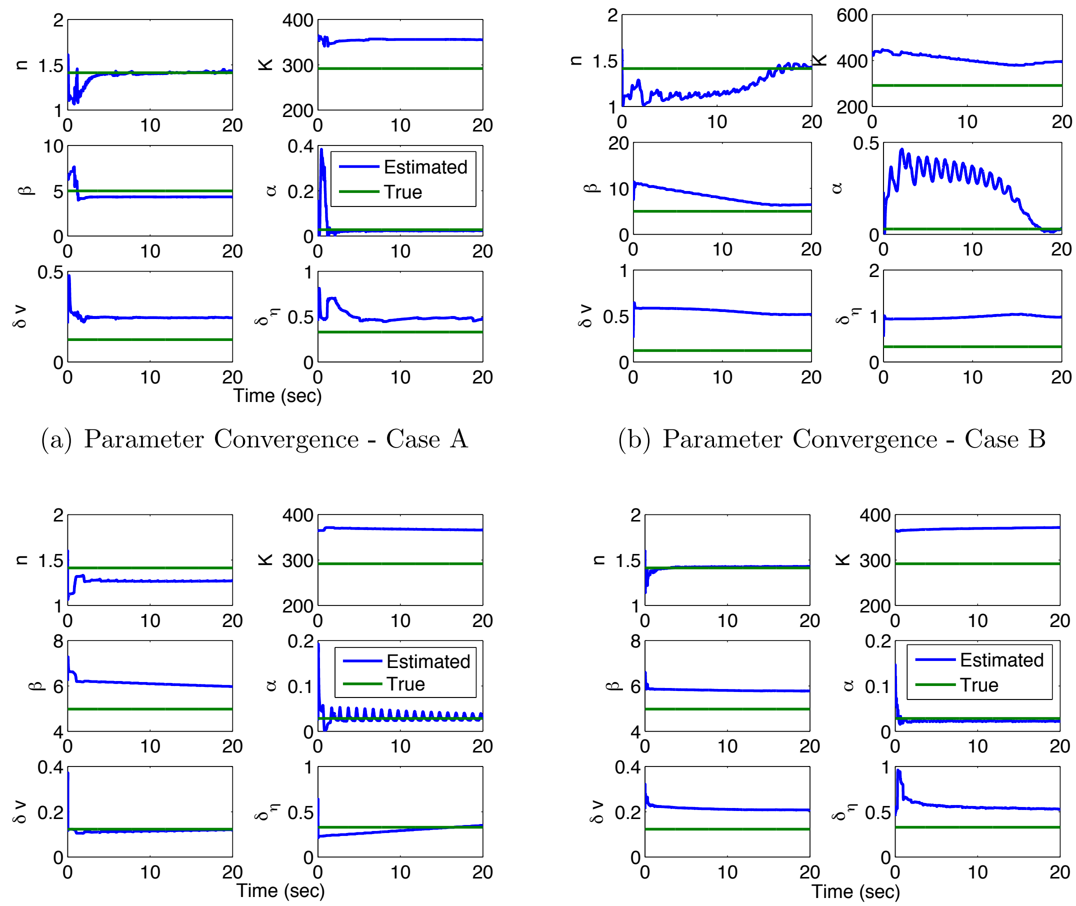


(b) E2-Case D



(c) E3-Case D

Figure 9.15.: Real-Time Model Updating Performance and Validation Using Various Excitations, EBW Model (Case D)



(a) Parameter Convergence - Case A (b) Parameter Convergence - Case B

(c) Parameter Convergence - Case C (d) Parameter Convergence - Case D

Figure 9.16.: Parameter Convergence for Excitation Completeness Analysis, Extended BW Model

Generally for hybrid simulation applications, if the initial numerical model is sufficiently accurate, model updating may not be necessary. For those cases where the initial model is less predictable and may be far away from the true behavior yielding more than 20% error, HSMU will improve the performance of the hybrid simulation in practice even though some estimation error may be present. Also, the choice between using a simpler model (GBW) or a more complex model (EBW) should be made with these challenges in mind. Relevant factors include the different physical specimen properties, the richness of the excitation and the participation of various model parameters, etc. are worth taking into consideration in the hybrid simulation planning stage.

9.5.2 Parameter Study of the Model Updating Algorithm

To examine the effects of the CUKF parameters (noise covariance matrices Q, R, initial parameter guess Θ_0 and parameter determine UT weighting λ) on model updating accuracy, we focus on case A with an initial excitation A of 0-1 Hz BLWN. The three errors defined in the excitation completeness study are computed. Both the GBW and EBW are considered, bur because the conclusion are found to be similar for both GBW and EBW, only GBW case is discussed.

Effect of Q and R Parameters on Model Updating Performance

For all methods in KF family, the error covariance matrices Q and R governs the accuracy and robustness of the estimation. Q is the process noise covariance for state estimation, and in the parameter estimation case Q dominates the rate of convergence in identifying the parameters. R is the measurement noise covariance, and in the identification case study R value sets the tolerable error (error threshold) between CUKF output estimation Y_{est} and the measurement Y_{mes}.

Without loss of generality, for this parametric study the measurement noise covariance is defined as $R = 10^{R_f}$ and the process noise covariance matrix takes the form of a diagonal matrix as:

$$Q = 10^{Q_f} \times \begin{pmatrix} \theta_0(1) & 0 & \cdots & 0 \\ 0 & \theta_0(2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \theta_0(L) \end{pmatrix}$$

Thus, the effect of the matrices Q and R is further simplified using the process noise factor Q_f and the measurement noise factor R_f, where the θ_0 is the initial guess of the parameter set which acts as a weighting factor based on the possible scale of each parameter.

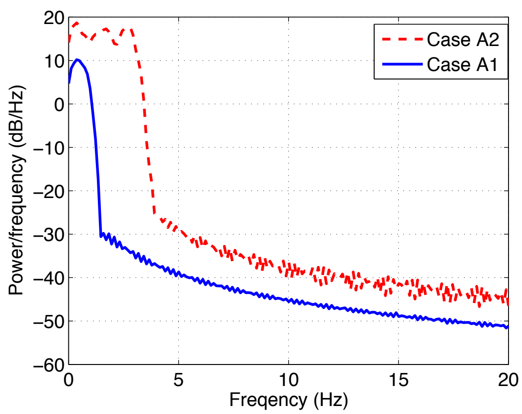
To examine the effect of the input on the choice of Q and R, two BLWN input cases are considered as indicated in Table 9.6 and Fig. 9.17(a). The results are shown in Fig. 9.17 for the GBW case and demonstrate that the accuracy of model updating (E3) depends strongly on the choice of Q_f. A smaller Q_f yields faster convergence (less variation in each parameter with time), and the steady state value of Q_f is reached around $Q_f = -7$ for both input cases. Trade-offs in accuracy occur when using larger R_f values for all error indices as shown in Fig. 9.17(c). For small R_f values Q_f may need tuning, as with $R_f = -1$, only a few cases, $Q_f = -7, -6, -5$ (solid blue line in Fig 9.17(b).) run successfully. A similar trend is obtained with case A2 when $R_f = 1$ (solid red line).

In the EBW case, a significant trend is also observed in the estimation output error indicator E1 as shown in Fig. 9.18(a). For larger Q_f E1 is reduced to under 2% for $R_f = 0$. When R_f is larger, a difference is present but to a lesser degree. The pattern in the model updating error indicator (E3) is less clear (Fig. 9.18(b)). As in the excitation incompleteness analysis conducted in the prior section, the global minima may be difficult to reach. Thus, the final parameter sets obtained with different Q_f may be affected by the choice of Q_f and R_f, and are simply several local minima.

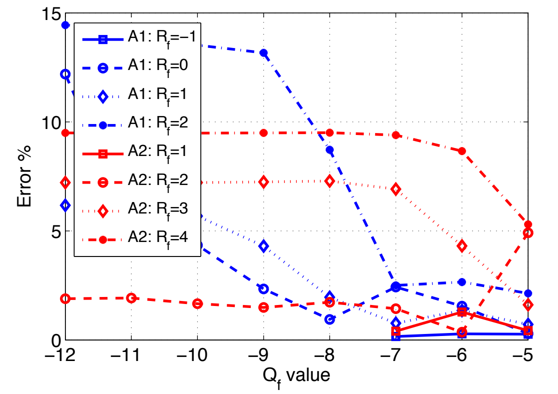
In general, an increase in R_f indicates a higher tolerance for estimation error and thus the model updating process tends to be more stable. The results here indicate that there is a co-dependence between R_f and Q_f, and it is worth mentioning that, depending on the test purpose and scenario (level of model uncertainty, convergence speed requirement, input signal and output signal magnitude), the absolute value of R_f and Q_f should be carefully examined.

Table 9.6.: Simulation Cases for Q and R Analysis

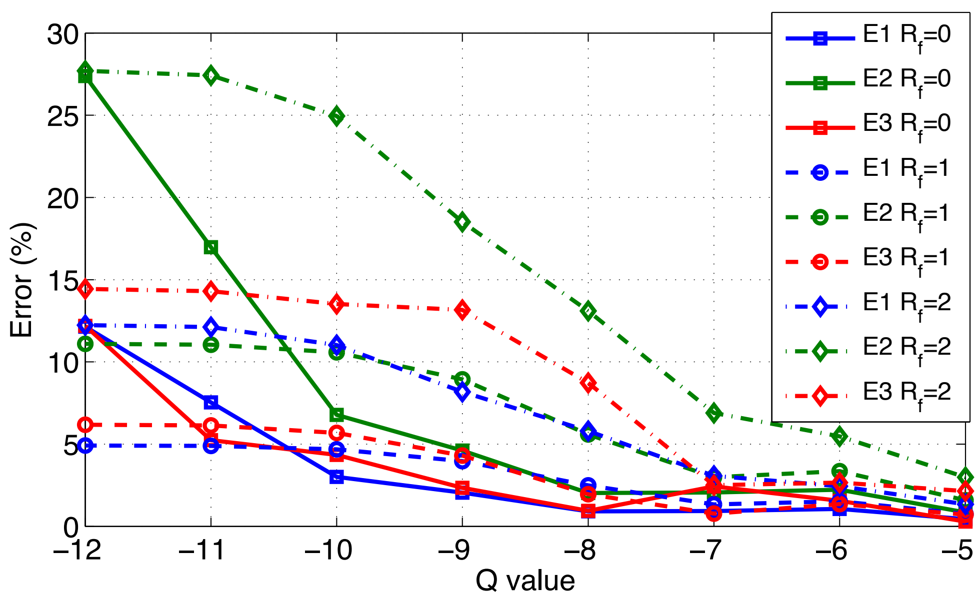
Case No.	Signal	R_f range	Q_f range
A1 (GBW)	0-1 Hz BLWN	[-1 : 1: 2]	[-12 : 1: -5]
A2 (GBW)	0-3 Hz BLWN	[1: 1 : 4]	[-12 : 1: -5]
A1 (EBW)	0-1 Hz BLWN	[0: 1: 3]	[-12 : 1: -5]
A2 (EBW)	0-3 Hz BLWN	[0: 1 : 3]	[-12 : 1: -5]



(a) Input Signal PSD - Cases A1 and A2

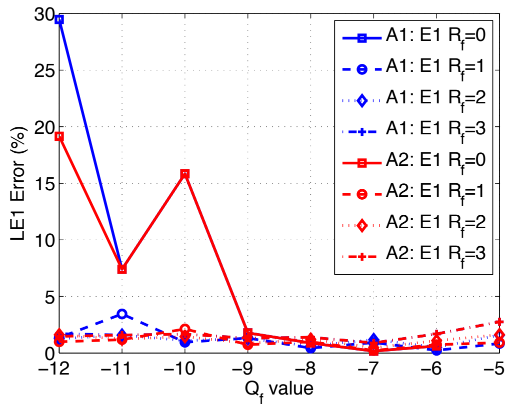


(b) Effect of Q and R on E3 - Case A1 and A2

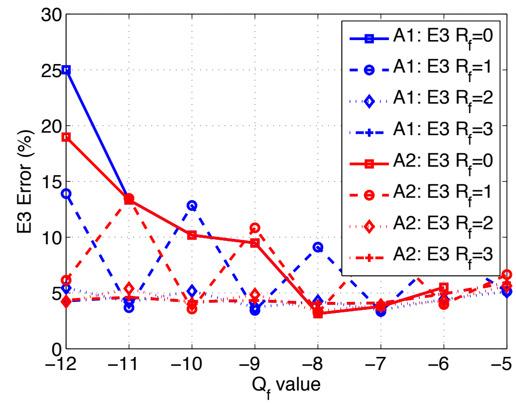


(c) Effect of Q and R on all error indicators - Case A1

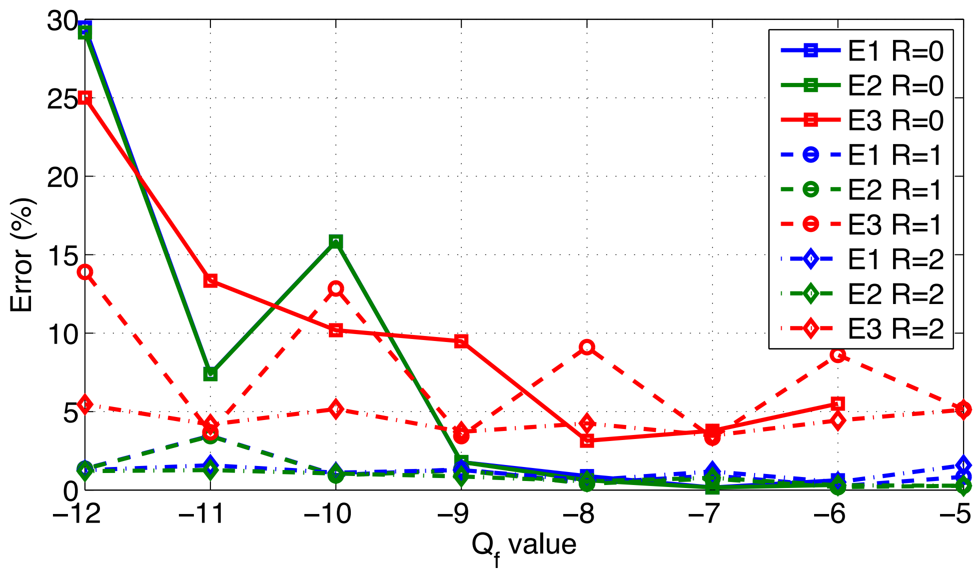
Figure 9.17.: Q and R effect on estimation accuracy, General BW model



(a) Q and R effect on E1 - Case A1 and A2



(b) Q and R effect on E3 - Case A1 and A2



(c) Q and R effect on all Error indicators - Case A1

Figure 9.18.: Q and R effect on estimation accuracy, Extended BW model

Effect of Initial Guess θ_0 Parameter on Model Updating Performance

In examining the performance of the model updating procedure, another question to address is how the estimation stability and accuracy may be affected by the initial guess (initial condition). To perform this analysis we use an initial guess factor F_g which θ_0 is defined as $\theta_0 = F_g \cdot \theta_e$, where θ_e is the nominal values of the model parameter. F_g is chosen from 0.2 to 2.2, where $F_g = 0.2$ indicates the initial guess in each parameter is 1/5 of its nominal value and $F_g = 2.2$ indicates 120% over estimation in each model parameter.

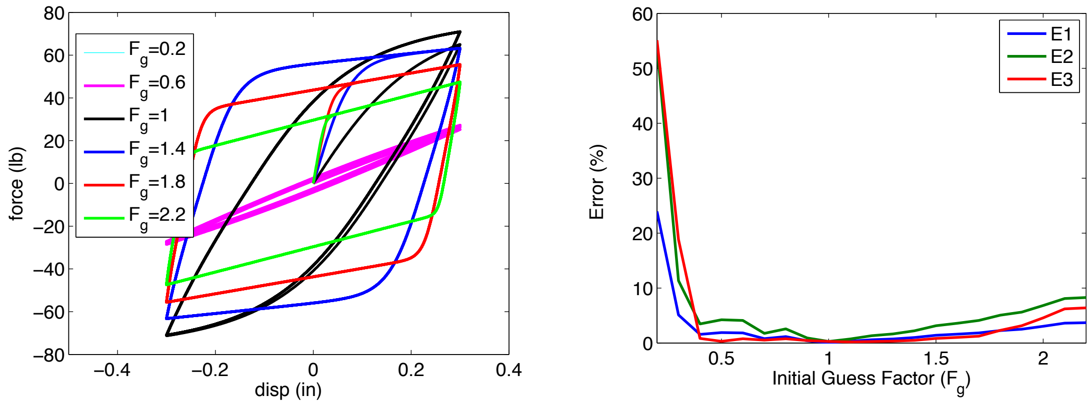
Five representative models with F_g=[0.2, 0.6, 1.4, 1.8, 2.2] and the exact model (F_g=1.0) are excited using a sinusoidal input, exhibiting a variety of hysteretic behaviors as demonstrated in Fig. 9.19(a). The output errors between these erroneous models and the exact models are listed in Table 9.7. It is clear that the chosen BW initial conditions produce a broad range of hysteretic behaviors and are significant different from the exact model performance.

However, although the initial guesses may be very far away (100% over estimation) from the exact solution, the CUKF model updating accuracy is not affected greatly as shown in Fig. 9.19(b). Another variable specific to CUKF algorithm λ_C, which defines the interval constrained unscented transformation (ICUT) as in Eq. 9.10 and 9.11. From simulation, λ_C is found to have little influence to model updating performance, thus, is not further discussed.

Another variable specific to CUKF algorithm λ_C, which defines the interval constrained unscented transformation (ICUT) as in Eq. 9.10 and 9.11. Known $\lambda_C + L > 0$, for the GBW model, $L = 6$, $\lambda_C > -6$, we use λ_C values ranging from -5 to 1 and perform the model updating procedure. The results are shown in Fig. 9.21 for the GBW , which demonstrate that λ_C has little influence on the estimation accuracy in each index.

Table 9.7.: Representative Model Cases

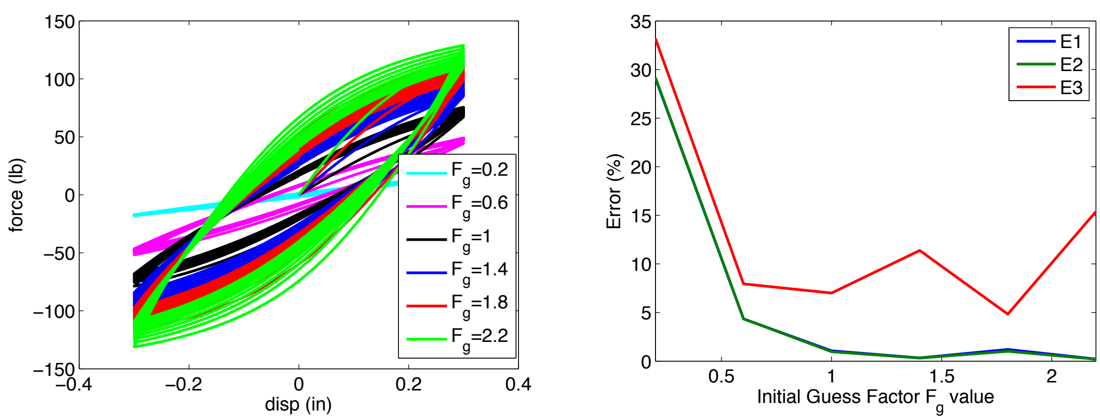
Case	Error in Response (GBW)	Error in Response (EBW)
F_g=0.2	93.7%	77%
F_g=0.6	52.98%	34.04%
F_g=1	0	0
F_g=1.4	42.76%	28.16%
F_g=1.8	63.94%	51.66%
F_g=2.2	78.42%	72.01%



(a) Hysteresis Loop with Different Initial Guess Parameter

(b) Estimation Error Indices for Various Initial Guesses

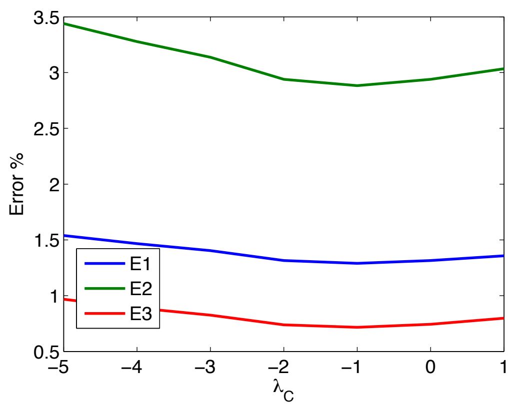
Figure 9.19.: Effect of Initial Guess on Estimation Accuracy, General BW model



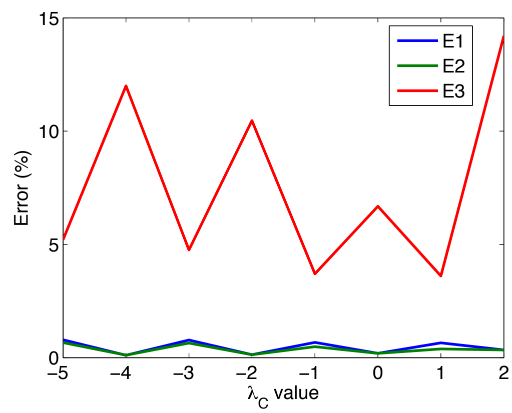
(a) Hysteresis loop with different initial guess parameter

(b) Estimation error indices for different initial guess

Figure 9.20.: Initial Guess Effect on Estimation Accuracy, Extended BW model



(a) GBW case



(b) EBW case

Figure 9.21.: λ_C Effect on Estimation Accuracy

9.6 Conclusion

HSMU performance is analyzed using a numerical example considering a practical case study with three identical energy dissipation plates in a structural system. In this simulation, one of these plates is identified as the experimental substructure in hybrid simulation, although it is simulated here, and the other two instances are in the numerical model. However, with unknown properties of the plates a large error in the global response would result. In simulation, HSMU reduces these errors by utilizing identified, real-time model parameters in the target model. The study leads to the following conclusions for implementation of such a model updating procedure.

1) HSMU results in improved accuracy in hybrid simulation when the initial target model behavior differs significantly from the true response. If the target model is relatively accurate, it may not always be necessary to implement model updating. Additionally, the initial guess of the target model parameter for CUKF does not significantly affect the stability and convergence of the updating procedure, which indicates that HSMU is likely to be successful even with a large initial error in the target model.

2) The accuracy of the model identification procedure depends on the richness of the information (frequency content) in the structural response as this is the input to the physical specimen. HSMU is not applicable in all situations because reaching the local minimum requires that the system is fully excited. Also, the excitation incompleteness may cause parameter oscillation in identification and not truly converge.

3) The target model complexity should be selected based on the conceptual specimen physical properties. All of the parameters in the model should participate sufficiently in the resulting model behavior. Otherwise, the system identification algorithm is hindered in its ability to estimate the true parameter set, resulting in residual errors.

Other conclusions from this work relate to the CUKF in general. The selection of process noise matrix Q and measurement noise matrix R combination has a large influence on estimation accuracy. Larger R indicates a larger tolerance on error between estimation and measurement, thus the procedure is more robust. Larger Q indicates a slower convergence on estimated parameter, and thus a more accurate solution in general. The combination of Q and R should be carefully studied before implementing CUKF.

10. LOCAL EVALUATION OF HYBRID SIMULATION WITH MODEL UPDATING

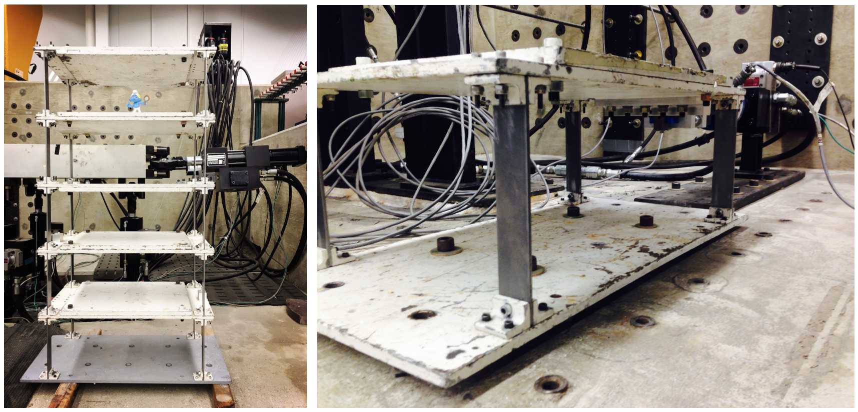
As discussed in chapter 8, the commonly used nonlinear models can be categorized into two groups, one is the phenomenological model, where macro mechanical behavior (displacement-force hysteresis and energy dissipation) is captured and described through differential equations. The other is the constitutive model which is associating with the constitutive relationship of structural materials. In the previous studies, the phenomenological Bouc-Wen model and Bouc-Wen-Baber-Noori model have been adopted widely among recent online model updating applications. Wu et al. applied the unscented Kalman Filter (UKF) to identify different Bouc-Wen model with degrading and pinching behavior, and validated with numerical examples [57]. Song et al. further extended this study to experimental study [122]. The Bouc-Wen model was first introduced in HSMU by Kwon et al. [56]. In the test, the numerical substructure had several identical components. The model of each component was a collection of Bouc-Wen models with different predetermined parameter sets. During model updating, a weighting factor was identified for each Bouc-Wen model until the summation of their weighted responses matched the measured response from the physical specimen. Thus, the accuracy of this approach depends on the selected initial collection of Bouc-Wen models. Later, identification directly on model parameters were integrated into hybrid simulation, where Hashemi et al. and Saho et al. implemented the unscented Kalman Filter algorithm in identifying the Bouc-Wen parameters of structural frame, and Wu et al. applied the constrained unscented Kalman Filter algorithm in identifying the parameters of a buckling restrained brace [58], [59], [60].

As an alternative to phenomenological models, constitutive relationship provides a deeper observation on structural components modeling, such as component damage level, status of structure serviceability and reliability. Furthermore, the tested components and its counterpart in numerical simulation do not need to be identical in geometry. Hazem and Elnashai first proposed a hybrid simulation framework with finite element software ZeusNL [61]. The finite element software is linked to a model updating algorithm such as genetic algorithms or neural network to identify the parameter of constitutive bilinear steel model and nonlinear concrete model [62]. This framework has been validated with numerical examples and offline experimental data, [63].

In this chapter, the HSMU approach is implemented and evaluated through an experimental study. The reference structure in the study is a five story structure with identical columns on each floor. In hybrid simulation, the first story is tested physically while the remaining stories are included in the numerical substructure. Two series of hybrid simulation tests are conducted. First, in case I, the entire frame is modeled with five decoupled Bouc-Wen models, parameters in each Bouc-Wen model are assumed to be identical. The parameters of the Bouc-Wen model are identified using measured response (displacement and force) from the physical specimen. Similarly, in the other case (namely the case II), the numerical stories are constructed in OpenSees with a bilinear steel model. A single story frame model is also established with OpenSees, where the steel material properties are identified using the physical specimen response. In this chapter, the model updating performance in HSMU is discussed. The estimation accuracy for both cases is presented, and the updated model responses are compared to the initial model. Parameters convergence due to incomplete excitation is continue discussed for both cases.

10.1 Model Updating Implementation for HSMU

In this experimental study, the entire structure is a five story steel frame, shown in Fig. 10.1(a), all five stories are identical. For each floor, a lumped mass of 23.2 kg is concentrated on the top plate, four columns (ASTM A36 steel, cold rolled) are fixed to the top and bottom plate, with effective length of 170.2 mm, the initial stiffness per floor is 1.82×10^5 N/m. The drawing of a typical story is shown in Fig. 10.2. In hybrid simulation, only the first story frame is tested physically (Fig. 10.1(b)), and the rest of the stories are modeled in the numerical substructure.



(a) Entire structure (b) Experimental Substructure

Figure 10.1.: Entire Structure and the Physical First Story in HSMU

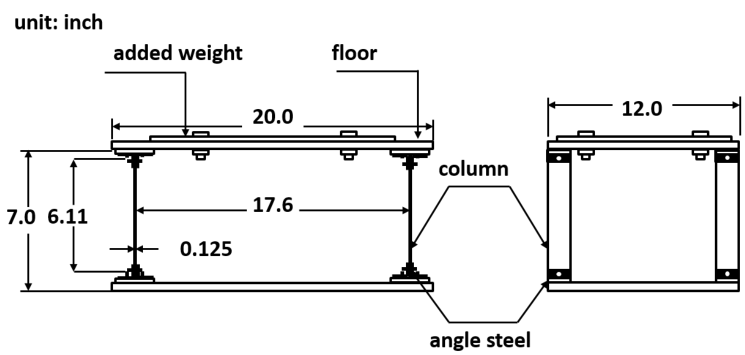


Figure 10.2.: Drawing of a Typical Story

As stated earlier, recall the formulation of HSMU is defined as:

$$M^N \ddot{x}^N + C^N \dot{x}^N + K^N x^N + F^E(x^E, \dot{x}^E) + R^N(x^N, \dot{x}^N, \tilde{\theta}_R) = -M\Gamma \ddot{x}_g \tag{10.1}$$

$$M^E\ddot{x}^E + C^E\dot{x}^E + K^E x^E + R^E(x^E, \dot{x}^E) = F^E(x^E, \dot{x}^E) \tag{10.2}$$

$$\tilde{\theta}_R = \Psi(R^E, x^E, \dot{x}^E, \theta_\Psi) \tag{10.3}$$

where the superscripts $(\ )^N$ and $(\ )^E$ denote the portions of the reference structure included in the numerical and experimental substructures, respectively, $M = M^E + M^N$, $C = C^E + C^N$, $K = K^E + K^N$. F^E denotes the measured force in the experimental substructure. Ψ indicates the model updating is performed in real-time, θ_Ψ is the parameter being updated through the chosen model updating algorithm, $\tilde{\theta}_R$ is the recursively identified nonlinear model parameters that minimize the associated cost function. The numerical restoring force $R^N(x^N, \dot{x}^N, \tilde{\theta}_R)$ is adapting in real-time based on the physical responses. In this section, two types of numerical substructures are developed for the model updating study, one is with a phenomenological Bouc-Wen model and the other is with a constitutive steel material model.

In this chapter, the selected model updating algorithm is also the Constrained Unscented Kalman Filter (CUKF), whose formulation is presented in the chapter 9. The discrete implementation of CUKF in HSMU on the k^{th} step is shown in Fig. 10.3.

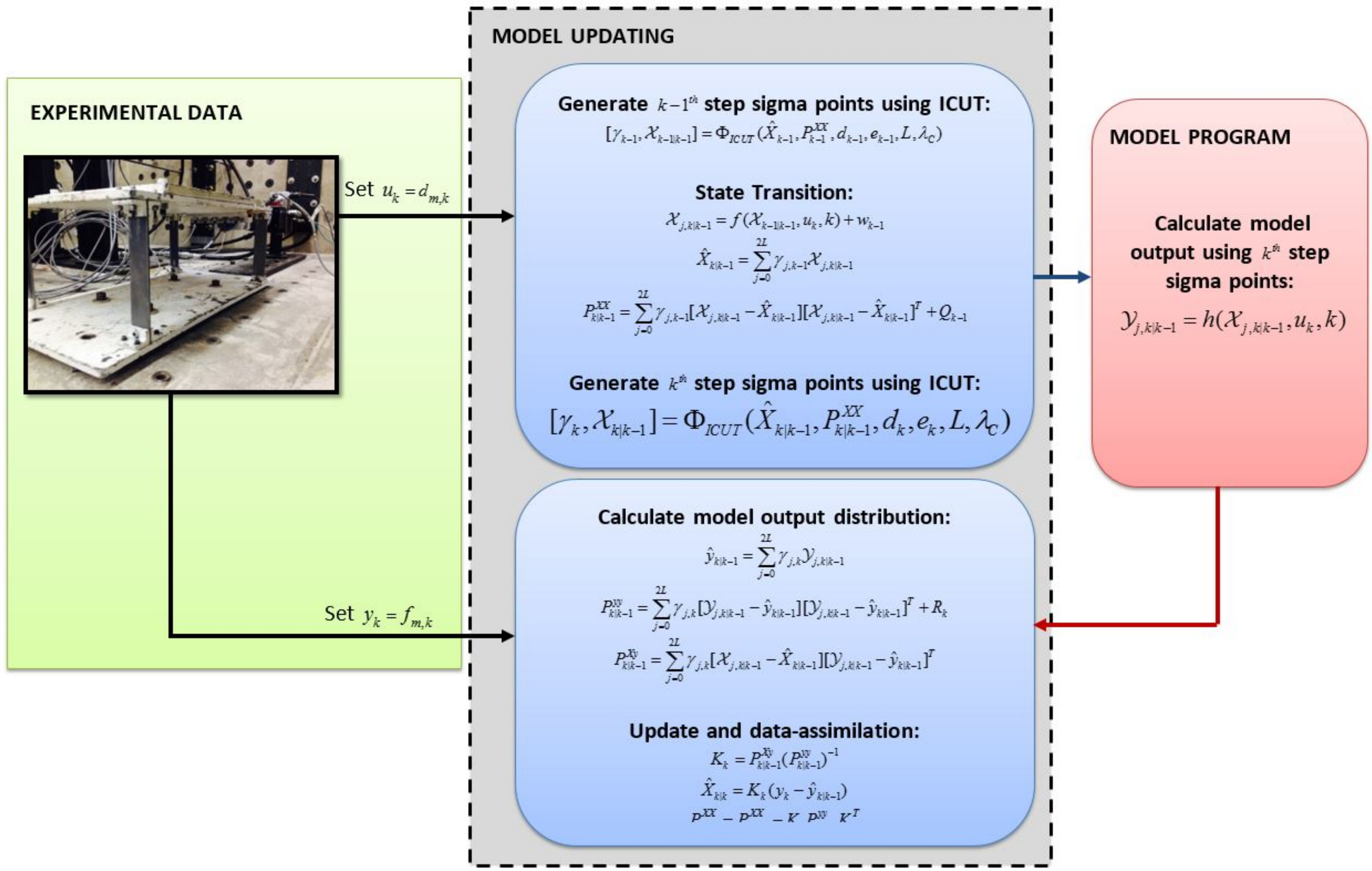


Figure 10.3.: CUKF Implementation at k^{th} Step

10.1.1 Formulation and Implementation of HSMU Case I

In HSMU case I, the physical substructure is first modeled with the phenomenological Bouc-Wen-Baber-Noori model [105], [100], as indicated before, here is also denoted as EBW model. This model can capture the pinching and degradation effects in a structure component, represented by Eq. 10.4 - 10.12 also is discussed in chapter 8.

$$R_{EBW}(x^E, z) = \alpha_{EBW} k_{EBW} x^E + (1-\alpha) k_{EBW} z \tag{10.4}$$

$$\dot{z} = h(z)\{\frac{\dot{x}^E - \nu(\varepsilon)(\beta_{EBW}|\dot{x}^E||z|^{n_{EBW}-1}z + \gamma_{EBW}\dot{x}^E|z|^{n_{EBW}}}{\eta(\varepsilon)}\} \tag{10.5}$$

where k_{EBW} is the stiffness coefficient and $0 \leq \alpha_{EBW} \leq 1$ determines the level of nonlinearity, $\alpha_{GBW} = 1$ indicates the system is purely linear and $\alpha_{GBW} = 1$ indicates the system is purely hysteretic. In the energy dissipation $E(t)$, response duration and severity is measured by $\varepsilon(t)$.

$$E(t) = \int (1-\alpha_{EBW}) k_{EBW} z \dot{x}^E dt, \varepsilon(t) = \int z \dot{x}^E dt \tag{10.6}$$

$$\nu(\varepsilon) = 1 + \delta_{\nu B2}\varepsilon \tag{10.7}$$

$$\eta(\varepsilon) = 1 + \delta_{\eta B2}\varepsilon \tag{10.8}$$

where $\nu(\varepsilon)$ and $\eta(\varepsilon)$ are degradation shape function, and $\delta_{\nu EBW}$, $\delta_{\eta EBW}$ are degradation parameters. To describe the pinching function, $h(z)$ is given by:

$$h(z) = 1 - \zeta_{1B2} e^{-[z \cdot sgn(\dot{x}^E) - q_{EBW} z_{x^E}]^2/\zeta_{2B2}^2} \tag{10.9}$$

$$\zeta_1(\varepsilon) = \zeta_{sB2}(1 - e^{-p_{EBW}\varepsilon}) \tag{10.10}$$

$$\zeta_2(\varepsilon) = (\Psi_{EBW} + \delta_{\Psi B2}\varepsilon)(\lambda_{EBW} + \zeta_1) \tag{10.11}$$

$$z_{x^E} = [\frac{1}{\nu(\varepsilon)(\beta_{EBW} + \gamma_{EBW})}]^{\frac{1}{n_{EBW}}} \tag{10.12}$$

The parameters λEBW, ζ_{sEBW}, p_{EBW}, q_{EBW}, Ψ_{EBW}, and $\delta_{\Psi EBW}$ are involved in describing the pinching effect. p_{EBW} quantifies the initial drop of the slope, ζ_{sEBW} relates to the total slip, Ψ_{EBW} is a parameter that contributes to the amount of pinching. $\delta_{\Psi EBW}$ specifies for the desired rate of pinching. Detail discussion of the hysteretic shape change to each parameter can also be found in chapter 8.

For EBW model updating, the model updating algorithm and numerical substructure are both realized in a Matlab program. Thus, the hysteresis model is modified into nonlinear dynamic form as in Eq. 10.13 and 10.14. where y_k is the model estimated hysteresis force at the k^{th} step, $\theta(k) = [\alpha,\, k,\, \beta,\, n,\, \delta_\eta,\, \delta_\nu,\, q,\, \gamma,\, \zeta_s,\, p,\, \Psi,\, \delta_\Psi,\, \lambda,\, \varepsilon\ z]^T$, $u(k) = [x^E(k)\ \dot{x}^E(k)]$.

$\dot{\theta}_(k) = [\mathbf{0}_{1\times 14},\, \theta_{15}u_2,\, h(\theta_{15})\{\frac{u_2-(1+\theta_6\theta_{14})(\beta|u_2||\theta_{15}|^{\theta_4-1}\theta_{15}+\theta_8 u_2|\theta_{15}|^{\theta_4}}{1+\theta_6\theta_{14}}\}]^T$.

$$\theta_k = \theta_{k-1} + \dot{\theta}_k \cdot dt + w_k \tag{10.13}$$

$$y_k = \theta_1\theta_2 u_1 + (1-\theta_1)\theta_2\theta_{15} + v_k \tag{10.14}$$

Recall, the equation of motion of a multistory frame is:

$$M\ddot{x} + C\dot{x} + R(x, \theta_{R1}) = -M\Gamma\ddot{x}_g \tag{10.15}$$

Here, in a MDOF, the restoring force at each level can be computed using several independent phenomenological EBW models $R(x_i, \theta)$, where x_i is the relative displacement for the EBW model, θ is the EBW parameter set identified at each step.

$$\mathbf{R}(x,\theta) = \begin{bmatrix} R(x_1,\theta) - R(x_2 - x_1, \tilde{\theta}) \\ R(x_2,\theta) - R(x_3 - x_2, \tilde{\theta}) \\ \ldots \\ R(x_5,\theta) \end{bmatrix}$$

In HSMU case I, with the first story as the experimental substructure:

$$M\ddot{x} + C\dot{x} + R^N(x, \hat{\theta}_{R1}) = -M\Gamma\ddot{x}_g - R^E \tag{10.16}$$

at the k^{th} step, the physical restoring force $R^E(k)$ and displacement $x_1(k)$ can be measured, based CUKF, an estimated EBW model parameter set is available $\hat{\theta}(k)$, with the estimated EBW model force is $r(x_1(k), \theta(k))$. Therefore, numerical restoring force R^N at k^{th} step is

$$R^N(x_i(k), \theta(k)) = \begin{bmatrix} -R(x_2(k) - x_1(k), \theta(k)) \\ R(x_2(k), \theta(k)) - R(x_3(k) - x_2(k), \theta(k)) \\ \dots \\ R(x_5(k), \theta(k)) \end{bmatrix}$$

As discussed in chapter 2, the information exchange in HSMU requires a coordinator and communication between physical components, numerical components, and model updating components. The coordinator program used here is the HyTest platform [69]. Fig. 10.4 shows the communication and information exchange in the HSMU case I. Both the model updating algorithm and the numerical substructure model are implemented in the Matlab code and the external loading to the experimental substructure is implemented by through LabVIEW program. In the Matlab code, the estimated parameter of the EBW is first identified through CUKF with a numerical model of the experimental substructure and the measured response R^E, later, the numerical substructure restoring force is calculated using the parameter $\tilde{\theta}_R$, by solving the equation of the motion, the structural response is calculated. Displacement at the numerical-experimental boundary is sent to the physical specimen through LabVIEW. This implementation step is demonstrated in Fig. 10.5.

10.1.2 Formulation and Implementation of HSMU Case II

Unlike HSMU case I, HSMU case II updates a constitutive relationship of a material model. Those constitutive model can be implemented in different commercial or open source software, or can be implemented by a user defined finite element code. In this study, the Open System for Earthquake Engineering Simulation (OpenSees) is selected as the software framework for modeling the numerical substructure, as

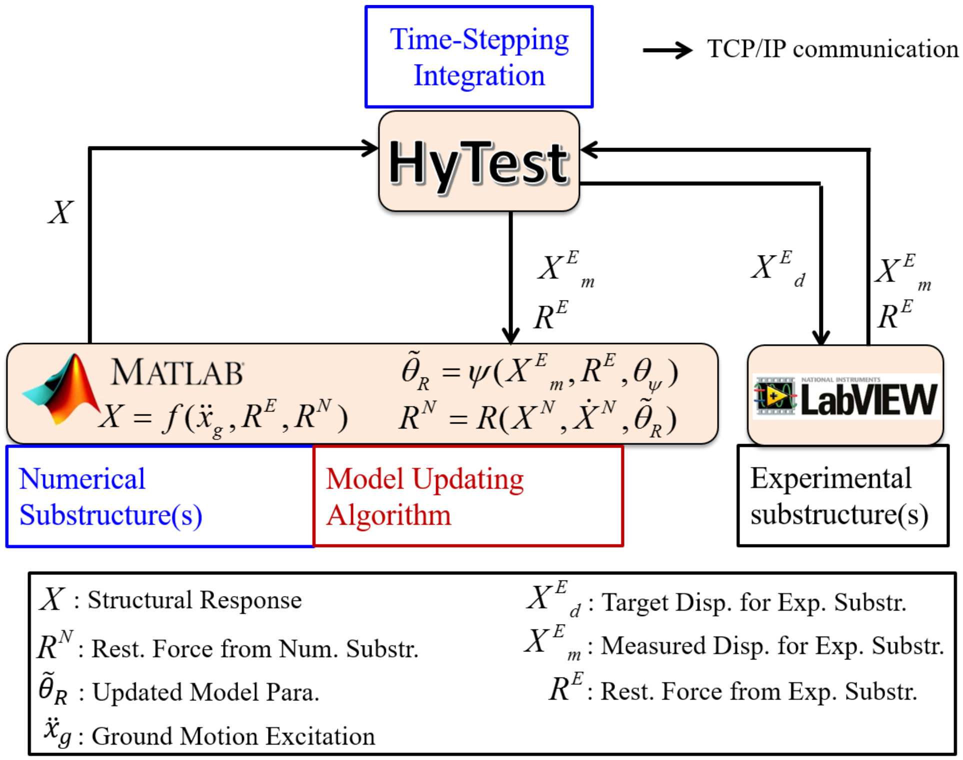


Figure 10.4.: Information exchange and communication in HSMU case I

well as the model of the physical substructure. Therefore, the numerical model of the physical substructure modeled by OpenSees performs as function $OpenSees(\theta, u)$, where θ here also is the parameter to be identified in CUKF, and u is the input to the OpenSees model, which is the measured displacement of the physical specimen.

As discussed in chapter 8, only considers the isotropic hardening, the simplified bilinear model can be described as:

$$E^p = b_s \cdot E \tag{10.17}$$

The parameters describe the hysteretic behavior of a steel component are: initial young's modulus E, stiffness hardening factor b_s, and yield stress σ_y. In case II, during HSMU, write the updating algorithm in the form of Eq. 10.13 and 10.14 as:

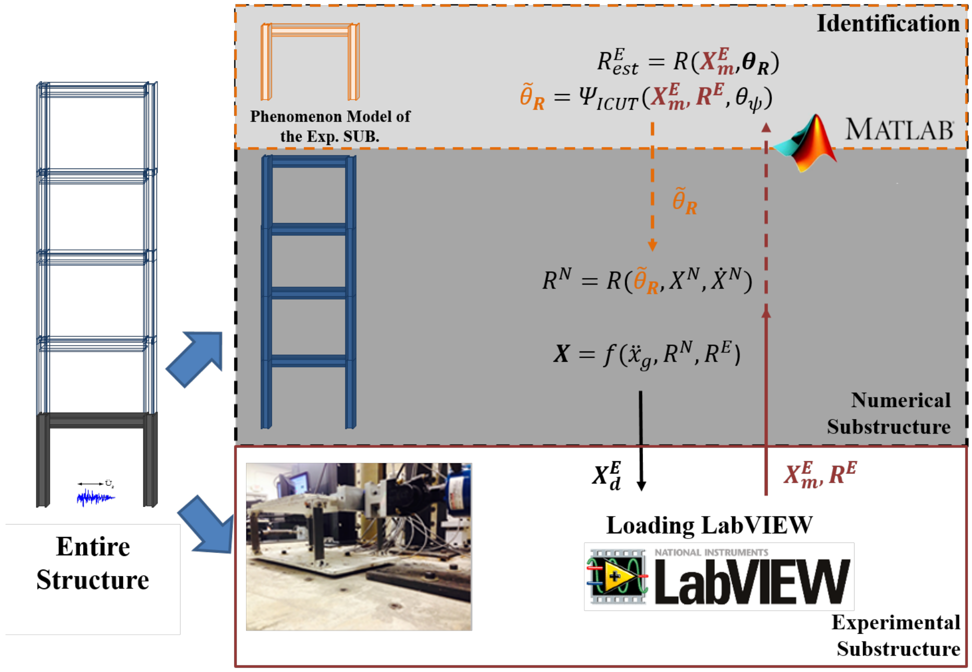


Figure 10.5.: Schematic Implementation of HSMU in case I

$$\theta_k = \theta_{k-1} + \dot{\theta}_k \cdot dt + w_k \tag{10.18}$$

$$y_k = OpenSees(\theta_k, u_k) + v_k \tag{10.19}$$

Similarly as in case I, the coordinator program in case II is the HyTest platform. In case II, the numerical substructure, model updating components, and the numerical model of the physical specimen are no longer modeled in the same software. Fig. 10.6 shows the communication and information exchange in the HSMU case II. Besides the conventional information exchange between a finite element code which governing the numerical restoring force computation and the physical components, information between model updating component it self needed to be considered. Because the model updating components contain 1) a model updating algorithm modeled in Matlab to implement the CUKF optimization, 2) an OpenSees model as a function which output is to be optimized to match the measured physical restoring force R^E with an

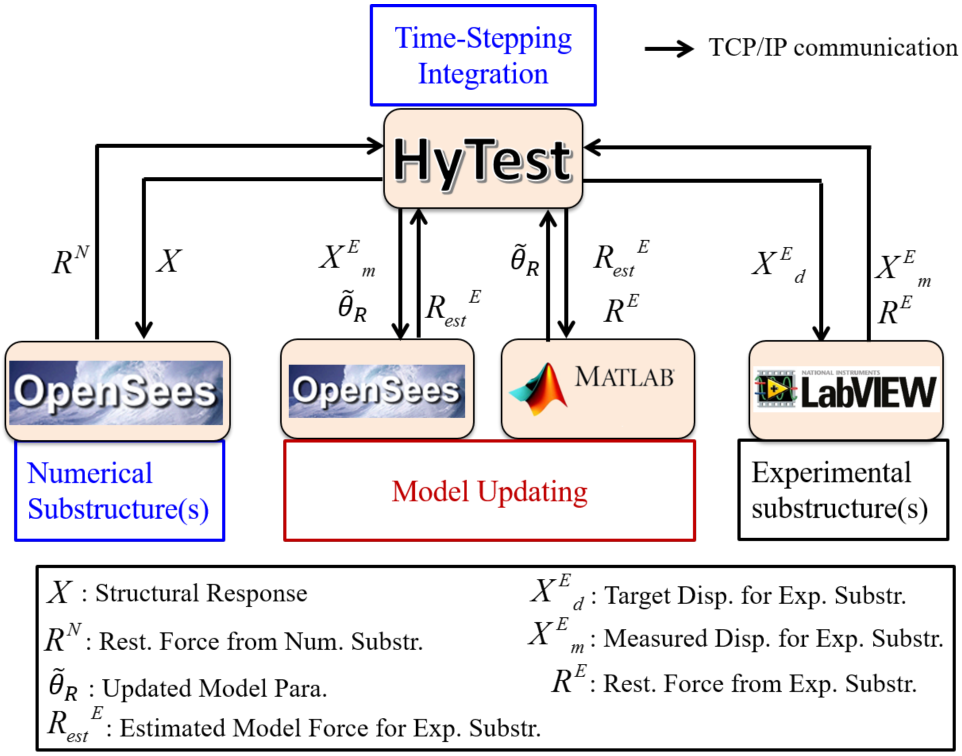


Figure 10.6.: Information exchange and communication in HSMU case II

optimized parameter set θ, information exchange between the two different software is implemented through TCP/IP communication protocol. Because $2 \times L + 1$ sigma points are required for the one CUKF optimization, there are $2 \times L + 1$ sets of $\tilde{\theta_R}$ sent to the OpenSees model to calculate the corresponding R^E_{est} for each iteration time step. This implementation step is demonstrated in Fig. 10.7.

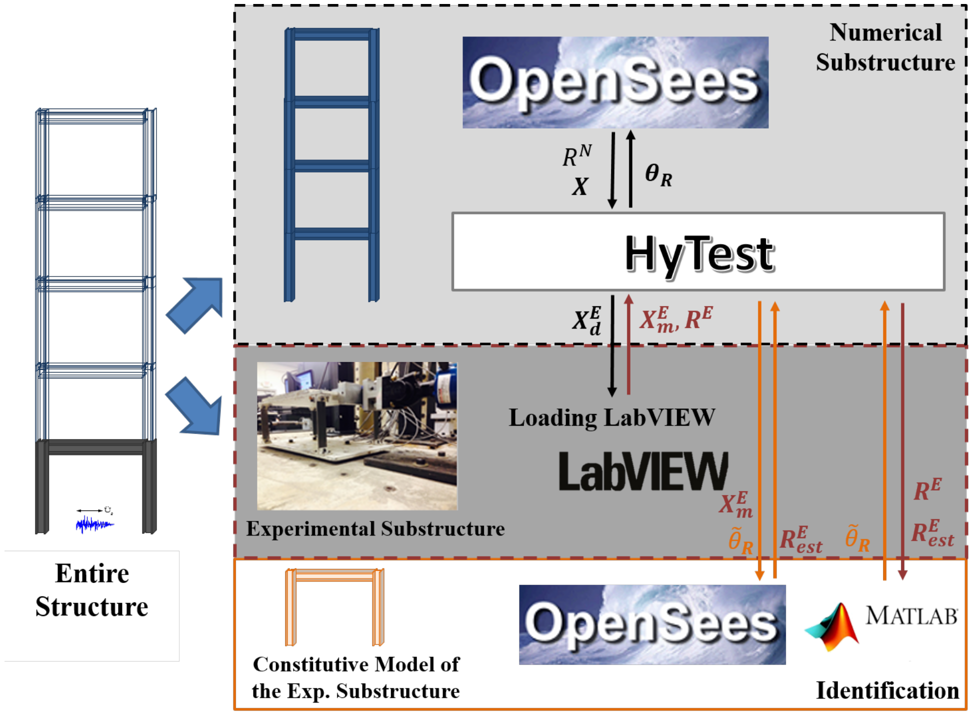
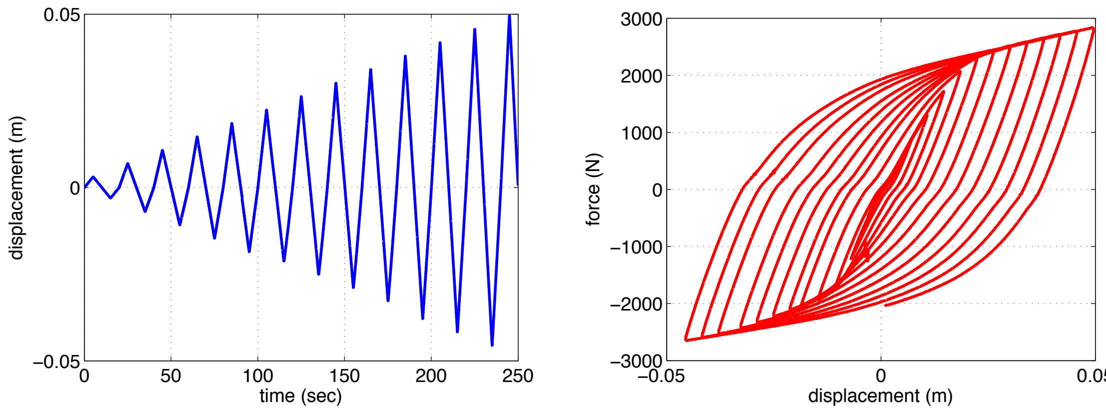


Figure 10.7.: Schematic Implementation of HSMU in case II

10.2 Initial Model Parameter Estimation

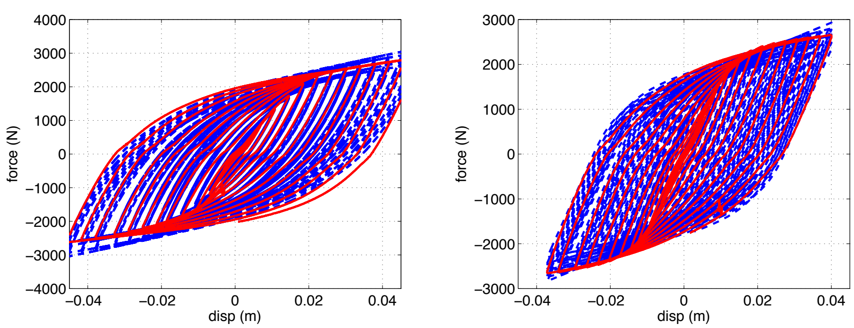
For implementing model updating in hybrid simulation, the CUKF requires an initial parameter set for the associated model. As indicated in chapter 8, one major drawback of the phenomenological model is that the parameters do not have physical meaning, therefore, it is very difficult to estimate a reasonable initial parameter set. Also, knowledge on previous component models and test results cannot be transmitted to a new specimen when geometry changes. Therefore, a quasi-static cyclic test is conducted to identify the initial parameters of the phenomenological model. The experimental setup of this cyclic testing is shown in Fig. 10.1(b).

The loading protocol and structural responses of the cyclic tests are demonstrated in Fig. 10.8. Several parameter sets satisfied the optimization criterion are generated from offline identification. Those parameters are spread across the range as listed in Table 10.1. Results indicate the parameter set describe the physical specimen hysteretic behaviors is not unique. In order to perform the UCKF, one initial parameter set is chosen, also the upper and lower bound for each parameter are determined.



(a) Cyclic Input Trajectory (b) Frame Hysteresis Behavior

Figure 10.8.: Cyclic Test and Structural Response



(a) Phenomenological Model Identification (b) Constitutive Model Identification

Figure 10.9.: Offline Identification Results for Cyclic Testing

For HSMU case II, similar process is conducted to estimate the initial parameter set for the bilinear model. From the offline identification, also more than one parameter sets are found to satisfy the optimization criterion. However, the results are less broadly distributed compared to case I, as listed in Table 10.2. Also, the initial parameter of the bilinear model can be estimated from the material test, as in Fig.

Table 10.1.: Model Parameters of EBW

Parameter	Offline ID Range	CUKF Range	HSMU Initial Parameter
δ_v	[1.02 2.56]	[0 20]	2.31
β	[55 168]	[0 200]	92.185
γ	[0.155 0.39]	[0 10]	0.94
δ_η	[1.17 4.3]	[0 20]	3.1142
α	[0.07 0.17]	[0 1]	0.156
n	[1 1.8]	[1 3]	1.1833
K	[1.4e+5 1.8e+5]	[1.0e+5 2.0e+5]	1.55e+5
δ_Ψ	[0.03 0.07]	[0 0.1]	0.05
ς_s	[0.56 1.41]	[0 5]	0.92
Ψ	[0.6 1.5]	[0 2]	0.94
p	[0.015 0.0375]	[0 0.05]	0.025
q	[0.022 0.07]	[0 0.1]	0.045
λ	[0.13 0.8]	[0 1]	0.476

10.10. Due to its physical meaning, the upper and lower bound for each parameter are less arbitrary.

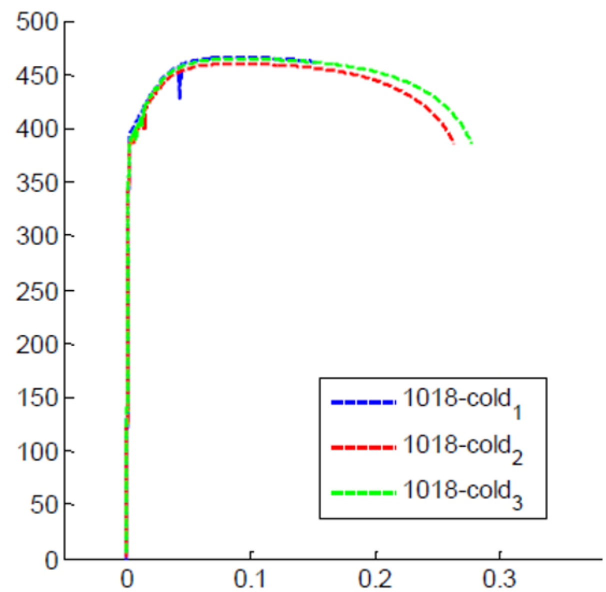


Figure 10.10.: Steel Material Test Result

Table 10.2.: Bilinear Model Parameters

Parameter	Offline ID Range	CUKF Range	HSMU Initial Parameter
F_y	[3.0e+8 5.2e+8]	[1.0e+8 8.0e+8]	4.8e+8
E	[2.0e+11 2.0e+11]	[1.5e+11 3.0e+11]	2.0e+11
bs	[0.04 0.15]	[0.01 0.8]	0.045

10.3 Model Updating Results in HSMU

In the experimental study, the physical substructure is attached to Shore Western (SW) actuator as in Fig. 10.11. The actuator is driven by voltage command from software SW6000 with an embedded PID controller. National Instruments (NI) and LabVIEW bridges structural dynamic response computation and the data acquisition hardware. In the loading step, the LabVIEW program receives the displacement command from the numerical solver, and then converts it into and analog signal to send to the SW6000.

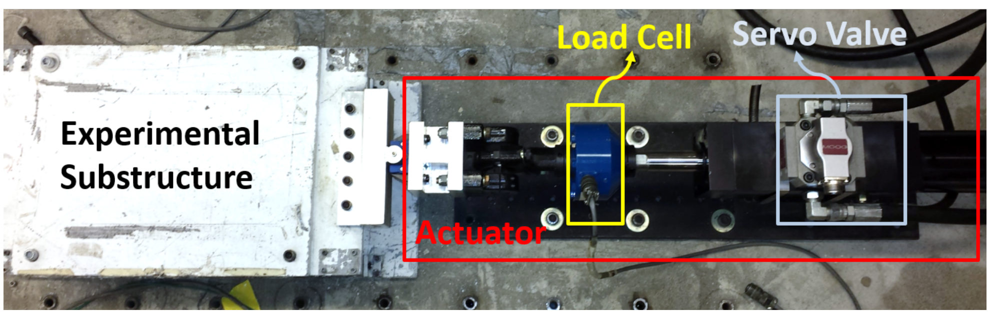


Figure 10.11.: Experimental Substructure with Actuator

The dynamic excitation to HSMU is a sequential combination of two scaled El-Centro earthquake (both in amplitude and time) records with different amplitude. This excitation is first applied to a shake table (will be discussed in chapter 11), the shake table implemented ground motion is shown in Fig. 10.12, where the measured shake table acceleration is used as the structure excitation in hybrid simulation. The entire excitation lasts 40 seconds, in the first 20 second, acceleration peak is 6.98 m/s^2, in the later 20 sec, the acceleration reaches its peak at 18.3 m/s^2. The objective of using two sequential earthquake excitation as one input is to investigate 1) the HSMU model updating performance on model estimation accuracy, parameter convergence; and 2) the model updating performance with incomplete excitation. In this experimental study, the model updating algorithm is taking a smaller excitation (physical specimen response is small, nonlinear behavior is less significant) and then a larger excitation (more significant hysteresis) later in the test. During the first 20 sec (Section one, labeled as S1 in the analysis) the smaller excitation can be considered as an incomplete excitation because the magnitude is smaller compared to the response in the later 20 sec (Section two, labeled as S2 in the analysis).

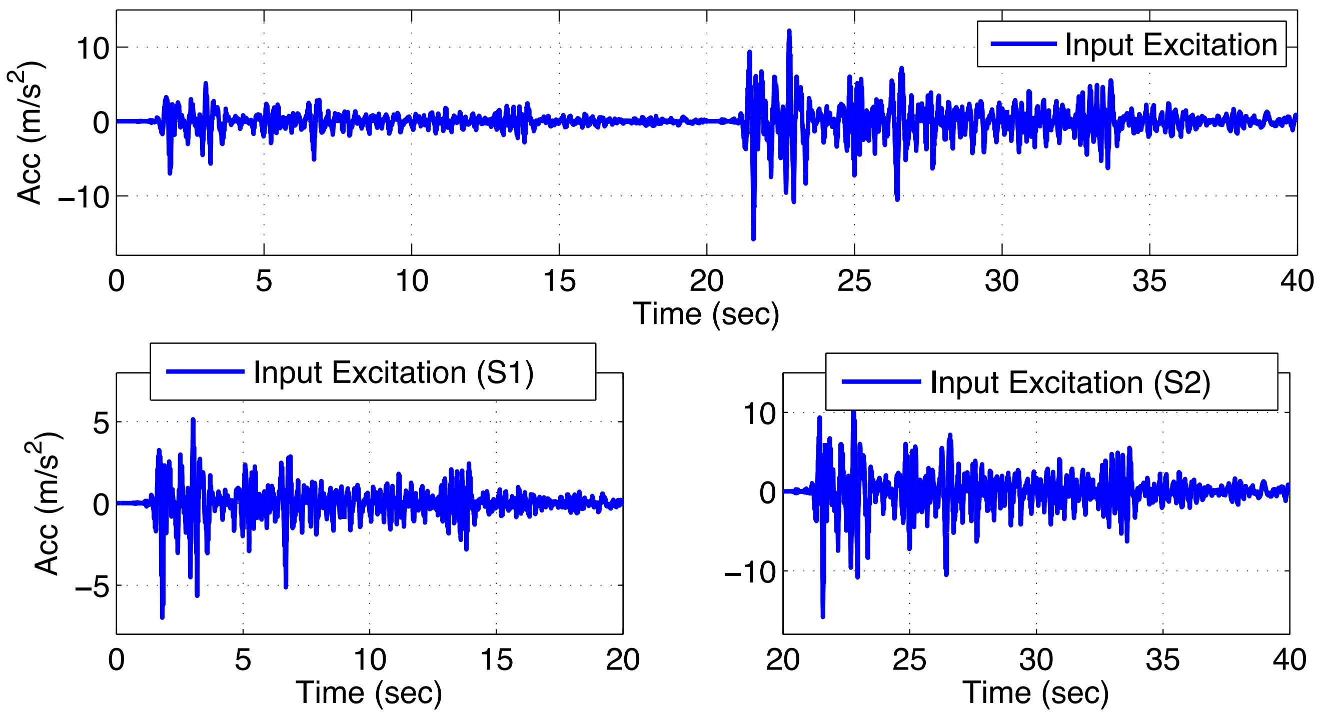


Figure 10.12.: Shake Table Tracking Performance

For both cases, the parameters convergence are presented for the entire time history (40 sec), period S1 (first 20 sec), and period S2 (later 20 sec), where an RMS model updating error indicator is used to assess the model updating effectiveness, defined as:

$$RMS_E = \sqrt{\frac{\sum_{i=1}^{n}(R_{est}(i) - R_m(i))^2}{n}} / \sqrt{\frac{\sum_{i=1}^{n}(R_m(i) - mean(R_m))^2}{n}} \tag{10.20}$$

where (R_{est} is either the estimated force from model updating for HSMU, or the force calculated with a numerical model with initial guess parameter, and R_m is the measured force. This model updating error indicator is applied also for the entire time history, S1 period, and S2 period.

10.3.1 HSMU Case I: Model Updating on the Phenomenological Model

In case I, the state vector contains 13 parameters and 2 states as derived earlier. During model updating, noise and estimation tolerance R is determined at 100 N, Q=diag[10^{-6}, 10^{-4}, 10^{-4}, 10^{-5}, $10^{-4.5}$,10^{-4}, 10^{4}, 10^{-5}, 10^{-6}, 10^{-6}, 10^{-6}, 10^{-6}, 10^{-5}, 10^{-12}, 10^{-12}], and P_0= $10 \times I_{15}$. Determination of R is related to system noise and tolerable error between estimation and measurement, Q matrix is related to the model uncertainty and sensitivity of each parameter and its magnitude. For example, the linear stiffness of the EBW model can be predetermined before testing with relatively higher accuracy, thus, it should be associated with a smaller value in the Q matrix. However, considering the magnitude of the stiffness k (1.5 $\times 10^5$), a value of 10^4 (variance 100 as 1% to its mean value) is reasonable. These model updating related parameters (Q, R, P_0) are still determined on a case by case basis. The selection of R and Q and their effect to model updating performance are discussed in chapter 9. Trade offs exist based on identification program robustness and estimation accuracy based on previously available data.

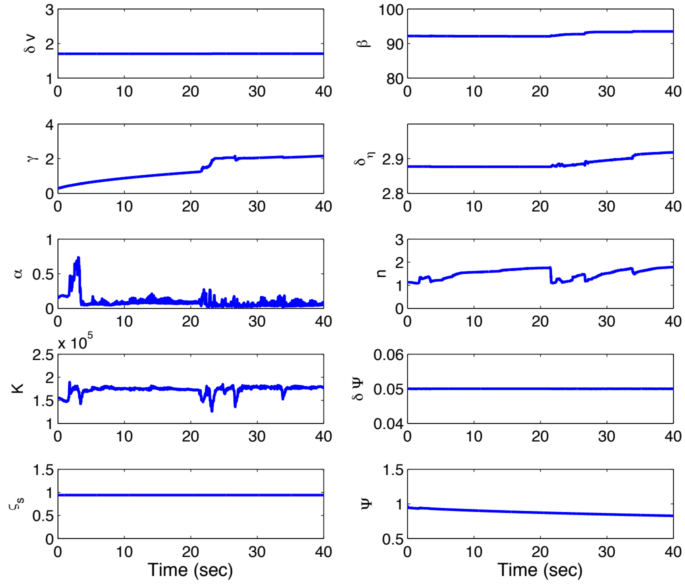
The identification results are illustrated in Fig. 10.13 and 10.14. For the first 20 second (S1), the results confirm that most of the EBW model parameters can converge 3 sec after the earthquake starts, where the first peak response occurs. In the later 20 seconds, a larger peak response occurs due to the increased ground excitation, most of the parameters vary and settle to another optimization. It can be concluded, in S1, the parameters converged to a local optima. When the the peak amplitude of the response evolves in S2, the converged parameters can no longer represent the specimen behavior, therefore the model updating algorithm continues to adjust the model parameters and bring them to a new converged set.

In the entire time history, the Bouc-Wen model can represent the steel frame nonlinearity well. The error between model estimation and measured response is negligible (with RMS error of 3.04%, 3.59%, and 3.34% for S1, S2, and entire time history), compared to the RMS error in the initial model which is 21.39%, 25.32% and 22.78%, respectively. CUKF shows to be effective on updating the phenomenological parameters.

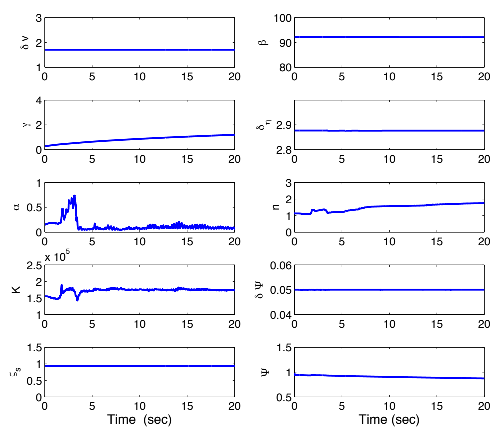
10.3.2 HSMU Case II: Model Updating on the Constitutive Model

For case II, as stated earlier, three parameters are identified in the steel bilinear model. During model updating, noise and estimation tolerance R is determined to be 200 N, Q=diag[10^{-5}, 10^{-5}, 10^{-7}]; P_0= $10 \times I_3$, $L = 3$, and $\lambda_L = -1$. d_k and e_k is the lower bound and upper bound of constrained parameters in Table 10.2, because each parameter has a clear physical meaning, the upper bound and lower bound can be reasonably determined. In the constitutive model updating, only three parameters are needed for identification, thus, due to the persistence excitation requirement, convergence is reached faster after the earthquake starts, as in Fig. 10.15(a).

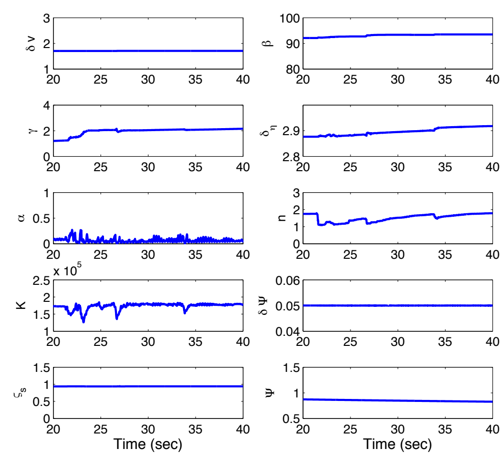
In S1, the estimated yield stress F_y is reduced from the initial parameter $480 N/m^2$ to $380 N/m^2$, Young's modulus E and the hardening factor b_s did not change. In S2, when a larger peak response occurs, this convergence is clearly affected, and eventually



(a) Entire Time History: Parameter Convergence



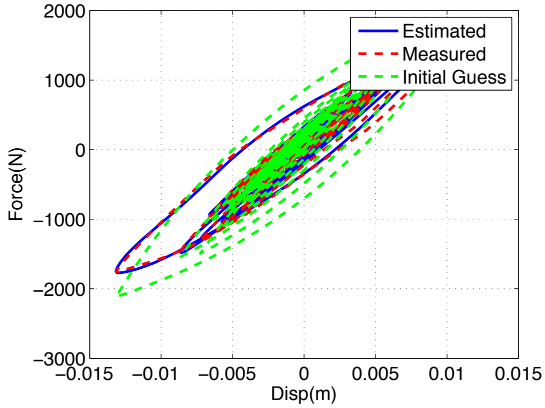
(b) S1: Parameter Convergence



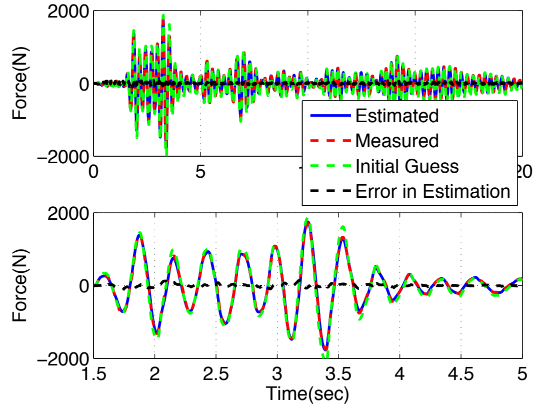
(c) S2: Parameter Convergence

Figure 10.13.: Parameter Convergence for Case I

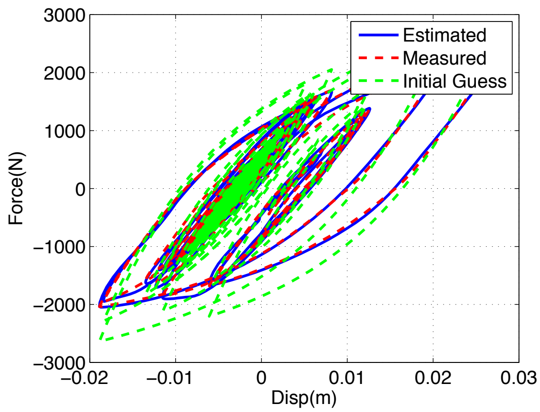
converged to $F_y = 560N/m^2$ and $b_s = 0.04$. Compared to the material test results in Fig. 10.10, steel starts to yield at $380N/m^2$ with initial hardening reduction of 0.2, later around $520N/m^2$, the associated hardening reduction factor is around 0.05. The results from HSMU are matching the material test data, in S1, the specimen



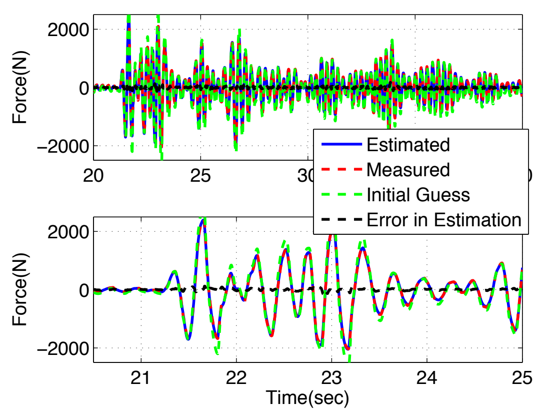
(a) Hysteretic Behavior Comparison



(b) Identification Time Domain Comparison



(c) Hysteretic Behavior Comparison



(d) Identification Time Domain Comparison

Figure 10.14.: Model Updating Performance Using EBW Model

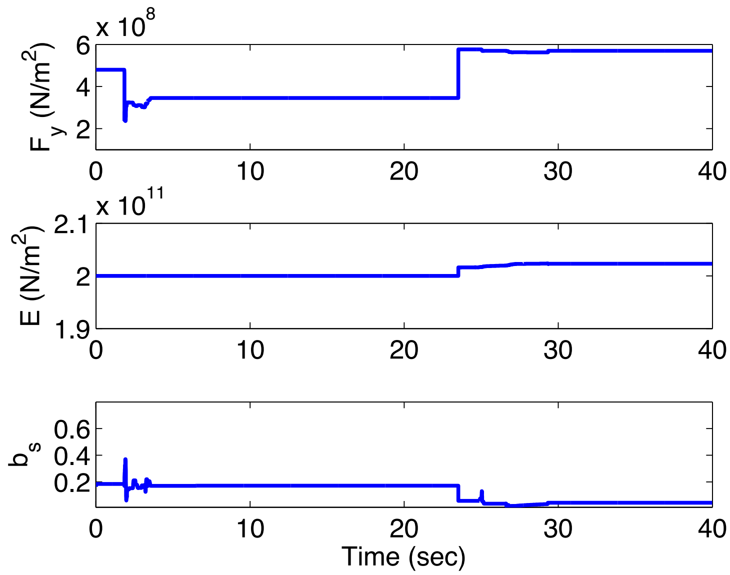
is yielding at $380N/m^2$ with the hardening ratio of 0.2, later in S2, the nonlinear behavior happens in the range around $480 - 550N/m^2$ where the reduction factor is around 0.05. The final estimation in S2 is closer to the initial guess. Therefore, a bilinear curve is not sufficient to describe the steel property, and a trilinear behavior is discovered.

Fig. 10.16 shows the time history comparison between the measured force and the estimated output, also the error between the two. In S1, frame hysteresis behavior is improved in HSMU with RMS error of 8.39%, where the initial model yields an RMS error of 26.41%. This is associated with the first stage bilinear behavior (yield at $380N/m^2$ with the hardening ratio of 0.2) which cannot be captured by the initial model parameters. In S2, the parameter converges at a new optimal that is closer to the initial guess (second stage bilinear), therefore, the estimation error in HSMU is 15.11% which is not improved significantly as the RMS error of 19.85% using the initial guess parameters. Fig. 10.16(a) illustrates that initial model under estimate the energy dissipated by the physical component in S1. Later in S2 as shown in Fig. 10.16(c), both initial model and updated model have similar behavior. This observation matches the conclusion from chapter 9, if the initial model is relatively accurate, it may not always be necessary to implement model updating.

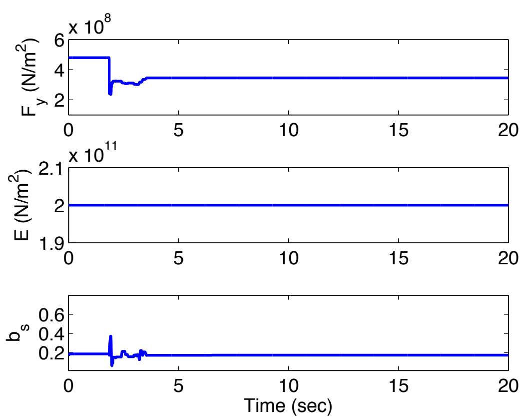
In addition, the results illustrate that the model updating performance is more related to the choice of the model. Comparing model updating accuracy in case I and II, the RMS error for the entire time history is 3.34 % for case I, and 13.21 % for case II. Because the EBW model can better capture the steel frame hysteresis than the bilinear model can. The model accuracy can further affect the fidelity of hybrid simulation results.

Table 10.3.: RMS Error in Two Model Updating Cases

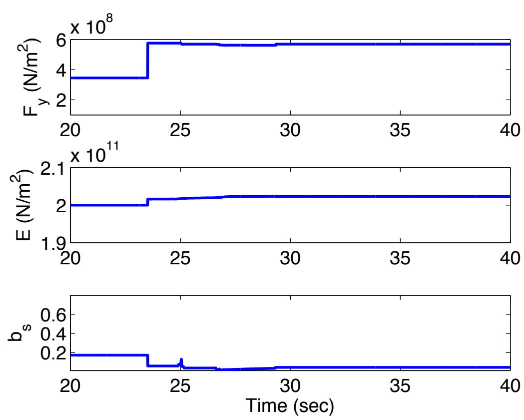
Error Case	Phenomenon Model	Constitutive Model
S1: ID Error	3.04%	8.39%
S1: Error with Initial Guess	21.39%	26.41%
S2: ID Error	3.59%	15.11%
S2: Error with Initial Guess	25.23%	19.85%
Entire Time History: ID Error	3.34%	13.21%
Entire Time History: Error with Initial Guess	22.78%	24.51%



(a) Entire: Parameter Convergence

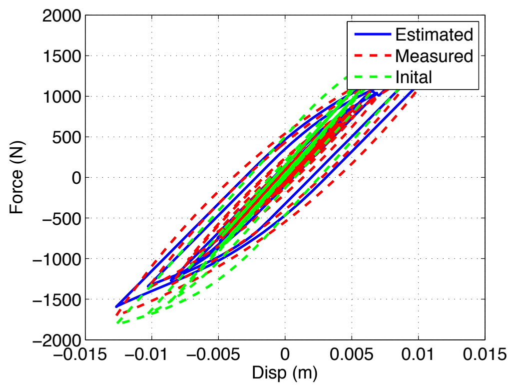


(b) S1: Parameter Convergence

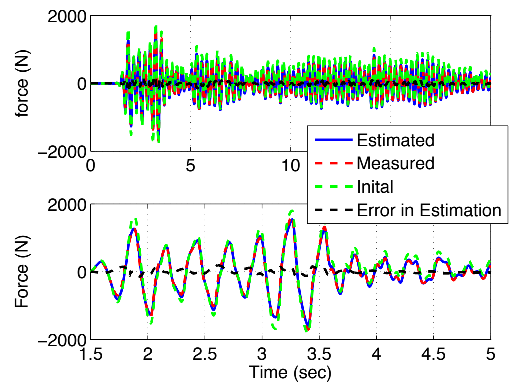


(c) S2: Parameter Convergence

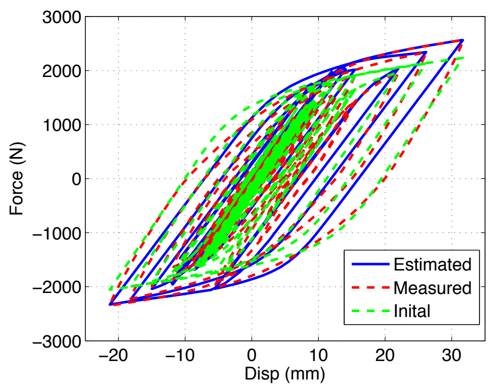
Figure 10.15.: Parameter Convergence for Case II



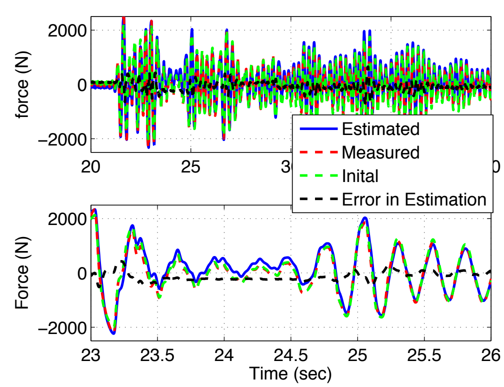
(a) S1: Hysteretic Behavior Comparison



(b) S1: Identification Time Domain Comparison



(c) S2: Hysteretic Behavior Comparison



(d) S2: Identification Time Domain Comparison

Figure 10.16.: Model Updating Performance Case II

10.4 Conclusion

To understand the model updating performance in HSMU, an experimental study is conducted. Both a phenomenological Bouc-Wen model and a bilinear steel constitutive model are used for numerical substructure and are updated during hybrid simulation. Two HSMU platforms and their implementation are presented. The main conclusions from the experimental study are as follows.

- The HSMU approach has been successfully implemented to both phenomenological and constitutive models. Results indicate that model response accuracy has been improved with model updating.

- Most model parameters converge quickly after the peak response occurs in the time history. This convergence can be the local optimization point for a given response history and a given model. If the peak response evolves largely later in the time history, the optimization set can adjust to a new convergence according to the updating process. A trilinear behavior is observed from model updating for the steel bilinear model, which matches the observation in the steel material test.

- The model updating estimation accuracy is largely improved in case I. However, due to the initial guess is more accurate and can be associated with its physical meaning in the bilinear model, the improvement in case II is less significant. Additionally, the bilinear model cannot represent the hysteresis behavior of the steel frame, the residual error after model updating is larger.

It should be left to the user to weight the trade off between model accuracy and complexity. Other constitutive models such as Menegotto-Pinto Model which are represented by structural physical meaning and can also accurately reproduce the Bauschinger effect are still desired. Thus, different model updating algorithms should be investigated in the future.

11. GLOBAL EVALUATION OF HYBRID SIMULATION WITH MODEL UPDATING

In the last couple of years, the HSMU concept has been successfully applied and validated in experimental examples. The assessment of the model updating performance emphasizes mainly on the local (model level) behavior. Evidence of the improvement of hybrid simulation fidelity are the model estimation accuracy and the parameter convergence during HSMU. In those studies, global responses are often compared to a baseline numerical simulation which may not represent the real response of the reference system when the structural behavior is highly nonlinear. In this chapter, to investigate the fidelity of HSMU on a global level, a shake table test is conducted. Structural displacement and acceleration responses in HSMU are compared with the measured structural responses in the shake table test. To further illustrate the fidelity improvement after HSMU, HSMU results are also compared with numerical simulations with the initial model parameters. Such an analysis is conducted for both case I (HSMU and simulation with phenomenological model) and case II (HSMU and simulation with phenomenological model), and observations on HSMU global performance using different models are also presented.

11.1 Shake Table Test Setup

For the shake table testing, the entire five story steel structure is mounted on a 6 DOFs shake table in the Intelligent Infrastructure System Lab. The shake table is controlled using the SW6000. The maximum stroke of actuators is ± 1.5 inch, the maximum acceleration is 10.2 g without payload, 3.1 g with 1000 lb payload. In this study, only the Y axis of the shake table is activated. During the shake table test, structural acceleration and displacement are measured using accelerometers and

LED sensors, as configured in Fig. 11.1. A high resolution data acquisition system VibPilot is used to acquire the acceleration data at 2048 Hz. A 6D tracking system Krypton is used to measure the position of LEDs which calculates the 6D position and dynamic movement of each LED. The sampling rate of the Krypton system is at 60 Hz.

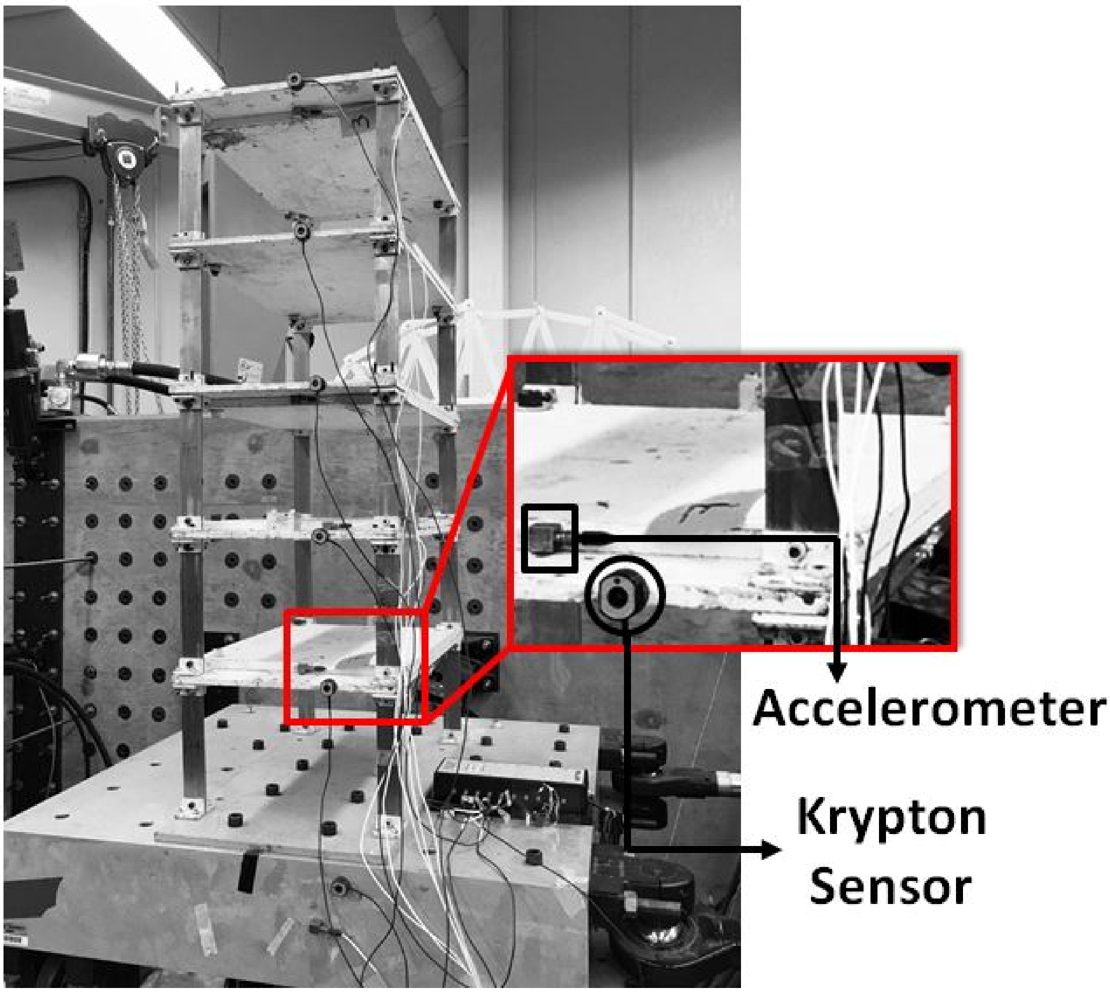
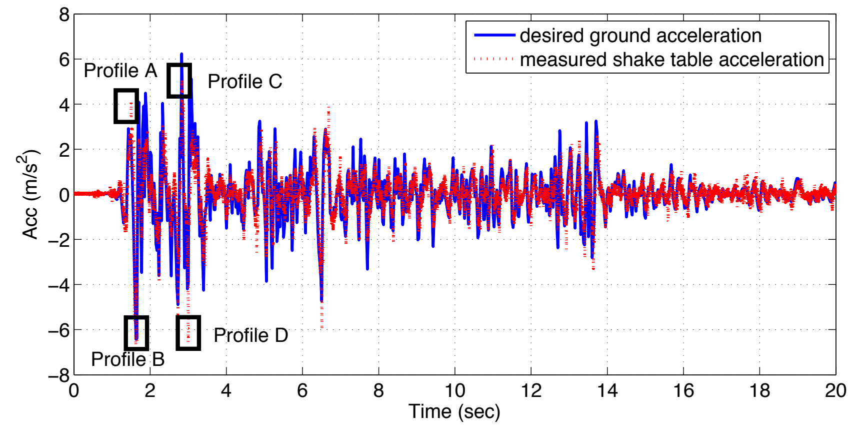
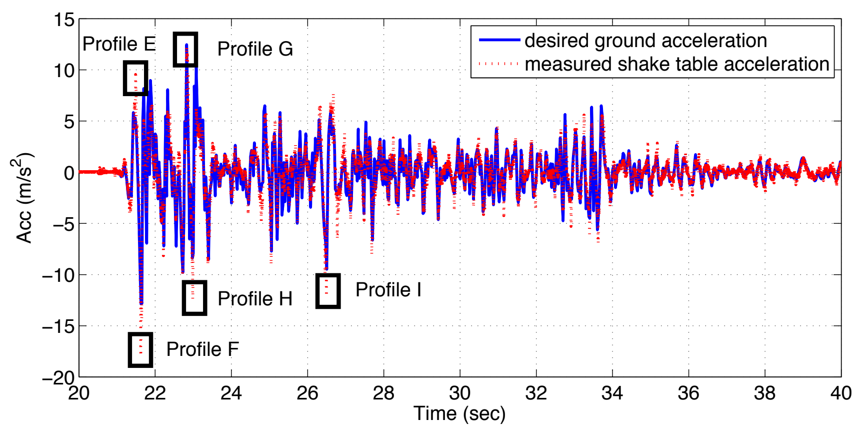


Figure 11.1.: Shake Table Test Configuration

As discussed in chapter 10, the ground excitation for HSMU is a measured acceleration response from the shake table test. This implemented shake table motion is a sequential combination of two scaled El-Centro earthquake records. As stated earlier, this ground excitation of the structure can be divided into section 1 (first 20 second, indicated as S1) and section 2 (last 20 second, indicated as S2). The comparison between desired excitation input and implemented acceleration of the shake table is shown in Fig. 11.2.



(a) S1: Shake Table Reccord



(b) S2: Shake Table Reccord

Figure 11.2.: Implementation of Shake Table Ground Motion

11.2 Response Comparison between HSMU and Shake Table Test: Section I

In this analysis, the shake table test measured responses are compared to HSMU case I and case II, (in this chapter, they are labeled as HSMU-BW for case I and

HSMU-BL for Case II). Additionally, computational simulation cases using initial guess parameters are conducted for each model, results are labeled as SIM-BW and SIM-BL, respectively. Time histories of displacement and acceleration responses at each floor are illustrated. In addition, structural displacement profiles and acceleration profiles at several peak responses are presented. The locations of those peaks are indicated in Fig. 11.2(a) and 11.2(b), labeled from A to I.

Two time domain error indices are introduced to quantify the performance at the peak responses, as following:

$$J_{1,j} = \sum_{i=1}^{5} \frac{x_j(i) - x_{s,j}(i)}{\sum_{i=1}^{5} x_{s,j}(i)} \tag{11.1}$$

$$J_{2,j} = \sum_{i=1}^{5} \frac{\ddot{x}_j(i) - \ddot{x}_{s,j}(i)}{\sum_{i=1}^{5} \ddot{x}_{s,j}(i)} \tag{11.2}$$

where J_1 is the peak displacement error, J_2 is the peak acceleration error, x_s indicates measured displacement from shake table test, $\ddot{x}_s$ indicates measured acceleration from shake table test, j is the profile case number.

In S1 (0-20 sec), the entire structure is excited with the first sequence of the El-Centro ground motion, the displacement and the acceleration responses time history are presented in Fig. 11.11 - 11.15 for each floor. This comparison is between measured response from the shake table, HSMU-BW, and HSMU-BL. As stated in chapter 10, the phenomenological model can better capture the specimen hysteresis behavior, response yielded from HSMU-BW is more accurate and falls on top of the shake table response, especially for displacement time histories. The bilinear model cannot capture the Bauschinger effect of each steel frame and it underestimates the energy dissipated in each hysteresis loop, therefore, some overshoots are observed in both displacement and acceleration responses in HSMU-BL.

In the displacement profile analysis, in Fig. 11.3, simulation using initial EBW model parameters (SIM-BW) underestimates the maximum displacement in profile B, C and D, which is due to the over estimation in energy dissipation. This can be demonstrated using hysteresis behavior in Fig. 11.16, in higher floors (2-5), energy dissipation is significant larger for SIM-BW case than HSMU-BW, therefore the displacement is smaller. In the contrast, simulation using initial bilinear steel model (SIM-BL) over estimates the maximum drift in case B, C, and D. However, the difference is less significant. This also can be demonstrated using hysteresis behavior in Fig. 11.17, HSMU-BL case yields nonlinear behavior up to 3^{rd} and 4^{th} floor, in which SIM-BL case they are linear due to a larger yield stress $F_y = 480\ N/m^2$.

The quantified displacement errors are listed in Table 11.1. Among all cases, HSMU-BW has the least error (in red) for profile A and C and D, and SIM-BW has the largest error for all locations (in blue). The improvement is significant after model updating, total error reduces from 1.526 to 0.1681. Error in the displacement is only slightly reduced from 0.4492 to 0.4021 as comparing the HSMU-BL with the SIM-BL. One explanation is that for the bilinear model, the model updating efficiency is taken over by the inherent modeling error (the selected model is not sufficient to represent a certain behavior).

Similarly, measured shake table acceleration responses are compared with HSMU-BW and HSMU-BL results. With the sampling frequency at 2048 Hz, the measured acceleration responses contain high frequency components. These high frequency contents can be contributed by responses from strong axis excitation and twisting modes, which are not captured using one dimensional model. Also, noise in the accelerometers is more significant at high frequency, especially when the acceleration data magnitude is small at the first two stories. In the displacement responses, observations are similar in which the SIM-BW case underestimates the acceleration response and SIM-BL yields an overestimates the peaks.

The peak acceleration locations are selected according to locations labeled in Fig. 11.2(a). Fig. 11.4 and Table 11.2 illustrate the acceleration profiles and J_2 errors for different cases. HSMU-BL has the best performance for case A and D, also the total error reduces from 0.75 to 0.41 compared to SIM-BL. The visualization of the two error indices for different cases are shown in Fig. 11.10. Among all cases, SIM-

BW has the largest total error of 1.1526 because it is difficult to start an accurate initial guess. HSMU-BW yields a total J_2 error of 0.4531, also confirms the significant improvement from SIM-BW to HSMU-BW.

In the frequency domain, the power spectrum of displacement at each floor is compared for all cases, which is shown through Fig. 11.5 - 11.9. The frequency results indicate that HSMU-BW can accurately estimate the first structural mode, where the higher modes are lightly more damped compared to the shake table response. But the improvement still is noticeable as compared to the SIM-BW case, in which higher modes are largely damped (especially in Fig. 11.8 and 11.9) due to the over estimation on energy dissipation. For bilinear model, in SIM-BL case, comparing to the shake table results, the third mode and fourth mode are shifted using the initial guess and are modified after apply model updating in HSMU. Also in SIM-BW, these higher modes are less damped compared to the shake table results. The energy dissipation on higher floors are much more accurate with HSMU-BL. This finding also demonstrate the effectiveness of model updating on both models.

Table 11.1.: J_1 Peak Displacement Error

Case	Profile A	Profile B	Profile C	Profile D	Total
HSMU-BL	0.1466	0.0475	0.1583	0.0497	0.4021
HSMU-BW	0.0381	0.0479	0.0373	0.0447	0.1681
SIM-BL	0.0473	0.1360	0.1630	0.1028	0.4492
SIM-BWBN	0.1552	0.3435	0.2959	0.3579	1.1526

Table 11.2.: J_2 Peak Acceleration Error

Case	Profile A	Profile B	Profile C	Profile D	Total
HSMU-BL	0.0826	0.1212	0.0738	0.1381	0.4157
HSMU-BW	0.1360	0.1400	0.0314	0.1439	0.4513
SIM-BL	0.1066	0.1074	0.2040	0.3333	0.7514
SIM-BW	0.1192	0.3142	0.1543	0.3068	0.8945

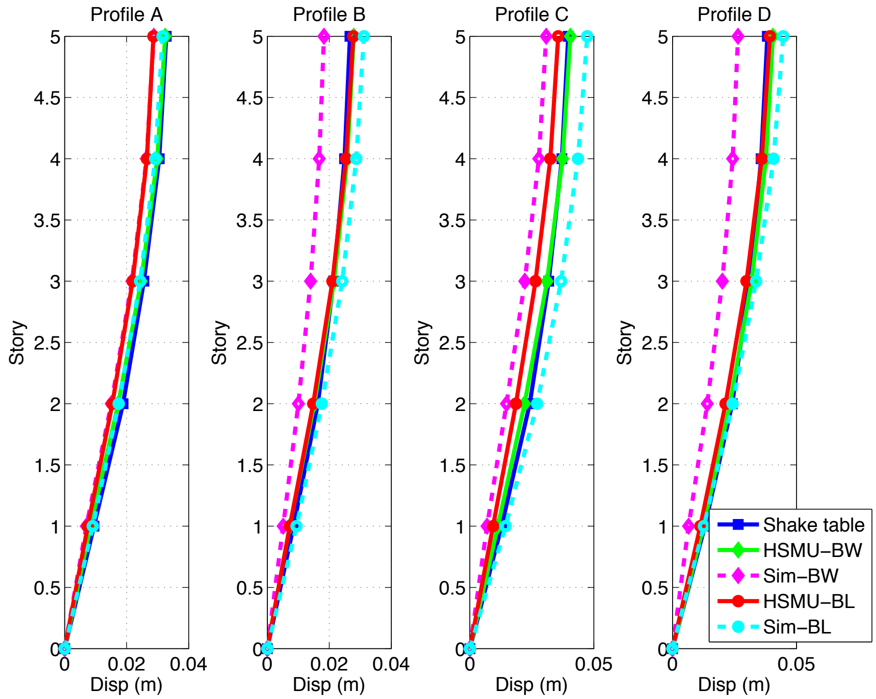


Figure 11.3.: Displacement Profile at Peak Locations

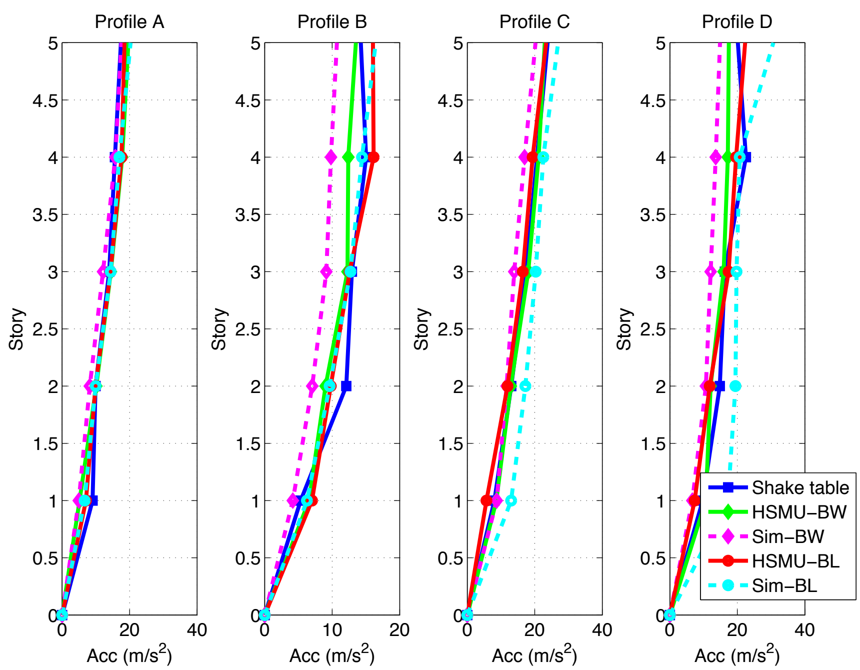


Figure 11.4.: Acceleration Profile at Peak Locations

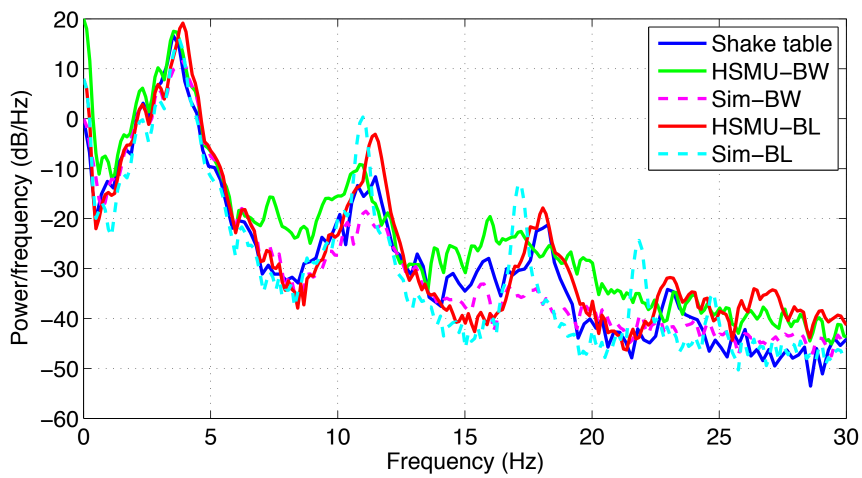


Figure 11.5.: S1: Frequency Domain Anlaysis Floor 1

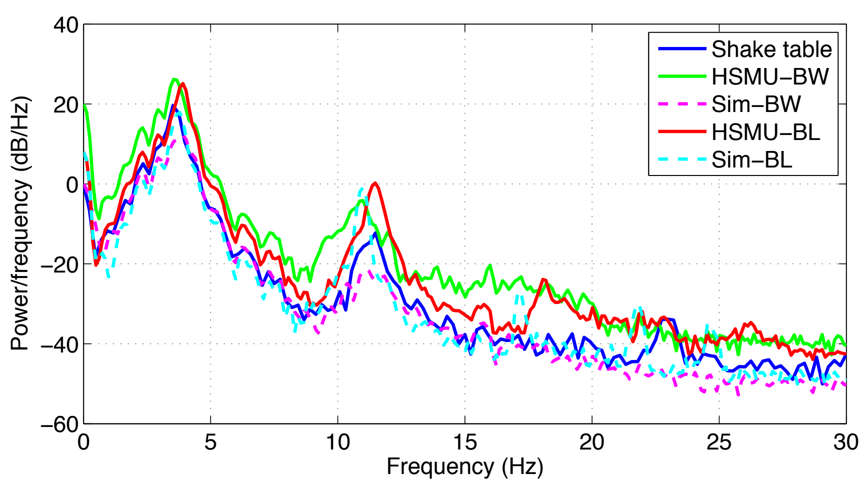


Figure 11.6.: S1: Frequency Domain Anlaysis Floor 2

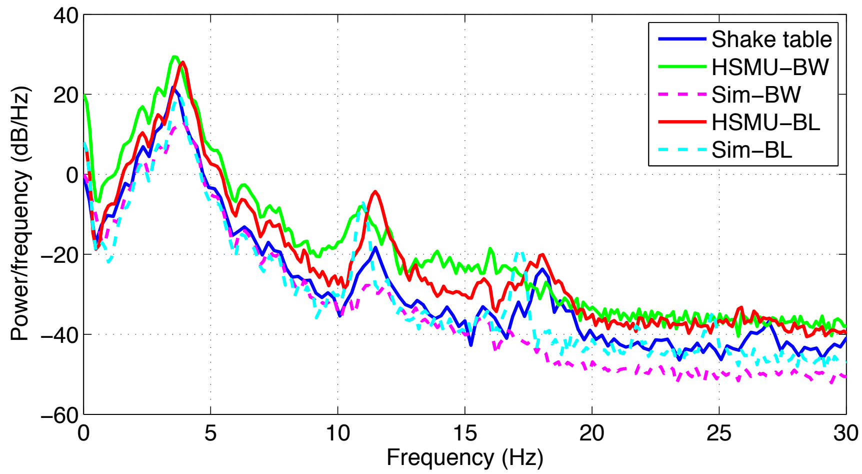


Figure 11.7.: S1: Frequency Domain Anlaysis Floor 3

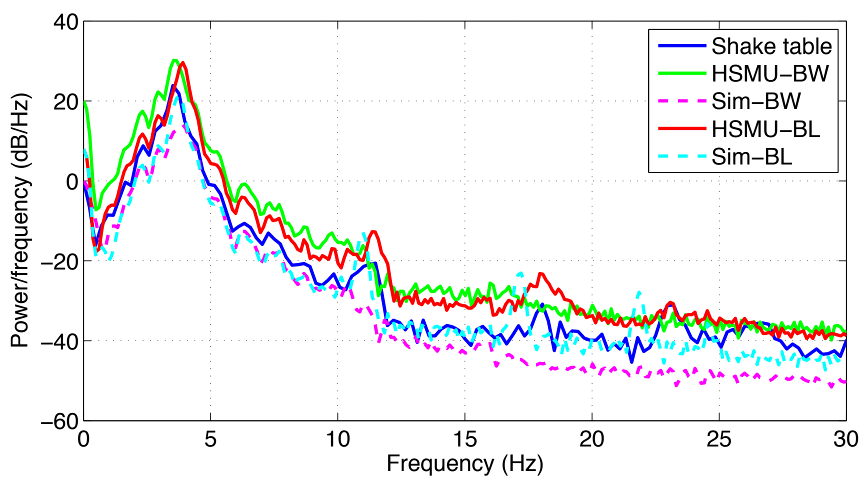


Figure 11.8.: S1: Frequency Domain Anlaysis Floor 4

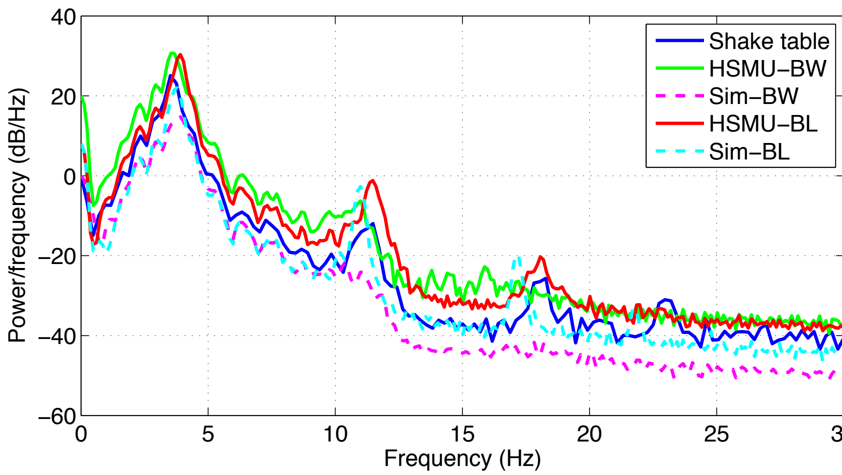
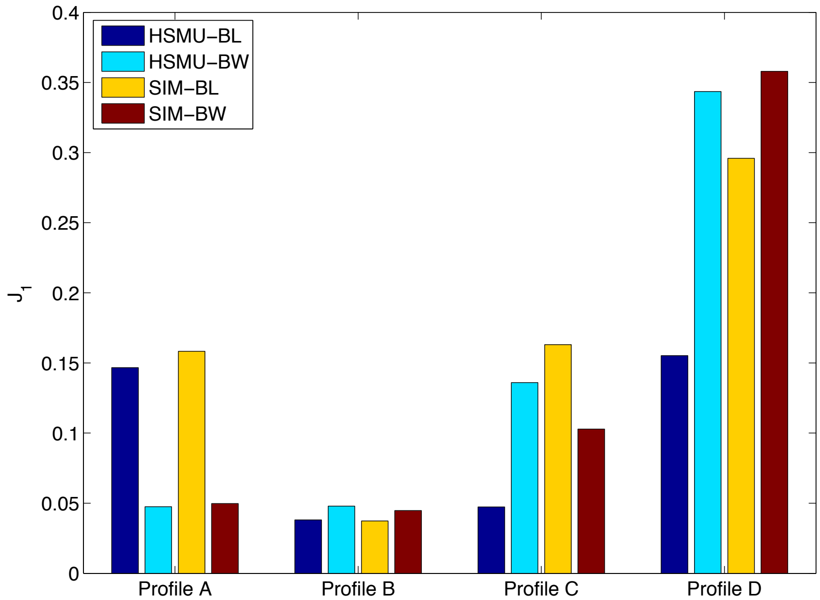
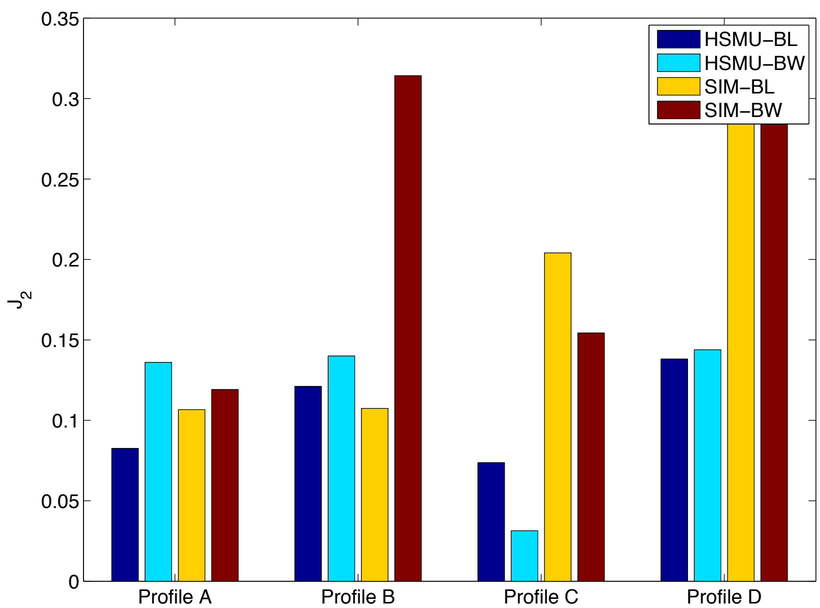


Figure 11.9.: S1: Frequency Domain Anlaysis Floor 5

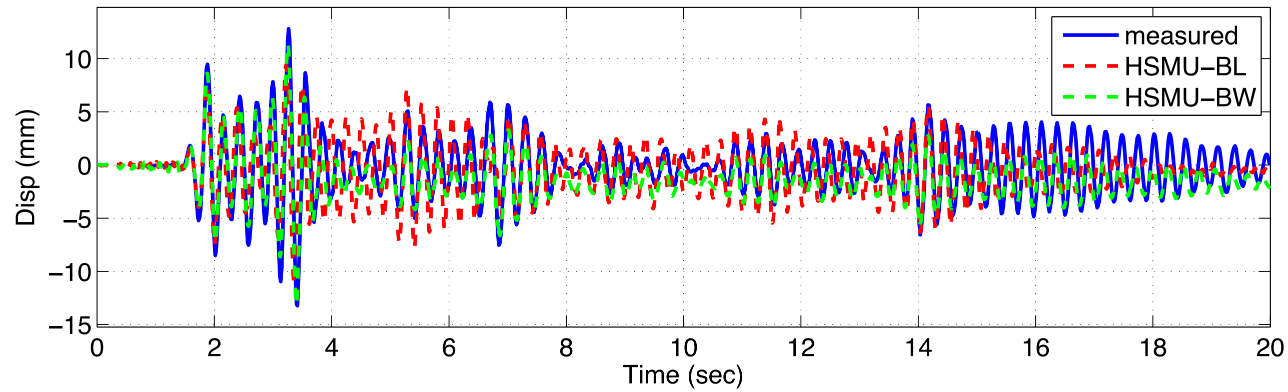


(a) S1: Peak Displacement Error J_1

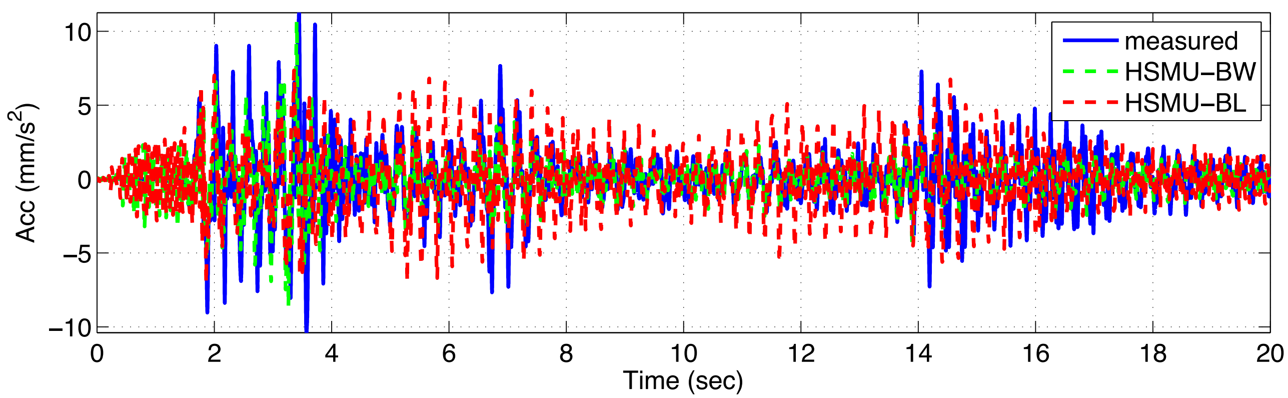


(b) S1: Peak Acceleration Error J_2

Figure 11.10.: S1: Error Indices Comparison for Peak Responses

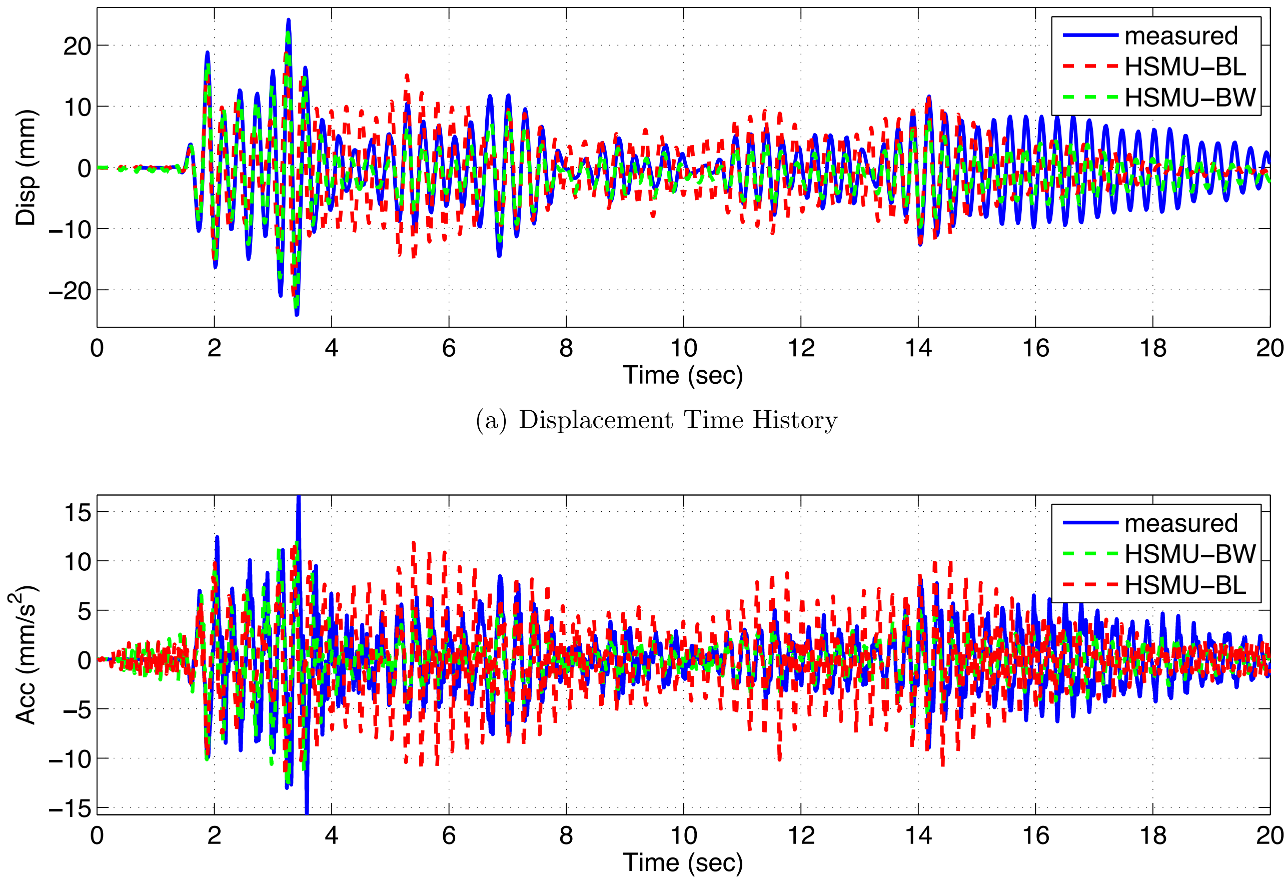


(a) Displacement Time History



(b) Acceleration Time History

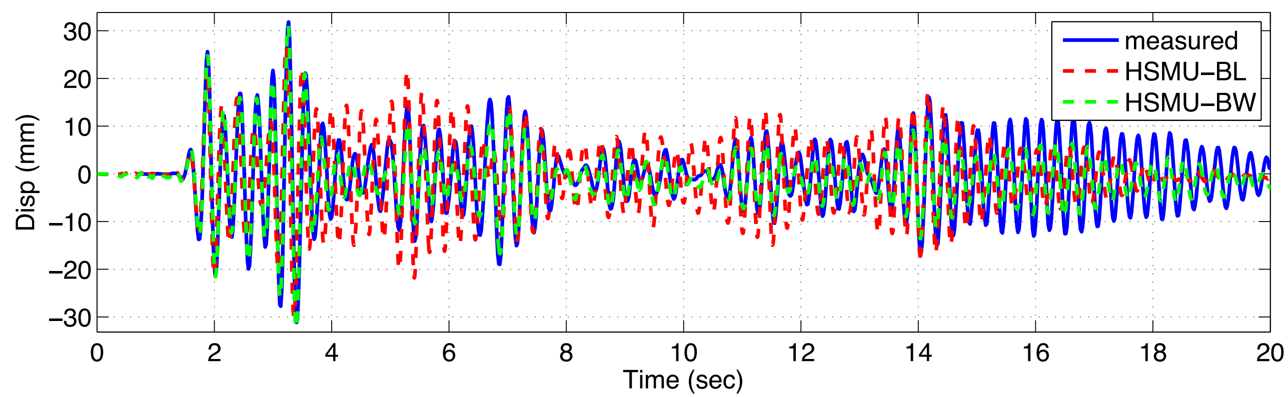
Figure 11.11.: S1: Floor 1 Time History Responses



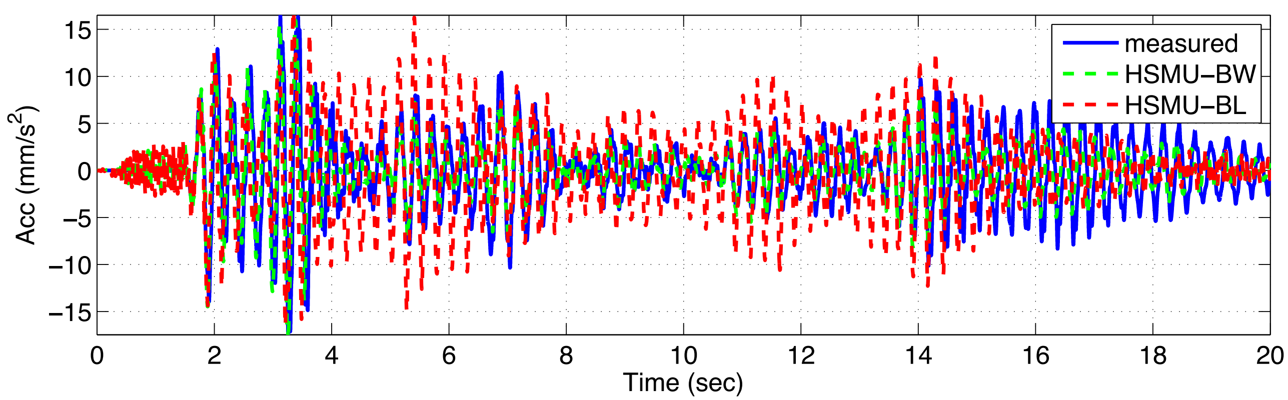
(a) Displacement Time History

(b) Acceleration Time History

Figure 11.12.: S1: Floor 2 Time History Responses

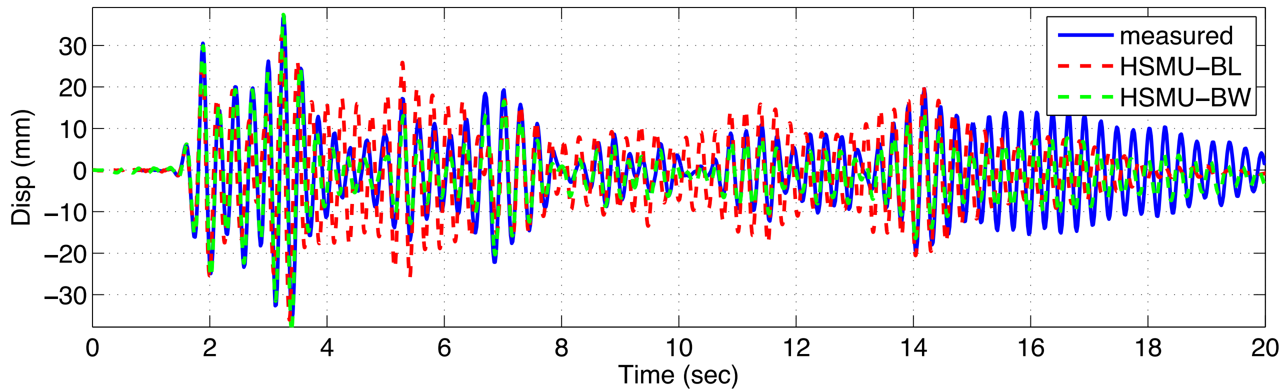


(a) Displacement Time History

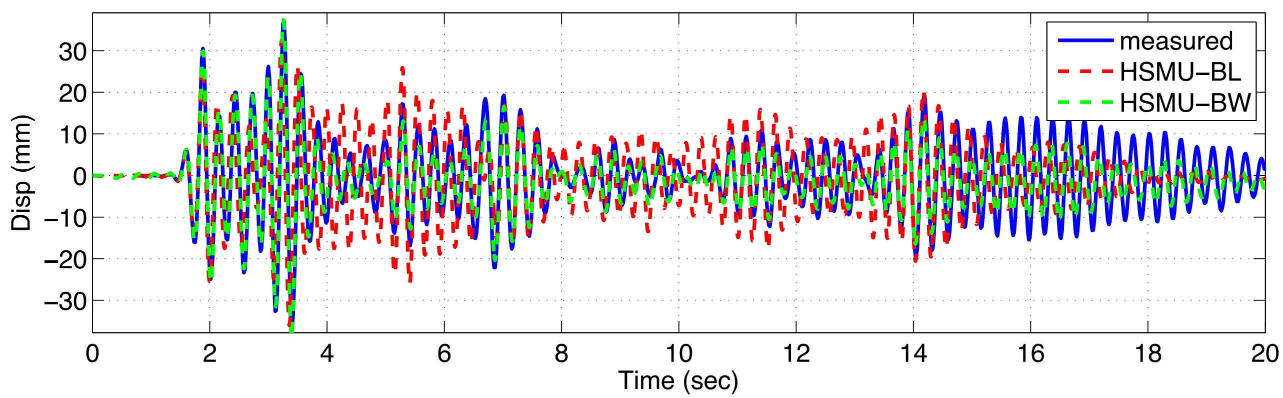


(b) Acceleration Time History

Figure 11.13.: S1: Floor 3 Time History Responses

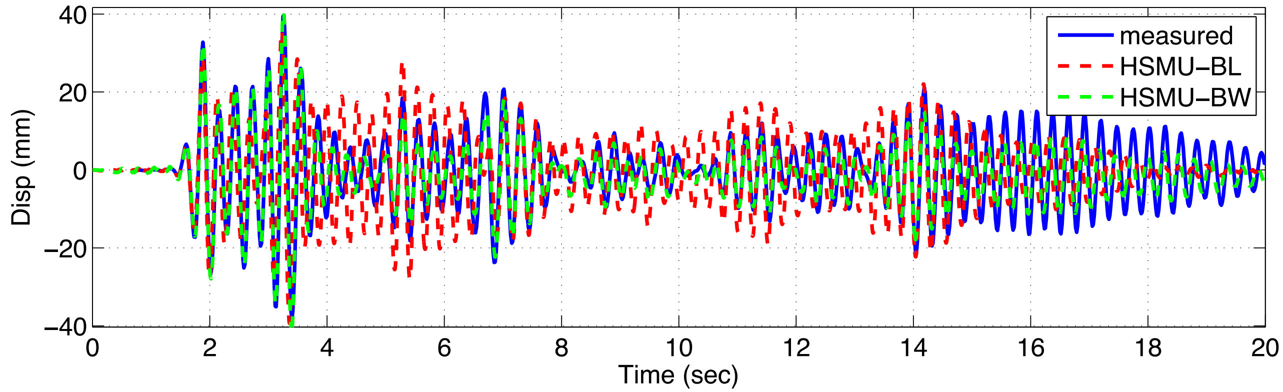


(a) Displacement Time History

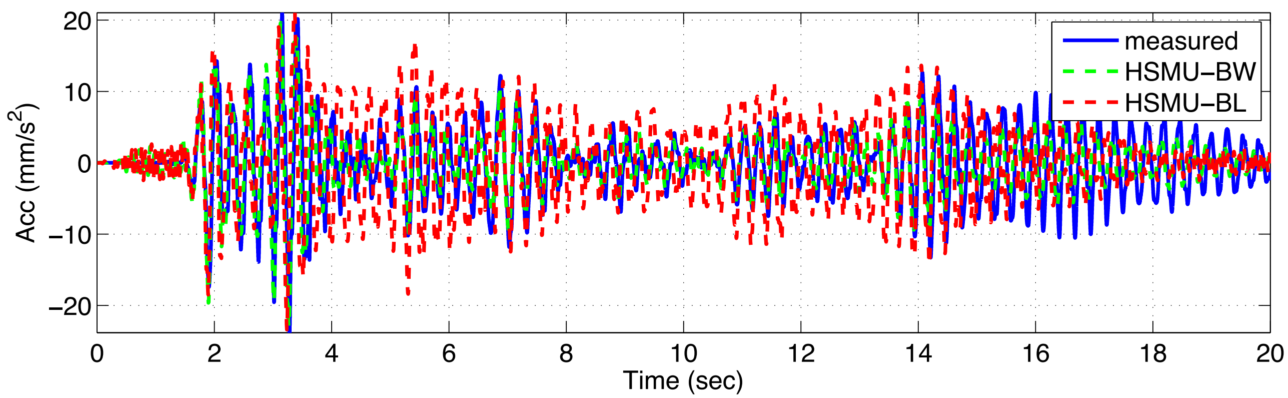


(b) Acceleration Time History

Figure 11.14.: S1: Floor 4 Time History Responses

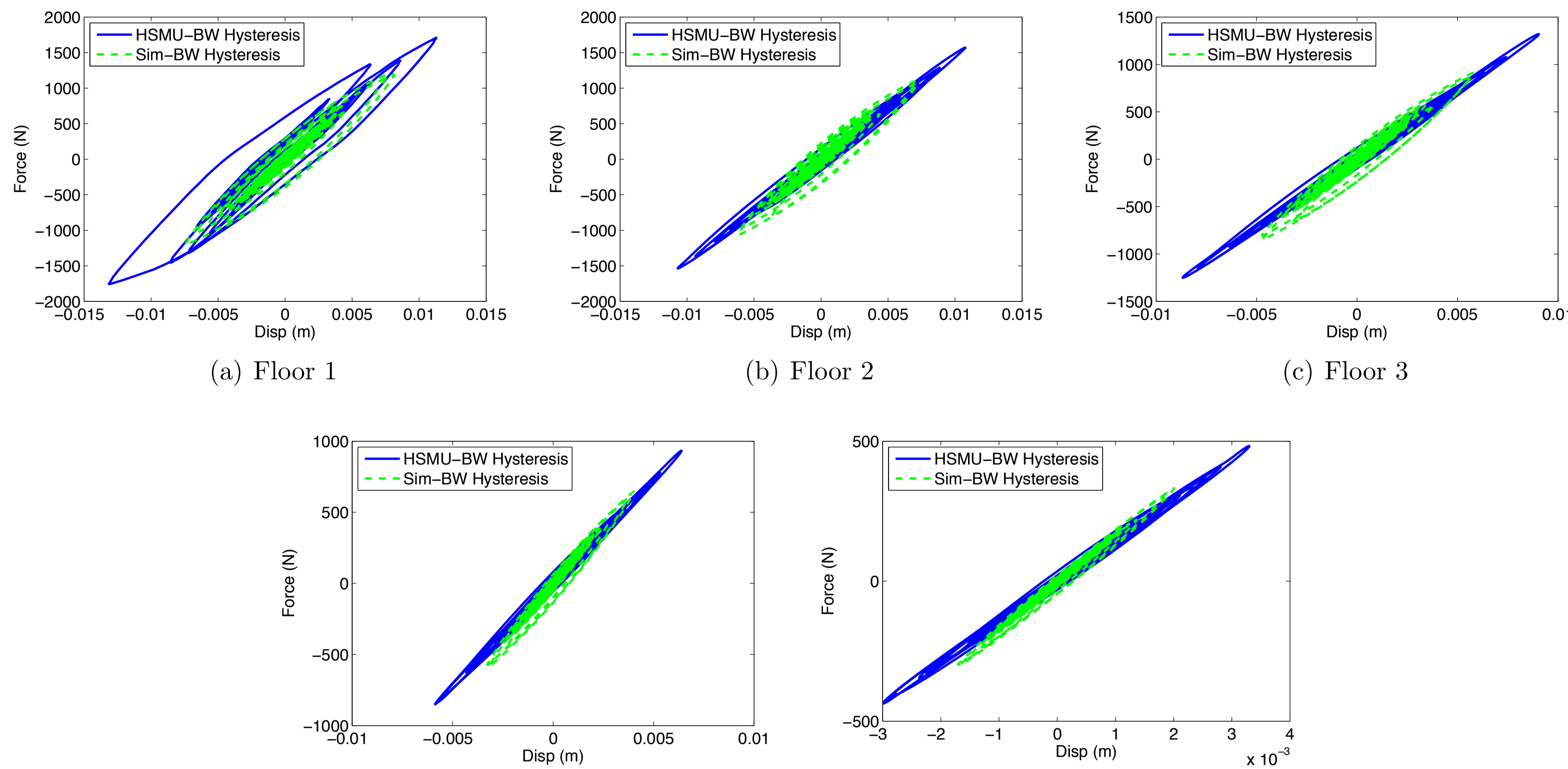


(a) Displacement Time History



(b) Acceleration Time History

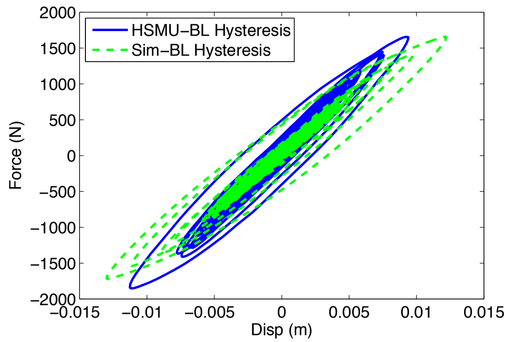
Figure 11.15.: S1: Floor 5 Time History Responses



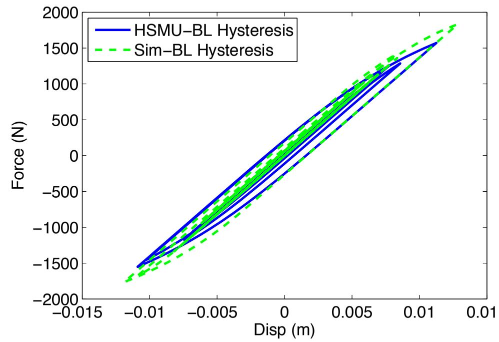
(a) Floor 1 (b) Floor 2 (c) Floor 3

(d) Floor 4 (e) Floor 5

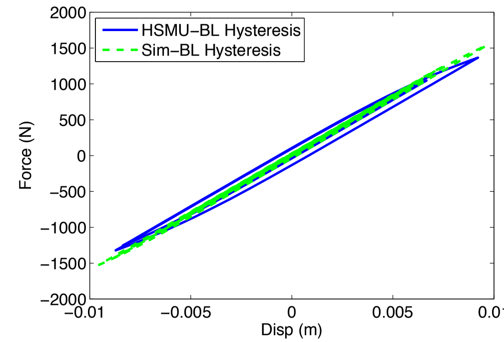
Figure 11.16.: S1: Hysteresis for Each Story - BW model



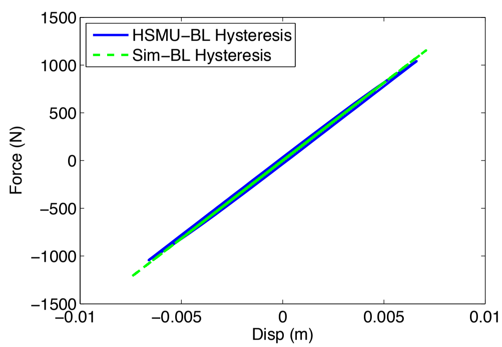
(a) Floor 1



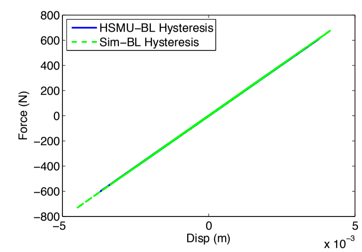
(b) Floor 2



(c) Floor 3



(d) Floor 4



(e) Floor 5

Figure 11.17.: S1: Hysteresis for Each Story - BL model

11.3 Response Comparison between HSMU and Shake Table Test: Section II

In section II, the steel frame is excited by a larger ground motion in which the peak reaches 18.3 m^2/s. The first story frame experienced a peak drift at 21.5 sec, labeled as location E. Later after E, a residual drift of 11.2 mm appears at the first floor, as shown in 11.26. This residual drift in the first floor also leads to residual drift in the higher floors, in Fig. 11.27 - 11.30. However, neither HSMU-BW nor HSMU-BW can capture such residual drift in the responses. Some possible explanations can be 1) deficiency in the connection manufacturing; 2) a fixed-end simplification of the connection is not sufficient; and 3) the inertia effect is numerical applied. Further studies to improve the residual drift prediction using simulation model are needed.

J_1 index reaches its maximum at peak location F for all cases, which is the first response peak (in the reverse direction) after the residual drift occurred at E. This can also be visualized in Fig. 11.19. In displacement profile G-I, because the residual drift of 11.2 mm always exists, the index value does not yield much meaningful information.

The hysteresis behavior comparison for BW model is shown in Fig. 11.31, clear improvement is shown for upper stories which indicates the effectiveness of model updating on this phenomenological model. For bilinear model, the hysteresis behavior comparison is shown in Fig 11.32. Because the nonlinear parameter F_y and b_s bounced back close to the initial guess value, the hysteresis behavior is similar for floor 2-5.

Results are more informative in the acceleration responses comparison, where the time histories are more accurate. From Fig. 11.26 - 11.30, HSMU-BW can better match the measured acceleration, where HSMU-BL yields overshoots at several peak responses. These observations are also shown in the quantification analysis, larger errors are presented for peak locations F, G, H, I in HSMU-BL, as compared to HSMU-BW. HSMU-BW yields the smallest error in J_2 (with total error 0.6642) which indicates the responses are more accurate. This error is reduced from 1.5968 as in the SIM-BW, which has the largest error among all the cases. Combining with the observations in chapter 10, it shows that the model updating process is very effective for the phenomenological model and can be adaptive to different excitation amplitudes.

The improvement in bilinear model updating is also observed, in which the total J_2 error improves from 1.43 as in SIM-BL to 0.95 as in HSMU-BL. This improvement is more significant compared to the results in S1 (first 20 secs). Because the parameters are close in SIM-BL and HSMU-BL, the main reason of the improvement of HSMU fidelity is that the first floor response of the HSMU-BL is the physical measurement from the experimental substructure. Even the bilinear model cannot capture the Bauschinger effect well for upper stories, the critical first floor response is the true response from the specimen.

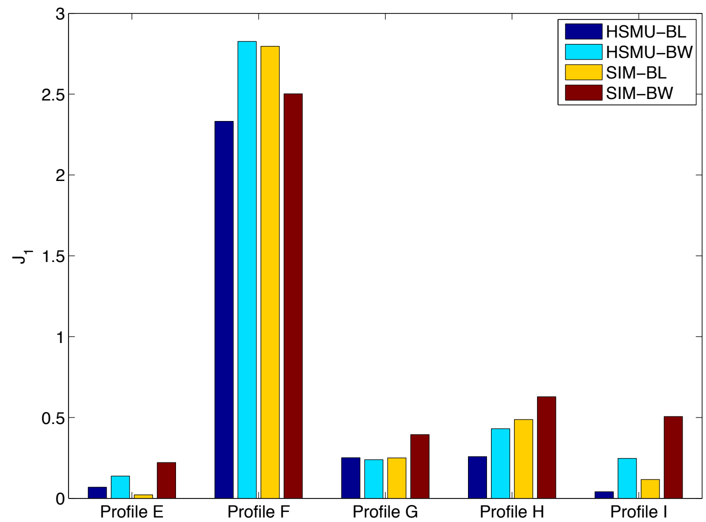
Frequency domain analysis are also carried on for all cases during S2, similar findings are illustrated as in S1, through Fig. 11.21 - 11.25. The frequency results indicate that HSMU-BW can accurately estimate the first structural mode. In SIM-BW case the higher modes are largely damped (especially in Fig. 11.8 and 11.9) which is the evidence of the overestimation on energy dissipation. For bilinear model, after apply model updating in HSMU, the performance on higher modes are improved compared to SIM-BW.

Table 11.3.: S2: J_1 Peak Displacement Error

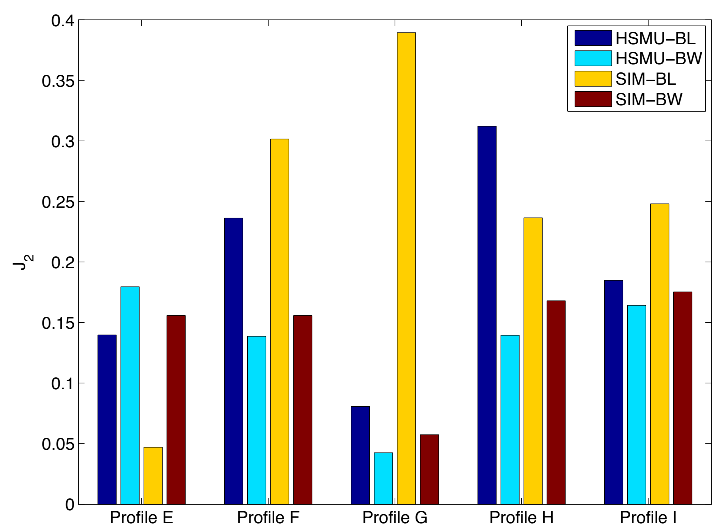
Case	Profile E	Profile F	Profile G	Profile H	Profile I	Total	Total w/o F
HSMU-BL	0.069	2.3316	0.2513	0.2579	0.0409	2.9507	0.6191
HSMU-BW	0.1375	2.8261	0.2390	0.4308	0.2472	3.8805	1.0544
SIM-BL	0.0217	2.7960	0.2498	0.4872	0.1164	3.6711	0.8751
SIM-BW	0.2215	2.5020	0.3941	0.6282	0.5058	4.2516	1.7496

Table 11.4.: S2: J_2 Peak Acceleration Error

Case	Profile E	Profile F	Profile G	Profile H	Profile I	Total
HSMU-BL	0.1397	0.2363	0.0807	0.3121	0.1849	0.9537
HSMU-BW	0.1795	0.1386	0.0424	0.1395	0.1642	0.6642
SIM-BL	0.0470	0.3016	0.5294	0.2999	0.2480	1.4259
SIM-BW	0.4612	0.1437	0.2180	0.2955	0.4784	1.5968



(a) Peak Displacement Error



(b) Peak Acceleration Error

Figure 11.18.: Error Indices Comparison for Peak Responses

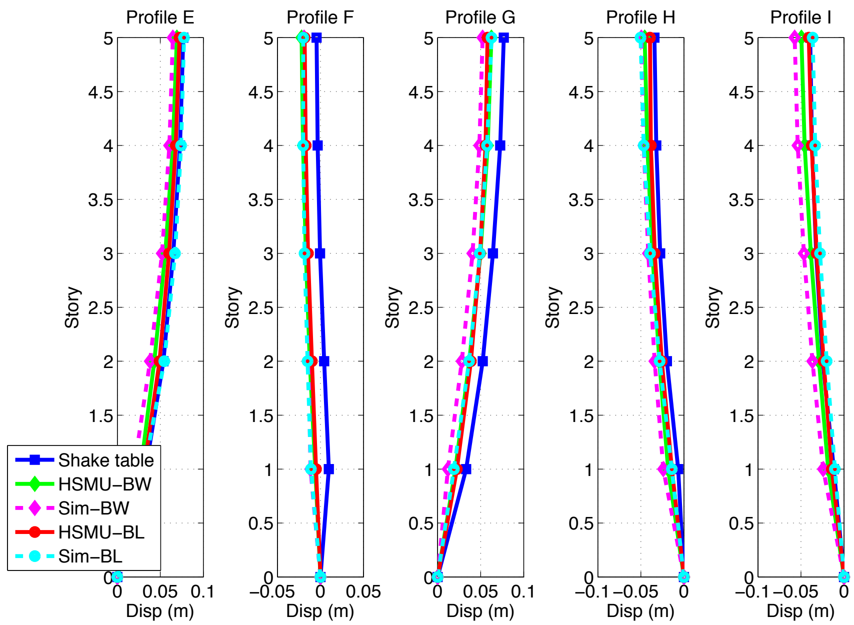


Figure 11.19.: Displacement Profile at Peak Locations

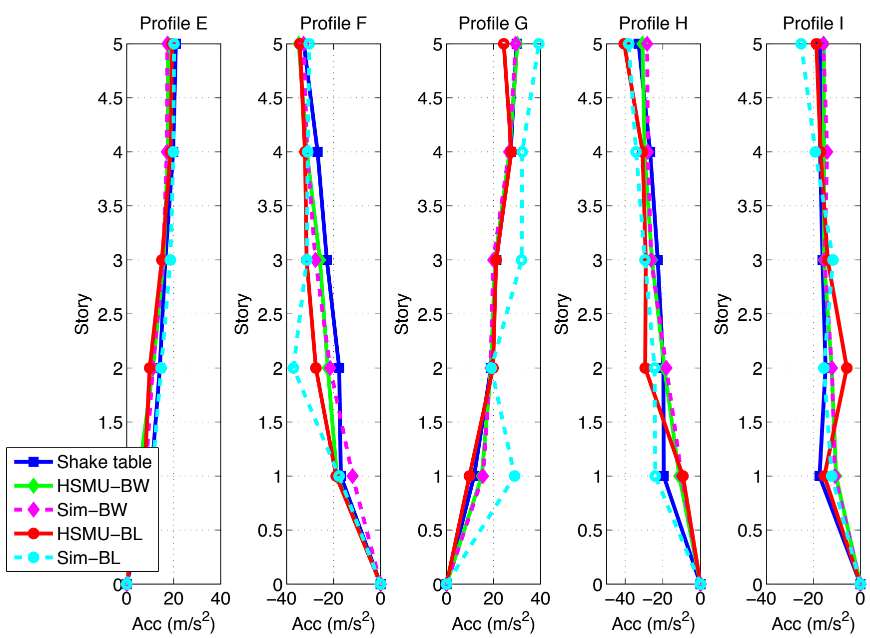


Figure 11.20.: Acceleration Profile at Peak Locations

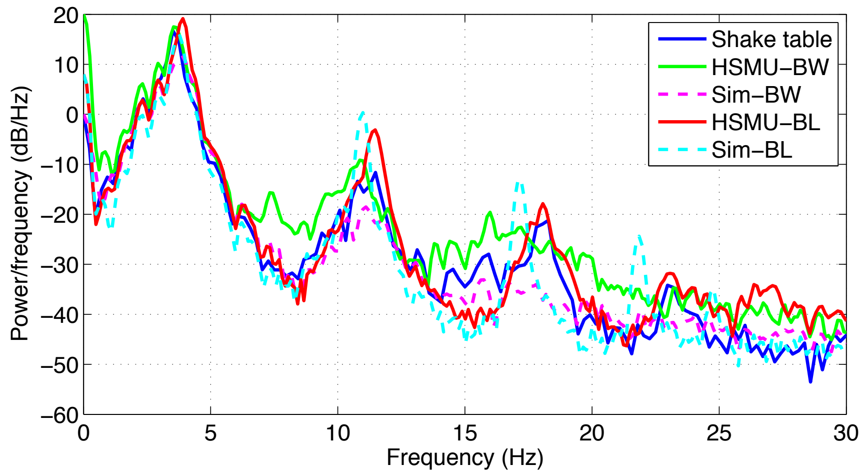


Figure 11.21.: S2: Frequency Domain Anlaysis Floor 1

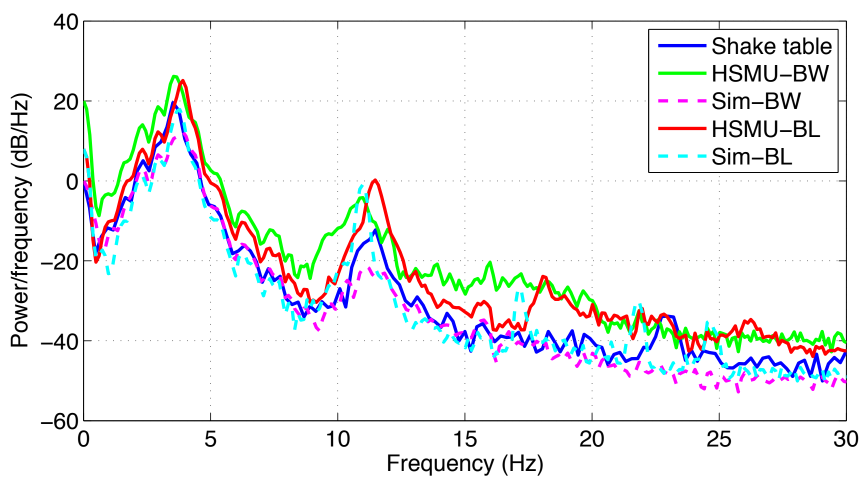


Figure 11.22.: S2: Frequency Domain Anlaysis Floor 2

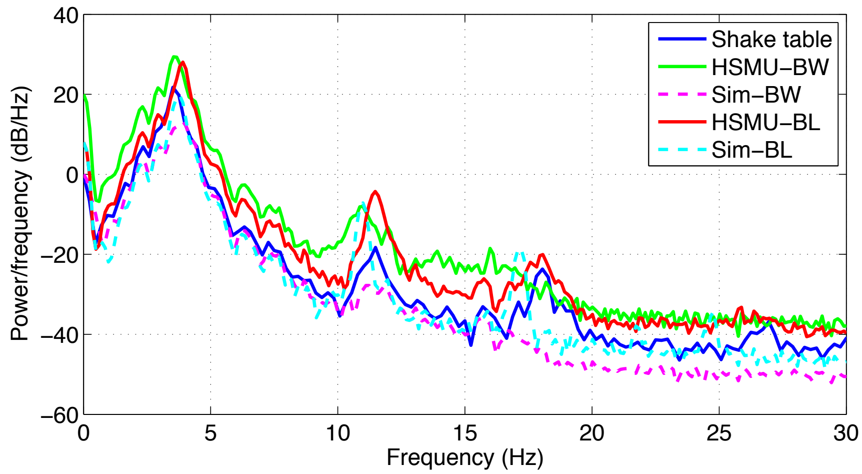


Figure 11.23.: S2: Frequency Domain Anlaysis Floor 3

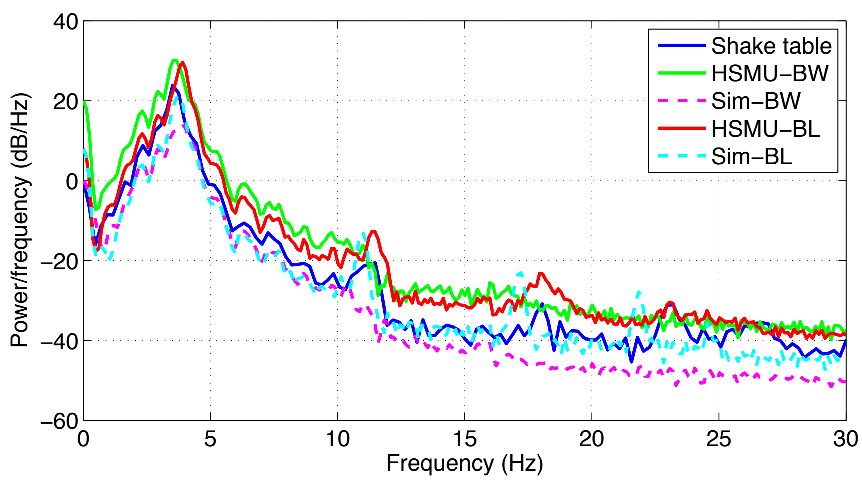


Figure 11.24.: S2: Frequency Domain Anlaysis Floor 4

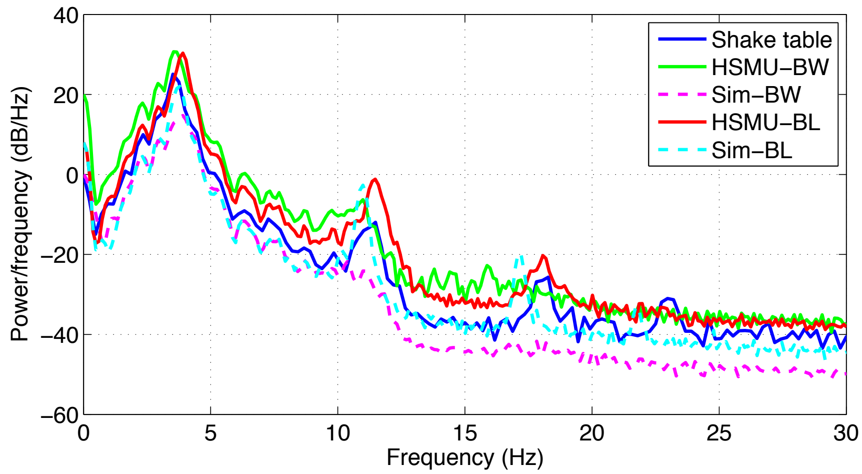
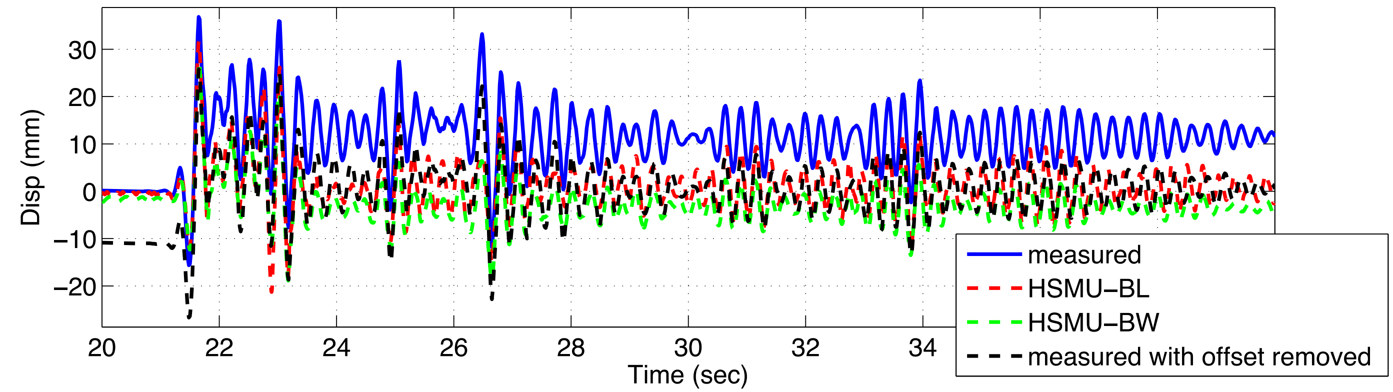
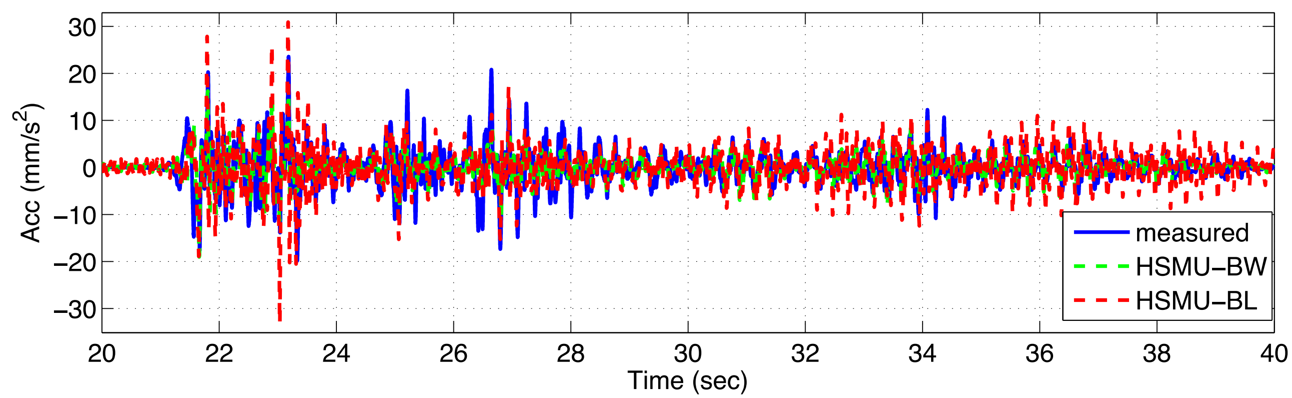


Figure 11.25.: S2: Frequency Domain Anlaysis Floor 5

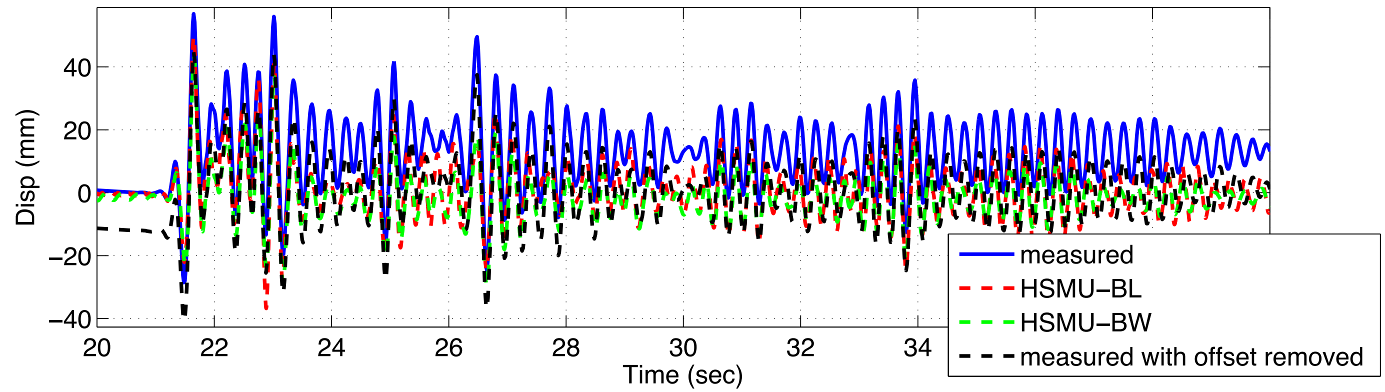


(a) Displacement Time History

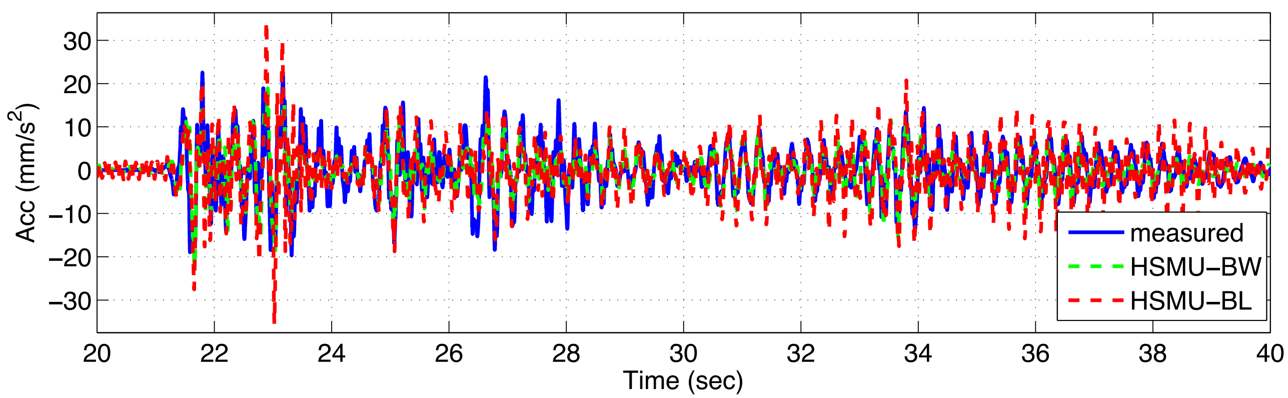


(b) Acceleration Time History

Figure 11.26.: S1: Floor 1 Time History Responses

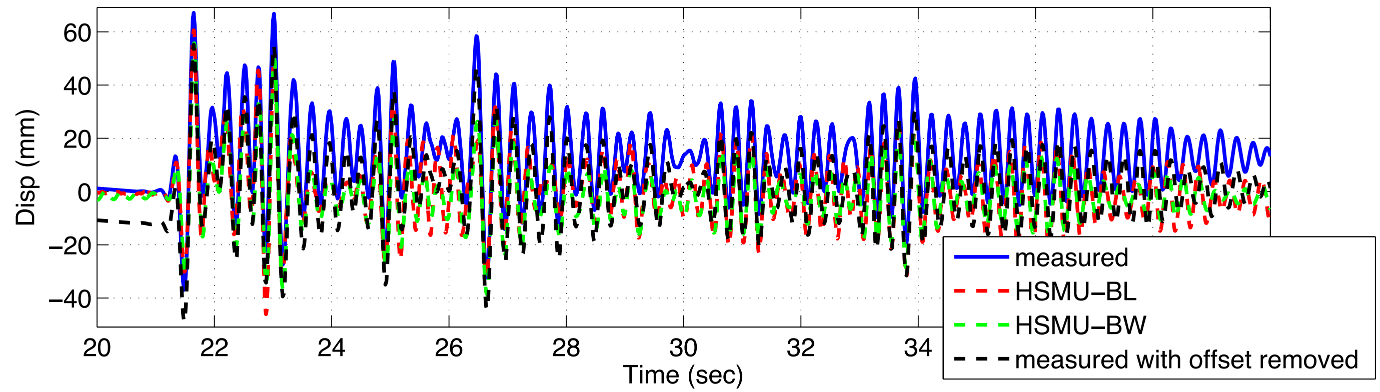


(a) Displacement Time History

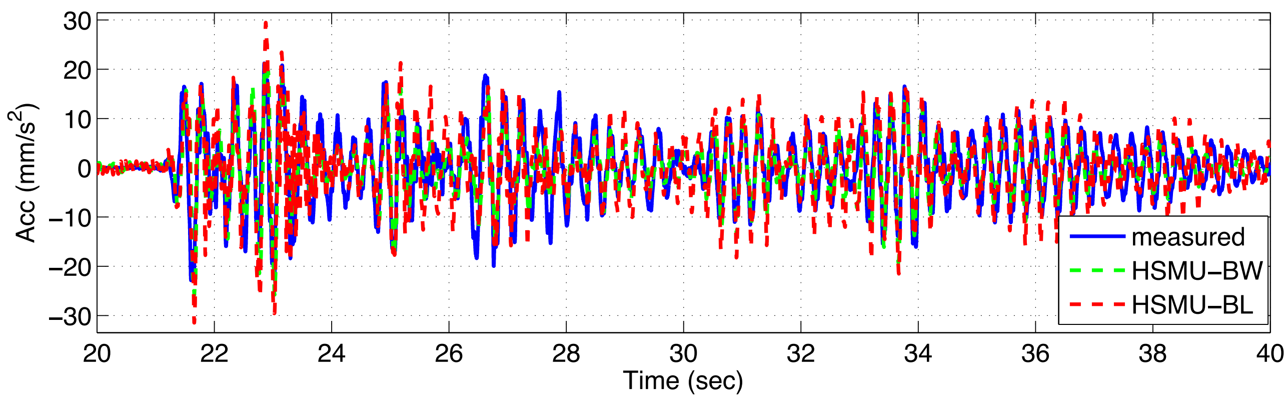


(b) Acceleration Time History

Figure 11.27.: S1: Floor 2 Time History Responses

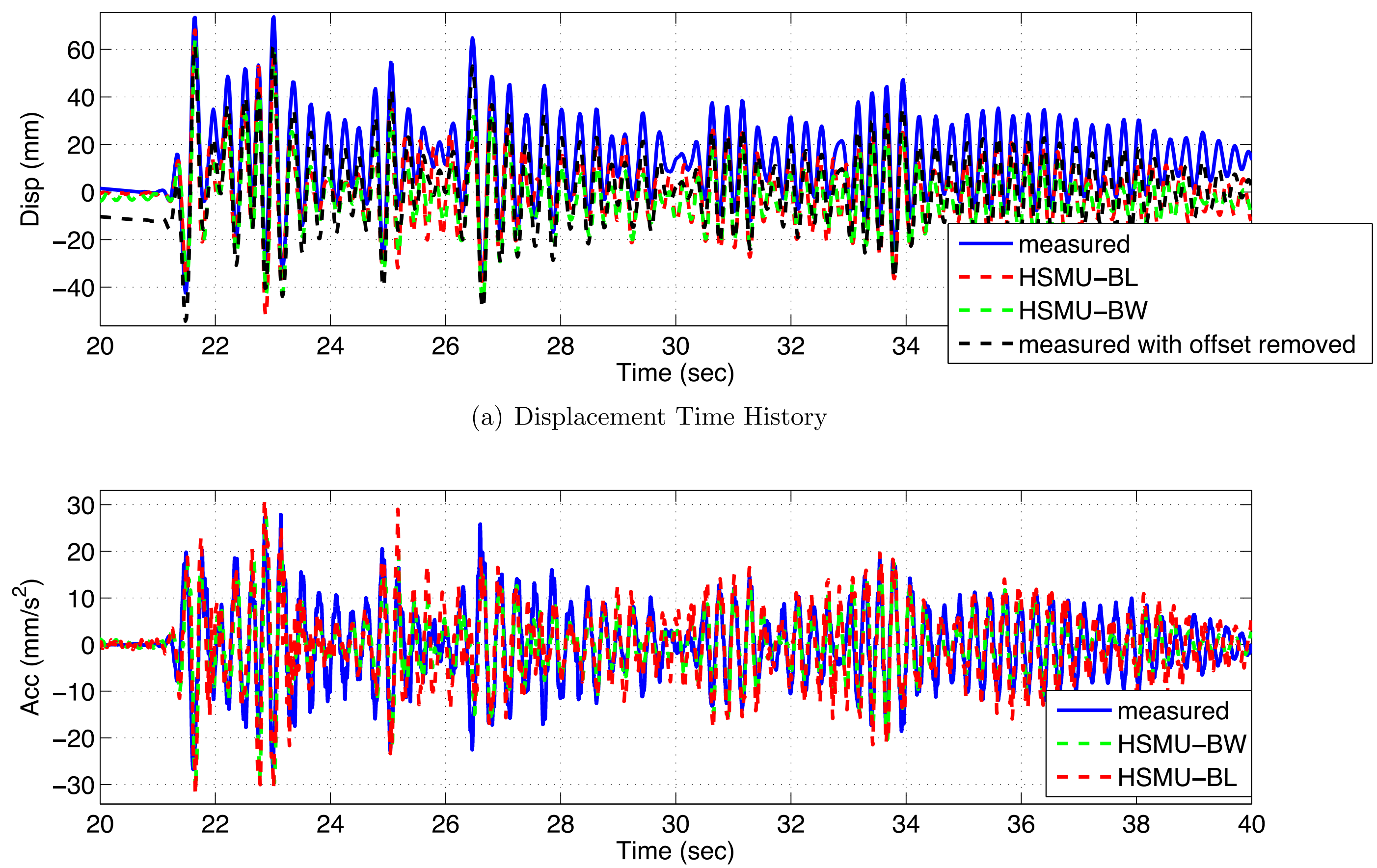


(a) Displacement Time History



(b) Acceleration Time History

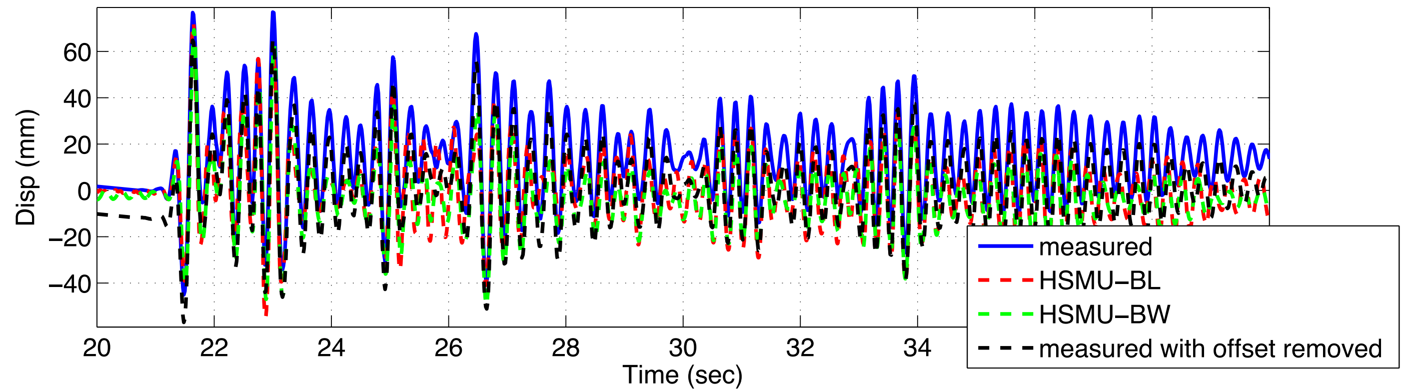
Figure 11.28.: S1: Floor 3 Time History Responses



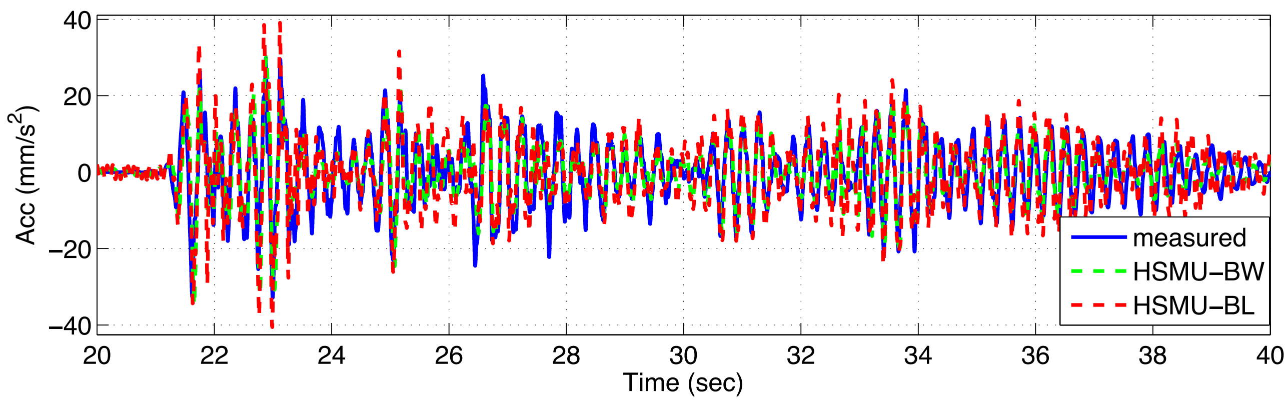
(a) Displacement Time History

(b) Acceleration Time History

Figure 11.29.: S1: Floor 4 Time History Responses

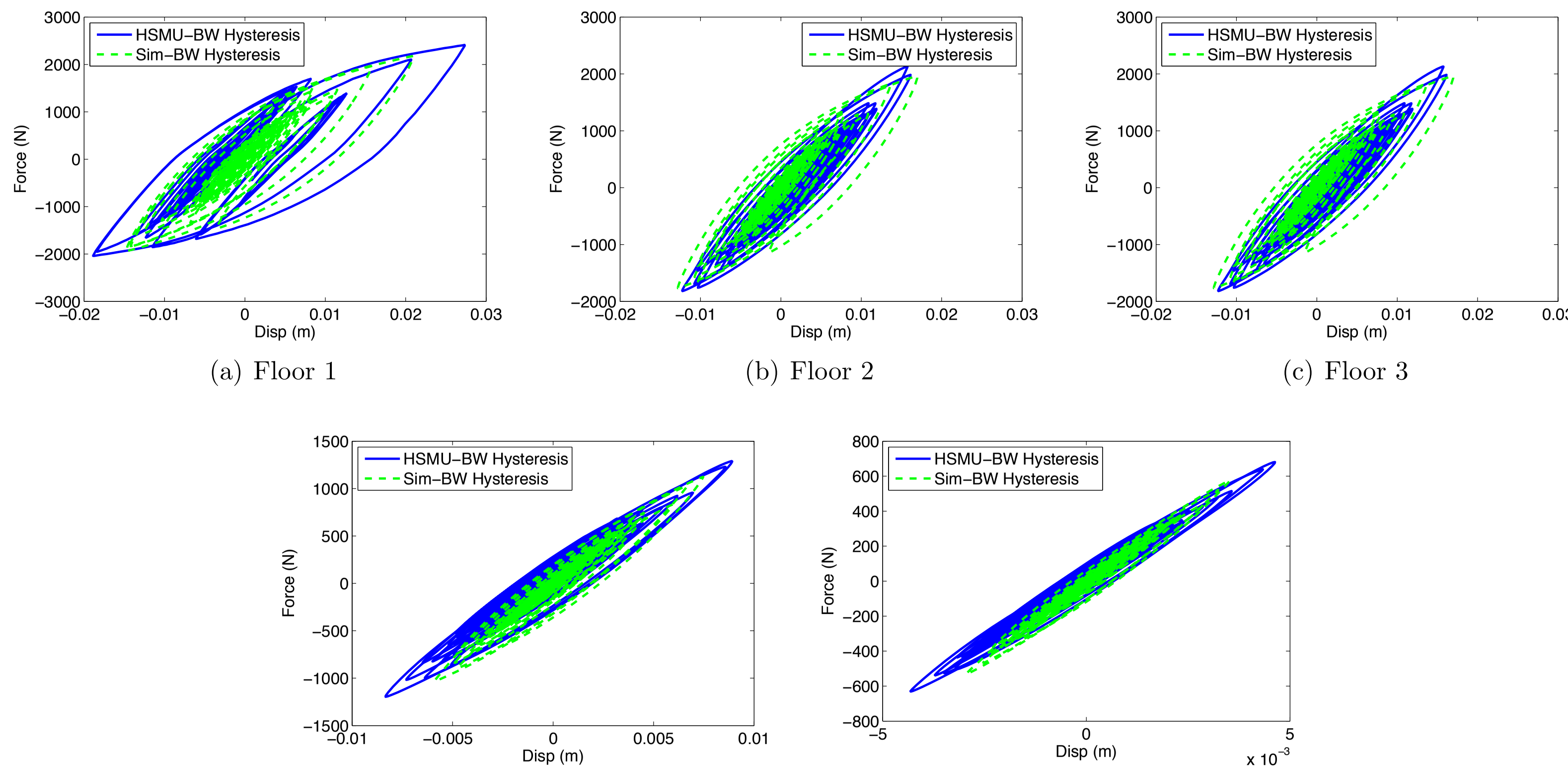


(a) Displacement Time History



(b) Acceleration Time History

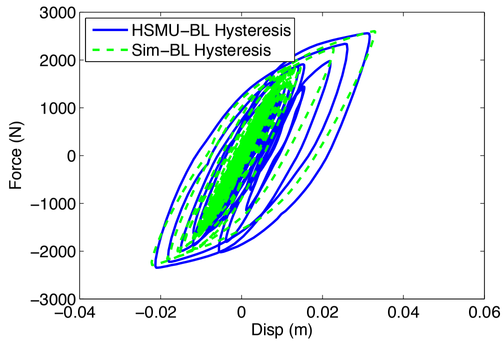
Figure 11.30.: S1: Floor 5 Time History Responses



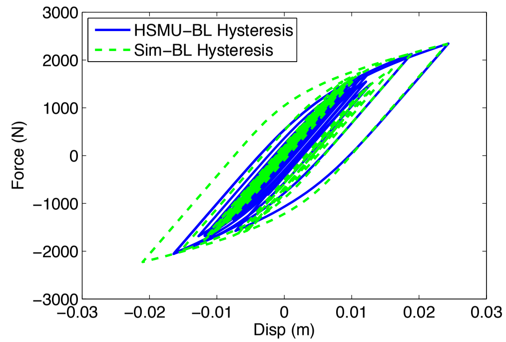
(a) Floor 1 (b) Floor 2 (c) Floor 3

(d) Floor 4 (e) Floor 5

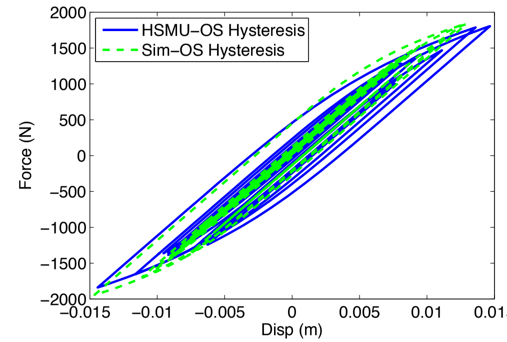
Figure 11.31.: S2: Hysteresis for Each Story - BW model



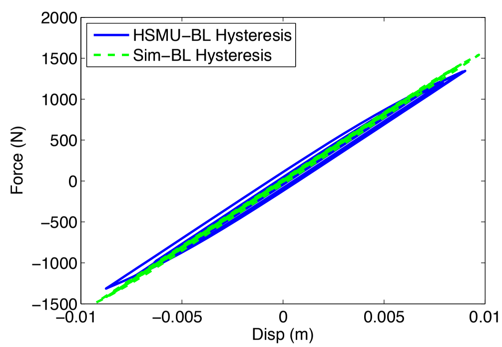
(a) Floor 1



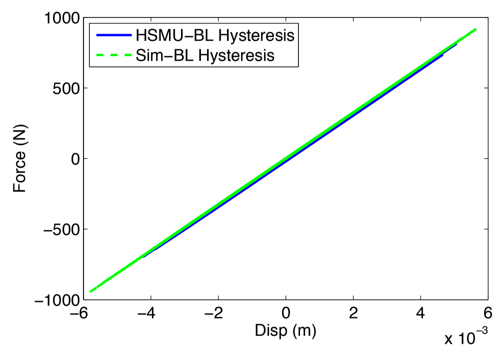
(b) Floor 2



(c) Floor 3



(d) Floor 4



(e) Floor 5

Figure 11.32.: S2: Hysteresis for Each Story - BL model

11.4 Conclusion

In this chapter, HSMU results are evaluated at the global level, where the displacement and acceleration responses from two HSMU tests are compared to a shake table test. Observations are made as follows, some observations are consistent with the conclusions from previous chapters:

- The HSMU-BW case yields the most accurate results, especially when the ground excitation is smaller, both displacement and acceleration time history match the measured shake table responses well.
- Results are improved when the phenomenological model is updated, where HSMU-BW has the smallest errors and SIM-BW case has largest errors. On the other hand, the improvement of HSMU-BL compared to SIM-BL is less significant.
- The phenomenological model can better capture the nonlinear behavior of each floor frame.
- Frequency analysis also demonstrates the effectiveness of model updating on both models. Higher modes on SIM-BW case shows larger damping for SIM-BW and less damping for SIM-BL, which are both improved with model updating, as more accurate modal frequency and damping are shown in HSMU-BW and HSMU-BL cases.
- When the structure experiences a large ground motion, a large residual drift shows in the shake table response at the first floor. Neither HSMU-BL or HSMU-BW can capture such residual drift in the responses.

The reason the large residual drift in the shake table test but not captured by hybrid simulations is not specifically investigated in this study. Some possible explanations can be 1) deficiency in the connection manufacturing; 2) a fixed-end simplification of the connection is not sufficient; 3) the inertia effect is applied numerical applied.

Additional investigation and further studies to improve the residual drift prediction using a more proper simulation model are needed.

12. CONCLUSIONS AND FUTURE WORK

12.1 Summary and Conclusions

The main objective of this dissertation is to develop a robust RTHS platform considering complex interactions between different components in the physical - computational system. Key contribution and findings in developing this platforms are:

- In RTHS, the loading servo-hydraulic system can be linearized into a fourth-order, component-based model. Genetic algorithms is used to optimize system characteristic parameters in this transfer function. The approach is found to be highly efficient and have fast convergence for this application and the results of the parametric identification is demonstrated to be effective.

- A new algorithm for actuator control is proposed. By integrating the most effective features to develop a flexible and versatile closed loop control system, the new robust integrated actuator control algorithm meets the needs of the RTHS user. The limitations of the original H_∞ design are overcome, while the robust stability is preserved. In both simulation and experimental results, the RIAC significantly reduced noise impact on the closed loop system, especially when the noise peak is in the desired control frequency range. RIAC enables the user to fully consider the system dynamics as well as the uncertainty (error or measurement noise) and still establish a design yielding highly accurate tracking.

- An inherent unit delay is identified in the force measurement of the experimental substructure in RTHS due to the sequential order of communication between the numerical and experimental substructures. This may cause instabilities or performance degradation of the test. The computation delay in RTHS changes the stability characteristics of the integration scheme. For an undamped structure, any partitioning ratio results in an unstable RTHS loop, and for lightly damped structures, the stable sampling intervals is significantly reduced to maintain closed loop stability. Also, larger partitioning ratios results in more restrictions on the selection of the sampling intervals.

- A modified Runge-Kutta integration algorithm is proposed to predict the feedback force measurement and minimize the effects of this inherent delay. The MRK integration includes three computation stages, 1) pseudo response calculation, 2) prediction of the measured force, and 3) corrected response calculation. Results illustrate that the modified Runge-Kutta improves the performance of RTHS. Further, a robustness analysis, considering modeling error in the experimental substructure, demonstrates that only under-estimation of structure stiffness (specimen stiffening) may affect MRK stability for the undamped case. For lightly damped structures with a high partitioning ratio, the MRK method is shown to be robust for up to 40% modeling error.

- An experimental test is also implemented to verify the effectiveness of the modified Runge-Kutta integration algorithm over conventional integration algorithms. A moment resisting frame with a large stiffness is tested as the experimental substructure in RTHS. Results indicated that the modified Runge-Kutta algorithm improves the accuracy of the RTHS and extends the stability limit of the test.

In addition, to improve the fidelity of the hybrid simulation where elements similar to the physical specimen are also represented in the numerical substructure, the online system identification algorithm is integrated into the hybrid simulation to update the numerical model parameters according to the physical specimen response. Investigations are performed using a model updating method to identify different structural models. The improvement in hybrid simulation fidelity is illustrated through the model updating performance as well as a global assessment by comparing the HSMU responses with the shake table test results. Some findings associated are listed as followings:

- Phenomenological models are capable of capturing the hysteretic shape of a structural component. One major drawback of these models is that the parameters do not have physical meaning, therefore, it is very difficult to start any model updating process with a reasonable initial guess. Also, knowledge of previous component models and test results cannot be transmitted to a new specimen. In addition, in Bouc-Wen models, the hysteretic shape is more sensitive to some parameters. Here it is demonstrated that this can affect model updating convergence.

- Both bilinear and the Menegotto-Pinto models are dominated by material properties such as the Young's modulus, the yield stress, and the strain-hardening ratio. There are shape parameters in the Menegotto-Pinto model to capture the Bauchinger effect of a steel component. However, the Menegotto-Pinto model is associated with loading path, and its implementation has been simplified with flag indications in computational implementation. Therefore, computational execution of the Menegotto-Pinto model is equivalent to piece-wise function, which makes it difficult to incorporate online (recursive) model updating algorithms.

- Hybrid simulation with model updating significantly improves the accuracy of hybrid simulation when the initial target model behavior differs significantly from the true response. If the target model is relatively accurate, it may not always be necessary to implement model updating. Additionally, the initial guess of the target model parameter for CUKF does not significantly affect the stability and convergence of the updating procedure, which indicates that HSMU is likely to be successful even with a large initial error in the target model.

- The accuracy of the model identification procedure depends on the richness of the information (frequency and amplitude) in the structural response. HSMU may not be applicable in all situations because reaching the global minimum requires that the system is fully excited. Also, the excitation incompleteness may cause parameter oscillation in identification and not truly converge.

- An experimental study is conducted to understand the model updating performance in HSMU. HSMU approach has been successfully implemented to both phenomenological and constitutive models with different platforms.

- Results indicate model response accuracy has been improved with model updating. Most model parameters converge fast after the peak response occurs in the time history. This convergence can be the local optimization point for a given response history and a given model. If the peak response evolves later in the time history, the optimization set can adjust to a new convergence according to the updating process. A trilinear behavior is observed from model updating for the steel bilinear model, which matches the observation in the steel material test.

- The model updating estimation accuracy is improved with the phenomenological model. Because the initial guess is more accurate and has a clear physical meaning in the bilinear model, the improvement in the HSMU-BL is less significant. Additionally, the bilinear model cannot represent the hysteresis behavior of the steel frame well and, the residual error after model updating is larger.

- Compared to the shake table test, hybrid simulation responses improved largely after the phenomenological model is updated. The improvement of HSMU-BL compared to SIM-BL is less significant. After model updating HSMU-BW case yields the most accurate results, especially when the ground excitation is smaller, both displacement and acceleration time history match the measured shake table responses well.

- Frequency analysis also demonstrates the effectiveness of model updating on both models. Higher modes on SIM-BW case shows larger damping for SIM-BW and less damping for SIM-BL, which are both improved with model updating, as more accurate modal frequency and damping are shown in HSMU-BW and HSMU-BL cases.

- When the structure experiences a large ground motion, a large residual drift shows at the first floor in the shake table test. Neither HSMU-BL or HSMU-BW can capture such residual drift in the responses.

12.2 Future Work

To broaden the applications of hybrid simulation and RTHS, continued development on the theory, methodology, and implementation of those testing techniques are needed. Based on the lesson learned and observations made in the course of preparing this dissertation, some topics for future research are recommended:

- The actuator model identification is implemented on a linearized simplification, in order to accurately predict the structure-actuator interacted behavior (known as control-structural interaction), a nonlinear model is needed which requires chamber pressure measurement. Also further study on large scale dynamic actuators is desired.

- To reduce the interaction between system noise and actuator control loop, the RIAC achieves a tradeoff between the model accuracy and system noise. To improve the accuracy of the RIAC under high noise/signal ratio, a high fidelity actuator model is desired. Model updating algorithm may also be applied to the control algorithm to form an adaptive controller. However, control stability with adaptive plant model should be carefully examined.

- The complex interaction between each component in RTHS needs to be further eliminated. Noise effect of the force transducer (structural response) to

RTHS loop should be investigated. Advanced methodology on dealing with such interaction should be established. It should be noted that any dynamic filtering technique (introducing phase lag) is not applicable because in RTHS no time delay is allowed in the feedback loop, also model predictive filter is not desired because the response of the experimental substructure is assumed to be unknown or hard to model.

- A component based platform or procedure to evaluate the feasibility of a RTHS including complex interactions are needed. Such platform needs to be compatible with different actuator control algorithms, numerical integration schemes, with considering system noise, and modeling uncertainties.

- An initial parameter set is required for implementing online identification on a given model. For phenomenological models, the initial parameters do not associate with physical properties. It is beneficial to be able to fast identify the initial parameters of phenomenological models close to true behavior without testing.

- The determination of parameters in the identification algorithm (in this case, for constrained unscented Kalman filter are Q, R, T_0, λ) based on a trial and error approach, a more systematic and versatile procedure or guideline is needed.

- New model updating algorithms and machine learning techniques are encouraged to be applied to HSMU. It is desired to have a new model updating algorithms can handle piece wise functions, so the Menegotto-Pinto model can be identified during hybrid simulation.

- Results indicate model updating performs well for small scale steel frame. Implementations of model updating on large scale specimen and composite materials should be studied.

REFERENCES

REFERENCES

[1] Ajrab, J., Pekcan, G., and Mander, J. (2004). Rocking wall frame structures with supplemental tendon systems. *Journal of Structural Engineering*, ASCE, **130**(6): 895-903.

[2] Eatherton, M.R., Hajjar, J.F., Deierlein, G.G., Krawinkler, H., Billington, S., and Ma, X., (2008). Controlled rocking of steel-framed buildings with replaceable energy-dissipating fuses. *Proceedings of the 14th World Conference on Earthquake Engineering, Beijing, China, October, 2008.*

[3] Eatherton, M.R., Ma, X., Krawinkler, H., Deierlein, G.G., and Hajjar, J.F., (2014). Quasi-static cyclic behavior of controlled rocking steel frames. *Journal of Structural Engineering*, ASCE, **140**(11): 04014083.

[4] Eatherton, M., Ma, X., Krawinkler, H., Mar, D., Billington, S., Hajjar, J., and Deierlein, G. (2014). Design concepts for controlled rocking of self-centering steel-braced frames. *Journal of Structural Engineering*, ASCE, **140**(11): 04014082.

[5] Lubell, A., Prion, H., Ventura, C., and Rezai, M. (2000). Unstiffened steel plate shear wall performance under cyclic loading. *Journal of Structural Engineering*, ASCE, **126**(4): 453-460.

[6] Zhao, Q. and Astaneh-Asl, A. (2004). Cyclic behavior of traditional and innovative composite shear walls. *Journal of Structural Engineering*, ASCE, **130**(2): 271-284.

[7] Buyukozturk, O. and Hearing, B. (1998). Failure behavior of precracked concrete beams retrofitted with FRP. *Journal of composites for construction*, **2**(3): 138-144.

[8] Shiozawa, K., Morii, Y., Nishino, S. and Lu, L. (2006). Subsurface crack initiation and propagation mechanism in high-strength steel in a very high cycle fatigue regime. *International Journal of Fatigue*, **28**(11),1521-1532.

[9] Hakuno, M., Shidawara, M., and Hara, T. (1969). Dynamic destructive test of a cantilever beam, controlled by an analog-computer. *Transactions of the Japan Society of Civil Engineers*. **171**(1-9): a1.

[10] Nakashima, M., McCormick, J., and Wang, T. (2008). Hybrid simulation: a historical perspective. *Hybrid Simulation: Theory, Implementation, and Applications*, **1**: 3-12.

[11] Takanashi, K. et al. (1975). Nonlinear earthquake response analysis of structures by a computer actuator on-line system (Part 1 details of the system).*Transactions of the Japan Society of Civil Engineers*. **229**: 77-83.

[12] Shing, P.B., Mahin, S.A. , (1983). Experimental error propogation in pseudodynamic testing. *Earthquake Engineering Research Report UCB/EERC-83/12*, University of California, Carlifornia, Berkeley, 168.

[13] Kato, H., Nakashima, M. and Kaminosono, T. (1985). Simulation accuracy of computer-actuator online testing. *Proceeding of the 7th Symposium on the Use of Computers in Building Structures, Architectural Institute of Japan*, Tokyo, Japan, 199-204 (in Japanese).

[14] Shing, P.B., Mahin, S. A. (1987). Cumulative experimental errors in pseudodynamic tests. *Earthquake Engineering and Structural Dynamics*, **15**(4): 409-424.

[15] Belytschko, T. (1976). A survey of numerical methods and computer programs for dynamic structural analysis. *Nuclear Engineering and Design*, **37**(1): 23-34.

[16] Shing, P.S.B. and Manivannan, T. (1990). On the accuracy of an implicit algorithm for pseudodynamic tests. *Earthquake Engineering and Structural Dynamics*, **19**(5):631-651.

[17] Hilber H.M., Hughes, T.J.R., and Taylor, R.L., (1977). Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and Structural Dynamics*, **5**(3): 283-292.

[18] Nakashima, M. et al. (1990). Integration techniques for substructure pseudodynamic test. *Proceedings of the 4th U.S. National conference on Earthquake Engineering*, Palm Springs, California, May, 1990, 515-524.

[19] Shing, P.B. and Vannan, M.T., (1991). Implicit algorithm for psuedodynamic tests: convergence and energy dissipation. *Earthquake Engineering and Structural Dynamics*, **20**: 809-819.

[20] Combescure, D. and Pegon, P., (1997). α-Operator splitting time integration technique for pseudodynamic testing error propagation analysis. *Soil Dynamics and Earthquake Engineering*, **16**(7): 427-443.

[21] Zhang, Y., Sause, R., Ricles, J.M. and Naito, C.J. (2005). Modified predictor-corrector numerical scheme for real-time pseudo dynamic tests using state-space formulation, *Earthquake engineering and structural dynamics*, **34**(3): 271-288.

[22] Lamarche, C.P., Bonelli, A., Bursi, O.S. and Tremblay, R. (2009). A Rosenbrock-W method for real-time dynamic substructuring and pseudo - dynamic testing. *Earthquake Engineering and Structural Dynamics*, **38**(9): 1071.

[23] Balendra, T., Lam, K.Y., Liaw, C.Y. and Lee, S.L. (1987). Behavior of eccentrically braced frame by pseudo-dynamic test. *Journal of Structural Engineering*, **113**(4): 673-688.

[24] Mutsuyoshi, H., Machida, A., Tanzo, W. and Mashiko, N. (1994). Inelastic seismic response behaviour of RC bridge pier using pseudo - dynamic test method. *Transactions of the Japan Concrete Institute*, **16**(1994): 265-272.

[25] Shing, P.B., Bursi, O.S. and Vannan, M.T., (1994). Pseudodynamic tests of a concentrically braced frame using substructuring techniques. *Journal of Constructional Steel Research*, **29**(1-3): 121-148.

[26] Molina, F.J., Verzeletti, G., Magonette, G., Buchet, P.H. and Geradin, M., (1999). Bidirectional pseudodynamic test of a full - size three-storey building. *Earthquake engineering and structural dynamics*, **28**(12): 1541-1566.

[27] Tsai, K.C., Hsiao, P.C., Wang, K.J., Weng, Y.T., Lin, M.L., Lin, K.C., Chen, C.H., Lai, J.W. and Lin, S.L., (2008). Pseudo-dynamic tests of a full-scale CFT/BRB frame-Part I: Specimen design, experiment and analysis. *Earthquake Engineering and Structural Dynamics*, **37**(7): 1081-1098.

[28] Stavridis, A. and Shing, P.B., (2010). Hybrid testing and modeling of a suspended zipper steel frame. *Earthquake Engineering and Structural Dynamics*, **39**(2): 187.

[29] Horiuchi, T. and Nakagawa, M. , (1996). Real-time hybrid experimental system with actuator delay compensation. *Proceedings of the 11th World Conference on Earthquake Engineering*, Acapulco, Mexico, June, 1996, Paper No. 660.

[30] Horiuchi, T. and Nakagawa, M. , (2001). A new method for compensating actuator delay in real-time hybrid experiments. *Proceedings of the Royal Society A*, **359**(1786): 1893-1909.

[31] Wallace,M.I. , Wagg, D.J. and Neild, S.A. (2005). An adaptive polynomial based forward prediction algorithm for multi-actuator real-time dynamic substructuring. *Proceedings of the Royal Society A*, **461**(2064), 3807-3826.

[32] Darby, A. P. , Williams, M. S. , and Blakeborough, (2002). Stability and Delay Compensation for Real-Time Substructure Testing. *Journal of Engineering Mechanics*, **128**(12): 1276-1284.

[33] Ahmadizadeh, A. , Mosqueda, G. , and Reinhorn, A.M. , (2008). Compensation of Actuator Dynamics for Real-time Hybrid Simulation. *Earthquake Engineering and Structural Dynamics*, **37**(1): 21-42.

[34] Nguyen, V.T. , and Dorka, U.E. , (2008). Phase lag compensation in real-time substructure testing based on online system identification. *Proceedings of the 14th World Conference of Earthquake Engineering*, Beijing, China, October, 2008.

[35] Chen, C. and Ricles, J. M. , (2009). Analysis of actuator delay compensation method for realtime testing. *Engineering Structures*, **31**(11): 2643-2655.

[36] Chen, C. and Ricles, J. M. , (2009). Improving the inverse compensation method for real-time hybrid simulation through a dual compensation scheme. *Earthquake Engineering and Structural Dynamics*, **38**(10): 1237-1255.

[37] Wu, B. , Wang, Z. , and Bursi, O.S. , (2013). Actuator dynamics compensation based on upper bound delay for realtime hybrid simulation. *Journal of Earthquake Engineering and Structural Dynamics*, **42**(12): 17491765.

[38] Carrion, J.E., and Spencer, B. F., (2006). Real-Time Hybrid Testing Using Model-Based Delay Compensation. *Proceedings of the 4th International Conference on Earthquake Engineering*, Taipei, Taiwan, October, 2006, Paper No. 299.

[39] Carrion, J.E. and Spencer, B. F., (2007). Model-based strategies for real-time hybrid testing. *Newmark Structural Engineering Laboratory Report Series*, University of Illinois at Urbana-Champaign, Urbana, IL, No. 6.

[40] Phillips, B. M., and Spencer, B. F., (2011). Model-Based Feedforward-Feedback Tracking Control for Real-Time Hybrid Simulation. *Newmark Structural Engineering Laboratory Report Series*, University of Illinois at Urbana-Champaign, Urbana, IL, No. 28.

[41] Phillips, B. M., and Spencer, B. F., (2013). Model-Based Feedforward-Feedback Actuator Control for Real-Time Hybrid Simulation. *Journal of Structural Engineering.* **139**(7): 1205-1214.

[42] Gao, X. , Castaneda, N. E. , and Dyke, S.J , (2011). Development and Validation of a Robust Actuator Motion Controller for Real-time Hybrid Testing Applications. *Intelligent Infrastructure System Laboratory Report Series*, Purdue University, West Lafayette, IN, No.1.

[43] Gao, X., (2012). Development of a Robust Framework for Real-Time Hybrid Simulation: From Dynamical System, Motion Control to Experimental Error Verification. *Doctoral Dissertation*, Purdue University, West Lafayette, IN.

[44] Gao, X. , Castaneda, N. E. , and Dyke, S.J., (2012). Real time hybrid simulation: from dynamic system, motion control to experimental error. *Earthquake Engineering and Structural Dynamics*, **42**(6), 815-832.

[45] Wu, B., Bao H., Ou, J., and Tian, S., (2005). Stability and accuracy analysis of the central difference method for real-time substructure testing. *Earthquake Engineering and Structural Dynamics*; **34**(7): 705-718.

[46] Wu, B., Xu, G., Wang, Q., and Williams, M. S., (2006). Operator-splitting method for real-time substructure testing. *Earthquake Engineering and Structural Dynamics*; **35**(3): 293-314.

[47] Chen, C., Ricles J. M., Marullo, T. M., and Mercan, O., (2009). Real-time hybrid testing using the unconditional stable explicit CR integration algorithm. *Earthquake Engineering and Structural Dynamics.* **38**(1): 23-44.

[48] Wu, B., Wang, Z., and Bursi, O. S., (2012). Nearly-perfect compensation for time delay in real-time hybrid simulation, *Proceedings of: 15WCEE*, Instituto Superior Tecnico, Lisbon, Portugal, September, 2012.

[49] Wu, B., Wang, Q., Shing, P.B., Ou, J., (2007). Equivalent force control method for generalized real-time substructure testing with implicit integration. *Earthquake Engineering and Structural Dynamics*; **36**(9):11271149.

[50] Shing, P. B., Spacone, E., Stauffer, E., (2002). Conceptual design of fast hybrid test system at the University of Colorado. *Proc. the 7th U.S. National Conf. on EQ Engineering*, Boston, Boston, Massachusetts, July, 2002.

[51] Chen, C., Ricles J. M., (2011). Analysis of implicit HHT-α integration algorithm for real-time hybrid simulation, *Earthquake Engineering and Structural Dynamics*, **41**(5): 1021-1041.

[52] Mosqueda, G., Stojadinovic, B., and Mahin, S., (2005). Implementation and accuracy of continuous hybrid simulation with geographically distributed substructures. *Technical Report, Earthquake Engineering Research Center*, University of California, Berkeley, November, 2005.

[53] Maghareh, A., Waldbjrn, J.P., Dyke, S.J., Prakash, A. and Ozdagli, A.I., (2016). Adaptive multi-rate interface: development and experimental verification for real-time hybrid simulation. *Earthquake Engineering and Structural Dynamics*.

[54] Gomez, D., Dyke,S. J., and Maghareh, A. (2015). Enabling role of hybrid simulation across NEES in advancing earthquake engineering, *Smart Structures and Systems*, **15**(3): 913-929.

[55] Dowden, D.M. and Bruneau, M., (2011). NewZ-BREAKSS: Post-tensioned rocking connection detail free of beam growth. *Engineering Journal(Chicago)*, **48**(2): 153-158.

[56] Kwon, O.S. and Kammula, V., (2013). Model updating method for substructure pseudo-dynamic hybrid simulation. *Earthquake Engineering and Structural Dynamics*, **42**(13): 1971-1984.

[57] Wu, M., and Smyth, A., (2008). Real-time parameter estimation for degrading and pinching hysteretic models. *International Journal of Non-Linear Mechanics*, **43**(9):822-833.

[58] Hashemi, M.J., Masroor,A., and Mosqueda, G., (2014). Implementation of online model updating in hybrid simulation. *Earthquake Engineering and Structural Dynamics*, **43**(3), 395-412.

[59] Shao, X., Mueller, A., and Mohammed, B. (2015). Real-time hybrid simulation with online model updating: methodology and implementation. *Journal of Engineering Mechanics*, 10.1061/(ASCE)EM.1943-7889.0000987, 04015074.

[60] Wu, B., and Wang, T., (2015). Model updating with constrained unscented Kalman filter for hybrid testing. *Smart Structure and Systems*, **14**(6): 1105-1129.

[61] Elnashai, A. S., V. Papanikolaou, and D. H. Lee., (2001). ZeusNL-A program for inelastic dynamic analysis of structures. *MAE Center, University of Illinois at Urbana-Champaign*, USA.

[62] Hazem, E.H., and Elnashai, A. S., (2014). On-line model updating in hybrid simulation tests, *Journal of Earthquake Engineering*, **18**(3):350-363, DOI: 10.1080/13632469.2013.873375.

[63] Hazem, E.H., and Elnashai, A. S., (2015). Framework for online model updating in earthquake hybrid simulations. *Journal of Earthquake Engineering*, DOI: 10.1080/13632469.2015.1051637.

[64] Nakata, N., Dyke S. J., Zhang, J., Mosqueda, G., Shao, X., Mahmoud, H., Head, M. H. , Blezinger, M. E., Marshall, G. A., Ou, G., (2014). Hybrid simulation primer and dictionary, *Hybrid Simulation Task Force meeting Report Series, Network of Earthquake Engineering Simulation*, Technical Report 001.

[65] Nakata, N., Spencer, B.F., and Elnashai, A. S., (2006). Mixed load/displacement control strategy for hybrid simulation. *4th International Conference on Earthquake Engineering*, Taipei, Taiwan, October, 2006.

[66] Spencer, B.F., Elnashai, A. S., Kwon, O., Park, K., and Nakata, N., (2007). The UI-SimCor hybrid simulation framework. *2nd World Forum on Collaborative Research in Earthquake Engineering*, Ispra, Italy.

[67] Schellenberg A., Kim H. K., Takahashi Y., Fenves G. L., and Mahin S. A., (2009). OpenFresco command language manual. Version: 2.6. University of California, Berkeley.

[68] OpenSees, (2009). Open system for earthquake engineering simulation. *http://opensees.berkeley.edu*, University of California, Berkeley, CA

[69] Yang, G., Wang, Z., and Wu, B., (2015). Development of HyTest for structural hybrid simulation. *Journal of Building Structures*, **36**(11): 149-156, (in Chinese).

[70] Jelali, M., Kroll, A., (2003). Hydraulic servo-system modeling, identification and control. *London: Springer Science & Business Media*.

[71] Zhao, J., (2003). Development of EFT for nonlinear SDOF systems. *Doctoral dissertation*, University of Minnesota.

[72] Carrion, J. E., Spencer, B. F., and Phillips, B. M., (2009). Real-Time hybrid simulation for structural control performance assessment. *Earthquake Engineering and Engineering Vibration*, **8**(4): 481-492.

[73] Nakata, N. (2012). Effective force testing using a robust loop shaping controller. *Earthquake Engineering and Structural Dynamics*, **42**(2): 261-275.

[74] Dyke, S. J., Spencer, B. F., Quast, P., and Sain, M., (1995). Role of control-structure interaction in protective system design. *Journal of Engineering Mechanics*, **121**(2): 322-338.

[75] Kristinsson, K. and Dumont, G., (1992). System identification and control using genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, **22**(5): 1033-1046..

[76] Yousefi, H., Handroos, H., and Soleymani, A. (2008). Application of differential evolution in system identification of a servo-hydraulic system with a flexible load. *Mechatronics* **18**(9): 513-528.

[77] Sarmady, S., (2007). An investigation on genetic algorithm parameters. *School of Computer Science, Universiti Sains Malaysia*, 2007.

[78] Kim, S. and Murrenhoff, H., (2012). Measurement of effective bulk modulus for hydraulic oil at low pressure. *Journal of Fluid Engineering*, **134**(2): 021201.

[79] Rathje, E. M., Abrahamson, N. A., and Bray, J. D., (1998). Simplified frequency content estimates of earthquake ground motions. *Journal of Geotechnical and Geoenvironmental Engineering*, **124**(2), 150-159..

[80] Nakashima, M. and Masaoka, N., (1999). Real-time online test for MDOF systems. *Earthquake Engineering and Structural Dynamics*, **28**(4): 393-420.

[81] Glover, K. and McFarlane, D., (1989). Robust stabilization of normalized coprime factor plant descriptions with H_∞ uncertainty, *IEEE Transactions on Automatic Control* **34**(8): 821-830.

[82] Ou, G., Ozdagli, A. I., Dyke, S. J., Wu, B., (2015). Robust integrated actuator control: experimental verification and real time hybrid simulation implementation. *Earthquake Engineering and Structural Dynamics*, **44**(3): 441460.

[83] Mahin, S.A. and Shing, P.B., (1985). Pseudodynamic method for seismic testing. *Journal of Structural Engineering*, ASCE, **111**(7): 1482-1503.

[84] Takanashi, K. and Nakashima, M., (1987). Japanese activities on on-line testing. *Journal of Engineering Mechanics*, ASCE, **113**(7): 1014-1032.

[85] Christenson, R.E., Lin, Y.Z, Emmons, A.T., and Bass, B. (2008). Large-scale experimental verification of semiactive control through real-time hybrid simulation. *Journal of Structural Engineering*, ASCE, **134**(4): 522-535.

[86] Nakashima, M. . (1995). Development, potential, and limitation of real-time on-line (pseudo-dynamic) testing. *Mathematical, Physical and Engineering Sciences*, **359**(1786): 1851-1867.

[87] Horiuchi, T. , and Nakagawa, M. , (1999). real time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber. *Earthquake Engineering and Structural Dynamics*, **28**(10): 1121-1141.

[88] Ning, C. and Zhang X. (2012). Study on vibration and noise for the hydraulic system of hydraulic hoist, *Proceedings of 2012 International Conference on Mechanical Engineering and Material Science*, Yangzhou, China, December, 2012.

[89] Klipec, B.E. (1967) Reducing electrical noise in instrument circuits. *IEEE Transactions on Industry and General Applications*, **IGA-2**(2): 90-96.

[90] Gao, X. and Dyke, S.J. (2014). Modeling and control of actuators for high performance structural dynamic testing. *Smart Materials and Structures*, **23**(5): 055008.

[91] Welch, G., and Bishop, G. (2006). An introduction to the Kalman filter. Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC. Available from http://www.cs.unc.edu/welch/kalman/kalmanIntro.html

[92] Maghareh A, Dyke S.J. (2013). Stability and performance analysis of SDOF real-time hybrid simulation, Technical Report 004, *IISL Technical Report Series* Intelligent Infrastructure Systems Lab, Purdue University.

[93] Christenson R, Botelho R, and Franco J. (2013). Exact stability analysis for uniaxial real-time hybrid simulation of 1-DOF and 2-DOF structural systems. *Engineering Mechanics Institute Conference*, Northwestern University, August, 2013;

[94] Kang, D., Haussmann, G., and Saouma V.E., (2012). Mercury optimized software for (single site) hybrid simulation from pseudo dynamics to real time hybrid simulation: theory manual ver.1. Department of Civil Environmental and Architectural Engineering, University of Colorado, Boulder, Colorado.

[95] Kang, D.H., (2010). An optimized computational environment for real-time hybrid simulation. *Doctoral Dissertation*, University of Colorado at Boulder.

[96] Ohtori, Y., Christenson, R., Spencer, B., Jr., and Dyke, S., (2004). Benchmark control problems for seismically excited nonlinear buildings. *Journal of Engineering Mechanics*, **4**(366), 366-385.

[97] Maghareh A. , Barnes M.J , and Sun Z., (2012), Evaluation of the 9-Story benchmark building shear model. https://nees.org/resources/4103.

[98] Bouc, R., (1971). Mathematical model for hysteresis. *Report to the Centre de Recherches Physiques*, Marseille, France, 1625.

[99] Wen, Y.K., (1976). Method of random vibration of hysteretic systems. *Journal of Engineering Mechanics Division*, ASCE, **102**(EM2):249263.

[100] Baber, T., and Noori, M.N., (1985). Random vibration of degrading, pinching systems. *Journal of Engineering Mechanics*, ASCE, **111**(8):10101026.

[101] Casciati, F., (1989). Stochastic dynamics of hysteretic media. *Structural Safety*, **6**(2-4): 259269.

[102] Reinhorn, A.M., Madan, A., Valles, R.E., Reichmann, Y., and Mander, J.B., (1995). Modeling of masonry infill panels for structural analysis. *Tech. Rep. NCEER-95-0018*, State University of New York at Buffalo, Buffalo, N.Y.

[103] Ma, F., Zhang, H., Bockstedte, A., Foliente, G.C., and Paevere, P., (2004). Parameter analysis of the differential model of hysteresis. *Journal of Applied Mechanics*, **71**(3):342-349.

[104] Song, J., and Kiureghian, A., (2006). Generalized Bouc-Wen model for highly asymmetric hysteresis. *Journal of Engineering Mechanics*, **132**:610-618.

[105] Baber, T., and Wen, Y.K., (1981). Random vibration of hysteretic degrading systems. *Journal of Engineering Mechanics*, ASCE, **107**(EM6):1069-1089.

[106] Menegotto, M., and Pinto, P.E. (1973). Method of analysis of cyclically loaded RC plane frames including changes in geometry and non-elastic behavior of elements under normal force and bending. *Proc. of IABSE symposium on resistance and ultimate deformability of structures acted on by well defined repeated loads*, 15-22, 1973

[107] Filippou F.C., Popov, E.P., and Bertero, V.V. (1983). Effects of bond deterioration on hysteretic behavior of reinforced concrete joints, *Report EERC 83-19, Earthquake Engineering Research Center*, University of California, Berkeley.

[108] Wang, T., Wu, B., (2013). Real-time hybrid testing with constrained unscented kalman filter. *5th International Conference on Advances in Experimental structural Engineering*, Taipei, Taiwan, November, 2013.

[109] Wu, B. and Wang, T., (2014). Model updating with constrained unscented Kalman filter for hybrid testing. *Smart Structures and Systems*, **14**(6): 1105-1129.

[110] Muller, A., (2014). Real-time hybrid simulation with online model updating, *Master Thesis*, Western Michigan University, Paper 506.

[111] Eatherton, M., Hajjar, J.F., Ma, X., Pena, A., Deierlein, G.G., Krawinkler, H., and Billington, S., (2008). Controlled rocking of steel framed buildings as a sustainable new technology for seismic resistance in buildings. *Creating and Renewing Urban Structures*, Chicago, Illinois, September, 2008.

[112] Deierlein, G., Billington, S., Hajjar, J., (2005-2010). Controlled rocking of steel-framed buildings. *Network of Earthquake Engineering Simulation*, Project 75 (https://nees.org/warehouse/project/75).

[113] Deierlein, G., Krawinkler, H., Ma, X., Eatherton, M., Hajjar, J., Takeuchi, T., Kasai, K., and Midorikawa, M., (2011). Earthquake resilient steel braced frames with controlled rocking and energy dissipating fuses. *Steel Construction*, **4**(3):171175.

[114] Kalman, R.E., (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, ASME, **82**(1):35-45.

[115] Rhudy, M., and Gu, Y., (2013). Understanding nonlinear kalman filters Part I: selection of EKF or UKF. *Interactive Robotics Letters*, West Virginia University, June, 2013.

[116] Kalman, R.E., and Bucy, R.S., (1961). New results in linear filtering and prediction theory. *Journal of Basic Engineering*, ASME, **83**(1961):95-108.

[117] Julier, S.J., and Uhlmann, J.K., (1997). A new extension of the Kalman filter to nonlinear systems. *InAeroSense'97*, 182-193, International Society for Optics and Photonics.

[118] Rhudy, M., and Gu, Y., (2013). Understanding nonlinear kalman filters Part II: an implementation guide. *Interactive Robotics Letters*, West Virginia University, June, 2013.

[119] Crassidis, J.L., (2006). Sigma-point Kalman filtering for integrated GPS and inertial navigation. *IEEE Transactions on Aerospace and Electronic Systems*, **42**(2):750-756.

[120] Fiorenzani, T., Manes, C., Oriolo, G., and Peliti, P., (2008). Comparative study of unscented kalman filter and extended kalman filter for position/attitude estimation in unmanned aerial vehicles. *Inst. for Systems Analysis and Computer Science (IASI-CNR)*, Rome, Italy, August, 2008.

[121] Wendel, J., Metzger, J., Moenikes, R., Maier, A., and Trommer, G.F., (2006). A performance comparison of tightly coupled GPS/INS navigation systems based on extended and sigma point Kalman filters. *Navigation*, **53**(1):21-31.

[122] Song, W., and Dyke, S.J., (2013). Real-Time Dynamic Model Updating of a Hysteretic Structural System. *Journal of Structural Engineering*, 140(3), 04013082.

[123] Simon, D.J., and Simon, D.L., (2010). Constrained Kalman filtering via density function truncation for turbofan engine health estimation. *International Journal of Systems Science*, **41**(2): 159-171.

[124] Ren, H., Chen, S., Liu, G., and Zheng, K., (2014). Vehicle state information estimation with the unscented Kalman filter. *Advances in Mechanical Engineering*, **6**:589397.

[125] Sobol, I.M., (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, **55**(1):271280.

[126] Sobol, I.M., (1969). Multidimensional quadrature formulas and Haar functions. *Izdat "Nauka"*, Moscow, in Russian.

[127] Astrom, K.J., and Wittenmark, B., (2008). Adaptive Control. *Courier Corporation.*

[128] Ikhouane, F. and Rodellar, J., (2007). Systems with hysteresis: analysis, identification and control using the Bouc-Wen model. *John Wiley & Sons.*

[129] Ismail, M., Ikhouane, F., and Rodellar, J., (2009). The hysteresis BoucWen model, a survey. *Archives of Computational Methods in Engineering*, **16**(2):161188.

VITA

VITA

Ge (Gaby) Ou was born in Changsha, China on March 13th, 1988. Before she joined the Ph.D program in Purdue University (West Lafayette, IN), she received both her B.E. in Civil Engineering from University of Sydney, Australia and B.S. in Theoretical and Applied Mechanics from Harbin Institute of Technology, China, in 2010.