REAL-TIME SEISMIC HYBRID SIMULATION PROCEDURES FOR RELIABLE STRUCTURAL PERFORMANCE TESTING

by

Mehdi Ahmadizadeh

September 18, 2007

A dissertation submitted to the
Faculty of the Graduate School of
the State University of New York at Buffalo
in partial fulfillment of the requirements for the
degree of
Doctor of Philosophy

Department of Civil, Structural and Environmental Engineering

ACKNOWLEDGEMENT

The writing of this dissertation has been one of the most challenging endeavors of my academic career. Without the help, guidance and encouragement of my friends, colleagues and professors the achievements of the work presented herein would not have been possible. Accordingly, I would like to take this opportunity to express my gratitude towards them.

I feel blessed to have had the privilege to work under the supervision of such a knowledgeable and respected dissertation committee: Dr. Gilberto Mosqueda, Professor Andrei Reinhorn, and Professor Andrew Whittaker. Their continuous support motivated me to work diligently on the research presented in this dissertation. I owe the deepest gratitude to my advisor, Dr. Gilberto Mosqueda, whose knowledge, guidance and support have been essential to the completion of this dissertation. He graciously provided me with ideas and critically assisted me in developing my own. His contribution to my Ph.D. studies was not limited to my research; he taught me how a person can be a mentor, a colleague and a friend at the same time. My sincere appreciation also goes to the SEESL Director, Professor Andrei Reinhorn, who was very supportive of my work. He provided me the opportunity to attend and participate in his insightful research group meetings, and was extremely helpful in improving my presentation

skills. I am also profoundly grateful of Professor Andrew Whittaker, for his valuable comments and ideas, and for reviewing my dissertation.

I also owe thanks to many of my other professors, especially the faculty and staff of the Department of Civil, Structural and Environmental Engineering at the University at Buffalo. I am especially grateful of Professor Tsu-Teh Soong's mentoring, who is truly a knowledgeable, dedicated and thoughtful professor. Other professors who I am honored to have been instructed by include Professor Gary Dargush, Professor Andre Filiatrault and Professor Tarunraj Singh. I also appreciate the support and guidance of Professor Cemal Basaran during my first year of study at this school. Lastly, I would like to thank the Department Chair, Professor Scott Weber, for his support, and the Department Secretary, Mrs. Kirsten Brown, for all of her assistance.

I gratefully acknowledge the assistance of SEESL staff, Mark Pitman, Scot Weinreber, Robert Staniszewski, Duane Kozlowski, and David Brugger, who readily helped me in my experimental efforts. I also appreciate the help of Nick Mangkalakiri from UC Irvine in the preparation of the SEESL hybrid simulation experimental setup. I would like to thank all of my colleagues and friends at the University at Buffalo, for their insight and enjoyable fellowship. I am particularly thankful to my closest friends: Rodrigo Retamales, Gian Paolo Cimellaro, Oren Lavan, David Keller, Charles Ekiert and Mohammad Abdulhamid. I also acknowledge the contributions of my best friends to my academic career: Mohammad Reza Jahanshahi, Zahra Tehrani, Amin Haghnegahdar, Ahmad Seyedabbasi, Arash Noshadravan and Reza Jafarkhani. Finally, I would like to express my sincere gratitude to my parents and my family, who wholeheartedly supported me throughout my education and stood beside me through all challenges of my life.

TABLE OF CONTENTS

ABSTRACT

The increased need for experimental verification of the seismic performance of conventional and novel structural systems has resulted in highly sophisticated dynamic test procedures. Hybrid simulation, including pseudo-dynamic testing of experimental substructures, offers an efficient method for assessment of dynamic and rate-dependent behavior of large-scale structural systems subjected to earthquake excitation. Compared to earthquake simulations using shake tables, hybrid simulation may have significant advantages in terms of cost, scale, geometry, and required physical mass of structures and components that can be tested. However, recent hybrid simulations have been limited to simplified structural models with only a few degrees of freedom. This is primarily due to the fact that hybrid simulation is a relatively new test method that is still being improved through research. Currently, the major challenges for using hybrid simulation in large and complex structural systems are the lack of robust simulation algorithms, and the sensitivity of the results to experimental errors in the presence of high-frequency modes. The main motivation for this research is to develop reliable test procedures that can be easily applied to fast and real-time hybrid simulations of large

and complex structural systems. It is also attempted to develop test procedures that are effective for geographically distributed hybrid simulations.

In this dissertation, recent developments to improve the accuracy and stability of hybrid simulation are described using the state-of-the-art pseudo-dynamic hybrid simulation system at the Structural Engineering and Earthquake Simulation Laboratory, University at Buffalo. In particular, delay compensation procedures are examined, and new methods are proposed. These methods are based on the correction of tracking errors in force measurement signal, and using the numerical integration procedure for prediction and compensation of command displacement signal. A new online procedure is proposed for estimation of delay during the simulation, and is shown to have better performance compared to existing online delay estimation methods. Furthermore, two numerical integration procedures are introduced for hybrid simulation, which are shown to improve the stability and accuracy properties of the simulation. The proposed integration algorithms use experimental measurements to iterate within implicit scheme and also take advantage of a new approach to estimate the tangent stiffness matrix of experimental substructures. For assessment of the reliability of hybrid simulation results, energy-based error monitors are proposed to examine the severity of experimental and numerical errors. These measures are then used to demonstrate the improved accuracy offered by new algorithms proposed here through analytical and numerical studies, and numerical and experimental simulations.

1 INTRODUCTION

The increasing need for identification of seismic performance of novel and existing structural systems has resulted in highly sophisticated dynamic test procedures. Further, advanced design methods, such as performance-based design require a better understanding of the behavior of structures well into their nonlinear response range. As a result, various forms of hybrid simulation [1-5] and effective force testing methods [6-8] have been of special interest in recent years, due to their appealing features for large-scale dynamic testing of nonlinear and complex structural systems.

Hybrid simulation, including real-time dynamic testing of substructures, is an efficient method for assessment of the dynamic and rate-dependent behavior of structural systems subjected to earthquake excitation. The method separates a structure into physical (experimental) and numerical substructures, only requiring the experimental simulation of parts of the structure that are difficult to model (Figure 1-1). By utilizing an incremental time-stepping solution technique and communication of interface forces and displacements, a parallel simulation can be carried out, which takes

advantage of numerical simulation for the well-identified parts of the structure, and experimental evaluation of complicated and nonlinear parts.

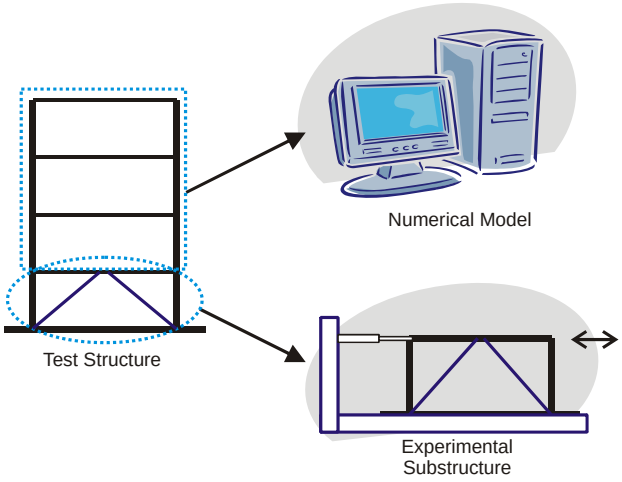

Figure 1-1 Illustration of hybrid structural simulation.

A real-time hybrid test may have significant advantages over an earthquake simulation test using a shake table in terms of the size, geometry, and required physical mass of structures that can be tested [4]. Substructure testing can also result in a better understanding of component behavior while interacting with the entire system, contrary to shaking table tests that provide information about the overall behavior of the test structure. This provides the opportunity to closely study the experimental behavior of structural components as well as the system-level response, and examine their effectiveness in achieving a better overall structural performance.

Performing an accurate and reliable hybrid simulation can be a challenging task. For example, setting up realistic boundary conditions for experimental substructures may be difficult. Similar to shaking table tests, the gravity loads may not be realistic when the experimental masses are small, and no replacement gravity forces are applied to the test

specimens. Furthermore, hybrid simulation procedures are sensitive to experimental and numerical errors, and current testing algorithms need to be improved to reliably test large-scale structures with many degrees of freedom. It is the primary goal of this dissertation to develop enhanced test procedures applicable to complex structural systems that are effective for fast and real-time hybrid simulations. The effectiveness of these procedures for geographically distributed hybrid simulations is also considered in their development.

1.1 PSEUDO-DYNAMIC HYBRID SIMULATION

In a hybrid simulation, the equation of motion of the combined numerical and experimental structure model can be expressed as:

$$\mathbf{Ma} + \mathbf{Cv} + \mathbf{Kd} + \mathbf{r} = -\mathbf{M}^{t}\boldsymbol{\iota}\ddot{u}_{g} \tag{1.1}$$

in which $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are mass, damping, and stiffness matrices of the numerical substructure, $\mathbf{M}^{t}$ is the total mass matrix of the structural model (including experimental mass), $\boldsymbol{\iota}$ is the influence vector, $\mathbf{d}$, $\mathbf{v}$, and $\mathbf{a}$ are displacement, velocity and acceleration vectors, respectively; $\ddot{u}_{g}$ is the input ground acceleration and $\mathbf{r}$ is the restoring force measured in the experimental substructures. The experimental restoring force vector may include strain-dependent, damping, or inertial forces, depending on the experiment rate and dynamic characteristics of the experimental substructure.

This dissertation mainly focuses on displacement-controlled hybrid simulation. In such experiments, the displacements computed by the numerical model are applied to the physical specimen, and the resisting force is measured and fed back into the

numerical model, as shown in Figure 1-2. This figure also shows that the displacements are imposed using servo-hydraulic actuators. In order to reduce the cost of the experiments in terms of the required mass and hydraulic power demand, and to improve the dynamic performance of servo-hydraulic actuators, the majority of mass is often considered in the numerical model. That is, the dynamic forces are created through a "virtual" mass. For this reason, this test method is sometimes called pseudo-dynamic hybrid simulation [9, 10], which may be performed at slow (static) or fast and real-time rates.

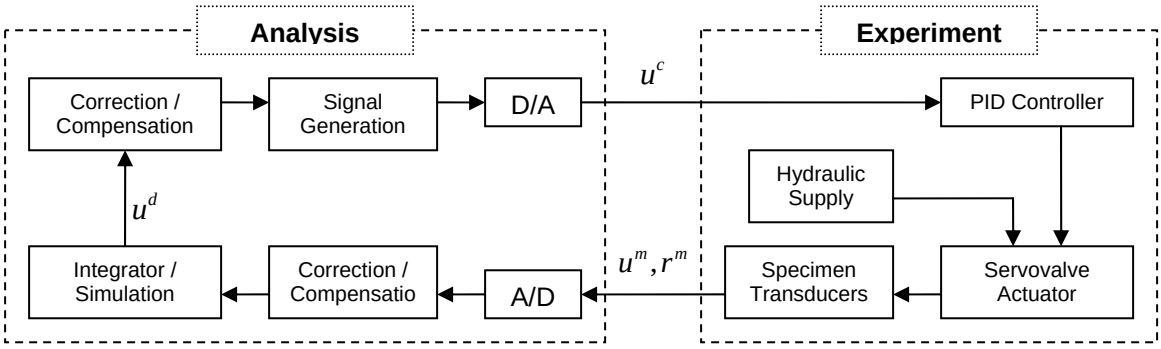

Figure 1-2 Block diagram of a typical displacement-controlled hybrid simulation.

In hybrid simulations, the experimental substructures may also be distributed in several laboratories in order to take advantage of their equipment for testing large structural systems. In this case, this test method is called geographically-distributed hybrid simulation [11]. Effective force testing is another method of testing structural components, in which actuators are used to apply the inertial forces to the experimental substructure [6-8]. A combination of force-controlled excitation of masses using actuators and base excitation using shake tables can also be used for substructure testing

[5, 9]. This constitutes a general test method that can be reduced to the above-mentioned test procedures by manipulating the roles of actuators and shaking tables.

Many of the components of a pseudo-dynamic hybrid simulation shown in the block diagram of Figure 1-2 are studied in this dissertation. As shown in this diagram, each step of a pseudo-dynamic experiment starts with the calculation of the desired displacement in the numerical simulation (integration) block. The numerical integration procedures are studied in Chapters 7 and 8. The desired displacement issued by the integration module is fed into a correction or compensation block (Chapters 5 and 6). This block may modify the desired displacement based on the expected dynamics of the test setup, in order to improve the agreement between the desired and achieved (measured) displacement signals. The output of this correction block is called compensated displacement, which is then used for generation of actuator command signal. The signal generation block re-samples the signal at the command update rate for the experimental setup, and its output is named command displacement signal. Only if the numerical integration and experiment command update rates are the same, the compensated and command displacement signals are identical. After conversion to analog form, the command displacement is fed into a controller, such as a proportional-integral-derivative (PID) controller, to produce the command voltage signal for actuator servovalve (Chapter 4). The actuators then apply the command displacement to the experimental substructure with the aid of hydraulic power supplies. The restoring force and achieved displacement are then measured using transducers and load cells installed in the experimental setup, and converted to digital form for use by the numerical

simulator. Before using the measurements in the integration module, they may be modified in another correction or compensation block (Chapter 6). This compensator can correct the signals for the actuator tracking errors, i.e. the differences between desired and measured displacement signals. These differences normally exist even when the displacement signal is carefully compensated for system dynamics. Finally, the numerical simulation module updates the states and calculates the next desired displacements using the most recent measurements to continue to the next simulation step.

1.2 MOTIVATIONS AND GOALS

Hybrid simulation has significant potential for testing of large-scale structural systems. From this standpoint, using geographically distributed testing facilities has received close attention in recent years, as it provides the possibility of testing large structures with many experimental substructures – beyond what can be performed in a single structural laboratory. Performing hybrid simulation at fast and real-time rates has also been of interest, in order to capture the rate-dependent behavior of experimental components such as viscous dampers, and to prevent the stress relaxations that may occur in static tests.

In order to utilize the hybrid simulation test method at its full potential, research in algorithm development and implementation techniques is still ongoing. To this day, real-time and distributed applications of hybrid simulations are very few, and the structures that are tested are commonly small, or considerably simplified. This is

primarily due to the limited robustness of error compensation or numerical simulation procedures that are sensitive to high natural frequencies, or have limited accuracy for testing highly nonlinear systems.

A brief look at the literature on hybrid simulation technique (presented in Chapter 2) shows that in addition to the improvement of laboratory equipment and test hardware [9, 11-13], substantial enhancements of hybrid simulation procedures are necessary. Particularly, the test procedures need to be extendable to multi-degree-of-freedom (MDF) experimental substructures, or systems that contain high-frequency modes. The simulation procedures should be able to capture highly nonlinear behavior of experimental substructures with reasonable accuracy. On the other hand, such procedures should have modest processing and communication requirements for real-time or distributed applications. Simplicity of these procedures is another important factor that makes them feasible for widespread experimental applications.

As a hybrid simulation includes both numerical and experimental errors, proper identification and compensation of these errors are critical. Hence, besides the essential simulation components, reliability measures should be established to assess the accuracy and reliability of simulation results.

In addition to the development of a basic pseudo-dynamic test system at Structural Engineering and Earthquake Simulation Laboratory (SEESL) at University at Buffalo, this study is aimed towards the improvement of current procedures used in hybrid simulation to achieve the above-mentioned goals. It is attempted to develop simple, computationally effective and robust procedures that expand the capabilities of hybrid

simulation to real-time and distributed experiments of large and complex structural systems. In this dissertation, the most common procedures for identification of system dynamics, error and delay compensation methods and numerical integration algorithms are examined, and improved procedures are introduced. In addition, new and existing accuracy measures are studied to assess and demonstrate the accuracy and stability of hybrid simulations.

1.3 DISSERTATION ORGANIZATION

Following the state-of-the-art of hybrid simulation test method outlined in Chapter 2, the hybrid simulation test system at SEESL is presented in Chapter 3. The major test components and simulation subroutines are introduced, and the software environment for creating simulation models is described. The hybrid simulation test setup that has been used for verification of test procedures is also presented in Chapter 3. In addition, this chapter introduces the numerical models used for computer simulations of hybrid simulation.

Chapter 4 is dedicated to an introduction of servo-hydraulic actuators and their control systems, since they play an important role in hybrid simulations. The mechanical principles governing the functions of these actuators are presented, and their numerical models are studied. This chapter also includes an introduction to PID controllers that are widely used with servo-hydraulic actuators. In Chapter 5, experimental and numerical error sources are explored, and methods for estimation and quantification of these errors are studied. Servo-hydraulic actuator delay, which is one of the most detrimental

experimental error sources in hybrid simulation, is studied in detail in this chapter. Sources and effects of delay, as well as its measurement methods are studied, and an improved online delay estimation procedure is introduced. Next, the experimental and numerical error indicators are studied, and an overall error measure based on energy balance is proposed.

Chapter 6 presents the existing and improved delay and error compensation methods for hybrid simulation. In this chapter, force correction methods are proposed for compensation of moderate delay and tracking errors, and displacement compensation is introduced using the numerical integration equations, as a better alternative to the widely-used polynomial extrapolation approach.

Numerical integration procedures for hybrid simulation are studied in Chapters 7 and 8. Chapter 7 presents the formulation of the most common numerical integration methods. The widely-used explicit and operator-splitting methods are also presented in this chapter. Chapter 7 serves as a foundation for the improved numerical integration methods that have been developed in this study and introduced in Chapter 8. The stability and accuracy properties of the numerical integration methods presented in Chapters 7 and 8 are compared analytically, numerically and experimentally, using the error indicators presented in Chapter 5. This dissertation is concluded in Chapter 9 with a summary of achievements, and a brief discussion of future research needs for further development of hybrid simulation.

2 HYBRID SIMULATION METHODS – STATE-OF-THE-ART

Development of improved hybrid simulation testing methods has been the subject of vast research in recent years. Extensive attempts have been made to develop improved test algorithms and enhanced procedures for numerical simulation and compensation of system dynamics, to improve the robustness of this test method. The development of George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES), which provides researchers with advanced testing, processing and communication tools, has also contributed to the increased interest in fast and geographically distributed hybrid simulations.

This chapter presents on overview of state-of-the-art research methods for hybrid simulation through a concise review of the literature. Existing publications are categorized into several groups based on their primary subjects, such as test algorithms, hardware, error estimation and compensation, and numerical integration methods. In this study, the physical components of hybrid simulation system, their configuration schemes and communication tools are termed the test hardware. Other simulation

components, such as test algorithms and computational subroutines for structural simulation, system identification, signal conditioning and error compensation are referred to as test procedures.

2.1 DEVELOPMENT OF FAST AND DISTRIBUTED HYBRID SIMULATION PROCEDURES

Fast and real-time hybrid simulation techniques owe their growth at early stages to the slow (quasi-static) pseudo-dynamic test procedures developed in past decades [1-3, 14-16]. The numerous shortcomings of static tests, such as their inability to capture the rate-dependent behavior of experimental components and problems with stress relaxation, have persuaded researchers to seek fast online implementations of this test technique. A real-time hybrid simulation, in which experimental substructures are loaded at realistic rates, has the ability to accurately identify the dynamic and rate-dependent properties of experimental substructures [4].

The pseudo-dynamic method was shown to be an effective method for seismic testing of structural systems in 1970's and 80's in Japan and the United States [1, 17, 18]. It was demonstrated that the pseudo-dynamic method of structural testing can be as reliable as earthquake simulations using shaking tables [1], while providing full scale testing capability of structural components. Mahin *et al.* [2] summarized the basics of pseudo-dynamic tests, their capabilities and limitations, and advantages over shaking tables in testing large and heavy specimens. They introduced the concept of

substructuring to allow for economical and realistic testing of structural sub-assemblages.

The Japanese activities on online testing until mid 80's have been summarized by Takanashi and Nakashima [18]. In addition to a summary of the development history of hybrid testing, they discussed the reliability issues of online tests, and extended the online test concept to fast online and substructure testing. They concluded that a fast online testing is necessary to capture the rate-dependent behavior of structural specimens and substructures.

Nakashima and Masaoka [19] successfully conducted simulations of a 5-story base-isolated building. Nakashima [4] studied the recent developments and limitations of online tests, and presented a test system for conducting such experiments. In that test system, the response analysis and signal generation tasks were separated in order to provide means for testing of complex structures; in each integration time step, the multitasking system continues to extrapolate the higher-rate command displacements as long as the lower-rate response analysis task is being completed. After that, an interpolation procedure calculates the command displacements for the rest of time step. Nakashima [4] identified an important advantage of hybrid simulation in reduction of the need for scaling (size and rate of loading), which can significantly affect the structural behavior. Nakashima also pointed out the need for further development of the procedures for stiff structures and application to structures loaded by multiple actuators.

2.1.1 GENERALIZED HYBRID SIMULATION METHODS

Sivaselvan [10] presented a unified view of hybrid simulation algorithms, from which many existing algorithms can be derived as particular cases. Dynamic and pseudo-dynamic test methods were identified based on physical existence of mass, and hence, the inertia forces. It was shown that an adaptive Smith predictor can be used for control and error compensation of test system. The use of Smith predictor for compensation of system dynamics was also studied by Reinhorn *et al.* [20] and Shao *et al.* [5] in force-controlled hybrid simulations.

Shao [9] introduced a unified approach to hybrid simulation by using shake tables and actuators for experimentation of substructures. A general formulation was developed that included the special cases of shake table, pseudo-dynamic, and effective force testing techniques. The development was verified using proof-of-concept experiments of a small linear test setup with uni-directional shake table and actuator.

2.1.2 SOFTWARE FRAMEWORKS

Recently, attempts have been made to develop general and flexible software environments for performing distributed hybrid simulations [21-24]. These software utilize object-oriented frameworks, and have been developed to connect general finite elements analysis software to local or distributed experimental subsystems. Development of generalized frameworks has been a challenging task for popularizing the use hybrid simulation test technique, due to the extensive diversity of laboratory equipment and complexity of the structures to be tested. For this reason, the source

codes of many of these software frameworks are open to the user, so that additional subroutines can be introduced for use in different laboratories.

The Open-Source Framework for Experimental Setup and Control (OpenFresco) was developed at the University of California – Berkeley by Schellenberg *et al.* [21, 22]. This package includes a set of interrelated software classes based on object-oriented software design methodologies, which form a framework for integrating experimental testing with the object-oriented finite element analysis software OpenSees (Open System for Earthquake Engineering Simulation [25]). OpenFresco mediates the instructions between the numerical simulation computer and local or remote laboratory equipment in a highly structured manner.

The Multi-Site Substructure Pseudo-Dynamic Simulation Coordinator (UI-SIMCOR) is another software package for this purpose that was developed at the University of Illinois at Urbana-Champaign by Kwon *et al.* [24]. This package uses a variety of communication protocols to integrate its numerical simulation with other analysis software or laboratory test equipment in local or remote sites.

2.1.3 HYBRID SIMULATIONS MODELS

Analytical models of hybrid simulation test systems have attracted the interest of several researchers. These models provide the potential for better identification of the complex dynamics of experimental subsystems, and hence, development of more effective compensation procedures. In addition, such numerical models can be used in

the preliminary evaluations of new test procedures on a computer, thus reducing the cost of such investigations and preventing potential damages to laboratory equipment.

Williams *et al.* [26] developed a computer model for hybrid simulations, including actuator, PID controller and experimental specimen. By comparing their results to experimental simulations, they showed that the model is able to predict the global system behavior, although their results were shown to be very sensitive to bulk modulus of the oil. As a result, the authors suggested the use of this model for a wide range of oil bulk modules to capture the expected global behavior of the experiment.

Jung and Shing *et al.* [27-29] introduced a transfer function for pseudo-dynamic test system. They demonstrated the necessity of removal of inertia effects from the experimental restoring forces in order to achieve the same resonance frequency as the exact solution. In fact, they did not consider the effects of experimental mass in the development of their transfer function, thus requiring the above-mentioned consideration.

Zhao *et al.* [30] developed mathematical models for real-time large-scale structural testing methods such as effective force technique and pseudo-dynamic experiments. Mercan *et al.* [31] demonstrated the design of PID and state-space controllers, pointing out the advantage of the state-space form, especially in multi-input-multi-output systems. The state-space controller was designed using an analytical model of the test system.

2.1.4 SPECIAL TEST PROCEDURES

Geographically distributed hybrid simulations have received close attention in recent years [11, 13, 32]. Using this approach, testing equipment in several laboratories can be utilized to test multiple experimental substructures of large structural systems. Mosqueda *et al.* [11] studied the implementation and accuracy of continuous hybrid simulations with geographically distributed experimental substructures. By comparing the results of geographically distributed experiments with those of conventional local tests, they demonstrated the effectiveness of this test method.

Pan *et al.* [13] presented an internet online test system developed for the simulation of earthquake response of structures. They carried out this test between two universities in Japan, and demonstrated the effectiveness of internet communications for distributed hybrid simulations. Guo *et al.* [32] demonstrated the application of a collaborative hybrid dynamic testing software based on network structural laboratories (NetSLab) along with a network communication platform and testing software to a distributed hybrid simulation. They demonstrated the validity of geographically distributed tests through pseudo-dynamic experiments of bridge columns. Several software frameworks have also been developed for geographically-distributed hybrid simulations [21-24] that provide means for communication of different analysis and experimentation components in local or remote sites (Section 2.1.2).

Nakata *et al.* [33] presented a mixed load-displacement control method for coupled multi-axial systems at static loading rates. They utilized Broyden formula [34] to update the stiffness matrix for use in their iterative integration scheme. The experimental

verification was carried out on reinforced concrete column specimens to demonstrate the effectiveness and robustness of their approach for highly inelastic systems.

Effective force testing methods [6-8] have also been subject to extensive developments in recent years. The major difference between this test method and pseudo-dynamic testing is that it is based on a force-controlled loading system. In addition, since the structural mass can normally be well-identified, the loading history is known prior to simulation, unless mass is modeled virtually, and experiment feedback is used in determination of loading history [6].

2.2 *HYBRID SIMULATION HARDWARE*

Mahin and Shing [1] pointed out that pseudo-dynamic experiments can be carried out with the basic equipment that is readily available in most structural simulation laboratories, implying the low cost of these experiments compared to shaking table tests. However, recent hardware designed especially for hybrid simulation has improved the speed and accuracy of testing over the past years. Digital controllers, servo-hydraulic actuators with enhanced tracking capabilities and reduced response lag, better measurement instrumentation, communication systems with extremely small delays in order of nanoseconds, faster processors and enhanced data storage equipment, along with other hardware improvements have enabled researchers to perform fast, real-time and geographically distributed hybrid simulations [11].

Mosqueda *et al.* [11] outlined a versatile hardware architecture for hybrid simulation with a distributed control scheme. Their improved hardware architecture

allows for a faster and continuous execution of hybrid simulations. By adopting multi-tasking in real-time applications, they separated signal generation and integration tasks, thus facilitating the use of more sophisticated integration and compensation procedures for complex analytical substructures. They also introduced a three-loop hardware architecture consisting of the integrator, corrector, and servo-control loops. The middle corrector loop is an intermediate loop that buffers the exchange of data between the integrator and servo-control loops. This loop also guarantees continuous simulation by carrying out interpolation of data when the analysis results are ready, and extrapolation of results when the analysis is still in progress [19].

Mosqueda *et al.* [11] used a Shared Common Random Access Memory Network (SCRAMNet) interface [35-37] as a fast communication device among simulation components. They also extended their hardware architecture to the experiments over the internet by employing an event-driven simulation scheme [38].

In order to carry out low-cost nonlinear hybrid simulations, Mosqueda *et al.* [11] developed an experimental setup consisting of a column with a clevis at the base. Steel coupons at a weaker capacity than the column inserted at the clevis provided nonlinear response, while the column remained linear for further simulations. The experimental setup used in this dissertation is developed on the same basis, and will be presented in Chapter 3.

Internet linkage of distributed test and analysis domains was studied by Yang *et al.* [12] and Pan *et al.* [13]. They demonstrated the architecture of simulation and experiment modules, equipment control and network communications, as well as data

acquisition and visualization systems. Reinhorn *et al.* [20] and Shao *et al.* [5, 9] designed a unified platform for an arbitrary state-of-the-art testing facility. The major components of platform were identified, and their functions were established. In their development, both actuators and shaking tables were used to apply dynamic forces on the experimental substructure. They demonstrated that their general test system can be simplified to the special cases of shake table, effective force, and pseudo-dynamic tests. The specific developments were based on the structural test facilities at University at Buffalo, which will be presented in Chapter 3.

2.3 NUMERICAL AND EXPERIMENTAL ERRORS

The random nature of physical experiments along with the imperfections of the test equipment, normally results in small but important differences among desired and achieved displacement histories. In an ordinary open-loop experiment, where the displacement or force history imposed by the actuators are previously known, these small differences may be tolerable, as the offline tools for processing experimental data generally make it possible to remove the majority of these errors.

In feedback systems, however, these errors are included in the signals that will be used to generate next actuator command; as a result, the errors can accumulate and increasingly contaminate the simulation results. Several researchers in different fields of engineering have faced this problem, and it has been shown that it is essential for the stability and accuracy of simulations to make corrections and compensations on the signals being transmitted among numerical and experimental substructures. These

errors, if not properly addressed, may make the simulation results unreliable, or even unstable, resulting in premature damages to the experimental specimens and test equipment.

The importance of errors and their potential effects on simulation results was recognized early on in slow pseudo-dynamic tests. Mahin and Shing [1] showed that both numerical and experimental errors can accumulate through simulation steps and contaminate the results, with the majority of errors originating from experimental sources. They also pointed out the difficulty associated with stiff (high-frequency) systems, which are extremely sensitive to experimental errors.

Thewalt and Mahin [3] studied pseudo-dynamic testing as an alternative to shaking table tests. They demonstrated the importance of the removal of experimental errors to prevent error propagation, and spurious excitation of higher-frequency modes. They also proposed procedures for performing real-time, or close to real-time hybrid simulations. Thewalt and Roman [16] introduced several performance parameters for identification of systematic errors and quantifying the uncertainties. Those parameters could also be used to apply signal corrections during the testing.

Monitoring the errors associated with hybrid simulations has been studied by Mosqueda *et al.* [11, 39-41]. In those studies, they introduced error indicators that help ensure the simulation quality during the experiment. The error measures are based on a comparison of the energy dissipated by the specimens and energy dissipation apparent to the numerical integrator. If large errors are detected early, the simulation can be stopped in order to prevent damages to the experimental specimens when the

simulation quality is not acceptable. Errors resulting mainly from uncompensated delay and their effects on the structural response were investigated, and it was shown that the accuracy of the simulation can be predicted based on these experimental energy error measures.

In order to estimate the extent of numerical errors in nonlinear seismic analyses, Filiatrault *et al.* [42, 43] utilized energy balance error as a measure of simulation accuracy. They provided examples of the use of energy balance concept, and demonstrated that the energy balance formulation better detects the errors than conventional peak parameters, such as displacements.

2.4 COMPENSATION OF SYSTEM DYNAMICS AND DELAY

As mentioned in the previous section, control of experimental and numerical errors is crucial for stability and accuracy of hybrid simulation. Careful consideration is necessary to minimize such errors and improve the quality of hybrid simulations. However, it is impossible to completely eliminate all error sources and achieve an ideal signal tracking from the actuator, or perfectly smooth and accurate measurements. For this reason, other procedures have been developed that attempt to correct the remaining errors and reduce the experimental imperfections.

One of the major sources of the above-mentioned errors is the dynamics and delay of servo-hydraulic actuators. Since the problem of control and delay compensation in feedback systems is faced in different engineering fields, including mechanical, chemical and electrical engineering, the research dedicated to this field is quite extensive. In the

field of structural engineering, the issue of actuator dynamics and delay has been extensively studied in active control of structures and hybrid simulations. These feedback systems mainly deal with extreme, transient and random effects of earthquake excitations. A concise review of the literature on delay compensation methods for structural applications is presented in the following sections.

2.4.1 DELAY COMPENSATION FOR ACTIVE CONTROL OF STRUCTURES

A number of recent and widely-used delay compensation methods for active control of structures in presented in this section. Agrawal *et al.* [44] studied the instability due to time delay and its compensation in active control of structures. They showed the dependency of maximum allowable time delay on the natural period of the structure and feedback gains. They presented upper limits on the time-delay to ensure the stability of the system. By considering the delay as transportation lag, the authors proposed a linear controller that compensates the delay by using the present states of the system and past control force information. Their method was shown to slightly increase the required energy for compensation of time-delay.

Qi and Kuang [45] used an extended linear Kalman optimal filter to compensate the time delay in active closed-loop structural control. This approach, however, is difficult to apply to hybrid simulations, as Kalman filter requires closed-form information about the state equations of the system, which is normally unavailable in a hybrid simulation.

Agrawal and Yang [46] presented a state-of-the-art review of available methods for time-delay compensation for control of civil engineering structures. Through numerical

simulations, they demonstrated the stability and control performance of several compensation procedures, including the recursive response method, state-augmented compensation, controllability-based stabilization, Smith Predictor, and Padé approximation. In addition, they evaluated the performance of phase shift method [47, 48] to show that this approach is not reliable for larger delays or high-gain controllers (high damping in various modes), and may destabilize the system. On the other hand, they proposed the recursive response and state-augmented compensation methods with better stability and control performance. They also showed that the controllability-based compensation outperforms the phase shift method for larger time-delays. They stated that the Padé approximation method is always stable and capable of compensation of all delay values with minimal performance degradation. Finally, the guaranteed stability of Smith Predictor and its ability to compensate small time-delays without considerable performance degradation were also demonstrated. A number of these procedures used in active control of structures may have the potential to be used in hybrid simulation, and will be discussed next and in Chapter 6.

2.4.2 DELAY COMPENSATION IN HYBRID SIMULATIONS

As mentioned in the previous section, Smith Predictor is one of the delay compensation procedures in active control of structures that have been shown to be stable and able to compensate small time delays without significant performance degradation. Smith Predictor has also been used in linear force-controlled hybrid simulations [5, 20], where the numerically computed forces are commanded to the actuator, and displacements are measured and fed back into the numerical model.

However, the major drawback of this procedure is that it needs an accurate numerical model of the system.

Compensation procedures for hybrid simulation should be able to eliminate the negative effects of actuator dynamics and delay without knowledge of a precise analytical model for the experimental setup. These models are normally unavailable in hybrid simulations, due to the complex servo-hydraulic test system and unidentified behavior of experimental substructures. For this reason, it is often impossible to directly implement active control compensation methods in hybrid simulation, and other procedures have been developed specifically for hybrid simulation.

In order to reduce the systematic errors in pseudo-dynamic experiments, Yi and Peek [49] considered adjustments in time step. They demonstrated that this approach is particularly successful in the elimination of systematic errors of single-degree-of-freedom (SDF) systems, as long the associated random errors are small, as the procedure may have an undesirable effect on them. They also showed that their approach improves the error propagation properties of MDF systems.

Horiuchi *et al.* [50] demonstrated the importance of the delay in real-time hybrid simulations, and its effect as a negative damping, which may lead to instability of the system. They developed a hybrid simulation system with a delay compensation procedure, which predicts the required displacement at a time equal to one time step ahead of current simulation time. By comparing the results of their hybrid simulation with those obtained from shaking table tests, they verified the accuracy and reliability of hybrid simulation with delay compensation. Horiuchi and Konno [51] improved the

above-mentioned delay compensation method by including velocity and acceleration in their predictions. In this method, the acceleration was linearly extrapolated from the current step, which can then be used to implicitly determine the predicted displacement. Since in an explicit integration procedure, the acceleration and velocity are only available for the previous step, the delay has to be increased by one integration time step in this method.

System dynamics and delay of displacement-controlled hybrid simulations were also studied by Zhao *et al.* [30]. They investigated the amplitude reduction and response delay phenomena in their simulations, and concluded that physical mass may result in amplitude reduction by limiting the maximum proportional gain. They also showed that first order phase-lead networks can be used to compensate system dynamics when hydraulic demands are small, while more advanced schemes are necessary to account for nonlinearities in servo-systems.

Darby *et al.* [52-54] studied real-time substructure testing methods using hydraulic actuators and delay compensation procedures. They pointed out the dependency of the actuator delay on the stiffness of the specimen, and proposed a method for estimation of delay on a real-time basis. They demonstrated the accuracy of the estimated delay in single- and two-degree-of-freedom systems subjected to sinusoidal excitations. The delay compensation was then carried out using the prediction of the required displacement at a time equal to the one time step ahead [50].

Carrion and Spencer [55] used an adaptive experimental tangent stiffness matrix to predict the restoring force due to the desired displacement and compensate for time

delay of real-time hybrid simulations. They used Broyden formula [34] to update the stiffness matrix, and demonstrated that larger natural frequencies that can be tested than when polynomial extrapolation is used [50].

2.5 *NUMERICAL INTEGRATION PROCEDURES*

Numerical integration procedures are time stepping algorithms to solve the equation of motion by updating the states (displacements, velocities, and accelerations) in each simulation time step. This is carried out in the numerical simulation module, and is one of the most challenging tasks in a hybrid simulation.

The direct implementation of iterative implicit integration procedures may be difficult in hybrid simulations due to the involvement of physical substructures. Particularly, physical iterations should be avoided, as they may unexpectedly damage the experimental substructures in displacement reversals and steps with failed convergence, or alter the energy-dissipation properties of test specimens. Estimation of an updated tangent stiffness matrix may also be difficult due to the noise in measurements. These limitations have made the explicit integration procedures quite popular in hybrid simulations [11] for their efficiency and accuracy. These procedures do not include iterations, and the command displacements sent to actuator directly reflect the desired displacement of the current time step.

Explicit central difference integration was originally used for implementation of online experiments by Takanashi *et al.* [17]. This method is considered as the basic integration method for relatively simple applications of hybrid simulations [19, 56].

More recently, Bonnet *et al.* [57] used explicit Newmark method to avoid iterations in hybrid simulation. They proposed an outer loop control (a compensator) to eliminate the dynamics of displacement imposition system. In this controller, the desired command displacement is used in a numerical model of the loading system (reference model), to give an estimate of the restoring force. The control signal is then composed of two signals: the estimated restoring force and measured displacement, multiplied by their corresponding gains. These gains are constantly updated by an adaptation routine using three pieces of information: estimated restoring force, measured displacement, and the displacement error signal (difference between measured and desired displacements). This method was implemented in a multitasking environment.

Despite their simplicity and applicability to hybrid simulations, explicit integration procedures are conditionally stable, and the maximum integration time step is restricted [58]. This stability limit is directly related to the highest natural frequency of the system and may require very small time steps, regardless of actual contribution of high-frequency modes to structural response. This results in the necessity of very small time steps that are impractical for stiff (high-frequency) systems, or unsuitable for networked applications of hybrid simulations. In addition to increasing the length of non-real-time experiments, an increase in the number of time steps will increase the potential for error propagation. Further, time delay will be more problematic when the time steps become very small compared to the delay amount. For these reasons, extensive research has been dedicated to the development of integration procedures with better stability and

accuracy characteristics that take special precautions for imposition of iterative displacements on experimental substructures.

Thewalt and Mahin [3] pointed out the difficulty associated with the estimation of tangent or initial stiffness matrix of experimental substructures. They used a modified Newmark integration method proposed by Hilber *et al.* [59], that includes a parameter that can be used to introduce additional damping in higher modes of the system. With this capability, the damping term of the equation of motion can be eliminated, and all required damping can be considered in the numerical damping of the integration procedure. This method, known as α-method, was used to determine the command displacement expression based on available data (explicit terms) and the restoring force of command displacement, which is not available before imposition of command displacement and digitization of force measurement.

In order to address this issue, Thewalt and Mahin [3] introduced a valve command signal that is the result of analog summation of explicit terms of command displacement (which is determined by computer in digital domain) and a feedback term, which forms the implicit portion of command displacement. As the command displacement is being applied on the test specimen, the feedback signal automatically uses the new restoring force to update the command voltage (control signal), and finally, the command displacement will be completely applied including the implicit term. In this way, the effect of actual restoring force will be applied in the command displacement determination, although it is not known by the numerical integrator.

While the above-mentioned iterative implicit method is theoretically feasible, there are several issues associated with its implementation. First, the force measurements in experiments are generally highly oscillatory; use of low-pass filter will then add to the delay of force feedback, and the implicit displacement may not be fully achieved in the allotted step time. In addition, the displacement and force should both be measured at the end of step, and there is no means for ensuring their accuracy. Further, the convergence of this feedback system may be bi-directional, which can result in erroneous energy dissipation or damage to the substructures. Similar implicit integration methods with feedback loops involving experimental substructures have been developed in digital domain [28, 29, 60, 61]. Shing *et al.* [28] also imposed a fixed number of iterations to ensure the completion of integration task within the allotted time.

Shing [61] proposed an implicit integration procedure, in which the iterative displacements imposed on substructures are not oscillatory; in other words, the convergence is attempted to be uni-directional by using a reduction factor on the iterative displacements of α-method in multi-degree-of-freedom systems. By implementing a large number of small increments in each step, the final convergence situation will be equivalent with the method proposed by Thewalt and Mahin [3] in the analog domain. They also completed this procedure by introducing easy-to-implement convergence criteria. The accuracy of this technique was demonstrated with error-propagation analysis for linear systems; in addition, it was shown that the procedure

adds damping to higher-frequency modes of the system that could otherwise be excited by measurement noise and convergence error.

The above-mentioned implicit integration procedure [61] was recently extended to MDF experimental substructures in real-time simulations by Jung *et al.* [62]. They adopted a fixed number of iterations in each integration step to ensure the completion of simulation task within the allotted analysis time. In order to avoid velocity fluctuations from one integration step to another, and to provide a smooth command displacement signal for the actuators, Jung *et al.* adopted the following procedure. In each iteration, the iterative displacement vector is calculated using an implicit formulation and the initial experimental stiffness matrix. Then, a fraction of the trial displacement at each experimental degree of freedom (computed using a quadratic interpolation) is applied to the test structure. Following the application of the trial displacement vector, the restoring force vector is measured and used for calculation of next trial displacements, starting from the last set of measurements. At the end of the iterative scheme of each integration step, an additional correction is made on the restoring force vector to account for actuator tracking errors, using the initial experimental stiffness matrix.

Wu *et al.* [63] used the equivalent force method for solving the nonlinear equation of motion in a real-time substructure test. In their approach, the implicit iterations were replaced with force-feedback control loop. The equivalent force feedback was a combination of external force and the pseudo-dynamic effects of the previous integration step. They verified the effectiveness of their integration approach in experimental studies of a magnetorheological damper.

Shing *et al.* [28, 29] developed transfer functions for all components of hybrid simulation hardware, and used the iterative α-method with a fixed number of iterations, to ensure completion of calculations within a practical amount of time. Instead of physical application of iterative displacements, they used initial stiffness for estimation of iterative forces, which reduces the communication requirements of hybrid simulation. The use of initial experimental stiffness matrix results in exact simulations for linear systems, but possesses limited accuracy for testing highly nonlinear systems, where the experimental tangent stiffness matrix significantly differs from its initial value.

In order to avoid physical iterations, Ghaboussi *et al.* [64] utilized an implicit integration method only in the numerical substructure; the interface forces between the numerical and experimental substructures were considered as external forces acting on the numerical model, which were assumed to remain constant throughout any step. As a result, there was no need for calculation of initial stiffness matrix, or application of iterative displacements to the experimental substructure. They presented a stability analysis of this approach and demonstrated its conditional stability, depending on the simulation time step normalized by the circular frequency of the highest natural vibration mode. Schneider and Roeder [65] utilized a similar approach using DRAIN-2D software and subtracting the incremental experimental restoring force from external excitation.

Nakashima *et al.* [14] studied the integration algorithms developed for hybrid simulation, and proposed the operator-splitting method for substructure pseudo

dynamic tests. In operator-splitting method, the desired displacement at each step is based on the implicit Newmark method. However, the command displacement signal is composed only of the explicit terms. After measuring the restoring force due to this command displacement, a correction step is conducted that updates the displacements to satisfy the original implicit equation. The restoring forces are then updated according to this change in displacement vector using the initial experimental stiffness matrix. Nakashima *et al.* demonstrated the unconditional stability of this method for linear systems, and stated that the stability can also be achieved in nonlinear systems with softening behavior.

Also using the initial experimental stiffness matrix, Chang [66] developed an integration method based on solving momentum equations. The momentum equations were adopted to eliminate the difficulties arising from the presence of high-frequency modes in online dynamic test signals, making explicit implementations of numerical integration effective. The improved error propagation properties of this method were demonstrated using linear analysis. Chang *et al.* [67-71] presented several other explicit integration procedures with unconditional stability. These procedures, which take advantage of the initial stiffness matrix of experimental substructures, were shown to improve the error propagation properties of the algorithm for displacement and force signals. However, the stability verifications were based on linear systems, and no discussions about stiffness matrix replacement were included in terms of applicability to highly nonlinear specimens.

Using the initial stiffness matrix approximation and α-method formulation, Combescure and Pegon [72] presented the α-operator-splitting integration scheme. They demonstrated the unconditional stability of this method for linear systems, and examined its error-propagation properties. Combescure and Pegon concluded that this method is effective for systems that do not lose a large portion of their initial stiffness during simulation.

The predictor-corrector method of integration was considered by Zhang *et al.* [73] in state-space form, which is more common in the field of structural control. Although the main issue in the use of this method is the use of initial linear stiffness matrix instead of the tangential stiffness in the completion of corrector step, only linear verifications were carried out in the above-mentioned work. In the predictor step of this approach, the forces corresponding to next displacement were required and should be determined by a polynomial extrapolation.

The correction of displacement and restoring force in operator-splitting family of integration procedures are generally based on the difference between the predictor (or measured) and corrector displacements. Wu *et al.* [74] extended this correction by considering the rate-dependent forces developed in the experimental substructures. In this approach, a predictor velocity is also determined, and after application of predictor displacement, the difference between the measured and corrector velocities is used for rate dependent restoring force correction. Through experimental simulations, the authors showed the improved performance of the procedure in comparison with the explicit central difference method, especially in cases that experimental substructures

exhibit a fair amount of damping. Wu *et al.* [75] later presented the discretized state-space form of operator-splitting method, and studied the effects of delay and its compensation on the stability of this integration algorithm.

In order to improve the accuracy of hybrid simulation for testing nonlinear systems while avoiding physical iterations, Pan *et al.* [13] introduced a method for estimation of tangential stiffness of SDF experimental substructures. For this purpose, they used linear functions fitted to most recent measured force and displacement data. As a result, the need for using the initial stiffness matrix was eliminated and no iterative displacements were applied to experimental substructures.

2.6 OTHER PUBLICATIONS AND CASE STUDIES

Applications of pseudo-dynamic test methods to specific cases of hybrid simulation have also been widely published. Through these example applications, the implementations of several test techniques were illustrated, and new procedures were introduced [13, 76, 77].

Fahnestock *et al.* [76] used the pseudo-dynamic testing method for a large-scale buckling-restrained braced frame and demonstrated their minimal strength and stiffness degradation. Elkhoraibi and Mosalam [77] conducted pseudo-dynamic experiments on reinforced concrete frame structures to study the effects of infill walls, using a mixed-formulation implicit algorithm. They used secant stiffness in their integration algorithm, assuming a nonlinear elastic experimental model.

Tsai and Lin *et al.* [78, 79] presented the bi-directional pseudo-dynamic test of a full-scale buckling-restrained braced frame, and evaluated the design and performance of gusset plates. Mercan and Ricles [80] described pseudo-dynamic testing of a 3-story steel moment resisting frame with elastomeric dampers in the first floor. They introduced a velocity feed forward controller in the servo-control system to minimize the time delay.

Yang *et al.* [81] used OpenSEES simulation software to test the first story of a zipper frame as the experimental substructure. Both numerical and experimental nonlinearities were considered, and it was shown that the results were in agreement with those of analytical simulation. Yang *et al.* [12] presented a transnational pseudo-dynamic experiment on a double-skinned concrete-filled tube pier bridge system. The specimens were located in two universities in Canada and Taiwan. They described the use of simulation software, network operations and advanced data acquisition systems, which enable the visitors to simultaneously observe the experiment progress through their web browsers.

3 SEESL PSEUDO-DYNAMIC HYBRID SIMULATION SYSTEM

In this chapter, the pseudo-dynamic test system at Structural Engineering and Earthquake Simulation Laboratory at University at Buffalo is described. First, an overview of the hybrid simulation system is provided and the software environment for the development of simulation models is introduced. The experimental setup and simulation model details are presented next, and a step-by-step guide for conducting a displacement-controlled hybrid simulation is provided. In Section 3.6, the PC simulation models of hybrid simulation are presented, which can be used to replace the experimental setup for preliminary analyses and development purposes.

3.1 EQUIPMENT AND HARDWARE ARCHITECTURE

Figure 3-1 shows the major components of SEESL hybrid simulation system [9]. As illustrated, the components are divided into three major groups: Data acquisition and streaming, hybrid simulation controllers, and testing controllers. The components of this test system are interconnected using Local Area Network (LAN) for model and data

transfer, and Shared Common Random Access Memory Network (SCRAMNet™) [35-37] for fast communications during real-time experiments.

As the name suggests, the data acquisition and streaming subsystem handles the storage and communication of experimental measurements command signals. In addition to local communications of test components, the NTCP server provides internet communication channels for use in geographically distributed hybrid simulations.

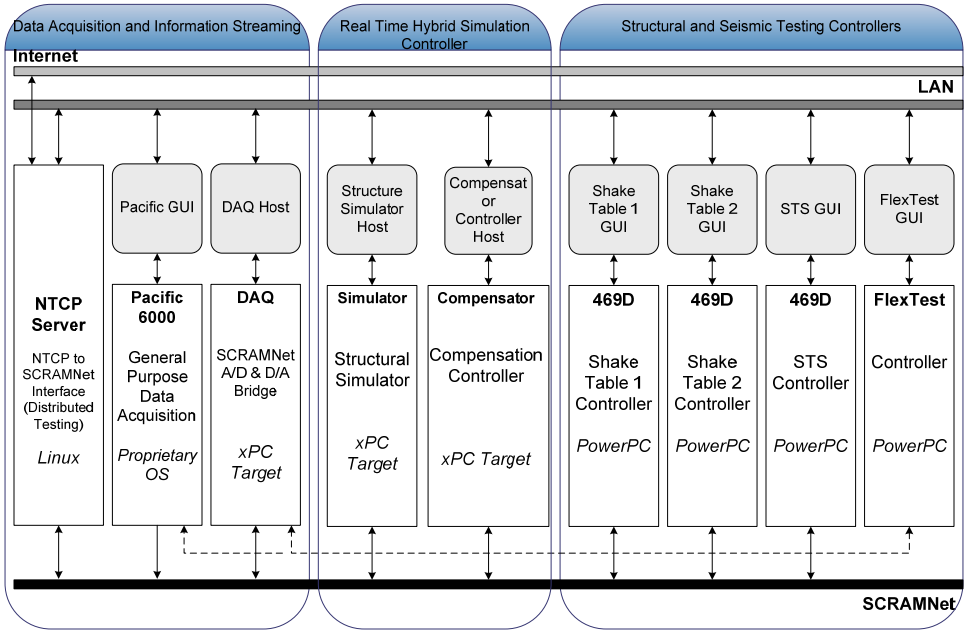

Figure 3-1 Hardware architecture of hybrid simulation test system at SEESL [9].

Hybrid simulation controllers, consisting of Host and xPC Target computers handle the numerical simulation and signal generation tasks. The host computers run MathWorks Simulink [82] software for development of simulation models and test procedures. These models will then be downloaded to real-time computers labeled xPC Targets for real-time online execution. The software and real-time environments are briefly described in the next section.

As shown in Figure 3-1, the test controllers consist of shake table and actuator controller computers, along with their graphical user interfaces (GUI). Note that the user interfaces of several components of the test system can be run on the same computer running a multimedia operating system such as Windows [83]. More information about these interfaces can be found in SEESL Laboratory Manuals [84].

The hardware arrangement in Figure 3-1 can be directly compared to the pseudo-dynamic test block diagram shown in Figure 1-2. In Figure 1-2, the experiment subsystem corresponds to test controllers, analysis subsystem represents hybrid simulation controllers, and data acquisition system handles the communication of the measurements between numerical and experimental subsystems.

3.2 SOFTWARE DEVELOPMENT ENVIRONMENT

The simulation model and test procedures are developed in MathWorks Simulink environment [82]. Simulink is a tool for modeling, simulating and analyzing dynamic systems. Its primary interface is a graphical block diagramming tool and a customizable set of block libraries. It is integrated with the rest of the MathWorks MATLAB [85] package, providing access to its extensive range of tools for algorithm development, visualization, data analysis and access, and numerical computation. Simulink is widely used in control theory and digital signal processing for simulation and design. Coupled with Real-Time Workshop [86], Simulink can automatically generate C code for real-time implementation of models.

For real-time[1] simulations, the models built in Simulink should be downloaded to a real-time environment called xPC Target. xPC Target provides a prototyping environment that enables the user to connect Simulink models to physical systems and execute them in real time on PC-compatible hardware. For this purpose xPC Target supports the use of I/O interface blocks in simulation models. The use of a real-time kernel such as xPC is important in simulations involving physical systems, to ensure accurate development of rate-dependent effects in the experimental substructures.

In order to improve the computational efficiency for real-time simulations, xPC Target uses a simple graphical interface that can be used for online visualization of only a few signals. For this reason, most of the simulation results need to be retrieved in the host PC for improved visualization and processing tools. That is, both pre- and post-processing tasks are preferred to be carried out in the host PCs, while the real-time simulation itself is performed on xPC Targets.

3.3 TEST SETUP

Figure 3-2 shows the two-degree-of-freedom test setup at SEESL facilities at University at Buffalo. As shown, the experimental specimen consists of two short columns mounted on top of two clevises. The specimen is loaded at two points close to tip of each column. Replaceable steel coupons can be inserted in the clevises to provide

[1] In this section, the term "real-time" has a different implication than that of real-time hybrid simulations. A real-time environment is able to execute simulation tasks with actual time intervals, and may be used to perform slow online, fast, or real-time hybrid simulations.

moment resistance at a weaker capacity than columns. As a result of lateral loading, the yielding first occurs in sacrificial coupons while columns remain elastic, providing the capability of low-cost nonlinear hybrid simulations.

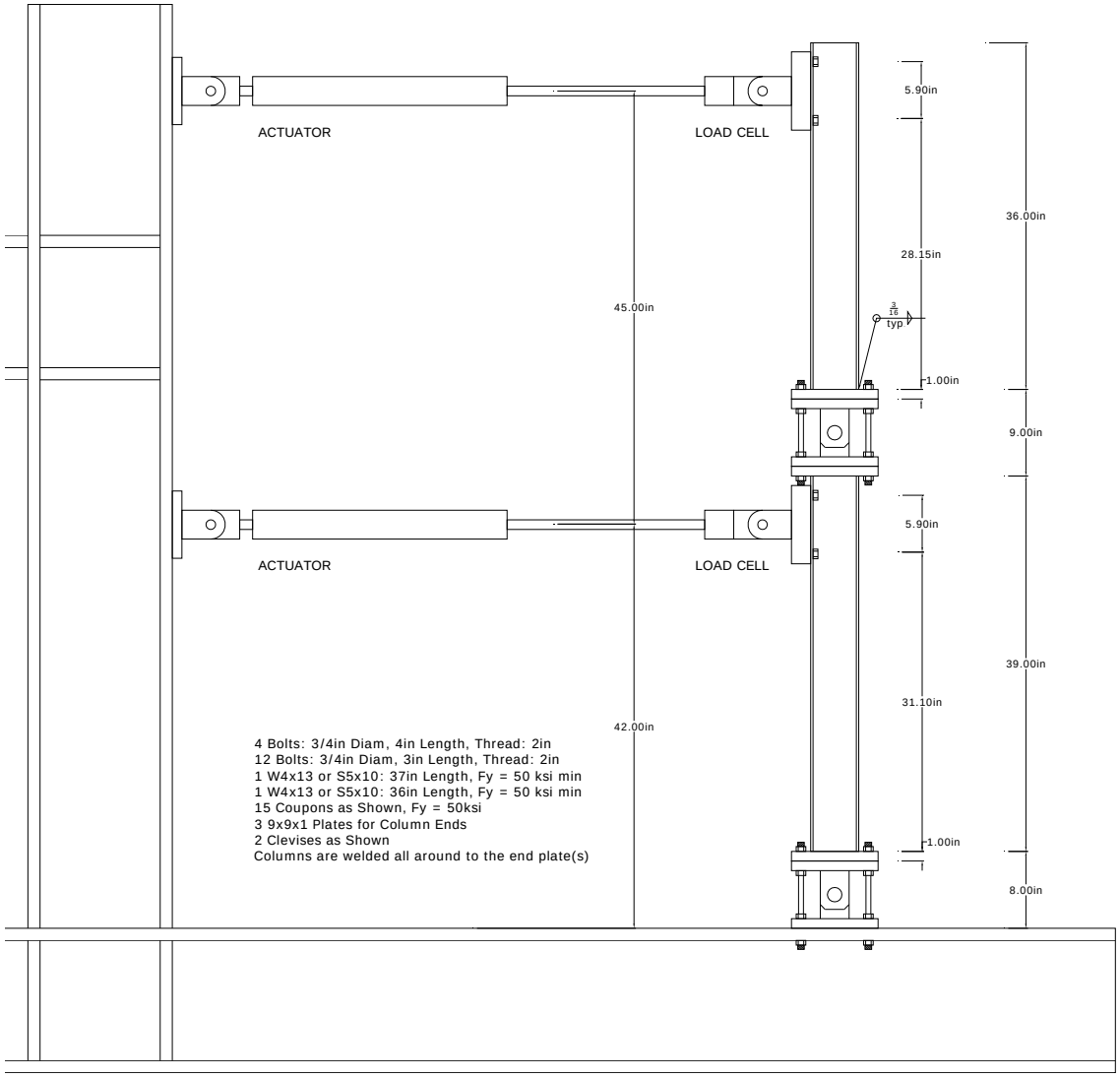

Figure 3-2 UB-SEESL two-degree-of-freedom experimental setup for hybrid simulation.

Up to three pairs of coupons can be inserted in each clevis to provide the desirable lateral resistance. The test setup picture in Figure 3-3 shows the lower clevis with two pairs of coupons. The number of coupons should be selected not to exceed the 22-kN (5-

kip) loading capacity of the actuators. Further, the excitation record should be scaled such that the expected displacements will not exceed the 76-mm (3-in) actuator stroke. As shown in Figure 3-4 a SDF experimental setup can be obtained by removing the upper actuator and column assembly.

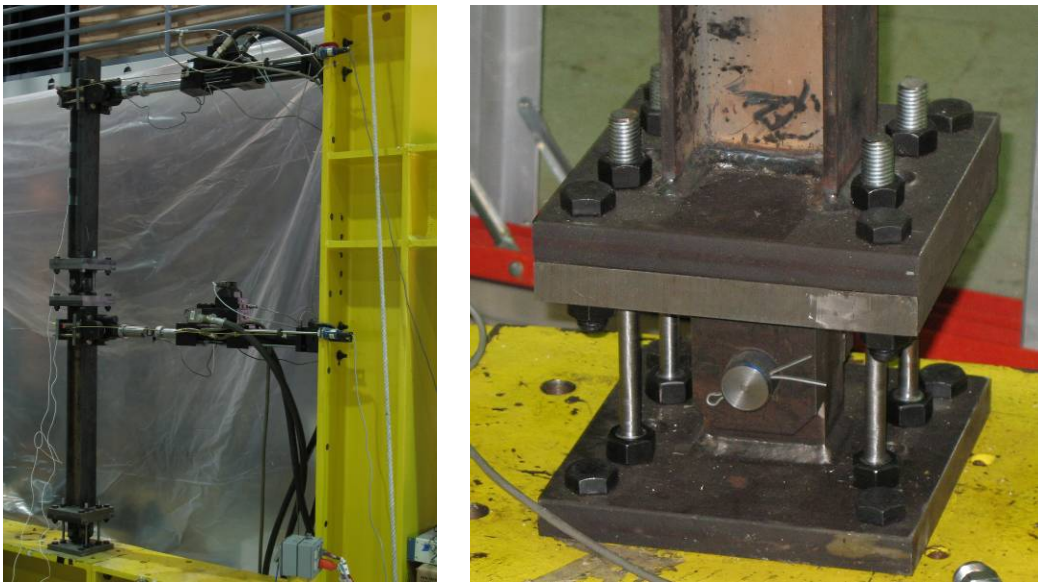

Figure 3-3 Pictures of two-degree-of-freedom test setup and the lower clevis.

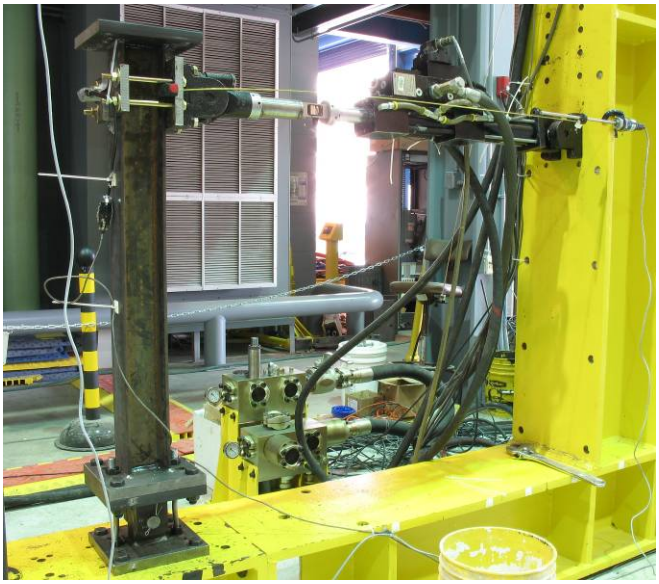

Figure 3-4 Single-degree-of-freedom experimental setup.

3.4 *HYBRID SIMULATION MODEL*

A hybrid simulation model performs the numerical integration and signal generation tasks. The inputs to a hybrid simulation model are raw measurements from the experimental setup, and the outputs are the experiment command signals. All inputs and outputs to this model are sampled at the experiment measurement and command update rate, 1024 Hz. Hence, the base interrupt rate of the numerical simulation model has to be 1024 Hz (sampling period $\delta t = 1/1024$ s).

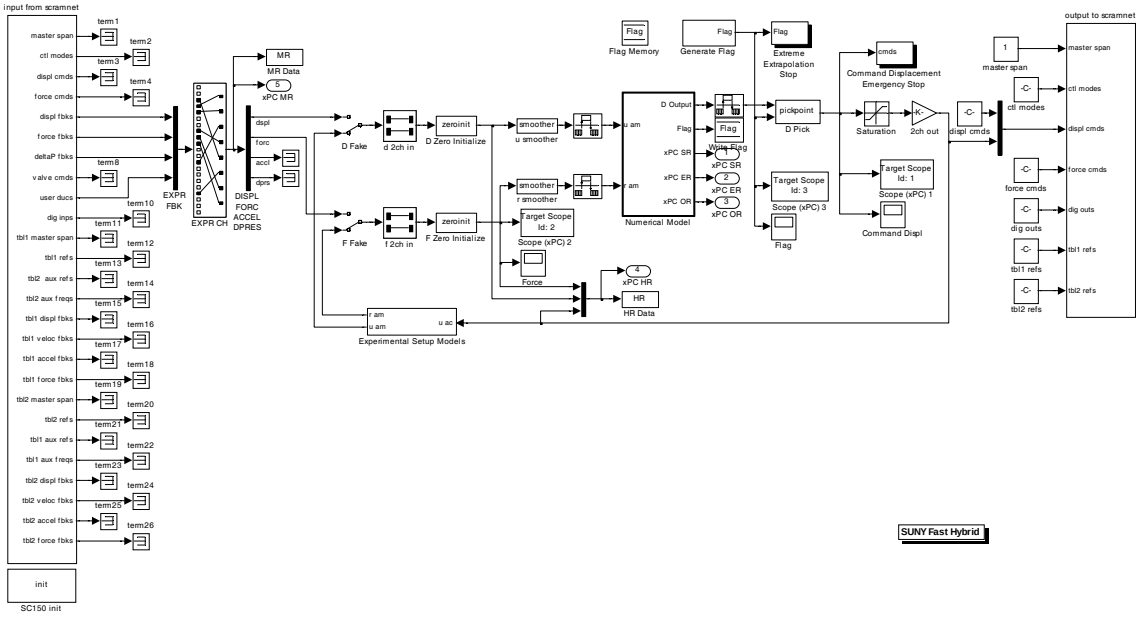

Figure 3-5 Combined Simulink model for numerical simulation and compensation.

A typical numerical simulation model is shown in Figure 3-5. This model includes the numerical model, input signal conditioners, output signal generators, data storage and visualization blocks, as well as numerical models of experimental setups for purely numerical simulations. It is shown that all the measurements and commands are read

from or written to the SCRAMNet blocks. The details of most of these components will be further described in the following sections.

As illustrated in Figure 3-5, the *numerical model* subsystem (the core of the simulation model) is located between input and output rate transition blocks. This shows the multitasking nature of the model, allowing for uninterrupted command signal generation task, which is of higher priority to ensure simulation continuity. The numerical simulation task is normally carried out at a rate slower than command update rate to improve computational efficiency for relatively large numerical simulations. The rate transition blocks ensure data integrity throughout the model by applying zero-order-hold to the signals to be sampled at lower rates, and double-buffering to the signals to be sampled at higher rates.

The proper execution of numerical simulation and signal generation tasks is ensured by using a *flag* variable. Depending on the integration time step Δt (normally integer multiples of command update time step δt), *flag* variable is increased by one each time the command signal is updated and decreased by the number of sub-steps, each time the simulation task is completed. The number of sub-steps is the number of command updates that should ideally occur during each integration time step:

$$n_s = \frac{\Delta t}{\delta t} \tag{3.1}$$

The *flag* variable is a positive integer, and will not exceed n_s unless the numerical simulation task fails to complete within one sampling period. In that case, larger-than-n_s values of *flag* indicate of the necessity of command signal extrapolation until simulation

results become available [19]. A faster computer is recommended when the flag variable tends to increase beyond $2n_s$.

In order to increase the computational efficiency of the numerical simulation and signal generation tasks, these can be separated on two real-time computers, as shown in the hardware architecture of Figure 3-1. Dedication of separate processors to these tasks may allow for the use of more sophisticated simulation and compensation procedures. The separate models for signal generation and numerical simulation are called *Controller* and *Simulator*, respectively.

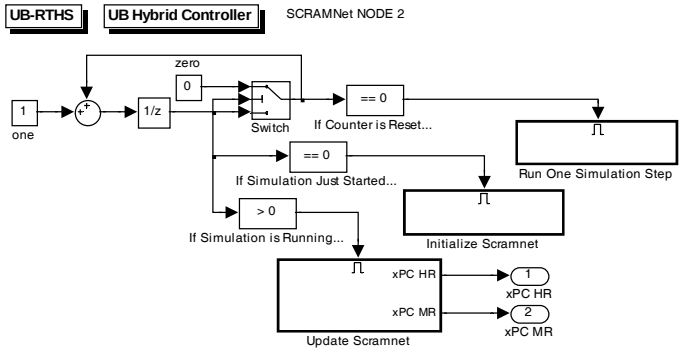

Figure 3-6 *Controller* Simulink model.

3.4.1 CONTROLLER

As the name suggests, the *Controller* model handles the experimental control at the sampling and command update rate, 1024 Hz. The model shown in Figure 3-6 shows that *Controller* takes appropriate actions during each timer interrupt based on a *counter* variable. *Counter* is initialized with zero at the beginning of a simulation, triggering an initialization of the SCRAMNet. Then, this variable is constantly increased with each execution of *Controller* model until it reaches the number of sub-steps, n_s, at which time it is reset to zero. Only when the *counter* is reset, the *Controller* runs a simulation step by

triggering the other hybrid simulation model, the *Simulator*. In any case, the *Update SCRAMNet* subsystem is executed, which handles the measurement signal conditioning and command signal generation tasks, as shown in Figure 3-7.

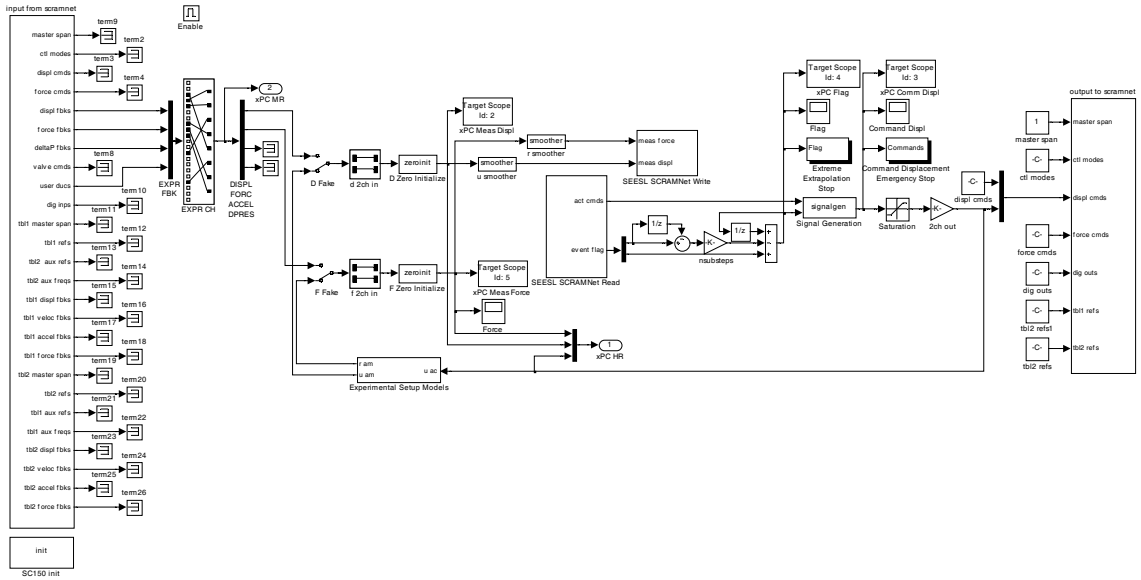

Figure 3-7 ***Update SCRAMNet*** **subsystem of Controller Simulink model.**

Figure 3-7 shows that the major difference of this subsystem with the combined simulation model shown in Figure 3-5 is that the *numerical simulation* subsystem is replaced by SCRAMNet read and write blocks, which can be accessed by *Simulator*. The *Controller* model contains all procedures for the tasks that have to be done at the experiment command update rate. These tasks include zero initialization and smoothing of measurements, signal generation, and numerical simulation of experimental setup. In addition, the selection of communication channels should be carried out by proper connections of shared memory blocks to the input and output blocks of this model. There are also safety measures that stop the simulation when *flag* variable becomes too large, or command displacement exceeds the actuator stroke. A saturation block also

prevents the submission of large displacement commands to the command displacement memory block.

3.4.2 SIMULATOR

The *Simulator* model is shown in Figure 3-8. As mentioned in the previous section, the execution of the simulator block is triggered when *counter* variable in *Controller* is reset to zero. Hence, *Simulator* may have a slower execution rate than that of *Controller*. Figure 3-8 shows that the *simulation* block is essentially idle until the trigger signal reaches unity. This trigger signal can be taken from a memory block called *master span* that shows the status of servo-hydraulic actuators. Alternatively, this signal can be set by a timer that delays the application of ground excitation by a user-specified time period. The former is suitable for random or harmonic excitations with infinite length, and the latter is useful for earthquake excitations, where the ground motion is of a finite length and simulation should stop after that. In this case, the initial excitation delay should be enough to allow the actuators master span to reach 100%.

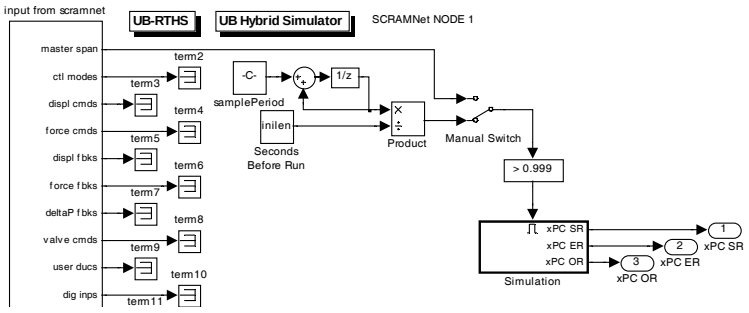

Figure 3-8 ***Simulator*** **Simulink model.**

The *simulation* subsystem of *Simulator* model, which essentially has the same components as the *numerical model* subsystem of Figure 3-5, is detailed in Figure 3-9. This

is the most important component of a hybrid simulation model, and contains blocks for numerical simulation, identification of system dynamics, signal correction and command preparation, as well as data streaming and visualization. The model communicates with the same memory partitions that are shown in Figure 3-7.

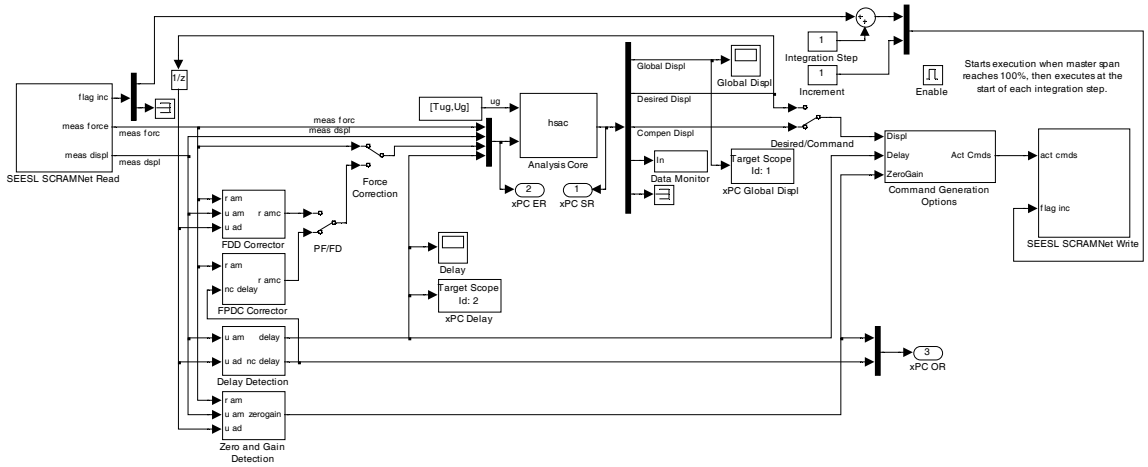

Figure 3-9 ***Simulation*** **subsystem of Simulator Simulink model.**

Like most of the blocks shown in Figure 3-9, the *analysis core* subsystem contains a Level-2 C S-Function[1] that carry out the numerical integration of test structure. The inputs to this block include measured forces and displacements, corrected forces, and actuator delays. Each of these signals is a vector of size N_{ACT}, which is the number of actuators involved in the simulation. The corrected force signal can be calculated using either of the force correction methods described in Chapter 6. As the name suggests, *delay detection* block estimates the delay of each dynamic actuator by comparing the

[1] A C S-Function is a MATLAB Standard Function written in C programming language. Level-2 C S-Functions are compatible with Simulink version 2.2 or later.

corresponding desired and measured displacement signals (Chapter 5). Another block determines the variable gain that should be applied to the command signal for reducing the undershooting of displacements in displacement reversals (Chapter 6).

The output signals of the *analysis core* are global displacements (a vector of size N, the number of degrees of freedom of test structure), actuator desired and command displacements, cumulative energy components, normalized energy error (Chapter 5), hybrid simulation error monitor [39], and other diagnostic outputs. The desired displacement signal is used only when delay compensation is performed outside of *analysis core* using polynomial extrapolation. This can be selected through the appropriate switches in *signal generation options* block, which also contains procedures for the application of the variable command gain. The existence of switches in the presented simulation models allows for quick selection of identification and compensation procedures, as shown in Figure 3-9.

3.5 A GUIDE TO SEESL PSEUDO-DYNAMIC HYBRID SIMULATION

In this section, the steps towards a pseudo-dynamic experiment using the available test procedures are outlined from an engineering standpoint. That is, the details of hybrid simulation operations, such as compensation methods and integration algorithms, are not discussed. Instead, only the minimum required information, data format, test preparation and execution, and result retrieval methods are presented. The currently available procedures allow for analysis of linear numerical substructures; hence, proper functions need to be developed for nonlinear systems. For this reason, a

brief discussion about possible extensions of current simulation models is provided for new and improved test procedures.

3.5.1 TEST STRUCTURE INFORMATION

The first step in performing a hybrid simulation is the selection of the test structure and its experimental substructure. The basic required information includes:

- Number of degrees of freedom N,
- Number of experimental degrees of freedom (number of actuators) N_{ACT},
- Stiffness matrix of the numerical substructure $\mathbf{K}$ $(N \times N)$,
- Total mass matrix $\mathbf{M}^{t}$,
- Mass matrix of the numerical substructure $\mathbf{M}$,
- Inherent damping ratio of the test substructure ξ,
- Numerical damping matrix $\mathbf{C}$ (for supplemental damping devices, if any),
- Dynamic influence vector $\boldsymbol{\iota}$ $(N \times 1)$,
- Transformation matrix $\mathbf{T}$ for displacements from global coordinate system to actuator coordinate system $(N_{\mathrm{ACT}} \times N)$,
- Command displacement factor,
- Measured force factor,
- Earthquake record and its scale factor.

Additional information may be required for specific integration procedures; initial $N_{\mathrm{ACT}} \times N_{\mathrm{ACT}}$ stiffness matrix $\mathbf{K}^{\mathrm{e}}$ of experimental substructure for operator-splitting integration method, and $N_{\mathrm{PAR}} \times N_{\mathrm{ACT}}$ displacement transformation matrix $\mathbf{T}_{\mathrm{p}}$ from actuator coordinate system to intrinsic (parameter) coordinate system, where N_{PAR} is the number of stiffness parameters to be estimated in each integration step. Details of these integration methods are presented in Chapters 7 and 8.

3.5.2 PRELIMINARY ANALYSIS

Similar to any other experimental study, it is necessary to perform pretest analyses to ensure proper structural response before the physical experiment. Any structural analysis program can be used for this purpose, and linear modeling of experimental substructure is normally sufficient. For this purpose, a rough numerical model of the experimental substructure is required based on its initial properties.

The use of numerical models of experimental setup presented in Section 3.6 is very effective for anticipation of expected force and displacement levels, as well as the required measurement quality. Using those models, the simulation data can be used for physical experiment without any change, which is useful in reduction of preparation tasks necessary for the physical experiment.

3.5.3 DATA PREPARATION

In order to initialize and execute a hybrid simulation, the test structure data needs to be in a format usable by the hybrid simulation models. This can be done through a MATLAB m-file for initialization of structural properties. A sample file for initialization

of simulation model is shown in Figure 3-10 for a 4-story structure with 2 experimental degrees of freedom (Figure 3-11). This file essentially contains the information required in Section 3.5.1, and should be saved in the same directory as the hybrid simulation models. The file name should then be specified in the initialization module of the hybrid simulation model, aci.m. This file also contains options for selection of earthquake excitation and scale factor, as well as analysis options such as time step, integration method, convergence tolerance, and so on.

```
% *** General information
NDOF=4; % number of degrees of freedom
NACT=2; % number of actuators involved in the simulation

% *** Numerical model data
MT = [7 0 0 0; 0 5 0 0; 0 0 3 0; 0 0 0 1]*1.25/g; % Total mass matrix
K = [30 -12 0 0; -12 20 -8 0; 0 -8 12 -4; 0 0 -4 4]; % Global analytical stiffness
C=zeros(NDOF,NDOF); % Global analytical damping
dr=0.05; % Damping ratio (stiffness proportional)
L=-MT*ones(NDOF,1); % Influence vector
% Coordinate system transformations
TDGA=[-1 1 0 0; -1 0 1 0]; % Displacement from global to actuator cs
TDAP=[1/l1 0; -l/l1/l2 1/l2]; % Actuator displacements to parameter cs
FDGA=1; % Displacement factor from global to actuator coordinates
FFAG=1; % Force factor from actuator to global coordinates

% *** Basic experimental setup data
ME=[0 0 0 0; 0 0.05 0 0; 0 0 0.025 0; 0 0 0 0]/g; % Experimental mass matrix

% *** Additional experimental setup data
    % 1. For explicit-implicit methods, the following data are not required.
    % 2. For operator-splitting methods, TDAP'*KEP*TDAP is the initial
    % stiffness of experimental substructure. Define TDAP as identity,
    % and KEP as the experimental stiffness matrix in actuator coordinates.
    % 3. For model-based integration, KEP is the parametric experimental
    % stiffness in actuator coordinate system (initial), and TDAP is the
    % transformation from actuator to intrinsic coordinate system.
k1 = 5.543*2;   % DOF 1    STORY 1 (two pairs of coupons)
k2 = 3.89;      % DOF 2    STORY 2
l1=43; % length of first story (in)
l2=46; % length of second story (in)
l=l1+l2;
NPAR=2; % number of important parameters for formation of stiffness matrix
KEP = [k1*l1^2 0; 0 k2*l2^2]; % Parametric experimental stiffness in intrinsic
coordinates system
TDAP=[1/l1 0; -l/l1/l2 1/l2]; % Displacements from actuator to parameter cs
```

Figure 3-10 Sample experimental setup data for initialization of hybrid simulation models.

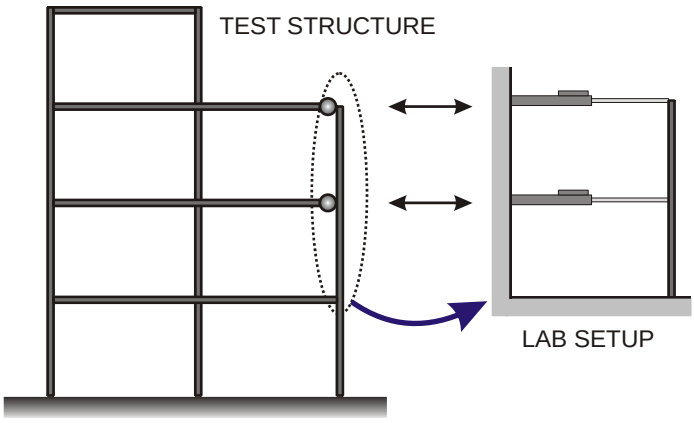

Figure 3-11 A 4-story structure and its corresponding experimental substructure.

If this file is also used for pre-test numerical simulations using numerical models of Section 3.6, the properties of the experimental substructure should be provided as well. This information depends on the numerical model selected for simulation of experimental setup as described in Section 3.6.

3.5.4 INITIALIZATION AND LOADING OF SIMULATION MODELS

After preparation of test structure information in a format usable for the hybrid simulation model, the model should be initialized and loaded into local memory. It is recommended that a MATLAB clear command should be executed before this step, to ensure proper initialization of all variables. The loading can then be done by running MATLAB files initialize.m and aci.m, which initialize the SCRAMNet partitions and numerical model, respectively. This procedure is automated in another m-file, ml.m.

3.5.5 DOWNLOADING MODELS TO XPC TARGETS

For real-time execution, the simulation models should be downloaded to xPC Target computers. At this point, it is recommended to save the workspace to keep track

of the simulation parameters and model properties as a part of the results. MATLAB rtwbuild command generates the C code for the simulation model, makes the executable file, and downloads it to the specified xPC Target. It should be noted that although Simulink supports the use of m S-Functions, as of version R2006b, all functions have to be written in C programming language for real-time execution.

Since the structural properties are specified during the initialization of the simulation models, any change in these properties should be followed by a fresh download of the model to xPC Targets. This procedure is automated in an m-file named xb.m.

3.5.6 TEST EXECUTION

Running the simulation on host PC is possible through start command from simulation menu. This is useful for non-real-time numerical simulations, in which physical substructures are not involved. Models that are downloaded to xPC can be executed by target.start command, where target is the user-defined name of the xPC Target computer. If the simulation is of finite length, it will stop at the end of the specified period, otherwise it should be stopped using target.stop command. For separate simulator/controller models described in Section 3.4, these commands should be executed for each of the targets. In order to simplify this process, the necessary commands are automated in m-files xr.m and xs.m for running and stopping the simulation, respectively.

3.5.7 RETRIEVAL AND PROCESSING OF RESULTS

After the simulation is completed, the results should be transferred to host PC for further analysis and processing. Depending on the selected data storage option, the results can be saved on xPC hard disk and then transferred to host using file transfer protocol (FTP) via LAN, or they can stream to host during the simulation. The transfer is carried out through LAN, which may not be fast enough for real-time data visualization on host PC. However it is sufficient for analysis of the results after the experiment, and does not have the 100-signal limit of the first data storage option on hard disk.

The streamed data can be accessed through target.outputlog array, which contains all the signals sent to model output ports in order of the port number. The timer is also stored in target.timelog array. An automated script named xpcretr.m retrieves the signals from target logs and stores the result in a MATLAB mat file.

The basic processing of the results, including plots of global displacements, command and measured displacements, measured and corrected forces, delay, energies, hysteretic behavior of experimental specimen, online estimated parameters and error indicators is available through an m-file named procexp.m. This script also calculates the final energy error and percentage of steps with specific properties, such as successful implicit iterations of stiffness corrections (see Chapters 6, 7 and 8 for more information).

3.5.8 MODEL MODIFICATIONS AND CUSTOMIZED SIMULATION SUBROUTINES

All the procedures that are currently available in the SEESL hybrid simulation models are designated for linear numerical substructure, although the formulations are

directly extendable to fully nonlinear systems. Hence, further development of integration procedures is necessary for application to nonlinear numerical systems. The majority of these procedures are located in the hybrid simulation analysis core file hsac.c, which is used by the block named *analysis core* (Figure 3-9). This function follows the MATLAB standard function format, which should be maintained throughout the modifications. Further, to avoid significant changes in the simulation models, the input and output structure of the blocks should remain intact.

The same considerations should be followed for development of other procedures of the simulation model, such as delay estimation and compensation, and signal generation and conditioning. In addition, all automated procedures described in the preceding sections should also be adapted to comply with the modified simulation models.

3.6 COMPUTER SIMULATIONS OF HYBRID EXPERIMENTS

Numerical models of hybrid simulation systems have significant advantages for performing low-cost computer simulations. These numerical evaluations are useful for pre-test analyses mentioned in the previous section, as well as initial investigations for the development of new test procedures. One important characteristic of these models for performance evaluation of new procedures is that the "exact" solution can be obtained for the simulation by simply eliminating all uncertainty sources of the experimental model. A direct comparison of the results will then give a measure of the performance of the procedure under evaluation. In addition, the properties of the

"experimental setup" model can be easily altered in a numerical simulation to conduct further studies that may be very costly using an actual experimental setup in the laboratory.

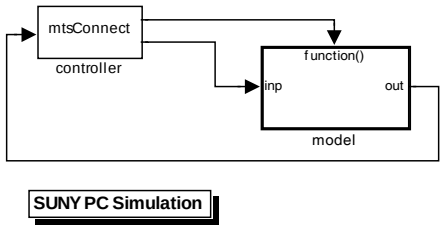

Figure 3-12 Simulink model for computer simulation of SEESL online hybrid simulation system.

The Simulink model shown in Figure 3-12 is the *PC Simulation* model designed for SEESL hybrid simulation system [87]. As illustrated, this model consists of *controller* and *model* blocks. The *controller* block is a subsystem for communication with the STS[1] software (in simulation mode) shown as a test controller in Figure 3-1. The *model* block, detailed in Figure 3-13, consists of actuator and specimen models, along with the *hybrid simulation* subsystem. The *hybrid simulation* subsystem gets the measurements and calculates the command signals. The contents of this subsystem are essentially similar to the *simulation* subsystem shown in Figure 3-9.

The details of a *dynamic channel* subsystem are shown in Figure 3-14. This subsystem receives the valve command from the STS, and calculates the piston displacement. The configuration of *actuator/valve* and *payload* blocks shown in this subsystem demonstrates that the piston displacement is actually the reaction of the attached specimen to the

[1] Structural Test System by MTS Systems, Inc.

actuator force. It should be noted that in Figure 3-14 the force measurement is calculated based on differential pressure on actuator piston, while in practice, load cells provide a more accurate estimate of the applied force. More detail about the development of actuator force as a result of valve command is described in Chapter 4.

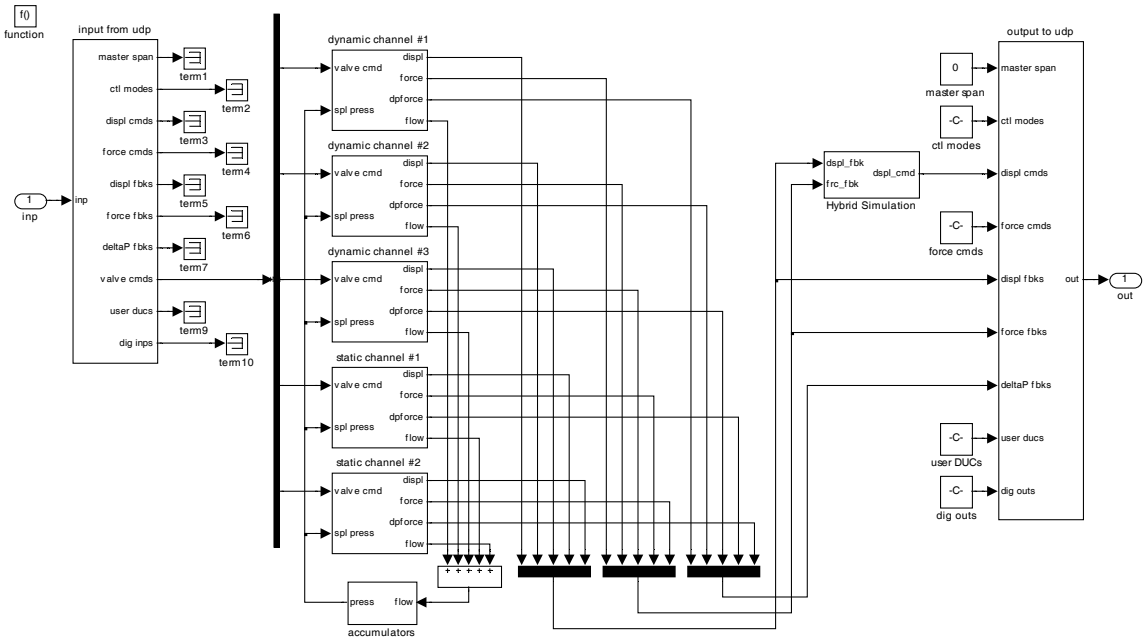

Figure 3-13 ***Model*** **subsystem of PC simulation Simulink model.**

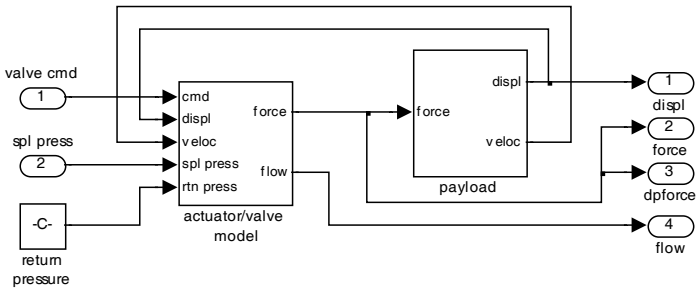

Figure 3-14 Detail of each ***channel*** **subsystem within model subsystem.**

Figure 3-15 shows the contents of a *payload* subsystem, which simulates the displacement response of a bilinear specimen to the force applied by the actuator. It is illustrated that the specimen restoring force may consist of inertia, strain and friction

forces. A viscous damping force can also be added by multiplying the velocity signal to a damping coefficient and subtracting the result form applied force. In addition, the bilinear model can be replaced by other hysteretic models that better simulate the response of the existing experimental setup.

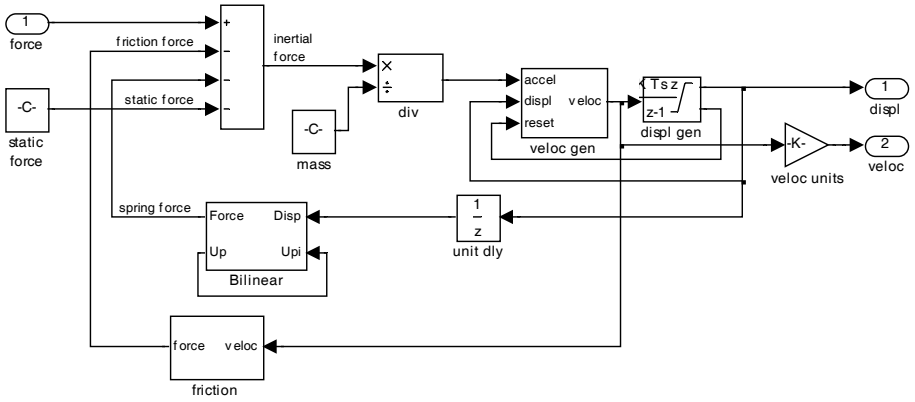

Figure 3-15 ***Payload*** **subsystem with bilinear specimen.**

Instead of using complex servo-hydraulic actuator models that generally require many parameters, simplified models of experimental setup can also be used for numerical simulations [39]. These models have been integrated into the hybrid simulation models shown in Figures 3-5 and 3-7.

The simplified model shown in Figure 3-16 simulates the response of a SDF experimental setup. The restoring force can be determined by any stiffness model, ranging from a linear one to complex and/or deteriorating inelastic models, such as those proposed by Sivaselvan and Reinhorn [88] or Bouc [89] and Wen [90]. In the present simulation model, a Bouc-Wen stiffness model is utilized, which is governed by the following relation:

$$r(t) = \alpha k u(t) + (1-\alpha) k u_y z(t) \tag{3.2}$$

where $z(t)$ can be calculated by solving:

$$\dot{z}(t) = \frac{1}{u_y}\left[\dot{u}(t) - \gamma\left|u(t)\right| z(t)\left|z(t)\right|^{n-1} - \beta\dot{u}(t)\left|z(t)\right|^{n}\right] \tag{3.3}$$

In the above relations, α is the post-yield stiffness ratio to the initial stiffness, β and γ govern the shape of hysteretic loops by altering yield initiation point and elastic recovery sharpness, and n modifies the rate of elastic-to-plastic transition of the model.

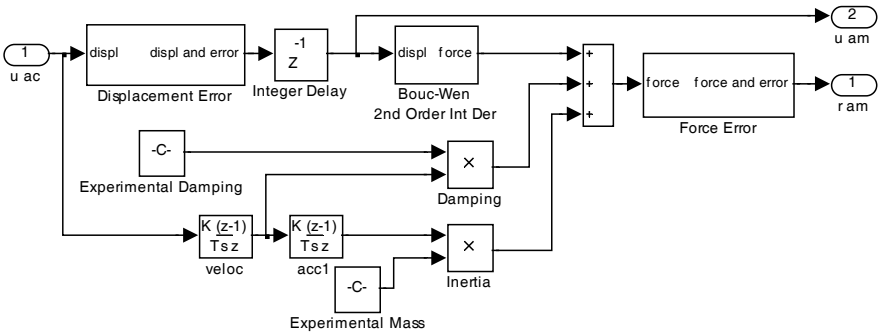

Figure 3-16 Simplified Simulink model for computer simulation of a SDF experimental substructure.

In order to include the dynamics of the experimental setup and measurement noise in the simplified models, a controlled error signal consisting of random and systematic error components is added to the displacement prior to being applied to the experimental model (Figure 3-17). The systematic error is generated by a normally distributed random number multiplied by the displacement increment, where the displacement increment serves as a first order approximation to velocity. The resulting errors are systematic with direction and magnitude proportional to velocity. Particularly, if the mean of the random number is less than zero, the added error signal will result in an apparent lag. Alternatively, the delay can be introduced by the *integer delay* block of Figure 3-16 as multiples of sampling period.

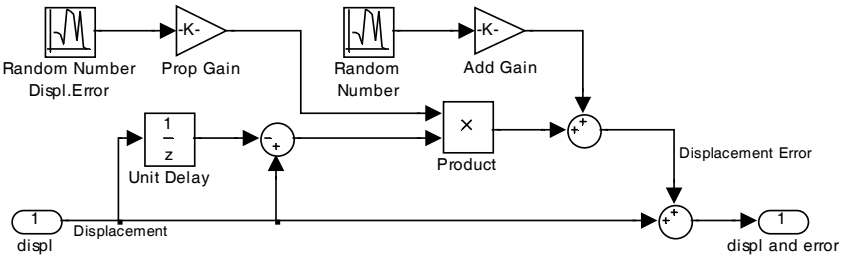

Figure 3-17 Simulink model for artificial introduction of systematic and random errors in displacement signal.

In addition to the velocity-dependent systematic error, a second source from a normally distributed random signal is added to the error signal to mimic the random noise contamination in the simulated measurements. A similar approach is followed for introduction of experimental errors and noise in the force signal. This simple model, which has shown to provide reasonable results when compared to experimentally measured errors [39], is widely used in the numerical studies presented in this dissertation. The actuator delay and measurement noise simulated in the model have been calibrated to actual experimental data and laboratory equipment information.

The simplified simulation models only require a few parameters, and can be easily modified for more complex and MDF experimental substructures. For instance, the current two-degree-of-freedom experimental setup at SEESL facilities (Figure 3-3) can be numerically modeled using the Simulink block diagram illustrated in Figure 3-18. In this model two SDF experimental models introduce delay and measurement noise in displacement and force signals. The experimental restoring force is governed by two Bouc-Wen stiffness models that are coupled using appropriate force and displacement coefficients based on the geometric properties of the experimental setup. The parameters of the Bouc-Wen model can be altered to achieve the desired experimental response.

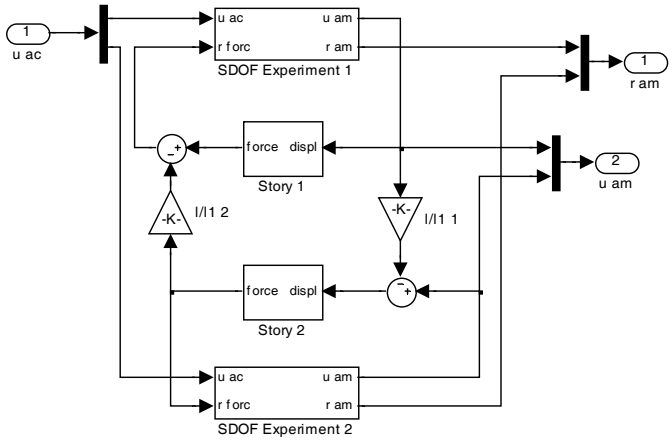

Figure 3-18 Simulink model for computer simulation of two-degree-of-freedom experimental setup at SEESL.

4 SERVO-HYDRAULIC ACTUATORS AND CONTROL TECHNIQUES

In hybrid simulations, servo-hydraulic actuators are widely used for the application of displacements and forces to the experimental substructures. As described in Chapter 1, feedback systems such as hybrid simulation are very sensitive to measurement and control errors that can propagate through the simulation and result in inaccurate or even unstable simulations. Actuator performance plays a central role in this regard, as the actuator tracking errors are known to be significant contributors to experimental errors in a hybrid simulation. An accurate understanding of the behavior of servo-hydraulic systems is essential for reduction of these errors through proper design of control systems and compensation procedures. In this chapter, the principles governing the functions of servo-hydraulic actuators are briefly presented and numerical models are described. The most common mechanical control techniques for servo-hydraulic actuators are also reviewed in this chapter.

4.1 MECHANICAL PRINCIPLES

The major components of servo-hydraulic actuators are a servovalve, a cylinder and a piston that slides in the cylinder. The servovalve controls the flow of the fluid from accumulators to actuator cylinders for driving the piston. An overall sketch of servovalve and actuator combination is shown in Figure 4-1. In summary, the input electric current to the servovalve results in the rotation of an electromagnetic flapper mechanism that displaces the spool from its neutral position. As a result of spool movement, the orifice sizes change and alter the fluid flow to the actuator cylinders. The resulting pressure difference between the two sides of the piston, known as the load pressure, drives the actuator piston [91].

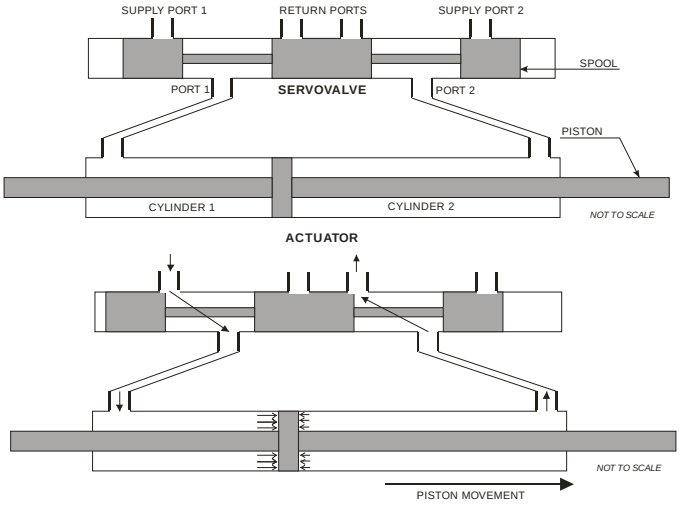

Figure 4-1 A schematic view of a servovalve actuator.

4.2 ORIFICE FLOW, PRESSURE AND FORCE RELATIONS

The details of the functions of a servo-hydraulic actuator are described in this section, and the governing physical relations are provided.

For turbulent flows, which hold for most of valve openings except for very small ones, the flow from side i to side j of an orifice is given by:

$$q_{ij} = K_v A_{ij} \sqrt{p_i - p_j} \tag{4.1}$$

in which p_i and p_j are the pressures upstream and downstream of the orifice, respectively. A_{ij} is the orifice area, and K_v is flow coefficient depending on the orifice geometry and fluid density. The flow is therefore dependent on the orifice area and geometry, which vary with the spool position, x. This relation is often linearized by substituting a truncated Taylor expansion of the square root term in Equation (4.1). This simplification therefore becomes more inaccurate with increment of spool displacement from its zero position.

Orifice flow coefficients with respect to the spool position can also be described by a relation such as that used in MTS PC Simulation models [87] developed in Simulink [82] environment:

$$K_v = \frac{q_n}{\sqrt{\frac{p_r}{2}\left[1 + A_{ij}^2 \cdot \left(q_n^2 / q_r^2 - 1\right)\right]}} \tag{4.2}$$

in which q_n and q_r are nominal and rated flows of the actuator ports, respectively, and p_r is the rated oil pressure. The port flow entering into each actuator cylinder is then determined by subtracting the corresponding servovalve chamber return flow from supply flow:

$$q_i = q_{\text{supply } i} - q_{\text{return } i} \tag{4.3}$$

in which i is the actuator cylinder or servovalve chamber number.

Many servovalves, including Moog servovalves [92] that are widely used in MTS actuators [84], are manufactured with symmetrical orifices, and the orifice areas are the same at any moment. The area varies linearly from zero at spool zero position (which generally includes an overlap) to a maximum when orifices are fully open. However, the flow may not be exactly zero even when the orifices are fully shut, as leakage flow occurs, and reaches its maximum when valves are closed. Williams *et al.* [26] introduced a nonlinear function to capture this behavior. At zero position, when the leakage flow is maximum, the orifice areas are assumed to be nonzero, which results in two equal leakage flows from supply to servovalve and then to return chambers on each side of servovalve. As the flows were assumed to be equal, equal pressure drops should then exist, and the pressure inside the actuator and servovalve will be:

$$p_{\text{closed}} = \frac{p_{\text{supply}} + p_{\text{return}}}{2} \tag{4.4}$$

Therefore, the leakage area when the spool is at zero position is given by:

$$A_{\substack{\text{leakage} \\ \text{closed}}} = \frac{q_{\text{max leak}}}{K_v} \sqrt{\frac{1}{2\left(p_{\text{supply}} - p_{\text{return}}\right)}} \tag{4.5}$$

where $q_{\text{max leak}}$ is the maximum leakage flow, which occurs when the valves are fully closed, and should be measured experimentally. Williams *et al.* then admitted a piecewise linear function to describe the overall orifice area as a function of spool displacement (Figure 4-2).

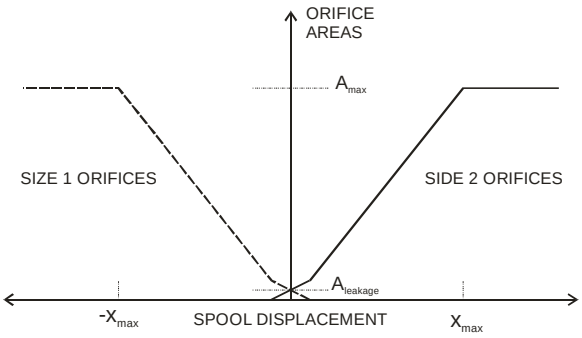

Figure 4-2 Orifice area as a piecewise linear function of spool displacement [26].

In MTS PC Simulation models, the leakage flow is simply related to the differential pressure of the orifice sides by a leakage factor:

$$q_{\text{leak}} = c_l \left(p_{C1} - p_{C2} \right) \tag{4.6}$$

in which p_{C1} and p_{C2} are oil pressures in servovalve chambers 1 and 2, respectively, and c_l is the experimentally-calibrated leakage factor (flow/pressure).

In addition to leakage, the compressibility of fluid results in a volume change of oil that can be considered as a virtual compressibility flow. If the kinematic flow is assumed to be multiples of cylinder internal cross-sectional area and piston velocity, as in:

$$q_{\text{kin } i} = A_i v_{\text{piston}} \tag{4.7}$$

then the compressibility flow is simply the result of subtraction of leakage and kinematic flows from the port flow:

$$q_{\text{comp } i} = q_i - q_{\text{kin } i} - q_{\text{leak } i} \tag{4.8}$$

This compression flow can then be converted to oil velocity by dividing it by cylinder area. Finally, oil displacement due to compression can be determined by integration of velocity:

$$v_{\text{comp } i} = {q_{\text{comp } i}}/{A_i} \tag{4.9}$$

$$d_{\text{comp } i}(t+\delta t) = d_{\text{comp } i}(t) + v_{\text{comp } i}(t)\cdot \delta t \tag{4.10}$$

This virtual oil displacement can then be used to determine the oil pressure in actuator cylinder:

$$p_i = \frac{K_{\text{oil } i}\cdot d_{\text{comp } i} + C_{\text{oil}}\cdot v_{\text{comp } i}}{A_i} \tag{4.11}$$

In the above equation, C_{oil} is oil damping coefficient, and $K_{\text{oil } i}$ is the axial stiffness of oil column in cylinder i. The damping coefficient has been used to compensate the reduced damping observed in simulations, compared to actual behavior of physically-tested actuators. This damping parameter is multiples of a given damping ratio, mass and oil column radial frequency. The stiffness of oil column can be determined by:

$$K_{\text{oil } i} = \frac{K_{\text{bulk}} A_i}{L_i} \tag{4.12}$$

in which K_{bulk} is the bulk modulus of oil, and L_i is the oil column length, which is equal to the sum of cylinder and end cushion lengths.

The piston force therefore develops as a result of pressure difference in actuator cylinders:

$$f = p_1 A_1 - p_2 A_2 \tag{4.13}$$

This force is applied on the specimen, and depending on its stiffness, mass and damping, piston displacement occurs (see Section 3.6 for models showing interaction of

actuator and test specimen). The control system then compares this displacement to the command displacement and issues the next valve command.

4.3 NUMERICAL MODELING OF SERVO-HYDRAULIC ACTUATORS

Early models of actuators have adopted the linearized form of system dynamics, such as those proposed by Merritt [93] and Stringer [94]. In these approximate models displacement of the servovalve spool is assumed to be a linear function of drive current, the linearized form of orifice flow relation is utilized, the effects of leakage flows are often simplified as additional damping, and the effect of piston movement on the cylinder volumes are not taken into account. These models do not consider the complex nonlinearities due to the torque motor dynamics, flow forces on the flapper and spool, dynamics of the armature/flapper assembly, spool dynamics, and oil compressibility.

Sixth order models incorporating the above-mentioned nonlinear effects have been proposed by Nikiforuk [95] and Wang [96] and compared with experimental results. They demonstrated that all nonlinear phenomena can be neglected without loss of accuracy in the frequency range 0-100 Hz, except for the armature/flapper dynamics. This frequency range is well beyond those needed for structural simulations. For this reason, a second-order model was proposed by Williams *et al.* [26] that only included nonlinearities due to armature/flapper dynamics. As shown in Figure 4-3, this model converts the drive current to spool velocity, which can be integrated to give displacement. The servovalve model requires only two parameters, namely the gain K_{sv} and time constant, T_{sv}.

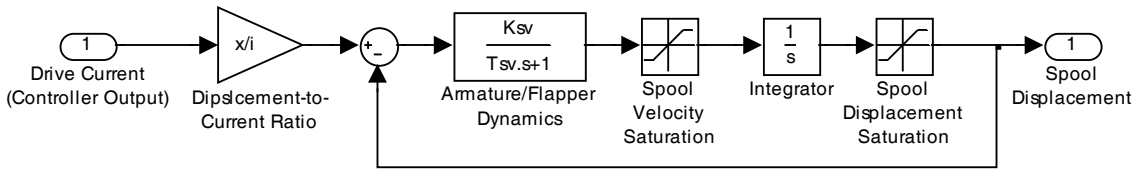

Figure 4-3 Block diagram of the servovalve model proposed by Williams *et al.* [26].

The MTS PC Simulation models use a simpler approach to consider the effect of servovalve on the response. In these models, the servovalve only introduces a delay in the control signal. The valve opening is otherwise linearly related to the control signal after removal of overlap displacement of spool. In this approach, an array of control signals is formed, and at each simulation step, this array moves forward by one element; the first element is the valve position, while the last one is the new control signal. The number of elements of this array is equal to the servovalve delay divided by the simulation step size.

Zhao *et al.* [30] provided another approach for consideration of three-stage servovalve dynamics by using a second order equation:

$$v = m_{ev}\ddot{x} + \beta_{ev}\dot{x} + k_{ev}x \tag{4.14}$$

in which x is spool position, v is the command voltage, and m_{ev}, β_{ev}, and k_{ev} are the equivalent mass, damping and stiffness of the servovalve, respectively. The authors stated that the second order dynamics is highly damped and rolls off at approximately $30\,\mathrm{Hz}$. In their numerical studies, however, they admitted a linear relation given by:

$$v = k_{ev}x \tag{4.15}$$

for a narrow frequency range of 0-10 Hz. Then the effect of amplitude reduction of higher frequencies was compensated using a first-order phase-lead network, whose transfer function is:

$$H_{LD} = \frac{T_d s + 1}{\alpha T_d s + 1} \tag{4.16}$$

in which T_d is a time constant and $\alpha = 0.1$.

Although many linear [93, 94, 97] and more complex nonlinear numerical models [26, 28-30, 95, 96] have been developed for systems involving servo-hydraulic actuators, the simpler linear models have been more widely used. This is primarily due to the complex dynamics of the system that may be sensitive to parameter uncertainties. In addition, the unidentified dynamic properties of test structures in hybrid simulations make it more difficult to establish a closed-form formulation for the entire test system. As a result, it is often attempted to develop test procedures that are independent of experimental setup dynamics.

4.4 CONTROL TECHNIQUES FOR SERVO-HYDRAULIC ACTUATORS

The servovalve command signal is generated by a controller that compares the position of actuator piston with the command displacement. A controller normally converts the command displacement to a voltage signal for the servovalve. In order to achieve an acceptable tracking performance, the controllers should be able to compensate the majority of system dynamics. In addition, the controllers should be

robust enough to make the actuators usable in a wide range of applications, such as test specimens with different static and dynamic properties.

In the design of mechanical control systems, existence of an analytical model of the servo-hydraulic system (the plant) is very beneficial. This analytical model is particularly useful for reduction of the effects of system dynamics. However, as mentioned in the previous section, a complete mathematical model of experimental setup is difficult to find due to the complex and diverse test system dynamics and unidentified properties of experimental substructure. For this reason, robust controllers have been the primary choice for control of servo-hydraulic actuators. These controllers often need to be tuned before the experiment for the intended range of application.

One of the most common controllers that are used with servo-hydraulic actuators are PID (proportional-integral-derivative) controllers. This type of controller is briefly studied in the following section. The discussion also includes its most common modifications, such as anti-integral windup, delta pressure stabilization, and feedforward control, which are made to enhance the tracking performance of the actuator.

4.4.1 PID CONTROLLERS

PID controllers are very common in mechanical engineering, due to the fact that they are able to control complex plants without the need for precise identification of their dynamics. The Controller compares a measured signal (actuator piston position) from a plant (the actuator) with the reference signal (command displacement). The

difference (the error signal) is then used to calculate a control signal value for the plant that brings the measured value of the plant back to its desired value. Unlike simpler control algorithms, the PID controller can adjust process outputs based on the history and rate of change of the error signal, which results in a more accurate and stable control. It can be shown mathematically that a PID control loop will produce accurate and stable control in cases where a simple proportional control would either have a steady-state error or would cause the process to become unstable. Unlike more complicated control algorithms based on optimal control theory, PID controllers do not require advanced mathematics to design and can be easily adjusted (tuned) to the desired application.

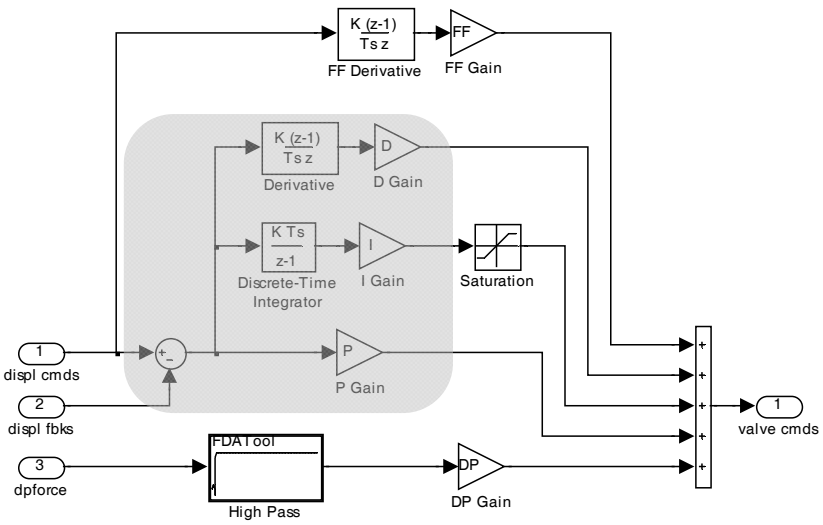

Figure 4-4 A Simulink model for PID controller with feedforward control and delta-pressure stabilizer.

4.4.1.1 Theory

The PID loop adds corrections to the control signal, removing errors from the controllable variable of the process, as shown in the shaded area of Figure 4-4 for the

displacement of an actuator. The error is calculated by subtracting the measured quantity (actuator piston position) from the reference value (command signal). The error signal is then used to generate a control signal adjustable by proportional, integral, and derivative gains. "PID" is named after its three correcting calculations, which add up and adjust the controlled quantity:

Proportional - To handle the present requirements, the error is multiplied by a constant P and added to the control signal. P is only valid in the band over which the controller output is proportional to the error of the system. When the error is zero, a proportional controller output is zero. The greater the proportional gain, the more the servovalve opens for a given error. As proportional gain is increased, the error decreases, resulting in a closer tracking of reference signal and reduced response time. On the other hand, increasing the gain decreases the stability margin of the system and increases the frequency of oscillation. If the proportional gain is set too high, unstable system operation and ringing can result. Consequently, proportional gain should be set as high as possible while maintaining stable system operation. Figure 4-5 shows the effect of proportional gain variation on the actuator response to a unit step input.

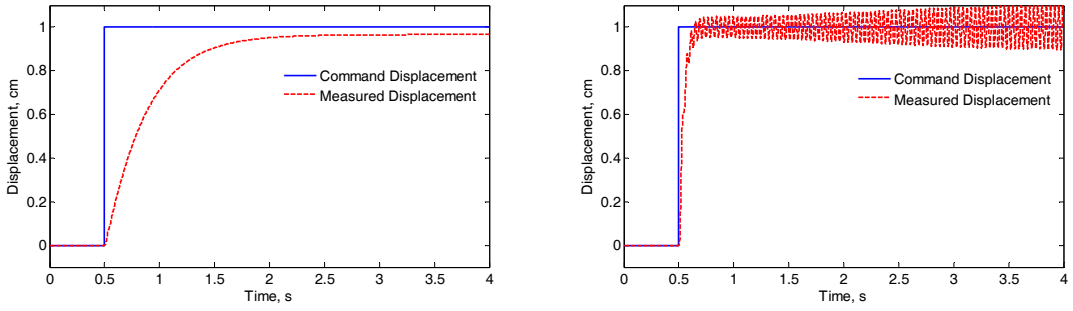

Figure 4-5 Effect of proportional gain on the actuator step response (only proportional control) – left: low gain, right: high gain.

Integral - To handle the past requirements, the error is integrated (added up) over a period of time, multiplied by a constant I, and added to the control signal. A simple proportional system may result in oscillatory response, as the overshooting errors are not removed by any other means. By adding a proportion of the average error from the plant input, the average difference between the process output and the reference value is always being reduced. As a result, the output of a plant controlled by a well-tuned PI loop will converge to the reference signal (zero steady-state error), leading to a reduced error between command and feedback. Integral adjustments affect system accuracy during static or low-frequency operation when the actuator cannot keep the commanded position (Figure 4-6).

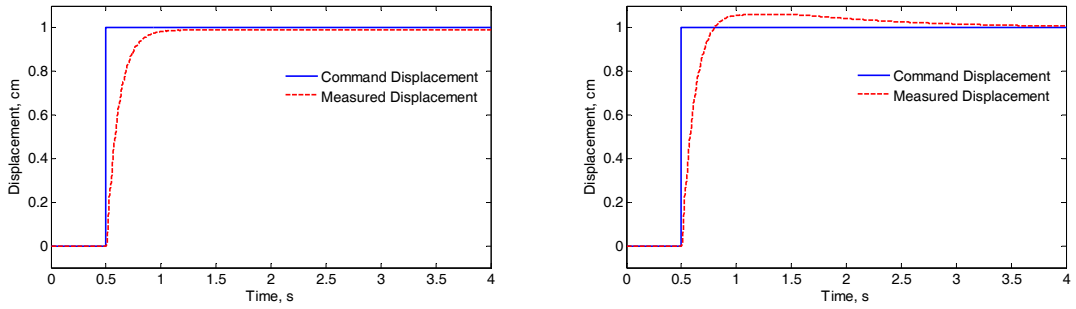

Figure 4-6 Effect of integral gain on the actuator step response (PI control) – left: zero gain, right: high gain.

Derivative - To handle the future requirements, the first derivative of the error (the slope of the error) over time is calculated, multiplied by a constant D, and added to the control signal. The derivative term controls the response to a change in the system. The larger the derivative term, the more rapidly the controller responds to changes in the process output. Its D term is the reason a PID loop is sometimes called a predictive controller. The D term is reduced when trying to dampen the controller response to

short term changes. Practical controllers for slow processes can even be utilized without D signal.

Derivative adjustments also affect the servo control loop dynamic stability as it reduces the overshoot at higher proportional gain settings. In addition, it can reduce the system bandwidth, closing the servovalve in anticipation of achieving the commanded position through the rate of change in feedback. Addition of derivative gain to a controller that has already been adjusted for proportional control results in oscillation effects shown in Figure 4-7.

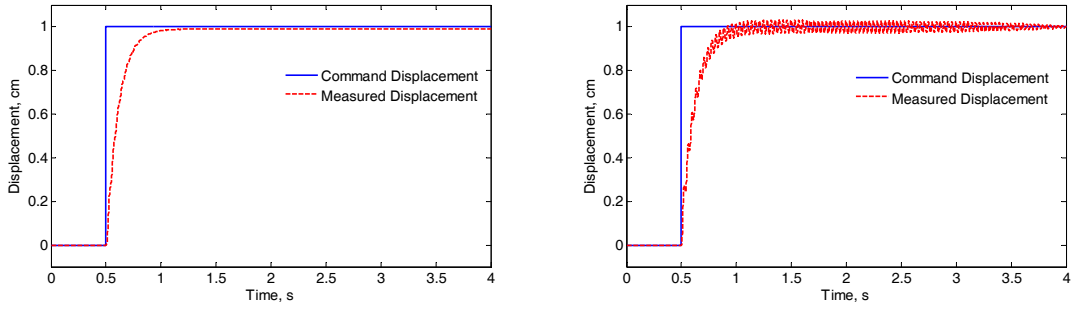

Figure 4-7 Effect of derivative gain on the actuator step response (PD control) – left: zero gain, right: high gain.

Transfer function - A PID loop can be characterized as a filter applied to a complex frequency-domain system. This form is commonly used for stability analysis of the process. The generic transfer function of a PID controller is as the following:

$$H(s) = \frac{Ds^2 + Ps + I}{s} \tag{4.17}$$

or sometimes, when P is used as a master gain, can be written as:

$$H(s) = P\frac{Ds^2 + s + I}{s} \tag{4.18}$$

In this configuration, the larger the proportional gain, the faster the response, as the control signal will be larger due to a given error signal with a higher master gain.

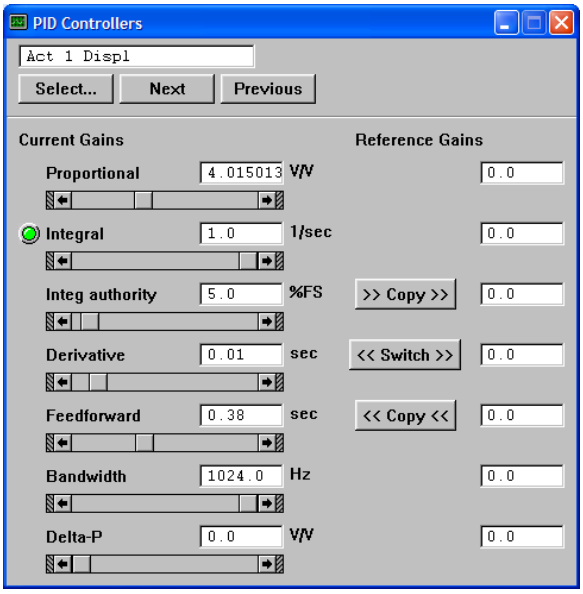

Figure 4-8 PID Controllers window of STS for adjusting control parameters [84].

In addition to the above components, the control signal may be filtered by a low-pass filter to remove the existing noises and improve the frequency content of valve command. The above-mentioned control gains can be adjusted through *PID Controllers* window of STS software, shown in Figure 4-8.

4.4.1.2 Anti-Integral Windup

One of the most common problems in using PID controller is integral windup. It may take too long for the output value to ramp up to the necessary value when the loop first starts up. For example, actuator may fail to achieve a target displacement by a large margin. This increases the integral share of the control signal, which can result in a large overshoot. Sometimes this can be fixed with a more aggressive differential term, or by preloading the control loop with an initial output. Another option is to disable the

integral function until the measured variable has entered the proportional band. That is, to avoid overcompensation of the integral component of control signal, the contribution of the integrated error to the control signal must be limited. This can be achieved using a saturation function as shown in Figure 4-4. In STS software, the integral term is limited by a percentage of full scale valve command, %FS. This is called the integral authority in STS controller parameters (Figure 4-8).

4.4.1.3 Feedforward Control

Feedforward control signal introduces a derivative of the command signal in the valve command. The feedforward component is used to minimize following errors or phase lag, which is of high importance in feedback systems. Feedforward control is especially useful because it does not affect the control loop stability. As illustrated in Figure 4-9, higher settings reduce following error during a sinusoidal command. However, extremely large values of this gain can result in a leading error, or large oscillations. Hence, this approach can only partially reduce the response lag of the servo-hydraulic system, and the remainder is often handled by using supplementary compensation procedures (Chapter 6).

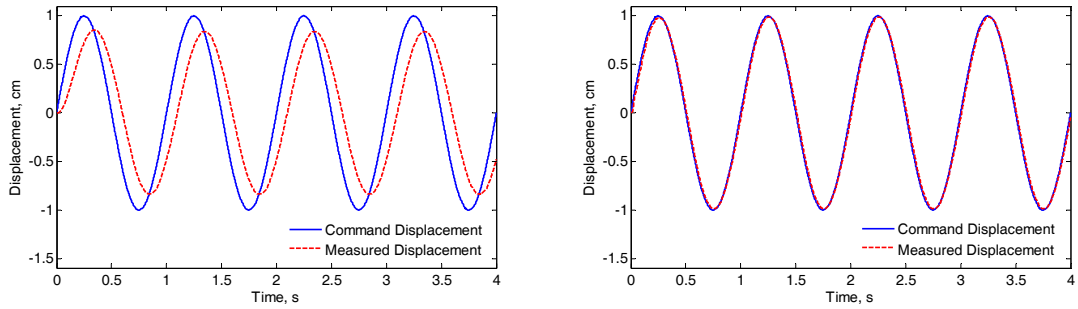

Figure 4-9 Effect of feedforward gain on the actuator sinusoidal response– left: zero gain, right: adjusted gain.

4.4.1.4 Delta-Pressure Stabilization

Oil column resonance is one the major problems that may arise in the control of servo-hydraulic actuators. The stiffness of the oil column coupled with the mass of the actuator piston and moving parts of the specimen create a resonant frequency that is normally outside the operational frequency range. However, this mode can be excited by the measurement or digitization noise, or oscillatory command displacements.

Delta-pressure gain uses the filtered signal from pressure differential in actuator cylinders to damp oil column vibration, as shown in Figure 4-4. The delta-p signal first passes through a high-pass filter, which intuitively should pass oil column and higher frequencies, and then is amplified and added to the control signal. Oil column frequency can be obtained by:

$$\omega_{\text{oil}} = \sqrt{\frac{K_{\text{oil 1}} + K_{\text{oil 2}}}{M}} \tag{4.19}$$

where $K_{\text{oil}\,i}$ is the oil axial stiffness in i^{th} actuator cylinder given by Equation (4.12), and M is the moving mass, including piston and moving parts of the experimental specimen. The effect of delta-pressure control is illustrated in Figure 4-10. It can be observed that less high-frequency oscillations occur when a nonzero gain is used for this control signal. It is important to note that the excitation of oil column frequency does not occur for all PID and feedforward parameter configurations, and is more serious for longer actuators that have lower oil column frequencies. Differential pressure (delta-p) stabilization is often used in MTS actuators to improve the fidelity of system performance (Figure 4-8).

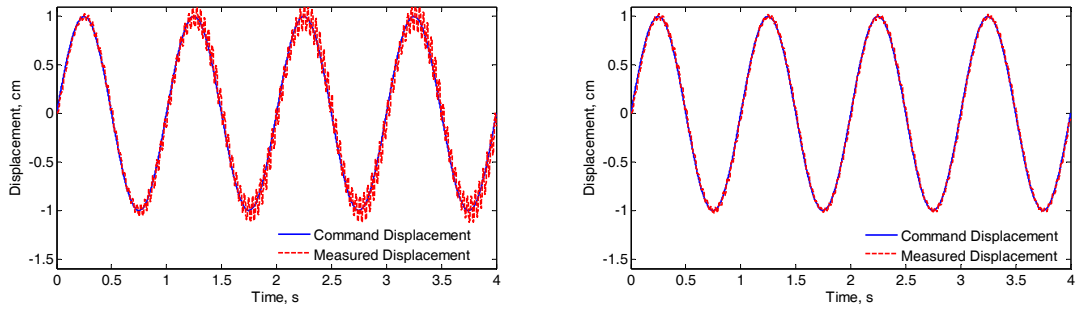

Figure 4-10 Effect of delta-pressure control on the actuator sinusoidal response – left: zero gain, right: adjusted gain.

4.4.1.5 Other Modifications and Limitations

When PID loops control a valve or similar mechanical device, the plant may not respond to small command signals as a result of wear of the valve or device. Replacing the valve can be a major maintenance cost. In these cases, the PID loop may have a deadband to reduce the frequency of activation of the mechanical device. This is accomplished by designing the controller to hold its output steady if the changes are small (within the defined deadband range). The calculated output must leave the deadband before the control signal changes. Then, a new deadband will be established around the new output value.

One problem with the differential (D) term of the control signal is that small amounts of noise can cause large amounts of change in the output. Sometimes it is helpful to filter the measurements, with a running average, or a low-pass filter. However, low-pass filtering and derivative control cancel each other out; therefore, a much better choice would be to reduce the measurement noise through improved instrumentations. Alternatively, the differential term can be turned off in many systems with little loss of control (a PI controller).

Digital implementations of a PID algorithm may have limitations owing to the sampling rate of the data, and the limits of internal calculation and precision. Another problem faced with PID controllers is that they are linear. As a result, they cannot completely eliminate the dynamics of nonlinear systems, or systems of higher order. For this reason, often PID controllers are tuned for a limited range of nonlinear operation of the plant, over which the plant can be approximated as linear.

4.4.1.6 Controller Tuning

Tuning a control loop is the adjustment of its control parameters to the optimum values for the desired system response. Generally stability of response is required and the process must not oscillate for any combination of plant conditions and reference signals. The response time of the process is another important factor, and the controller is normally tuned to minimize response lag of the system.

There are several methods for tuning a PID control loop. The choice of the proper method largely depends on whether or not the loop can be taken offline for tuning, and the response rate of the system. If the system can be taken offline, the best tuning method often involves subjecting the system to a step change in input, measuring the output as a function of time, and using this response to determine the best control parameters.

If the system must remain online, one tuning method is to first set the I and D values to zero. P should then be increased until the output of the loop starts to oscillate. Then I gain is increased until oscillation stops. Finally, D should be increased until the

loop is acceptably quick to follow its reference signal. A fast PID loop tuning usually overshoots slightly to reach the reference signal more quickly; however, not all systems can tolerate response overshoot. The effect of PID control parameters on the response are listed in Table 4-1.

Table 4-1 Effect of control parameters on the performance of PID Controllers.

Parameter	Rise Time	Overshoot	Settling Time	Steady-State Error
P	**Decrease**	**Increase**	**Small Change**	**Decrease**
I	**Decrease**	**Decrease**	**Increase**	**Eliminate**
D	**Small Change**	**Decrease**	**Decrease**	**Small Change**

Another tuning method is formally known as the Ziegler-Nichols method [98]. It starts in the same way as the method described before: first I and D gains are set to zero and then P is increased until the output of the loop starts to oscillate. This gain is called the critical gain, K_c, and the oscillation period is termed P_c. Then P, I and D controls can be adjusted using the parameters listed in Table 4-2.

Table 4-2 Ziegler-Nichols method parameters.

Control	P	T_I	T_D
P	$0.5K_c$	-	-
PI	$0.45K_c$	$P_c/1.2$	-
PID	$0.6K_c$	$P_c/2$	$P_c/8$

In the above table, T_I and T_D are called integral time and derivative time, respectively, and are related to PID control parameters by the following relations:

$$T_I = \frac{P}{I} \tag{4.20}$$

$$T_D = \frac{D}{P} \tag{4.21}$$

Most modern industrial facilities no longer tune loops using the manual calculation methods described above. Instead, PID tuning and loop optimization software are used to ensure consistent results. These software packages will gather data, develop process models, and suggest optimal tuning parameters. As an example, the STS software includes a spectrum analyzer and an FRF (frequency response function) plotter, which shows the average transfer function of the actuator in the setup for the specified frequency range. The control parameters can then be adjusted in order to achieve a unity transfer function magnitude over the designated frequency range, and a phase plot with minimal slope that implies minimum delay. The reference signal in this tuning process is generally a band limited white noise with small amplitude to keep the specimen in linear range [84].

5 ERRORS IN HYBRID SIMULATIONS

Hybrid simulation takes advantage of analysis of the numerical substructures combined with the physical testing of experimental substructures. Hence, both numerical and experimental errors are expected to occur during a hybrid simulation. Usually, numerical errors can be reduced beyond the desired precision for results, by following certain modeling and analysis guidelines. The errors in experimental substructures can also be reduced by proper tuning and calibration of test equipment and using high-performance instrumentation, although it is virtually impossible to entirely eliminate experimental errors.

In feedback systems like hybrid simulation, even small errors can accumulate during the experiment and significantly alter the simulation outcome, yielding inaccurate or unstable results. This is due to the fact that in time-stepping integration algorithms, experimental measurements contaminated by errors are used to compute subsequent commands. Hence, it is imperative to recognize the most important sources of error in hybrid simulation and seek ways to minimize and compensate these errors.

In this chapter, the major errors in hybrid simulation are briefly described and their most important sources are identified. In particular, the issue of actuator delay (a systematic experimental error) is discussed in detail; the sources and effects of delay are explored, and the offline and online methods for estimation of delay are studied. An improved online procedure for accurate and fast estimation of delay during hybrid simulations is introduced, and its performance is compared to the existing methods. The assessment of the reliability and accuracy of hybrid simulation results is discussed next, and quantitative error indicators are suggested for this purpose. This chapter concludes by examining the overall energy balance error for assessment of numerical and experimental errors in hybrid simulations.

5.1 SOURCES OF ERROR

In a hybrid simulation, errors are introduced from several sources: the structural model idealizations, the approximate numerical methods used to solve the equation of motion, and the experimental setup. A brief summary of these error sources follows.

5.1.1 MODELING AND IMPLEMENTATION TECHNIQUES

The ability of hybrid simulation to accurately capture the structural characteristics of the prototype structure is highly dependent on the selected numerical and experimental models. Approximations leading to idealized lumped-mass models, spatial discretization of elements, and characteristics of the utilized time integration procedure, all affect the simulation results. Most of the resulting errors are well identified, and it is often possible to minimize these errors by following certain guidelines in numerical

modeling, such as proper selection of finite element mesh size and geometry. In addition to modeling errors, the utilized simulation algorithm also influences the accuracy of the results; for example, simulations scaled in time may contain errors in rate-dependent forces developed in experimental specimen, or result in stress relaxations in ramp-and-hold-type experiments.

A number of the modeling errors resulting from the above-mentioned simplifications are sometimes admitted to exist in simulation, in order to alleviate the difficulties associated with physical testing, such as test equipment limitations, slow processing tools, and cost. As a result of these issues, often highly simplified numerical models are used in fast hybrid simulations. Simplification of boundary conditions for numerical and experimental substructures is another example of such errors. It is normally difficult to produce seamless boundary conditions between numerical and experimental substructures, mainly due to the limitations of testing equipment. From this standpoint, the separation of numerical and experimental substructures should be carried out carefully, to minimize the number of constraints on the boundaries of the physical test setup. It is expected that these errors can be reduced with the use of enhanced processing tools and improved testing equipment.

Another potential error source is the utilized numerical integration method. These algorithms offer various stability and accuracy characteristics along with different levels of difficulty in terms of implementation, which should be considered in their selection for hybrid simulation. Numerical integration errors have been extensively studied by Shing and Mahin [58, 99]. They pointed out that errors in numerical integration may

result in frequency distortions, which in turn may lead to significant errors in the response of linear undamped structures. This is primarily due to the sensitivity of maximum response amplitude of these structures to their natural period T. Nonetheless, the errors can be significantly reduced if the selected integration time step Δt is considerably smaller than the natural period. In addition, when viscous damping exists, or when inelastic deformations occur, the sensitivity of the structural response to frequency distortions greatly reduces. Since structural components are usually loaded into their nonlinear response range in hybrid simulations, and viscous damping is often included, small frequency distortions can often be neglected.

Commonly, the selection of proper integration time step for numerical simulations should follow these criteria: (i) it should satisfy the stability conditions of the utilized integration algorithm, (ii) it should be small enough to avoid large frequency distortions, and (iii) it should be small enough to accurately capture the nonlinear behavior of the test structure. It should be noted that the accuracy limits on $\Delta t / T$ ratio are generally more stringent than stability limits, and ratios selected for accuracy normally satisfy stability conditions for most integration methods. It should be noted that the limitations of hybrid simulation may impose additional restrictions on the simulation time step. For example, the time step may need to be larger than a lower limit to meet the available communication capacity for the intended experiment rate, or to mitigate the error propagation problems in the simulation.

5.1.2 EXPERIMENTAL SETUP

The experimental setup introduces various sources of error in a hybrid simulation that can influence the computed structural response. These errors include actuator tracking errors and mistuning, calibration errors of instrumentation, and noise generated in measurement instrumentation and A/D converters. Experimental errors are known to have the most substantial impact on the simulation results, as they are unidentified before the simulation, and may be large in improperly tuned test setups.

The experimental errors can be classified as random or systematic. Random errors have no distinguishable pattern, while a regular pattern of occurrence can be identified for systematic errors. As an example, the lag in the displacement response of an actuator, can produce systematic uncertainties, as the resulting displacement errors have a magnitude and direction proportional to the velocity demand [11]. The effects of delay become more significant when it is considerably larger than the simulation time step. Hence, selection of very small time steps that may be required by the numerical integration component for acceptable simulation accuracy, may degrade the overall results, and is not always advisable for hybrid simulation. Due to the importance of delay in hybrid simulation, the sources, effects, and estimation and compensation procedures for this error are studied in detail in the following sections and Chapter 6.

Errors in tuning and calibration of instrumentation are also categorized as systematic, as their direction and magnitude follow a regular pattern. These errors should be minimized in hybrid simulations before the experiment, as they can directly affect the simulation results. When a full elimination of these errors is impossible, proper

compensation methods should be employed to prevent their accumulation during the simulation (Chapter 6). Other systematic errors can occur in experimental elements scaled in size, or as a result of nonrealistic gravitational forces from reduced experimental masses.

The measurement noise resulting from the instrumentation and A/D converters can be classified as random. Actuator tracking errors also include random components in addition to the systematic components described above. Random errors are an integral part of all experiments, and reasonable amounts of these errors are normally acceptable in the results of ordinary open-loop experiments. However, their effects are more pronounced in hybrid simulation as a feedback system; these errors can be considered as an external source of high-frequency excitation signal that may potentially amplify and destabilize the system through the excitation of higher-frequency modes and unmodeled dynamics.

5.2 *DELAY*

Delay in the experimental hardware is one of the most important systematic errors in feedback systems. In the field of structural engineering this issue has been faced in active structural control and hybrid simulations, due to the inherent delay of servo-hydraulic actuators. The importance of delay and its effects on hybrid simulation results are discussed in this section.

5.2.1 SOURCES OF DELAY

Delay in a hybrid simulation may result from lags in computation, communication, and servo-hydraulic system dynamics. Computational delays occur when the processing tools are too slow to complete the numerical integration task of the numerical substructure within one sampling period. Delays in communication are more important in geographically distributed hybrid simulations, where data transmission between numerical simulation and experiment sites takes a finite amount of time. Communication delays normally fluctuate and may occasionally be large. Event-oriented simulation methods have been developed to handle these situations [11, 22].

Delay in servo-hydraulic actuators is known to be the most important part of overall system delay; the actuator always follows the command displacement with a lag. Actuator delay may be reduced by proper selection of control parameters. However, it cannot be completely eliminated due to its tradeoff with control loop stability as was discussed in Chapter 4.

5.2.2 EFFECTS OF DELAY

In order to demonstrate the effect of delay in hybrid simulation, a linear SDF experimental specimen is considered. The hysteretic graph (force-displacement diagram) for a linear specimen should be a line, whose slope is equal to the stiffness. However, in the presence of delay, the measured force r^m corresponds to the measured (achieved) displacement u^m, which has a delay compared to the command displacement u^c (Figure 5-1). As a result, the apparent (observed) behavior of the specimen (command

displacement *versus* measured forces) is similar to a viscoelastic material with negative damping that tends to add energy into the system. The additional energy is indicated as a shaded area within a reverse hysteretic loop shown in Figure 5-1. If this negative damping is greater than the total damping of the system, the system may become unstable. Delay is also shown to reduce the hysteretic energy dissipation of nonlinear substructures in Figure 5-1. In this case, the instability occurs when the negative damping effect of delay is greater than combined hysteretic and damping energy dissipation capacities of the system. Note that delay always introduces errors in hybrid simulation through the negative damping effect, regardless of simulation stability.

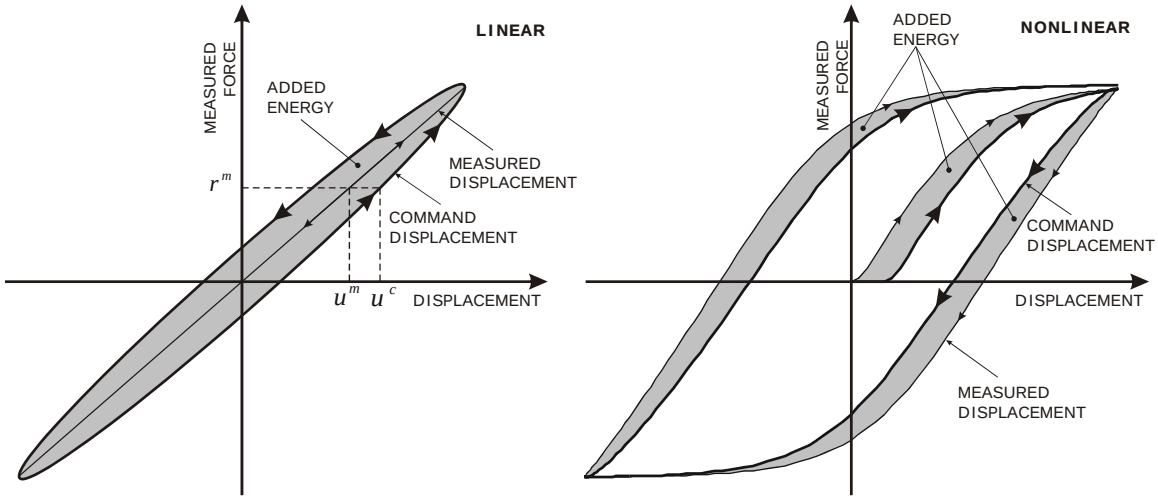

Figure 5-1 Effect of delay as negative damping in linear and nonlinear simulations.

To examine the effect of delay in the equation of motion (1.1), it is assumed that the experimental force vector $\mathbf{r}$ is delayed by a delay τ. The equation of motion for a linear SDF system with purely experimental restoring force will then be:

$$m\ddot{u}(t) + c\dot{u}(t) + k^e u(t-\tau) = f(t) \tag{5.1}$$

where k^{e} is the stiffness of the experimental specimen, and $f(t)$ is the output excitation term at time t, and is equal to $-m^{t}\ddot{u}_{g}(t)$ for an earthquake excitation. Defining the natural frequency $\omega_n = \sqrt{k^E/m}$ and damping ratio $\xi = c/2m\omega_n$, Equation (5.1) can be expressed as:

$$\ddot{u}(t) + 2\xi\omega_n\dot{u}(t) + \omega_n^2 u(t-\tau) = \frac{f(t)}{m} \tag{5.2}$$

By using Laplace Transform, the transfer function of the system between the input force and displacement response will be:

$$T_d(s) = \frac{U(s)}{F(s)} = \frac{1/m}{s^2 + 2\xi\omega_n s + \omega_n^2 e^{-\tau s}} \tag{5.3}$$

This transfer function can be converted to frequency domain by substituting $s = i\omega$ with ω being the excitation frequency of a harmonic load of the form $f(t) = f_0 \sin(\omega t)$ [39]:

$$T_d(i\omega) = \frac{1}{k}\frac{\left[\cos(\beta\phi_d) - \beta^2\right] - i\left[2\xi\beta - \sin(\beta\phi_d)\right]}{\left[\cos(\beta\phi_d) - \beta^2\right]^2 + \left[2\xi\beta - \sin(\beta\phi_d)\right]^2} \tag{5.4}$$

in which $\phi_d = \tau\omega_n$ is the phase delay and $\beta = \omega/\omega_n$. Comparing Equation (5.4) with the transfer function of a SDF system without time delay:

$$T(i\omega) = \frac{1}{k}\frac{\left[1 - \beta^2\right] - i\left[2\xi\beta\right]}{\left[1 - \beta^2\right]^2 + \left[2\xi\beta\right]^2} \tag{5.5}$$

one can observe that the delayed system is equivalent to the SDF system with the following effective damping ratio:

$$\xi_{eff} = \sqrt{\xi^2 - \xi\sin\phi_d - (1 - \cos\phi_d)/2} \tag{5.6}$$

which is calculated by comparing the resonant peaks of the two transfer functions at $\beta = 1$ [39].

The transfer function given by Equation (5.4) for 5% damping is plotted against the frequency ratio β for different values of phase delay, ϕ_d in Figure 5-2. These plots demonstrate the effect of delay as a phase delay, whose value is proportional to the natural vibration frequency. Therefore, a small delay can significantly reduce the effective damping ratio of higher-frequency modes, and possibly result in instability of the simulation. Consequently, more artificial damping in the numerical integration algorithm may be required to suppress the spurious excitation of higher-frequency modes in the presence of delay.

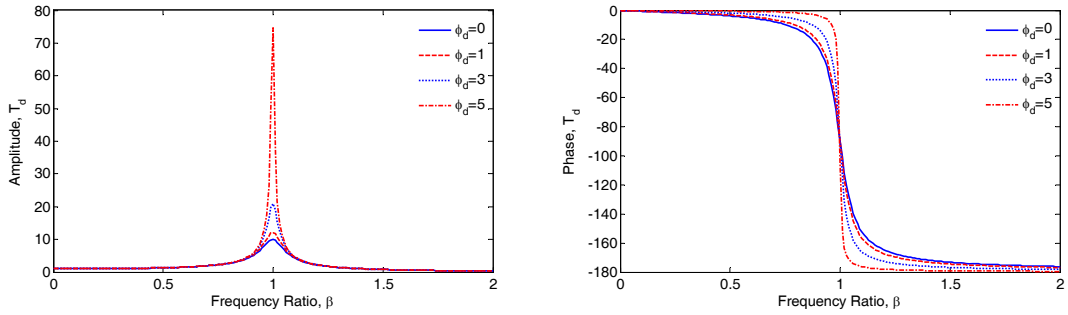

Figure 5-2 Amplitude and phase of system transfer function with delay (5% inherent damping).

In order to have a stable transfer function, the poles should be on the left half of the complex plane, i.e. the required damping ratio in Equation (5.4) should satisfy:

$$\xi_{req} > \frac{\sin(\beta\phi_d)}{2\beta} \tag{5.7}$$

The maximum values of the above equation occur for small values of β, where $\sin(\beta\phi_d) \simeq \beta\phi_d$ and the condition can be simplified as[1]:

$$\xi_{req} > \frac{\phi_d}{2} \tag{5.8}$$

It is important to note that the above-mentioned stability condition is developed for a continuous system, and does not consider the effects of time discretization and numerical integration algorithm. More stringent stability ranges for damping and/or time step size may exist when those considerations are taken into account.

In order to further demonstrate the effects of delay, a hybrid simulation is carried out for a SDF system shown in Figure 5-3. The natural period of the system is selected to be 0.5 seconds. The complete restoring force of the system is modeled experimentally using a vertical cantilever loaded by a transverse actuator (Figure 3-4) with a fairly constant delay of 15 milliseconds. The majority of the mass and the entire 5% of critical damping of the system are modeled numerically. The simulation is carried out for 1978 Tabas earthquake (a near-fault record with peak ground acceleration of 0.85g) scaled in amplitude by 2.5%. Low-level excitation is used to ensure a linear response in the experimental substructure, and prevent damage.

[1] Equation (5.8) can also be found from Equation (5.3) with a truncated Taylor Series expansion of exponential term in the denominator.

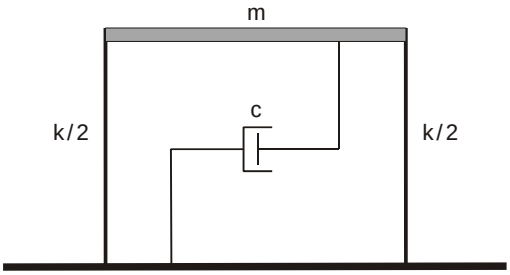

Figure 5-3 A single-degree-of-freedom structure.

The displacement history shown in Figure 5-4 demonstrates the unstable system response that triggers a system shut-down after the preset displacement limit of 5mm is exceeded. Note that a linear numerical simulation of the same system without delay produces a peak displacement less than 2.5mm. However, the simulation with delay becomes unstable even before strong earthquake shaking begins. The energy artificially added to the simulation can be clearly observed by comparison of measured or actual hysteretic loops (measured force *versus* measured displacement) of the experimental specimen with that observed by the numerical integrator (feedback force *versus* desired displacement) in Figure 5-4. The differences between the two hysteretic plots represent the errors introduced into the hybrid simulation as a result of actuator tracking errors. In this case, the behavior of the specimen observed by the numerical integrator consists of reverse hysteretic loops (negative damping) in the observed hysteresis.

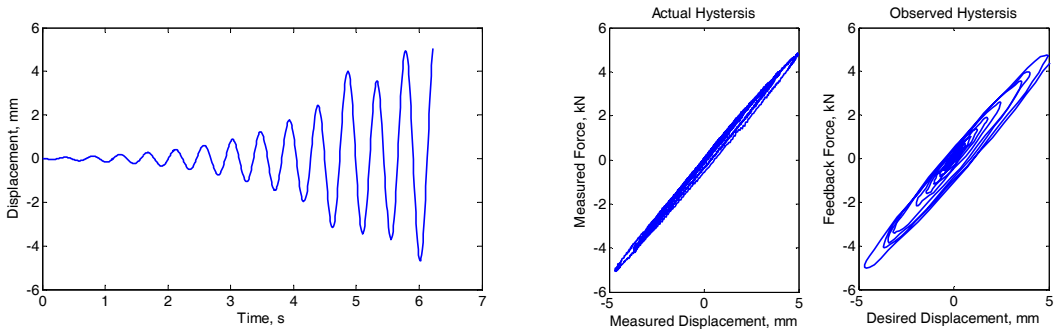

Figure 5-4 Displacement history of SDF hybrid simulation with delay, and actual and observed hysteretic loops.

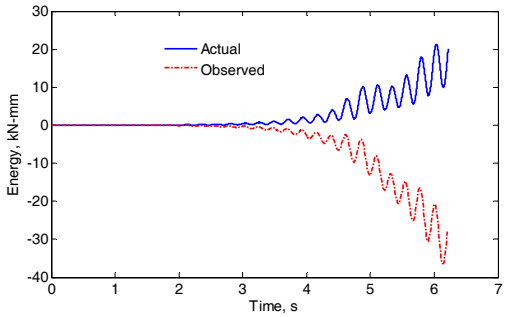

Figure 5-5 Actual and observed energy dissipation histories in SDF hybrid simulation with delay.

The negative damping effect of delay can also be observed in the energy histories shown in Figure 5-5. In this figure, the energy histories show the energy stored (in the forms of kinetic or elastic strain) or dissipated (through damping or inelastic strain) in the experimental substructure. It is illustrated that although the experimental substructure is actually dissipating energy, the observed energy plot shows that energy is being added to the system as a result of delay. The difference between these two histories shows the net energy added through negative damping as a result of delay errors. In Section 5.4 it will be shown that this difference in energy dissipation histories can be normalized by input energy and used as a measure of simulation accuracy [40].

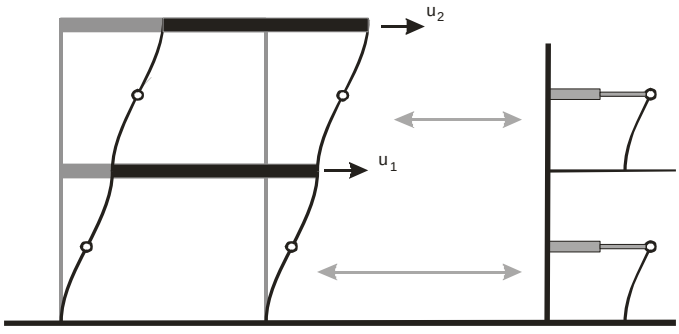

Figure 5-6 Two-degree-of-freedom system with columns modeled as experimental substructures [11].

To demonstrate the increased sensitivity of higher-frequency modes to delay, a two-degree-of-freedom system as shown in Figure 5-6 is considered with properties listed in

Table 5-1. Each experimental substructure represents the story shear *versus* story drift. The inherent damping is selected to be 5% of critical, which is assumed to be entirely in the analytical model. The majority of the mass is also assumed to be in the numerical model. The natural periods of this system are selected to be 0.42 and 0.16 seconds. The 2DF structure was subjected to 50% of Tabas earthquake. SDF numerical models described in Chapter 3 are used to replace the experimental substructure. The exact simulation result of this system can therefore be determined by eliminating all artificial error sources and utilizing a small-time-step implicit Newmark integration procedure.

Table 5-1 Structural properties of 2DF model.

	Weight (kN)	Stiffness (kN/mm)	Yield Displacement (mm)
Story 2	**222.5**	**6.57×2**	**20**
Story 1	**222.5**	**6.57×2**	**20**

The delays in the simulated actuators of the first and second story setups were selected to be about 15 and 7 milliseconds, respectively. Figure 5-7 shows the simulated results with delay, when no delay compensation or signal correction procedure is used. By observing the top floor displacement of the system, it is evident that the response is spuriously dominated by the second natural mode of vibration. In a simulation without errors, the actual contribution of this mode to the response is considerably smaller. This comparison confirms that the effective damping of the second mode is highly reduced as a result of delay in the actuator models. The hysteretic loops for the specimens of the first and second stories also show that significant amount of energy is being added to the system through reverse hysteretic loops.

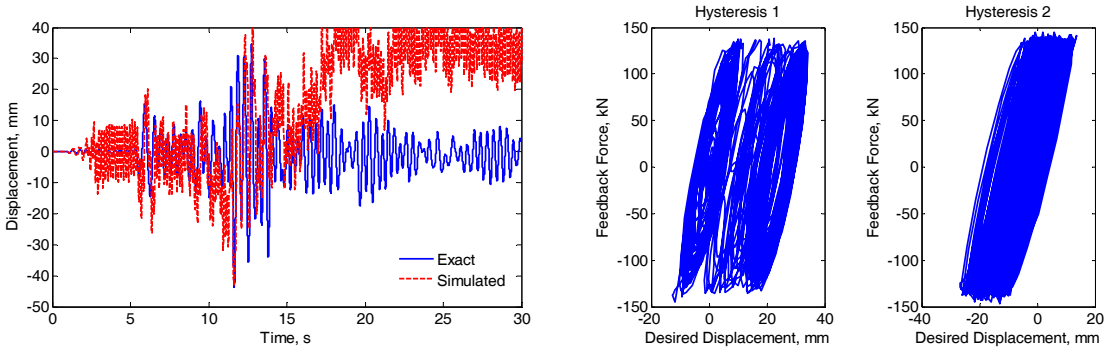

Figure 5-7 Simulation results of 2DF system without delay compensation – top floor displacement and observed hysteretic loops of experimental specimens.

5.3 *DELAY ESTIMATION*

Accurate estimation of actuator delay is essential for compensation and correction of signals to minimize the effects on simulation results. In most hybrid simulations, delay can be assumed constant throughout the simulation and can be calibrated offline. However, when delay is expected to significantly change during a simulation, online estimation is recommended for proper compensation.

5.3.1 OFFLINE ESTIMATION

Offline estimation of delay can be easily carried out by considering the system transfer function (Bode gain and phase plots) for a band-limited white noise excitation with small amplitude. In the system block diagram (Figure 1-2), time delay can be thought of as an extra block in the forward path that adds phase to the system but has almost no effect on the gain. That is, a time delay can be represented as a block with magnitude of 1 and phase $\omega(\mathrm{rad/s}) \times \tau(\mathrm{s})$. Therefore, Bode phase plot has the following simple relationship with system delay, τ:

$$\tau = \frac{\text{phase (degrees)}}{360^\circ \times \text{frequency (Hz)}} \tag{5.9}$$

In a physical test setup, the phase plot should be averaged for a relatively long excitation period to result in a fairly constant delay over the considered frequency range.

Also in Bode frequency plots, the phase margin[1] [100] can be used to measure the system tolerance to time delay. If there is a time delay greater than $180/\omega_{pc}$ (where ω_{pc} is phase crossover frequency, corresponding to a phase shift of 180˚), the system will become unstable in closed loop control.

5.3.2 ONLINE ESTIMATION

Although offline calibration is the easiest and most accurate way of measuring delay, the actual delay may not remain constant during a simulation. For example, the dependency of servo-hydraulic actuator delay on specimen stiffness was demonstrated by Darby *et al.* [54]. Hence, specimens with large stiffness variations due to buckling, fracture, or semi-active stiffness devices may require variable time delay compensation during the simulation.

Darby *et al.* [54] proposed an online procedure to measure the delay τ at step n during a simulation by multiplying constant proportional gains weighted by the velocity

[1] The phase margin is the difference in phase between the phase curve and -180˚ line at gain crossover frequency, ω_{gc} (the frequency corresponding to a gain of 0dB).

to the difference between the desired displacements (u^d) and measured displacements (u^m):

$$\tau_n = \tau_{n-1} + C_p \tanh\left(C_v \frac{u_n^d - u_{n-1}^d}{\Delta t}\right)\left(u_{n-1}^d - u_{n-1}^m\right) \tag{5.10}$$

The constants C_p and C_v should be determined experimentally by a tuning process, in which the delay history is considered during linear simulations involving the designated earthquake record and the actuator [54]. The multiplication of delay estimate at each step by the velocity increases the weights of the higher-velocity points of displacement history, where delay can be determined more accurately. As shown by Darby *et al.* [54], larger gains result in faster delay adjustments, but with larger oscillations.

In a numerical simulation of the 0.5-second-period SDF system described in Section 5.2.2 subjected to 50% of Tabas earthquake record, with $C_p = 0.01$ and $C_v = 10C_p = 0.1$, the estimated delay history is shown in Figure 5-8. The delay converges to the actual delay of 15ms after a few seconds and remains fairly constant for this model. Note that there are small oscillations close to the peak earthquake excitation.

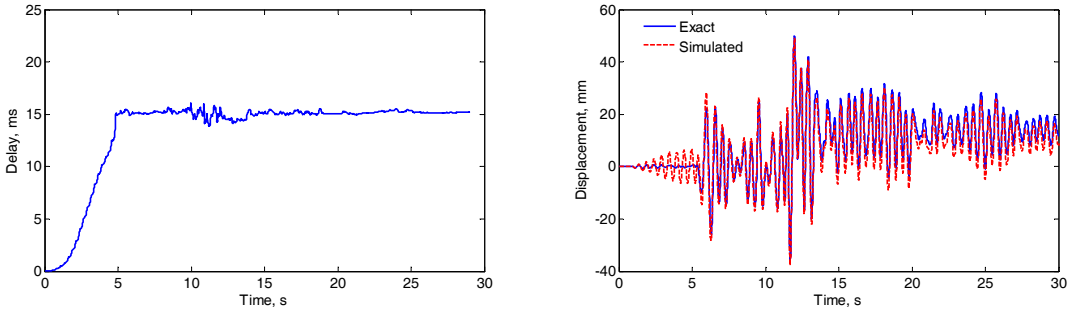

Figure 5-8 Actuator delay history estimated using Equation (5.10) and corrected simulation results.

The displacement history of the simulation using a variable delay compensation procedure (described in detail in Chapter 6) is also shown to be stable. However, in the beginning of the simulation, the estimated delay is significantly smaller than the actual delay, resulting in spurious response during the first 5 seconds in this 30-second simulation. In order to reduce this initial instability, it is necessary to increase the values of the constants C_p and C_v, which in turn, increases the amplitude of oscillations in the estimated delay near the peak earthquake excitation. The delay history is plotted in Figure 5-9 for $C_p = 0.04$. While it takes more than 3 seconds for the estimation procedure to reach the actual delay value, the oscillations are considerably large, ranging from negative values to several multiples of actual delay. Thus, the values of C_p and C_v need to be carefully selected for proper implementation of this method.

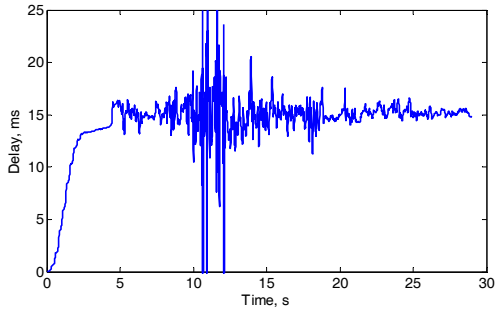

Figure 5-9 Estimated delay history using Equation (3) with higher gains ($C_p = 0.04$).

A new procedure is proposed here to improve the online delay estimation procedure. This approach directly measures the delay from the desired and measured displacement histories in the actuator coordinate system. As shown in Figure 5-10, the horizontal distance between the desired and measured time-history displacement signals is measured by using linear fits to the last three data points of each signal. Since

the points are equally spaced in time, the fit can be carried out with minimal computational effort. The delay estimated in step n can then be obtained from:

$$\tau_n = \tau_{n-1} + 2G\Delta t \frac{u_n^{ad} - u_n^{am}}{u_n^m - u_{n-2}^m} \tag{5.11}$$

where:

$$u_n^{ad} = \frac{u_n^d + u_{n-1}^d + u_{n-2}^d}{3}; \quad u_n^{am} = \frac{u_n^m + u_{n-1}^m + u_{n-2}^m}{3} \tag{5.12}$$

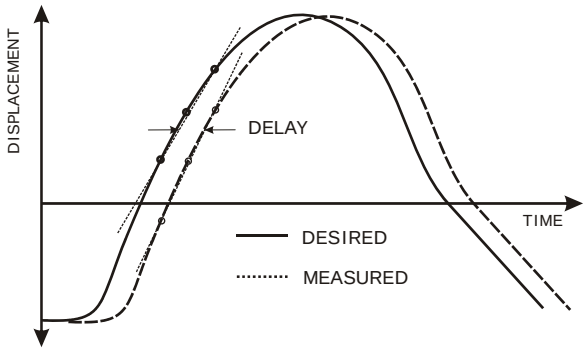

Figure 5-10 Direct estimation of delay using desired and measured displacement histories.

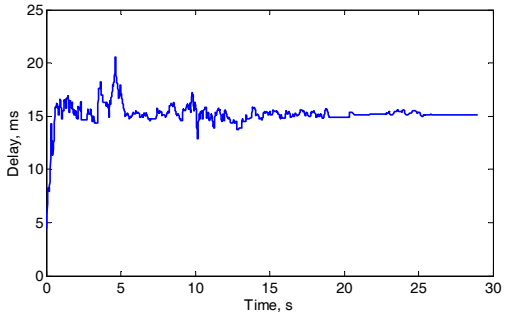

Figure 5-11 Estimated delay history using the procedure proposed in Equation (5.11).

and G is a learning gain. The estimated delay using Equation (5.11) is plotted in Figure 5-11, with a gain value of 0.1. As shown, the final value of the delay is similar to that obtained by Equation (5.10). However, the convergence is considerably faster, and the subsequent oscillations are reduced compared to Figure 5-10. Similar performance has

been observed in most numerical and experimental simulations carried out in this study. Therefore, this relatively simple estimation procedure shows improved characteristics for fast and accurate adjustments of delay during a hybrid simulation, compared to Equation (5.10).

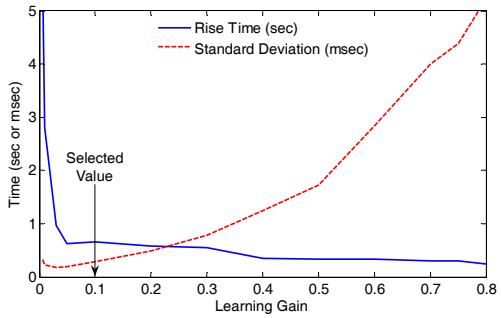

Figure 5-12 Effect of learning gain on rise time and standard deviation of the estimated delay after rise time.

Similar to the calibration constants in Equation (5.10), the learning gain G in Equation (5.11) should be properly selected, based on the quality of measurements and amount of expected delay variation. For this purpose, low-level physical simulations can be carried out to select the best value for G, while avoiding damage to the experimental specimens. Higher gain values result in faster estimates of delay but with larger oscillations through the simulation. Hence, the best value for G is the largest value that does not result in highly oscillatory estimations. Figure 5-12 shows the variation in rise time (the time required for the estimate to rise from 10% to 90% of the final value) and standard deviation after passing the rise time for different gain values for a SDF subjected to 1940 El Centro earthquake. Based on this graph, any value between 0.03 and 0.2 can be chosen to achieve a rise time less than 1 second and a standard deviation less than 0.5 milliseconds. As shown in Figure 5-12, this range includes the above-mentioned

gain of 0.1, which has been selected for the numerical and experimental simulations presented here with different ground motions. The selection of G may be dependent on the instrumentation used and on the quality of measurements, and should be calibrated for each experimental setup. In order to ensure delay estimation based on quality data, velocity of the step can also be used as a part of this gain as suggested in [54], or one can turn this adjustment off if the detected velocity is less than some limit, say, 50% of the maximum expected velocity.

As shown in the delay estimation history in Figure 5-11, delay can be considered constant during this particular experiment since the yielding of the specimen was limited. In this case, it is preferable to assume a constant delay to eliminate the extra computational effort associated with the estimation procedure. In addition, noise in delay estimation can lead to additional noise in the command displacement signal, which in turn can lead to contaminated measurements. Hence, it is recommended that online estimation of delay only be used in experiments involving significant stiffness variations, such as semi-active stiffness devices or fracturing specimens.

In order to reduce the initial time necessary for estimation procedures to converge to the actual delay value, a reasonable initial value for delay can be specified based on previous experience or pre-test calibrations. This is particularly useful in cases where the strong motion of the earthquake record starts after a few seconds of initial low-amplitude acceleration – the period in which accurate estimation of delay is difficult.

The amount of delay in most of experimental and numerical simulations carried out in this study was known, from offline calibrations and previous experiments. However,

Equation (5.11) is always used for online estimation with all compensation methods presented in Chapter 6, starting with zero delay estimation. This has been done to further evaluate the efficiency of the proposed method in experiments with little *a priori* information about the test setup. In these studies, it has been observed that it is often unnecessary to adjust the learning gain value before each test conducted using the same experimental setup, regardless of amplitude of testing and earthquake excitation. This shows the relatively low sensitivity of the proposed delay estimation procedure to small variations of the learning gain value.

5.4 ASSESSMENT OF ACCURACY

Before using the results of a hybrid simulation, their accuracy and reliability should be verified. As mentioned earlier, hybrid simulation is prone to both numerical and experimental errors; hence, the accuracy assessment procedures should consider both of these error sources.

It is often difficult to extend the numerical accuracy and stability limits of individual test procedures to hybrid simulations due to system nonlinearities and experimental errors. Particularly, simulation instability may occur well before reaching the stability limits that are calculated analytically for linear SDF systems. For example, when the negative damping resulting from delay in a linear numerical simulation becomes greater than the structural damping, instability occurs. However, instability may occur in a linear hybrid simulation with smaller delays due to measurement noise that acts similar to a high-frequency excitation signal. On the other hand, a nonlinear hybrid simulation

with larger delay may remain stable (but inaccurate) as a result of hysteretic energy dissipation. These nonlinearities and experimental errors also affect the analytical stability limits of numerical integration procedures. For this reason, it is important to develop error indicators that include nonlinearities and experimental errors, preferably without dependency on the numerical and experimental models.

Unlike purely numerical simulations, repeating a hybrid simulation due to unacceptable accuracy can be very costly, as the experimental substructures may sustain damage during the experiment and need to be replaced. To avoid these situations, excessive accumulation of errors should be detected at initial experiment stages to stop the experiment and investigate the underlying problems. For this purpose, online error indicators can be used for early detection of the errors [40], possibly before damaging the experimental substructure. These error indicators provide real-time information about proper functioning of simulation components such as experimental setup, numerical analysis, and compensation and correction of signals.

The hybrid simulation error indicators are discussed in the next two sections. In Section 5.5, an energy-based experimental error monitor is examined that indicates the proper detection of the behavior of the experimental substructure in the simulation. Next, an overall energy balance error is introduced in Section 5.6 for the assessment of combined numerical and experimental errors in hybrid simulations.

5.5 *RELIABILITY MEASURES FOR EXPERIMENTAL ERRORS*

One of the main goals of a hybrid simulation is the accurate identification of structural properties of the experimental substructure. It is important to ensure these properties are correctly captured in the simulation.

One way of assessing the accuracy of the captured behavior of experimental substructure is through the observed hysteresis; that is, the hysteretic behavior of the experimental substructure that is used in the numerical simulation. This hysteretic behavior should be in agreement with the actual behavior of the experimental substructure. In a hybrid simulation, the closest hysteretic behavior of an experimental specimen to the actual one is the measured hysteresis, which is a plot of measured forces *versus* measured displacements. For this reason, the actual hysteresis and measured hysteresis terms are used interchangeably in this study.

Delay is the most important experimental error that can lead to significant discrepancy between actual and observed behaviors of the experimental substructure. This was shown in the simulation of a linear SDF system in Section 5.2.2. Figure 5-4 shows that the observed hysteretic behavior significantly differs from actual behavior as a result of uncompensated delay. The difference can be evaluated by comparing the energy dissipated through these hysteretic loops [40]:

$$E_{\mathrm{E}}^{\mathrm{err}} = E_{\mathrm{E}}^{\mathrm{O}} - E_{\mathrm{E}} = \int \left(\mathbf{r}^{\mathrm{m}}\right)^{\mathrm{T}} \mathbf{du}^{\mathrm{d}} - \int \left(\mathbf{r}^{\mathrm{m}}\right)^{\mathrm{T}} \mathbf{du}^{\mathrm{m}} \tag{5.13}$$

in which $\mathbf{r}$ and $\mathbf{du}$ are experimental restoring force and incremental displacement vectors, and superscripts d and m denote the desired and measured values,

respectively; E_{E} is the energy stored in, or dissipated by the experimental substructures, $E_{\mathrm{E}}^{\mathrm{O}}$ is that observed by the numerical analysis subsystem, and $E_{\mathrm{E}}^{\mathrm{err}}$ is the experimental energy dissipation error. This equation takes into account the difference between the desired and measured displacement, but does not account for the corrections made in the measured force vector, if any. This energy error term can be normalized by input energy to give a non-dimensional error indicator that is merely dependent on the experimental errors (hybrid simulation error monitor, HSEM) [40]:

$$\mathrm{HSEM} = \frac{E_{\mathrm{E}}^{\mathrm{err}}}{E_{\mathrm{I}} + E_{\mathrm{E}}^{\max}} \tag{5.14}$$

where:

$$E_{\mathrm{I}} = -\int \left[\mathbf{M}^{\mathrm{t}} \boldsymbol{\iota} \ddot{u}_g(t)\right]^{\mathrm{T}} \mathbf{du} \tag{5.15}$$

is the input energy from earthquake excitation, and:

$$E_{\mathrm{E}}^{\max} = \frac{1}{2} \mathbf{u}_0^{\mathrm{T}} \mathbf{K}^{\mathrm{e}} \mathbf{u}_0 \tag{5.16}$$

is the maximum experimental strain energy and is used to prevent large values of error indicator in the beginning of simulation, when the input energy is very small. $\mathbf{K}^{\mathrm{e}}$ is the initial stiffness matrix, and $\mathbf{u}_0$ is an experimental displacement vector, which can be roughly selected as the maximum expected displacement, or the yield displacement of the experimental substructure. The choice of this displacement vector depends on the available information, and since it is used in the normalization of energy error, it should be considered in the selected limit for error indicator. Mosqueda *et al.* [39] showed that

one can limit the displacement and force errors of a hybrid simulation by limiting the amount of the above-mentioned error indicator.

Since the majority of errors in a hybrid simulation are likely from experimental sources, the above-mentioned HSEM can be a suitable choice for monitoring the simulation quality. As an example, the effect of delay compensation in the simulation of the two-degree-of-freedom structure (Figure 5-6) is shown in Figure 5-13. The negative value of HSEM shows a significant amount of energy is being added to the system when the delay is not properly compensated.

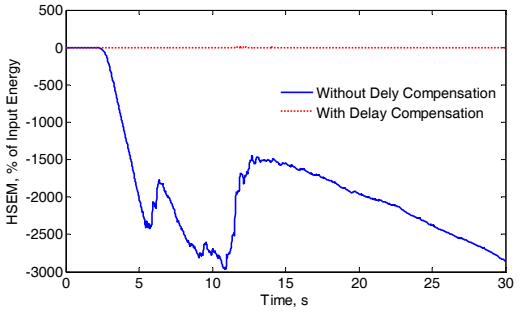

Figure 5-13 Effect of delay compensation on energy error monitor value.

A number of compensation procedures and integration methods that will be described in the following chapters apply corrections on the measured force, or modify desired displacements and measured forces to improve the stability. In order to observe the effects of those modifications on the accuracy of the captured experimental hysteresis, the energy error between the experimental and analytical subsystems can be defined as:

$$E_{\mathrm{EA}}^{\mathrm{err}} = E_{\mathrm{E}}^{\mathrm{C}} - E_{\mathrm{E}} = \int \mathbf{r}^{\mathrm{T}} \mathbf{du} - \int \left(\mathbf{r}^{\mathrm{m}}\right)^{\mathrm{T}} \mathbf{du}^{\mathrm{m}} \tag{5.17}$$

in which $\mathbf{r}$ and $\mathbf{du}$ are the final force and displacement values used in numerical analysis. These values are found at the end of each integration step (possibly after some modifications or iterations), and are used to calculate the next state values for continuing the simulation. For this reason, the hysteretic behavior using these values is called corrected or converged hysteresis, and the corresponding energy dissipation is termed $E_{\mathrm{E}}^{\mathrm{C}}$.

In order to ensure proper identification of experimental substructure properties, the energy error $E_{\mathrm{EA}}^{\mathrm{err}}$ should be monitored rather than $E_{\mathrm{E}}^{\mathrm{err}}$, since the latter only considers the errors that may occur outside of numerical simulation module. Hence, it is important to note that not all errors in $E_{\mathrm{EA}}^{\mathrm{err}}$ have experimental sources, and part of them may be originating from numerical simulation module. This error can also be normalized by a relation similar to Equation (5.14) for online monitoring of simulation errors.

5.6 *Overall Energy Balance for Evaluation of Total Errors*

Experimental studies are often accompanied with analytical models that are used to verify the results, or models that are developed based on the results obtained from the experiment. In any case, the discrepancies between analytical and experimental results cannot be fully eliminated. Furthermore, a hybrid simulation is carried out partially due to lack of accurate numerical models for experimental substructure, and analytical results most probably are not available during the simulation. For this reason, a comparison between fully analytical results and those of hybrid simulation may not be a suitable accuracy measure in most cases. In this section, an energy-based error measure

is introduced for hybrid simulation that includes both numerical and experimental errors, and does not require a numerical model of the test system.

Filiatrault *et al.* [42, 43] proposed the use of energy balance equation to estimate the extent of numerical errors in nonlinear seismic analyses. They showed that the energy balance is a better accuracy measure than a comparison among peak response parameters, such as displacements and accelerations. The error index introduced in this section also uses the energy balance for online assessment of simulation accuracy. In order to include both numerical and experimental errors in this index, the energy balance evaluation procedure is slightly modified as described below.

The energy balance equation of a simulation can be obtained by integrating the equation of motion (1.1) over displacement:

$$E_{\mathrm{K}} + E_{\mathrm{D}} + E_{\mathrm{S}} + E_{\mathrm{E}}^{\mathrm{C}} = E_{\mathrm{I}} \tag{5.18}$$

in which E_{K} is the kinetic energy of numerical mass, E_{D} is the energy dissipated through viscous damping in numerical substructure, E_{S} is the strain energy stored or dissipated in numerical substructure:

$$E_{\mathrm{K}} = \frac{1}{2}\mathbf{v}^{\mathrm{T}}\mathbf{M}\,\mathbf{v} \tag{5.19}$$

$$E_{\mathrm{D}} = \int \mathbf{v}^{\mathrm{T}}\mathbf{C}\,\mathbf{du} \tag{5.20}$$

$$E_{\mathrm{S}} = \int \mathbf{u}^{\mathrm{T}}\mathbf{K}\,\mathbf{du} \tag{5.21}$$

and $E_{\mathrm{E}}^{\mathrm{C}}$ is the energy stored or dissipated in the experimental substructure from an analytical standpoint, as discussed in the preceding section.

Both numerical and experimental errors affect how well the energy balance is maintained. For example, experimental errors make experimental energy E_{E} differ from $E_{\mathrm{E}}^{\mathrm{C}}$ used in numerical analysis to satisfy the equation of motion. On the other hand, numerical truncation errors or relaxed convergence tolerances may result in small differences between left- and right-hand sides of Equation (5.18). Hence, an overall energy error can be defined as:

$$E^{\mathrm{err}} = E_{\mathrm{I}} - \left(E_{\mathrm{K}} + E_{\mathrm{D}} + E_{\mathrm{S}} + E_{\mathrm{E}}\right) \tag{5.22}$$

Within the engineering precision requirements, and if the convergence tolerance is sufficiently small, the energy error obtained from Equation (5.22) will be very close to $E_{\mathrm{EA}}^{\mathrm{err}}$ from Equation (5.17). That is, it essentially includes the difference between actual experimental and converged energies[1]. Particularly, it cannot capture the errors of numerical integration procedure, since all integration methods satisfy the equation of motion and its integral form, Equation (5.18). However, perfect satisfaction of equation of motion is not sufficient for an accurate and stable simulation; the numerical

[1] This is based on the assumption that experiment and numerical analysis are in phase. Later, it will be shown that some integration procedures may load the experimental substructure beyond current integration step, which results in some additional energy error. Unlike other errors, the errors resulting from this phase difference do not add up during the simulation, and should reduce to zero at the end of excitation, when the system stops oscillating.

simulation procedure should also maintain proper kinematic relations between displacement, velocity and acceleration. To include the kinematic errors that may occur in the numerical simulation module of hybrid simulation, it is proposed to replace the velocity in Equations (5.19) and (5.20) by the first derivative of displacement:

$$E_{\mathrm{K}} = \frac{1}{2}\dot{\mathbf{u}}^{\mathrm{T}}\mathbf{M}\dot{\mathbf{u}} \tag{5.23}$$

$$E_{\mathrm{D}} = \int \dot{\mathbf{u}}^{\mathrm{T}}\mathbf{C}\,\mathbf{du} \tag{5.24}$$

With this modification, any error in the kinematic relation between displacement and velocity (and hence, between displacement and acceleration) will be reflected as a discrepancy of kinetic and damping energies from those satisfying Equation (5.18).

The energy terms of Equation (5.22) in incremental form are:

$$E_{\mathrm{I},\,n} = E_{\mathrm{I},\,n-1} - \frac{1}{2}\left\{\mathbf{M}^{\mathrm{t}}\boldsymbol{\iota}\left[\ddot{u}_g\left(t_n\right) + \ddot{u}_g\left(t_{n-1}\right)\right]\right\}^{\mathrm{T}}\Delta\mathbf{u}_{\mathbf{n}} \tag{5.25}$$

$$E_{\mathrm{K},\,n} = \frac{1}{2\left(\Delta t\right)^2}\left(\Delta\mathbf{u}_n\right)^{\mathrm{T}}\mathbf{M}\,\Delta\mathbf{u}_n \tag{5.26}$$

$$E_{\mathrm{D},\,n} = E_{\mathrm{D},\,n\text{-}1} + \frac{1}{\Delta t}\left(\Delta\mathbf{u}_n\right)^{\mathrm{T}}\mathbf{C}\,\Delta\mathbf{u}_n \tag{5.27}$$

$$E_{\mathrm{S},\,n} = E_{\mathrm{S},\,n-1} + \frac{1}{2}\left(\mathbf{u}_n + \mathbf{u}_{n-1}\right)^{\mathrm{T}}\mathbf{K}\,\Delta\mathbf{u}_n \tag{5.28}$$

$$E_{\mathrm{E},\,n} = E_{\mathrm{E},\,n-1} + \frac{1}{2}\left(\mathbf{r}_n^{\mathrm{m}} + \mathbf{r}_{n-1}^{\mathrm{m}}\right)^{\mathrm{T}}\Delta\mathbf{u}_n \tag{5.29}$$

where $\Delta\mathbf{u}_n = \mathbf{u}_n - \mathbf{u}_{n-1}$. Similar to Equation (5.14), a normalized error index can be calculated based on overall unbalanced energy:

$$\mathrm{EEI} = \frac{E^{\mathrm{err}}}{E_{\mathrm{I}} + E_{\mathrm{E}}^{\max}} \tag{5.30}$$

The energy error index given by Equation (5.30) has been extensively used to evaluate the effectiveness of compensation and integration procedures in the following chapters. In the calculations leading to the above-mentioned energy balance error, velocities are derived from displacements, and the actual energy dissipation of the experimental substructure E_{E} is considered. Hence, the resulting error is expected to include the effects of the numerical integration errors and kinematic inconsistencies, as well as the differences between the observed and actual behaviors of the experimental substructure.

6 DELAY COMPENSATION METHODS

It is virtually impossible to eliminate all experimental errors from a hybrid simulation. Examples of these errors are random noise in measurements and systematic errors like delay. In an ordinary open-loop experiment, small experimental errors are tolerable, as they can be easily removed after the experiment by a variety of offline correction procedures, such as baseline correction and gain adjustment, and signal smoothing using moving window averaging or low-pass filters. However, since hybrid simulation is a feedback system with direct use of measurements, it can be extensively affected by these errors, and their correction cannot be left until post-experiment data processing.

Hence, in addition to the attempts to minimize experimental errors, online correction and compensation procedures are employed to mitigate the effects of leftover errors on hybrid simulation results and prevent their accumulation. This is normally a challenging task, since the corrections should be carried out online within a short analysis time, with only available data belonging to the previous steps of the

experiment. Furthermore, it is difficult to establish a complete analytical model of the test system to use in compensation procedures.

Servo-hydraulic actuator delay is one of the most critical sources of errors in a hybrid simulation. It was demonstrated in Chapter 5 that the effect of time delay in a feedback system is similar to negative damping that may result in an unstable simulation. In this chapter, different approaches for compensation of delay in feedback systems used in structural engineering are presented. The procedures that are most widely used in hybrid simulations are studied, and new methods are introduced. The performance of the compensation methods are then compared using analytical studies and energy errors in numerical and experimental simulations.

6.1 DELAY COMPENSATION METHODS FOR ACTIVE STRUCTURAL CONTROL

The problem of delay has been faced in feedback systems used in many areas of engineering. In structural engineering, the first real-time feedback control systems were developed for active control of structures [101, 102]. Since an active control process involves structural response measurement, computation of control forces, and transmission of command signals to actuators, a time-delay always exists in the application of control forces to the structure. Similar to hybrid simulations, the negative damping introduced by delay may significantly degrade the performance of active control systems.

An important difference between active control of structures and hybrid simulation is the availability of a fairly complete numerical model of the feedback system. Often, a

reasonably accurate numerical model is available in control problems that allows for prediction of states [45, 103-105]. With a numerical model for the system, the effects of delay can also be formulated and augmented into the system equations [104, 106], or added to the feedback system as a new subsystem [44, 107]. The effects of delay can then be considered in the design of control system. Further, it has been shown that an appropriate choice of delay can be beneficial for active control (time-delayed control of structures) [108, 109].

By comparing the transfer functions of closed-loop systems with and without delay, Smith [110, 111] proposed a novel method of time-delay compensation, known as Smith Predictor method. In this approach, the compensator is designed such that the characteristic equation of the delayed system with Smith Predictor is the same as that of the ideal system without delay. Hence, the stability of the structure using the Smith Predictor can be analytically guaranteed. An accurate analytical model of the structure is essential for Smith Predictor to adequately compensate small to moderate delays.

In addition to the above procedures, delay compensation methods have been developed for control problems that do not explicitly use the numerical model of the system. Using the past measurements in a Taylor Series expansion for prediction of states [47], and fitting curves to sampled response quantities and extrapolating for compensation of delay [112] are examples of these methods. The phase shift method is another way of delay compensation [47, 48, 113, 114] based on the idea of changing the phase angle between the control force and structural response to obtain an acceptable response reduction. For this purpose, dominant frequency of vibration is necessary to

modify the gain matrix. To extend this method to MDF systems, the mode shapes and natural frequencies of the open-loop system are also required [113, 114].

6.1.1 APPLICABILITY TO FAST HYBRID TESTS

A wide range of delay compensation procedures have been developed for active control of structures that take advantage of the numerical model of the feedback system. The performance of these methods is highly dependent on the accuracy of the utilized mathematical model. A hybrid simulation, however, is carried out for identification of the structural behavior of complex or highly nonlinear experimental components. In addition, it involves complex dynamics of servo-hydraulic test system. Hence, the properties of the feedback system are often unavailable during the experiment for use in delay compensation procedures. For this reason, mostly methods that require less information about the numerical model of the system have been adapted to hybrid simulation. An example of these procedures is the curve fitting method that does not need a numerical model of the system, and has been widely used in hybrid simulations. More detail about this compensation procedure is presented in the next section.

Smith Predictor is another active control delay compensation method that has been applied to force-controlled hybrid simulations [5, 20, 115]. However, as an accurate numerical model of the system is required for this compensator, it has only been applied to linear systems. The implementation strategy consists of an ordinary feedback loop plus an inner loop that introduces two extra terms directly into the feedback paths. The first term is an estimate of the process variable in the absence of time delay, and the second is an estimate of process variable with time delay. Subtracting the estimated

process variable from the actual process variable yields an estimation error, which can be added to the predicted process variable to create a feedback signal without time delay. As a result of using Smith Predictor, the system becomes a closed loop feedback control system with the time delay term outside the loop.

As illustrated in Figure 6-1 [10], it is expected that the model imperfections result in a signal named *modeling error*, which is used to modify feedback and compensate for the imperfections. If the actuator model is perfect, this model is able to completely remove its dynamics from simulation results. Regarding the delay effect, Figure 6-1 shows that the majority of the *interface force* feedback is taken from a numerical model with minimal delay, and the servo-hydraulic actuator delay only exists in the error portion of the feedback signal to the computational substructure.

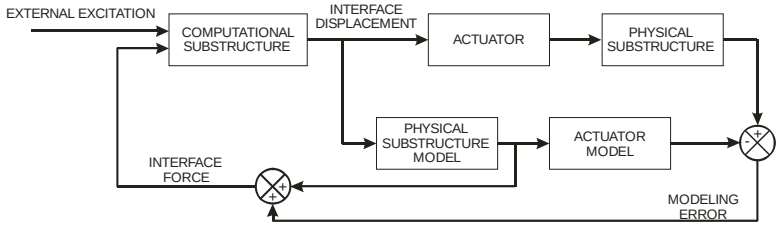

Figure 6-1 Displacement-controlled hybrid simulation with Smith Predictor.

The block diagram in Figure 6-1 does not provide any means for updating the substructure model; in other words, the experimental substructure is modeled using an invariable transfer function. As a result, any change in the properties of the physical substructure during the simulation leads to an increase in the modeling error signal, and degrades the performance of compensation system. To account for specimen nonlinearities, the modeling error signal can be used to modify the parameters of the physical substructure model, as illustrated in Figure 6-2 [10]. This configuration

resembles a classic problem of model reference adaptive control. Instead of modifying the feedback force, the modeling error is used to update the parameters of the numerical model and improve its performance. The model update can be carried out at the numerical integration rate, which is normally slower than the experiment command update and measurement rate, to reduce the computational costs [116].

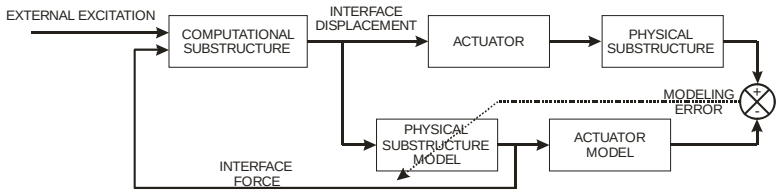

Figure 6-2 Displacement-controlled hybrid simulation with adaptive Smith Predictor.

One problem in using the Smith Predictor is its sensitivity to the actuator model used in the simulation. It is evident that this compensator cannot eliminate the non-modeled dynamics of the actuator. Further, if the amount of delay in the model is different than the actual delay present in the actuator an incorrect error signal results. Another challenge is that the adaptation laws and learning rates should be carefully selected to ensure the use of quality data in updating the model parameters. The learning rates should be fast enough to enable the model to capture rapid changes in system behavior while avoiding the instabilities that may occur using large learning gains.

6.2 DELAY COMPENSATION METHODS FOR FAST HYBRID TESTS

In this section, the most widely-used compensation procedures for hybrid simulation are reviewed, and new methods are proposed. The performances of these methods are also examined through numerical and experimental studies.

Based on the way the correction procedures modify hybrid simulation signals, the compensation methods can be categorized in two major groups [117]: i) procedures that modify or compensate the command displacement signal, and ii) procedures that correct the force measurement signal. These two approaches appear as two correction/compensation blocks in Figure 1-2, one before sending the commands to the actuator controller, and the other before returning the measurements to numerical integration block.

6.2.1 COMMAND DISPLACEMENT MODIFICATION METHODS

In a typical command displacement delay compensation procedure shown in Figure 6-3, the command displacement at time t, denoted by d_n', is predicted ahead of the desired displacement d_n by the expected delay, τ. Since a predicted command displacement is sent to the actuator, the actuator response is then expected to be close to the desired displacement d_n at the end of the numerical integration time step, Δt. Based on this nomenclature, the desired and command displacement signals in Figure 1-2 (u^{d} and u^{c}, the output signals of the simulation and the first correction/compensation blocks) consist of d_n and d_n' values, respectively.

The most-widely used delay compensation procedure for hybrid simulation was introduced by Horiuchi *et al.* [50]. In their recommended procedure, a third order polynomial is fitted to the last four desired displacements, and used to predict the command displacement ahead of simulation time by extrapolation on the fitted polynomial. In addition to the simplicity of this approach, the fitted polynomials can

also be used to generate command signal at time intervals equal to δt using desired displacements calculated by integration module at Δt intervals [19]. Further, the polynomial extrapolation can continue in case of computation or communication delays, until the next numerical integration results become available [11].

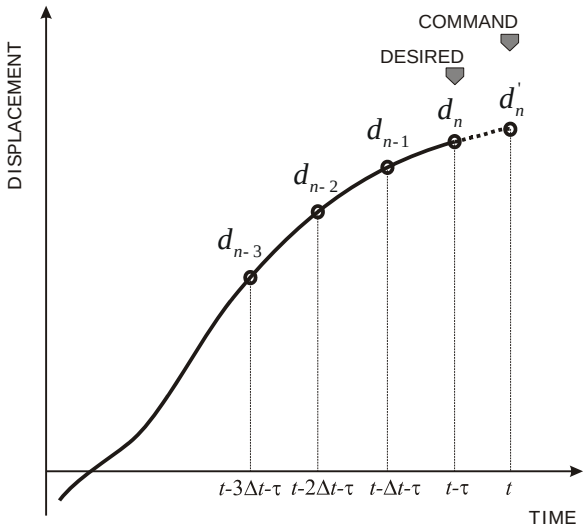

Figure 6-3 General extrapolation problem for command displacement signal.

Horiuchi and Konno [51] proposed another prediction algorithm by assuming a linear variation of acceleration, which also provides third order displacement variation. Since most of the integration procedures that are currently being used in hybrid simulation are explicit (e.g. Newmark's Beta method [118]), the velocity and acceleration at desired displacement time $t-\tau$ remain unknown until the force feedback at the desired displacement is available. Therefore, the command displacement can be determined based on the states at times $t-\tau-\Delta t$ and $t-\tau-2\Delta t$, using the following equations:

$$a_n' = a_{n-1} + \frac{\Delta t + \tau}{\Delta t}\left(a_{n-1} - a_{n-2}\right) = \left(2 + \frac{\tau}{\Delta t}\right)a_{n-1} - \left(1 + \frac{\tau}{\Delta t}\right)a_{n-2} \tag{6.1}$$

$$d'_n = d_{n-1} + (\Delta t + \tau)\, v_{n-1} + \frac{1}{3}(\Delta t + \tau)^2\, a_{n-1} + \frac{1}{6}(\Delta t + \tau)^2\, a'_n \tag{6.2}$$

In the above relations, subscripts denote the integration step number, and primed variables denote a predicted quantity.

The command displacement delay compensation procedure proposed herein, is based on the modification of time step in the numerical integration procedure. In this method, the delay-compensated command displacement d'_n is determined using the same numerical integration equations used to determine d_n from the available states at time $t - \tau - \Delta t$. For example, many numerical integration procedures use an explicit Newmark's method ($\beta = 0$) to determine the desired displacement at step n:

$$d_n = d_{n-1} + \Delta t\, v_{n-1} + \frac{1}{2}(\Delta t)^2\, a_{n-1} \tag{6.3}$$

which is a finite difference kinematics equation assuming constant step acceleration. The compensated displacement can then be predicted by simply increasing the integration time step:

$$d'_n = d_{n-1} + (\Delta t + \tau)\, v_{n-1} + \frac{1}{2}(\Delta t + \tau)^2\, a_{n-1} \tag{6.4}$$

Since the desired and command displacements are determined in the same way in the proposed method, it can be incorporated in the numerical simulation module. Note that the increased time step is only used in Equation (6.4) for calculation of command displacement, and should not affect the integration algorithm time step Δt.

It should be noted that Equation (6.4) can be derived from Equation (6.2) for the case of constant acceleration. That is, Equation (6.2) has an additional term given by:

$$d'_{\mathrm{n,\,Horiuchi}} - d'_{\mathrm{n,\,Proposed}} = \frac{1}{6}(\Delta t + \tau)^3 \frac{a_{n-1} - a_{n-2}}{\Delta t} \simeq \frac{1}{6}(\Delta t + \tau)^3 \, \dot{a}_{[n-1,n-2]} \tag{6.5}$$

which is a third-order term, resulting from the assumption of linear variation for acceleration.

6.2.2 FORCE CORRECTION METHODS

As an alternative to command displacement modification procedures, corrections can be applied to the measured force signals. From an analytical standpoint, correction of forces can be as effective as correction of displacements, because force measurement signal is the main feedback to the numerical model for continuing the simulation. The force corrections can also be carried out in combination with displacement extrapolation procedures to correct the actuator tracking errors, which may exist even when displacements are carefully compensated for delay.

One method of delay compensation in force measurements is using the same polynomial extrapolation procedures described in the preceding section. If this force extrapolation is accompanied by a displacement extrapolation procedure, care should be taken not to compensate for delay twice; force extrapolation should compensate for the portion of delay that was not previously compensated in displacement signal. For this reason, the online delay estimation task should yield two values as shown in Figure 3-9. The first output is *delay* that should be compensated in displacement signal, and the second is *uncompensated delay*, which should be corrected in force signal. The former is

calculated cumulatively using Equation (5.11), and the latter reflects the instantaneous delay between desired and measured displacement signals. *Delay* output increases until the time lag between the desired and measured displacements diminishes and remains constant afterwards; at this point, the *uncompensated delay* reduces to zero and force extrapolation stops until a change in system delay disturbs this condition.

A major problem in polynomial extrapolation of force measurements is the noise contamination, which is often worse than in displacement signals. As a result, force predictions often have a reduced quality compared to displacement predictions. Another issue that further supports the use of displacement extrapolation methods is that the actuator follows the desired displacement signal more closely when command displacement is properly compensated, than when only force is corrected.

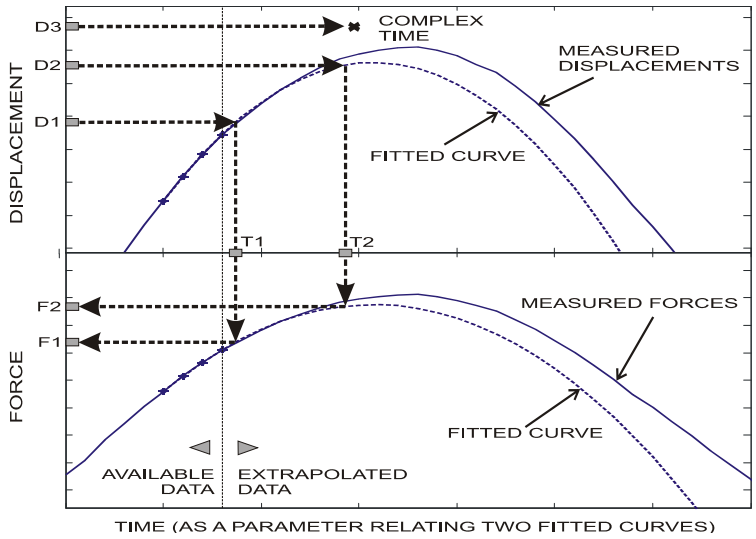

Figure 6-4 Estimation of force corresponding to the desired displacement using measurements.

A different approach is considered for correction of measured force by seeking the time at which the desired displacement is achieved and its corresponding force within

the measured data. As shown in Figure 6-4, two second-order polynomials are fitted to the latest few force and displacement measurements in actuator coordinate system, as:

$$r(t) = a_r t^2 + b_r t + c_r \tag{6.6}$$

$$u(t) = a_u t^2 + b_u t + c_u \tag{6.7}$$

By using the displacement polynomial, the time corresponding to the desired displacement d_n can be determined as:

$$t_d = \frac{-b_u \pm \sqrt{b_u^2 - 4a_u(c_u - d_n)}}{2a_u} \tag{6.8}$$

from which the value closer to current step time will be chosen. The force can then be found by replacing the above-mentioned time into the force polynomial given by Equation (6.6). Note that if delay in a displacement-controlled test is overcompensated in displacement signal, this procedure will only interpolate on the measured force data. This procedure eliminates the need for exact estimation of delay since it directly seeks the force corresponding to the instant the desired displacements is achieved. As a result, it can be combined with a displacement extrapolation procedure to compensate for tracking errors before the estimated delay reaches the actual value. Such correction can improve the stability of the simulation in initial low-amplitude excitation periods, such as the one shown in Figure 5-8. After the estimated delay reaches the actual value, this force correction procedure attempts to reduce the effects of remaining random actuator tracking errors.

The use of time to relate the two fitted polynomials has several advantages. First, since the measured data points are equally spaced in time, determination of fitted polynomial coefficients is computationally efficient. In addition, the effects of specimen nonlinearities will be less pronounced on time histories compared to force-displacement curves, providing better quality curve fitting. Clearly, the prediction error using the fitted polynomials increases when extrapolating further from the fitted points. To minimize errors, limits can be imposed on the variation of time parameter within one step to avoid extremely large extrapolations. The extrapolation limit should be selected sufficiently large for compensation of maximum expected delay in a simulation.

One problem in using force compensation procedures is that Equation (6.8) can result in a complex time value, particularly during undershooting of displacement at displacement reversals. This occurs when the desired displacement d_n does not satisfy $4a_u d_n \geq 4a_u c_u - b_u^2$ (Figure 6-4). Inserting the resulting complex-valued time in the fitted force polynomial then results in a complex force value. If the desired displacement remains close to the measured displacement peak, the imaginary part of the resulting complex force is small, and its real part increases proportionally beyond the force peak. Consequently, the absolute value of the complex force, which is approximately equal to its real part, provides an adequate estimate of the force. Application of this force correction procedure in steps with small imaginary force components has shown to be advantageous in minimizing experimental errors compared to simulations without corrections in steps with complex-valued time.

As each of the peaks of the command displacement will be achieved with a delay, at which time the command has already reduced from its maximum, an undershooting is more likely to happen during displacement reversals. In order to mitigate the systematic undershooting of displacements during displacement reversals, a variable gain can be applied to the delay compensated command displacement. The gain can be calculated online based on previous comparisons of desired and measured displacement peaks with moving window averaging to smoothen the results. Since the restoring force from the attached specimen tends to move the actuator piston back to its equilibrium point, the command signal gain is applied to the position of the piston with respect to the this point, termed zero displacement. The equilibrium can be found by seeking the zero-force feedback from the actuator load cell. As the zero displacement of a nonlinear specimen may drift from its original position, it should also be determined online during the simulation.

It has been observed that the use of a variable gain on command displacements may increase the number of simulation steps with successful force correction in SDF experimental substructures. An example of the variable gain and zero displacement history estimated online in an experiment is presented in the next section. For MDF experimental substructures, however, determination of zero displacement is difficult due to the interaction of actuators on the experimental substructure. As a result, the gain estimations become oscillatory and ineffective in MDF experimental substructures.

As mentioned earlier, force measurements may contain a significant amount of noise. The majority of this noise can be removed by using conventional low-pass filters

at the expense of additional system delay. In any case, the use of the minimum number of points to fit polynomials is likely to result in a fit highly contaminated by the noise present in the measurements. To avoid this situation, it is recommended that additional points be used to fit the polynomial. The extra computational effort for using more points is small, since normally the measured points are equally spaced in time. In the numerical and experimental studies presented in this dissertation, quadratic polynomials are fitted to the four most recent measurements (one more than minimum required points), to obtain an adequate fit.

6.3 COMPARISONS OF COMPENSATION PROCEDURES

In this section, the above-mentioned delay compensation procedures are compared analytically, numerically and experimentally. These comparisons also include a number of existing delay compensation methods that are widely used in hybrid simulations.

6.3.1 DISPLACEMENT EXTRAPOLATION METHODS

The performances of displacement extrapolation methods are studied by examining their ability to predict a sinusoidal displacement history. This analytical approach is similar to that used by Nakashima and Masaoka [19] in evaluating the polynomial extrapolation method. Assuming a sinusoidal displacement signal with frequency ω and amplitude A, the exact displacement at time t is given by:

$$d = A\sin\omega t \tag{6.9}$$

It is assumed the command (predicted) displacement is sought at time t, while the displacement, velocity and acceleration of time $t-\Delta t-\tau$ and prior steps are given by:

$$d_{n-i} = A\sin\omega\{t-\tau-i\Delta t\} \qquad i \geq 1 \tag{6.10}$$

and its derivatives. Further, only the displacement d_n at time $t-\tau$ is assumed to be known through an explicit Newmark's kinematics equation (6.3). After some manipulations, the predicted displacement at time t can be expressed in the form:

$$d_n' = m'A\sin(\omega t+\theta') \tag{6.11}$$

which is different from the exact displacement given by equation (6.9) by an amplitude magnification m' and phase shift θ'.

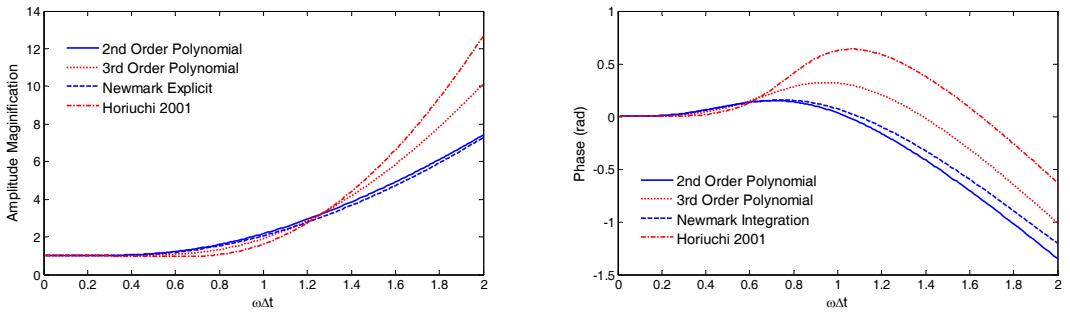

Figure 6-5 Amplitude magnification and phase errors for different displacement prediction methods.

The resulting phase and amplitude errors are plotted in Figure 6-5 for a normalized integration time step $\omega\Delta t$ with $\Delta t = 0.01\text{s}$, $\delta t = 0.001\text{s}$, and $\tau = 0.009\text{s}$. It can be observed that all procedures give a minimal dispersion when $\omega\Delta t < 0.25$. For larger values of $\omega\Delta t$, different procedures offer various advantages and disadvantages. However, it is evident that with selection of higher order procedures, better precision can be achieved for small $\omega\Delta t$ (less than 0.3), but for mid-range values, lower order procedures appear to perform relatively better.

Although Figure 6-5 suggest that polynomial extrapolation procedures have a comparable performance to procedures assuming constant or linear acceleration, the latter methods show significant advantage in force measurements during earthquake simulations. As an example, Figure 6-6 shows close-up views of displacement histories for a 0.5-second-period SDF subjected to Tabas earthquake using different compensation methods. It is evident that the kinked command displacements from polynomial extrapolation result in small oscillations in the displacement measurement, which in turn, lead to significant noise in the measured force signal. The Fourier amplitude plot of force measurements in Figure 6-7 confirms that polynomial extrapolation procedure results in larger amplitude noise in the higher frequency range. The results provided here are based on numerical simulations where other sources of uncertainty are better controlled. Similar benefits to using kinematics expressions have been observed in actual hybrid tests.

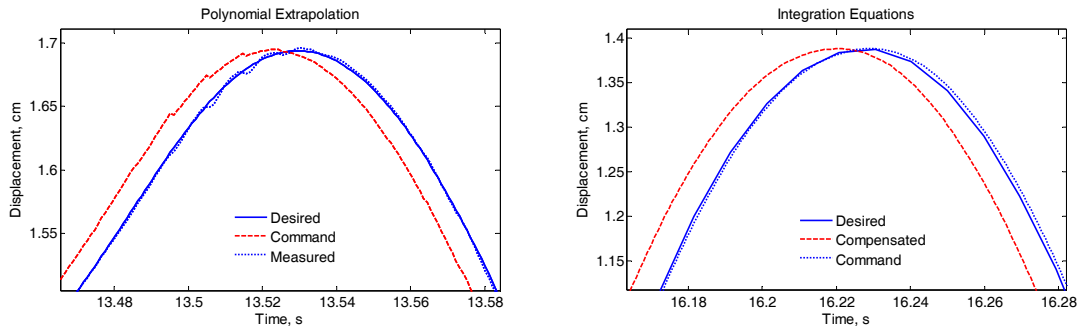

Figure 6-6 Comparison of command displacement signals using polynomial extrapolation and numerical integration ($f = 2\text{Hz}$, $\omega\Delta t = 0.12$).

6.3.2 PERFORMANCE IN NUMERICAL AND EXPERIMENTAL SIMULATIONS

Numerical and experimental simulations have also been used to verify the effectiveness of the compensation procedures described in this chapter. A few examples

of simulation results are presented and the energy error indicators introduced in Chapter 5 are used to compare the performance of these compensation procedures.

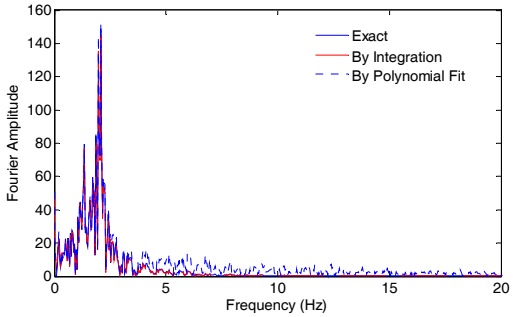

Figure 6-7 Fourier amplitude of the force measurements resulting from numerical simulations with different displacement extrapolation methods.

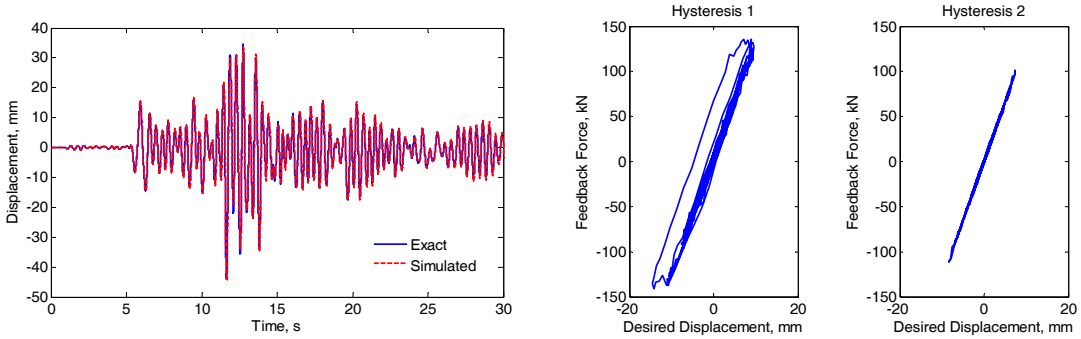

Figure 6-8 Simulation results of 2DF system with delay compensation using kinematics equation – top floor displacement and observed hysteretic loops of experimental specimens.

Figure 6-8 shows the numerical simulation results for the two-degree-of-freedom structure described in Section 5.2.2, using the explicit kinematics equation (Equation (6.4), assuming constant step acceleration) for compensation of delay. Compared to Figure 5-7, it is evident that this compensation procedure eliminates the majority of the energy errors and suppresses the erroneous second mode response. The hysteretic loops demonstrate minimal change in energy dissipation without the negative damping effect. The second story behaves in the elastic range during the entire simulation and the first story yields slightly. The variation of HSEM for this simulation shown in Figure 5-13

also confirms the effectiveness of this compensation procedure. This comparison demonstrates that in absence of compensation procedures, the majority of the energy input into the simulation originates from systematic actuator delay errors as opposed to the earthquake excitation.

The estimated actuator delays using Equation (5.11) were also observed to quickly converge to the actual values during this simulation, as shown in Figure 6-9. Since the delay values assumed for the actuator models were constant and independent from the specimen behavior, no significant oscillations were observed in the estimated delay histories.

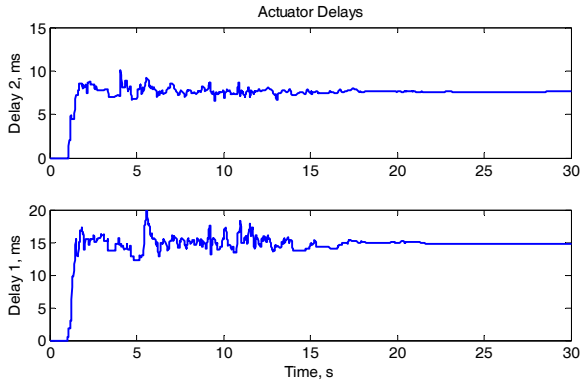

Figure 6-9 Estimated delay history in the simulation of 2DF structure using the proposed online delay estimation method.

The simulation results using other compensation procedures, namely, the force correction based on desired displacement and delay compensation in measured force signal have also been observed to be effective in minimizing the negative damping effects. For large delay values (larger than 20ms for the structure in question), however, the force correction procedures fail to produce acceptable results, which can be attributed to the noise in force measurements. Another reason can be the fact that with a

large delay, the desired and measured displacements are significantly different, and larger extrapolations may be necessary to estimate the force corresponding to the desired displacements.

The SDF experimental setup shown in Figure 3-4 is utilized for the experimental verification of correction and compensation procedures examined in this study. The natural period of the system has been selected to be 0.5 seconds. With a measured stiffness of about 880 N/mm, the required mass is 5.59 kN·s²/mm, out of which 0.01 kN·s²/mm is estimated to be present in the experimental specimen. The stiffness is completely modeled in the experimental specimen, while the entire 5% of critical damping is modeled numerically. In these experimental studies, the measurement and actuator command update rate and integration time step were 1024Hz and 10/1024s, respectively. The explicit form of Newmark's Beta integration procedure is used to solve the equation of motion in the numerical simulation module.

A 16-millisecond delay in actuator response was measured, which is considerably larger than other published amounts of delay [39, 50, 51, 54] and delays that have been recently measured at SEESL using newer actuators and servovalve models. Figure 6-10 shows the estimated delay history using Equation (5.11) in a nonlinear experiment with delay compensation using kinematics displacement equation. As illustrated, the delay estimate quickly converges from zero to values close to the actual delay with minor oscillations in the initial low level excitation. This and other experiments have shown that the proposed delay estimation procedure provides reliable results.

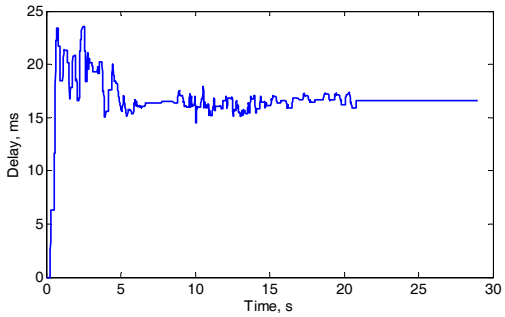

Figure 6-10 Estimated delay history using Equation (5.11) in a nonlinear experiemental simulation.

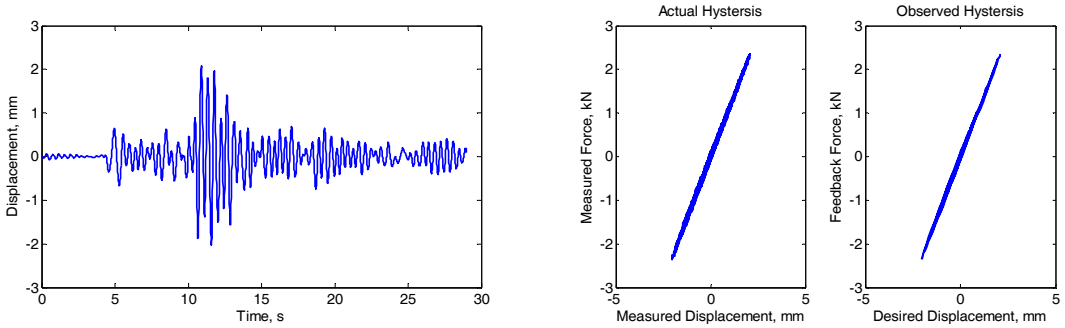

Figure 6-11 Hybrid simulation results with polynomial extrapolation of command displacements – 2.5% Tabas erthquake.

All delay compensation procedures discussed in this chapter were observed to be successful in compensation of 16 milliseconds delay for a real-time hybrid simulation of the above-mentioned SDF system. Typical experimental results for different compensation procedures are shown in Figures 6-11 to 6-14. The results are shown for the 0.5-s period system subjected to Tabas earthquake record with different amplitude scales for linear and nonlinear response. As demonstrated, these compensation procedures effectively compensated the delay to achieve stable simulations. The actual hysteretic behaviors of the specimen (measured forces *versus* measured displacements) are also compared to those observed by the numerical integrator (feedback forces corrected by compensation procedure *versus* displacements required by the numerical integrator). These comparisons demonstrate good agreement between actual and

observed specimen behavior, and imply minimal energy alteration of the system due to delay.

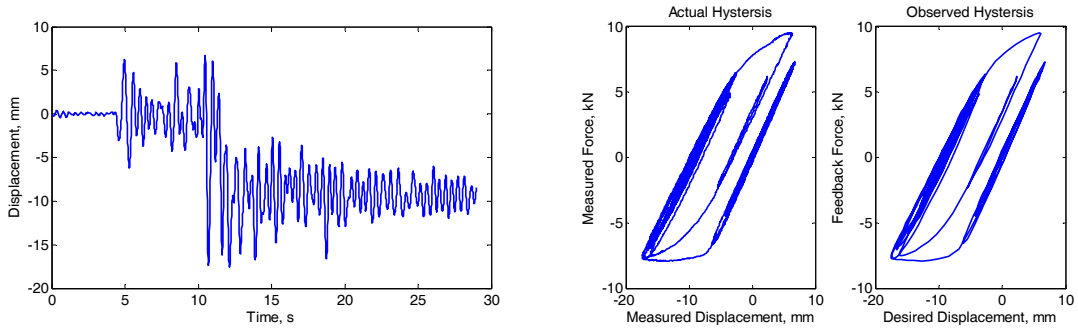

Figure 6-12 Hybrid simulation results with extrapolation of displacements using kinematics equation – 20% Tabas erthquake.

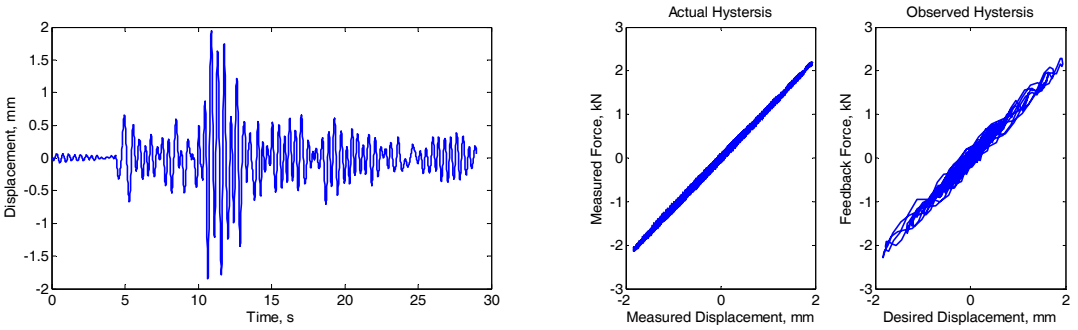

Figure 6-13 Hybrid simulation results with delay compensation in force signal – 2.5% Tabas erthquake.

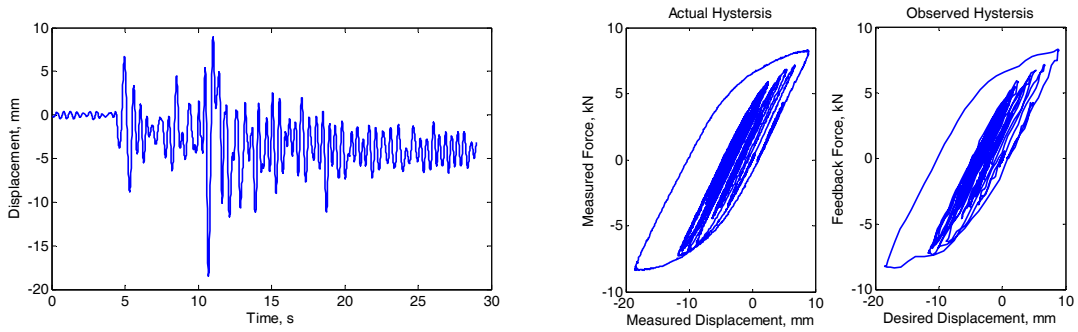

Figure 6-14 Hybrid simulation results with correction of force mesurements based on desired dispalcements – 25% Tabas earthquake.

The energy errors added to a simulation have been observed to be relatively smaller in displacement modification procedures, as listed in Table 6-1 for linear experiments. Other tests with 30 milliseconds delay also showed that the force correction procedures

failed to effectively compensate for larger delay, while displacement correction procedures provide reasonable results. If the specimen is not sensitive to the rate of loading, time scaling may be necessary to carry out the simulations with large delay in actuator response.

Table 6-1 Final energy error value using different compensation procedures (linear experiments).

Compensation Method	Energy Error (% of Input Energy)	
	Tabas 1978	El Centro 1940
Kinematics Equation	**0.22**	**0.20**
Displacement Polynomial Extrapolation	**0.26**	**0.35**
Force Polynomial Extrapolation	**1.15**	**1.84**
Force Estimation Based on Desired Displacement	**2.53**	**2.09**

Numerical and experimental simulation results also support the implementation of delay compensation procedures on the displacement signals. This can be partially attributed to the higher amounts of noise in force measurements that make large extrapolations difficult, as mentioned earlier. In order to further explain the advantage of displacement modification procedures over force correction methods, it should be noted that in the latter methods, the force feedback corresponds to a displacement that may not be physically applied to the experimental substructure. As a result, the actual energy dissipation will progressively differ from the feedback amount as the delay increases.

It should be noted that the energy error values listed in Table 6-1 are obtained using small-amplitude excitations (2.5% Tabas earthquake and 7.5% El Centro Earthquake). With these small excitations, the force measurements will be small, and the effect of noise in them will be more significant. As a result, the force-based correction methods

are considerably less effective in these simulations. These differences will be smaller in higher-amplitude excitations. For example, the final energy error in a higher-amplitude simulation with force correction based on desired displacement (Figure 6-14, 25% Tabas earthquake) reduces to 0.97% of input energy. This is closer to 0.63% error that resulted from a similar nonlinear experiment with the use of kinematics displacement equation for delay compensation (Figure 6-12, 20% Tabas earthquake).

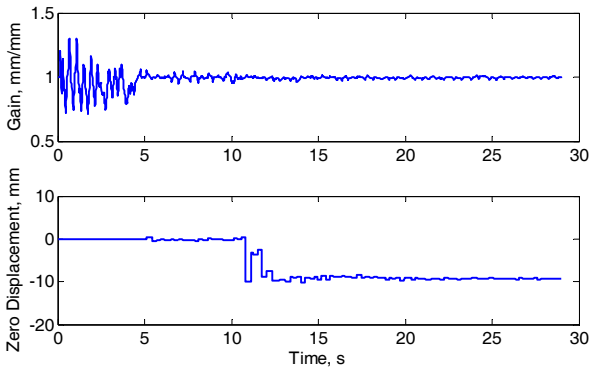

Figure 6-15 Gain and zero displacement histories estimated during a nonlinear simulation.

A typical gain and zero displacement history (introduced in Section 6.2.2 for reduction of actuator undershooting) estimated online in a nonlinear simulation (Figure 6-12) is shown in Figure 6-15. The zero displacement history is in agreement with the plastic drifts that can be observed in Figure 6-12. The gain history shows some large oscillations in the beginning of the simulation, where the excitation amplitude is small. After the initial low-level excitation, the gain takes values very close to unity, showing small tracking errors at displacement reversals. Proper tuning of the actuator with constant control parameters is always of priority, as variable control gains may increase the noise level in the command signal and lead to the excitation of high-frequency modes of the system.

6.3.3 DETAILS OF SIGNAL CORRECTION PROCEDURES

In order to illustrate the complete conditioning process of displacement signals, a short period of a typical displacement history of an experimental degree of freedom is shown in Figure 6-16. As illustrated, the desired displacement is first compensated for delay to give the compensated displacement signal. Next, this signal is re-sampled at experiment command update rate to generate the command displacements. This signal is then sent to the actuator, and the achieved displacements are measured and fed back to the numerical integrator. The feedback displacement signal is sampled at numerical simulation rate from normally-higher-rate measured displacement signal. As shown in Figure 6-16, the feedback displacement may further be modified by a force correction procedure, or by the numerical simulation module to give the converged displacement signal.

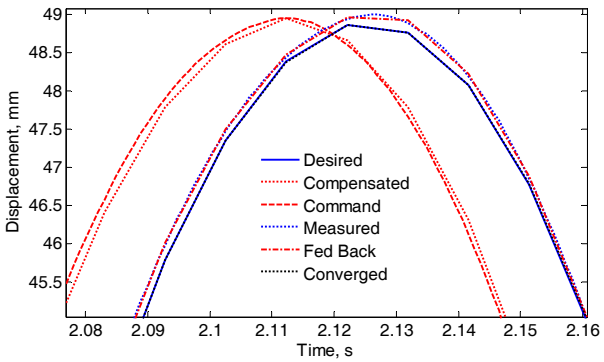

Figure 6-16 A typical displacement history in a hybrid simulation with displacement compensation.

Typical displacement and force histories for a hybrid simulation with force correction are shown in Figure 6-17. This figure shows that in this case, the command displacement signal is directly calculated from the desired displacement signal and sent to the actuator. The measured force and displacement signals are then fed into a

correction module after re-sampling at numerical simulation rate, for correction of actuator tracking errors. Note that the feedback signals may also be filtered to reduce measurement noise and produce smoother signals. The corrected force and displacement signals are finally fed back to the numerical simulation module.

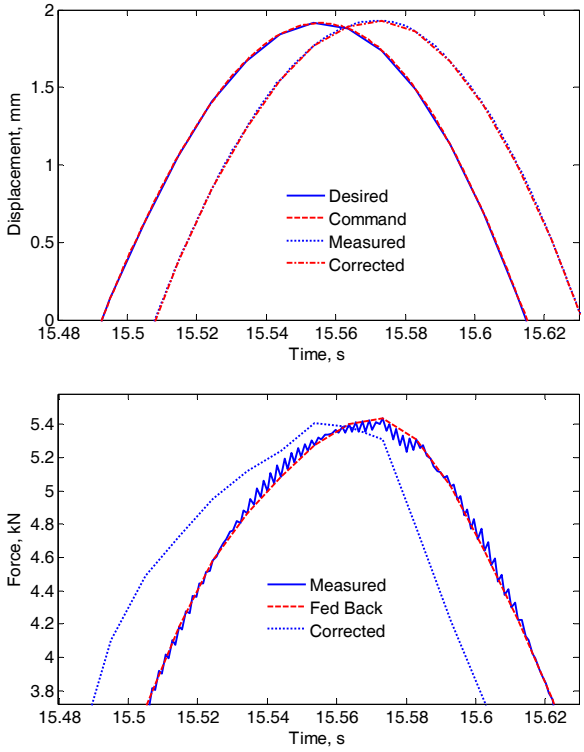

Figure 6-17 Typical displacement and force histories in a hybrid simulation with force correction.

6.3.4 EXPERIMENTAL STUDIES OF MDF AND HIGHLY NONLINEAR SYSTEMS

Experimental studies have also been conducted to investigate the effectiveness of delay compensation procedures for MDF experimental substructures. It has been observed that the above-mentioned delay estimation and compensation procedures are able to minimize the effects of delay from the test system, as long as they are individually calibrated and applied to each of the actuators. In this regard, there is no

restriction on the similarity of delay or dynamic behavior of actuators. On the other hand, the interaction of actuators makes the determination of zero-force displacement and hence, the variable gain difficult. As a result of off-diagonal terms in the stiffness matrix of MDF experimental substructures, the equilibrium point of each actuator constantly drifts even in linear systems. Hence, the tangent stiffness of the experimental substructure is necessary for detection of zero-force displacement, which is normally unavailable in most hybrid simulations. In this study, no variable gains were applied to most of the numerical and experimental simulations involving MDF experimental models.

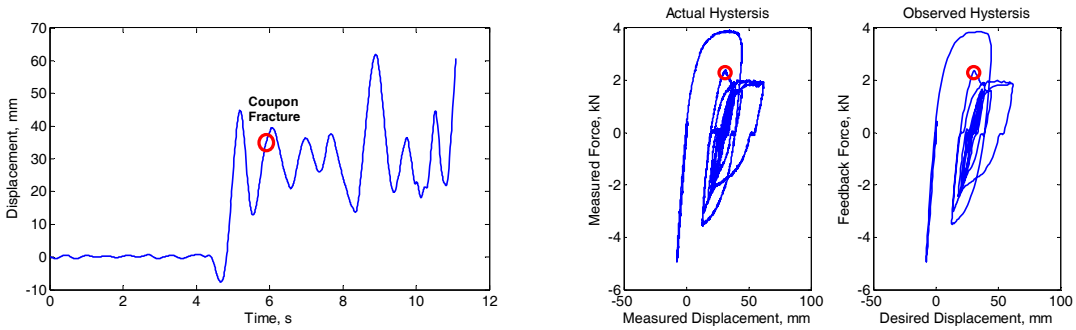

Figure 6-18 Hybrid simulation results with extrapolation of displacements using kinematics equation – 25% Tabas erthquake, with coupon fracture during the experiment.

Through several numerical and experimental simulations, it has been observed that the delay compensation procedures studied in this chapter have low sensitivity to the nonlinearities of the experimental substructure. This can be attributed to the fact that these procedures do not use any numerical models of the test system. Furthermore, the compensation procedures act within small time intervals, within which the nonlinearities are only pronounced mildly. As an example, Figure 6-18 shows the results of a nonlinear simulation, in which one of the coupons was fractured (Figure 6-19). The

hysteretic loops show abrupt strength reduction at the 5.9 second of the simulation, and a distorted behavior afterwards. The simulation stops at 11th second, when the command displacement exceeds the preset limit of 63.5 mm (2.5 in).

Figure 6-19 Fractured coupon during an experiement.

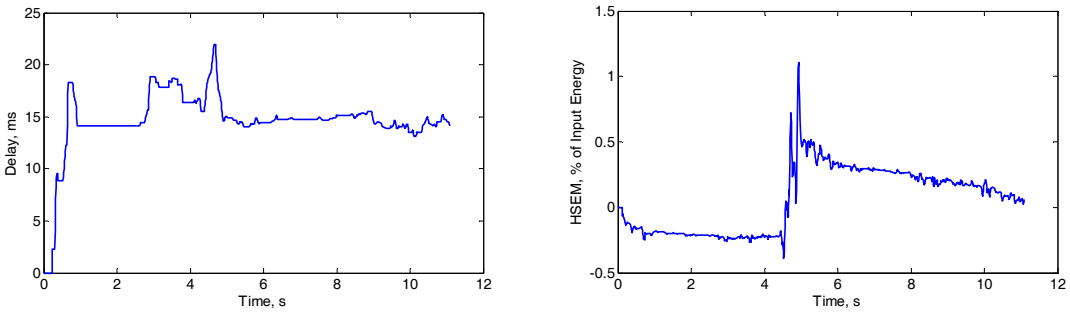

Figure 6-20 Delay history and HSEM during a simulation involving specimen fracture.

Figure 6-20 shows the delay history estimated using Equation (5.11) and the values of hybrid simulation error monitor introduced in Section 5.5. The delay history shows some oscillations during the low-amplitude excitation period at the beginning of the simulation. As the strong motion of earthquake begins, another local maximum can be recognized in the delay history, which coincides with an abrupt increase in the HSEM value. These phenomena can be attributed to the increase in the excitation amplitude, which results an increase in all simulation signals. It should be noted that the coupon fracture occurred about 1.5 s after the beginning of earthquake strong motion, and no significant change is observed in the error monitors or delay history as a result of this fracture. Note that in this simulation, the experimental restoring force remains smaller than 5 kN, which is about 25% of actuator capacity. After the fracture, the restoring force in the positive displacement range drops to values less than 2 kN, while in the negative side the force peaks at about 3 kN. That is, the actuator force demand drops to about 15% of its capacity as a result of coupon fracture. This reduction, however, did not lead to a significant and meaningful change in the estimated delay as observed by Darby *et al.* [54], which can be attributed to the relatively small force demands compared to actuator capacity.

7 FORMULATION OF NUMERICAL INTEGRATION METHODS

Numerical integration algorithms are time stepping procedures used to solve the equation of motion that update the states (displacements, velocities, and accelerations) in the numerical simulation module (Figure 1-2). Similar integration methods are used in both hybrid and purely numerical simulations. In a hybrid simulation, they are also used to calculate the desired displacement vector of each simulation step. The application of numerical integration procedures to hybrid simulations is often more challenging due to the involvement of nonlinear experimental substructures and experimental errors.

This chapter presents the numerical integration methods that are widely used in hybrid simulations and their accuracy and stability characteristics. The general formulations of common numerical integration methods are presented first, and the modifications required for their implementation to hybrid simulation are discussed in Section 7.2. This chapter also presents a formulation for the separation of rate-dependent effects and extension to nonlinear numerical models. This chapter concludes with a

short discussion on the actual performance of integration methods in hybrid simulations including nonlinearities and experimental errors. Later in Chapter 8, new numerical integration procedures are proposed with improved accuracy and stability properties that utilize the formulations presented in this chapter.

7.1 NUMERICAL INTEGRATION FORMULATIONS

Formulations of the most common numerical integration methods are presented in this section. These methods are primarily developed to solve the following equation of motion:

$$\mathbf{Ma} + \mathbf{Cv} + \mathbf{Kd} = -\mathbf{M}\boldsymbol{\iota}\ddot{u}_g \tag{7.1}$$

where $\mathbf{M} = \mathbf{M}^{\mathrm{t}}$ (the total mass matrix) for a purely numerical simulation. This equation is developed for a linear system response; nonlinear numerical models will be considered in Section 7.2.3. The Newmark's Beta method, and Hilber, Hughes and Taylor's Alpha method are described in the following sections.

7.1.1 NEWMARK'S BETA METHOD

One of the most widely used numerical integration method was introduced by Newmark [118]. This method adopts the trapezoidal integration rule, and attempts to satisfy the following equations at the n^{th} integration step:

$$\mathbf{Ma}_n + \mathbf{Cv}_n + \mathbf{Kd}_n = \mathbf{M}\boldsymbol{\iota}\ddot{u}_g\left(t_n\right) \tag{7.2}$$

$$\mathbf{d}_n = \mathbf{d}_{n-1} + \Delta t\,\mathbf{v}_{n-1} + \left(\frac{1}{2} - \beta\right)(\Delta t)^2\,\mathbf{a}_{n-1} + \beta(\Delta t)^2\,\mathbf{a}_n \tag{7.3}$$

$$\mathbf{v}_n = \mathbf{v}_{n-1} + (1-\gamma)\Delta t\,\mathbf{a}_{n-1} + \gamma\,\Delta t\,\mathbf{a}_n \tag{7.4}$$

in which t is time and parameters β and γ define the variation of acceleration over a time step and affect the stability and accuracy of the algorithm. Selection of $\gamma = 1/2$ and $1/6 \le \beta \le 1/4$ are normally recommended to achieve the best stability and accuracy from Newmark's method [119]. Unless β is selected to be zero (explicit integration scheme), iterations are normally required to solve the implicit equations, as the unknown acceleration vector $\mathbf{a}_n$ appears at the right side of Equation (7.3). Only in linear systems, it is possible to avoid iterations by rearranging these equations to directly give the unknown states at step n. Newmark's method is stable if:

$$\omega_n \Delta t \le \frac{\sqrt{2}}{\sqrt{\gamma - 2\beta}} \tag{7.5}$$

where ω_n is the largest natural frequency of the system. Selection of $\gamma = 1/2$ and $\beta = 1/4$ results in an unconditionally stable implicit solution scheme with the assumption of constant acceleration in each integration step. For linear acceleration variations within a time step the parameters should be $\gamma = 1/2$ and $\beta = 1/6$. In this case, the stability condition becomes $\omega_n \Delta t \le 3.46$. For an explicit scheme equivalent to the central difference method ($\gamma = 1/2$ and $\beta = 0$), the time step should satisfy $\omega_n \Delta t \le 2$ for stability. In order to achieve adequate accuracy in the results, it is generally required to restrict the time step to about one-tenth of the natural period, which is generally more restrictive than the above-mentioned stability conditions for SDF systems. In MDF

systems, however, large natural frequencies of higher modes often make it necessary to use unconditionally stable schemes.

7.1.2 HILBER, HUGHES AND TAYLOR'S ALPHA METHOD

It is generally considered that the contribution of higher natural vibration modes to seismic structural response is significantly smaller than the first few modes. However small truncation errors in numerical simulation may lead to spurious excitation of higher modes, which in turn, may result in instability of the simulation. This issue becomes more important in hybrid simulations, where the experimental errors and measurement noise introduce high-frequency error signals in the system. Introduction of numerical damping in high-frequency modes is an effective way to improve the stability of the simulation by preventing spurious high-frequency excitations.

Numerical damping can be introduced in the Newmark method at the expense of degrading the order of accuracy [120]. The Alpha-method, introduced by Hilber *et al.* [59], overcomes this limitation. In this method, the time-discrete equation of motion is modified as follows:

$$\mathbf{M}\mathbf{a}_n + \mathbf{C}\mathbf{v}_n + \mathbf{K}\mathbf{d}_n + \alpha\left[\mathbf{C}\left(\mathbf{v}_n - \mathbf{v}_{n-1}\right) + \mathbf{K}\left(\mathbf{d}_n - \mathbf{d}_{n-1}\right)\right] = \mathbf{M}\boldsymbol{\iota}\ddot{u}_g\left(t_n + \alpha\Delta t\right) \tag{7.6}$$

where α is a parameter that controls the numerical damping. Other finite difference formulas, Equations (7.3) and (7.4), are retained in this integration method. Selection of $-1/3 \le \alpha \le 0$, $\gamma = (1-2\alpha)/2$ and $\beta = (1-\alpha)^2/4$ results in an unconditionally stable, second-order accurate scheme. With the selection of $\alpha = 0$ this formulation reduces to that of the Newmark's Beta method. Decreasing α from zero increases the amount of

numerical energy dissipation in higher modes, with small effects on lower-frequency modes of the system.

7.2 *NUMERICAL INTEGRATION FOR HYBRID SIMULATION*

Existence of an experimental restoring force slightly changes the equation of motion from purely numerical form of Equation (7.1) to the hybrid form of Equation (1.1), repeated here:

$$\mathbf{Ma} + \mathbf{Cv} + \mathbf{Kd} + \mathbf{r} = -\mathbf{M}^{\mathrm{t}}\boldsymbol{\iota}\ddot{u}_g \tag{7.7}$$

The experimental restoring force vector $\mathbf{r}$ may include strain-dependent, damping, or inertial forces. As a result, the time-discrete equation of motion in Alpha method will be modified as follows:

$$\begin{aligned}&\mathbf{Ma}_n + \mathbf{Cv}_n + \mathbf{Kd}_n + \mathbf{r}_n \\ &+ \alpha\left[\mathbf{r}_n - \mathbf{r}_{n-1} - \mathbf{M}^{\mathrm{e}}\left(\mathbf{a}_n - \mathbf{a}_{n-1}\right) + \mathbf{C}\left(\mathbf{v}_n - \mathbf{v}_{n-1}\right) + \mathbf{K}\left(\mathbf{d}_n - \mathbf{d}_{n-1}\right)\right] = \mathbf{M}^{\mathrm{t}}\boldsymbol{\iota}\ddot{u}_g\left(t_n + \alpha\Delta t\right)\end{aligned} \tag{7.8}$$

where $\mathbf{M}^{\mathrm{e}} = \mathbf{M}^{\mathrm{t}} - \mathbf{M}$ is the experimental mass matrix. The equivalent time-discrete equation of motion for Newmark's method can be obtained by setting $\alpha = 0$ in the above equation. Other finite difference equations for updating displacement and velocity vectors (Equations (7.3) and (7.4)) can be used without change in numerical and hybrid simulations.

Note that in Equation (7.8) the inertial force contribution of the experimental mass to the measured restoring force should be removed from the increments multiplied by α. This requires the knowledge of the experimental mass matrix when α is non-zero. The mass present in experimental substructures is generally selected to be very small

compared to total mass, to reduce the power requirements and achieve better actuator performance. A series of numerical simulations to study the effects of mass estimation errors on the simulation results is presented in Section 7.2.2.

7.2.1 SEPARATION OF RATE-DEPENDENT PROPERTIES

In Equation (7.8), matrices $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ (without superscripts) are properties of the numerical substructure in real-time simulations. When non-unity time scales are used, the experimentally developed rate-dependent forces will also be scaled. To account for this change in the components of the experimental restoring force, the rate-dependent numerical matrices should be modified. For example, the total inertial force in a fast simulation with a time scale S_t is:

$$\mathbf{f}_{\text{inertia}}^{\text{t}} = \mathbf{M}\mathbf{a} + \mathbf{M}^{\text{e}}\frac{\mathbf{a}}{S_t^2} \tag{7.9}$$

which is equivalent to a real-time simulation with the following experimental mass matrix:

$$\mathbf{M}^{\text{e}} = \frac{1}{S_t^2}\mathbf{M}_{\text{actual}}^{\text{e}} \tag{7.10}$$

where $\mathbf{M}_{\text{actual}}^{\text{e}}$ is the actual physical mass. As a result, the numerical mass matrix should also be modified:

$$\mathbf{M} = \mathbf{M}_{\text{actual}} + \left(1 - \frac{1}{S_t^2}\right)\mathbf{M}_{\text{actual}}^{\text{e}} \tag{7.11}$$

in which $\mathbf{M}_{\text{actual}} = \mathbf{M}^{\text{t}} - \mathbf{M}^{\text{e}}_{\text{actual}}$ is the actual numerical mass. The above equations suggest that the effect of experimental mass quickly diminishes as the time scale increases. As an extreme case, the analytical mass should be equal to the total mass in a slow (conventional pseudo-dynamic) test. A similar consideration should be taken into account for damping if present in the experimental substructure of a simulation with a reduced time scale:

$$\mathbf{C} = \mathbf{C}_{\text{actual}} + \left(1 - \frac{1}{S_t}\right)\mathbf{C}^{\text{e}}_{\text{actual}} \tag{7.12}$$

where $\mathbf{C}_{\text{actual}}$ and $\mathbf{C}^{\text{e}}_{\text{actual}}$ are actual numerical and experimental damping matrices, respectively. One problem in using Equation (7.12) is that accurate estimation of experimental damping properties of the experimental substructure may be difficult. In these cases, a real-time hybrid simulation ($S_t = 1$) eliminates the need for estimation of this damping matrix, and best captures the rate-dependent behavior of the experimental substructure directly in the measurements.

7.2.2 EFFECTS OF EXPERIMENTAL MASS ESTIMATION ERRORS

In the Alpha-method, the experimental mass matrix is necessary if Equation (7.8) is used with non-zero α, even in real-time simulations. Accurate estimation of experimental mass is often possible through free vibration tests or direct measurements of the experimental setup. In order to assess the effects of mass estimation errors, the numerical models of hybrid simulation described in Chapter 3 have been used to examine the sensitivity of simulation results to these errors.

Figure 7-1 shows the final energy balance errors of two separate sets of simulations of a two-degree-of-freedom model (shown in Figure 5-6) with $\alpha = -1/3$, in which 1% and 5% of total mass is considered as the experimental mass. For each of these simulations, experimental mass in numerical analysis is slightly modified from its actual value. As shown, the alterations of energy balance are generally small, and more importantly, random. Hence, it can be concluded that the estimation uncertainties of experimental mass have small effects on the simulation, comparable to the order of system randomness in instrumentation noise and other experimental errors. However, this is only valid for small amounts of experimental mass, which is commonly the case in hybrid simulations. It should be mentioned that the effects of above-mentioned variations on displacement histories are negligible.

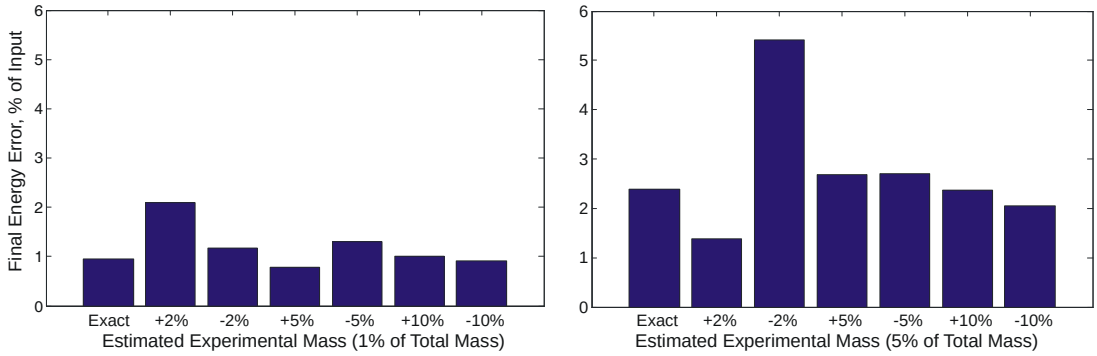

Figure 7-1 Effect of mass estimation error on the overall energy balance of simulation for cases of $\mathbf{M}^{e} = 0.01\mathbf{M}$ and $\mathbf{M}^{e} = 0.05\mathbf{M}$.

7.2.3 NONLINEAR NUMERICAL MODELS

The formulations presented in the preceding sections can account for the nonlinearities of experimental substructures. These formulations can be easily modified to an incremental form for application to hybrid simulations with nonlinear numerical substructures. To account for a nonlinear restoring force developed in numerical

substructure, the stiffness-dependent force in Equation (7.7) can be replaced by a generic force vector, as in:

$$\mathbf{Ma} + \mathbf{Cv} + \mathbf{r}^{\mathrm{a}} + \mathbf{r} = -\mathbf{M}^{\mathrm{t}}\boldsymbol{\iota}\ddot{u}_g \tag{7.13}$$

where $\mathbf{r}^{\mathrm{a}}$ is the numerical restoring force. Adopting a piecewise linear behavior for numerical substructure, the restoring force can be obtained from an incremental relation, such as:

$$\mathbf{r}_n^{\mathrm{a}} = \mathbf{r}_{n-1}^{\mathrm{a}} + \mathbf{K}_n\left(\mathbf{d}_n - \mathbf{d}_{n-1}\right) \tag{7.14}$$

with $\mathbf{K}_n$ being the tangent stiffness matrix of numerical substructure in step n. Substituting the above in Equation (7.8), the time-discrete equation of motion becomes:

$$\begin{aligned}&\mathbf{Ma}_n + \mathbf{Cv}_n + \mathbf{r}_{n-1}^{\mathrm{a}} + \mathbf{K}_n\left(\mathbf{d}_n - \mathbf{d}_{n-1}\right)\\ &+ \alpha\left[\mathbf{r}_n - \mathbf{r}_{n-1} - \mathbf{M}^{\mathrm{e}}\left(\mathbf{a}_n - \mathbf{a}_{n-1}\right) + \mathbf{C}\left(\mathbf{v}_n - \mathbf{v}_{n-1}\right) + \mathbf{K}_n\left(\mathbf{d}_n - \mathbf{d}_{n-1}\right)\right] = \mathbf{M}^{\mathrm{t}}\boldsymbol{\iota}\ddot{u}_g\left(t_n + \alpha\Delta t\right)\end{aligned} \tag{7.15}$$

Given that only a force vector $\mathbf{r}_{n-1}^{\mathrm{a}}$ that remains constant in step n is added to the equation of motion, the above modification does not have a significant effect on the solution algorithm. Only the numerical tangent stiffness matrix should be updated in each step based on element internal forces and plasticity states. Based on this fact, and to simplify the formulations presented in this dissertation, numerical substructure is always assumed to be linear. The above procedure can be followed to modify these relations for application to hybrid simulations with nonlinear numerical substructures.

7.3 EXPLICIT INTEGRATION

Explicit integration methods have been widely used in hybrid simulations due to their simplicity. As mentioned earlier, setting $\beta = 0$ in Equation (7.3) results in an explicit scheme, in which the calculation of desired displacement at step n does not require the knowledge of acceleration vector at this step:

$$\mathbf{d}_n = \mathbf{d}_{n-1} + \Delta t\, \mathbf{v}_{n-1} + \frac{1}{2}(\Delta t)^2\, \mathbf{a}_{n-1} \tag{7.16}$$

The displacement vector given by Equation (7.16) is transformed to the actuator (local) coordinate system using displacement transformation matrix $\mathbf{T}$:

$$\mathbf{d}_n^{\mathrm{l}} = \mathbf{T}\mathbf{d}_n \tag{7.17}$$

and applied on the experimental substructures. The measured experimental restoring force can then be transformed to global coordinates:

$$\mathbf{r}_n = \mathbf{T}^{\mathrm{T}}\mathbf{r}_n^{\mathrm{l}} \tag{7.18}$$

and used in Equations (7.4) and (7.8) to calculate velocity and acceleration vectors of step n. Note that a superscript l denotes quantities in actuator local coordinate system. Equations (7.4) and (7.8) can be combined to directly give the new acceleration vector:

$$\mathbf{a}_n = \mathbf{A}^{-1}\left\{\begin{array}{l}\mathbf{M}^{\mathrm{t}}\iota\ddot{u}_g\left(t_n + \alpha\Delta t\right) \\ \quad -\left[\begin{array}{l}\mathbf{C}\left(\mathbf{v}_{n-1} + (1+\alpha)\dfrac{\Delta t}{2}\mathbf{a}_{n-1}\right) \\ +(1+\alpha)\left(\mathbf{K}\mathbf{d}_n + \mathbf{r}_n\right) + \alpha\left(\mathbf{M}^{\mathrm{e}}\mathbf{a}_{n-1} - \mathbf{K}\mathbf{d}_{n-1} - \mathbf{r}_{n-1}\right)\end{array}\right]\end{array}\right\} \tag{7.19}$$

where

$$\mathbf{A} = \mathbf{M} + (1+\alpha)\frac{\Delta t}{2}\mathbf{C} - \alpha\mathbf{M}^{e} \tag{7.20}$$

The above acceleration vector can then be used in Equation (7.4) to find velocity. The simulation continues to the next step by calculating the next displacement vector using Equation (7.16). A simulation algorithm using an explicit integration method is summarized in Table 7-1. Note that the procedures to minimize the experimental errors (such as identification of system dynamics, signal generation, delay compensation, and signal conditioning and correction) are not included in the concise algorithm of Table 7-1 and all other integration algorithms presented in this and the following chapters.

Table 7-1 Explicit integration algorithm.

Simulation Steps	Equations and Other Information
• **Calculate desired displacement vector at step** n	**Equation (7.16)**
• **Transform displacements to actuator coordinate system**	**Equation (7.17)**
• **Apply desired displacement vector on experimental substructure**	
• **Measure experimental restoring forces**	
• **Transform measured forces to global coordinate system**	**Equation (7.18)**
• **Calculate the acceleration and velocity of step** n	**Equations (7.4) and (7.8), or (7.4), (7.20), and (7.19)**
• **Set** $n+1 \rightarrow n$ **and go to the next integration step**	

7.3.1 CHARACTERISTICS

Explicit integration methods do not require iterations, and have the least communication requirements: the communication of interface forces and displacements occurs only once in each integration step. With small integration time steps, the accuracy of explicit methods can be comparable to implicit ones for structures with a few degrees of freedom. Further, since these integration algorithms directly use the measurements without any modifications, they do not have the potential to erroneously alter the

observed behavior of the experimental substructure; that is, the observed and converged hysteretic behaviors of the experimental substructure are identical in this approach (see Section 5.5).

However, in large MDF structures the stability condition of $\omega_n \Delta t \leq 2$ for the Newmark explicit method becomes very restrictive due to the existence of high-frequency modes. This stability condition may also require very small time steps in stiff systems. Use of very small time steps is not normally desirable in hybrid simulations, since it increases the communication needs, intensifies error propagation problems, and increases the sensitivity of the system to delay. Furthermore, small time steps may lead to small displacement increments that are within the actuator noise level. For this reason, conditionally stable integration methods such as Newmark's explicit approach are not suitable for hybrid simulations of complex structural systems with high-frequency modes.

Using Equation (7.8) in the above-mentioned explicit integration algorithm ensures the satisfaction of equilibrium in each integration step. Hence, its integral form, the energy balance equation is also satisfied throughout the simulation. In order to understand the nature of errors introduced by an explicit integration algorithm, it should be noted that Equation (7.16) is exact only when the acceleration remains constant throughout the integration step. That is, the kinematic relationship between displacement and acceleration is correct only if the acceleration $\mathbf{a}_n$ obtained from Equation (7.8) is equal to $\mathbf{a}_{n-1}$, which is rarely the case. As a result, the kinematic relations among displacement, velocity and acceleration are not exactly maintained

using an explicit procedure, and kinematic errors may increase in consecutive integration steps.

In order to take the kinematic relations into account in the energy balance of the simulation, the velocity vectors should be replaced by the derivatives of the displacement vector, as described in Section 5.6; that is, damping and kinetic energies (E_{D} and E_{K}) should be calculated using Equations (5.23) and (5.24). This formulation will be used later to examine the accuracy and stability of explicit integrators.

7.4 OPERATOR-SPLITTING METHOD

The use of operator-splitting integration method in hybrid simulation was introduced by Nakashima *et al.* [14]. The general form of the procedure for alpha-method formulation is based on Equations (7.3), (7.4) and (7.8), solved using a predictor-corrector formulation. In the predictor step, the displacement vector is first obtained from Equation (7.3) by temporarily eliminating the terms that are dependent on the new acceleration vector $\mathbf{a}_n$:

$$\tilde{\mathbf{d}}_n = \mathbf{d}_{n-1} + \Delta t\, \mathbf{v}_{n-1} + \left(\frac{1}{2} - \beta\right)(\Delta t)^2\, \mathbf{a}_{n-1} \tag{7.21}$$

The predictor displacement vector is transformed to actuator coordinate system and applied on the experimental substructure. The restoring force $\tilde{\mathbf{r}}_n^j$ is then measured in actuator coordinate system. Next, in the corrector step, the displacement vector is updated to satisfy the implicit expression of Equation (7.3):

$$\mathbf{d}_n = \tilde{\mathbf{d}}_n + \beta(\Delta t)^2\, \mathbf{a}_n \tag{7.22}$$

As a result of this change in displacement vector, the restoring force vector should also be modified. Since estimation of a tangent stiffness matrix of experimental substructure may be difficult in an online simulation, the initial stiffness matrix of experimental substructure $\mathbf{K}^{\mathrm{e,l}}$ is used in this correction:

$$\mathbf{r}_n^{\mathrm{l}} = \tilde{\mathbf{r}}_n^{\mathrm{l}} + \mathbf{K}^{\mathrm{e,l}}\left(\mathbf{d}_n^{\mathrm{l}} - \tilde{\mathbf{d}}_n^{\mathrm{l,m}}\right) \tag{7.23}$$

in which $\tilde{\mathbf{d}}_n^{\mathrm{l,m}}$ is the measured displacement vector in local coordinate system. Equation (7.23) not only updates the force vector due to the displacement correction of Equation (7.22), but also attempts to correct for actuator tracking errors by using $\tilde{\mathbf{d}}_n^{\mathrm{l,m}}$ in place of $\tilde{\mathbf{d}}_n^{\mathrm{l}}$.

Combining Equations (7.4), (7.8), (7.22) and (7.23), the acceleration vector at step n can be calculated from:

$$\mathbf{a}_n = \mathbf{B}^{-1}\left\{\begin{array}{l}\mathbf{M}^{\mathrm{t}}\boldsymbol{\iota}\ddot{u}_g\left(t_n + \alpha\Delta t\right) - \mathbf{C}\left[\mathbf{v}_{n-1} + (1+\alpha)(1-\gamma)\Delta t\,\mathbf{a}_{n-1}\right] \\ \quad + \alpha\left(\mathbf{r}_{n-1} + \mathbf{K}\mathbf{d}_{n-1} - \mathbf{M}^{\mathrm{e}}\mathbf{a}_{n-1}\right) - (1+\alpha)\left[\tilde{\mathbf{r}}_n + \mathbf{K}\tilde{\mathbf{d}}_n + \mathbf{T}^{\mathrm{T}}\mathbf{K}^{\mathrm{e,l}}\left(\tilde{\mathbf{d}}_n^{\mathrm{l}} - \tilde{\mathbf{d}}_n^{\mathrm{l,m}}\right)\right]\end{array}\right\} \tag{7.24}$$

where:

$$\mathbf{B} = \mathbf{M} - \alpha\mathbf{M}^{\mathrm{e}} + \gamma\Delta t(1+\alpha)\mathbf{C} + \beta\Delta t^2(1+\alpha)\left(\mathbf{K} + \mathbf{K}^{\mathrm{e}}\right) \tag{7.25}$$

can be evaluated prior to simulation. In the above relations, $\mathbf{K}^{\mathrm{e}}$ is the initial experimental stiffness matrix in global coordinate system. This stiffness matrix can be obtained from the local stiffness matrix of the experimental substructure using displacement transformation matrix $\mathbf{T}$:

$$\mathbf{K}^{\mathrm{e}} = \mathbf{T}^{\mathrm{T}}\mathbf{K}^{\mathrm{e,l}}\mathbf{T} \tag{7.26}$$

After calculation of acceleration vector from Equation (7.24), Equations (7.22), (7.23), and (7.4) can be used to determine $\mathbf{d}_n$, $\mathbf{r}_n$ and $\mathbf{v}_n$. The next integration step begins with calculation of next predictor displacement using Equation (7.21). A summary of this integration algorithm is listed in Table 7-2.

Table 7-2 Operator-splitting integration algorithm.

Simulation Steps	Equations and Other Information
• **Calculate predictor displacement vector at step** n	**Equation (7.21)**
• **Transform displacements to actuator coordinate system**	$\tilde{\mathbf{d}}_n^1 = \mathbf{T}\tilde{\mathbf{d}}_n$
• **Apply predictor displacement vector on experimental substructure**	
• **Measure experimental displacements and restoring forces**	
• **Calculate the acceleration vector of step** n	**Equations (7.24) and (7.25)**
• **Calculate new displacement, velocity, and force vectors**	**Equations (7.22), (7.23), and (7.4)**
• **Set** $n+1 \rightarrow n$ **and go to the next integration step**	

7.4.1 CHARACTERISTICS

By modifying the predictor displacement in Equation (7.22) or (8.2) to satisfy the implicit formulation, the operator-splitting integration method ensures accurate kinematic relationships among displacement, velocity and acceleration. This improvement over the explicit integration methods have been shown to result in better stability properties. In particular, this integration method is considered unconditionally stable as long as the experimental nonlinearity is of softening type [14]. It has also been shown that acceptable accuracy can be obtained by selecting an integration time step sufficiently shorter than the dominant natural periods of the test structure. It is not necessary to consider the high-frequency modes with negligible contributions to the response in the selection of time step.

The operator-splitting integration method also does not require iterations, and has basic communication needs, which greatly facilitate its implementations. These features, along with its improved stability in test structures with many degrees of freedom, have made this method very popular in hybrid simulations. The only additional requirements of this integration method in comparison with the explicit approach described in Section 7.3, are the need for initial stiffness matrix of the experimental substructure, and slightly increased computational cost. The initial stiffness can normally be measured using low-amplitude static tests before the main simulation. The increased computational cost is also small compared to the capabilities of currently-available processing tools.

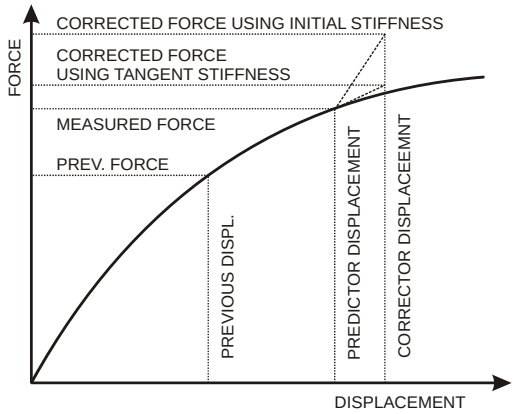

Figure 7-2 Comparison of the use of initial stiffness and tangent stiffness for correction of the restoring force.

In addition to establishing proper kinematic relationships among displacements, velocities and accelerations, the operator-splitting integration method satisfies the equation of motion in each integration step. Hence, the energy balance (Equation (5.18)) is also maintained throughout the simulation. However, the force correction is only approximated using initial stiffness (see Figure 7-2), and hence, the experimental energy dissipation conceived by the equation of motion will be different than the actual energy

dissipated in the experiment. Consequently, when the actual experimental energy term E_{E} is used in energy balance equation, the effects of this approximation will be reflected in the energy error given by Equation (5.22). This is an important error, as one of the major goals of a hybrid simulation is the identification of the actual behavior of the experimental substructure.

It should be noted that the correction of the restoring force vector in Equation (7.23) does not account for experimental rate-dependent forces. This limits the applications of operator-splitting method to experimental substructures that are not essentially rate-dependent. Moderate rate-dependencies can still be handled by this integration algorithm, as the experimental damping is ignored only in the corrector step, and predictor and corrector displacements are normally close. An extension of this method to highly rate-dependent experimental substructure is studied in [74]. Such an extension, however, requires an accurate knowledge of experimental damping properties.

7.4.2 OTHER INTEGRATION METHODS USING INITIAL STIFFNESS MATRIX

Existence of a tangent stiffness matrix greatly facilitates the integration task by eliminating the need for iterative solution of equation of motion and finite difference kinematics equations. However, due to the difficulties associated with online estimation of experimental tangent stiffness matrix, often initial stiffness matrix has been used as an approximation in many other integration methods [66-71, 73]. It has been shown that these integration algorithms have better error propagation and stability properties compared to explicit methods. However, one important problem in deriving these

conclusions is that they are based on conventional stability and error propagation analyses, which normally do not consider the nonlinearities and experimental errors. In addition, admitting the initial stiffness matrix instead of the tangent stiffness clearly reduces the accuracy of the algorithm in nonlinear systems. The errors resulting from initial stiffness approximation, along with the effects of system nonlinearities and experimental errors are further studied in Sections 7.6 and 8.4.3.

7.5 IMPLICIT INTEGRATION METHODS FOR HYBRID SIMULATION

Implicit integration methods have also been proposed in literature for hybrid simulation as discussed in Chapter 2. A number of these methods utilize implicit formulations by introducing feedback loops involving SDF experimental substructures [3, 28, 29, 60, 61], and more recently, MDF substructures [62]. These procedures have onerous communication requirements between experimental and numerical substructures, and require specialized control strategies to avoid unwanted displacement reversals during iterations on experimental substructures. For these reasons, implicit integration methods with physical iterations have not been widely used in hybrid simulations. In order to mitigate these difficulties, other integration methods have been introduced that apply implicit iterations only in numerical substructure by assuming the interface forces as constant external forces [64, 65]. These methods can be compared to operator-splitting method, but without correction of the experimental restoring force to account for the changes in displacement vector. More advanced procedures such as one introduced by Pan *et al.* [13] measure the instantaneous behavior of the test specimen to estimate the tangent stiffness for full implementation of implicit

integration equations. This procedure was developed for SDF experimental substructures. A more complete discussion on these integration methods is available in Chapter 2.

7.6 PERFORMANCE OF NUMERICAL INTEGRATION METHODS – EXPERIMENTAL ERRORS AND NONLINEAR PROBLEMS

In numerical problems, the accuracy and stability properties of integration methods are usually carried out for linear SDF problems. However, hybrid simulations are usually carried out to evaluate the nonlinear and complex behavior of the experimental substructures. It can be shown that numerical integration methods with unconditional stability can become unstable when applied to nonlinear problems with large time steps [58]. This can occur due to the approximate solution schemes such as iterative methods, and the discrepancy between the actual nonlinear structural behavior and the piecewise linear behavior detected by the integration algorithm. These errors increase as longer integration time steps are selected, and may render the simulation unstable by adding erroneous energy to the system.

In addition to the above-mentioned nonlinearities, the experimental errors in hybrid simulations significantly affect the stability and accuracy properties of the numerical integration procedures. Experimental errors may destabilize the simulation well before reaching the stability limits of the utilized numerical integration methods. That is, the stability and accuracy of hybrid simulations cannot be determined based on the numerical properties of the integration algorithm alone.

In order to include the above-mentioned errors in the assessment of accuracy and stability of the numerical integration methods introduced in this and the following chapters, they are often compared using the energy balance error. This error measure was introduced in Chapter 5 to take both numerical and experimental errors into account; as mentioned in Chapter 5, the energy balance error includes the discrepancies between actual and converged hysteretic behaviors of the experimental substructures, in addition to the errors of the numerical integration procedure. A limitation of this approach is that this error measure can only be evaluated to show the stability and accuracy of the integration algorithm for each particular simulation, and the results cannot be extended to other simulations in general. The energy-based error measures are used in the next chapter to demonstrate the advantages of the proposed integration methods.

8 IMPROVED NUMERICAL INTEGRATION METHODS

Two improved numerical integration methods for hybrid simulation are introduced in this chapter. These integration methods use the formulations presented in the previous chapter, as well as those of the improved operator-splitting method introduced in Section 8.2. The first method implements an implicit formulation to solve the equation of motion when possible, and defaults to an explicit or operator-splitting scheme to continue the simulation in case of convergence failure. The second method is based on the operator-splitting integration scheme. In this method, an experimental tangent stiffness is estimated in each integration step and used in the operator-splitting method to improve its accuracy for testing nonlinear systems.

Before introducing the improved integration methods, a short discussion on the hybrid simulation implementation issues is presented in the following section. It is attempted to demonstrate the difficulties associated with numerical integration in hybrid simulation leading to the currently-available simplified integration methods, and to highlight the need for improved numerical integration methods.

8.1 IMPORTANCE OF IMPROVED NUMERICAL INTEGRATION METHODS

In hybrid simulations, the path-dependent behavior of experimental substructures does not allow for direct implementations of iterative implicit integration procedures. Physical iterations may unexpectedly damage the experimental substructure, or erroneously dissipate energy. Further, the convergence of the iterative solution scheme cannot be guaranteed in nonlinear problems. Unlike numerical simulations, it may be impossible to "roll back" to the last converged solution and continue the simulation with different parameters, due to physical damages. In addition, unconverged solutions may lead to excessively large displacements that can severely damage the experimental setup with significant replacement costs. In any case, if a simulation is stopped because of convergence failure, it is likely for the experimental substructure to have already sustained unrecoverable damages.

In order to avoid physical iterations and the difficulties associated with them, a numerical model of the experimental substructure (such as experimental tangent stiffness matrix) can be used. This numerical model can provide estimates of forces corresponding to iterative displacements, or it can be used to solve the state equations without iterations. This also reduces the required communications between numerical and experimental substructures. However, due to the experimental errors and measurement noise, estimation of experimental tangent stiffness matrix may be difficult, and often, the initial experimental stiffness matrix has been used for this purpose.

The above are the most important underlying reasons for the widespread use of explicit and operator-splitting integration methods (presented in the previous chapter)

in hybrid simulations. As mentioned before, these methods are either conditionally stable, leading to stringent time step requirements for MDF structures, or have limited accuracy for testing highly nonlinear systems. For this reason, development of improved numerical integration methods has been of great interest in recent years.

In addition to providing better stability and accuracy properties, the improved integration methods should be easily applicable to hybrid simulation. Particularly, in fast and real-time simulations, the numerical simulation task and the communication of the results must be completed within a small period. That is, the solution algorithm should be very efficient, and be able to update the states and prepare the results within a finite amount of time. If a geographically distributed hybrid simulation is intended, the integration procedure should also limit the amount of communications that are necessary in each integration step. In summary, improved numerical integration procedures for hybrid simulation should possess the following characteristics:

- have improve stability for using longer time steps and testing stiff systems,
- have better accuracy by capturing the instantaneous behavior of experimental substructure instead of using its initial stiffness,
- require minimal information about the experimental setup, which is normally unavailable before hybrid simulations,
- eliminate iterations on experimental substructures to prevent unexpected damages,

- limit the communication and processing costs for fast and distributed experiments,
- guarantee continuous simulation without interruptions from crashing of numerical integration algorithm.

In the remainder of this chapter, two improved numerical integration methods are introduced for hybrid simulation. It is attempted to develop hybrid simulation integration procedures that address most of the above-mentioned implementation issues.

8.2 OPERATOR-SPLITTING METHOD WITH IMPROVED PREDICTOR

This section introduces a modification to the conventional operator-splitting procedure to improve its accuracy. In this method, instead of eliminating the implicit terms of Equation (7.3), the predictor displacement vector is calculated from an explicit expression by temporarily setting $\beta = 0$ in Equation (7.3):

$$\tilde{\mathbf{d}}_n = \mathbf{d}_{n-1} + \Delta t\, \mathbf{v}_{n-1} + \frac{1}{2}(\Delta t)^2\, \mathbf{a}_{n-1} \tag{8.1}$$

Consequently, in the corrector step the displacement should be modified as follows:

$$\mathbf{d}_n = \tilde{\mathbf{d}}_n + \beta(\Delta t)^2 (\mathbf{a}_n - \mathbf{a}_{n-1}) \tag{8.2}$$

in which the correction term $\beta\Delta t^2 (\mathbf{a}_n - \mathbf{a}_{n-1})$ is normally smaller than $\beta\Delta t^2\mathbf{a}_n$ used in the original operator-splitting formulation, thus providing more accurate predictor displacements. When the difference between the corrector and predictor displacement decreases, the effect of initial stiffness matrix approximation in Equation (7.23) also

reduces, and better accuracy will be achieved in capturing the behavior of experimental substructures. This integration method is termed herein the improved operator-splitting method. When this form of operator-splitting method is used, Equation (7.24) also changes to:

$$\mathbf{a}_n = \mathbf{B}^{-1}\left\{\begin{array}{l}\mathbf{M}^{\mathrm{t}}\boldsymbol{\iota}\ddot{u}_g\left(t_n+\alpha\Delta t\right)-\mathbf{C}\left[\mathbf{v}_{n-1}+(1+\alpha)(1-\gamma)\Delta t\,\mathbf{a}_{n-1}\right]\\ +\alpha\left(\mathbf{K}\mathbf{d}_{n-1}+\mathbf{r}_{n-1}-\mathbf{M}^{\mathrm{e}}\mathbf{a}_{n-1}\right)\\ -(1+\alpha)\left[\tilde{\mathbf{r}}_n+\left(\mathbf{K}+\mathbf{K}^{\mathrm{e}}\right)\left(\tilde{\mathbf{d}}_n-\beta\Delta t^2\mathbf{a}_{n-1}\right)-\mathbf{T}^{\mathrm{T}}\mathbf{K}^{\mathrm{e,l}}\tilde{\mathbf{d}}_n^{\mathrm{l,m}}\right]\end{array}\right\} \tag{8.3}$$

The improved integration algorithm is summarized in Table 8-1.

Table 8-1 Improved operator-splitting integration algorithm.

Simulation Steps	Equations and Other Information
• **Calculate predictor displacement vector at step** n	**Equation (8.1)**
• **Transform displacements to actuator coordinate system**	$\tilde{\mathbf{d}}_n^{\mathrm{l}} = \mathbf{T}\tilde{\mathbf{d}}_n$
• **Apply predictor displacement vector on experimental substructure**	
• **Measure experimental displacements and restoring forces**	
• **Calculate the acceleration vector of step** n	**Equations (8.3) and (7.25)**
• **Calculate new displacement, velocity, and force vectors**	**Equations (8.2), (7.23), and (7.4)**
• **Set** $n+1 \rightarrow n$ **and go to the next integration step**	

8.3 IMPLICIT OR EXPLICIT INTEGRATION STEPS FOR HYBRID SIMULATION

A new integration technique is proposed here, in which recent experimental measurements are used to capture the instantaneous behavior of experimental substructures in a fully implicit iterative scheme. The iterations are implemented numerically, without physical imposition of iterative displacements on the experimental substructures using the following procedure. First, the actuator command displacements are predicted using an explicit expression to load the experimental substructures.

Second, the displacements and forces measured through the load path are used in the iterative scheme to satisfy an implicit formulation. Since convergence cannot be guaranteed in every step of an iterative scheme, the procedure defaults to the explicit solution only in steps with failed convergence to continue the simulation. One advantage to this approach is that errors will not accumulate as in fully explicit methods because the implicit steps tend to minimize the unbalanced energy in the equation of motion. The improved accuracy and stability of the proposed integration procedure is demonstrated through experimental and numerical simulations.

The proposed integration algorithm is based on the Alpha method formulation described in Sections 7.1 and 7.2. This procedure is described through modification of an explicit integration scheme. Later in this section, the combination of implicit steps with operator-splitting integration method is also discussed.

In an explicit integration scheme, Equation (7.16) is first used to calculate the desired displacement at step n. This equation can be used to predict a displacement command signal for the actuator to load the experimental substructure, after considering the compensation of system delay and dynamics. For an explicit solution, the measured force vector, $\mathbf{r}_n$, corresponding to desired displacements $\mathbf{d}_n$ is then directly used in Equations (7.19) and (7.4) to determine acceleration and velocity at step n.

In the proposed implicit approach, the initial displacement vector $\mathbf{d}_n^0$ is assumed equal to the desired displacement from Equation (8.1), with predictor velocity given by $\mathbf{v}_n^0 = \mathbf{v}_{n-1} + \Delta t\, \mathbf{a}_{n-1}$. The experimental substructures are then loaded to the predicted

displacement, similar to the explicit procedure. Instead of using the force measurements directly to calculate acceleration and velocity vectors at the current step, the forces and displacements measured through the load path are used in an iterative scheme to correct the predicted displacement until the implicit equations (7.8), (7.3) and (7.4) are satisfied. The iterative procedure is repeated until a convergence criterion is satisfied, such as:

$$\frac{\left\|\mathbf{d}_n^i - \mathbf{d}_n^{i-1}\right\|}{\left\|\mathbf{d}_n^i\right\|} < \varepsilon \tag{8.4}$$

where ε is the convergence tolerance for the normalized displacement increment, and superscripts denote the iteration number.

The major challenge in implementing implicit integration algorithms in a hybrid simulation is that iterative displacement reversals may result in unrecoverable damage to experimental specimens or erroneous energy dissipation. Therefore, it is not advisable to measure experimental restoring forces, $\mathbf{r}_n^{i+1}$, by physically imposing the iterative displacements.

In order to avoid iterations on experimental substructures, a force estimation procedure for iterative displacements is followed, similar to that used in force correction approach described in Section 6.2.2. After imposing the predictor displacement determined by Equation (7.16), recent measurements are used to fit second-order polynomials to both measured displacement and force histories in local actuator coordinates (Equations (6.6) and (6.7)). As shown in Figure 6-4, these polynomials are used to estimate forces corresponding to each of the iterative displacements.

In each iteration, the trial displacements are determined in actuator coordinate system, but they are not physically imposed on the specimen. Instead, the displacement polynomial of each actuator is first used to estimate the time corresponding to the instant the target displacement was achieved. The corresponding force is then determined by replacing the computed time into the force polynomial. Therefore, the integration algorithm can be completed by addition of another step to the iterative scheme of Equations (7.8), (7.3) and (7.4), which estimates the restoring force vector $\mathbf{r}_n^i$ for the iterative displacements $\mathbf{d}_n^i$. It is evident that this procedure does not require additional communication between experimental and numerical, since the iterative force estimations are obtained from the fitted polynomials that can be executed numerically local to the integrator.

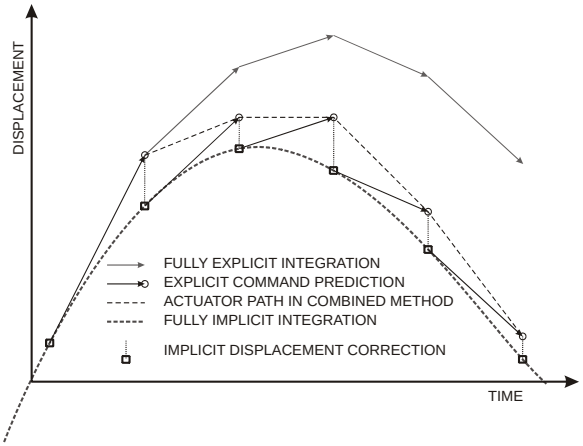

Figure 8-1 Combined implicit or explicit steps in comparison with fully implicit and fully explicit methods.

A schematic view of the performance of the combined implicit or explicit integration method is shown in Figure 8-1. It is demonstrated that the actuator follows the explicit desired displacement path, while an attempt is made to bring the displacement to the implicit result in the following step. Consequently, the actuator

displacement will remain close to the results of an implicit formulation and numerical errors will not accumulate as much compared to a fully explicit scheme. Of course, the actuator may not achieve the exact desired displacements as shown in Figure 8-1, and the polynomials should be fitted to measured (achieved) displacements and forces to capture the actual specimen behavior.

Polynomials fitted to force and displacement measurements provide an instantaneous relation between the restoring force and the actuator displacement [13]. As mentioned in Section 6.2.2, fitting time histories and relating force and displacement polynomials through the data acquisition time stamp has several advantages: (i) computation of fitted polynomial coefficients with equally spaced points is efficient; and (ii) the effects of specimen nonlinearities will be less pronounced on time histories compared to force-displacement curves. Further, the accuracy of fitted polynomials in representing the actual specimen behavior can be controlled by restricting the range of variations of time, thus avoiding excessively large extrapolations.

The time corresponding to each iterative displacement can be obtained from Equation (6.8) by replacing the iterative displacement for desired displacement:

$$t_n^i = \frac{-b_u \pm \sqrt{b_u^2 - 4a_u\left(c_u - d_n^i\right)}}{2a_u} \tag{8.5}$$

which can result in a complex value if actuator undershoots the displacement in a displacement reversal. In this case, the iterative displacement does not satisfy $4a_u d_n^i \geq 4a_u c_u - b_u^2$, and inserting the resulting complex-valued time in the fitted force polynomial results in a complex-valued force. Similar to the delay compensation method

described in Section 6.2.2, the iterative restoring force is approximated as the absolute value of the resulting complex value, as long as its imaginary component remains small. Admitting this approximation increases the number of successful implicit integration steps, which has been observed to improve the overall accuracy of the simulation.

8.3.1 CONVERGENCE ISSUES

As with most iterative schemes on non-linear systems, convergence cannot be guaranteed in each step, especially for a hybrid simulation that also involves experimental errors. The failed integration steps can be identified by detection of excessive time parameter variation, or convergence failure after maximum number of iterations. In a hybrid simulation, failure to converge to a solution that satisfies the implicit integration formulation could be detrimental to the experimental substructures. Stopping and restarting the simulation may not be possible if the experimental substructures are damaged during the simulation; hence, it is necessary to have alternate solution strategies that will allow the simulation to continue. To handle integration steps with failed convergence, it is proposed to revert to an explicit procedure by selecting the displacement of Equation (7.16) as the final solution for the step. The measured restoring force vector $\mathbf{r}_n$ is then directly used to determine acceleration and velocity vectors at step n using Equations (7.19) and (7.4). If the initial stiffness matrix of the system is available, an operator-splitting method can also be utilized in the steps with failed implicit iterations. In this case, the explicit displacement vector of Equation (7.16) is used as the predictor displacement; in the corrector step, the measured force vector $\mathbf{r}_n$ should be

used in Equation (8.3) for acceleration, followed by Equations (7.4) and (8.2) for velocity and corrector displacement.

It is shown in Figure 7-2 that using measurements in the corrector step will result in a better force estimate compared to using the initial stiffness matrix in operator-splitting method. Hence, the proposed integration method results in an improved agreement between actual and converged hysteretic behaviors of the experimental substructure, and reduces energy balance errors described in Section 5.4. It should be noted that the proposed procedure may have errors in force estimation due to experimental errors not shown in this figure. The simulation algorithm using the proposed integration method is summarized in Table 8-2.

Table 8-2 Numerical integration algorithm with combined implicit or explicit step.

Simulation Steps	Equations and Other Information
• **Calculate predictor displacement vector at step** n	**Equation (8.1)**
• **Transform displacements to actuator coordinate system**	$\tilde{\mathbf{d}}_n^1 = \mathbf{T}\tilde{\mathbf{d}}_n$
• **Apply predictor displacement vector on experimental substructure**	
• **Measure experimental displacements and restoring forces**	
• **Fit polynomials to most recent force and displacement measurements**	
• **Calculate the predictor velocity vector**	**Equation (7.4) with** $\gamma = 0$
• **Solve iteratively:**	
- **Using fitted polynomials, estimate forces corresponding to iterative displacements**	**Equations (7.8), (7.3), (7.4) and (8.4)**
- **Calculate new accelerations, velocities, and displacements**	
- **Check the convergence norm**	
• **Iterative solution scheme:**	
- **Failed: use explicit expressions to update accelerations and velocities, or perform a one-step correction using the initial stiffness (operator-splitting) to update accelerations, velocities, and displacements**	**Equations (7.19) and (7.4), or (8.3) and (7.4)**
- **Succeeded: admit the iterative solution**	
• **Set** $n+1 \rightarrow n$ **and go to the next integration step**	

8.3.2 CHARACTERISTICS

The proposed approach is similar to a fully implicit integration method for SDF substructures since the fitted polynomials provide an instantaneous relation between the measured displacements and restoring forces. However, for MDF experimental substructures, the coupling between actuators is not considered in the force estimation procedure. This is due to the fact that the force and displacement polynomials are fitted individually for each actuator, which is equivalent to ignoring the off-diagonal terms of the experimental tangent stiffness matrix. This should not have a dramatic effect on the simulation results, but may increase the required number of iterations to converge.

The proposed integration method is mainly for experimental substructures with dominant strain-dependent behavior as opposed to strain-rate or acceleration dependent behavior. This is mainly due to the procedure for estimation of forces, which is carried out for displacements without consideration of velocity or acceleration changes. In addition, when this integration method is combined with operator-splitting, the correction of feedback force in Equation (7.23) only considers displacement-dependent forces. Therefore, this procedure may not accurately capture the behavior of highly rate-dependent experimental substructures such as viscous damping devices. Nonetheless, it should be noted that these effects are only ignored in the iterative corrections, in which variations of the states are generally small. As a result, slight rate-dependent behavior of experimental substructures will not considerably affect the efficiency of this procedure.

Accurate estimation of forces in the iterative procedure is central to the accuracy and stability of the proposed integration procedure. In fact, the proper use of

experimental measurements improves the stability of the integration scheme over explicit methods, and provides better accuracy over the operator-splitting method in nonlinear systems. However, this is only possible if the fitted polynomials accurately capture the behavior of experimental substructures. As shown in Figure 6-4, the fitted polynomials follow the actual measurement as long as the extrapolation distance (variation of time parameter) is small. Therefore, it is important to limit the time variation range within iterations to ensure adequate correlation between the measurements and fitted polynomials. In the approximations involving complex-valued time, the imaginary component should also be limited to small values.

Another important issue that may affect the accuracy of the proposed integration procedure is the noise contaminations in the measurements. To reduce the effect of measurement noise, it is recommended that a number of data points larger than the minimum required by the polynomial should be used in the fitting process. Since the data points are equally spaced in time, use of additional data points will not significantly increase computational costs, but will help reduce noise contamination in the fitted polynomials. Using four points to fit second-order polynomials (one point more than minimum required) has been observed to result in adequate fits in the numerical and experimental studies presented in this dissertation.

From Figure 6-4, it is evident that the polynomial curve fitting procedure is most effective for interpolation. In the proposed integration scheme with explicit displacement predictions, the final corrected displacement is expected to be near the last measured data point used for curve fitting. Consequently, the iterative trial

displacements may be interpolated or extrapolated. In order to reduce extrapolations, Equation (8.1) can be modified to over predict displacement by increasing the prediction time to twice the integration time step as in:

$$\tilde{\mathbf{d}}_n = \mathbf{d}_{n-1} + 2\Delta t \mathbf{v}_{n-1} + 2\Delta t^2 \mathbf{a}_{n-1} \tag{8.6}$$

The measurements used for curve fitting will then include data from one step ahead of calculations to increase the interpolated force values. This modification is particularly effective for smaller values of time steps, where an increase of prediction distance will not significantly affect the accuracy of desired displacement. It is important to note that Equation (8.6) is only used to command the actuators, and does not modify the initial iterative displacement vector determined using Equation (8.1). Several simulations have demonstrated that using Equation (8.6) increases the number of integration steps with successful completion of iterative scheme.

8.3.2.1 Stability

In the proposed integration method with implicit or explicit steps, the implicit steps prevent the accumulation of errors and hence, improve the stability compared to fully explicit algorithms. As long as the implicit procedure does not fail in a large number of consecutive integration steps, its stability should remain unconditional similar to the implicit method [59, 121]. However, stability criteria for linear systems cannot be directly extended to hybrid simulation of nonlinear systems with experimental errors such as servo-hydraulic actuator delay, tracking errors and measurement noise. The stability of the proposed integration algorithm is also highly dependent on the curve

fitting procedure used to obtain the force-displacement relation in each step. Thus the unconditional stability of the integration procedures cannot be guaranteed for hybrid simulation because of the additional sources of errors and system nonlinearities. In this case, the overall stability of the proposed algorithm can as well be limited by explicit or operator-splitting methods when a large number of consecutive integration steps fail to converge. However, experimental verification tests described later demonstrate that this situation does not normally occur for a properly tuned experimental setup. In this section, a procedure from the field of structural control is implemented to investigate the stability of the combined integration algorithm.

The combination of explicit and implicit integration steps for solving the equation of motion can be considered as a particular type of "variable structure control system" [122]. These control systems are recognized as a number of feedback control laws and a decision rule. Here, the selection of control law is based on the ability of implicit iterations to solve the equation of motion and maintain proper kinematic relations among the states. Combining several control laws has the advantage of utilizing the useful properties of each of the control systems, namely the stability of the implicit procedure, and the simplicity and guaranteed continuity of the explicit approach. If the control selection logic can be formulated as a function of the states, it is called a "sliding surface", and "sliding mode control" results.

In order to illustrate the improved stability of the integration procedure with implicit or explicit steps, the ability of this method for simulation of free vibration

response of a system is evaluated. The system is assumed to be linear, and random error signals are assumed to exist, as in:

$$\mathbf{M}\mathbf{a}_n + \mathbf{C}\mathbf{v}_n + \mathbf{K}\mathbf{d}_n - \mathbf{u}_n^{\text{err}} = \mathbf{0} \tag{8.7}$$

where $\mathbf{u}_n^{\text{err}}$ is an $N \times 1$ vector of random disturbance signals representing errors in force measurements. For example, this error can be a force measurement noise from an experimental strain-dependent substructure. Here, it is assumed that systematic errors such as mistuning of test equipment are minimal, and delay is properly compensated using procedures described in Chapter 6. For this purpose, the Newmark integration relations (Equations (8.7), (7.3) and (7.4)) can be put in a state-space form as follows:

$$\begin{bmatrix}\mathbf{x}\\ \dot{\mathbf{x}}\end{bmatrix}_n = \mathbf{A}_{\text{D}}\begin{bmatrix}\mathbf{x}\\ \dot{\mathbf{x}}\end{bmatrix}_{n-1} + \mathbf{B}_{\text{D}}\begin{bmatrix}\mathbf{u}_n^{\text{err}}\\ \mathbf{u}_{n-1}^{\text{err}}\end{bmatrix} \tag{8.8}$$

where $\mathbf{x}$ and $\dot{\mathbf{x}}$ are column vectors of displacements and velocities, respectively, and $\mathbf{A}_{\text{D}}$ and $\mathbf{B}_{\text{D}}$ are time-discrete state equation matrices given by:

$$\mathbf{A}_{\text{D}} = \begin{pmatrix}\mathbf{M} + \gamma\Delta t\,\mathbf{C}\\ +\beta(\Delta t)^2\,\mathbf{K}\end{pmatrix}^{-1}\begin{bmatrix}\mathbf{M} + \gamma\Delta t\,\mathbf{C} - \left(\frac{1}{2}-\beta\right)(\Delta t)^2\,\mathbf{K} & \Delta t\,\mathbf{M} - \left(\frac{1}{2}-\gamma\right)(\Delta t)^2\,\mathbf{C}\\ +\left(\beta - \frac{1}{2}\gamma\right)(\Delta t)^3\,\mathbf{C}\mathbf{M}^{-1}\mathbf{K} & +\left(\beta - \frac{1}{2}\gamma\right)(\Delta t)^3\,\mathbf{C}\mathbf{M}^{-1}\mathbf{C}\\ -\Delta t\,\mathbf{K} & (\beta-\gamma)(\Delta t)^2\,\mathbf{K} - (1-\gamma)\Delta t\,\mathbf{C}\\ +\left(\frac{1}{2}\gamma - \beta\right)(\Delta t)^3\,\mathbf{K}\mathbf{M}^{-1}\mathbf{K} & +\mathbf{M} + \left(\frac{1}{2}\gamma - \beta\right)(\Delta t)^3\,\mathbf{K}\mathbf{M}^{-1}\mathbf{C}\end{bmatrix} \tag{8.9}$$

$$\mathbf{B}_{\text{D}} = \begin{pmatrix}\mathbf{M} + \gamma\Delta t\,\mathbf{C}\\ +\beta(\Delta t)^2\,\mathbf{K}\end{pmatrix}^{-1}\begin{bmatrix}\beta(\Delta t)^2\,\mathbf{I} & \left(\frac{1}{2}-\beta\right)(\Delta t)^2\,\mathbf{I} + \left(\frac{1}{2}\gamma - \beta\right)(\Delta t)^3\,\mathbf{C}\mathbf{M}^{-1}\\ \gamma\Delta t\,\mathbf{I} & (1-\gamma)\Delta t\,\mathbf{I} + \left(\beta - \frac{1}{2}\gamma\right)(\Delta t)^3\,\mathbf{K}\mathbf{M}^{-1}\end{bmatrix} \tag{8.10}$$

The above matrices are found by expressing Equations (8.7), (7.3) and (7.4) in a matrix form and solving for state vector at step n. When $\beta = 1/4$ and $\gamma = 1/2$, the state equation takes the implicit form, while for $\beta = 0$ and $\gamma = 1/2$ an explicit formulation results. In the proposed integration method, the variable structure control system takes one of these forms in each step of a simulation.

Fully explicit and fully implicit simulations have been carried out for a 5-story structure, with $(\omega\Delta t)_{\max} \approx 2$ and 5% of critical damping (stiffness-proportional). MDF structural model is selected to demonstrate the stability in the presence of high-frequency modes. Random signals with zero mean and amplitudes of 10% of initial restoring forces are used for $\mathbf{u}_n^{\text{err}}$. By applying an initial displacement (90mm at the top) the phase diagrams of the top story states are shown in Figure 8-2. This figure demonstrates that the simulation using implicit method remains stable up to the vicinity of the origin, where the response continues to oscillate due to the disturbance signals $\mathbf{u}^{\text{err}}$. The explicit simulation becomes unstable, although it should remain stable according to numerical stability limit of $\omega\Delta t = 2$ for undamped structures. It should be mentioned that in the beginning of simulation, the internal forces are significantly greater than the force error signal. The error effects become more evident as the oscillation amplitude reduces and likely result in the instability. The integration errors gradually accumulate until a higher mode is excited, leading to boundless increase of response. Note that the simulation was stopped after a few integration steps with unstable response to prevent excessive graph distortions. The disturbance signal has the same standard deviation from zero in both simulations.

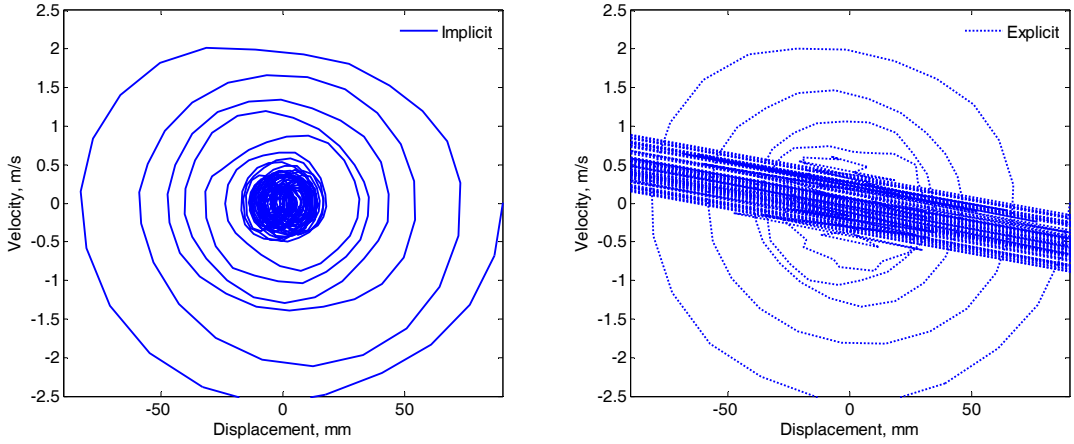

Figure 8-2 Phase diagram of a free vibration response with fully implicit or fully explicit integration methods – $(\omega\Delta t)_{\max} \approx 2$.

In the proposed integration method with implicit or explicit steps, the control structure is selected based on the possibility of using the implicit solution scheme. That is, only if the implicit structure control system fails, the explicit approach is admitted. These situations usually occur when excessive errors prevent accurate calculation of forces, or severe experimental nonlinearities cannot be captured within the maximum number of iterations. Due to the random nature of most experimental errors, the control decision logic cannot be expressed in closed form as a definite sliding surface. Further, because random excitations are normally considered in seismic simulations, the occurrence of the explicit integration steps is expected to be random.

Figure 8-3 shows the phase diagram of a simulation with combined implicit or explicit integration steps. Here, the selection of implicit or explicit control system is made using a random decision logic, providing a 20% probability for explicit steps. This probability is more than what is expected to occur in hybrid simulations with properly tuned experimental setups. In the simulation shown in Figure 8-3, 78.8% of integration

steps are completed using implicit form of the state equation. As illustrated, similar to a fully implicit procedure, the combined method results in a stable simulation. It is shown that the implicit steps (shown by solid lines) are occasionally interrupted by one or more explicit steps (shown by dotted lines). Since the explicit steps are mostly isolated, the accumulation of errors is prevented, and the simulation remains stable.

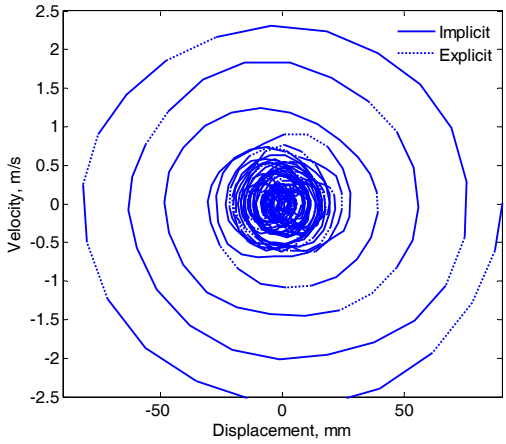

Figure 8-3 Phase diagram of a free vibration response with combined implicit or explicit integration steps – $(\omega\Delta t)_{\max} \approx 2$.

Simulations with higher-frequency modes or longer time steps also show that the combined integration method can produce stable results when explicit procedures fail. The simulation results shown in Figure 8-4 are determined for the same structure as above, but with $(\omega\Delta t)_{\max} = 2.79$. The explicit results are shown to immediately become unstable, while the combined integration method (with 80.2% implicit integration steps) remains stable throughout the simulation.

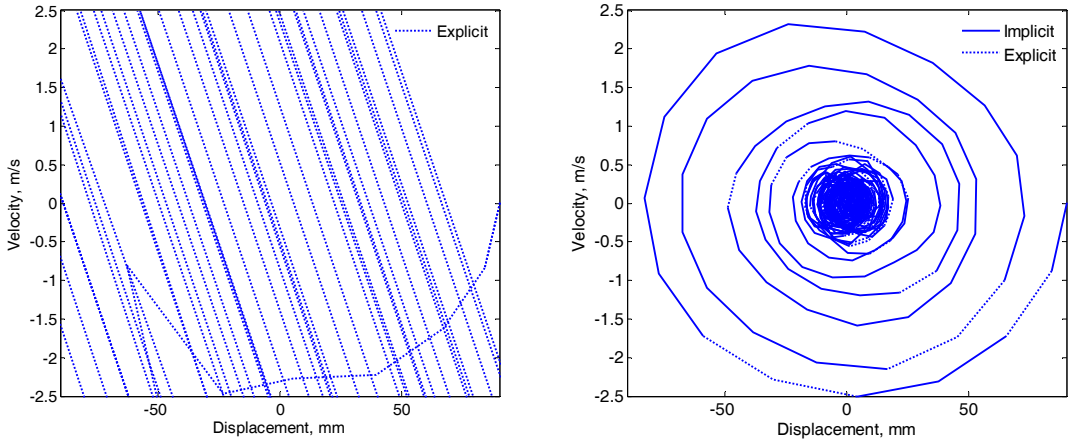

Figure 8-4 Phase diagram of a free vibration response with fully explicit and combined implicit or explicit integration steps – $(\omega\Delta t)_{\max} = 2.79$.

To further study the accuracy and stability properties of the combined integration method, a series of parametric studies has been carried out. In these analyses, the same simulation model and initial conditions as above are considered. The unbalanced energy errors of 5-second simulations with different integration time steps and probabilities for explicit steps are calculated and shown in Figure 8-5. Sharp increases in the energy error shown in this figure indicate the transition of response from stable to unstable. Note that energy errors larger than 100% are considered obsolete, and are not shown in this figure.

As illustrated in Figure 8-5, the energy balance error shows a consistent increase with increase of $\omega\Delta t$ or the probability of explicit integration steps. Note that large errors and the instability of the explicit integration method occurs before the numerical limit of $\omega\Delta t = 2$ due to the existence of an error signal, although the assumed damping of 5% of critical should extend the stability range. When 50% or more of the integration steps are explicit, the combined integration method does not show a significant reduction of energy error compared to a fully explicit integration scheme. On the other

hand, the improvement of accuracy is evident with the reduction of explicit steps below 40% of all integration steps. Particularly, when the percentage of explicit steps is limited to 20% or less, stable results can be obtained over a relatively wide range of $\omega\Delta t$.

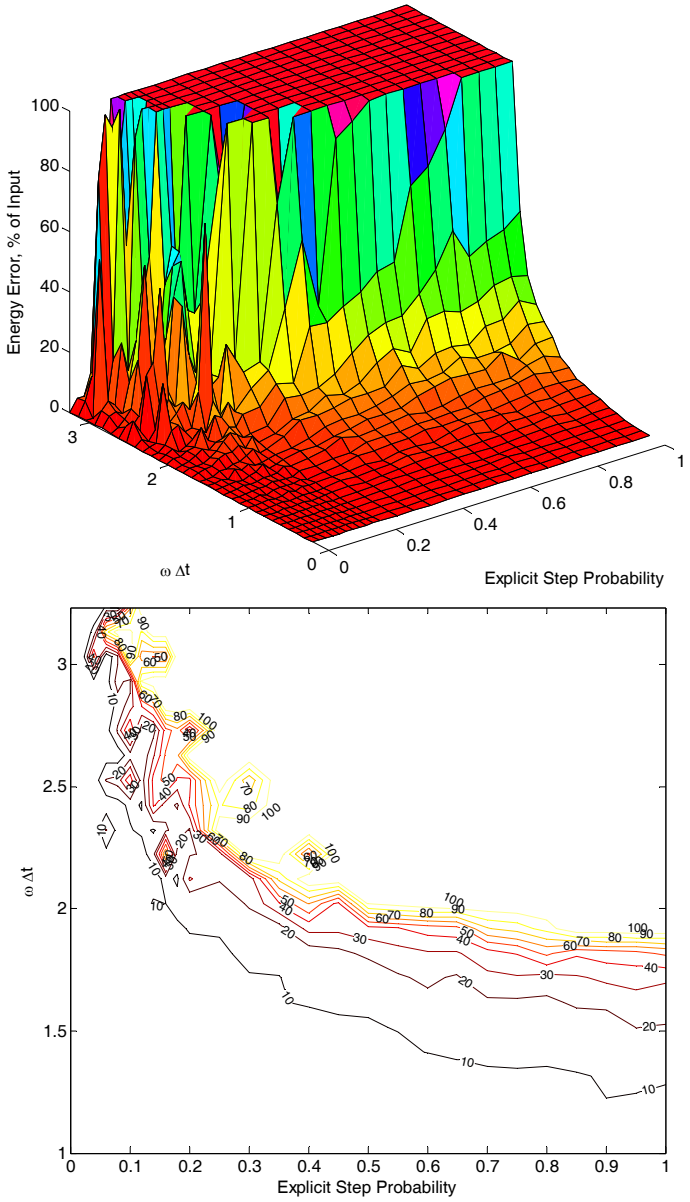

Figure 8-5 Energy error as a function of $\omega\Delta t$ and percentage of explicit steps.

Hence, the proposed integration method is shown to extend the stability limits of the simulation in the presence of high-frequency modes. That is, by selecting an integration time step sufficiently smaller than only significant natural frequencies of the system, accurate and stable results can be obtained using this integration method. The integration time step should also be small enough to accurately capture the hysteretic behavior of nonlinear components using a piecewise linear approximation. Of course, other stability and accuracy conditions should also be considered, such as properly tuned and calibrated experimental setup, reasonable levels of random experimental errors and measurement noise, and accurate delay compensation. The stability of the proposed integration method will be further studied through numerical simulations in Section 8.3.3.3.

8.3.2.2 Delay Compensation

Delay issue should be properly addressed in real-time or continuous hybrid simulations to ensure stability and accuracy. Delay compensation procedures that modify command displacements or force measurement signals can be used with this integration procedure (Chapter 6). However, measured force correction procedures should equally modify the measured displacements, as their phase difference can result in unpredictable simulation performance from this integration approach.

In addition to polynomial extrapolation, the same explicit expression for desired displacements can be used for delay compensation in the command displacement signal,

as described in Section 6.2.1. Therefore, Equation (8.1) will result in the following expression for delay compensated command displacement vector for j^{th} actuator:

$$ {}^{j}\mathbf{d}_{n}^{c} = \mathbf{d}_{n-1} + \left(\Delta t + \tau_j\right)\mathbf{v}_{n-1} + \frac{\left(\Delta t + \tau_j\right)^2}{2}\mathbf{a}_{n-1} \tag{8.11} $$

in which τ_j is the j^{th} actuator's delay. The command displacement for this actuator is then the result of transformation of the above displacement vector to the actuator local coordinate system.

8.3.3 NUMERICAL AND EXPERIMENTAL VERIFICATIONS

Single- and two-degree-of-freedom experimental substructures are considered for experimental verification of the proposed integration procedure with combined implicit or explicit steps. The experiments have been carried out using test specimens shown in Figures 3-3 and 3-4, with the 1978 Tabas earthquake record selected as the input. The simulations are conducted at the real-time event rate to demonstrate the modest computational cost of the proposed integration procedure.

8.3.3.1 Single-Degree-of-Freedom Experiments

A SDF system with natural period of 0.5 seconds has been selected. The properties of this system are the same as those described in Section 6.3.2. The test setup, including actuator response lag and measurement digitization was identified to have a total delay of 16 milliseconds, which is compensated using Equation (8.11).

Simulations of the SDF system using the combined implicit-explicit integrator with a 1024Hz experiment sampling rate and a 10/1024-second integration time step are

shown in Figure 8-6. In these cases, the explicit procedure also produces similar results. In order to demonstrate the advantages of the proposed integration method, the experiment was repeated using a larger integration time step of 100/1024 seconds. The earthquake is scaled down to 2.5% of the full record to maintain a linear response that can be duplicated in both simulations. As shown in Figure 8-7, while the result of explicit approach is on the verge of instability, the combined procedure remains accurate and, more importantly, stable, despite the large time step compared to the natural period of the system.

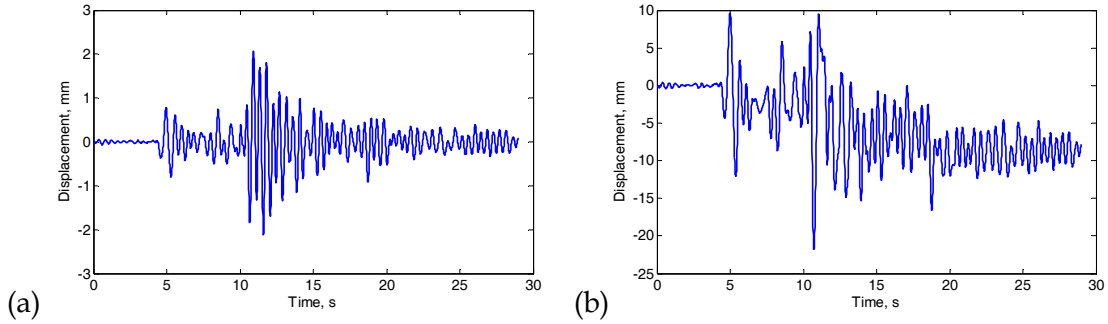

Figure 8-6 Displacement histories of experimental simulation for 0.5-second period SDF system using combined implicit or explicit integration – (a) linear (2.5% amplitude Tabas earthquaje), (b) nonlinear (25% amplitude Tabas earthquake).

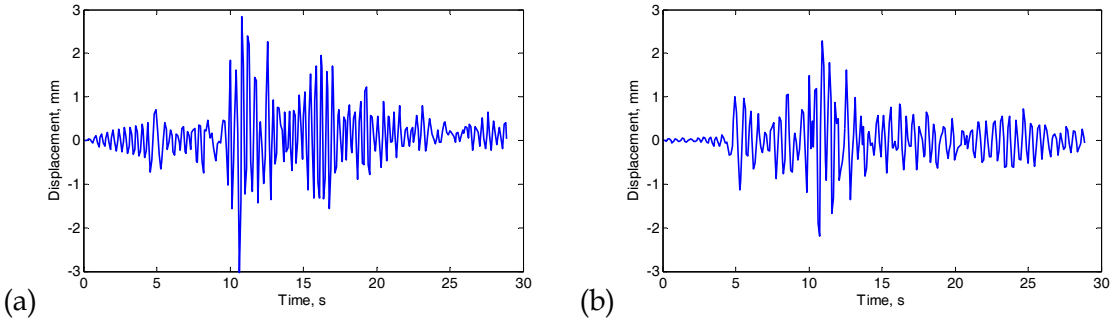

Figure 8-7 Displacement history of linear experimental simulation 0.5-second period system with increased integration time step – (a) explicit Newmark, (b) combined implicit or explicit integration.

The overall energy balance of the system is also observed to be maintained closer to zero with the proposed integration procedure. Based on the experimental results, it can

be concluded that the proposed integrator shows significant improvement when using larger time steps. This can have substantial benefits for testing more complex structural systems including high-frequency modes, or for distributed hybrid simulations. In this test, 84.5% of the integration steps successfully satisfied the implicit formulation.

8.3.3.2 Two-Degree-of-Freedom Experiments

The two-story structure considered in this section and corresponding experimental setup are shown in Figure 8-8. The test setup shown in Figure 3-3, illustrates that the two-degree-of-freedom experimental substructure consists of two SDF setups mounted on top of each other. With two pairs of coupons in the lower clevis, and one pair in the upper one, static tests have been carried out for estimation of the initial stiffness matrix of the experimental substructure:

$$\mathbf{K}^{e} = \begin{bmatrix} 4.86 & -1.41 \\ -1.41 & 0.68 \end{bmatrix} \mathrm{kN/mm} \tag{8.12}$$

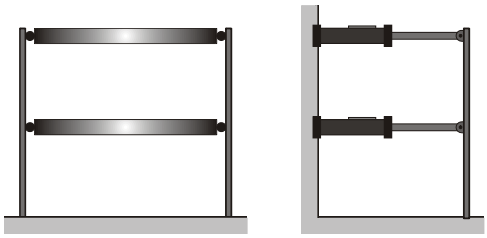

Figure 8-8 Two-degree-of-freedom structure and corresponding laboratory setup for a column.

which is doubled in the numerical simulation to account for both columns of the structure. A mass matrix is then selected to attain natural periods of 0.50 and 0.13 seconds. For comparison purposes, and to keep the specimens in linear range, two simulations with 2.5% of Tabas earthquake were carried out, one with explicit central

difference method and the other with the proposed combined implicit-explicit integration. The integration time step was selected to be 10/1024 seconds in both simulations.

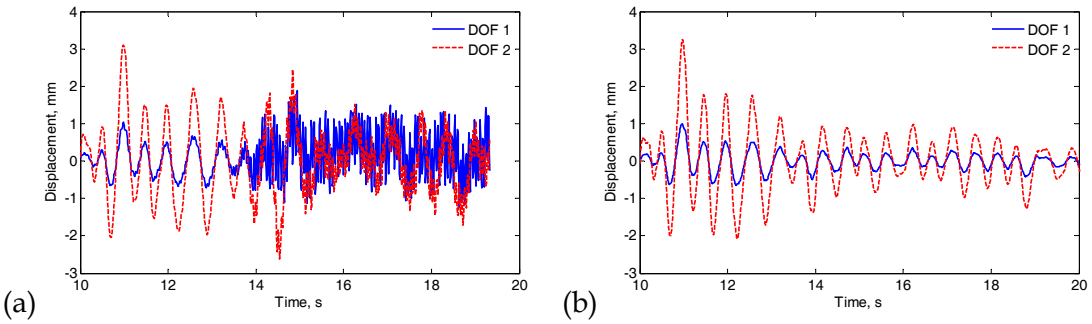

Figure 8-9 Displacement history of linear experimental simulation of two-degree-of-freedom system – (a) explicit central difference, (b) combined implicit or explicit integration.

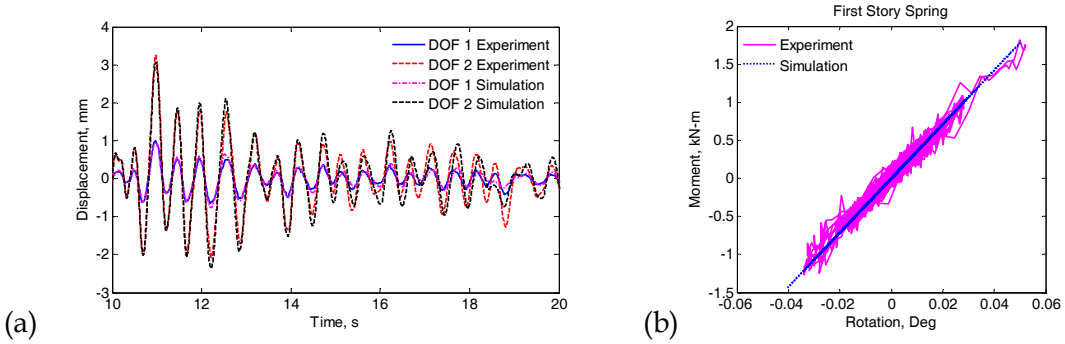

Figure 8-10 Comparison of hybrid simulation results with those of numerical simulation – (a) displacement history, and (b) hysteretic behavior of first story spring.

As shown in Figure 8-9, the explicit integration method is unstable and results in the spurious excitation of the second mode of the system. However, the proposed method is stable throughout the simulation. For this linear simulation, numerical analysis results are also calculated and compared to experimental results in Figure 8-10. The good agreement of the result confirms the accuracy of the proposed integration algorithm; the small discrepancies can be contributed to experimental errors and measurement noise, as shown in the hysteresis in Figure 8-10(b) for the plastic hinge (a nonlinear rotational

spring) in the first story. This rotational behavior is selected over the shearing behavior of columns due to the fact that the column is continuous past the first story, and hence, its behavior is essentially governed by bending and plastic hinge rotation rather than shearing of the column.

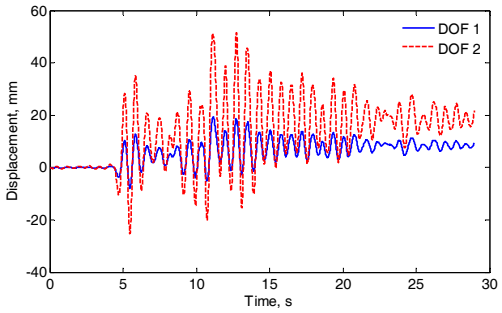

Figure 8-11 Displacement history of nonlinear experimental simulation of two-degree-of-freedom system.

In order to demonstrate the effectiveness of the proposed integration method in nonlinear simulations, the internal forces have been increased by adding numerical mass and increasing the earthquake amplitude scale factor. Figure 8-11 shows the displacement results of a two-degree-of-freedom system with natural periods of 0.60 and 0.15 subjected to the Tabas earthquake scaled in amplitude by 20%. In this simulation, 87.1% of integration steps were successfully completed with implicit corrections; implicit steps are recognized by a flag variable that stores values of 0 or 1 to indicate explicit or implicit steps, respectively. This variable is plotted against time for a portion of simulation in Figure 8-12, along with the number of iterations at each step. It is shown that the steps with failed implicit iterations are not consecutive for the most part, which is important to maintain the stability of the simulation. The number of required iterations at each step is generally small, and only a few steps need more than 5

iterations for convergence. Some of integration steps showed bi-directional convergence, implying that their physical imposition on the experimental substructure could result in erroneous energy dissipation.

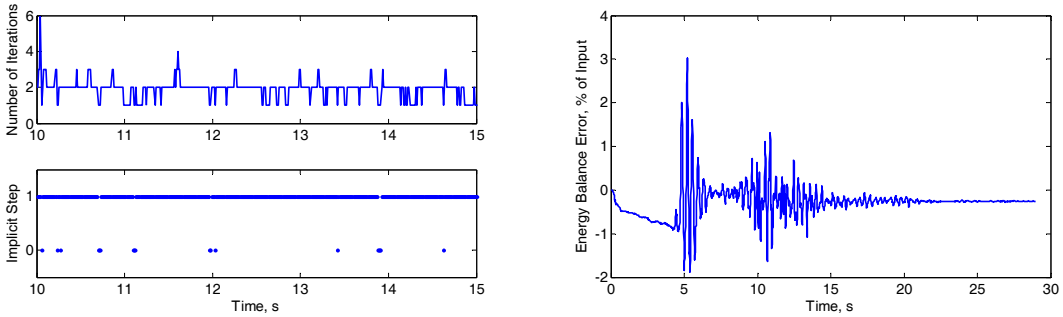

Figure 8-12 Number of iterations in implicit steps and energy balance.

Figure 8-12 also shows that the energy balance error remains close to zero during this simulation. Note that in the beginning of the simulation the excitation amplitude is small, resulting in highly-contaminated force measurements. Consequently, the implicit iterations fail in a large number of integration steps within this period, leading to a gradual drift of energy balance error within the first 5 seconds. As the excitation amplitude increases, this error is normalized by the increased input energy (see Equation (5.30)) and becomes less significant.

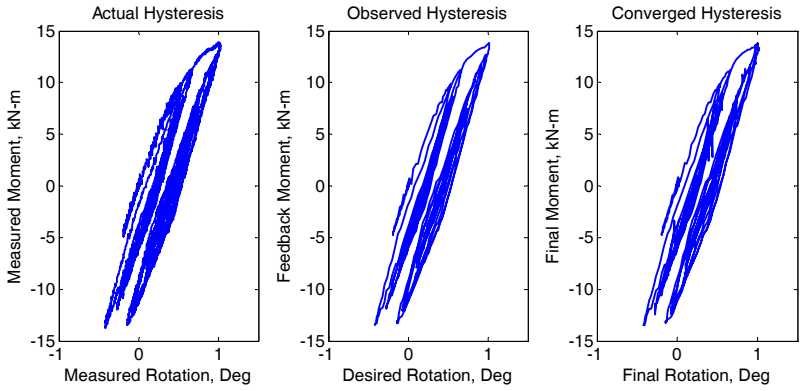

Figure 8-13 Hysteretic behavior of first story plastic hinge.

The comparison between actual, observed, and converged hysteresis of first story plastic hinge is shown in Figure 8-13. The actual hysteretic loop of the presumed plastic hinge in the lower level is simply a plot of measured forces *versus* measured displacements as recorded by the data acquisition system. The observed hysteretic loop is a plot of feedback forces down-sampled to the integration time step, *versus* explicit desired displacements determined using Equation (8.1). The desired displacement may be different from command displacements due to implementation of delay compensation procedures. In this simulation, Equation (8.11) was used for compensation of delay. Finally, the converged hysteretic loops demonstrate the force and displacements at the end of implicit iterations. The similarity of these graphs shows that fitted polynomials have been successful in capturing the actual hysteretic behavior of the test specimen.

Experimental studies have also shown that the performance of the proposed integration scheme for small nonlinearities (as in the previous example) is comparable to results achieved using an operator-splitting method. However, this procedure does not utilize the initial stiffness matrix, and hence, is expected to provide more accurate results for systems that are highly nonlinear. Due to current limitations of the experimental setup, this issue has been further investigated using numerical simulations presented in the next section.

8.3.3.3 Numerical Simulations

One of the most important goals in the development of proposed integration algorithm with implicit or explicit steps is to improve the stability of numerical

simulation. Section 8.3.2 included a discussion on the stability of this method. It should be noted that this integration procedure uses measurements in an iterative solution scheme, and hence, it is largely dependent on the experimental errors. Hence, analytical stability limits that are developed using linear control theory have limited applicability in the evaluation of its stability. For this reason, the energy balance error is considered as an accuracy and stability measure in this section to compare explicit and combined integration methods.

Nonlinear SDF simulations with initial natural periods ranging from 0.08 seconds to 0.4 seconds have been carried out using explicit and combined integration methods. The integration time steps have been selected to range from 0.002 s to 0.02 s. The Tabas earthquake scaled in amplitude to 50% is selected as input excitation. Experimental errors and measurement noise, along with actuator delay have been introduced in the simulations using the numerical models described in Chapter 3. The final energy balance errors in these simulations are plotted against normalized integration time step in Figure 8-14. It is shown that the proposed approach results in smaller energy errors, and remains stable for a wider range of $\omega\Delta t$. Another important observation is that the explicit integration method becomes unstable well before its analytical stability limit, $\omega\Delta t = 2$, as a result of experimental errors and system nonlinearities. This was also observed in the analytical and numerical studies in Section 8.3.2.1. Of course, if larger values of $\omega\Delta t$ are selected in a simulation, the proposed integration method also results in large energy errors, because: (i) the accuracy of simulation degrades with the increase of $\omega\Delta t$ through piecewise linear approximation of nonlinear hysteresis, and (ii) with

large $\omega\Delta t$, polynomials fitted to a few widely-spaced data points fail to accurately capture the behavior of the experimental substructure. Consequently, using large $\omega\Delta t$ the number of steps with failed implicit iterations increases and the procedure eventually reduces to an explicit integration scheme.

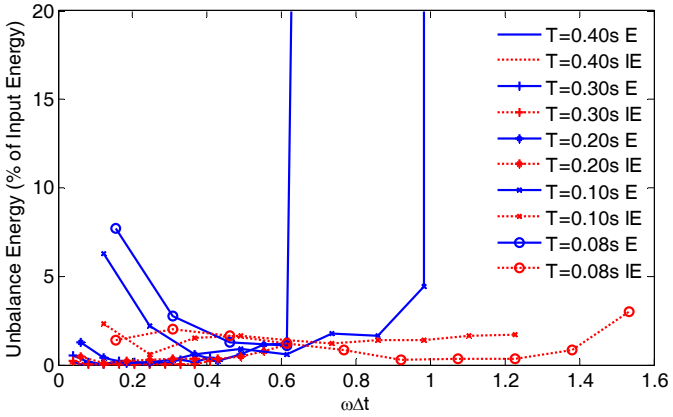

Figure 8-14 Energy balance errors of SDF numerical simulations with explicit (E) and combined (IE) numerical integration.

It was previously mentioned that the proposed integration procedure is expected to provide more accurate results for highly nonlinear systems compared to operator-splitting approach. In this section, the two-degree-of-freedom system of Figure 8-8 is considered in a numerical study to demonstrate this advantage. The properties of the test structure and experimental model are selected to be similar to those of the experimental studies in the previous section (natural periods of 0.60 and 0.15 seconds), except for a reduced yield displacement of the experimental substructure to achieve larger ductility. The input excitation for these numerical simulations is the Tabas earthquake scaled in amplitude by 25%.

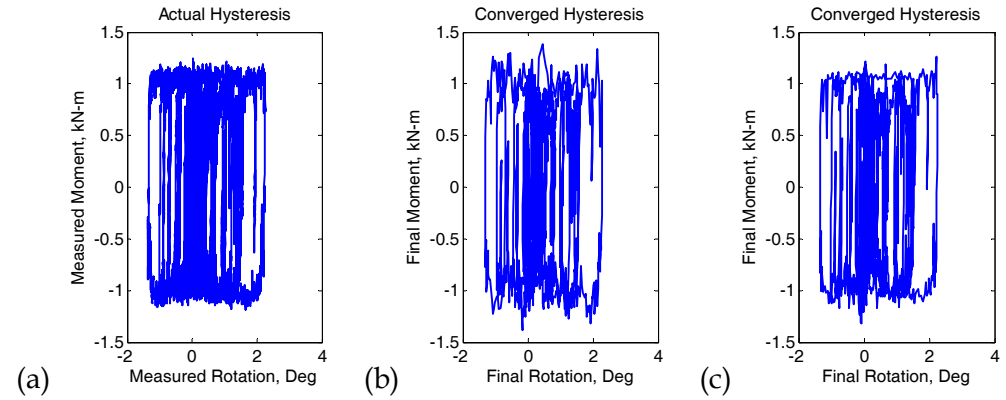

Figure 8-15 Hysteretic behavior of first story plastic hinge in a highly nonlinear numerical simulation – (a) actual hysteresis, (b) operator-splitting result, (c) combined implicit or explicit integration result.

Numerical simulation results of the two-degree-of-freedom system with elasto-plastic behavior and large ductility are shown in Figure 8-15. This figure shows the actual hysteresis of the experimental model, and converged hysteresis for operator-splitting and the proposed methods. It can be observed that the converged hysteresis obtained from operator-splitting method in Figure 8-15(b) has larger force estimates in some steps due to corrections using initial stiffness, as illustrated in Figure 7-2. Figure 8-15(c) shows that the proposed method results in an improved agreement between the actual and converged hysteretic behavior of the experimental substructure. In addition, the proposed implicit procedure does not require the initial stiffness matrix of the structure.

The proposed integration procedure has also been numerically verified to be effective for structural models with strength degradation. Experimental studies involving highly nonlinear experimental substructures are necessary for better comparisons of these two integration methods.

8.4 MODEL-BASED INTEGRATION

In this section, an integration method is introduced that takes advantage of a numerical model for the experimental substructure. Assuming predominantly strain-dependent experimental substructures, this model reduces to a tangent stiffness matrix. The experimental tangent stiffness matrix is updated during the simulation, in order to establish an accurate relationship between experimental forces and displacements. The stiffness update process only uses the readily available force and displacement measurements. The updated experimental stiffness matrix is used to improve the accuracy of a modified operator-splitting integration scheme for testing highly nonlinear experimental substructures. The application and effectiveness of the proposed approach is demonstrated through hybrid simulations with MDF experimental substructures.

8.4.1 OPERATOR-SPLITTING INTEGRATION USING EXPERIMENTAL TANGENT STIFFNESS

A modified operator-splitting approach has been adopted following the formulation of α-method, described in Sections 7.1.2 and 8.2. Here, a tangent stiffness is determined in each integration step that replaces the initial experimental stiffness matrix. That is, after application of predictor displacement from Equation (8.1) and measuring the experimental restoring force, the force vector is updated according to a modified version of Equation (7.23):

$$\mathbf{r}_n^{\mathrm{l}} = \tilde{\mathbf{r}}_n^{\mathrm{l}} + \mathbf{K}_n^{\mathrm{e,l}} \left(\mathbf{d}_n^{\mathrm{l}} - \tilde{\mathbf{d}}_n^{\mathrm{l,m}} \right) \tag{8.13}$$

where $\mathbf{K}_n^{\mathrm{e,l}}$ is the $m \times m$ experimental tangent stiffness matrix at step n expressed in the actuator coordinate system, where m is the number of actuators (and load cells). According to this equation, the acceleration in the corrector step is obtained from:

$$\mathbf{a}_n = \mathbf{B}_n^{-1} \left\{ \begin{array}{l} \mathbf{M}^{\mathrm{t}} \boldsymbol{\iota} \ddot{u}_g \left(t_n + \alpha \Delta t\right) - \mathbf{C}\left[\mathbf{v}_{n-1} + (1+\alpha)(1-\gamma)\Delta t\, \mathbf{a}_{n-1}\right] \\ \quad + \alpha\left(\mathbf{K}\mathbf{d}_{n-1} + \mathbf{r}_{n-1} - \mathbf{M}^{\mathrm{e}}\mathbf{a}_{n-1}\right) \\ \quad - (1+\alpha)\left[\tilde{\mathbf{r}}_n + \left(\mathbf{K} + \mathbf{K}_n^{\mathrm{e}}\right)\left(\tilde{\mathbf{d}}_n - \beta \Delta t^2 \mathbf{a}_{n-1}\right) - \mathbf{T}^{\mathrm{T}} \mathbf{K}_n^{\mathrm{e,l}} \tilde{\mathbf{d}}_n^{\mathrm{l,m}}\right] \end{array} \right\} \tag{8.14}$$

where:

$$\mathbf{B}_n = \mathbf{M} - \alpha \mathbf{M}^{\mathrm{e}} + \gamma \Delta t (1+\alpha)\mathbf{C} + \beta \Delta t^2 (1+\alpha)\left(\mathbf{K} + \mathbf{K}_n^{\mathrm{e}}\right) \tag{8.15}$$

Equations (7.4), (8.2) and (8.13) can then be used to update $\mathbf{v}_n$, $\mathbf{d}_n$ and $\mathbf{r}_n$.

8.4.2 ESTIMATION OF STIFFNESS MATRIX

An updated tangent stiffness matrix of the experimental substructure can be used to solve the equations of motion without iterations, or correct the force measurements in iterative schemes without physical application of iterative displacements. The tangent stiffness matrix can be used in a variety of integration methods to replace the initial stiffness matrix [14, 66, 70, 73, 74], including the modified operator-splitting method. By updating the experimental tangent stiffness during the simulation, the accuracy of the integration procedure is expected to improve in highly nonlinear experiments.

Estimation of tangent stiffness of the experimental substructures has already been attempted in hybrid simulations. These tangent stiffness matrices have been used for error calculations [16], delay compensation [55] and establishing a force-displacement relation for SDF experimental substructures to improve the numerical integration [13].

The first two applications were extended to MDF experimental substructures, which are briefly discussed in this section. The use of tangent stiffness matrix has not been attempted for numerical integration in hybrid simulations with MDF experimental substructures.

The estimated tangent stiffness matrix of the experimental substructure should satisfy the following incremental force-displacement relation at the n^{th} integration step:

$$\Delta \mathbf{r}_n^{\text{l}} = \mathbf{K}_n^{\text{e,l}} \Delta \mathbf{u}_n^{\text{l}} \tag{8.16}$$

where $\Delta \mathbf{r}_n^{\text{l}}$ and $\Delta \mathbf{u}_n^{\text{l}}$ are measured incremental force and displacement vectors of the experimental substructure in actuator local coordinate system, respectively.

The conventional static test sequence for estimation of experimental stiffness matrix cannot be applied to online hybrid simulations. The required procedures should estimate the tangent stiffness using $m \times 1$ measured force and displacement vectors of experimental substructure. Thewalt and Roman [16] developed such a procedure based on the BFGS formula [123], and used it to estimate energy errors of hybrid simulations. In their approach, the stiffness matrix is updated using $\mathbf{K}_n^{\text{e,l}} = \mathbf{K}_*^{\text{e,l}} + \Delta \mathbf{K}_*^{\text{e,l}}$, where:

$$\Delta \mathbf{K}_*^{\text{e,l}} = \left[1 + \frac{\left(\Delta \mathbf{u}_n^{\text{l}}\right)^{\text{T}} \mathbf{K}_*^{\text{e,l}} \Delta \mathbf{u}_n^{\text{l}}}{\left(\Delta \mathbf{r}_n^{\text{l}}\right)^{T} \Delta \mathbf{r}_n^{\text{l}}} \right] \frac{\Delta \mathbf{r}_n^{\text{l}} \left(\Delta \mathbf{r}_n^{\text{l}}\right)^{\text{T}}}{\left(\Delta \mathbf{r}_n^{\text{l}}\right)^{\text{T}} \Delta \mathbf{u}_n^{\text{l}}} - \frac{\Delta \mathbf{r}_n^{\text{l}} \left(\Delta \mathbf{u}_n^{\text{l}}\right)^{\text{T}} \mathbf{K}_*^{\text{e,l}}}{\left(\Delta \mathbf{r}_n^{\text{l}}\right)^{\text{T}} \Delta \mathbf{u}_n^{\text{l}}} - \frac{\mathbf{K}_*^{\text{e,l}} \Delta \mathbf{u}_n^{\text{l}} \left(\Delta \mathbf{r}_n^{\text{l}}\right)^{\text{T}}}{\left(\Delta \mathbf{r}_n^{\text{l}}\right)^{\text{T}} \Delta \mathbf{u}_n^{\text{l}}} \tag{8.17}$$

and $\mathbf{K}_*^{\text{e,l}}$ is equal to $\mathbf{K}_{n-1}^{\text{e,l}}$ or $\mathbf{K}_0^{\text{e,l}}$, whichever results in a $\Delta \mathbf{K}_*^{\text{e,l}}$ with a smaller Frobenius norm. The above relation results in a symmetric stiffness matrix that satisfies Equation (8.16). Thewalt and Roman also introduced criteria for selection of reliable measurements and data that result in positive-definite stiffness matrices.

Carrion and Spencer [55] employed the Broyden formula [34] to update the experimental stiffness matrix:

$$\mathbf{K}_n^{\mathrm{e,l}} = \mathbf{K}_{n-1}^{\mathrm{e,l}} + \frac{\left(\Delta\mathbf{r}_n^{\mathrm{l}} - \mathbf{K}_{n-1}^{\mathrm{e,l}}\Delta\mathbf{u}_n^{\mathrm{l}}\right)\left(\Delta\mathbf{u}_n^{\mathrm{l}}\right)^{\mathrm{T}}}{\left(\Delta\mathbf{u}_n^{\mathrm{l}}\right)^{\mathrm{T}}\Delta\mathbf{u}_n^{\mathrm{l}}} \tag{8.18}$$

and used it for model-based compensation of delay. The resulting stiffness matrix will also satisfy Equation (8.16), but may not be symmetric or positive-definite. A brief discussion on the characteristics of Equations (8.17) and (8.18) is presented in Section 8.4.5.

A new procedure for online estimation of tangent stiffness matrix is proposed here for MDF experimental substructures. Similar to the above-mentioned procedures, it is assumed that the only available information during the simulation is the incremental measured forces and displacements. For this reason, it is first attempted to reduce the number of unknowns required to update the tangent stiffness matrix. For this purpose, and to facilitate the estimation of stiffness matrix elements, a coordinate system is sought, in which the stiffness matrix is diagonal. By transforming force and displacement increments to this coordinate system, the decoupled stiffness matrix elements can be obtained by dividing corresponding force and displacement pairs. Two approaches are presented here; the first method uses the physical test setup information to identify an intrinsic coordinate system with the above-mentioned properties, and the second method is the classical decomposition of stiffness matrix using its eigenvectors.

8.4.2.1 Decomposition of Stiffness Matrix Using Test Structure Information

The first method of decomposition of stiffness matrix takes advantage of some information about the physical test system and experimental element configuration. A brief look at structural analysis problems reveals that the stiffness matrices of most structural elements consist of terms that are a combination of a few geometric and material properties, such as modules of elasticity, lengths, areas and section modules. These quantities are subject to change in nonlinear numerical analyses. From a macroscopic standpoint, similar intrinsic parameters often exist that determine the resistance of a structure to loads imposed by actuators. For example, the lateral stiffness of a bracing system subjected to horizontal displacements provides a sufficient force-displacement relation, regardless of the configuration of individual elements. As another example, the entire $N \times N$ stiffness matrix of an N-story shear building can be found from N story stiffnesses. Further, if only m ($m < N$) stories are subjected to loading, the required number of essential stiffness terms reduces to m, some of which may represent the equivalent stiffness of several stories.

By only considering the key intrinsic parameters, the stiffness matrix $\mathbf{K}_n^{\mathrm{e,l}}$ of the experimental substructure in the actuator coordinate system can often be expressed as:

$$\mathbf{K}_n^{\mathrm{e,l}} = \mathbf{T}_{\mathrm{p}}^{\mathrm{T}} \mathbf{P}_n \mathbf{T}_{\mathrm{p}} \tag{8.19}$$

where $\mathbf{P}_n$ is a diagonal $p \times p$ matrix of essential stiffness parameters. The transformation matrix $\mathbf{T}_{\mathrm{p}}$ transform displacements from the actuator local (substructure) coordinate system to an intrinsic (parameter) coordinate system with a presumed diagonal stiffness

matrix $\mathbf{P}_n$. Normally, only the parameters that are expected to change in a simulation should be included in $\mathbf{P}_n$, and $\mathbf{T}_\mathrm{p}$ can include all other parameters, such as numbers, angles, and lengths. For the example of shear building, $\mathbf{P}_n$ is a diagonal matrix of story stiffness, and $\mathbf{T}_\mathrm{p}$ simply transforms the displacements to story drifts. In a flexural element, the slope of one end with respect to the other end can be used as a component of intrinsic coordinate system. In such a case, the substructure local displacements should be projected to the rotation of one end of each element with respect to the tangent to the other end, to form each row of the transformation matrix, $\mathbf{T}_\mathrm{p}$.

In order to calculate the terms of the diagonal stiffness matrix $\mathbf{P}_\mathrm{n}$, the incremental displacement and force vectors should be transformed to the above-mentioned intrinsic coordinate system. Regarding displacements, the transformation is relatively simple, and can be done through the same transformation matrix described above:

$$\Delta\mathbf{u}_n^\mathrm{p} = \mathbf{T}_\mathrm{p}\Delta\mathbf{u}_n^\mathrm{l} \tag{8.20}$$

in which $\Delta\mathbf{u}_n^\mathrm{p}$ is the displacement vector in the intrinsic coordinate system. The transformation of displacements from global to actuator coordinate system can be carried out using $\Delta\mathbf{u}_n^\mathrm{l} = \mathbf{T}\Delta\mathbf{u}_n$.

The transformation of forces to the intrinsic coordinate system depends on the static determinacy of the experimental substructure. For statically determinate structures, the intrinsic forces can simply be found by equilibrium. Therefore, each column of force transformation matrix $\mathbf{T}_\mathrm{p}^{(-\mathrm{T})}$ from local to intrinsic coordinates can be found by applying a unit force in a local degree of freedom and calculating forces in intrinsic degrees of

freedom. The transformation of local incremental force vector $\Delta\mathbf{r}_n^l$ to intrinsic coordinates ($\Delta\mathbf{r}_n^{\mathrm{p}}$) will then be:

$$\Delta\mathbf{r}_n^{\mathrm{p}} = \mathbf{T}_{\mathrm{p}}^{(-\mathrm{T})}\Delta\mathbf{r}_n^{\mathrm{l}} \tag{8.21}$$

The superscript $(-\mathrm{T})$ represents a pseudo-inverse of the matrix transpose, due to the fact that transformation matrices $\mathbf{T}$ and $\mathbf{T}_{\mathrm{p}}$ are generally rectangular [124]. Further, the above matrices satisfy $\mathbf{T}_{\mathrm{p}}^{\mathrm{T}}\mathbf{T}_{\mathrm{p}}^{(-\mathrm{T})} = \mathbf{I}_p$ and $\mathbf{T}^{\mathrm{T}}\mathbf{T}^{(-\mathrm{T})} = \mathbf{I}_m$, where $\mathbf{I}_p$ and $\mathbf{I}_m$ are $p \times p$ and $m \times m$ identity matrices, respectively.

If the experimental substructure is statically indeterminate (for example, if the number of stiffness parameters to be updated in each step is greater than m), the calculation of forces in intrinsic coordinates requires the stiffness matrix of the system for a structural analysis. In this case, the structure should be analyzed to find local displacements from the measured local force vector, $\Delta\mathbf{r}_n^{\mathrm{l}}$. The resulting local displacements can then be transformed to the intrinsic coordinate system using Equation (8.20). The intrinsic forces will be the forces corresponding to the intrinsic displacement vector using diagonal stiffness matrix $\mathbf{P}_n$:

$$\Delta\mathbf{r}_n^{\mathrm{p}} = \mathbf{P}_n\mathbf{T}_{\mathrm{p}}\left(\mathbf{K}_n^{\mathrm{e,l}}\right)^{-1}\Delta\mathbf{r}_n^{\mathrm{l}} \tag{8.22}$$

It should be noted that the stiffness matrices used in Equation (8.22) are constantly updated. As a result, the transformation may be different from one step to another. Further, since the stiffness matrix is updated after being used for this transformation in the same integration step, an iterative procedure is necessary to ensure the use of up-to-

date transformation relations. This may require computations too demanding for fast or real-time hybrid simulations, but given the small number of degrees of freedom in most experimental substructures, the iterative stiffness estimation procedure may be acceptable. Nonetheless, the iterative procedure may be omitted by replacing Equation (8.22) by:

$$\Delta \mathbf{r}_n^{\mathrm{p}} = \mathbf{P}_{n-1} \mathbf{T}_{\mathrm{p}} \left(\mathbf{K}_{n-1}^{\mathrm{e,l}} \right)^{-1} \Delta \mathbf{r}_n^{\mathrm{l}} \tag{8.23}$$

updated once at the beginning of each integration step. Using the previous stiffness matrix in transformation of force vector at the current step slightly reduces the stiffness matrix update rate. With the customary integration time steps that are required to achieve reasonable simulation accuracy, the effects of this reduction of stiffness update rate are normally insignificant.

After determination of forces and displacements in the intrinsic coordinate system, each diagonal element of the updated parameter matrix can be found by dividing the corresponding elements of force vector by the displacement vector. Put in matrix form, the expression will be:

$$\mathbf{P}_n = \mathrm{diag}\left(\Delta \mathbf{u}_n^{\mathrm{p}} \right)^{-1} \mathrm{diag}\left(\Delta \mathbf{r}_n^{\mathrm{p}} \right) \tag{8.24}$$

Following Equation (8.19), the global stiffness matrix of the experimental substructures can then be found using:

$$\mathbf{K}_n = \mathbf{T}^{\mathrm{T}} \mathbf{K}_n^{\mathrm{e,l}} \mathbf{T} \tag{8.25}$$

8.4.2.2 Modal Decomposition of Stiffness Matrix

As demonstrated in the previous section, transformation of the stiffness matrix to a coordinate system, in which the stiffness matrix is diagonal, facilitates the estimation of stiffness terms using the measurements. In this section, the classical method of diagonalization of stiffness matrix is presented for use in the above-mentioned stiffness estimation procedure.

An $m \times m$ matrix is diagonalizable if it has m linearly-independent eigenvectors. This is the case when the matrix has m distinct eigenvalues [125]. Given these conditions, the following relation can be used to diagonalize the tangent stiffness matrix at step n:

$$\mathbf{P}_n = \mathbf{\Phi}_n^{-1} \mathbf{K}_n^{\mathrm{e,l}} \mathbf{\Phi}_n \tag{8.26}$$

in which:

$$\mathbf{\Phi}_n = \left[\boldsymbol{\phi}_1 \, \boldsymbol{\phi}_2 \cdots \boldsymbol{\phi}_m \right]_n \tag{8.27}$$

is a matrix of normalized eigenvectors (modal matrix) of the local stiffness matrix at step n, $\mathbf{K}_n^{\mathrm{e,l}}$. In addition to the above, the symmetry of the stiffness matrix results in the orthogonality of eigenvectors (or the dynamic mode shapes with an identity mass matrix) [119], which further facilitates the diagonalization process by changing Equation (8.26) to:

$$\mathbf{P}_n = \mathbf{\Phi}_n^{\mathrm{T}} \mathbf{K}_n^{\mathrm{e,l}} \mathbf{\Phi}_n \tag{8.28}$$

Hence, a general choice of the transformation matrix $\mathbf{T}_{\mathrm{p}}$ is the transpose of the eigenvectors matrix:

$$\mathbf{T}_{\mathrm{p}} = \mathbf{\Phi}_n^{\mathrm{T}} \tag{8.29}$$

Since this transformation matrix is square, the number of stiffness parameters to be estimated will be equal to m, which is the same as the number of measured force and displacement pairs. Note that the orthogonality of the normalized eigenvectors matrix results in:

$$\mathbf{\Phi}_n^{-1} = \mathbf{\Phi}_n^{\mathrm{T}} \tag{8.30}$$

The transformation matrix given by Equation (8.29) and its inverse from Equation (8.30) can be used in the procedure presented in the previous section to transform forces and displacements to modal coordinate system and update the diagonal experimental tangent stiffness matrix. The experimental tangent stiffness matrix in actuator local coordinates can then be found using the inverse transformation.

It should be noted that since the stiffness matrix is updated in each integration step, the mode shapes may change, and the transformation matrix needs to be updated. That is, an eigenvalue problem involving the tangent stiffness matrix should be solved in each integration step, regardless of the static determinacy of the experimental substructures. Hence, this procedure is more computationally expensive than that of previous section for statically determinate experimental substructures. Furthermore, the change in the stiffness matrix within an integration step leads to an iterative procedure for simultaneous update of stiffness and transformation matrices. Again, it has been

observed that iterations can be avoided by using the tangent stiffness matrix in the previous step, $\mathbf{K}_{n-1}^{e,l}$, with negligible loss of accuracy.

The modal decomposition method provides a general procedure for diagonalization of stiffness matrix. This approach be used when an intrinsic coordinate system, in which the stiffness matrix is diagonal, cannot be directly recognized based on the geometry and element configuration of the experimental substructure. The coupled axial and bending behavior of columns, or the combined horizontal, vertical and rotational testing of bracings, such as the zipper frame shown in Figure 8-16, are examples of this situation. Furthermore, this procedure does not need the above-mentioned *a priori* information about the experimental substructure.

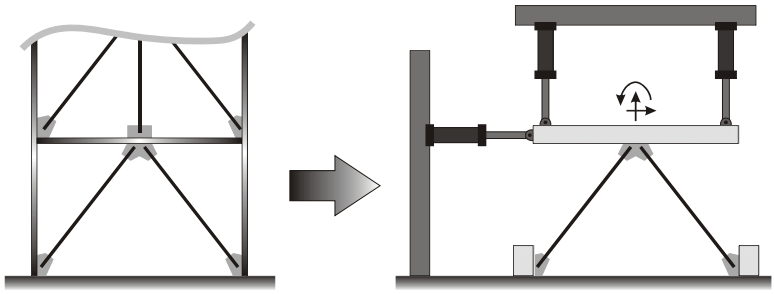

Figure 8-16 A zipper frame and the first story bracing as the experimental substructure [81].

8.4.2.3 Selection of Reliable Measurements

The fidelity of displacement and force measurements used in Equation (8.24) is essential to the accuracy of the estimated tangent stiffness matrix. Hence, it is important to minimize the amount of noise in the measured force and displacement vectors. For this purpose, filters and signal smoothing procedures can be used to improve the measurements. In addition, it is suggested that the stiffness matrix update process should be carried out only in integration steps with significant displacement increments

sufficiently larger than the noise level. For example, the incremental displacement vector should satisfy:

$$\left\|\Delta \mathbf{u}_n^1\right\|_\infty > \delta \tag{8.31}$$

where $\left\|\Delta \mathbf{u}_n^1\right\|_\infty$ denotes the largest element of the absolute incremental displacement vector, and δ is displacement increment threshold. The displacement threshold δ should be greater than the measurement noise level, but small enough to capture steps with significant displacement increments. Pretest simulations with zero input excitation can be used to determine the root-mean-square (RMS) of the noise signal in displacement and force measurements. Recommended value of δ is the greater of: 10 times the RMS of displacement noise, or a value that results in a force (using initial stiffness) 10 times greater than the RMS of force noise.

The proposed integration procedure using experimental tangent stiffness in the modified operator-splitting algorithm is summarized in Table 8-3.

8.4.3 CHARACTERISTICS

Compared to the operator-splitting integration with linear experimental stiffness matrix, the proposed integration procedure is more computationally expensive. The additional processing cost primarily originates from the stiffness update procedure described in the preceding section. However, for experimental substructures with a few degrees of freedom, this additional task can be easily handled by currently-available processing tools. In terms of communication requirements, interface forces and displacements are communicated once in each step of this integration method. Hence, it

is a reasonable procedure for implementation in geographically-distributed hybrid simulations.

Table 8-3 Operator-splitting integration using experimental tangent stiffness matrix.

Simulation Steps	Equations and Other Information
• **Calculate predictor displacement vector at step** n	**Equation (8.1)**
• **Transform displacements to actuator coordinate system**	$\tilde{\mathbf{d}}_n^{\mathrm{l}} = \mathbf{T}\tilde{\mathbf{d}}_n$
• **Apply predictor displacement vector on experimental substructure**	
• **Measure experimental displacements and restoring forces**	
• **Compare displacement increments with threshold:**	**Equation (8.31)**
- **Displacement increment norm is greater than threshold: update stiffness matrix:**	
▪ **Transform force and displacement increments to intrinsic coordinates**	**Equations (8.20) to (8.25)**
▪ **Update intrinsic stiffness parameters**	
▪ **Transform the updated parameters to actuator local coordinates**	
- **Displacement increment norm is smaller than threshold: keep the experimental tangent stiffness unchanged**	
• **Calculate the acceleration vector of step** n	**Equations (8.14) and (8.15)**
• **Calculate new displacement, velocity, and force vectors**	**Equations (7.4), (8.2) and (8.13)**
• **Set** $n+1 \rightarrow n$ **and go to the next integration step**	

The proposed integration method uses a more accurate force-displacement relationship to reduce the discrepancy between the actual hysteretic behavior of the experimental substructure and that obtained by the numerical integrator at the end of corrector step (termed herein the converged hysteresis). As shown in Figure 7-2, the use of a tangent stiffness that is obtained using the measurements will improve the force estimate corresponding to the corrector displacement. Consequently, the difference between actual energy dissipated in the experimental substructure E_{E} and that observed by the numerical integrator $E_{\mathrm{E}}^{\mathrm{C}}$ reduces, and a smaller energy balance error results from

Equation (5.22). This improvement will be demonstrated in experimental and numerical verifications in Section 8.4.5.

It should be noted that the proposed integration algorithm using experimental tangent stiffness is not iterative. This simplifies the implementation of this integration algorithm in hybrid simulations. However, the effects of changes in displacement vector in the corrector step are not considered in the estimation of the tangent stiffness matrix. As a result, the equation of motion is exactly satisfied only when the tangent stiffness matrix does not change between predictor and corrector displacements. For this reason, a reduced discrepancy between the actual and converged behaviors of the experimental substructure is still expected to exist using this integration method.

8.4.3.1 Stability

Similar to the conventional operator-splitting integration method, the proposed approach results in accurate kinematic relationships among displacements, velocities, and accelerations, and prevents the accumulation of errors. It has been shown that when initial stiffness matrix is used in the integration procedure, the stability is guaranteed as long as the specimen nonlinearity is of softening type [14]. This condition can be released when the stiffness matrix is properly updated in each integration step, since the actual stiffness is not expected to be larger than the tangent stiffness estimated in the integration procedure. Therefore, the stability of this integration method is equivalent to, or better than the conventional operator-splitting method.

It should be noted that besides improving the accuracy of operator-splitting method, using measurements for updating the stiffness matrix creates a potential for instability due to experimental errors. For example, the conventional operator-splitting method may remain stable in the presence of very large tracking errors as long as they are corrected using Equation (7.23). As an extreme case, if the actuators completely fail to respond to the commands ($\mathbf{d}_n^{\mathrm{l,m}} = \mathbf{0}$), the conventional operator-splitting method will result in a simulation with linear response from experimental substructure (stable, but most probably, inaccurate). Such independence from experimental errors, however, cannot be expected from the proposed stiffness update procedure.

Low-quality measurements, or very small displacement threshold for selection of significant displacement increments, may result in large noises in the estimated stiffness matrix elements, leading to an inaccurate or unstable simulation. Nonetheless, as long as the experimental setup is well tuned for the intended application range, and only reliable measurements are used in the stiffness update process by proper selection of displacement threshold, it is expected that the proposed integration algorithm remains stable. Hence, the stability of this integration method depends on the extent of experimental errors; the stability and accuracy of simulation can be assured using this integration method if (i) the experimental setup is properly tuned and calibrated, (ii) delay is accurately compensated, (iii) the experimental errors and measurement noise remain within reasonable limits, and (iv) the integration time step is small enough to capture the behavior of nonlinear components with reasonable accuracy. Note that these

conditions are required for a reliable simulation using any integration method, and the method proposed in this section does not impose other stability conditions.

8.4.3.2 Delay Compensation

The proposed integration procedure with experimental tangent stiffness uses an explicit expression for calculation of predictor displacement vector, similar to other integration methods studied in this dissertation. Hence, in addition to the polynomial extrapolation method, the same Equation (8.11) can be used to compensate the delay. Again, if force correction procedures are utilized for delay compensation, they should retain the phase agreement of forces and displacements for use in the estimation of tangent stiffness matrix.

8.4.4 USE OF TANGENT STIFFNESS MATRIX IN ITERATIVE SCHEMES

In the integration scheme described above, the experimental tangent stiffness is updated based on the measurements up to the predictor displacement of integration step n. The use of this stiffness matrix in the corrector step without change assumes a constant tangent stiffness matrix between predictor and corrector displacements. It was mentioned in Section 8.4.3 that because of this assumption, the experimental energy dissipation may be slightly different from that converged at the end of corrector step. Although the difference between these two displacement vectors are generally small, for exact satisfaction of equation of motion and finite difference kinematics relations, an iterative implicit integration scheme should be employed.

In iterative implicit integration methods, the experimental measurements should be updated in each iteration, and a new experimental tangent stiffness matrix should be determined and used to update the states. The first problem in using this procedure appears to be the iterative scheme, which is not suitable for simulations involving nonlinear physical subsystems. A possible workaround for this problem is using the fitted polynomials described in Section 8.2 for estimation of forces corresponding to iterative displacements, to avoid physical iterations.

Another problem in implementation of an iterative algorithm is to create an uninterrupted integration scheme. Other procedures need to be established for handling the integration steps with failed iterations. For this purpose, all cases that may prevent iterative calculation tasks from proper functioning should be identified and properly addressed. The following lists these situations, and the corresponding measures that can be taken to continue the simulation:

- The maximum number of iterations can be reached before the convergence criterion is met.
    - In this situation, the integration step can be completed using a one step correction using tangent stiffness. The stiffness matrix may or may not be updated in this integration step, depending upon the displacement increments size.
- The tangent stiffness matrix update may fail for one or more of the following reasons:

- The displacement increments are too small,
- The force estimation procedure fails to estimate forces corresponding to one or more iterative displacements due to excessive variation of time parameter,
- The matrix $\mathbf{B}_n$ given by Equation (8.15) cannot be inverted.
    - When the experimental stiffness matrix is not updated, the next iteration will yield identical state vectors, resulting in a "false convergence." In this situation, the solution to the last iteration can be left unchanged, since up to that point, the iterative corrections can be thought as successful. However, the convergence error is expected to be more than the accepted tolerance of Equation (8.4). Alternatively, the same strategy as that for unconverged integration steps can be followed here.

Note that a number of above problems can also occur in the non-iterative scheme described above, which can be handled by leaving the stiffness matrix unchanged in that integration step. As shown in the summarized algorithm of Table 8-4, an iterative integration algorithm based on the above-mentioned approach will be more computationally expensive. Further, by using polynomials for estimation of forces corresponding to the iterative displacement, a result similar to that of integration method with combined implicit or explicit steps may result. The properties of this

integration algorithm will be further studied through numerical simulations in Section 8.4.5.2.

Table 8-4 Iterative implicit integration procedure using experimental tangent stiffness.

Simulation Steps	Equations and Other Information
• **Calculate predictor displacement vector at step** n	**Equation (8.1)**
• **Transform displacements to actuator coordinate system**	$\tilde{\mathbf{d}}_n^l = \mathbf{T}\tilde{\mathbf{d}}_n$
• **Apply predictor displacement vector on experimental substructure**	
• **Measure experimental displacements and restoring forces**	
• **Fit polynomials to most recent force and displacement measurements**	
• **Calculate the predictor velocity vector**	**Equation (7.4) with** $\gamma = 0$
• **Solve iteratively:**	
- **Using fitted polynomials, estimate forces corresponding to iterative displacements, and calculate displacement and force increments from the beginning of step**	
- **Compare displacement increments with threshold:**	**Equation (8.31)**
▪ **Displacement increment norm is greater than threshold: update stiffness matrix:**	
• **Transform force and displacement increments to intrinsic coordinates**	**Equations (8.20) to (8.25)**
• **Update intrinsic stiffness parameters**	
• **Transform the updated parameters to actuator local coordinates**	
▪ **Displacement increment norm is smaller than threshold: keep the experimental tangent stiffness unchanged**	
- **Calculate new accelerations, velocities, and displacements**	**Equations (8.14), (8.15), (7.4), (8.2) and (8.13)**
- **Check the convergence norm**	**Equation (8.4)**
• **Iterative solution scheme:**	
- **Failed: Use a one-step correction using the last successfully-updated tangent stiffness matrix**	**Equations (8.20) to (8.25), (8.14), (8.15), (7.4), (8.2) and (8.13)**
- **Succeeded: admit the iterative solution**	
• **Set** $n+1 \rightarrow n$ **and go to the next integration step**	

8.4.5 NUMERICAL AND EXPERIMENTAL VERIFICATIONS

In this section, numerical and experimental simulations have been carried out to illustrate the use of the proposed integration algorithm using experimental tangent stiffness matrix. This integration method is also compared to the conventional operator-splitting method, where only the initial experimental matrix is used.

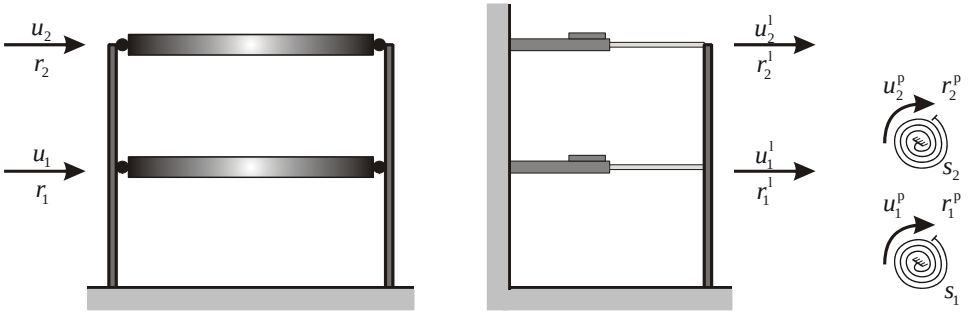

Figure 8-17 Two-degree-of-freedom structure, corresponding laboratory setup for experimental substructure column, and stiffness components in the intrinsic coordinate system.

8.4.5.1 Experimental Simulations

The two-story structure shown in Figure 8-17 is considered in the experimental studies for verification of the proposed integration method. The entire stiffness of the hybrid model is represented by a two-degree-of-freedom experimental substructure shown in Figure 3-2. Damping is numerically modeled to be 5% of critical in the first mode. The numerical mass matrix is selected to obtain natural periods of 0.60 and 0.15 seconds based on the measured initial substructure stiffness. The response of the structure subjected to the Tabas earthquake is simulated at the real-time event scale with integration time step of 10/1024 seconds. Real-time hybrid testing is used to demonstrate the computational efficiency of the integration scheme. Based on the observed measurement noise level, a displacement threshold of 0.1 mm has been chosen

to select steps with reliable data using Equation (8.31), for the estimation of stiffness matrix.

The stiffness matrix of the experimental substructure, given by Equation (8.12) is doubled in the numerical simulation to account for both columns of the two-story prototype structure. Since the column is continuous past the first story, the rotation of one end of the column with respect to the other end best describes the flexural behavior of each of experimental elements. Hence, the intrinsic coordinates for this structure can be defined as the deformations of two rotational springs at first and second story plastic hinges and their corresponding moments, as illustrated in Figure 8-8. That is, the diagonal stiffness matrix in intrinsic coordinate system is given by:

$$\mathbf{P} = \begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix} \tag{8.32}$$

where s_1 and s_2 are the stiffnesses of the first and second story springs, respectively. Considering that actuator local and global coordinate systems are the same in this problem ($\mathbf{T} = \mathbf{I}_2$), the displacement transformation matrix:

$$\mathbf{T}_p = \begin{bmatrix} 1/l_1 & 0 \\ -(l_1 + l_2)/l_1 l_2 & 1/l_2 \end{bmatrix} \tag{8.33}$$

transforms actuator displacements to spring rotations. In Equation (8.33), l_1 and l_2 are the lengths of the first and second story columns, respectively. Since the experimental substructure is statically determinate, the force transformation matrix can be found based on equilibrium:

$$\mathbf{T}_{\mathrm{p}}^{(-\mathrm{T})} = \begin{bmatrix} l_1 & l_1 + l_2 \\ 0 & l_2 \end{bmatrix} \tag{8.34}$$

This matrix transforms actuator forces to moments acting on the springs in the intrinsic coordinate system. Note that these transformation matrices satisfy $\mathbf{T}_{\mathrm{p}}^{\mathrm{T}}\mathbf{T}_{\mathrm{p}}^{(-\mathrm{T})} = \mathbf{I}_2$.

The structural response of the two-degree-of-freedom structure is simulated with 30 seconds of Tabas earthquake, scaled in amplitude by 35%. The global displacement histories are shown in Figure 8-18. As illustrated, the response is nonlinear, and residual drifts of 14 and 28 millimeters can be observed in the first and second stories, respectively. The energy balance of the system is well maintained throughout the simulation, as shown in Figure 8-19. This figure shows that the sum of analytical and experimental energies shows an excellent agreement with the input energy. Note that the simulation model does not have any numerical stiffness and numerical strain energy is zero throughout the simulation. The final energy balance error is less than 0.02%, which is very small. A similar experiment using a constant initial stiffness matrix for experimental substructure (conventional operator-splitting method), shows about 0.45% energy error at the end of simulation. This difference is small for this test structure, due to the fact that the amount of yielding is limited in the present experimental setup by the available actuator stroke. The improvement is expected to be larger in experiments with highly nonlinear experimental substructures, and will be demonstrated through numerical simulations in the next section.

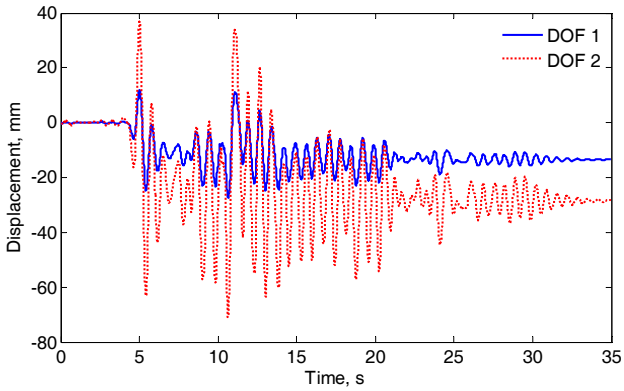

Figure 8-18 Displacement history of the two-degree-of-freedom structure subjected to 35% Tabas earthquake record.

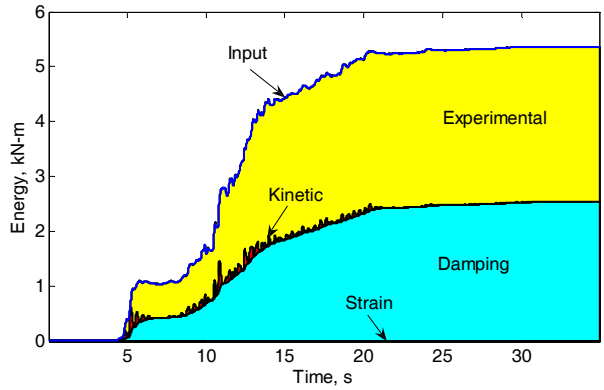

Figure 8-19 Energy histories computed through the hybrid simulation.

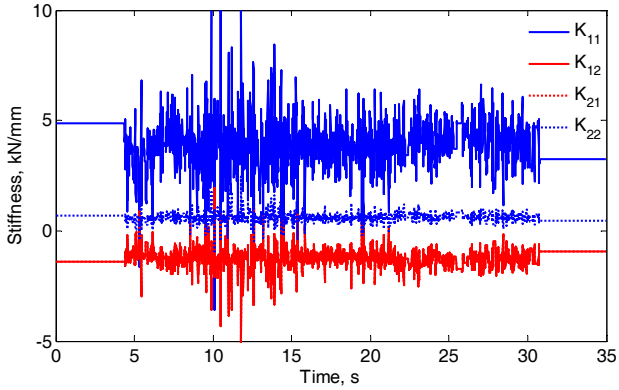

Figure 8-20 Stiffness matrix elements estimated through the simulation in actuator degrees of freedom.

In the above experimental simulation, the stiffness matrix was updated in 64.0% of the integration steps (85.3% for the period of significant response between 5 and 30

seconds). In other steps, the displacement norm was less than the noise threshold. The terms of the experimental tangent stiffness matrix through the simulation are shown in Figure 8-20. The stiffness matrix appears to have a fair amount of noise, which can be reduced by improved filtering of the measurements. Even so, this estimate is enough to reduce the energy error of the simulation well below that of conventional operator-splitting approach.

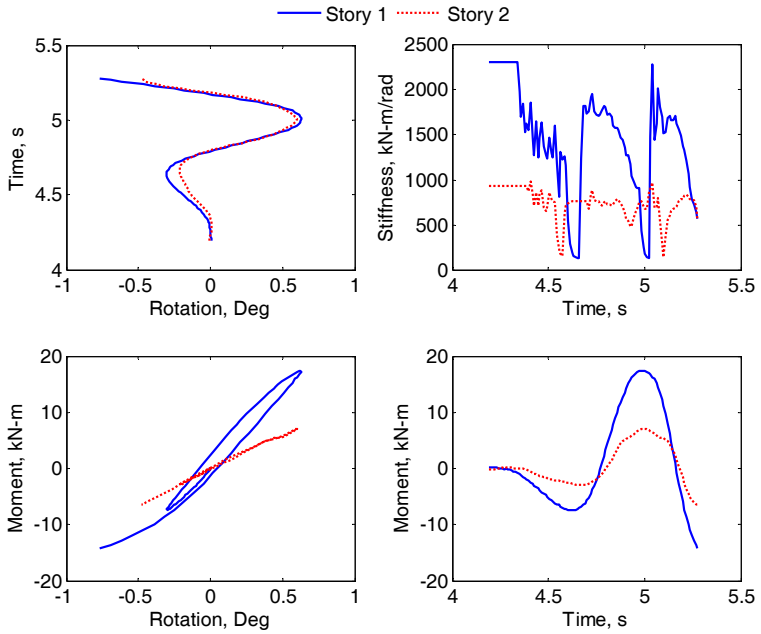

Figure 8-21 Variation of stiffness matrix elements and hysteretic behvior of the experimental setup in intrinsic coordinates (spring rotation).

The good agreement of the estimated spring stiffness with instantaneous behavior of the experimental substructure is shown in Figure 8-21 for a short period of simulation. Note that the gradual reductions in the stiffness coincide with softening behavior in hysteretic loops. The consequent increases in stiffness are faster, which correspond to elastic recovery at displacement reversals. The stiffness of the second-story spring remains close to the initial value, except for two peaks that occur immediately after the

rotation rate in this hinge becomes negative, which is in agreement with its force-displacement diagram. Since only one pair of coupons was used in the second story, this apparent reduction of stiffness is believed to originate from loose components of the experimental setup. The accuracy of the estimated experimental stiffness matrix can also be observed through the agreement of the converged and actual experimental hysteretic behavior of the first story hinge in Figure 8-22.

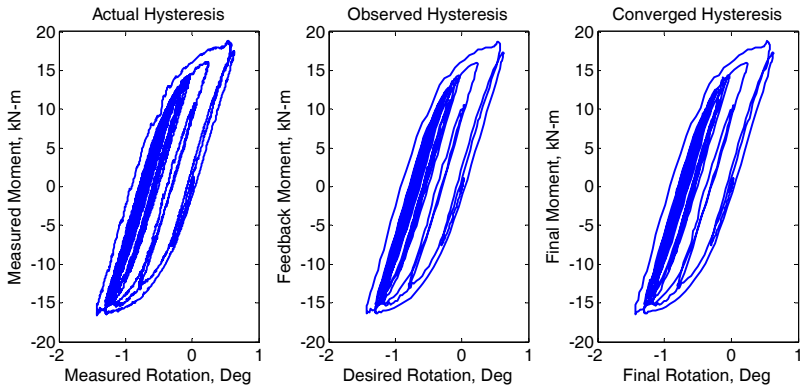

Figure 8-22 Comparison of actual, observed, and converged experimental hysteretic behavior.

8.4.5.2 Numerical Simulations

Numerical simulations have been carried out to compare different stiffness matrix estimation methods presented earlier, and to demonstrate the improved effect of using tangent stiffness matrix over the initial stiffness matrix in highly nonlinear experiments. The numerical models described in Chapter 3 are used to replace the experimental setup with similar properties. The delays have been selected to be 15ms for each actuator, and measurement noise has been calibrated based on actual experimental data and laboratory equipment information.

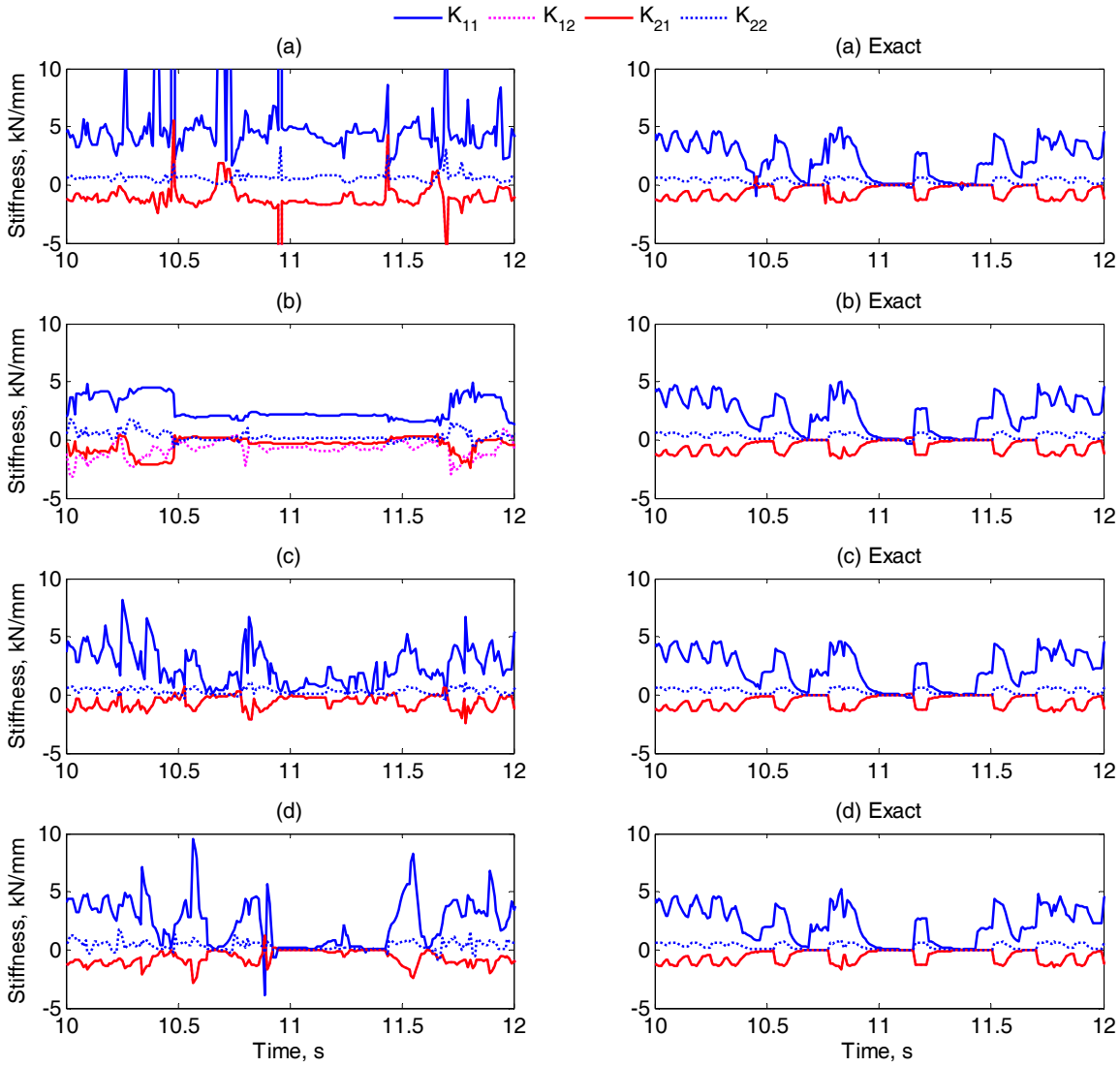

Figure 8-23 Estimation of tangent stiffness matrix using different methods (a) BFGS method, (b) Broyden formula, (c) proposed method using test setup information, and (d) proposed method using modal decomposition.

Comparison of Stiffness Matrix Estimation Procedures – In this section, the performance of the stiffness matrix estimation methods studied in this chapter are compared through numerical simulations. The seismic response of the structural model in Figure 8-17 subjected to Tabas earthquake with 50% amplitude scale is simulated, and the stiffness matrix histories estimated using different methods are compared. The same

displacement threshold of 0.1 mm is used for the all methods of stiffness estimation. In these simulations, the exact stiffness values are also calculated by directly using the force and displacement signals of the Bouc-Wen hysteretic models simulating the experimental response. That is, the input and output pairs of *Story 1* and *Story 2* blocks in Figure 3-18 are recorded and used to calculate the instantaneous stiffness of these models. These results do not include force errors, and reflect the actual stiffness of individual Bouc-Wen models in the experimental stiffness matrix. Note that such comparison is not possible in a hybrid numerical and experimental simulation.

The elements of stiffness matrix during a short period of these simulations are shown in Figure 8-23. This figure shows two sets of stiffness matrix histories: the first group on the left show the results of stiffness estimation methods presented in this chapter, and the group on the right demonstrate the exact results obtained during each simulation using the above-mentioned approach. Figure 8-23(a) shows that the stiffness matrix estimated using equation (8.17) is symmetric, but has a significant amount of large-amplitude spikes. This noise has been observed to result in considerable distortions in the converged hysteretic behavior discussed earlier. The agreement of the estimated stiffness matrix elements with exact values is also relatively poor. On the other hand, Equation (8.18) has a slower stiffness update rate, and some matrix elements do not adequately reflect yielding of the experimental model, as illustrated in Figure 8-23(b). Reducing the displacement threshold has also been observed to significantly increase the noise before achieving the desired adjustment rates. It can also be observed that Equation (8.18) results in unsymmetrical stiffness matrices in this simulation.

Similar observations have been made in simulations using other structures or earthquake records. Overall, the stiffness estimation procedures using Equations (8.17) and (8.18) did not show sufficient accuracy improvement in the simulation results over using the initial stiffness matrix.

Figure 8-23(c) and (d) demonstrate that the proposed stiffness estimation methods provide quick updates of the stiffness matrix elements with reduced noise. It can be observed that these methods result in the best agreement of the estimations with the exact results. The energy balance errors of the simulations using these methods are also distinctly smaller than using Equation (8.17), Equation (8.18), or the initial stiffness matrix of the experimental substructure. From Figure 8-23, it also appears that using the test setup information for diagonalization of the stiffness matrix produces more accurate estimations than using the modal matrix, but with no significant advantage in terms of energy balance. Similar results have been observed in other numerical and experimental simulations.

Effects of Using Updated Stiffness Matrix - In a numerical simulation, the level of nonlinearity of the experimental substructure can be easily altered by selecting different values for yield displacements of the experimental elements. Hence, it is possible to study the performance of the integration methods at different ductility levels. Numerical studies have been carried out for experimental models with yield displacements ranging from 5 to 70 millimeters resulting in ductility of about 6.0 to linear response. The structural response subjected to 50% Tabas earthquake is simulated using integration time steps of 10/1024 and 20/1024 seconds.

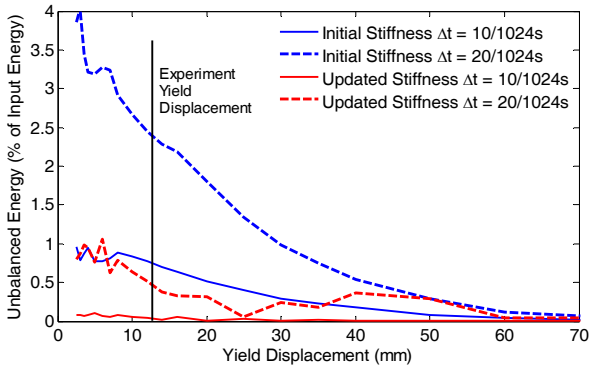

Figure 8-24 Absolute energy balance error at the end of simulation for different yield displacements.

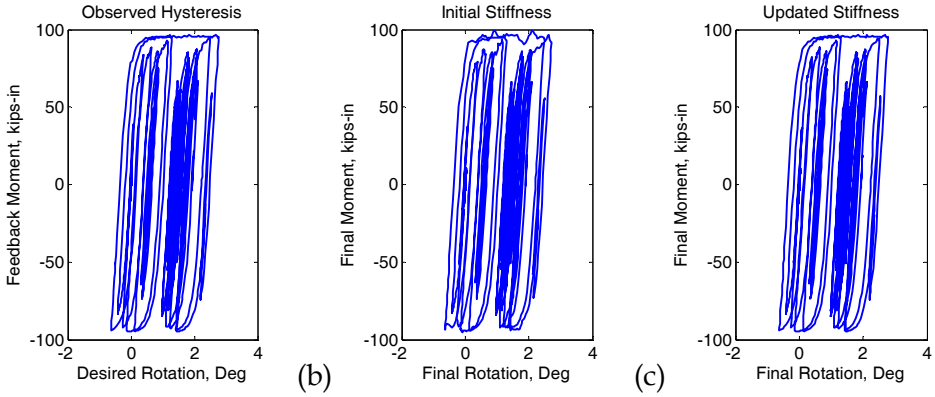

Figure 8-25 Hysteresis behavior of first story spring (a) observed (b) converged using initial stiffness matrix (c) converged using updated stiffness matrix.

Figure 8-24 illustrates that the unbalanced energy is always smaller when the tangent stiffness matrix is updated. This error continuously increases with higher levels of nonlinearity, but increases at a higher rate when the initial stiffness matrix is used. The use of initial stiffness is shown to result in comparable accuracy only when the response is close to linear (large yield displacements). Hence, it can be concluded that the stiffness matrix update method presented in this section improves the accuracy of simulation of nonlinear systems and becomes more beneficial for higher ductility levels. The comparisons of hysteretic loops for a yield displacement of 10mm and integration time step of 10/1024s are shown in Figure 8-25. It is illustrated that the use of initial

stiffness matrix may result in distortions in the yielded portions of hysteretic loops (Figure 8-25(b)), while such distortions are eliminated when the tangent stiffness matrix is used.

The benefit of using experimental tangent stiffness matrix is also evident when larger time steps are used in simulation of nonlinear systems, as shown in Figure 8-24. This can be attributed to the use of Equation (8.13) for updating the feedback force vector. With larger time steps, the differences between predictor and corrector displacements increase, and the effect of force corrections will be more pronounced in simulation results. The increased freedom to use larger time steps without significant loss of accuracy is therefore another advantage of the integration procedure presented in this section.

Use of Tangent Stiffness Matrix in Iterative Schemes – The effects of iterative estimation of stiffness matrix using polynomials fitted to the measurements are studied in this section. Numerical simulations have been carried out for the two-degree-of-freedom model subjected to 50% amplitude Tabas earthquake. The operator-splitting method using the tangent stiffness matrix has been observed to result in a small energy balance error of 0.05%, and the converged hysteretic behavior is in agreement with the actual behavior as shown in Figure 8-26. Figure 8-27 shows the elements of stiffness matrix during the simulation. In this simulation, 71.6% steps were completed with successful updates of stiffness matrix.

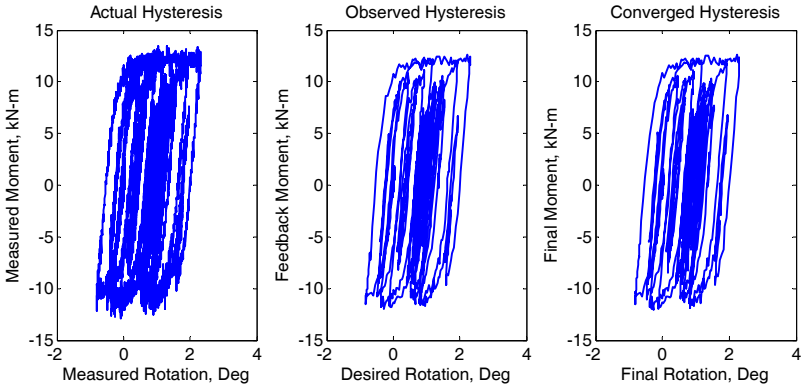

Figure 8-26 Hysteretic behavior of the first story spring obtained by operator-splitting procedure using tangent stiffness matrix.

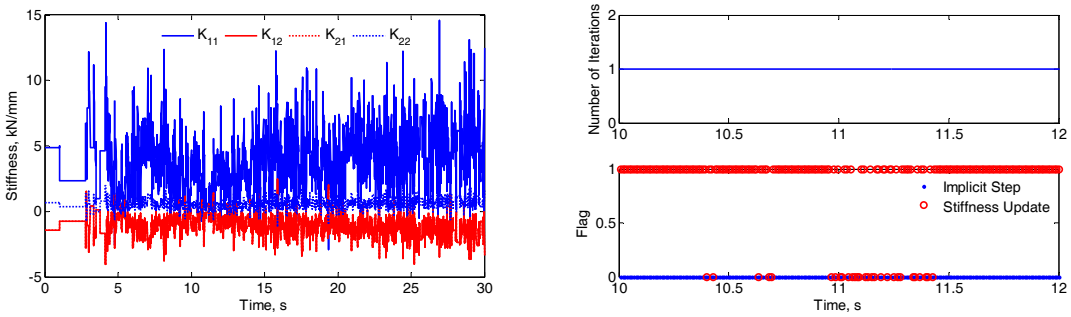

Figure 8-27 Stiffness matrix elements, number of iterations, and flags for implicit steps and stiffness update.

The simulation results shown in Figures 8-28 and 8-29 are obtained using the iterative stiffness update procedure described in Section 8.4.4. Here, the iterative results with failed stiffness update procedure are accepted as solutions with increased convergence tolerance. With this strategy, 100% of the steps are considered implicit; that is, the convergence is achieved before the maximum number of iterations. However, the stiffness update has been successful only in 30.0% of steps. This reduced percentage can be attributed to the fact that stiffness update is considered successful only when it is successful in all iterations up to the convergence. In all other steps, the iterative procedure had a "false convergence" due to unchanged stiffness matrix in iterations. The energy balance error at the end of this simulation was about 0.90%.

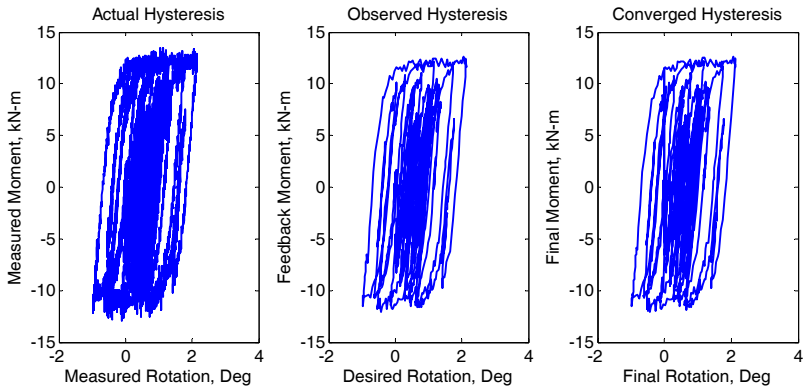

Figure 8-28 Hysteretic behavior of the first story spring obtained by iterative integration procedure using tangent stiffness matrix – admitting the last iterative solution in case of stiffness update failure.

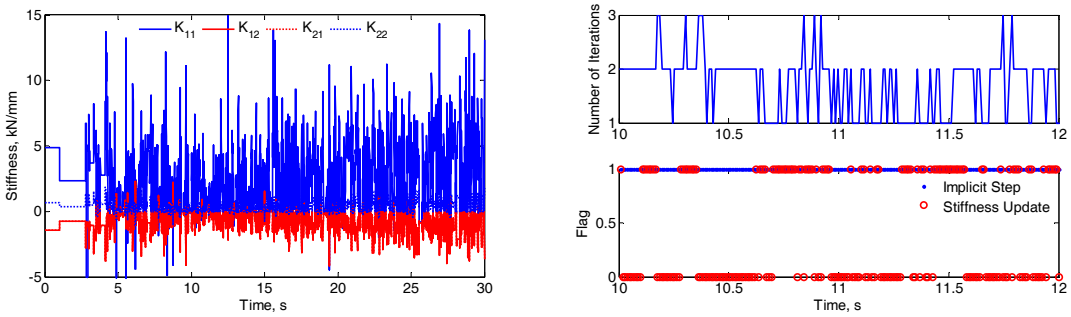

Figure 8-29 Stiffness matrix elements, number of iterations, and flags for implicit steps and stiffness update.

In another simulation, the steps with failed stiffness update have been considered non-implicit, and an operator splitting procedure has been utilized to replace the results using experimental tangent stiffness matrix. In this case, the percentage of steps with successful iterative scheme reduces to about 31.1%. On the other hand, the stiffness update is successful in about 72.3% of steps, which has been used either in iterative, or operator-splitting scheme. The results of this simulation are shown in Figures 8-30 and 8-31. The energy balance error at the end of simulation is about 0.11%. The energy balance errors obtained from iterative schemes seem to be more than that of operator-splitting with tangent stiffness. However, the converged hysteretic loops are in good

agreement with the actual ones in both iterative integration schemes. The noise in the estimated stiffness matrix elements also appear to be larger in iterative methods.

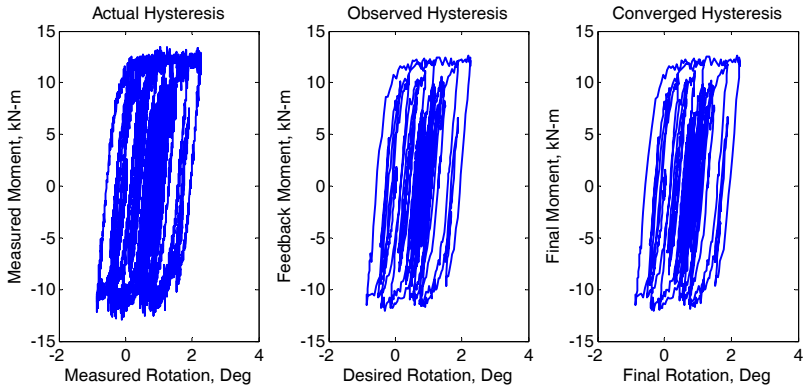

Figure 8-30 Hysteretic behavior of the first story spring obtained by iterative integration procedure using tangent stiffness matrix – reverting to operator-splitting in case of stiffness update failure.

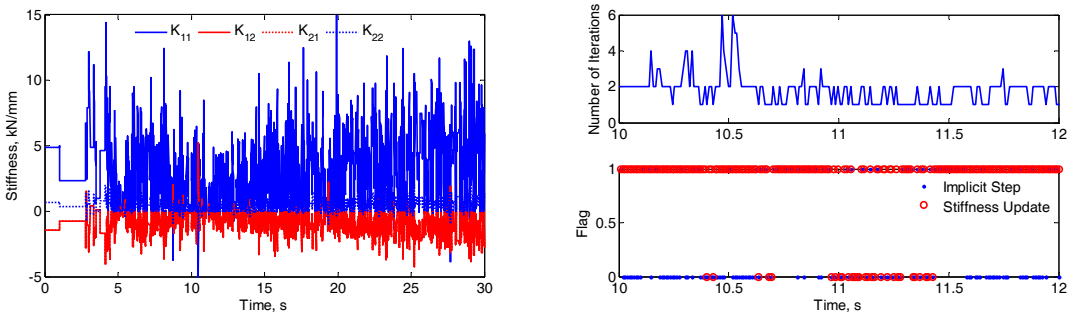

Figure 8-31 Stiffness matrix elements, number of iterations, and flags for implicit steps and stiffness update.

The amounts of energy balance error found in these simulations are observed to be sensitive to the noise in force measurements. Table 8-5 shows how energy balance and percentage of iterative solutions are affected by the level of noise in force measurements. Based on this data, the iterative solution (with reverting to operator-splitting in cases of stiffness update failure) shows better results in the absence of force measurement noise. Hence, the use of iterative stiffness update procedure does not show a noticeable advantage over a single-step correction of states using tangent stiffness due to experimental errors and measurement noise. However, this iterative integration scheme

may be beneficial for use with commercial finite element analysis programs that employ iterative solution schemes for nonlinear numerical and experimental substructures. With this approach, fully implicit integration methods can be used for both experimental and numerical substructures.

Table 8-5 Energy errors and percentages of iterative solutions for different integration methods.

Method Description	Full Force Measurement Noise			25% Force Measurement Noise			No Force Measurement Noise		
	Energy Error	Stiffness Update	Iterative Solution	Energy Error	Stiffness Update	Iterative Solution	Energy Error	Stiffness Update	Iterative Solution
Operator-splitting with tangent stiffness	**0.05%**	**71.6%**	**0%**	**0.08%**	**78.3%**	**0%**	**0.08%**	**79.0%**	**0%**
Iterative with admitting the last iterative solution in case of stiffness update failure	**0.90%**	**30.0%**	**100%**	**0.61%**	**31.8%**	**100%**	**0.28%**	**33.3%**	**100%**
Iterative with reverting to operator-splitting with tangent stiffness in case of stiffness update failure	**0.11%**	**72.3%**	**31.0%**	**0.05%**	**78.5%**	**37.1%**	**0.04%**	**79.3%**	**39.0%**

An Example for Using Modal Diagonalization of Stiffness matrix – In this section, the zipper frame structures shown in Figure 8-32 is simulated using stiffness matrix estimation with modal diagonalization approach. A similar substructure was tested pseudo-dynamically by Yang *et al.* [81]. As shown, six global degrees of freedom have been selected for this three-story frame. The three vertical degrees of freedom are considered to observe the effect of vertical elements in carrying vertical unbalanced loads when buckling occurs in the lower braces. The mass and stiffness matrices are selected to result in a first-mode period of 0.6 s, with the natural period of the highest

mode being 0.05 s. Damping is assumed to 5% of critical and is modeled in the numerical substructure.

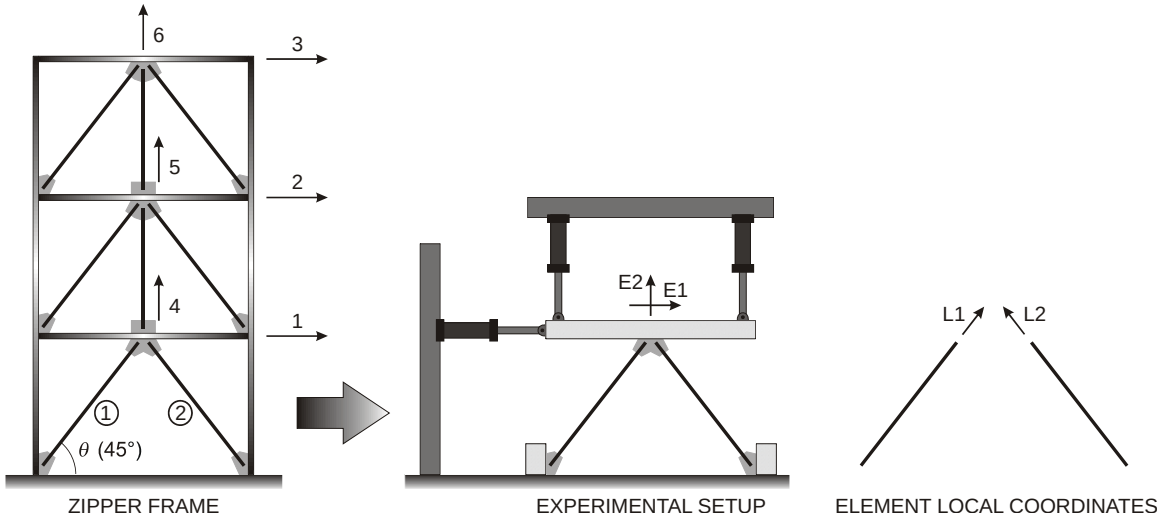

Figure 8-32 Example zipper frame structure and first story brace as the experimental substructure.

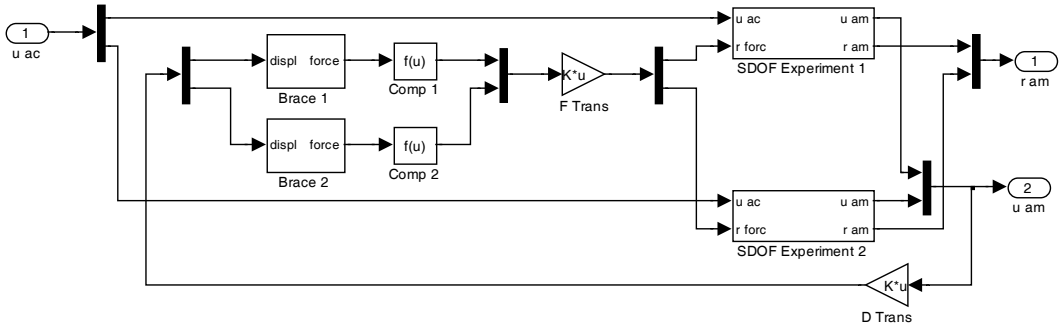

Figure 8-33 Simulink model for numerical simulation of experimental substructure.

The Simulink model shown in Figure 8-33 is used to model the behavior of the first story braces on the computer. This model first transforms the displacements to element local coordinates, and uses two Bouc-Wen models to calculate the restoring force for each element. The elements are assumed to be softer in compression, by reducing the negative restoring forces by 30%. This simple approach results in an overall behavior similar to that of a bracing system with buckled elements. The forces are then

transformed to actuator coordinate system. Delay and other errors are also introduced in the signals using the same methods presented in Chapter 3.

In order to simulate the structural response using the stiffness estimation methods proposed in this chapter, it is necessary to select a diagonalization approach. Using the experimental setup information, the intrinsic coordinate system can be selected as the combination of the element local coordinate systems. That is, the transformation matrix is given by:

$$\mathbf{T}_{\mathrm{p}} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\cos\theta & \sin\theta \end{bmatrix} \tag{8.35}$$

which transforms the displacements from actuator to element coordinates. For force transformation, the above matrix should be transposed and inverted, to give member axial forces from forces in actuator coordinate system. Alternatively, the modal diagonalization method described in Section 8.4.2.2 can be selected for the stiffness update process, without using the above information about the test setup. In any case, the initial stiffness matrix of the experimental substructure is required to start the simulation. Here, the initial stiffness matrix is selected based on the symmetric behavior of elements in tension and compression, which is only the case within the linear range, before the occurrence of buckling.

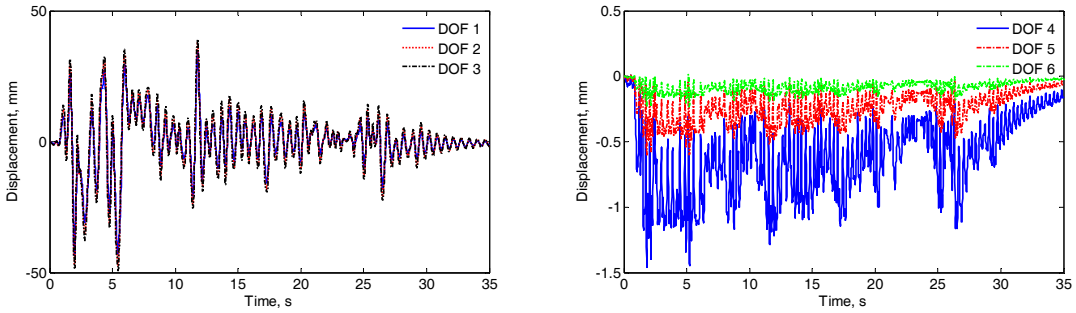

Figure 8-34 Global displacement histories of the example zipper frame.

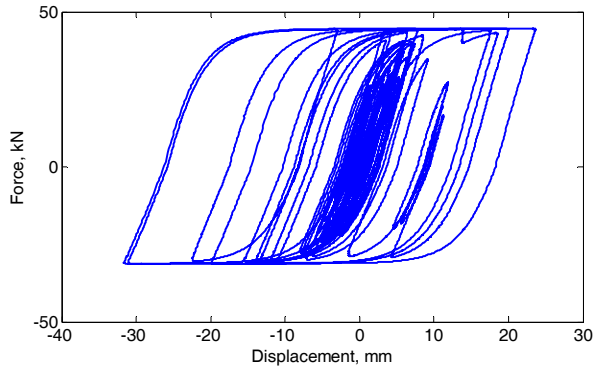

Figure 8-35 Hysteretic behavior of brace number 1, with compressive stiffness being 30% less than tensile stiffness.

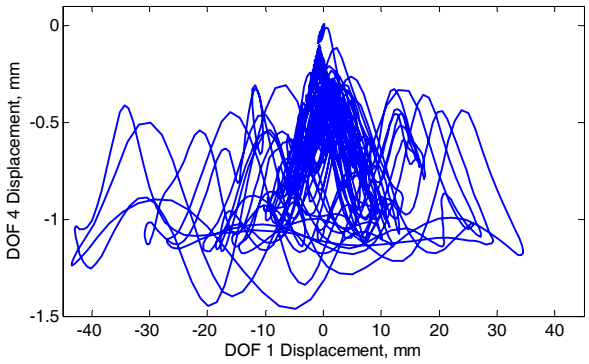

Figure 8-36 Horizontal and vertical displacements of the top node of first story brace.

The displacement results of the simulation of the zipper frame subjected to El Centro ground motion is shown in Figure 8-34. Note that the nonlinearities are only limited to the braces in the first story, which are modeled as the experimental substructure. The hysteretic behavior of the experimental brace number 1 is illustrated

in Figure 8-35, which shows the reduced stiffness of the brace in compression. This unsymmetrical behavior of the elements results in the top experimental node to sustain a permanent downward dislocation, as shown in Figure 8-36. This figure shows the oscillation of this node in both horizontal and vertical directions, with vertical displacements being significantly smaller than horizontal displacements. It is also evident that vertical oscillations occur with a higher frequency than horizontal vibrations.

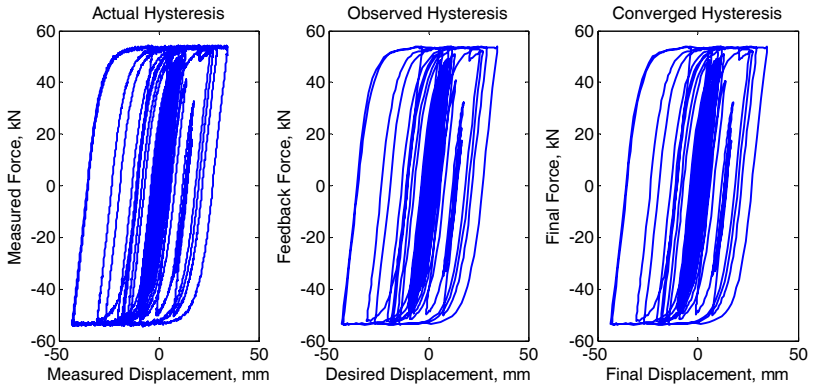

Figure 8-37 Force-displacement diagram of the horizontal experimental degree of freedom (E1).

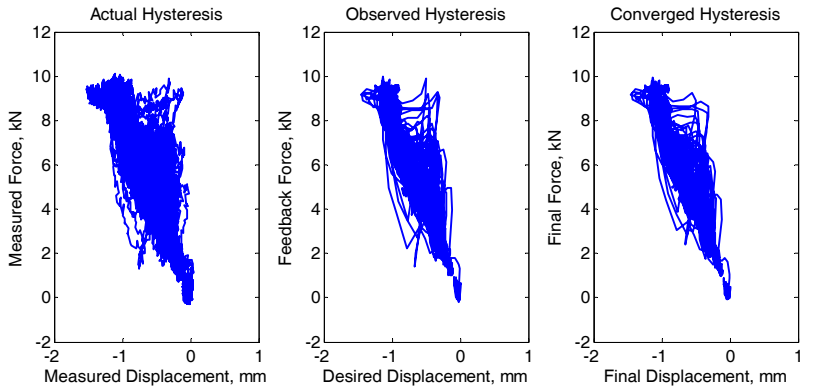

Figure 8-38 Force-displacement diagram of the vertical experimental degree of freedom (E2).

The force-displacement diagrams for both actuators are shown in Figures 8-37 and 8-38. The horizontal force-displacement diagram shows a typical nonlinear behavior, while the vertical behavior is more complicated. The overall trend in vertical direction

appears to show a negative stiffness. This implies the development of tensile forces in the vertical actuators as the top node of the bracing underneath moves downward. This is in agreement with the role of the vertical elements of the zipper frame in carrying the unbalanced vertical loads of lower bracing elements after buckling [126]. The agreement of converged and actual hysteretic behaviors is also evident in Figures 8-37 and 8-38, demonstrating the accuracy of the stiffness estimations.

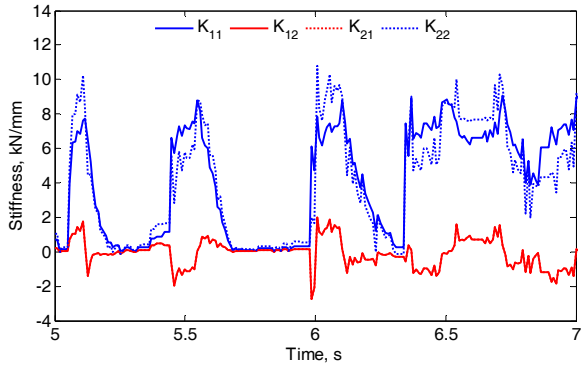

Figure 8-39 Stiffness matrix elements during the peak structural response.

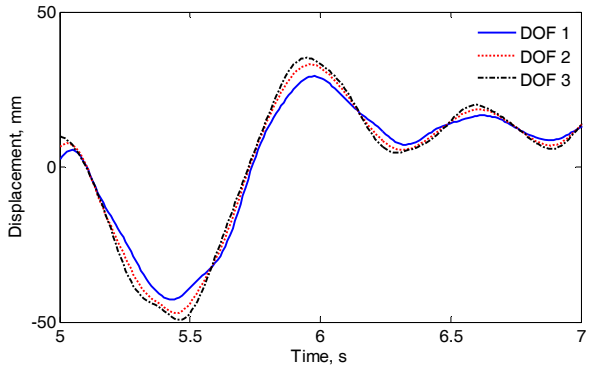

Figure 8-40 Displacements at the first three degrees of freedom during the peak structural response.

The elements of experimental tangent stiffness matrix in actuator local coordinate system during the peak structural response are shown in Figure 8-39. The displacement histories of horizontal degrees of freedom during this period are also detailed in Figure 8-40. It can be observed that during large drifts, the stiffness of the brace elements

reduce to values close to zero. At displacement reversals, the stiffness matrix elements are observed to sharply increase, and then, gradually decrease in the opposite direction.

It should be noted that a linear 45° bracing system should have a diagonal stiffness matrix with identical stiffness in horizontal and vertical directions. However, during the simulation period shown in Figure 8-39, the horizontal stiffness appears to be greater when velocity is positive. In negative-velocity portions, the vertical direction shows larger stiffness values. These periods are accompanied by negative and positive off-diagonal terms of stiffness matrix, respectively. These observations can be attributed to the unsymmetrical behavior and yielding of the elements in tension and compression, as described above. In this simulation, the experimental stiffness matrix was updated in 76.2% of integration steps, and the final energy balance error was less than 0.1% of input energy.

9 SUMMARY AND CONCLUSIONS

A summary of key contributions and concluding remarks from this dissertation are presented in this chapter. Research areas for further developments and future studies are also identified and briefly discussed.

9.1 SUMMARY

This work was conducted for the development of pseudo-dynamic hybrid simulation system at SEESL, and the improvement of hybrid simulation procedures for real-time and geographically distributed experiments. Only real-time experiments are presented in this dissertation, but the minimal communication requirements of the proposed methods make them suitable for distributed testing at slower rates. In this section, a concise summary of this dissertation is presented.

The state-of-the-art of hybrid simulation test technique was presented in chapter 2. It was found that hybrid simulation has been widely recognized as a reliable and effective test technique for evaluation of dynamic performance of structural components and systems. As it stands, the most important challenges in the extension of this test

method to fast and distributed testing of complex structural systems are: (i) difficulties in setting up proper boundary conditions for experimental substructures, (ii) complex and unidentified dynamics of the test setup, (iii) effects of numerical and experimental errors on simulation results, (iv) lack of robust numerical simulation procedures for conducting stable and accurate experiments, and (v) absence of a general, user-friendly and fully-developed software framework for real-time hybrid simulations. This study was conducted to address issues related to items ii, iii, and iv above, by developing improved simulation procedures.

Basic simulation models were developed for the Structural Engineering and Earthquake Simulation Laboratory at University at Buffalo. In addition, a general two-degree-of-freedom experimental setup was developed and presented in Chapter 3 for experimental verification of test procedures. The hybrid simulation models take advantage of MATLAB Simulink development environment and xPC target computers for real-time simulations. All the numerical integration, delay compensation, and signal conditioning procedures studied in this dissertation are implemented in the simulation models. In hybrid simulations, these models are executed on real-time computers, and communicate with test setup via SCRAMNet.

Also in Chapter 3, numerical models were described that can be used to replace the experimental setup for numerical simulations of hybrid simulation. These models mimic the performance of experimental setups by artificially introducing errors and delay in force and displacement signals. Pretest simulations and preliminary evaluation of new test procedures are two most important uses of these numerical models. Since the

experimental forces are numerically calculated in these models, numerical simulations can be carried out in either real-time or non-real-time environments.

The functions of servo-hydraulic actuators and their most common control systems were studied in Chapter 4. It was shown that the control system parameters considerably affect the stability and tracking performance of the actuators. The properties of the most widely used PID controllers were also studied; it was shown that these linear controllers are able to yield satisfactory actuator response, if they are properly tuned for the intended frequency range. For this reason, the tuning should be repeated after any major change in the experimental setup or specimen properties. It was demonstrated that the properties of PID controllers can be improved by small adjustments to the procedure, such as delta-pressure stabilization for elimination of oil column resonance, and feedforward gain for reduction of response delay. It was also pointed out that some delay must be admitted to exist in the actuator response to achieve a sufficiently stable behavior from this controller.

Numerical models developed for servo-hydraulic actuators were also studied in Chapter 4, ranging from simple linear models to complex models including the nonlinear dynamics of servovalve and orifice flow. The linear models have been shown to produce acceptable results for a limited actuator response range, where the behavior can be assumed linear. The more complex and nonlinear models normally produce better results, but they were shown to be sensitive to the accuracy of the provided system parameters, such as oil bulk modulus. Consequently, it may be necessary to repeat the numerical simulations for a wide range of system parameters to account for

their uncertainties. These models can be used to predict the system dynamics during simulations, and design controllers and compensators to improve the actuator performance.

In Chapter 5, it was shown that both numerical and experimental errors affect the accuracy and stability of hybrid simulations. The majority of numerical errors normally occur due to the use of discretized numerical models and integration methods that are highly simplified for use in hybrid simulations. Compared to experimental errors, numerical errors are better identified, and normally can be restricted by following appropriate modeling guidelines.

Experimental errors, which can be categorized as random or systematic, often originate from actuator tracking errors and measurement noise. These errors were shown to have the most detrimental effects on the stability and accuracy of hybrid simulations. In particular, servo-hydraulic actuator delay was studied in detail, and it was shown that uncompensated delay may destabilize the system by adding energy through an apparent negative damping effect. These effects were shown to be more significant in the presence of high-frequency modes. Hence it is important to first make every effort to reduce this systematic error, and then, properly compensate the remainder during fast and real-time hybrid simulations.

It was shown that delay can be measured offline if it is expected to remain constant during the simulation. However, it has been demonstrated that delay may change during simulations with large stiffness variations. For this reason, the existing online delay estimation procedures were studied and an improved method was proposed in

Chapter 5. Error indicators were also studied in this chapter for the assessment of accuracy and stability of hybrid simulations. An experimental error indicator was examined and used to show the importance of proper delay compensation for accuracy and stability of hybrid simulations. In order to include both numerical and experimental errors, an overall energy balance error was formulated and applied to hybrid simulations.

Delay compensation procedures for hybrid simulation were studied in Chapter 6. It was pointed out that in hybrid simulations, the properties of the experimental substructures are unidentified before the experiment; this is in contrast with delay compensation problem for active control of structures, where normally a reasonable numerical model of the structural system is available. Consequently, mostly compensation procedures that need the least information about test structure have been adapted to hybrid simulations. In particular, the use of Smith Predictor was demonstrated in linear hybrid simulations. It was shown that an adaptive implementation of Smith Predictor may be suitable for compensation of delay in nonlinear simulations.

The polynomial extrapolation method was also studied in Chapter 6 as the most common delay compensation method in displacement signal. In addition, alternative methods were proposed for delay compensation in hybrid simulations that apply corrections in force signal or use the numerical integration procedure to predict the delay-compensated displacements.

Numerical integration procedures for hybrid simulation were studied in Chapters 7 and 8. In Chapter 7, the formulations for the most common numerical integration methods were presented, and the necessary modifications were applied for use in hybrid simulations. The most widely-used explicit and operator-splitting integration methods were also presented, and their stability and accuracy properties along with their benefits and shortcomings for hybrid simulation were studied. Following this chapter, improved numerical integration methods were introduced in Chapter 8.

9.2 CONCLUSIONS

In this section, a summary of the major contributions of this dissertation is presented. These include an online delay estimation procedure, an energy-based error indicator, delay compensation methods based on force correction and numerical integration, and two improved numerical integration methods for hybrid simulation.

9.2.1 ONLINE DELAY ESTIMATION

In hybrid simulations with significant stiffness variation, delay may change during the simulation, requiring variable compensation of delay. For this purpose, a procedure for online estimation of delay was proposed that compares the desired and measured displacement signals in the actuator coordinate system. This procedure only requires a learning gain, and does not need any *a priori* information about the experiment setup. Computer simulations and experiments demonstrated its fast convergence with reduced oscillations, compared to the existing online delay estimation procedures.

The proposed delay estimation procedure was used with all compensation methods studied herein to demonstrate its effectiveness with minimal dependency on the simulation properties. It was observed that the adjustment of learning gain may be necessary only when significant modifications are applied to the experimental test setup and instrumentation.

9.2.2 DELAY COMPENSATION METHODS

Servo-hydraulic actuator delay is one of the most influential experimental errors in real-time hybrid simulations. Since it is virtually impossible to completely eliminate the delay in the experimental setup, compensation procedures are essential for stable and accurate hybrid simulations. Hybrid simulation delay compensation procedures were considered in the command signal sent to the actuator and force measurement signal fed back to the numerical integration procedure (Chapter 6).

As an alternative to polynomial extrapolation method, the use of the numerical integration procedure for prediction of command displacement was proposed. In particular, delay compensation using the finite difference kinematic expression assuming constant acceleration was successfully implemented in SDF and MDF experiments. Similar to the polynomial extrapolation method, this method can be used with any integration procedure by modifying the time step of the equation estimating the next actuator command displacement.

Modifications of force measurements were also studied for compensation of delay and actuator tracking errors. It was observed that estimation of the forces corresponding

to the desired displacements using recent measurements is an effective compensation method. This procedure can be combined with the above-mentioned displacement extrapolation methods to correct the force signal for uncompensated delay and tracking errors.

Command displacement delay compensation methods were compared by studying their phase and amplitude errors in prediction of a sinusoidal signal. Analytically, the use of numerical integration procedure demonstrated a comparable performance to that of polynomial extrapolation. However, in earthquake simulations, the proposed procedure was shown to reduce the high-frequency noises in the force measurements by producing a smoother command displacement signal than that of polynomial extrapolation.

Numerical and experimental simulations were also carried out for the verification of the performance of different delay compensation methods. This study showed that command displacement compensation procedures are more effective in the compensation of relatively large amounts of delay, while both procedures showed acceptable results for compensation of moderate delays.

9.2.3 ENERGY-BASED ERROR INDICATOR

For assessment of the reliability of hybrid simulation results, an error indicator based on the overall energy balance was introduced in Chapter 5. This procedure expands the existing methods that only monitor the experimental errors by considering both experimental and numerical errors. The energy balance equation was modified to

include the effects of: (i) the discrepancies between the actual experimental energy dissipation (hysteretic behavior) and that conceived by the numerical simulation module, (ii) the improper kinematic relations between displacement, velocity, and acceleration, and (iii) the inexact satisfaction of the governing equation of motion. The hybrid simulation error indicators studied in this dissertation can be monitored during the experiments to ensure the proper functioning of test components and stop the simulation in case of excessive errors, possibly before damaging the experimental substructure.

9.2.4 NUMERICAL INTEGRATION METHODS

In Chapter 8, new integration procedures were proposed for hybrid simulation with improved stability and accuracy properties. First, the conventional operator-splitting method was slightly modified to provide a predictor displacement closer to the corrector displacement in each integration step. Then, two improved integration methods were introduced: an integration method with combined implicit or explicit steps, and a method for estimation of experimental tangent stiffness matrix for use in operator-splitting method.

9.2.4.1 Integration Method with Combined Implicit or Explicit Steps

A new integration scheme with combined implicit or explicit step was proposed in Section 8.3 for real-time hybrid simulations. This method captures the instantaneous behavior of the experimental substructures by using the most recent measurements to satisfy an implicit formulation in a majority of the integration steps. Only measured

forces and displacements from experimental substructures are used in the iterative scheme, to avoid the direct application of iterative displacements that may damage the experimental substructures. By using the measurements in the force estimation procedure, this method does not require the initial experimental stiffness matrix, and better captures the actual behavior of experimental setup.

The accurate estimation of forces corresponding to iterative displacements and convergence of the iterative scheme cannot be guaranteed due to system nonlinearities and experimental errors. To address this issue in the proposed integration method, the implicit integration scheme is slightly modified such that the states are updated similar to an explicit procedure. This modification enables the procedure to revert to an explicit or operator-splitting scheme in cases of convergence failure, to ensure the completion of the integration step and continuity of the simulation.

It was shown that the proposed integration method is able to eliminate spurious excitation of high-frequency modes and use longer time steps compared to explicit methods. This approach was shown to produce stable and accurate simulations when the explicit integration methods fail. Smaller overall energy balance errors were also obtained compared to the conventional operator-splitting method in nonlinear simulations. In addition, this integration method reduces the required communications between numerical and experimental subsystems, as the exchange of command displacements and acquisition of measurements occur only once within each integration step. These features make the proposed integration algorithm appealing for testing

large, stiff or highly nonlinear systems, or applications in geographically distributed simulations.

9.2.4.2 Estimation of Experimental Tangent Stiffness Matrix

An improved numerical integration procedure was proposed in Section 8.4 based on the estimation of tangent stiffness matrix of the experimental substructure. In this approach, necessary stiffness parameters are first identified and updated during the simulation using the incremental measured force and displacement vectors. Only significant force-displacement pairs are used to update these parameters; steps with small displacement increments are ignored to ensure the fidelity of the results. The estimated stiffness parameters can then be used to determine a reduced experimental stiffness matrix through a simple coordinate system transformation.

In order to use the measurements to update the stiffness matrix, a transformation was established to an intrinsic coordinate system, in which the stiffness matrix is diagonal. This transformation can be found by considering the geometry and element configuration of the experimental substructure, or by using the modal matrix of the tangent stiffness matrix for classical diagonalization.

The estimated tangent stiffness matrix can replace the initial elastic experimental stiffness matrix for improved accuracy in a variety of integration procedures. In this study, a modified operator-splitting method was employed using the experimental tangent stiffness matrix. It was demonstrated that the use of the updated experimental stiffness matrix improves the overall accuracy of nonlinear hybrid simulations. This

improvement was demonstrated through the reduced energy balance error in experimental and numerical simulations. In numerical simulations, it was shown that the improved accuracy of the proposed approach is more evident in highly nonlinear experiments, where using the initial stiffness matrix may no longer be a reasonable approximation.

Estimation of the experimental tangent stiffness matrix increases the computational cost of the proposed integration procedure. However, since most experimental substructures have only a few degrees of freedom, the additional computational burden is often tolerable using currently available processing tools. Further, in each step of this integration method, the interface forces and displacements are transferred once between analytical and experimental subsystems. Hence, this procedure has the minimum communication requirements and is effective in geographically-distributed experiments.

In order to fully implement an iterative implicit integration scheme in hybrid simulation, the use of experimental tangent stiffness matrix was also studied in iterative schemes. In this method, it is attempted to include the possible changes of the tangent stiffness matrix between predictor and corrector displacements. For this purpose, polynomials fitted to the force and displacement measurements were used to avoid physical applications of iterative displacements. Using these polynomials, the force and displacement increments are updated in each iteration, and used to estimate a new tangent stiffness matrix. In the numerical simulations, this method did not show a significant advantage over the single-step operator-splitting integration scheme with tangent stiffness, primarily due to the noise in force measurements. However, this

method can be beneficial for simulations involving commercial finite element analysis software that use iterative implicit solution schemes.

9.3 *FUTURE DEVELOPMENTS*

Identification of seismic performance of new and complex structural components and devices at large scale is one of the most important features of hybrid simulation technique. This method becomes more appealing as novel structural systems and more sophisticated design procedures are developed. Examples of these components are semi-active dampers and variable stiffness devices, isolation systems or sensitive nonstructural components. Performance-based design of structures is also an example of design procedures that require understanding of the nonlinear behavior of structures and components. Hybrid simulation methods also have a potential for accurate, safe, and low-cost evaluation of the behavior of structural components through structural collapse.

In order to achieve the above-mentioned goals, robust numerical simulation procedures that are able to carry out nonlinear and large-deformation analyses parallel to experimental tests should be implemented in simulation models. The current numerical simulation models at SEESL only include basic procedures for conducting relatively simple numerical simulations. Further development of these models is a necessary step for effective and prevalent use of this test technique. In addition, to make hybrid simulation a usable procedure by researchers in different areas of civil engineering, software packages that have the above-mentioned capabilities and a user-

friendly interface should be developed. This development may involve the use of software packages developed in other laboratories across the NEES. In this case, these packages should be customized for the test system at SEESL and their necessary subroutines should be produced.

Further development of computer simulation models of hybrid simulation is also required for more robust numerical experiments that take into account nonlinearities in both experimental and numerical substructures. To extend their capabilities to highly nonlinear problems such as collapse analysis, more general material models (e.g. rate-dependent or deteriorating models) should be considered in the development of the new numerical models of hybrid simulation.

The present hybrid simulation experimental setup at SEESL has a limited ability for nonlinear simulations, primarily due to short actuator strokes. For this reason, often numerical models were used in this study to perform highly nonlinear simulations. Experimental verifications are necessary to better demonstrate the benefits and deficiencies of test procedures in highly nonlinear systems. For this purpose, setting up a more robust experimental setup is a valuable step in the verification of test procedures studied in this dissertation.

Currently, mostly linear control systems are used for control of servo-hydraulic actuators. Although these systems are simple and often do not need a numerical model of the actuators (the plants), their performance is limited, and they need to be exclusively tuned for each experimental setup. Any improvement to the actuator control systems to reduce errors and achieve better tracking performance is greatly valuable for

hybrid simulations. As a novel family of controllers, neurocontrollers have recently received considerable attention in different engineering areas, including active control of structures. It has already been shown that these controllers are able to handle delay and measurement noise better than ordinary feedforward controllers. In order to take advantage of their promising features, utilization of these novel controllers for servo-hydraulic actuators should be studied in hybrid simulations.

In this study, for mitigation of actuator undershooting in displacement reversals, a variable gain was applied to the command displacements. The gain was calculated online using the difference between peak command and measured displacements, with respect to the actuator equilibrium point (zero displacement). This procedure, however, was only developed in simulations with SDF experimental substructures. The interaction of actuators in MDF experimental substructures makes the determination of actuator equilibrium point difficult, leading to an oscillatory command gain. More work is needed in the extension of this compensation method to MDF experimental substructures. Utilization of numerical models of experimental setup that are updated during the simulation may help better calculate the zero displacement and command displacement gain in MDF systems.

The hybrid simulation error monitor and energy error indicator presented in Chapter 5, provide information about the amount of unbalanced energy in hybrid simulations. This unbalanced energy may be used as a feedback for correction or compensation of simulation signals to improve energy balance of the simulation system. As an example, it may be possible to modify the damping ratio of the test structure, or

select the integration parameters to alter the numerical energy dissipation towards a better energy balance. Implementation of this technique requires further study of the energy errors and their relations with structural properties and simulation parameters.

For an integration procedure, the importance of a numerical model of the experimental substructure is that it eliminates the need for iterations, or physical application of iterative displacements on experimental substructures. The model-based integration method introduced in Section 8.4 was based on this concept. However, since it only uses a stiffness matrix that is updated during the simulation, this method is more suitable for essentially strain-dependent experimental substructures. For further extension of this procedure, more general models of the experimental substructure and setup should be used that allow for accurate testing of highly rate-dependent experimental substructures. In addition, efficient procedures should be established to accurately update these numerical models. For example, a well adapted Smith Predictor appears to have promising features for use in a model-based integration method. An updated model of experimental substructure can also be used in compensation of delay and correction of feedback forces for actuator tracking errors. More work is required for the development of these models, their adaptation laws and selection of optimum learning gains for use in hybrid simulations of MDF systems.

REFERENCES

1. Mahin SA and Shing PSB. Pseudodynamic method for seismic testing. *Journal of Structural Engineering-ASCE*, 1985; **111**(7): p. 1482-1503.

2. Mahin SA, Shing PSB, Thewalt CR, and Hanson RD. Pseudodynamic test method - current status and future directions. *Journal of Structural Engineering-ASCE*, 1989; **115**(8): p. 2113-2128.

3. Thewalt CR and Mahin SA. Hybrid solution techniques for generalized pseudodynamic testing. *Report UCB/EERC-87/09*. 1987, University of California, Berkeley, CA.

4. Nakashima M. Development, potential, and limitations of real-time online (pseudo-dynamic) testing. *Philosophical Transactions of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences*, 2001; **359**(1786): p. 1851-1867.

5. Shao X, Reinhorn AM, and Sivaselvan M. Real time dynamic hybrid testing using force-based substructuring. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

6. Chen C and Ricles JM. Effective force testing method using virtual mass for real-time earthquake simulation. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

7. Dimig J, Shield C, French C, Bailey F, and Clark A. *Effective Force Testing: A Method of Seismic Simulation for Structural Testing*. 1999, ASCE. p. 1028-1037.

8. Shield C, French C, and Timm J. Development and implementation of the effective force testing method for seismic simulation of large-scale structures. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2001; **359**(1786): p. 1911-1929.

9. Shao X. *Unified control platform for real-time dynamic hybrid simulation (PhD Dissertation)*, in *Department of Civil, Structural, and Environmental Engineering*. 2006, University at Buffalo: Buffalo, NY.

10. Sivaselvan M. A unified view of hybrid seismic simulation algorithms. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

11. Mosqueda G, Stojadinovic B, and Mahin SA. Implementation and accuracy of continuous hybrid simulation with geographically distributed substructures. *Report EERC 2005-02*. 2005, Earthquake Engineering Research Center, University of California, Berkeley, Berkeley, CA.

12. Yang YS, Wang SJ, Wang KJ, Lin ML, Weng YT, Cheng WC, Chang YY, Tsai KC, Lau DT, Hsieh SH, Lin FP, and Lin SY. Network system for a transnational collaborative pseudo-dynamic experiment on a DSCFT-Pier bridge system. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

13. Pan P, Tada M, and Nakashima M. Online hybrid test by internet linkage of distributed test-analysis domains. *Earthquake Engineering & Structural Dynamics*, 2005; **34**(11): p. 1407-1425.

14. Nakashima M, Kaminoso T, Ishida M, and Kazuhiro A. Integration techniques for substructure online test. Proceedings, *4th US National Conference of Earthquake Engineering*. 1990. p. 515-524. Palm Springs, CA: Earthquake Engineering Research Institute.

15. Nakashima M, Kato H, and Takaoka E. Development of Real-Time Pseudo Dynamic Testing. *Earthquake Engineering & Structural Dynamics*, 1992; **21**(1): p. 79-92.

16. Thewalt CR and Roman M. Performance parameters for pseudodynamic tests. *Journal of Structural Engineering -- ASCE*, 1994; **120**(9): p. 2768-2781.

17. Takanashi K, Udagawa K, Seki M, Okada T, and Tanaka H. Nonlinear earthquake response analysis of structures by a computer-actuator on-line system. *Bulletin of Earthquake Resistant Structure Research Center, University of Tokyo*, 1975(8).

18. Takanashi K and Nakashima M. Japanese activities on online testing. *Journal of Engineering Mechanics - ASCE*, 1987; **113**(7): p. 1014-1031.

19. Nakashima M and Masaoka N. Real-time on-line test for MDOF systems. *Earthquake Engineering & Structural Dynamics*, 1999; **28**(4): p. 393-420.

20. Reinhorn AM, Sivaselvan M, Weinreber S, and Shao X. Real-time dynamic hybrid testing of structural systems. Proceedings, *Third European Conference on Structural Control*. 2004. Vienna, Austria.

21. Schellenberg A and Mahin SA. Integration of hybrid simulation within the general-purpose computational framework OpenSEES. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

22. Schellenberg A, Mahin SA, and Fenves GL. A software framework for hybrid simulation of large structural systems. Proceedings, *ASCE Structures Congress*. 2007. Long Beach, CA.

23. Takahashi Y and Fenves GL. Software framework for distributed experimental-computational simulation of structural systems. *Earthquake Engineering & Structural Dynamics*, 2006; **35**(3): p. 267-291.

24. Kwon OS, Nakata N, Elnashai AS, and Spencer BF. Technical Note: A framework for multi-site distributed simulation and application to complex structural systems. *Journal of Earthquake Engineering*, 2005; **9**(5): p. 741-753.

25. McKenna F and Fenves GL. *OpenSees Manual*. 2001, PEER Center, University of California - Berkeley.

26. Williams DM, Williams MS, and Blakeborough A. Numerical modeling of a servohydraulic testing system for structures. *Journal of Engineering Mechanics-ASCE*, 2001; **127**(8): p. 816-827.

27. Jung RY and Shing PB. Performance evaluation of a real-time pseudodynamic test system. 2006; **35**(7): p. 789-810.

28. Shing PB, Stavridis A, Wei Z, Stauffer E, Wallen R, and Jung RY. Validation of a fast hybrid test system with substructure tests. Proceedings, *17th Analysis and Computation Specialty Conference*. 2006. St. Louis.

29. Shing PSB and Jung RY. System dynamics in real-time pseudodynamic testing. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

30. Zhao J, French C, Shield C, and Posbergh T. Considerations for the development of real-time dynamic testing using servo-hydraulic actuation. *Earthquake Engineering & Structural Dynamics*, 2003; **32**(11): p. 1773-1794.

31. Mercan O, Zhang X, and Ricles JM. State space control design for real-time pseudodynamic testing applications. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

32. Guo YR, Xiao Y, Fan YL, Dong XH, and Hu Q. Development and application of a collaborative hybrid dynamic testing software based on NetSLab. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

33. Nakata N, Spencer BF, and Elnashai AS. Mixed load/displacement control strategy for hybrid simulation. Proceedings, *4th International Conference on Earthquake Engineering*. 2006. Taipei, Taiwan.

34. Broyden CG. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, 1965; **19**: p. 577-593.

35. Hanley JP. SCRAMNet DAQ Users Manual. *Report SEESL-2004-02, V1.0.0*. 2004, SEESL/University at Buffalo, Buffalo, NY.

36. Systran / Curtiss Wright Controls Embedded Computing. *Shared Common Random Access Memory Network. http://www.cwcembedded.com/products/0/1/71.html*. 2007.

37. Hanley JP. SCRAMNet DAQ Developers Manual. *Report SEESL-2004-01, V1.0.0*. 2004, SEESL/University at Buffalo, Buffalo, NY.

38. Stojadinovic B, Mosqueda G, and Mahin SA. Event-driven control system for geographically distributed hybrid simulation. *Journal of Structural Engineering-ASCE*, 2006; **132**(1): p. 68-77.

39. Mosqueda G, Stojadinovic B, and Mahin SA. Real-time error monitoring for hybrid simulation. II: structural response modification with error. *Journal of Structural Engineering*, 2007; **133**(8): p. 1109-1117.

40. Mosqueda G, Stojadinovic B, and Mahin SA. Real-time error monitoring for hybrid simulation. I: methodology and experimental verification. *Journal of Structural Engineering*, 2007; **133**(8): p. 1100-1108.

41. Mosqueda G, Stojadinovic B, and Mahin SA. Energy-based procedure for monitoring experimental errors in hybrid simulations. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

42. Filiatrault A, Leger P, and Tinawi R. On the computation of seismic energy in inelastic structures. *Engineering Structures*, 1994; **16**(6): p. 425-436.

43. Filiatrault A and Tinawi R. The use of energy balance in nonlinear seismic analysis. Proceedings, *10th World Conference on Earthquake Engineering*. 1992. p. 4111-4116. Rotterdam: Balkema.

44. Agrawal AK, Fujino Y, and Bhartia BK. Instability due to time-delay and its compensation in active control of structures. *Earthquake Engineering & Structural Dynamics*, 1993; **22**(3): p. 211-224.

45. Qi K and Kuang JS. Time delay compensation in active closed-loop structural control. *Mechanics Research Communications*, 1995; **22**(2): p. 129-135.

46. Agrawal AK and Yang JN. Compensation of time-delay for control of civil engineering structures. *Earthquake Engineering & Structural Dynamics*, 2000; **29**(1): p. 37-62.

47. McGreevy S, Soong TT, and Reinhorn AM. An experimental study of time delay compensation in active structural control. Proceedings, *6th International Modal Analysis Conference*. 1988. p. 733-739. Orlando, FL.

48. Soong TT. *Active structural control: theory and practice*. 1990, New York: Wiley. 194.

49. Yi WH and Peek R. Posterior time-step adjustment in pseudodynamic testing. *Journal of Engineering Mechanics-ASCE*, 1993; **119**(7): p. 1376-1386.

50. Horiuchi T, Inoue M, Konno T, and Namita Y. Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber. *Earthquake Engineering & Structural Dynamics*, 1999; **28**(10): p. 1121-1141.

51. Horiuchi T and Konno T. A new method for compensating actuator delay in real-time hybrid experiments. *Philosophical Transactions of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences*, 2001; **359**: p. 1786-1893.

52. Darby AP, Blakeborough A, and Williams MS. Real-time substructure tests using hydraulic actuator. *Journal of Engineering Mechanics-ASCE*, 1999; **125**(10): p. 1133-1139.

53. Darby AP, Blakeborough A, and Williams MS. Improved control algorithm for real-time substructure testing. *Earthquake Engineering & Structural Dynamics*, 2001; **30**(3): p. 431-448.

54. Darby AP, Williams MS, and Blakeborough A. Stability and delay compensation for real-time substructure testing. *Journal of Engineering Mechanics - ASCE*, 2002; **128**(12): p. 1276-1284.

55. Carrion JE and Spencer BF. *Real-time hybrid testing using model-based delay compensation*, in *4th International Conference on Earthquake Engineering*. 2006: Taipei, Taiwan.

56. Magonette G. Development and application of large-scale continuous pseudo-dynamic testing techniques. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2001; **359**(1786): p. 1771-1799.

57. Bonnet PA, Lim CN, Williams MS, Blakeborough A, Neild SA, Stoten DP, and Taylor CA. Real-time hybrid experiments with Newmark integration, MCSmd outer-loop control and multi-tasking strategies. *Earthquake Engineering & Structural Dynamics*, 2007; **36**(1): p. 119-141.

58. Shing PB and Mahin SA. Pseudodynamic test method for seismic performance evaluation. *Report UCB/EERC-84/01*. 1984, Earthquake Engineering Research Center, University of California, Berkeley, Berkeley, CA.

59. Hilber HM, Hughes TJR, and Taylor RL. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 1977; **5**: p. 283-292.

60. Bayer V, Dorka UE, Fullekrug U, and Gschwilm J. On real-time pseudo-dynamic sub-structure testing: algorithm, numerical and experimental results. *Aerospace Science and Technology*, 2005; **9**(3): p. 223-232.

61. Shing PSB, Vannan MT, and Cater E. Implicit time integration for pseudodynamic tests. *Earthquake Engineering & Structural Dynamics*, 1991; **20**(6): p. 551-576.

62. Jung RY, Shing PSB, Stauffer E, and Thoen BK. Performance of a real-time pseudodynamic test system considering nonlinear structural response. *Earthquake Engineering & Structural Dynamics*, 2007; **36**(2): p. 1785-1809.

63. Wu B, Wang Q, Shing PB, and Ou J. Equivalent force control method for generalized real-time substructure testing with implicit integration. *Earthquake Engineering & Structural Dynamics*, 2007; **36**(9): p. 1127-1149.

64. Ghaboussi J, Yun GJ, and Hashash YMA. A novel predictor-corrector algorithm for sub-structure pseudo-dynamic testing. *Earthquake Engineering & Structural Dynamics*, 2006; **35**(4): p. 453-476.

65. Schneider SP and Roeder CW. An inelastic substructure technique for the pseudodynamic test method. *Earthquake Engineering & Structural Dynamics*, 1994; **23**: p. 761-775.

66. Chang SY. Application of the momentum equations of motion to pseudo-dynamic testing. *Philosophical Transactions of The Royal Society of London Series A-Mathematical Physical And Engineering Sciences*, 2001; **359**(1786): p. 1801-1827.

67. Chang SY. Improved numerical dissipation for explicit methods in pseudodynamic tests. *Earthquake Engineering & Structural Dynamics*, 1997; **26**(9): p. 917-929.

68. Chang SY. An improved on-line dynamic testing method. *Engineering Structures*, 2002; **24**(5): p. 587-596.

69. Chang SY. Explicit pseudodynamic algorithm with unconditional stability. *Journal of Engineering Mechanics-ASCE*, 2002; **128**(9): p. 935-947.

70. Chang SY and Sung YC. An enhanced explicit pseudodynamic algorithm with unconditional stability. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

71. Chang SY, Tsai KC, and Chen KC. Improved time integration for pseudodynamic tests. *Earthquake Engineering & Structural Dynamics*, 1998; **27**(7): p. 711-730.

72. Combescure D and Pegon P. Alpha-operator splitting time integration technique for pseudodynamic testing - error propagation analysis. *Soil Dynamics and Earthquake Engineering*, 1997; **16**(7-8): p. 427-443.

73. Zhang YF, Sause R, Ricles JM, and Naito CJ. Modified predictor-corrector numerical scheme for real-time pseudo dynamic tests using state-space formulation. *Earthquake Engineering & Structural Dynamics*, 2005; **34**(3): p. 271-288.

74. Wu B, Xu G, Wang Q, and Williams MS. Operator-splitting method for real-time substructure testing. *Earthquake Engineering & Structural Dynamics*, 2006; **35**(3): p. 293-314.

75. Wu B, Wang Q, and Ou J. Stability analysis of operator-splitting method for real-time substructure testing with actuator delay and compensation. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

76. Fahnestock LA, Ricles JM, and Sause R. Experimental study of a large-scale buckling-restrained braced frame using the pseudo-dynamic testing method. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

77. Elkhoraibi T and Mosalam KM. Pseudo-dynamic experiment on one-story RC structure with and without masonry infill. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

78. Lin ML, Tsai KC, and Tsai CY. Bi-directional sub-structural pseudo-dynamic tests of a full-scale 2-story BRBF, Part 2: Compressive behavior of Gusset Plates. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

79. Tsai KC, Weng YT, Wang KJ, Tsai CY, and Lai JW. Bi-directional sub-structural pseudo-dynamic testing of a full-scale 2-story BRBF, Part 1: Seismic design, analytical and experimental performance assessments. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

80. Mercan O and Ricles JM. Implementation of real-time hybrid pseudodynamic test method for evaluating seismic hazard mitigation measures. Proceedings, *ASCE Structures Congress*. 2007. Long Beach, CA.

81. Yang TY, Stojadinovic B, and Moehle J. Hybrid simulation evaluation of innovative steel braced framing system. Proceedings, *8th National Conference on Earthquake Engineering*. 2006. San Francisco, CA.

82. The MathWorks® Inc. *Simulink® Reference*. 1994-2007: Natick, MA.

83. Microsoft Corp. *Windows Operating System*. 1985-2007: Redmond, Washington.

84. MTS Systems Corporation. *SUNY Buffalo real time hybrid structural test system user's manual*. 2004: Buffalo, NY.

85. The MathWorks® Inc. *MATLAB User's Guide*. 1994-2007: Natick, MA.

86. The MathWorks® Inc. *Real-Time Workshop User's Guide*. 1994-2007: Natick, MA.

87. MTS Systems Corporation. *Computer simulation models of actuators*. 2004: Eden Prairie, MN.

88. Sivaselvan M and Reinhorn AM. Hysteretic models for deteriorating inelastic structures. *Journal of Engineering Mechanics*, 2000; **126**(6): p. 633-640.

89. Bouc R. Forced vibration of mechanical systems with hysteresis. Proceedings, *4th Conference on Nonlinear Oscillations*. 1967. Prague, Czechoslovakia.

90. Wen Y. Method for random vibration of hysteretic systems. *Journal of Engineering Mechanics*, 1976; **102**(2): p. 249-263.

91. Nachtigal CL. *Instrumentation and Control: Fundamentals and Applications*. 1990: Wiley Interscience.

92. Moog Inc. *Moog Servovalves*. 1958-2007: East Aurora, NY.

93. Merritt HE. *Hydraulic control systems*. 1967, New York: John Wiley and Sons Ltd.

94. Stringer J. *Hydraulic systems analysis: an introduction*. 1976, New York: John Wiley and Sons Ltd.

95. Nikiforuk PN, Ukrainetz PR, and Tsai SC. Detailed analysis of a two-stage four-way electrohydraulic flow-control valve. *Journal of Mechanical Engineering Science - London*, 1969; **11**(2): p. 169-174.

96. Wang D, Dolid R, Donath M, and Albright J. Development and verification of a two-stage flow control of a servovalve model. Proceedings, *ASME - Fluid Power Systems and Technology Division*. 1995. p. 121-129.

97. Conte JP and Trombetti TL. Linear dynamic modeling of a uni-axial servo-hydraulic shaking table system. *Earthquake Engineering & Structural Dynamics*, 2000; **29**(9): p. 1375-1404.

98. Ziegler JG and Nichols NB. Optimum settings for automatic controllers. *Intech*, 1995; **42**(6): p. 94-100.

99. Shing PB and Mahin SA. Experimental error propagation in pseudodynamic testing. *Report UCB/EERC-83/12*. 1983, Earthquake Engineering Research Center, University of California, Berkeley, CA.

100. Ogata K. *Modern Control Engineering*. 4 ed. 1997, Upper Saddle River, NJ: Prentice Hall.

101. Agrawal AK and Yang JN. Effect of fixed time delay on stability and performance of actively controlled civil engineering structures. *Earthquake Engineering & Structural Dynamics*, 1997; **26**(11): p. 1169-1185.

102. Chu SY, Soong TT, and Reinhorn AM. *Active, Hybrid, Semi-Active Structural Control - A Design and Implementation Handbook*. 2005: John Wiley and Sons.

103. Rodellar J, Barbat AH, and Martin Sanchez JM. Predictive control of structures. *Journal Of Engineering Mechanics - ASCE*, 1987; **113**(6): p. 797-812.

104. Chung LL, Lin CC, and Lu KH. Time-delay control of structures. *Earthquake Engineering & Structural Dynamics*, 1995; **24**: p. 687-701.

105. Agrawal AK and Yang JN. Compensation of time-delay for active control of engineering structures. Proceedings, *12th Engineering Mechanics Conference*. 1998. p. 225-228. La Jolla, CA.

106. Abdel-Rohman M. Structural control considering time delay effect. *Transactions of Canadian Society for Mechanical Engineering*, 1985; **9**(4): p. 224-227.

107. Sain PM, Spencer BF, Sain MK, and Suhardjo J. Structural control design in the presence of time-delays. Proceedings, *Ninth Engineering Mechanical Conference - ASCE*. 1992. p. 812-815. College Station, TX.

108. Kumar R. Effects of time delays on the stability of collocated and noncollocated point control of discrete dynamic structural systems. *Applied Mathematics and Computation*, 1995; **70**(2-3): p. 215-232.

109. Udwadia FE, von Bremen H, and Phohomsiri P. Time-delayed control design for active control of structures: principles and applications. *Structural Control and Health Monitoring*, 2007; **14**(1): p. 27-61.

110. Smith OJM. Close control of loops with dead time. *Transactions of Chemical Engineering Progress*, 1957; **53**: p. 217-219.

111. Smith OJM. A Controller to overcome dead time. *ISA Transactions*, 1959; **6**(2): p. 28-33.

112. Abdel-Mooty M and Roorda J. Time delay compensation in active damping of structures. *Journal of Engineering Mechanics*, 1991; **117**(11): p. 2549-2570.

113. Chung LL, Lin RC, Soong TT, and Reinhorn AM. Experimental study of active control for MDOF seismic structures. *Journal of Engineering Mechanics*, 1989; **115**(8): p. 1609-1627.

114. Reinhorn AM, Soong TT, and Lin CC. 1:4 scale model studies of active tendon systems and active mass damper for aseismic protection. *Report NCEER-89-0026*. 1989, NCEER, Buffalo, NY.

115. Shao X, Reinhorn AM, and Sivaselvan M. Force controlled actuators in hybrid testing. Proceedings, *4th World Conference on Structural Control and Monitoring*. 2006. San Diego, CA.

116. Hang CC, Lim KW, and Chong BW. A dual-rate adaptive digital Smith Predictor. *Automatica*, 1988; **25**(1): p. 1-16.

117. Ahmadizadeh M, Mosqueda G, and Reinhorn AM. Compensation of actuator delay and dynamics for real-time hybrid structural simulation. Proceedings, *4th World Conference on Structural Control and Health Monitoring*. 2006. San Diego, CA.

118. Newmark NM. A method of computation for structural dynamics. *Journal of Engineering Mechanics*, 1959; **85**: p. 67-94.

119. Chopra AK. *Dynamics of Structures - Theory and Applications to Earthquake Engineering*. 2 ed. 2001, Upper Saddle River, NJ: Prentice-Hall, Inc.

120. Hughes TJR. *The Finite Element Method - Linear Static and Dynamic Finite Element Analysis*. 2000, Mineola, NY: Dover Publications.

121. Bathe KJ. *Finite Element Procedures*. 2 ed. 1995: Prentice Hall.

122. Utkin V. Variable structure systems with sliding modes. *IEEE Transactions on Automatic Control*, 1977; **22**(2): p. 212-222.

123. Igarashi A, Seible F, and Hegemeier GA. Development of pseudo-dynamic technique for testing a full scale 5-story shear wall structure. Proceedings, *Japan Seminar on the Development and Future Directions of Structural Testing Techniques*. 1993. Japan.

124. McGuire W, Gallagher RH, and Ziemian RD. *Matrix Structural Analysis*. 2 ed. 2000, New York: John Wiley and Sons, Inc.

125. Greenberg MD. *Advanced Engineering Mathematics*. 2 ed. 1998: Prentice Hall.

126. Schachter M and Reinhorn AM. Analytical modeling of zipper frames subjected to shaking table testing. Proceedings, *ASCE Structures Congress*. 2007. Long Beach, CA.