NSEL Report Series
Report No. NSEL-049
September 2018

Development and Implementation of a Multi-axial Real-time Hybrid Simulation Framework

G.A. Fermandois
and
B.F. Spencer, Jr.

Department of Civil and Environmental Engineering
University of Illinois at Urbana-Champaign

UILU-ENG-2018-1801

ISSN: 1940-9826

The Newmark Structural Engineering Laboratory (NSEL) of the Department of Civil and Environmental Engineering at the University of Illinois at Urbana-Champaign has a long history of excellence in research and education that has contributed greatly to the state-of-the-art in civil engineering. Completed in 1967 and extended in 1971, the structural testing area of the laboratory has a versatile strong-floor/wall and a three-story clear height that can be used to carry out a wide range of tests of building materials, models, and structural systems. The laboratory is named for Dr. Nathan M. Newmark, an internationally known educator and engineer, who was the Head of the Department of Civil Engineering at the University of Illinois [1956-73] and the Chair of the Digital Computing Laboratory [1947-57]. He developed simple, yet powerful and widely used, methods for analyzing complex structures and assemblages subjected to a variety of static, dynamic, blast, and earthquake loadings. Dr. Newmark received numerous honors and awards for his achievements, including the prestigious National Medal of Science awarded in 1968 by President Lyndon B. Johnson. He was also one of the founding members of the National Academy of Engineering.

Contact:
Prof. B.F. Spencer, Jr.
Director, Newmark Structural Engineering Laboratory
2213 NCEL, MC-250
205 North Mathews Ave.
Urbana, IL 61801
Telephone (217) 333-8630
E-mail: bfs@illinois.edu

This technical report is based on the first author's doctoral dissertation of the same title, which was completed in May 2018. The second author served as the dissertation advisor for this work.

Financial support for this research was provided in part by the following funding agencies: "Comisión Nacional de Investigación, Ciencia y Tecnología" (CONICYT – Chile) under Becas Chile Scholarship No. 72140204; the Fulbright Foreign Student Fellowship under IIE Grant No. 15130588; the Universidad Técnica Federico Santa María under Faculty Development Fellowship Grant No. 208-13; and funding from the Nathan M. and Anne M. Newmark Endowed Chair in Civil Engineering.

The cover photographs are used with permission. The Trans-Alaska Pipeline photograph was provided by Terra Galleria Photography (http://www.terragalleria.com/).

ABSTRACT

Real-time hybrid simulation is an efficient and cost-effective experimental testing technique for performance evaluation of structural systems subjected to earthquake loading with rate-of-loading behavior. To assess the response of structural components with multi-axial loading, a loading assembly with multiple parallel actuators connected to a rigid moving platform is required to impose realistic boundary conditions on physical components. This loading assembly is expected to exhibit significant dynamic actuator coupling and suffer from systematic errors and potential instabilities. One approach to reduce experimental errors considers a multi-input, multi-output (MIMO) modeling approach to design controllers that could compensate for these undesired effects.

In this study, a framework for three-dimensional, multi-axial real-time hybrid simulation is presented. The methodology consists in designing a real-time system platform to perform dynamic test experiments by controlling the interface boundary conditions on the physical specimen in Cartesian (global) coordinates. First, a kinematic transformation is derived to impose the six-degree-of-freedom motion to the loading platform in three-dimensional Cartesian space. Then, a linearized model of the multi-actuator loading assembly is obtained through nonparametric frequency domain system identification techniques. Subsequently, a feedforward-feedback compensator is developed for reference tracking of the multivariate transient signals, which should be sufficiently robust to rule out any disturbances and measurement noises in the experimental closed-loop system. Finally, the numerical substructure, compensators, and kinematic transformations are implemented over an embedded system with a micro-controller unit and digital signal processing capabilities for real-time applications.

The proposed framework is validated using a small-scale version of the Load and Boundary Condition Box (LBCB) from Newmark Civil Engineering Laboratory at University of Illinois, Urbana-Champaign. A one-story, two-bay, moment frame was considered as the reference structure, where the experimental substructure was chosen as a steel column with fixed ends. The hybrid system was subjected to earthquake ground motions chosen according to its importance and destructive characteristics. Comparisons of different compensation strategies are made, and excellent performance is achieved for all situations that incorporates the multivariate controller.

CONTENTS

Chapter 1

INTRODUCTION

1.1 Motivation

Natural and human-made disasters have a dramatic impact on the development of our society. According to the International Civil Defense Organisation (ICDO, 2016), the total economic and human impact of natural disasters in the period between 2000 and 2011 has been estimated to be around USD $1.3 trillions of damage, with 2.7 billion citizens been directly affected by the disaster, and with a death toll of 1.1 millions. If only the de-aggregated statistics per natural hazard are considered, earthquakes are the most deadly natural hazard with more than 750 thousand people killed in this period; moreover, earthquakes have the second highest economic impact (USD $636 billion in damage), right after storm hazards (USD $720 billion in damage).

Indeed, understanding the risks of natural hazards and promoting mitigation plans is fundamental for our society to achieve sustained development. Therefore, one of the main objectives in urban planning is to support the growth of sustainable and resilient cities, and civil engineers play a very important role in this regard. Civil engineers are entrusted to plan, design, build, and maintain the civil infrastructure that serves as the engine of our economic development. Also, civil engineers collaborate in multi-disciplinary teams to enable risk management of lifelines (e.g. highways, railroads, water and electrical supply), critical facilities (e.g. hospitals, governmental and defense buildings), and residential property.

Moreover, structural engineers are commissioned to assess the performance and reliability of structural systems, which is a fundamental ingredient for any risk management and mitigation plan. Structural performance assessment can be accomplished by two means: (i) numerical simulation, where the structural responses are predicted using numerical models; and (ii) physical observations, where structural systems and/or components are subjected to loading conditions, and the physical responses are measured directly to evaluate the structural performance. The latter is usually conducted through either field inspection after an extreme event (e.g., large earthquakes) or by experimental testing in a laboratory environment.

Numerical simulation has experienced significant improvements over the years, with faster computers capable of running multiple simulations of very large structural systems with complex loading scenarios. However, experimental testing is still necessary. For example, most computational models and constitutive relationships are developed and evaluated

on the basis of experimental testing results. Furthermore, when the response characteristics of a structural system are not well understood or difficult to model numerically (e.g., inelasticity, nonlinear effects, and rate-dependent behavior), physical testing provides the only accurate way to study the performance of structural systems.

Therefore, experimental testing is considered an essential tool in civil engineering, to understand the behavior of structural systems and construction materials, which provides empirical evidence that can be used to calibrate numerical models for reliable, cost-effective analysis and design, and eventually be considered for the development of building codes used by practitioners. However, experimental testing has its limitations. More often than not, experimental tests are constrained by laboratory space, equipment payloads and size, and more importantly, by project budget and timeframes.

Different experimental techniques are currently available, which have been extensively used in the field of earthquake engineering. Most of these techniques can be classified into three main methods: (i) cyclic (static) testing; (ii) shaking table (dynamic) testing; and (iii) hybrid simulation. In particular, shaking table testing is regarded as the most realistic approach to conduct experiments on structural systems, due to the fact that it can reproduce similar conditions associated with large earthquake events. Therefore, the results obtained from shaking table testing are considered to be more representative of the physical phenomena compared to cyclic testing, because it incorporates dynamic behavior on the structural response. But, due to high costs and complexity of the loading systems, shaking table tests of large-scale buildings and bridges is extremely challenging. As an alternative, reduced-scale specimens can be considered to study the global behavior of real structures. Unfortunately, local effects of construction materials, such as fracture or local buckling, do not scale well with size. Hence, the results from small-scale shake table testing may not be an adequate representation of the physical phenomena.

Thus, hybrid simulation has proven to be a reliable, flexible and practical alternative to shake table testing. Hybrid simulation is a *hybrid* procedure that integrates experimental testing with online computer simulation, for cost-effective performance evaluation of structural systems and components when subjected to extreme loading events, such as large and destructive earthquakes. This approach allows for experimental testing of only critical components of interest (i.e. where damage is expected) from the reference structural system; while other, better-understood portions of the structure can be modeled numerically in the computer. Both experimental and numerical components of the hybrid system are interconnected by actuators, sensors, and digital signal processing hardware, forming a feedback loop to solve the governing equations of motion at every integration time step.

Hybrid simulation was originally proposed in the late 1970s, and from the mid-1980s it gained much attention from the engineering community. Many developments in hybrid simulation testing have been successfully proposed in the last few decades, each of which focus on particular aspects of the experimental, computational, and networking capabilities of this approach. One of the most recent applications from this scope is real-time hybrid simulation (RTHS). This technique was proposed for dynamic testing purposes, especially for structural systems with rate-dependent behavior. During the last two decades, RTHS have demonstrated many advances and improvements. Still, research on RTHS testing has not reached its mature stage yet, and many challenges are yet to be solved in different technical

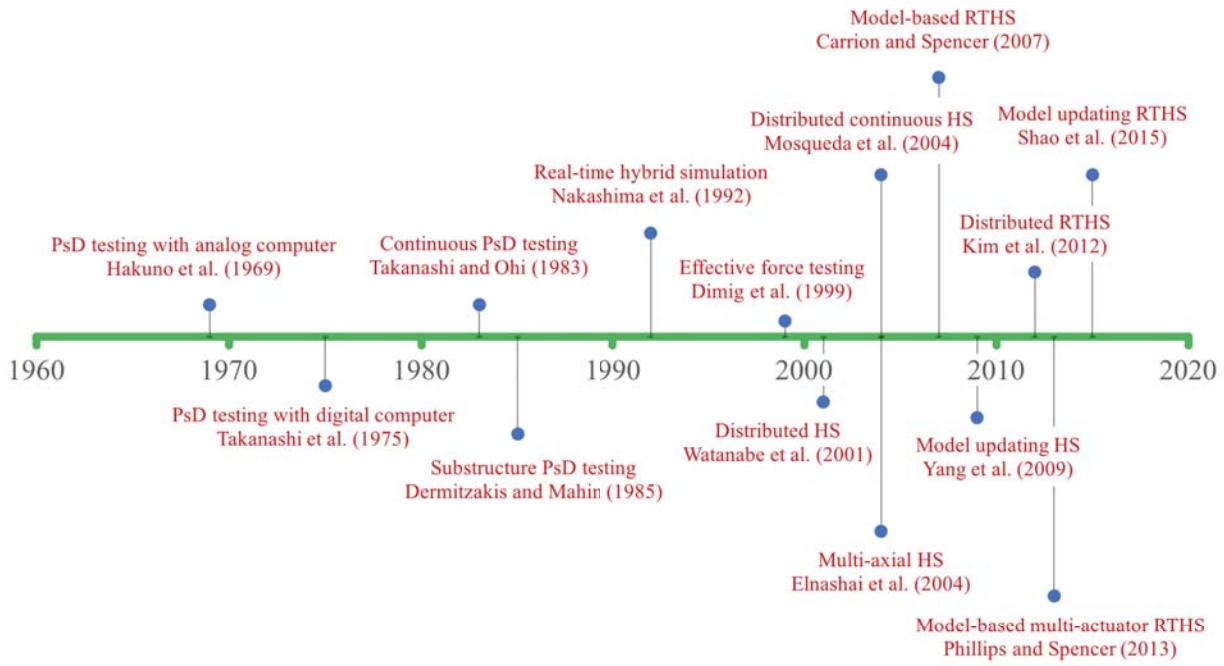


Figure 1.1: Major milestones in the development of hybrid simulation techniques

areas.

1.2 Hybrid simulation: a historical perspective

Hybrid simulation (HS) testing, also called *pseudo-dynamic (PsD) testing*, is an experimental testing method that has proven to be an attractive alternative to traditional cyclic (quasi-static) and shaking table testing. The original concept was proposed in the late 1960s by Hakuno et al. (1969), where a single-degree-of-freedom system under ground motion excitation was studied. In this seminal work, the test specimen was loaded by an electromagnetic actuator while using an analog computer to perform "online" numerical integration of the equations of motion. Later, with the advent of modern computers, Takanashi et al. (1975) was the first to incorporate a digital computer to control the experimental equipment for online testing. Moreover, the first implementation of substructuring techniques along with pseudo-dynamic testing was presented by Dermitzakis and Mahin (1985). These three seminal contributions, along with incorporation of the microprocessor and advanced electro-servo controllers, opened the door for the extensive use of what is commonly known these days as *hybrid simulation (HS)* testing in experimental research. Major milestones on the development of hybrid simulation are presented in Figure 1.1.

The goal of hybrid simulation is to study the responses of a reference structural system by combining numerical simulations and experimental observations in a test setup. The method consists on the identification of critical components of interest from a reference

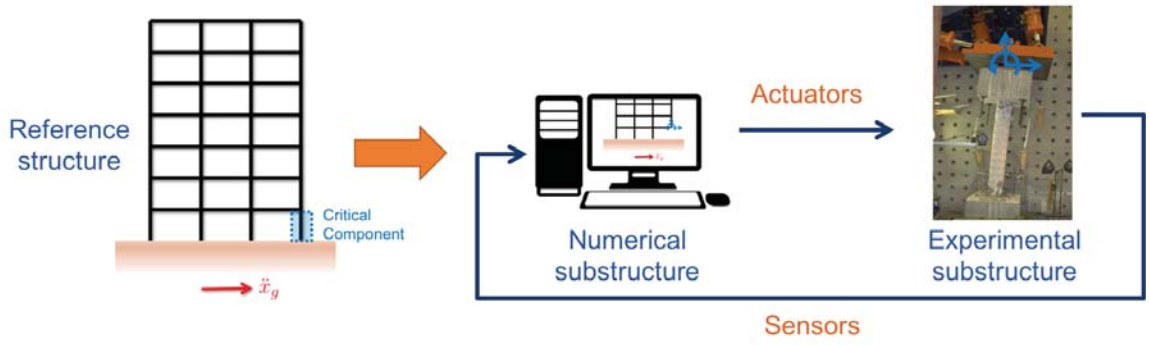


Figure 1.2: Substructuring in hybrid simulation testing

structural system (e.g., where damage is expected). These components are then isolated from the reference system, thus creating two separate substructures, as shown in Figure 1.2: (i) a *experimental substructure*, where test specimens are built and installed in a experimental facility; and (ii) a *numerical substructure*, where all the structural elements are modeled numerically for simulation. To solve the governing equations of motion from the reference system, both experimental and numerical substructures are interconnected with a series of actuators and sensors, to enforce equilibrium and compatibility at the designated interfaces between substructures (Mahin and Shing, 1985).

At the early stages of development of hybrid simulation, two major challenges were identified. The first one is associated to experimental error propagation, while the second concerns the real-time constraints of loading and computational hardware. About the former, it was observed that experimental errors can affect the reliability of the hybrid simulation, due to differences between target and measured displacements of the actuator, or by incorrect force measurements from the test specimen. Because hybrid simulation is a closed loop feedback system, any experimental error can be introduced and accumulate with time, thus decreasing the accuracy of the results and jeopardizing the stability of the experimental setup. This effect is exacerbated when actuators are used to simulate dynamic loads. On the other hand, the speed for solving the equations of motion of the numerical substructure is highly dependent on the model size and the computational resources. More often than not, computers require to perform several algebraic operations and run iterations to obtain an accurate numerical solution. Hence, time restrictions occur, because both computer and loading assembly must be perfectly synchronized to perform the test.

Thus, the loading over the specimen is usually applied very slowly in conventional hybrid simulation, by means of a *ramp-hold* procedure. In other words, the test specimen system is loaded with an extended time scale. Regardless, this ramp-hold procedure have a major problem that results on loss of accuracy and representativeness of the results. During the hold stage, the specimen may experience force relaxation which introduces unwanted errors in the experimental test. Hence, researchers were looking for alternatives to move the actuators in a smooth manner, without hold phases.

Later, Takanashi and Ohi (1983) developed a "fast" (continuous) hybrid simulation technique, which introduced for the first time a dynamic actuator and improved servo controller. In this method, the command displacements are applied continuously to the test

specimen; thus, the actuators move without stopping until the simulation ends. But, to keep the actuators moving continuously, the responses from the numerical substructure must be predicted while the computer is busy solving the next time integration step. Hence, the *predictor-corrector method* was introduced, which is the key component to allow for continuous HS. The technique was validated using a single-degree-of-freedom system with a steel beam as test specimen (Takanashi and Nakashima, 1987). From the tests, it was observed that the continuous HS behaved well, while no significant differences were observed in the overall responses for quasi-static and "fast" tests. Still, this method was not sufficient to conduct dynamic testing in real-time, primarily because of inaccurate control and error propagation during the test.

For dynamic testing purposes, *real-time hybrid simulation (RTHS)* has demonstrated many advances and improvements in the past 20 years. Testing is executed in real-time, meaning that all calculations, imposed boundary conditions on physical specimens, measured forces, and displacements, and digital data acquisition, must be performed in very short time intervals, typically less than 1 millisecond (Nakashima, 2001). Moreover, the boundary conditions must also be imposed at fast rates, meaning that dynamic actuators are required for this task. Therefore, the fundamental requirements to perform RTHS tests is the implementation of both fast hardware and software to achieve a stable and accurate result. The literature offers many comparative studies between RTHS and conventional shaking table testing, for both steel specimens (Lamarche et al., 2010) and concrete specimens (Chae et al., 2017; Saouma et al., 2014), with very good agreement between the two testing methods. Moreover, RTHS studies have been carried out for a number of structural systems with rate-dependent components, such as sliding bearing devices (Nakashima, 2001); passive energy dissipation devices (Chae et al., 2013b; Horiuchi et al., 1999; Nakashima et al., 1992); and semi-active control devices (Asai et al., 2013; Brodersen et al., 2016; Carrion et al., 2009; Chae et al., 2014; Jiang et al., 2013; Phillips et al., 2010). These results confirm that RTHS is an accurate, cost-effective, flexible, and repeatable alternative to conventional shake table testing. Nonetheless, RTHS testing has not yet reached maturity; many challenges remain to be resolved in the areas of servo-hydraulic dynamic compensation, control design, digital communications, and fast numerical computations, among others.

A critical aspect of RTHS testing is that any experimental errors must be mitigated during real-time execution to avoid inaccurate and unstable responses (Darby et al., 2002; Horiuchi et al., 1996). Thus, the engineering community has greatly focused on the development of compensation techniques in RTHS, with sufficient performance and robustness guarantees, such that RTHS experiments are conducted safely. In particular, Carrion and Spencer (2007) was the first study to consider a model-based approach to the design of compensators to run RTHS experiments with sufficient guarantees of stability and accuracy. This approach considered feedforward and feedback controllers that were designed according to a model of the experimental system, such that it minimizes the tracking errors while conducting the dynamic experiments in real time. Thereafter, many researchers focused on the improvement of this concept to include nonlinear phenomena and model uncertainty (Gao et al., 2013; Ou et al., 2015a; Phillips and Spencer, Jr., 2012).

Furthermore, recent RTHS studies are looking for alternatives to reduce the burden over micro-controller units when executing the integration time-stepping algorithms of large

numerical substructures in real-time. For example, Kim et al. (2011a) proposed a convolution integral method that can reduce the numerical computations during real-time execution by means of a determining a pre-calculated response of the bare numerical substructure without specimen interaction. While, Maghareh et al. (2016b) proposed an adaptive multi-rate interface for two machines working at different sampling rates: (i) a computer working at a slow sampling rate for numerical integration of a high fidelity numerical substructure; and (ii) a micro-controller unit working at a higher sampling rate, where the control strategies are executed.

On the other hand, many researchers have been interested in alternatives of hybrid simulation where specimens are tested under force control, rather than displacement control, specially for very stiff structures (Thewalt and Mahin, 1987). The *effective force testing (EFT)* was proposed as a dynamic force-controlled method. Dimig et al. (1999) presented the first experimental implementation of this approach. In this method, the effective (inertial) force, which is a function of the ground acceleration and mass of the structure, is applied directly to the test specimen. Thus, the experimental loading pattern can now be force-controlled. This method was attractive at first, because the effective force loading was known "a priori" for any given earthquake ground acceleration record. But, it was found that hydraulic actuators were incapable of applying forces at the natural frequency of lightly damped structures, due to a phenomenon called *control-structure interaction* (Dyke et al., 1995). Moreover, because experimental force measurement is usually contaminated with high noise levels, stable and robust force feedback control must be considered (Nakata, 2013; Nakata et al., 2014). Although, the EFT method does not allow for substructuring, there has been some contributions on force-control RTHS testing (Shao and Reinhorn, 2012; Sivaselvan et al., 2008).

For three-dimensional loading of full-scale test specimens, *multi-axial HS* was proposed by Elnashai et al. (2004), as an alternative to six-degree-of-freedom shaking table tests. For this purposes, multi-actuator loading assemblies were introduced to impose six-degree-of-freedom (6DOF) Cartesian boundary conditions over the test specimen: three translations and three rotations, all of them controlled at one point in space. A concise review of this method is provided in the next Section 1.3.

Subsequently, *geographically-distributed HS* was inspired by the fundamental concept of substructuring, and benefited from modern technological advances in digital communications. Conceptually, the reference structure is partitioned in several substructures, and each substructure do not need to be located in the same laboratory facility. Thus, each distributed substructure can be linked through computer networking, such as the Internet. This method can substantially boost national and international collaboration between experimental sites with larger loading assembly capacities and computational sites with powerful, super-computer clusters. Other advances have been proposed in the area of geographically-distributed HS, such as distributed continuous HS (Mosqueda et al., 2004), and distributed RTHS (Kim et al., 2012).

Other important developments in hybrid simulation have gained recent attention of the scientific community. In particular, *model-updating hybrid simulation* was proposed by Yang et al. (2009), where the numerical substructure is modified during the simulation to incorporate knowledge from the test specimen. The reason for the modifications in struc-

tural parameters of the numerical substructure is associated with the level of knowledge of structural behavior of some of its components. If subjective judgements were made in the numerical modeling of a component, and it shares common characteristics with the test specimen, the structural properties of the numerical component are "updated" by using the online observations from the real physical component. For this purpose, an optimization problem is usually stated, where a set of optimal structural parameters are sought such that an objective function is minimized. This objective function is usually a residual between measured and predicted responses from the structural component of interest. The model-updating approach for HS testing have been studied by Hashemi et al. (2014) and Elanwar and Elnashai (2016a,b), among others. Also, the model-updating approach was extended for RTHS testing, where nonlinear Kalman filters have been used for online parameter estimation (Ou et al., 2017; Shao et al., 2015).

1.3 Multi-axial hybrid simulation testing

As stated previously, the purpose of experimental testing is to evaluate the performance of structural systems for extreme environmental loading. A well known fact is that the accuracy achieved in performance evaluation can significantly affect the design process and/or risk assessment. Moreover, if reliable measures of structural performance from seismic loading are required, then a more realistic loading scenario is needed. Then, the effects of multi-directional ground motions over structural systems are required in both numerical simulations and experimental testing.

Different solutions are available for three-dimensional loading in experimental testing. As discussed previously, a six-degree-of-freedom shake table can be considered for this purpose, but the cost and time required to conduct large-scale testing makes it very difficult to implement. Thus, Elnashai et al. (2004) proposed a multi-axial hybrid simulation framework, based on state-of-the-art hydraulic simulators that could impose six-degree-of-freedom load and boundary conditions to a physical specimen for seismic performance evaluation.

In essence, multi-axial HS testing requires a multi-actuator loading assembly to impose the three-dimensional loads over the test specimen. This type of equipment is very unique, and only three facilities in the world have introduced large-scale multi-actuator systems for three-dimensional multi-axial loading, as shown in Figure 1.3. Each of these three facilities has the capability to conduct multi-axial hybrid simulation (HS) testing.

First, the Multi-Axial Subassemblage Testing (MAST) system was first commissioned by the University of Minnesota (USA). Basically, the MAST system consists of eight coupled hydraulic actuators for the application of multi-directional loading as shown in Figure 1.3a, and it was first proposed by French et al. (2004) mainly for cyclic (quasi-static) testing (e.g., Nojavan et al., 2015). More recently, the Smart Structures Laboratory from Swinborne University of Technology (Australia) have commissioned a similar large-scale MAST system, which is capable of conducting both cyclic and HS testing (Hashemi et al., 2015). A picture of the MAST system installed at Swinborne is shown in Figure 1.3b.

In 2003, the Newmark Civil Engineering Laboratory from the University of Illinois at Urbana-Champaign (USA) commissioned three large-scale multi-actuator loading assem-

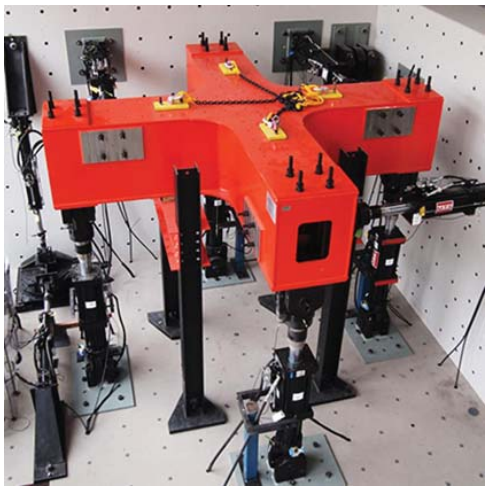
blies, called the *Load and Boundary Condition Box (LBCB)*. Each LBCB consists of six hydraulic actuators mounted to a boxed frame, where each moving piston is connected in parallel configuration to a rigid loading platform for controlled three-dimensional rigid body motion, as shown in Figure 1.3c. In particular, the large-scale LBCBs at Illinois are a modular solution for experimental testing that could be mounted to either strong walls or strong floor, something that is not possible with the MAST system. The experimental facility at Illinois allows for up to three experimental substructures to be tested simultaneously in cyclic (quasi-static) or hybrid simulation. Because of its modularity,the LBCBs can also be used to impose multiple boundary conditions at different positions of a single test specimen. This feature is very attractive for sophisticated substructuring (e.g., three numerical substructures interconnected through a single experimental substructure). Also, modularity allows for the use of two LBCBs working in parallel to test very stiff physical specimens, where a single LBCB is not enough to impose the required loads for testing. In addition, three small-scale versions are available at the same facility, primarily intended for training and academic purposes.

Nakata et al. (2007) proposed a systematic procedure for calibration and control of the multi-actuator loading assembly. The main contributions of that study were the development of a sensitivity-based external calibration method for multi-actuator systems controlled in global Cartesian coordinates, and a mixed-mode (force and displacement) coupled control strategy for multi-axial HS testing. The latter was very important to account for the coupling between vertical actuators responsible of creating the gravity loads, with the lateral actuators that primarily impose displacements to the test specimen.

Subsequently, researchers adopted this framework to conduct novel three-dimensional hybrid simulation testing of civil infrastructure. Kim et al. (2011b) examined the effect of coupled horizontal and vertical earthquake ground motion on a series of large-scale reinforced concrete bridge piers. In addition, an external feedback control system was developed in order to overcome control issues of the LBCB when interacts with test specimens with high axial stiffness. Mahmoud et al. (2013) investigated the system-level performance of steel moment frames with semi-rigid connections. The test specimen consisted on a full-scale beam-column subassembly, that was loaded by the combination of two LBCBs to impose the required displacements and boundary conditions at two different ends of the subassemblage. Frankie et al. (2013) performed a three-dimensional seismic test of a curved four-span bridge system. The test specimen was a full-scale reinforced concrete bridge pier, and the numerical substructure was comprised of the bridge deck, cap beams and abutments. Afterwards, rigorous model calibration was performed using the experimental results in order to improve the accuracy of analytical predictions from the bridge numerical model (Abdelnaby et al., 2014). Chang et al. (2015) proposed a high-precision external positioning correction method for multi-axial hybrid simulation systems, that could account for any unanticipated displacements in the test setup due to flexibility of the reaction structure or insufficient friction at the interfaces with LBCBs.

Recently, an experimental study of a representative 10-story reinforced concrete (RC) frame structure under severe pulse-type seismic ground motion was conducted (Murray and Sasani, 2016). A single RC column with insufficient shear capacity was selected as a test specimen to study the near-collapse system response. A following study by Murray and

(a) MAST, University of Minnesota



(b) MAST, Swinburne University of Technology

(c) LBCB, University of Illinois

Figure 1.3: Available experimental solutions for three-dimensional multi-axial loading

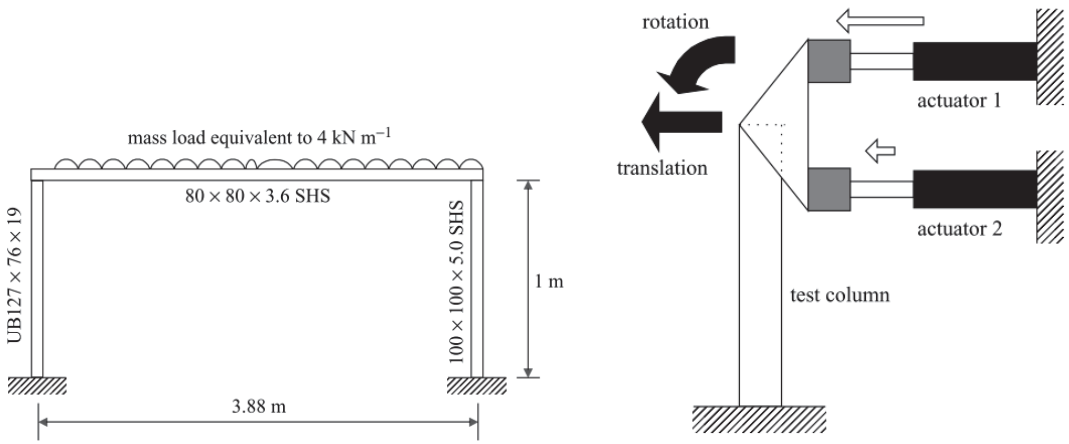
Sasani (2017) considered two shear-critical RC columns as test specimens, and the objective was to assess the collapse resistance of a 7-story structure after combined shear-axial column failure. In both studies, each test column was loaded using an LBCB from the Illinois facility. Similarly, Hashemi et al. (2017) performed a comparison study of two identical, full-scale RC columns, tested to collapse, by using both cyclic and HS testing. In these tests, the limited-ductility RC columns were taken from first-story, corner-columns, of a 5-story RC frame prototype structure, and the loading of each test specimen was applied by using the MAST system at Swinborne. Because multi-axial HS tests provide more realistic boundary effects, the associated fragility analysis is deemed more credible compared to cyclic test results.

This body of literature has proved that multi-axial HS testing has been successfully used for structural performance assessment of full-scale structural systems with multi-directional loading. However, the current multi-axial HS framework is not capable to reproduce real-time, dynamic, three-dimensional loading, because it does not consider either loading assemblies with dynamically-rated servo-hydraulic actuators, or they are not designed with compensation schemes to reduce real-time tracking errors. Dynamic three-dimensional loading could open the door to examine the behavior of strain-rate dependent structural components, such as materials with visco-elastic or visco-plastic relationships, cohesive materials, components with frictional-contact mechanics, etc. Also, even though over-strength effects of concrete and steel due to strain-rate effects are considered to be small, are not negligible and would be desirable to incorporate those effects in experimental testing for better understanding of the physical phenomena. Moreover, strain-rate effects may be very significant for the evaluation of stiff structures (i.e. large natural frequencies) that are subjected to pulse loading (Shing and Mahin, 1988).

In addition, the aforementioned framework has shown some shortcomings due to accuracy problems associated to force relaxation of the physical specimens due to sustained loads from hold sequences during the integration of the numerical substructure (i.e., not performed in continuously or in real-time). Force relaxation up to 10% of the yield capacity for steel structures on a lapse of 0.5 sec has been reported by Mahin et al. (1989). While, some control strategies have been proposed to minimize force relaxation errors in geographically-distributed HS tests (Stojadinovic et al., 2006). Furthermore, an excessive time dilation of the overall experiment could also affect the experiment budget and timeframes for successive testing.

1.4 Multi-actuator dynamic coupling in real-time hybrid simulation

As suggested in the previous sections, performing real-time hybrid simulation (RTHS) testing with a multi-actuator loading assembly would allow researchers to improve the accuracy and reliability of experimental testing for structural performance assessment. Nevertheless, to realize this concept, investigation of multi-actuator dynamic coupling effects is fundamental for successful real-time dynamic tests. For that matter, two cases have been identified from the literature to date: (i) multi-actuators coupled through a rigid link; and, (ii) multi-actuators coupled through a flexible specimen.



(a) Reference structure (b) Experimental substructure

Figure 1.4: RTHS test with twin-actuators coupled through a rigid link (Blakeborough et al., 2001)

Blakeborough et al. (2001) reported one of the earliest experimental studies on real-time hybrid simulation with rigidly-coupled actuators. The reference structure was a portal frame (see Figure 1.4a), where the experimental substructure was chosen as the left-hand column. The RTHS test was composed of two servo-hydraulic actuators that were coupled together with a rigid loading bracket, in order to impose two-degree-of-freedom boundary conditions (translation and rotation) to the free end of a cantilever column specimen, as shown in Figure 1.4b. Similarly, a polynomial extrapolation delay compensation approach (Horiuchi et al., 1996) was considered to control each actuator independently.

This test setup was highly prone to dynamic instabilities, mainly because of the dynamic coupling between the two actuators. The reason for this behavior was not fully explained and understood, but the evidence clearly showed that it was extremely difficult to ensure stability of a multi-degree-of-freedom RTHS test setup. An ad-hoc solution was devised by artificially increasing the value of numerical damping (up to 5% of critical value). This artificial damping had the effect of offsetting the equivalent negative damping associated to actuator dynamics, which causes the test to become unstable. Unfortunately, this approach sacrificed the accuracy of the RTHS overall response, so better solutions to this problem were needed.

In addition, Darby et al. (2002) perform a detailed study of the stability of two-actuator RTHS test setup based on the previous study by Blakeborough et al. (2001). The purpose of the study was to inspect the sensitivity of delay compensation parameters on the stable response of the twin-actuator system. Stability analysis was performed to determine the maximum error in estimated delay of each actuator by trial and error. The main result from this study was that either delay over-compensation or under-compensation of the twin-actuator system can cause instability of the RTHS system. Therefore, the coupled actuator system requires a robust compensation algorithm to avoid unstable responses.

Afterwards, an adaptive delay compensation of the twin-actuator system was proposed. A polynomial extrapolation/interpolation algorithm was used for delay compensation of each

actuator independently, where an online delay estimation technique was proposed. In this test, a damping ratio of 2% was used, which is a more realistic assumption compared to the previous study by Blakeborough et al. (2001), while the the delay compensation with online delay estimation provided a stable response of the RTHS test. Although, the response of the test specimen was constrained to the linear range for the 2DOF RTHS test.

Moreover, this paper provided additional evidence of the effect of specimen stiffness in actuator delay, as it was previously explained by Dyke et al. (1995). For this purpose, a single actuator was attached to the end of the cantilever column, and various specimen were chosen with varying lateral stiffness. Then, actuator delay estimation was conducted. The results showed that actuator delay increased almost linearly with specimen stiffness. Therefore, for test scenarios where a stiffer specimen is considered for RTHS testing, careful assessment and synthesis of delay compensation methods is required in order to avoid any unstable response during the test.

Subsequently, Bonnet et al. (2007) presented an improved investigation on the effects of a highly-stiff coupling system between two identical actuators. The study consisted on the analysis of a three-degree-of-freedom (3DOF) mass-spring system, with two nominally identical dynamic actuators on opposing ends. An adaptive model-based compensator was proposed for RTHS testing using this multi-actuator loading system. The stiffness of the springs was increased to study the phenomenon of actuator coupling. The extreme case considered all three masses joined into one by using two stiffer springs between them, as shown in Figure 1.5. As a result, any attempt to excite the actuators in opposite directions were not able to be run due to instability. This effect was explained as an inability of the outer-loop controller to adapt quickly to the sudden changes in measured force at both boundaries of the specimen. However, when the excitation caused both actuators to move in phase, tests were able to be completed in a stable manner.

To illustrate the later effect, a sine sweep test with 3.5 mm amplitude and frequency range from 0 to 4 Hz on both actuators was conducted. Figures 1.6a and 1.6b shows the synchronization subspace plot and tracking error vs. time plot, respectively. The tracking errors were considerably high, and the average relative error for actuators 1 and 2 was 7.1% and 13.0%, respectively. Also, the test configuration with stiff coupling experienced undesired high frequency oscillations at around 22 Hz, close to the expected rigid body mode of vibration the reference 3DOF structural system. Hence, the authors declared that accurate control of systems with strong dynamic actuator coupling was still a major remaining challenge in multi-actuator RTHS.

Then, Saouma et al. (2014) conducted a study of RTHS with three actuators coupled through a rigid beam, in order to command planar motion (i.e., two translations and one rotation) over the test specimen. The reference structure was a three-story reinforced concrete (RC) moment frame, where the experimental substructure was a first-floor, interior column with non-ductile detailing, as shown in Figure 1.7. Furthermore, a restoring force correction was considered to mitigate experimental errors. Unfortunately, the test implementation suffered extreme challenges that were only possible to be corrected by using ad-hoc solutions. High-frequency vibrations and undesired rotations at the controlled point were experienced during the tests. Also, the peak drift tracking error was reported as 3.29%. A restoring force correction was considered. A large effort was conducted for both PID fine-tuning of

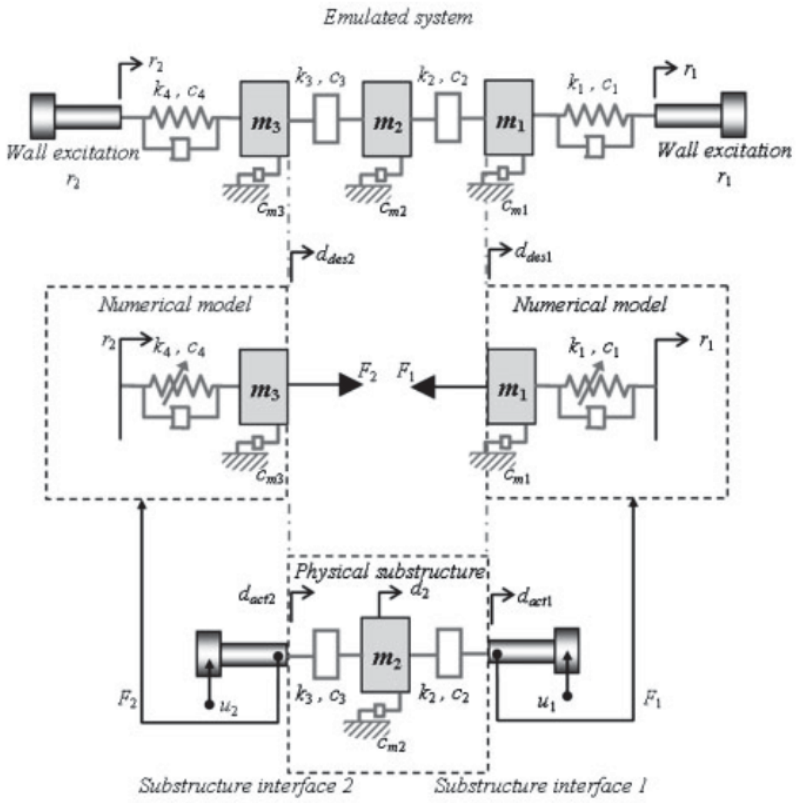
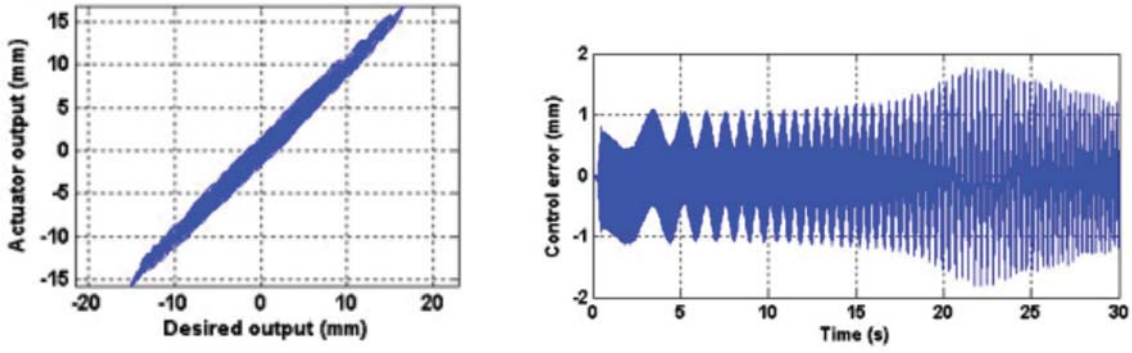


Figure 1.5: RTHS experiment with high actuator coupling (Bonnet et al., 2007)



(a) Synchronization subspace plot

(b) Tracking error

Figure 1.6: Main results from RTHS tests with high multi-actuator coupling (Bonnet et al., 2007)

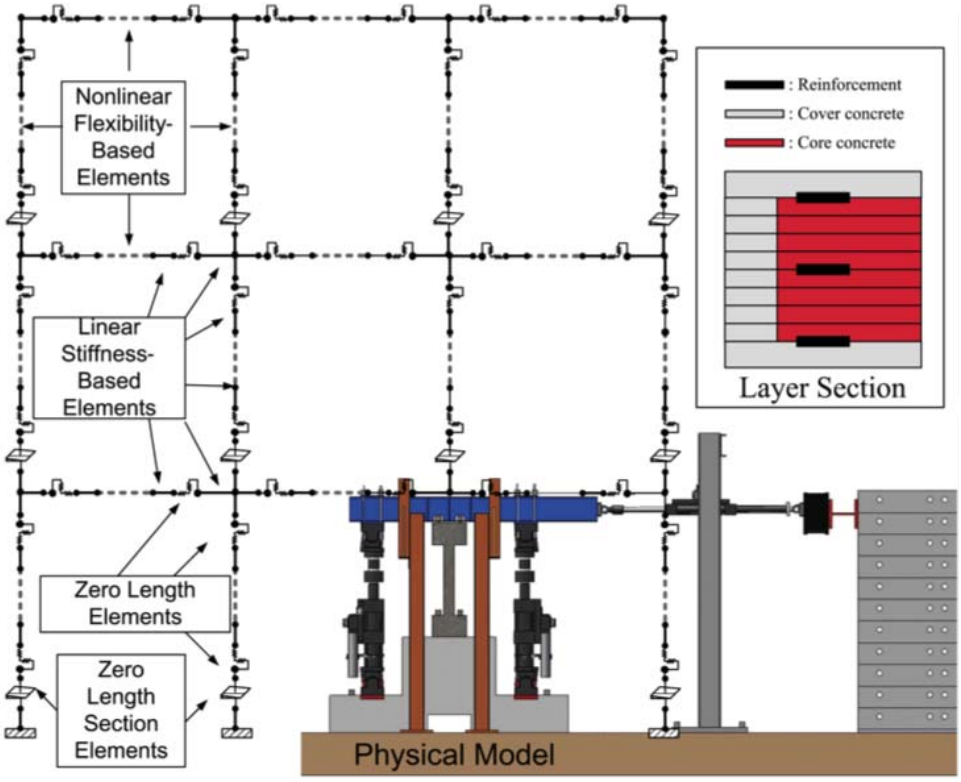


Figure 1.7: RTHS test of a three-story RC moment frame, with multi-actuator coupling through a rigid link (Saouma et al., 2014)

the servo-controller, and system identification, with destructive tests of 20 specimens in total. The lack of a rational approach for compensation of both the actuator dynamics and coupling effects could be the main reason for the undesired responses reported in this study.

Furthermore, the literature provides some results of RTHS testing for soft-coupling of multi-actuator systems when connected to one flexible test specimen. Investigations of two coupled actuators (Gao et al., 2014; Jung et al., 2007; Wallace et al., 2005), as well as three coupled actuators (Cha et al., 2014; Dong et al., 2015; Friedman et al., 2015; Phillips and Spencer, Jr., 2013b) have been reported in the literature. In particular, Phillips and Spencer, Jr. (2013b) was the first study to account directly for the multi-input, multi-output (MIMO), actuator coupling effects in the design of model-based compensators for multi-actuator RTHS. This analytical study considered a three-story shear building system, where the numerical substructure included only the inertial effects and intrinsic damping, while the physical substructure included both linear (columns) and nonlinear (MR damper) structural components, as shown in Figure 1.8. While, three identical actuators were located at each story level to impose the displacements after integration of the numerical equations of motion. Numerical simulations portrayed a significant improvement of accuracy for the RTHS testing when the multi-actuator coupling effects were considered explicitly on the design of model-based compensators. This novel technique for multi-actuator RTHS was implemented and validated in an experimental test of a three-story steel frame equipped

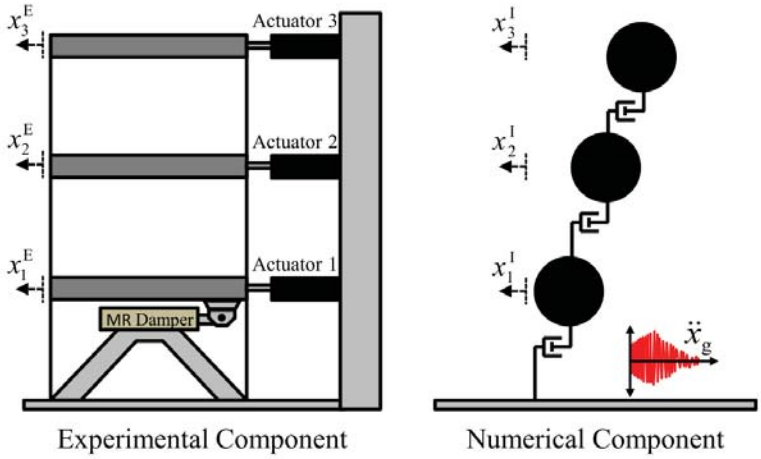


Figure 1.8: Multi-actuator coupling through a test specimen (Phillips and Spencer, Jr., 2013b)

with MR dampers (Cha et al., 2014).

On the other hand, Nakata and Krug (2013) and Nakata et al. (2014) studied two-coupled actuators for multi-degree-of-freedom effective force testing (EFT). Other examples of multi-actuator RTHS have been proposed with the addition of shake tables. For example, (Reinhorn et al., 2004, 2005) conducted an investigation on force-controlled real-time hybrid simulation, where a one-story building specimen was connected between a six-degree-of-freedom (6DOF) shake table and a dynamically rated actuator. The compensation of the single dynamic actuator was conducted by increasing the mechanical compliance installing a spring in series with the actuator; but, shake table control was not included in the design of dynamic compensators for RTHS purposes. Subsequent studies improved the force-controlled RTHS by including compensation of only unidirectional shake tables (Shao and Reinhorn, 2012; Shao et al., 2011).

Nevertheless, the use of more than three coupled dynamic actuators for RTHS testing has not been reported in the literature to date. More importantly, the research on multi-actuator RTHS with rigid coupling has only explored situations when the specimen behaves in the linear range. Also, research on this topic for nonlinear specimens is lacking in the literature. Indeed, as the discussed in the previous section, the advancement of dynamic multi-axial testing will require a minimum of six coupled dynamic actuators, in order to impose realistic three-dimensional dynamic loading over structural specimens.

1.5 Overview

In this study, a novel framework to conduct *multi-axial real-time hybrid simulation (maRTHS)* testing is proposed. In maRTHS testing, the experimental substructures will be subjected to realistic three-dimensional dynamic loading, which consists of a total of six-degree-of-freedom (6DOF): three translational and three rotational motion. The 6DOF loading considered for this research will be displacement-controlled in Cartesian space.

1.5.1 Main objectives

The main objectives of this research are:

1. Develop and validate a reliable and efficient framework for three-dimensional, multi-axial, real-time hybrid simulation (maRTHS) testing.

2. Enable the multi-actuator loading assembly to accurately impose three-dimensional dynamic boundary conditions on a physical specimen for realistic performance assessment of structural systems, especially for materials with rate-dependent behavior.

3. Provide sufficient guarantees for stable and robust maRTHS testing of physical specimens with different stiffness levels.

1.5.2 Broader impacts

In carrying out the vision of the proposed research, the scientific community will have improved and cost-efficient tools to promote for seismic resiliency of civil infrastructure. The proposed research will have a considerable impact on the earthquake engineering research community, pushing forward the development of real-time hybrid simulation testing to new heights, while promoting large-scale, cost-effective, experimental evaluation for dynamic characterization of new innovative systems that are not well understood or are difficult to model numerically.

The development of the proposed framework for multi-axial real-time hybrid simulation (maRTHS) of complex systems under complex loading has the potential of increasing the class of structures that can be experimentally tested using the hybrid simulation technique, while enabling significant reductions on costs through substructuring methods. In addition, it may allow experimental testing of full-scale specimens where some of their mechanical properties cannot easily be scaled down in smaller prototypes, i.e. grain size in concrete, defects on materials that causes propagation of cracks, geometric and material instabilities, etc. This opportunity opens a promising field that could incorporate new materials and structural systems into future versions of building design codes.

This maRTHS framework could allow for full-scale testing of massive structures such as high-rise buildings and long-span bridges, while overcoming current payload and/or spatial restrictions of state-of-the-art laboratory facilities with large shaking tables (see Figure 1.9).

In addition, there are many interesting applications that could take advantage of this proposed framework, including but not limited to:

- influence of vertical loading effect on the lateral stability of elastomeric bearing (Sanchez et al., 2013) and friction pendulum bearing (Ryan and Dao, 2016) isolators subjected to three-dimensional seismic loading;

- design of supplemental energy dissipation devices with multi-axial loading, e.g. viscoelastic (VE) coupling dampers (Christopoulos and Montgomery, 2013) installed on mid-rise to high-rise buildings subject to either seismic and wind vibration (Figure 1.10);

Figure 1.9: Large-scale multi-axial shaking table facility (E-Defense, Japan)

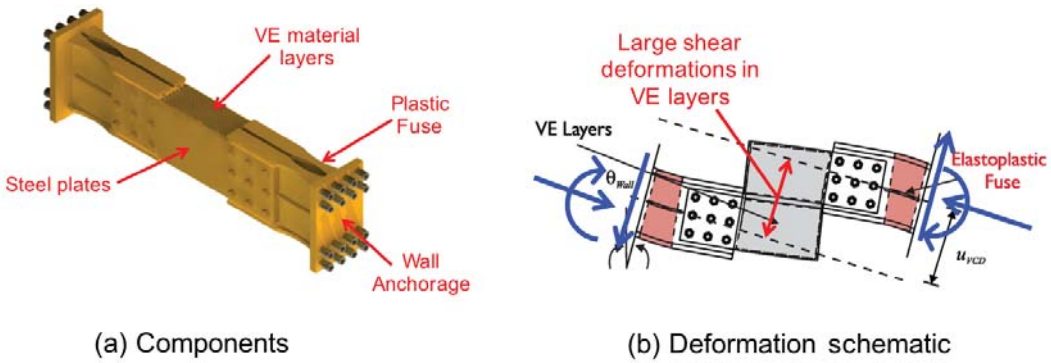


Figure 1.10: Viscoelastic (VE) coupling damper for energy dissipation of building systems (Christopoulos and Montgomery, 2013)

- self-centering friction connections for steel moment frame systems, as shown in Figure 1.11 (Kim and Christopoulos, 2008);
- soil-structure interaction, where problems of soil mass inertial loading, the permeability of the soil in partially saturated conditions, and liquefaction potential, are critical issues that need to be assessed experimentally. Studies on rocking isolation systems with elasto-plastic / visco-plastic soil specimens (Anastasopoulos et al., 2013) and dynamic soil-structure interaction of wind turbines (Lombardi et al., 2013) are promising applications for maRTHS testing, as shown in Figures 1.12 and 1.13;
- large-scale studies of magneto-rhelogical elastomer (MRE) bearings for semi-active base isolation (Li et al., 2013);
- other studies, such as earthquake pulse-like response on non-ductile systems, system-level performance assessment of civil infrastructures subject to progressive collapse scenarios, etc.

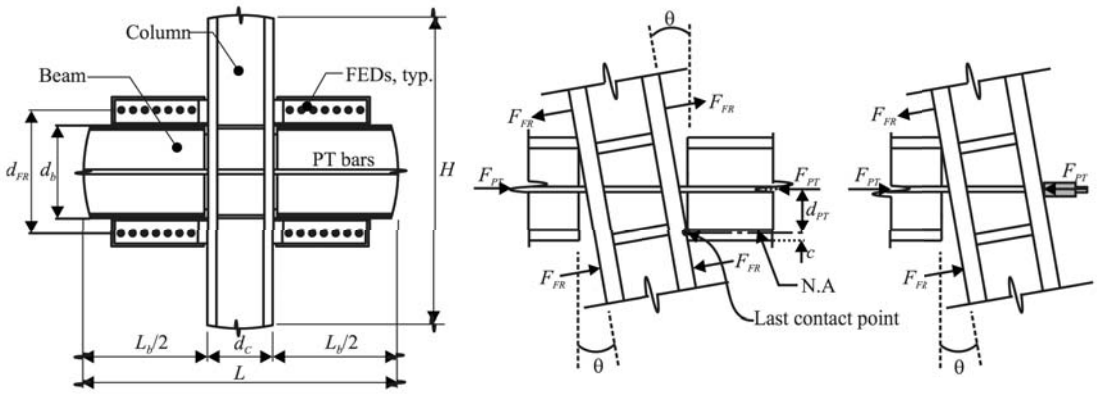


Figure 1.11: Self-centering, post-tensioned (PT), steel moment frame connection with friction energy dissipation (FED) devices, subjected to multi-axial loading conditions (Kim and Christopoulos, 2008)

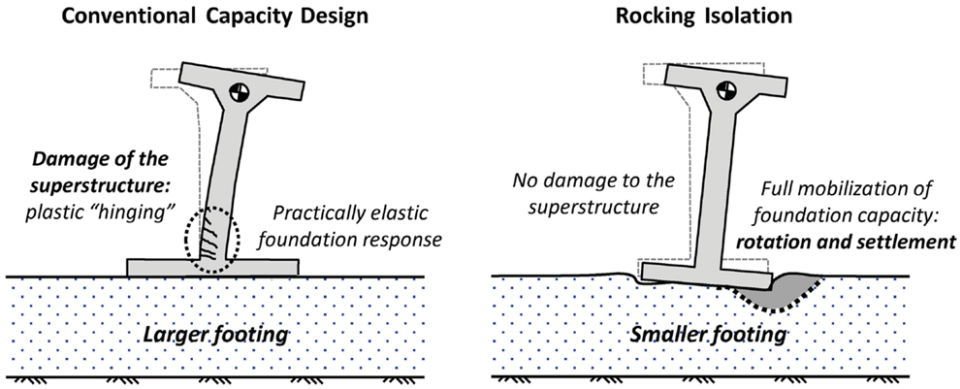


Figure 1.12: Rocking isolation design for bridge piers with soil-structure interaction (Anastasopoulos et al., 2013)

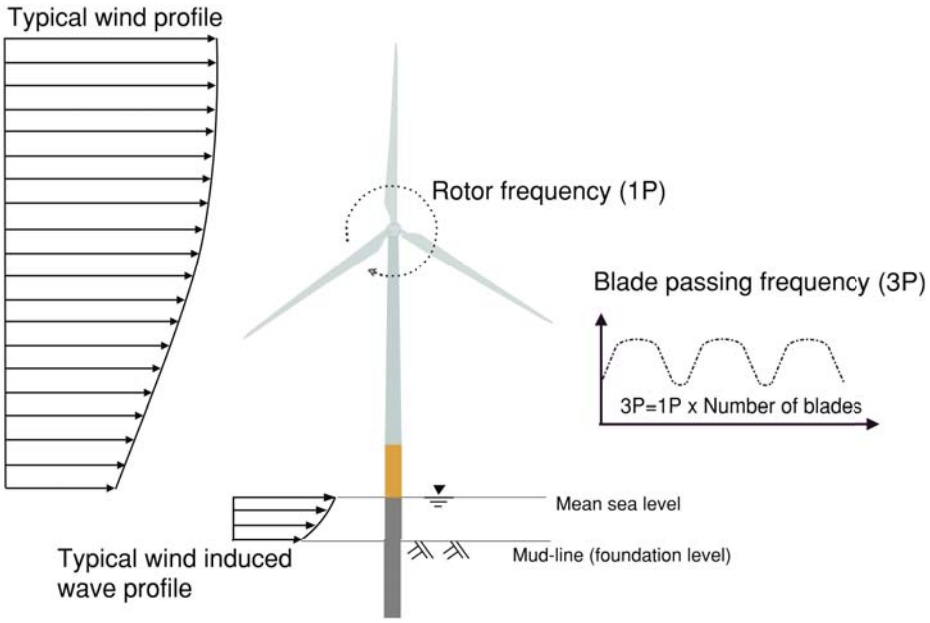


Figure 1.13: Dynamic effects of offshore wind turbine supported on monopile foundation (Lombardi et al., 2013)

1.5.3 Report organization

Chapter 2 will provide the necessary background to conduct this research. An overview of techniques used in real-time hybrid simulation (RTHS) will be presented, such as substructuring, numerical integration schemes, and compensation algorithms. Then, an overview of the governing equations for servo-hydraulic actuators will be provided, in order to briefly discuss about its dynamical properties. This knowledge is fundamental for effective compensation in RTHS testing. Finally, a brief summary of modern control theory for linear systems will be provided, with emphasis to multi-input, multi-output (MIMO) systems. This information will be fundamental to develop model-based compensation algorithms for maRTHS in the following sections.

Chapter 3 will explain the proposed framework for multi-axial real-time hybrid simulation (maRTHS). In order to allow for real-time execution of the proposed framework, fast and reliable hardware is required. As such, the objective is to integrate control and measurement systems to enable maRTHS using real-time hardware. This hardware will consists in a fast micro-controller, including multi-channel digital-to-analog (D/A) converters for servo-valve command voltage signals, and multi-channel analog-to-digital (A/D) converters for multiple sensor measurements (i.e. load cells, displacement transducers, etc.). The framework implementation on a small-scale LBCB assembly will be presented.

Chapter 4 introduces the topic of kinematic transformations for tracking of multi-actuator systems in Cartesian space. The target displacements from the numerical substructure are applied to the physical specimen by using multiple servo-hydraulic actuators attached to the loading platform. The actuator's piston can only be commanded to move along its primary axis, therefore if the multiple actuators of the loading assembly are not aligned with the global Cartesian system of coordinates, a kinematic transformation between Actuator and Cartesian space coordinates will be required for successful maRTHS testing. The goal of this task is to develop an explicit solution of inverse and forward kinematic transformations using an external sensor approach attached to the loading platform. In addition, a calibration procedure for actuators and external displacement transducers will be developed, in terms of both command and measured strokes, using a contact-less dynamic measuring machine. This will offer a practical way to perform fast and accurate measurements for the position of the controlled motion in Cartesian coordinates.

Chapter 5 explains the system identification procedure for multi-actuator systems. In order to provide good tracking and robustness to the overall system, a very accurate representation of the dynamics of the experimental setup is needed for the purpose of developing compensators for stable and robust system responses. The objective for this task is to obtain an accurate model of the multi-input, multi-output (MIMO) experimental system, that incorporates all the parameters from the servo-hydraulic actuators, the test specimen interaction, and the actuator dynamic coupling effects. To achieve this, a nonparametric frequency-domain system identification procedure is considered . The data collected to create this models are obtained from multi-input random vibration excitations to the physical component. The final outcome will be a MIMO physical model of the experimental system that will be used for compensation design. In addition, this model will be fundamental to increase our understanding of the multi-actuator dynamic coupling, and the effects of

specimen interaction.

Chapter 6 illustrates the compensation design approach for multi-actuator systems in the context of maRTHS testing. For the purposes of this research, a model-based controller is considered in this study. Consequently, the goals for this task are to develop a multi-input, multi-output (MIMO) model-based controller to perform maRTHS testing. This controller will allow for stable and accurate tracking of the overall system response, without the need of any ad-hoc artifice like in previous research. Moreover, the controller is designed considering the global Cartesian displacements of the physical specimen, quite different from other RTHS solutions where single actuator feedback was considered. In particular, a feedforward-feedback control architecture will be considered. The feedforward compensator will be designed using the inverse model of the experimental system, and will be responsible of tracking the target displacements with zero-phase delay error. Similarly, the feedback controller will consider an LQG/LTR approach to provide additional robustness to the system when the feedforward is not able to perfectly track the target displacements due to model uncertainty. The performance of the designed controllers will be assessed in terms of analytical simulations and experimental tests for specimens with varying degree of relative stiffness.

Chapter 7 presents the results of a small-scale experimental test, carried out for validation purposes of the proposed framework. To demonstrate the effectiveness of this framework, successive tests and adjustments must be considered to guarantee that the experimental results satisfy the main goals of this proposed research. Therefore, the next steps on this development are to perform a series of small-scale tests that will incrementally increase the degree of actuator dynamic coupling, the range of kinematic transformations, and the nature of the physical specimens to be tested (elastic and inelastic). This will enable a test-bed to verify and evaluate the proposed framework and methods developed during this research.

Finally, **Chapter 8** will provide some final remarks of this research, and will specify future directions to continue the research on this topic for continued improvement.

Chapter 2

BACKGROUND

2.1 Substructuring method

Consider an initial boundary value problem (IBVP) with a domain defined by Ω and boundary conditions Γ_g and Γ_h for prescribed displacements and loads, respectively. This IBVP represents the dynamic response of a reference structural system, such as a building, or a bridge. After discretization of the IBVP (e.g., by finite elements method), the dynamic response of the structural system is obtained in terms of degrees-of-freedom (DOF) $\mathbf{u}(t) \forall t \in [0, t_f]$, by solving the following equation of motion:

$$\Omega : \quad \mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{r}\left(\mathbf{u}(t)\right) = \mathbf{p}(t) \tag{2.1}$$

where $\mathbf{u}$, $\dot{\mathbf{u}}$ and $\ddot{\mathbf{u}}$ are the displacement, velocity and acceleration vectors, respectively; $\mathbf{M}$ is the mass matrix; $\mathbf{C}$ is the linear damping matrix; $\mathbf{r}(\mathbf{u})$ is the nonlinear internal (restoring) force vector; and $\mathbf{p}$ is the vector of externally applied forces. For earthquake loading, the external force vector is determined as $\mathbf{p} = -\boldsymbol{\mu}\ddot{u}_g$, where $\boldsymbol{\mu}$ is the seismic participation vector, and $\ddot{u}_g$ is the ground acceleration. In addition, the initial conditions at time $t = 0$, $\mathbf{u}(0) = \mathbf{u}_0$ and $\dot{\mathbf{u}}(0) = \dot{\mathbf{u}}_0$, are required to solve the system of coupled ordinary differential equations (ODEs) for all time $t \in [0, t_f]$, where $t_f > 0$ is the final time of the numerical simulation.

In the context of this research, instead of solving the IBVP for the entire domain, a process known as *substructuring* can be employed to subdivide the domain into smaller subdomains, such that the order of large and complex structural systems is reduced for efficient computations. Each subdomain can be solved independently, provided that coupling between components is enforced by means of compatibility and equilibrium conditions at their interfaces (de Klerk et al., 2008).

For example, the domain can be defined as the union of two smaller subdomains, $\Omega = \Omega^N \bigcup \Omega^E$, as shown in Figure 2.1. The subdomains Ω^N and Ω^E are called the numerical and experimental substructures, respectively. Each subdomain can have its own boundary conditions for prescribed displacements (Γ_g^N and Γ_g^E) and prescribed loads (Γ_h^N and Γ_h^E), as shown in Figure 2.2.

Let the displacement vector of the associated numerical and experimental substructures be defined as:

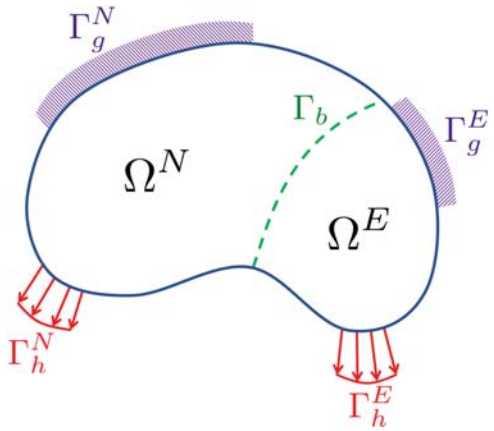


Figure 2.1: Initial boundary value problem (IBVP) for a structural system

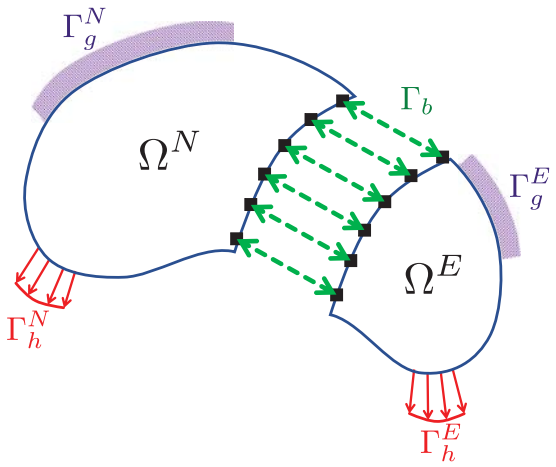


Figure 2.2: Substructuring of dynamical system

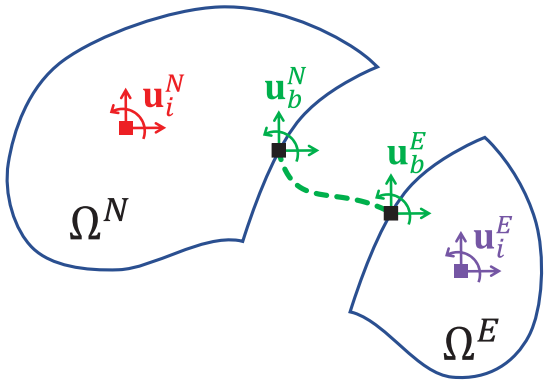


Figure 2.3: Degrees-of-freedom (DOF) of numerical (Ω^N) and experimental (Ω^E) substructures

$$\mathbf{u}^N = \begin{Bmatrix} \mathbf{u}_i^N \\ \mathbf{u}_b^N \end{Bmatrix}, \quad \mathbf{u}^E = \begin{Bmatrix} \mathbf{u}_i^E \\ \mathbf{u}_b^E \end{Bmatrix} \tag{2.2}$$

where the indices "N" and "E" denote the numerical and experimental substructures, respectively; and "i" and "b" refer to the interior and boundary DOFs, respectively. The displacement vectors for each substructure is better illustrated in Figure 2.3. Then, the equations of motion (EOM) for both numerical (Ω^N) and experimental (Ω^E) coupled substructures are expressed as follows:

$$\Omega^N : \quad \mathbf{M}^N\ddot{\mathbf{u}}^N + \mathbf{C}^N\dot{\mathbf{u}}^N + \mathbf{r}^N(\mathbf{u}^N, \dot{\mathbf{u}}^N) = \mathbf{p}^N + \mathbf{g}^N \tag{2.3}$$

$$\Omega^E : \quad \mathbf{M}^E\ddot{\mathbf{u}}^E + \mathbf{C}^E\dot{\mathbf{u}}^E + \mathbf{r}^E(\mathbf{u}^E, \dot{\mathbf{u}}^E) = \mathbf{p}^E + \mathbf{g}^E \tag{2.4}$$

and the coupling force vector for each substructure is defined by:

$$\mathbf{g}^N = \begin{Bmatrix} \mathbf{g}_i^N \\ \mathbf{g}_b^N \end{Bmatrix}, \quad \mathbf{g}^E = \begin{Bmatrix} \mathbf{g}_i^E \\ \mathbf{g}_b^E \end{Bmatrix} \tag{2.5}$$

To solve this coupled problem, both compatibility and equilibrium conditions must be satisfied. First, the compatibility condition is given by:

$$\mathbf{u}_b^N = \mathbf{u}_b^E \tag{2.6}$$

The main assumption in this formulation is that the substructures are only coupled through the boundary Γ_b. Therefore, the coupling forces at interior DOFs for each substructure should be equal to zero:

$$\mathbf{g}_i^N = \mathbf{0}_i^N \tag{2.7}$$

$$\mathbf{g}_i^E = \mathbf{0}_i^E \tag{2.8}$$

Furthermore, the equilibrium condition for boundary DOFs is given by:

$$\mathbf{g}_b^N + \mathbf{g}_b^E = \mathbf{0}_b \tag{2.9}$$

Therefore, by substituting (2.9) and (2.5) into (2.3), the following "coupled" numerical substructure EOM is obtained:

$$\mathbf{M}^N \ddot{\mathbf{u}}^N + \mathbf{C}^N \dot{\mathbf{u}}^N + \mathbf{r}^N(\mathbf{u}^N) = \mathbf{p}^N + \begin{Bmatrix} \mathbf{0}_i^N \\ -\mathbf{g}_b^E \end{Bmatrix} \tag{2.10}$$

where $\mathbf{g}_b^E$ is the coupling force vector from the experimental component, which includes all the effects associated with nonlinear restoring forces, nonlinear damping, and inertial forces, along with any external excitation that can be induced directly to the experimental substructure:

$$\mathbf{g}^E = \begin{Bmatrix} \mathbf{0}_i^E \\ \mathbf{g}_b^E \end{Bmatrix} = \mathbf{M}^E \ddot{\mathbf{u}}^E + \mathbf{C}^E \dot{\mathbf{u}}^E + \mathbf{r}^E(\mathbf{u}^E) - \mathbf{p}^E \tag{2.11}$$

while noting that the coupling vector $\mathbf{g}_b^E$ is a function of displacement vector $\mathbf{u}_b^N$ to satisfy (2.6):

$$\mathbf{u}^E = \begin{Bmatrix} \mathbf{u}_i^E \\ \mathbf{u}_b^E = \mathbf{u}_b^N \end{Bmatrix} \tag{2.12}$$

To obtain an admissible solution, compatibility and equilibrium must be satisfied for all boundary DOFs at all times. Therefore, an algorithm should be considered to prescribe displacements and forces at the boundary Γ_b for the solution of the dynamical system. The algorithm chosen for hybrid simulation is illustrated in Figure 2.4. After solving the EOM (2.10) of numerical substructure Ω^N through a time-stepping integration algorithm (see Section 2.3), the output $\mathbf{u}_b^N$ is commanded to the experimental substructure Ω^E to satisfy compatibility condition at the interface Γ_b. Then, the coupling force $\mathbf{g}_b^E$ is measured directly from the test specimen after displacement-controlled loading, using load cell sensors in a laboratory facility, and this output is inserted back into the numerical substructure Ω^N, to satisfy the equilibrium condition at interface Γ_b. This "hybrid loop" procedure is repeated until the simulation reaches the final simulation time t_f. Moreover, due to the time constraints imposed in real-time systems, the hybrid loop must be performed in a fast and timely manner in order to conduct RTHS testing.

Finally, other substructuring techniques have been proposed for hybrid simulation testing, such as overlapping methods (Hashemi and Mosqueda, 2014), where the substructures are overlapped by more than the interface nodes, while it can also share redundant elements. This overlapping technique is conceived for the main purpose of alleviating the requirements on number of actuators at the interface of experimental subassemblies. Also, mixed-mode control (Nakata et al., 2007) and switch control (Yang et al., 2017) has been proposed to command forces in addition to displacements over the experimental substructure. However, real-time force control of actuators is much more complex than real-time displacement control due to multi-actuator dynamic coupling, specimen interaction, and measurement errors from load cells. Therefore, substructuring with only displacement control will be primarily

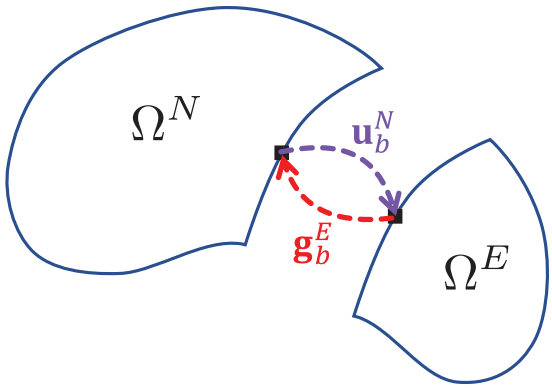


Figure 2.4: Coupling of numerical (Ω^N) and experimental (Ω^E) substructures

presented in this study for maRTHS testing purposes.

2.2 Real-time systems

As discussed in Section 1.2, *real-time hybrid simulation* (RTHS) is an experimental technique to conduct dynamic testing of physical specimens coupled with numerical structures. To allow for dynamic loading and fast computations of numerical substructures and feedback control algorithms, among others, a *real-time system* is required.

The concept of a real-time system comes from computer science and is used to describe the collection of hardware and software systems that are subjected to *timing and resource constraints*. The goal of real-time systems is to provide necessary guarantees of predictable and timely computational behavior under all expected operating conditions when the system interacts with the physical environment. In other words, computer programs must perform "tasks" within specified "deadlines". If timing constraints are not met by the computer response, then the real-time system could cause performance degradation or even lose reliability, i.e., the probability of system failure increases.

In general, real-time systems (RTS) are classified by the consequence of missing a deadline: (i) *hard RTS* is such that missing any deadline will cause total system failure; and (ii) *soft RTS* is when the system allows for frequent missing of deadlines without failure, but at the expense of system performance (Buttazzo, 2011). In the context of this research, RTHS testing can be considered as hard RTS. In general, RTHS hardware and software must ensure that all deadlines of the hybrid loop are strictly met. If this condition is not guaranteed, it could cause failure of the experimental setup and damage to laboratory equipment and its surroundings.

Therefore, to ensure that constraints are strictly met for RTHS testing, the real-time system must be designed taking into account the tradeoff between high-fidelity numerical models and simulation/integration time steps. Large and complex numerical substructures are necessary to capture local phenomena that affect global behavior. It may be possible that sophisticated material and element models (e.g. finite deformations, inelasticity, etc.) are

necessary to estimate the expected phenomena. But, this leads to tasks with the increasing use of computational resources, thus requiring a significant amount of time to solve with the potential of missing deadlines in real-time execution. On the other hand, if the time to meet a deadline is larger, the reliability of the system is guaranteed, but at the expense of a reduction on the computational resources available for numerical integration and other arithmetic operations. In this regard, it may be possible that only macro (coarse) models are allowed for RTHS tests with this particular RTS design, which may not offer accurate results compared to more refined models.

For example, Huang et al. (2010) considered the relationship between numerical model fidelity and timing constraints in RTHS testing. For this study, different multi-story shear buildings (5 to 280 DOFs) were considered, where only the first floor is chosen as the experimental substructure. The test specimen was a linear-elastic spring loaded by a servo-hydraulic actuator. The numerical substructure was solved using an Intel Core2 Quad 2.66 Hz CPU machine, with 4GB RAM memory, and NI-DAQ card for analog/digital communication with actuators and sensors. Also, the software consisted of C++ and Matlab code running on a standard Linux platform. A sampling rate of 1280 Hz was chosen, and 10 trials were conducted for each RTHS test. The experimental results show a direct dependency between a number of deadline misses and size of the numerical substructure. For this experiment, if the numerical substructure was larger than 255 DOFs, then the computational load becomes too large for the RTS to keep a simulation rate of 1280 Hz, therefore reducing its system reliability.

Hence, careful choice of hardware and software must be considered for a reliable and accurate RTHS platform. In terms of hardware, a faster real-time target machine would imply that timing constraints are less restrictive. The literature offers a wide variety of target machine solutions for RTHS testing; for example: dSpace (Phillips and Spencer, Jr., 2012), Speedgoat (Gao et al., 2013), National Instruments (NI) (Liu et al., 2016), among others. These target machines differ primarily in computer architecture, timer speed, shared memory, and digital signal processing (DSP) capabilities. Also, target machines usually work with a real-time operating system (RTOS), which is responsible for task scheduling and handling of both computational resources and timing constraints. Examples of RTOS kernels are Matlab's Simulink Real-Time (formerly xPC Target), NI Linux Real-Time, LabView Real-Time, etc.

Moreover, software applications are developed to work over the RTOS platform, to perform the required operations in real-time. Most standard software packages have been developed in Matlab/Simulink IDE for direct integration over target machines. For example, HybridFEM (Karavasilis et al., 2009) and RT-Frame2D (Castaneda et al., 2012) have been developed especially for RTHS testing.

In case that standard structural analysis software is required to perform the calculations of the numerical substructure (e.g., OpenSees, Abaqus, Zeus-NL), then a simulation coordinator is required. In traditional hybrid simulation (HS) testing, the coordinator is responsible for communications between multiple actors in the hybrid loop and usually performs time-stepping integration locally. These actors can be numerical and/or experimental substructures located in the same laboratory or even geographically-distributed substructures. Examples of standard coordinator packages are UI-SIMCOR (Kwon et al., 2007) and

OpenFresco (Schellenberg et al., 2009). But, in RTHS tests, simulation coordination must be performed in real-time; hence, the real-time software application is usually designed to serve the role of simulation coordinator and is tailored specifically for a particular experimental setup. In that case, the coordinator is just a simple communication link between numerical and physical substructures located in the same facility.

Finally, some studies have focused on extending the RTHS capabilities by allowing RTS systems with multiple periodic rates, to solve multiple tasks with different deadlines. Maghareh et al. (2016b) presented a multi-rate transition between numerical and experimental substructures, which solve the numerical model at a larger time-step; while, an adaptive polynomial interpolation-extrapolation scheme is used to send command displacements continuously to the experimental setup. In addition, parallel computing has been explored to relax the constraints of real-time systems in RTHS testing.Bunting (2016) considered a multi-time-step (MTS) approach, where the numerical model is partitioned into refined and coarser models, depending on its proximity to the experimental substructure. Then, the refined and coarse models are integrated at different time steps to ensure strict timing constraints and efficiency of the solutions.

2.3 Numerical integration of equations of motion

A fundamental aspect of RTHS testing is the choice of fast and accurate numerical integration schemes to solve the multi-degree-of-freedom (MDOF) equations of motions from the numerical substructure (see Section 2.1). Consider the following nonlinear equation of motion (EOM):

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{C}\dot{\mathbf{u}}(t) + \mathbf{r}(\mathbf{u}(t)) = \mathbf{p}(t) \tag{2.13}$$

where $\mathbf{u}$ is the displacement vector, $\mathbf{p}$ is the external force vector. The previous equation can also be formulated using a state-space representation. Let the state vector $\mathbf{x}$ be defined as:

$$\mathbf{x} = \begin{Bmatrix} \mathbf{u} \\ \dot{\mathbf{u}} \end{Bmatrix} \tag{2.14}$$

Then, (2.13) can be reformulated as a system of nonlinear ordinary differential equations (ODEs) in terms of state vector $\mathbf{x}$ and input force $\mathbf{p}$:

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left(t, \mathbf{x}(t), \mathbf{p}(t)\right) \tag{2.15}$$

To solve this dynamical system, numerical time-integration methods are generally considered. Therefore, by assuming a time-discretization $t_k = k\Delta t, \forall k = \{0, 1, 2, \ldots, t_f/\Delta t\}$, where Δt is the time step, and t_f is the final simulation time. By defining the discrete vectors, $\mathbf{x}_k = \mathbf{x}(t_k)$, $\mathbf{r}_k = \mathbf{r}(\mathbf{x}_k)$, and $\mathbf{p}_k = \mathbf{p}(t_k)$, then the EOM (2.13) can be expressed as:

$$\mathbf{M}\ddot{\mathbf{u}}_k + \mathbf{C}\dot{\mathbf{u}}_k + \mathbf{r}_k = \mathbf{p}_k \tag{2.16}$$

Similarly, the system of ODEs (2.15) can be discretized in time as follows:

$$\dot{\mathbf{x}}_k = \mathbf{f}\left(t_k, \mathbf{x}_k, \mathbf{p}_k\right) \tag{2.17}$$

Clearly, the derivatives $\ddot{\mathbf{u}}_k$ and $\dot{\mathbf{u}}_k$ for EOM, or $\dot{\mathbf{x}}_k$ for ODEs, must be approximated by using finite difference approximations. Hence, the solution to the dynamical problem can be expressed by either *explicit* or *implicit* algorithms. In explicit integration, the calculation of the states of the system at a future time is done by evaluating an expression that depends only on the current states. Also, implicit integration involves the solution of a nonlinear equation of both the current and future states. The following algebraic equations for state vector $\mathbf{x}$ represents the structure of both algorithms:

$$\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{p}_k) \qquad \text{(explicit)} \tag{2.18}$$

$$\mathbf{G}(\mathbf{x}_{k+1}, \mathbf{x}_k, \mathbf{p}_k) = \mathbf{0} \qquad \text{(implicit)} \tag{2.19}$$

where $\mathbf{F}(\cdot)$ and $\mathbf{G}(\cdot)$ are functions derived from implementing explicit and implicit integration schemes, respectively. Explicit algorithm provides a direct solution to the dynamic problem for future state $\mathbf{x}_{k+1}$, even though $\mathbf{F}(\cdot)$ may be a nonlinear function. But, when $\mathbf{G}(\cdot)$ is a nonlinear function, then the solution for future state $\mathbf{x}_{k+1}$ will require additional iterations to achieve convergence, hence the solution is implicit.

Although, a major disadvantage of explicit algorithms is the fact that is *conditionally stable*, i.e., the choice of time step Δt will affect its numerical stability. On the other hand, most implicit algorithms are unconditionally stable, but not all of them have this property. In the context of RTHS testing, iterations may pose a serious problem because of time constraints of the real-time system when the hybrid loop is implemented (see Section 2.2). For this reason, explicit algorithms have been extensively used in RTHS testing, such that integration would be performed as fast as possible (McCrum and Williams, 2016; Shing, 2008).

Hence, the choice of numerical integration schemes is a crucial aspect of RTHS implementation. Historically, the first RTHS tests were implemented using explicit integration, mainly because they do not require any iteration sub-steps to solve the equations of motion in real-time, with the requirement that stability margins must be obtained in order to guarantee that the numerical results are bounded. But some implicit algorithms have been reported in the literature as well. A representative set of numerical integration algorithms used in RTHS testing is presented in the following sections.

2.3.1 Nonlinear systems

For a MDOF system with a nonlinear restoring force vector, let the EOM be:

$$\mathbf{M}\ddot{\mathbf{u}}_k + \mathbf{r}(\mathbf{u}_k, \dot{\mathbf{u}}_k) = \mathbf{p}_k \tag{2.20}$$

The nonlinear term $\mathbf{r}(\mathbf{u}_k, \dot{\mathbf{u}}_k)$ can be linearized using Taylor series expansion:

$$\mathbf{r}(\mathbf{u}_{k+1}, \dot{\mathbf{u}}_{k+1}) = \mathbf{r}(\mathbf{u}_k, \dot{\mathbf{u}}_k) + \frac{\partial \mathbf{r}}{\partial \mathbf{u}}(\mathbf{u}_k, \dot{\mathbf{u}}_k)\Delta\mathbf{u} + \frac{\partial \mathbf{r}}{\partial \dot{\mathbf{u}}}(\mathbf{u}_k, \dot{\mathbf{u}}_k)\Delta\dot{\mathbf{u}} + \mathcal{O}(\|\Delta\mathbf{u}\|^2, \|\Delta\dot{\mathbf{u}}\|^2) \quad (2.21)$$

Defining the tangent stiffness and tangent damping matrices:

$$\mathbf{K} = \frac{\partial \mathbf{r}}{\partial \mathbf{u}}(\mathbf{u}_k, \dot{\mathbf{u}}_k) \quad (2.22)$$

$$\mathbf{C} = \frac{\partial \mathbf{r}}{\partial \dot{\mathbf{u}}}(\mathbf{u}_k, \dot{\mathbf{u}}_k) \quad (2.23)$$

and the incremental terms:

$$\Delta\mathbf{a} = \mathbf{a}_{k+1} - \mathbf{a}_k \quad (2.24)$$

$$\Delta\mathbf{v} = \mathbf{v}_{k+1} - \mathbf{v}_k \quad (2.25)$$

$$\Delta\mathbf{u} = \mathbf{u}_{k+1} - \mathbf{u}_k \quad (2.26)$$

where $\mathbf{v}_k \triangleq \dot{\mathbf{u}}_k$ and $\mathbf{a}_k \triangleq \ddot{\mathbf{u}}_k$. Then, the linearized EOM is expressed in incremental form:

$$\mathbf{M}\Delta\mathbf{a} + \mathbf{C}\Delta\mathbf{v} + \mathbf{K}\Delta\mathbf{u} = \Delta\mathbf{p} \quad (2.27)$$

where $\Delta\mathbf{p}$ is the residual force vector:

$$\Delta\mathbf{p} = \mathbf{p}_k - \mathbf{M}\mathbf{a}_k - \mathbf{r}(\mathbf{u}_k, \mathbf{v}_k) \quad (2.28)$$

Hence, the solution of the nonlinear dynamic system is obtained incrementally, by solving the linearized equilibrium equation (2.27) at step k, and obtaining the increment values $\Delta\mathbf{u}$, $\Delta\mathbf{v}$ and $\Delta\mathbf{a}$, to update the responses for next step $k+1$. The solution of (2.27) is typically performed through iteration, with finite difference approximations for $\mathbf{u}_{k+1}$, $\mathbf{v}_{k+1}$ and $\mathbf{a}_{k+1}$, depending on each numerical integration algorithm.

2.3.2 Central difference method

The central difference method (CDM) is one of the most used numerical integration algorithms, mainly due to its explicit nature and simple implementation (Carrion and Spencer, 2007; Darby et al., 1999; Horiuchi et al., 1999; Horiuchi and Konno, 2001; Nakashima et al., 1992; Nakashima and Masaoka, 1999; Phillips and Spencer, Jr., 2013a; Shing et al., 1996; Wu et al., 2005).

Without loss of generality, let the EOM for a linear system be defined by:

$$\mathbf{M}\mathbf{a}_k + \mathbf{C}\mathbf{v}_k + \mathbf{K}\mathbf{u}_k = \mathbf{p}_k \quad (2.29)$$

where $\mathbf{v}_k \triangleq \dot{\mathbf{u}}_k$ and $\mathbf{a}_k \triangleq \ddot{\mathbf{u}}_k$ are the velocity and acceleration vectors at step "k", respectively. Then, the CDM considers the following approximations for velocity and acceleration vectors using finite differences:

$$\mathbf{v}_k = \frac{\mathbf{u}_{k+1} - \mathbf{u}_{k-1}}{2\Delta t} \tag{2.30}$$

$$\mathbf{a}_k = \frac{\mathbf{u}_{k+1} - 2\mathbf{u}_k + \mathbf{u}_{k-1}}{\Delta t^2} \tag{2.31}$$

Substituting the later in (2.29), the explicit solution of displacement $\mathbf{u}_{k+1}$ is obtained through the following algebraic equation:

$$\mathbf{u}_{k+1} = \left(\mathbf{M} + \frac{\Delta t}{2}\mathbf{C}\right)^{-1} \left[\Delta t^2 \mathbf{p}_k + \left(2\mathbf{M} - \Delta t^2 \mathbf{K}\right)\mathbf{u}_k - \left(\mathbf{M} - \frac{\Delta t}{2}\mathbf{C}\right)\mathbf{u}_{k-1}\right] \tag{2.32}$$

For the calculation of the first step, an estimation of $\mathbf{u}_{-1}$ is required. By combining (2.30) and (2.31), we obtain the following expression to initialize the CDM algorithm:

$$\mathbf{u}_{-1} = \mathbf{u}_0 - \Delta t \mathbf{v}_0 + \Delta t^2 \mathbf{a}_0 \tag{2.33}$$

where $\mathbf{u}_0$ and $\mathbf{v}_0$ are the initial conditions of displacement and velocity, respectively. In addition, the initial acceleration $\mathbf{a}_0$ can be determined through equilibrium:

$$\mathbf{a}_0 = \mathbf{M}^{-1}\left(\mathbf{p}_0 - \mathbf{C}\mathbf{v}_0 - \mathbf{K}\mathbf{u}_0\right) \tag{2.34}$$

The CDM algorithm is an explicit and conditionally stable scheme, with a criterion for integration time step Δt given by the equation $\omega_n \Delta t \leq 2$, where ω_n (rad/s) is the highest natural frequency of the structural system.

The CDM algorithm was originally proposed by Nakashima et al. (1992) for RTHS testing; but, in order to guarantee that the actuator will receive displacement commands without interruption, a staggered scheme was considered, where the integration time interval is $2\Delta t$ instead of Δt, while Δt corresponds to the sampling time of the digital controller for the actuator. Then, Nakashima and Masaoka (1999) proposed the CDM algorithm with an interpolation/extrapolation strategy to ensure a continuous movement of the actuator. Later, Wu et al. (2005) considered a modified version of the CDM, with an explicit forward difference formulation for velocity $\mathbf{v}_{k+1}$, although sacrificing accuracy and stability, thus only working for lightly damped experimental substructures.

2.3.3 Newmark-β and HHT-α methods

The Newmark family of integration methods is given by the following finite differences expressions:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \Delta t \mathbf{v}_k + \Delta t^2 \left[\left(\frac{1}{2} - \beta\right)\mathbf{a}_k + \beta \mathbf{a}_{k+1}\right] \tag{2.35}$$

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \Delta t \left[(1-\gamma)\mathbf{a}_k + \gamma \mathbf{a}_{k+1}\right] \tag{2.36}$$

These two expressions are combined with (2.29) for the linear case, to solve for accelerations $\mathbf{a}_{k+1}$ or displacements $\mathbf{u}_{k+1}$, depending on the corresponding implementation.

In terms of numerical stability, if the structural system is linear, and if condition $1/2 \leq \gamma \leq 2\beta$ is met, then the Newmark method is *unconditionally stable*, i.e., the method is stable regardless of the choice of time step Δt. But, if $\gamma < 1/2$, then the method is conditionally stable, which implies a restriction for possible values of Δt for numerical integration.

Also, there are a number of classical algorithms derived from the Newmark method family. For instance, if $\beta = 1/4$ and $\gamma = 1/2$, then the algorithm is the implicit unconditionally stable Newmark method, also known as the *constant average acceleration method (CAAM)*, or *trapezoidal rule*. Also, the choice of $\beta = 0$ and $\gamma = 1/2$ leads to the explicit CDM algorithm discussed previously. But, the explicit form of the Newmark method is better than the CDM in terms of its numerical conditioning (Shing, 2008).

On the other hand, HHT-α method is a generalization of the Newmark method, when the introduced parameter $\alpha = 0$. For a general case when $\alpha \neq 0$, the HHT-α method modifies the discrete linear EOM (2.29) into the expression:

$$\mathbf{M}\ddot{\mathbf{u}}_{k+1} = (1+\alpha)\left[\mathbf{p}_{k+1} - \mathbf{C}\dot{\mathbf{u}}_{k+1} - \mathbf{K}\mathbf{u}_{k+1}\right] - \alpha\left[\mathbf{p}_k - \mathbf{C}\dot{\mathbf{u}}_k - \mathbf{K}\mathbf{u}_k\right] \tag{2.37}$$

where the following conditions must be met for second-order accuracy and unconditionally stability:

$$\alpha \in \left[-\frac{1}{3}, 0\right] \tag{2.38}$$

$$\beta = \frac{(1-\alpha)^2}{4} \tag{2.39}$$

$$\gamma = \frac{1}{2} - \alpha \tag{2.40}$$

The incorporation of the α parameter is useful to improve numerical dissipation of undesired high-frequency responses from the numerical solution of MDOF systems, without degrading the accuracy as much.

For HS testing, Chang (2002) considered a modification of the Newmark method, that allows for explicit integration through the introduction of two weighting matrices, $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$, that are computed before the test starts, and are based on the initial elastic stiffness of the structure:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \Delta t \boldsymbol{\beta}_1 \mathbf{v}_k + \Delta t^2 \boldsymbol{\beta}_2 \mathbf{a}_k \tag{2.41}$$

This Newmark-Chang method is considered to be second-order accurate and unconditional stable for linear systems; but, a thorough study of its accuracy and stability for nonlinear systems is still lacking.

Furthermore, the HHT-α implicit method with a fixed number of iterations was proposed by Jung et al. (2007) and Mercan and Ricles (2009) for RTHS testing. In addition, this implementation considered a polynomial extrapolation-interpolation technique to ensure a

smooth motion of the actuator to impose the command displacements over the test specimen. However, a degradation on the test performance was observed, as a direct consequence of the increased numerical damping associated with the α parameter.

Later, Bonnet et al. (2008) studied different numerical integration schemes from the Newmark family for RTHS testing purposes. The schemes considered for this study were: Newmark explicit (CDM), Newmark-Chang explicit, Newmark implicit, and HHT-α implicit. It was concluded that the two implicit schemes were extremely slow, risking overrun situations on the real-time system even for a relatively small number of DOFs on the numerical substructure. On the other hand, the explicit schemes could manage the integration of larger numerical substructures in real-time. The Newmark-Chang method was demonstrated to be computationally efficient, and it was recommended by this study for situations when the numerical substructure does not meet the stability conditions of the Newmark explicit (CDM) scheme.

2.3.4 Operator-Splitting method

The Operator-Splitting (OS) method was developed to allow for two parts of the mesh (or operator) to be solved by implicit and explicit algorithms, simultaneously. This idea was developed specially for "stiff" subdomains, where implicit algorithms are required for stability purposes; while, "soft" subdomains are solved explicitly for efficiency.

The OS method consists in the combination of an implicit integrator, with a compatible explicit predictor-corrector scheme, while a modified EOM is taken into account for the synthesis of both algorithms.

First, the predictor-corrector variables are defined. The predictors of displacement and velocity, $\widetilde{\mathbf{u}}_{k+1}$ and $\widetilde{\mathbf{v}}_{k+1}$, are given by:

$$\widetilde{\mathbf{u}}_{k+1} = \mathbf{u}_k + \Delta t \mathbf{v}_k + \frac{\Delta t^2}{2}(1-2\beta)a_k \tag{2.42}$$

$$\widetilde{\mathbf{v}}_{k+1} = \mathbf{v}_k + \Delta t(1-\gamma)a_k \tag{2.43}$$

where the correctors for displacement and velocity, $\mathbf{u}_{k+1}$ and $\mathbf{v}_{k+1}$, are obtained by:

$$\mathbf{u}_{k+1} = \widetilde{\mathbf{u}}_{k+1} + \beta\Delta t^2 a_{k+1} \tag{2.44}$$

$$\mathbf{v}_{k+1} = \widetilde{\mathbf{v}}_{k+1} + \gamma\Delta t a_{k+1} \tag{2.45}$$

Then, the implicit scheme of the linear EOM is given by:

$$\mathbf{M}\mathbf{a}_{k+1} + \mathbf{C}\mathbf{v}_{k+1} + \mathbf{K}\mathbf{u}_{k+1} = \mathbf{p}_{k+1} \tag{2.46}$$

Next, the explicit predictor-corrector method is defined by:

$$\mathbf{M}\mathbf{a}_{k+1} + \mathbf{C}\widetilde{\mathbf{v}}_{k+1} + \mathbf{K}\widetilde{\mathbf{u}}_{k+1} = \mathbf{p}_{k+1} \tag{2.47}$$

Finally, by performing the following "splitting" procedure between implicit and explicit parts of the mesh, "I" and "E", respectively:

$$\mathbf{C} = \mathbf{C}^I + \mathbf{C}^E \tag{2.48}$$

$$\mathbf{K} = \mathbf{K}^I + \mathbf{K}^E \tag{2.49}$$

the modified EOM that combines implicit-explicit schemes with the predictor-corrector approach is given by:

$$\mathbf{M}^* \mathbf{a}_{k+1} = \mathbf{p}_{k+1} - \mathbf{C}\widetilde{\mathbf{v}}_{k+1} - \mathbf{K}\widetilde{\mathbf{u}}_{k+1} \tag{2.50}$$

$$\mathbf{M}^* = \mathbf{M} + \gamma \Delta t \mathbf{C}^I + \beta \Delta t^2 \mathbf{K}^I \tag{2.51}$$

This expression is exactly the same as the implicit Newmark method, when:

$$\begin{aligned} \mathbf{C}^I = \mathbf{C}, \quad \mathbf{K}^I = \mathbf{K} \\ \mathbf{C}^E = \mathbf{0}, \quad \mathbf{K}^E = \mathbf{0} \end{aligned} \tag{2.52}$$

Nakashima et al. (1990) first implemented the OS method for HS testing, with parameters $\beta = 1/4$ and $\gamma = 1/2$. The OS method considered an explicit prediction sub-step, where the displacement predictor is imposed onto the test specimen, to obtain the experimental coupling force. Also, an implicit correction sub-step is employed to solve for the nonlinear response of the numerical substructure. This OS method proved to be unconditionally stable for HS testing purposes.

Afterwards, Wu et al. (2006) extended the OS method for RTHS testing. The explicit scheme is responsible for the calculation of both displacement and velocity predictors. Then, both predictors are directly imposed onto the test specimen, through a polynomial extrapolation-interpolation scheme (Horiuchi et al., 1999). Unfortunately, the OS method cannot preserve the unconditional stability of for RTHS testing, and it has proved to offer sufficient stability guarantees only for softening type systems (e.g., degrading strength and/or stiffness).

Finally, Bonnet et al. (2008) considered the α-OS method for RTHS testing, a modification of the OS method to incorporate the α parameter from the HHT-α method discussed in the previous section. This α-OS method was intended to include numerical damping to reduce experimental errors due to uncompensated actuator dynamics.

2.3.5 CR/KR methods

Chen and Ricles (2008) developed the CR algorithm, which is an explicit, unconditionally stable algorithm, that was developed by studying the Newmark-β method from the perspective of digital control theory. This Also, this scheme has demonstrated similar accuracy to the implicit Newmark method, which is an additional feature of the method.

Inspired by the Newmark-Chang method (Chang, 2002), the displacement and velocity increments are obtained by including two weighting matrices, $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$, as shown in the following expressions:

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \Delta t \boldsymbol{\alpha}_1 \mathbf{a}_k \tag{2.53}$$

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \Delta t \mathbf{v}_k + \Delta t^2 \boldsymbol{\alpha}_2 \mathbf{a}_k \tag{2.54}$$

where $\mathbf{a}_k$ is solved through equilibrium equation (2.29), and the weighting matrices are equal to:

$$\boldsymbol{\alpha}_1 = \boldsymbol{\alpha}_2 = 4\left(4\mathbf{M} + 2\Delta t\mathbf{C} + \Delta t^2\mathbf{K}\right)^{-1}\mathbf{M} \tag{2.55}$$

The matrices $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ were determined by pole placement in discrete frequency domain (z-domain), such that the integration scheme satisfies unconditional stability, but only for linear and nonlinear structures with stiffness softening response. In addition, the matrices $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ are computed once, before the simulation starts.

Subsequently, Kolay and Ricles (2014) proposed the KR-α method, which is a family of explicit, unconditionally stable algorithms, based on the analysis of the HHT-α method through digital control theory. In this method, the parameter $\rho_\infty \in [0, 1]$ is introduced, which is defined as the *high-frequency spectral radius*. This parameter is useful for tuning of controllable numerical energy dissipation, that will dampen high-frequency spurious modes; while, low-frequency energy dissipation is minimized. This feature is particularly useful for numerical substructures with a large number of DOFs, when the time step for numerical integration is too large due to timing constraints in real-time systems. Then, the response increments are obtained by (2.53) and (2.54), with the following modification to the equilibrium equation:

$$\mathbf{M}\hat{\mathbf{a}}_{k+1} + \mathbf{C}\mathbf{v}_{k+1-\alpha_f} + \mathbf{K}\mathbf{u}_{k+1-\alpha_f} = \mathbf{p}_{k+1-\alpha_f} \tag{2.56}$$

where

$$\hat{\mathbf{a}}_{k+1} = (\mathbf{I} - \boldsymbol{\alpha}_3)\mathbf{a}_{k+1} + \boldsymbol{\alpha}_3\mathbf{a}_k \tag{2.57}$$

$$\mathbf{v}_{k+1-\alpha_f} = (1-\alpha_f)\mathbf{v}_{k+1} + \alpha_f\mathbf{v}_k \tag{2.58}$$

$$\mathbf{u}_{k+1-\alpha_f} = (1-\alpha_f)\mathbf{u}_{k+1} + \alpha_f\mathbf{u}_k \tag{2.59}$$

$$\mathbf{p}_{k+1-\alpha_f} = (1-\alpha_f)\mathbf{p}_{k+1} + \alpha_f\mathbf{p}_k \tag{2.60}$$

The weighting matrices are defined as:

$$\boldsymbol{\alpha}_1 = \boldsymbol{\alpha}^{-1}\mathbf{M} \tag{2.61}$$

$$\boldsymbol{\alpha}_2 = \left(\frac{1}{2} + \gamma\right)\boldsymbol{\alpha}_1 \tag{2.62}$$

$$\boldsymbol{\alpha}_3 = \boldsymbol{\alpha}^{-1}\left[\alpha_m\mathbf{M} + \alpha_f\gamma\Delta t\mathbf{C} + \alpha_f\beta\Delta t^2\mathbf{K}\right] \tag{2.63}$$

$$\boldsymbol{\alpha} = \left(\mathbf{M} + \gamma\Delta t\mathbf{C} + \beta\Delta t^2\mathbf{K}\right) \tag{2.64}$$

and

$$\alpha_m = \frac{2\rho_\infty - 1}{\rho_\infty + 1}, \quad \alpha_f = \frac{\rho_\infty}{\rho_\infty + 1} \tag{2.65}$$

$$\gamma = \frac{1}{2} - \alpha_m + \alpha_f, \quad \beta = \frac{1}{4}(1 - \alpha_m + \alpha_f)^2 \tag{2.66}$$

The special case when $\rho_\infty = 1$ yields the CR algorithm, hence no numerical energy dissipation is introduced. On the other hand, when $\rho_\infty = 0$, maximum numerical energy dissipation is considered. Similarly to the CR algorithm, the KR-α method is explicit and unconditionally stable, only for linear or nonlinear structures with stiffness softening response. Later, Kolay et al. (2014) implemented the KR-α method for an RTHS test, and conducted a study on the determination of optimal values for ρ_∞. Finally, Kolay and Ricles (2017) proposed a modification to the KR-α method, to mitigate overshoot for high-frequency modes, but at the expense of period elongation errors and numerical dissipation in the low-frequency regime.

2.3.6 Runge-Kutta methods

Let the MDOF EOM be defined by a system of nonlinear ODEs:

$$\dot{\mathbf{x}}_k = \mathbf{f}\left(t_k, \mathbf{x}_k, \mathbf{p}_k\right) \tag{2.67}$$

The Runge-Kutta methods are single-step integration schemes that replace higher-order derivatives by finite difference approximations based on values of the nonlinear function $\mathbf{f}(\cdot)$ at points between t_k and t_{k+1}. A well-known Runge-Kutta method is the fourth-order Runge-Kutta (RK4) explicit scheme:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \frac{\Delta t}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right) \tag{2.68}$$

where

$$\mathbf{k}_1 = \mathbf{f}\left(t_k, \mathbf{x}_k, \mathbf{p}_k\right) \tag{2.69}$$
$$\mathbf{k}_2 = \mathbf{f}\left(t_k + \Delta t/2, \mathbf{x}_k + (\Delta t/2)\mathbf{k}_1, \mathbf{p}_k\right) \tag{2.70}$$
$$\mathbf{k}_3 = \mathbf{f}\left(t_k + \Delta t/2, \mathbf{x}_k + (\Delta t/2)\mathbf{k}_2, \mathbf{p}_k\right) \tag{2.71}$$
$$\mathbf{k}_4 = \mathbf{f}\left(t_k + \Delta t, \mathbf{x}_k + \Delta t\mathbf{k}_3, \mathbf{p}_k\right) \tag{2.72}$$

In particular, the Runge-Kutta methods have special attributes compared to methods previously examined. For example, the method is self-starting at the beginning of the integration, i.e., the calculation of $\mathbf{x}_1$ only depends on the initial condition $\mathbf{x}_0$. Also, it allows for variable time-step during the integration. And the fact that is easy to implement, and that is readily available in many mathematical software packages, makes it a very popular numerical integration technique for nonlinear ODEs. But, the Runge-Kutta method is not very efficient for "stiff" ODE problems, or when highly accurate solutions are required.

In the context of RTHS testing, Carrion et al. (2009) and Phillips and Spencer, Jr. (2012) considered the explicit RK4 method, with a sufficiently small fixed time step $\Delta t = 0.005$ s (i.e., 2000 Hz sampling rate) for improved stability margins. Additionally, Li et al. (2017) considered the RK4 scheme for numerical integration of an RTHS test with geographically-distributed substructures, where the sampling rate of $1,024$ Hz was chosen for numerical integration.

Furthermore, Ou et al. (2015b) looked further on the classical Runge-Kutta scheme and proposed a modified Runge-Kutta (MRK) method for RTHS that is aimed at providing better numerical stability margins. The MRK method is a model-based predictor-corrector scheme, where the experimental coupling force is predicted before measurement takes place, in order to compensate for undesired experimental delays during the numerical integration of RTHS tests. The MRK scheme consists of three main steps: (1) pseudo response calculation; (2) coupling force prediction, based on identified initial stiffness and damping of the test specimen; and (3) response correction. Nonetheless, it was observed from experimental results that situations of lightly damped structures and test specimens with hardening effects and/or underestimated initial stiffness, may affect the stability of this numerical scheme.

2.3.7 Rosenbrock-based method

The Rosenbrock method is related to implicit Runge-Kutta methods and is classified as L-stable, a property that makes this method generally good to solve stiff ODE problems. Let the solution to (2.67) be given by an s-stage Rosenbrock method:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \sum_{i=1}^{s} b_i \mathbf{k}_i \tag{2.73}$$
$$[\mathbf{I} - \gamma \Delta t \mathbf{J}]\mathbf{k}_i = \mathbf{f}\left(t_k + \alpha_i \Delta t, \mathbf{x}_k + \sum_{j=1}^{i-1} \alpha_{ij}\mathbf{k}_j\right)\Delta t + \mathbf{J}\sum_{j=1}^{i-1} \gamma_{ij}\mathbf{k}_j \Delta t \tag{2.74}$$

where $\alpha_i = \sum_{j=1}^{i-1} \alpha_{ij}$, γ_{ij} and b_i are predefined parameters, and $\mathbf{J}$ is the Jacobian matrix:

$$\mathbf{J} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \tag{2.75}$$

Bursi et al. (2008) proposed a variant of the Rosenbrock method for RTHS testing. Two L-stable real-time (LSRT) algorithms were developed from the Rosenbrock scheme: two-stage (LSRT2) and three-stage (LSRT3). In addition, these algorithms were developed to avoid the burden of Jacobian matrix update; the Jacobian matrix is evaluated only once at the start of the algorithm and kept constant for the entire simulation. Also, the parameters γ, γ_{ij}, α_i, α_{ij}, and b_i, must meet specific conditions for unconditional stability. In addition, LSTR2 and LSTR3 are classified as second-order and third-order accurate, respectively. The LSRT algorithms were implemented and verified through RTHS tests of SDOF and MDOF systems with linear and nonlinear test specimens (Lamarche et al., 2009). Also, other studies focused on the examination of numerical performance for stiff numerical substructures (Bursi et al., 2010), and the extension for nonlinear solutions with subcycling (mixed time step integration) strategies (Bursi et al., 2011).

2.4 Compensation of actuator dynamics

Due to actuator dynamics, there exists an inherent delay in the response of the actuator with respect to the commanded motion. This effect could not only degrade the accuracy of the RTHS test (i.e. measured response does not follow the expected response), but it could also result in an unstable response, which could have negative consequences on the experimental equipment.

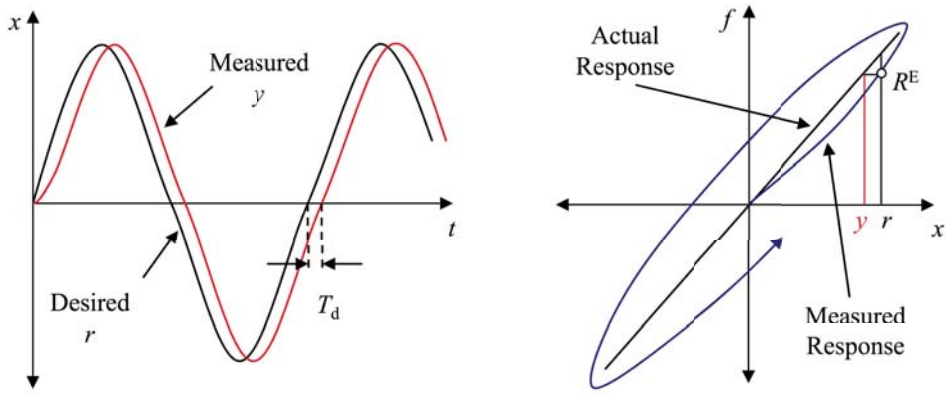


Figure 2.5: Effects of time delay on RTHS with linear physical specimen (Phillips and Spencer, Jr., 2012)

Horiuchi et al. (1996) observed the effects on the stability and accuracy of RTHS tests caused by the dynamic response of servo-hydraulic actuators. The dynamics of a servo-hydraulic actuator can be idealized as a *pure time delay* T_d, as shown in Figure 2.5. Therefore, the measured and target (desired) displacements of the actuator are separated

in time-domain by T_d. It was demonstrated that this delay causes an instability problem during the execution of the RTHS test. The reason behind this issue was that the delay increases the total energy supplied to the RTHS experiment, which is equivalent to the case where *negative damping* is artificially introduced into the numerical substructure. If the negative damping is greater or equal to the numerical damping of the system, then an unstable response occurs.

The solution devised to fix this problem was to include a *delay compensation method* in RTHS testing. A "prediction" of the actuator motion, by means of an n-th order polynomial extrapolation, is commanded to the servo-hydraulic system in order to balance the time delays. Unfortunately, the compensation method failed to achieve its objective when very stiff specimens are tested, or when the numerical substructure is very flexible (e.g. tall buildings).

Henceforth, compensation of actuator dynamics has become a key challenge for RTHS development. The formulations reported in the literature can be classified in time-domain or frequency-domain. The former is based on the numerical analysis of the equations of motion and it considers extrapolation relations to "predict" command signals that will compensate for time delays; while, the latter methods were derived from the adoption of tools from modern control theory. Either way, any compensator design should allow for stable and accurate RTHS test, that ideally could provide some robustness guarantees of the controller design (i.e., controller performes well under model uncertainty or random disturbance). The following is a brief list of significant contributions on this field.

2.4.1 Polynomial extrapolation

Horiuchi et al. (1996) proposed a delay compensation based on polynomial extrapolation. Let $r \triangleq u^{\text{target}}$, $u \triangleq u^{\text{cmd}}$, and $y \triangleq u^{\text{meas}}$, be the target, command, and measured displacements, respectively. Also, let the sampling time Δt, and discrete time $t_k = k\Delta t, \forall k \in \{0, 1, 2, \ldots, t_f/\Delta t\}$.

Assume that the actuator dynamics can be idealized by a constant pure time delay T_d. Hence, a key assumption is that the measured signal is delayed with respect to the target displacement, i.e., $y(t_k) = r(t_k - T_d)$. Then, the discrete-time command signal $u_k = u(t_k)$ that compensates for this delay effect is obtained by:

$$u_k = \sum_{i=0}^{n} a_i r_{i,k} \tag{2.76}$$

where $r_{i,k} = r(t_k - iT_d)$ is the discrete-time target signal by adding shifts of T_d by integer values of i. The polynomial coefficients a_i were determined using the Lagrange basis functions. A third- and fourth-order polynomial extrapolation is commonly used for delay compensation in RTHS:

$$u_k = 4r_{0,k} - 6r_{1,k} + 4r_{2,k} - r_{3,k} \qquad \text{(third-order)} \tag{2.77}$$

$$u_k = 5r_{0,k} - 10r_{1,k} + 10r_{2,k} - 5r_{3,k} + r_{4,k} \qquad \text{(fourth-order)} \tag{2.78}$$

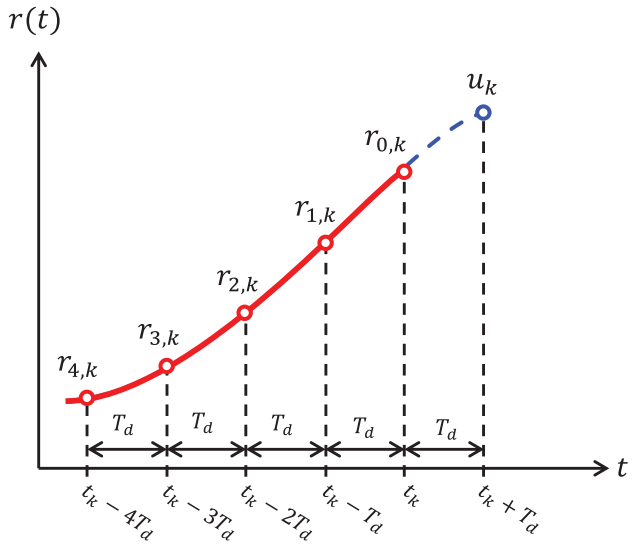


Figure 2.6: Delay compensation through polynomial extrapolation

The discrete-time command signal can be understood as a "prediction" of the target signal shifted T_d once into the future, i.e., $u_k \triangleq r(t_k + T_d)$. After commanding this signal to the actuator, the discrete-time measured and target signals will be approximately equal, $y_k \approx r_k$, thus compensating for the actuator dynamics.

Unfortunately, perfect compensation is not possible, mainly due to the fact that actuators have more complex dynamics that cannot be represented using only a pure time delay model. In addition, this technique is only suitable when the pure time delay is small compared to the fundamental period of the structure (i.e. $T_d << T_n$). Also, accurate estimation of time delay T_d is a requirement to implement this compensation method.

Other alternatives to this polynomial extrapolation method have been reported in the literature, such as the linear acceleration extrapolation (Horiuchi and Konno, 2001), which is developed to consider explicit predictions of target displacement, velocity, and acceleration, for the determination of the controller signal.

2.4.2 Phase-lead compensation

The phase-lead compensator (Jung et al., 2007; Zhao et al., 2003) is developed from the perspective of classical control theory. The actuator dynamics can be analyzed in frequency-domain, where in general the transfer function $G(s)$ shows a phase shift that could be assumed to be linear. This phase shift can be cancelled out by adding a lead compensator $K(s)$. Thus, obtaining an approximate zero-phase response for the open-loop transfer function $G(s)K(s)$ at a given frequency bandwidth.

Let T_d be the pure time delay of the actuator. Then, the transfer function of the phase-lead compensator is given by:

$$K(s) = \frac{T_d s + 1}{\alpha T_d s + 1}, \quad \alpha < 1 \tag{2.79}$$

where α is the phase-lead constant, $s = j\omega$ is the Laplace variable, ω is the natural frequency (rad/s), and $j = \sqrt{-1}$ is the complex number. In general, this compensator provides a maximum phase increase that depends only on the choice of α. For example, for $\alpha = 0.1$, the phase-lead compensator can contribute a maximum of 60° to the phase angle at a frequency equal to ωT_d. Similarly, this method requires accurate estimation of the time delay T_d through parameter estimation.

2.4.3 Inverse compensation

Bonnet et al. (2007) proposed an inverse-based feedforward compensator. This controller is designed to compensate the dynamics of a simple first-order actuator model:

$$G_m(s) = \frac{b_m}{s + a_m} \tag{2.80}$$

where $G_m(s)$ is transfer function of the actuator model. Then, to compensate for the actuator dynamics, an inverse model is considered for the design of feedforward compensator:

$$K(s) = G_m^{-1}(s) = \frac{s + a_m}{b_m} \tag{2.81}$$

where the control signal is $u(s) = K(s)r(s)$. Then, the tracking error is given by:

$$\begin{aligned} e(s) &= y(s) - r(s) \\ &= G(s)u(s) - r(s) \\ &= \left[G(s)G_m^{-1}(s) - 1\right] r(s) \approx 0 \end{aligned} \tag{2.82}$$

This idea was further extended by Chen and Ricles (2009) for discrete control systems:

$$K(z) = \frac{\alpha z - (\alpha - 1)}{z}, \quad \alpha \geq 1 \tag{2.83}$$

where α was determined such that the time delay satisfies the following relationship:

$$T_d = (\alpha - 1)\Delta t \tag{2.84}$$

and Δt is the sampling time. Both algorithms require an accurate estimation of the time delay parameter T_d prior to conducting the RTHS test. For example, Chen and Ricles (2009) proposed values of $\Delta t = 1/1024$ s and $\alpha = 36$ for their experimental implementation, and open loop system (i.e., $L(s) = G(s)K(s)$) showed almost unit-gain, zero-phase for frequency range 0 – 5 Hz.

Unfortunately, the time delay of an actuator system is not constant along the frequency space, making the optimal estimation of time delays for robust compensation a laborious task.

2.4.4 Adaptive compensation

Adaptive compensation was introduced to increase the robustness of RTHS test. Compared to previous compensator solutions, this algorithm provides an on-line estimation and adaption of compensator parameters, in order to adjust to model uncertainty and/or nonlinear response of the experimental system. Indeed, this solution offers an attractive solution for the study of nonlinear or very stiff experimental substructures, compared to schemes with fixed parameters.

Darby et al. (2002) proposed a polynomial extrapolation compensator, where the time delay τ was adapted using the following on-line estimator:

$$\tau_k = \tau_{k-1} + C_p \tanh\left[C_v\left(\frac{r_k - r_{k-1}}{\Delta t}\right)\right](r_{k-1} - y_{k-1}) \tag{2.85}$$

where r_k is the k-th step of discrete-time target displacement; while, C_p and C_v are adaptation gains. These gains are tuned such that the estimator converges sufficiently fast and at the same time does not become unstable.

Afterwards, Ahmadizadeh et al. (2008) proposed polynomial extrapolation compensation with a modified on-line time delay estimator through linear fits:

$$\tau_k = \tau_{k-1} + 2G\Delta t\left(\frac{r_k^{\text{avg}} - y_k^{\text{avg}}}{y_k - y_{k-2}}\right) \tag{2.86}$$

where

$$r_k^{\text{avg}} = \frac{r_k + r_{k-1} + r_{k-2}}{3}, \quad y_k^{\text{avg}} = \frac{y_k + y_{k-1} + y_{k-2}}{3} \tag{2.87}$$

and G is the learning gain, chosen equal to $G = 0.1$. This implementation proved to converge faster and with reduced oscillations, compared to Darby et al. (2002).

Then, Wallace et al. (2005) proposed an adaptive forward prediction (AFP) compensator, which considered a polynomial extrapolation compensator with least-squares polynomial fitting, instead of using Lagrange basis functions. The controller signal is obtained as follows:

$$u_k = k_a \sum_{i=0}^{n} \theta_i P^i \tag{2.88}$$

where k_a is a gain to remove amplitude errors (i.e., undershoot/overshoot problems); θ_i are the adaptive polynomial coefficients; and P is the number of time steps to be predicted forward, which does not need to be an integer multiple of sampling time step Δt. Then, the adaptation of the polynomial coefficients is obtained through an on-line least-squares estimation:

$$\boldsymbol{\theta} = \mathbf{X}_P\left[\left(\mathbf{X}_k^T\mathbf{X}_k\right)^{-1}\mathbf{X}_k^T\right] \tag{2.89}$$

$$\mathbf{X}_k = \begin{bmatrix} 1 & r_k & r_k^2 & \dots & r_k^N \\ 1 & r_{k-1} & r_{k-1}^2 & \dots & r_{k-1}^N \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & r_{k-n} & r_{k-n}^2 & \dots & r_{k-n}^N \end{bmatrix} \tag{2.90}$$

where $\mathbf{X}_P = [1, P\Delta t, \dots, P^N \Delta t^N]$ is the forward prediction vector; $\boldsymbol{\theta} = [\theta_0, \theta_1, \dots, \theta_n]$ is the adaptive polynomial vector; n is the number of target points, and N is the order of the polynomial fit. In particular, values of $n = 3$ and $N = 2$ were chosen for application to an RTHS test.

Subsequently, Lim et al. (2007) extended the idea of inverse compensation with a modified minimal controller synthesis (MCS) algorithm. The adaptive controller is based on a reference model of the actuator, given by:

$$G_m(s) = \frac{b_m}{s + a_m} \tag{2.91}$$

Then, the control signal was obtained from a feedforward-feedback controller:

$$u(t) = K_{\text{ff}}(t) \left(\frac{s + a_m}{b_m} \right) r(t) + K_{\text{fb}}(t) e(t) \tag{2.92}$$

The purpose of the adaptive controller is to minimize the tracking error $e(t) = r(t) - y(t)$. Hence, the adaptation of feedforward $K_{\text{ff}}(t)$ and feedback $K_{\text{fb}}(t)$ gains are given by the following expressions:

$$K_{\text{ff}}(t) = K_{\text{ff}}(0) + \beta e(t) r'(t) + \alpha \int_0^t e(\tau) r'(\tau) d\tau \tag{2.93}$$

$$K_{\text{fb}}(t) = K_{\text{fb}}(0) + \beta e(t) y(t) + \alpha \int_0^t e(\tau) y(\tau) d\tau \tag{2.94}$$

where α and β are adaptive weights, and the initial conditions for feedforward and feedback gains are chosen as $K_{\text{ff}}(0) = 1$ and $K_{\text{fb}}(0) = 0$, respectively. This framework was also validated experimentally by Bonnet et al. (2007) in a twin-actuator RTHS setup.

Following these steps, Chen and Ricles (2010) considered an adaptive inverse compensator:

$$K(z) = \frac{(\alpha_{es} + \Delta\alpha) z - (\alpha_{es} + \Delta\alpha - 1)}{z} \tag{2.95}$$

where α_{es} is the estimated actuator delay, and $\Delta\alpha$ is an evolutionary variable that is determined by the following expression:

$$\Delta\alpha(t) = k_p \text{TI}(t) + k_i \int_0^t \text{TI}(\tau) d\tau \tag{2.96}$$

where $\text{TI}(t)$ is a tracking error indicator; k_p and k_i are proportional and integral gains for

the adaptive law. Also, the initial condition of the adaptive law is chosen as $\Delta\alpha(0) = 0$.

Similarly, Chae et al. (2013a) proposed an adaptive time series (ATS) compensation, where the discrete-time controller displacement at step k is obtained by:

$$u_k = a_{0k} r_k + a_{1k} \dot{r}_k + a_{2k} \ddot{r}_k \tag{2.97}$$

where the target velocity and acceleration are approximated using finite differences:

$$\dot{r}_k = \frac{r_k - r_{k-1}}{\Delta t}, \quad \ddot{r}_k = \frac{r_k - 2r_{k-1} + r_{k-2}}{\Delta t^2} \tag{2.98}$$

and the adaptive parameters $\mathbf{A} = [a_{0k}, a_{1k}, a_{2k}]^T$ are obtained using an on-line least-squares estimation approach:

$$\mathbf{A} = \left(\mathbf{X}_m^T \mathbf{X}_m\right)^{-1} \mathbf{X}_m^T \mathbf{U}_c \tag{2.99}$$

where $\mathbf{X}_m = [\mathbf{x}_m, \dot{\mathbf{x}}_m, \ddot{\mathbf{x}}_m]$, is a matrix of observed responses, where $\mathbf{x}_m = [y_{k-1}, y_{k-2}, \ldots, y_{k-q}]^T$; $\mathbf{U}_c = [r_{k-1}, r_{k-2}, \ldots, r_{k-q}]^T$ is a vector of observed targets; and q is the number of target points considered for polynomial fitting.

Later, Chen et al. (2014) proposed an adaptive, discrete, second-order, phase-lead compensator:

$$K(z) = \frac{[W_1 + (W_1 + W_2 + 1)T_d]\, z^2 + [W_2 - (W_1 + W_2 + 1)T_d]\, z + 1}{W_1 z^2 + W_2 z + 1} \tag{2.100}$$

where W_1 and W_2 are weighting parameters, and $z = e^{j\omega}$ is the complex number in z domain. In this case, the time delay T_d is modified using a gradient adaptive law.

Finally, Chen et al. (2015) proposed self-tuning model-based compensator approach, where the feedforward controller

$$K_{\text{ff}}(s) = a_3 s^3 + a_2 s^2 + a_1 s + a_0 \tag{2.101}$$

have parameters a_i that are modified using a gradient adaptive control law:

$$\dot{\boldsymbol{\theta}} = \boldsymbol{\Gamma} \varepsilon \mathbf{w} \tag{2.102}$$

$$\varepsilon = \frac{z - \boldsymbol{\theta}^T \mathbf{w}}{m_s^2} \tag{2.103}$$

where $\boldsymbol{\theta} = [a_3, a_2, a_1, a_0]^T$ is the estimated parameter vector; z and $\mathbf{w}$ are known signals; ε is the estimation error; $m_s^2 = 1 + \alpha \mathbf{w}^T \mathbf{w}$ is a normalization factor, with $\alpha > 0$; and $\boldsymbol{\Gamma}$ is the adaptive gain matrix. In addition, parameter projection was considered to enforce constraints over the adaptive gains $\boldsymbol{\theta}$, such that Routh's stability criterion was satisfied.

2.4.5 Model-based compensation

Previous compensator solutions require an estimate of time delay, which is taken as a constant for these classical implementations. But, a pure time delay is an approximation that does not consider more fundamental dynamics of an actuator. For example, actuator systems have a frequency response with nonlinear phase, which is equivalent to frequency-dependent time delays. Therefore, relying on one constant time delay estimation may not be suitable for compensation of complex RTHS setups. Moreover, the test specimen and actuator system have a dynamic interaction (Dyke et al., 1995), which is an effect that may not be possible to capture with constant pure time delays.

Hence, Carrion and Spencer (2007) proposed a model-based compensation scheme, based on the advances of modern control theory. The control algorithms are designed by taking into account a model of the experimental system, for improved performance while conducting RTHS tests. The original approach considered a linearized model of the experimental system:

$$G(s) = \frac{k}{\prod_{i=1}^{n}(s - p_i)} \tag{2.104}$$

where k is the gain, and p_i are the poles of the transfer function. Then, an inverse-based feedforward controller is obtained, with a unit-gain lowpass filter to allow the resulting transfer function to be proper and causal:

$$K_{\text{ff}}(s) = \alpha^n \frac{\prod_{i=1}^{n}(s - p_i)}{\prod_{i=1}^{n}(s - \alpha p_i)} \tag{2.105}$$

Moreover, a feedback term is included to reduce the effect of model uncertainty in the design of the feedforward compensator. A proportional feedback controller with constant gain $K_{\text{fb}}(s) = K_{\text{fb}}$ was considered in this case. For stability purposes, a root locus plot was required to determine the maximum proportional gain to satisfy both performance and stability specifications.

Later, Phillips and Spencer, Jr. (2012) extended this idea and proposed a discrete-time feedforward compensator:

$$u_k^{\text{ff}} = a_0 r_k + a_1 \dot{r}_k + a_2 \ddot{r}_k + a_3 \dddot{r}_k \tag{2.106}$$

where higher order derivatives $\dot{r}_k$, $\ddot{r}_k$, and $\dddot{r}_k$, are approximated using finite difference method for causal implementation in a real-time system. Furthermore, the feedback controller was replaced by an LQG optimal control algorithm, which is also based on a nominal model of the actuator dynamics.

Afterwards, Gao et al. (2013) studied the incorporation of robust control strategies into the model-based compensation approach. This study recognized that the compensation schemes must provide sufficient guarantees of robustness against model uncertainties and external disturbances. H_∞ loop shaping design was proposed to satisfy both performance and robustness specifications. Another implementation was the robust integrated actuator control (RIAC), based on feedforward controller with H_∞ loop shaping feedback, and a Kalman filter for noise reduction (Ou et al., 2015a).

2.5 Compensation of coupling force measurements

A fundamental aspect of HS testing is the accurate measurement of coupling forces from the experimental substructure, to avoid error propagation when these values are inserted back to the numerical substructure. Unfortunately, force measurements in experimental tests contain significant noise. Usually, high-frequency noise can be filtered by using lowpass filters, but at the expense of introducing additional dynamics to the experimental system, something that is not beneficial for RTHS test purposes.

Therefore, methods to compensate for errors in coupling force measurements have been considered in the literature. Most methods were focused on force compensation by using an estimation of the tangent stiffness of the test specimen. Then, the compensated coupling force is given by

$$\mathbf{g}^E_{\text{corr}} = \mathbf{g}^E_{\text{meas}} + \hat{\mathbf{K}}^E(\mathbf{u}_{\text{meas}} - \mathbf{u}_{\text{target}}) \tag{2.107}$$

where $\mathbf{g}^E_{\text{corr}}$ is the corrected (compensated) coupling force; $\mathbf{g}^E_{\text{meas}}$ is the measured (uncompensated) coupling force; $\hat{\mathbf{K}}^E$ is the estimated tangent stiffness of the test specimen; $\mathbf{u}_{\text{meas}}$ and $\mathbf{u}_{\text{target}}$ are the measured and target displacements, respectively. Hence, the amount of compensation is proportional to the displacement tracking error.

In the context of HS tests, Nakashima et al. (1990) proposed corrections of the coupling force measurements by using a previously identified initial elastic stiffness of the test specimen. Later, Thewalt and Roman (1994) proposed an on-line tangent stiffness estimation for force compensation, based on the BFGS algorithm typically used in nonlinear optimization problems to estimate Hessian matrices. Also, Hung and El-Tawil (2009) proposed an updated tangent stiffness, which is estimated using a least-squares method, and applied with particular rules depending on loading/unloading trajectories of the test specimen.

Force compensation in RTHS tests has only been considered. Carrion and Spencer (2007) proposed the use of a Kalman filter, which was designed using an estimated initial stiffness of the test specimen. In addition, an extended Kalman filter (EKF) was considered for force compensation of nonlinear specimens, although the method was found to be sensitive to the degree of inelastic behavior of the specimen and the choice filter parameters. Later, Ahmadizadeh et al. (2008) proposed a polynomial extrapolation for force compensation. But, the proposed method can result in complex-valued restoring forces, something that does not have a physical meaning. Afterwards, Chen and Tsai (2013) proposed a moving average tangent stiffness estimation (MATSE). The accuracy of this stiffness estimator was highly dependent on the choice of sample size N used in the moving window: if N was too small, then the estimated stiffness is highly sensitive to sensor noise; on the other hand, if N was too large, the estimation is not able to represent the sudden changes of the specimen stiffness.

The main challenges in force compensation of the experimental substructure are related to obtaining accurate and stable predictions of tangent stiffness, under the presence of sensor noise and nonlinear hysteretic response of the test specimen. Indeed, compensation of actuator dynamics have a great impact on force compensation. But, if the test specimen is stiff, even the ideal case where tracking errors are close to zero could produce some undesired

results in terms of coupling force errors.

2.6 Error quantification in hybrid simulation

Experimental errors in hybrid simulation can be classified in two categories: (i) *tracking errors*, which are associated with actuator dynamics, and cause de-synchronization of experimental and numerical substructures; and (ii) *communication errors*, which are associated to problems of coordination between geographically-distributed substructures. The evaluation of how experimental errors are propagated while conducting the tests is essential to avoid any problems of accuracy and stability.

Therefore, assessment measures have been proposed in order to highlight any experimental inconsistencies for hybrid simulation tests in general. These assessment measures can be classified as either local or global response assessment indices (Christenson et al., 2014). The former focuses only on the performance of the experimental system, where actuator tracking errors are primarily monitored to ensure accurate and stable response of the RTHS tests. While, the later considers the full substructuring problem, which could be useful to identify problems on either numerical or experimental substructures.

2.6.1 Local assessment measures

Wallace et al. (2005) proposed the use of a synchronization subspace plot (Ashwin, 1998) for online visual evaluation of tracking errors. The synchronization subspace plot (SSP) consists of plotting the experimentally measured response vs. target displacement, as shown in Figure 2.7. Perfect tracking occurs when the data follows a straight line with 1:1 slope (angle of 45°). Also, changes on the slope are associated to undershoot/overshoot of the response, and hysteresis (not straight lines) are associated to lag/lead of the response.

Mercan and Ricles (2009) adopted the synchronization subspace plot to propose two performance indices: a tracking indicator (TI) and amplitude indicator (θ_{pc}). The former is used to determine if any time lead or time lag error is present in the experiment; while, the latter is used to assess any amplitude error in the actuator measured displacement during real-time execution. These indices were obtained by numerically computing the enclosed area of the SSP, and the major axis inclination through principal component analysis (PCA), respectively. For perfect tracking, values of $TI = 0$ and $\theta_{pc} = 45°$ are expected.

Similarly, Hessabi and Mercan (2012) proposed a phase and amplitude error indices (PAEI) that were derived as coefficients of ideal ellipsoids that match the shape of the SSP. This allowed improved computation compared to previous tracking and amplitude indices.

Afterwards, Guo et al. (2014) proposed the frequency evaluation index (FEI), which is a weighted sum of the ratio between the auto-power spectral densities $S_{yy}(\omega)$ and $S_{rr}(\omega)$, associated to response $y(t)$ and target $r(t)$ displacements, respectively. Then, generalized amplitude and phase measures are obtained:

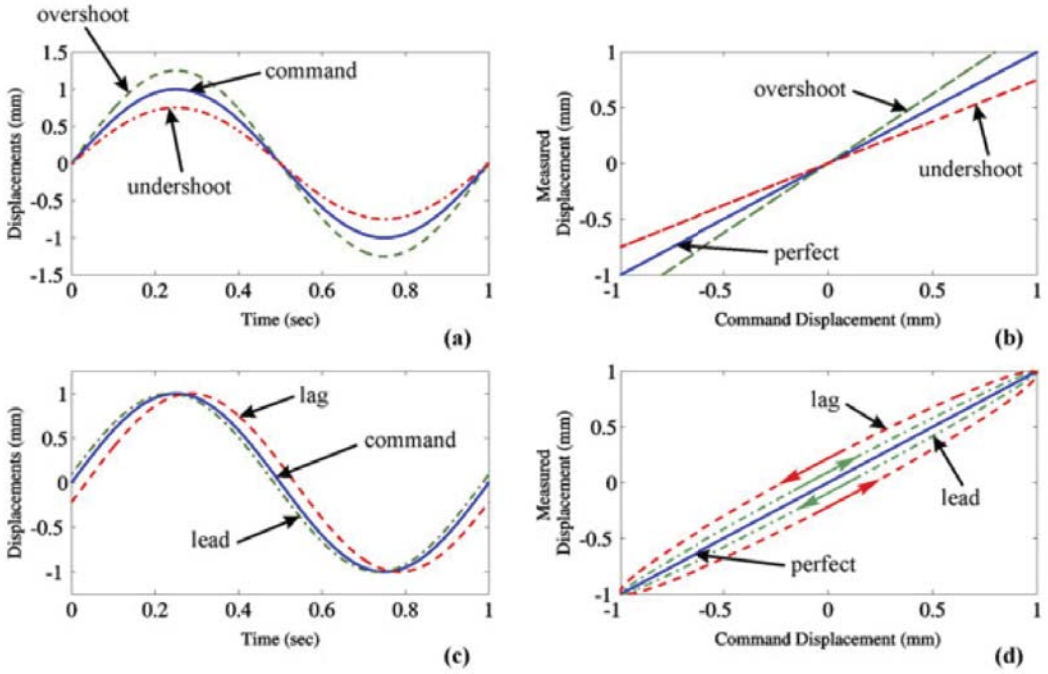


Figure 2.7: Synchronization subspace plot (SSP) of simulated response with amplitude and tracking errors (Hessabi and Mercan, 2012)

$$A = \|\text{FEI}\| \tag{2.108}$$

$$\varphi = \arctan\left[\frac{\text{Im(FEI)}}{\text{Re(FEI)}}\right] \tag{2.109}$$

where in the case of perfect tracking, $A = 1$ and $\varphi = 0$.

Subsequently, Mosqueda et al. (2007a,b) proposed a hybrid simulation error monitor (HSEM), that was derived from energy principles. A normalized energy error estimator is determined by estimating the energy that was introduced to the hybrid simulation test by systematic errors on the experimental setup:

$$\text{HSEM}_k = \frac{E_k^{\text{error}}}{E^{\text{input}} + E^{\text{strain}}} \tag{2.110}$$

$$E^{\text{input}} = \int \mathbf{p}^T d\mathbf{u} \tag{2.111}$$

$$E^{\text{strain}} = \frac{1}{2}\mathbf{u}_y^T \mathbf{K} \mathbf{u}_y \tag{2.112}$$

where E_k^{error} is the energy error from the experimental setup at time step k; E^{input} is the total input (excitation) energy; and E^{strain} is the maximum recoverable strain energy, which is estimated for elasto-plastic materials with yield displacement $\mathbf{u}_y$. Also, the energy error

is calculated as follows:

$$E_{k+1}^{\text{error}} = E_k^{\text{error}} + \left(E_k^{BE} - E_k^{E}\right) \tag{2.113}$$

where E_k^{BE} is the best estimate of the energy in the experimental substructure; and E_k^{E} is the energy in the experimental substructure observed by the numerical substructure. Both quantities are defined in actuator coordinates, where the kinematic transformations $\mathbf{q} = \mathbf{T}\mathbf{u}$ and $\mathbf{r} = \mathbf{T}^T\mathbf{t}$ are considered. In this case, $\mathbf{q}$ is the displacement in actuator coordinates, $\mathbf{t}$ is the force in actuator coordinates, and $\mathbf{T}$ is the kinematic transformation matrix.

$$E_k^{BE} = \frac{1}{2}(\mathbf{t}_{k-1}^{\text{meas}} + \mathbf{t}_k^{\text{meas}})^T(\mathbf{q}_{k-1}^{\text{meas}} - \mathbf{q}_k^{\text{meas}}) \tag{2.114}$$

$$E_k^{E} = \frac{1}{2}(\mathbf{t}_{k-1}^{\text{meas}} + \mathbf{t}_k^{\text{meas}})^T(\mathbf{q}_{k-1}^{\text{target}} - \mathbf{q}_k^{\text{target}}) \tag{2.115}$$

These indices provide a way to track and control individual actuators in order to improve overall hybrid simulation results. Nevertheless, it was reported that the relationship between the growth of hybrid simulation accuracy measures and the values of HSEM are influenced by the structural model and ground motion considered. Therefore, numerical studies to assess the effect of simulation error measures are required prior to the experimental testing, in order to determine bounds for the HSEM index.

Other measures local assessment measures have been proposed in the literature. For example, normalized root mean square error (NRMSE) offers a good indication of tracking errors for a particular boundary DOF. Assuming uniaxial loading in RTHS test:

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(u_k^{\text{meas}} - u_k^{\text{target}}\right)^2}}{m_N} \tag{2.116}$$

where m_N is a normalization factor, that can be chosen as one of the following options:

1. $m_N = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(u_k^{\text{target}}\right)^2}$
2. $m_N = \max\left(|u_k^{\text{target}}|\right)$
3. $m_N = \left|\max\left(u_k^{\text{target}}\right) - \min\left(u_k^{\text{target}}\right)\right|$

Also, the cross-correlation function between target and measured signals and the transfer function of the experimental system can be considered for the same purposes.

2.6.2 Global assessment measures

Ahmadizadeh and Mosqueda (2009) proposed an extension of the HSEM index to incorporate both numerical and experimental errors global assessment measured for HS tests. A non-dimensional energy error index (EEI) was proposed based on overall unbalanced energy:

$$\text{EEI} = \frac{E^{\text{error}}}{E^{\text{input}} + E^{\text{strain}}} \tag{2.117}$$

where in this case E^{error} is the energy error of the whole substructuring problem, i.e., including numerical an experimental errors. This quantity is computed as follows:

$$E^{\text{error}} = E_I - (E_K + E_D + E_S + E_E) \tag{2.118}$$

in which E_I is the input energy, E_K is the kinematic energy, E_D is the dissipated energy through viscous damping of the numerical substructure, E_S is the strain energy stored by the numerical substructure, and E_E is the experimental energy (stored + dissipated) from the experimental substructure:

$$E_I = \int \mathbf{p}^T d\mathbf{u} \tag{2.119}$$

$$E_K = \frac{1}{2}\dot{\mathbf{u}}^T \mathbf{M} \dot{\mathbf{u}} \tag{2.120}$$

$$E_D = \int \dot{\mathbf{u}}^T \mathbf{C} d\mathbf{u} \tag{2.121}$$

$$E_S = \int \mathbf{r}^T d\mathbf{u} \tag{2.122}$$

$$E_E = \int \left(\mathbf{r}^{\text{meas}}\right)^T d\mathbf{u}^{\text{meas}} \tag{2.123}$$

In principle, the EEI measure is lower bounded by the HSEM index, that is the global measure cannot be less that the local measure. Therefore, the combination of these two global and local measures is quite valuable to assess the performance and robustness of the HS experimental design.

More recently, the predictive performance and stability indicators (PPI and PSI) were established to assess the impact of substructuring choices and the sensitivity of tracking errors on the stability of the RTHS tests (Maghareh et al., 2014). Both measures can be determined before conducting the experimental test, for the study of different experimental design alternatives. First, the steady-state variance of the tracking error is equal to:

$$\sigma^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{ee}(\omega) d\omega \tag{2.124}$$

$$S_{ee}(\omega) = |H_{eg}(\omega)|^2 S_{gg}(\omega) \tag{2.125}$$

where $S_{ee}(\omega)$ is the auto-spectral density of the tracking error; $S_{gg}(\omega)$ is the auto-spectral density of the input excitation (e.g., ground motion); and $H_{eg}(\omega)$ is the frequency response function of the tracking error $e(t)$ given an input $g(t)$. Then, the PPI measure is defined as the normalized variance of the tracking error, and is obtained by the formula:

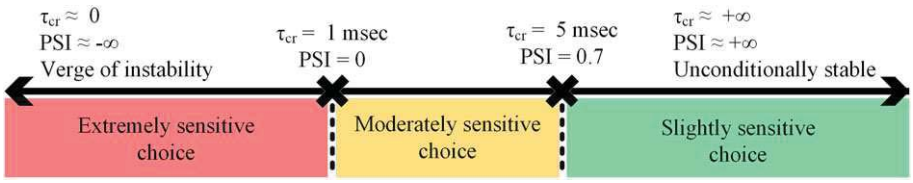


Figure 2.8: Relationship between predictive stability measures and RTHS stability (Maghareh et al., 2016a)

$$\text{PPI} = \frac{\omega_n^3 \sigma^2}{S_g} \tag{2.126}$$

where ω_n is the natural frequency of the reference structure; S_g is the power spectral density of zero-mean, stationary, white noise excitation; and σ^2 is the steady-state variance of the tracking error.

In general, the parameter PPI is closely related to the value of $|\gamma - \alpha|$, where $\gamma = M^N/M$ and $\alpha = K^N/K$ are the normalized mass and stiffness from the numerical substructure, respectively; while, M and K are the mass and stiffness matrices of the reference structural system, respectively. Then, to achieve better performance and stability for single-degree-of-freedom (SDOF) RTHS tests, three options can be applied in combination: (i) decrease PPI, which is associated to a decrease of $|\gamma - \alpha|$; (ii) decrease the natural frequency ω_n of the reference structural system; and (iii) increase the numerical damping.

Likewise, the predictive stability indicator (PSI) was proposed as:

$$\text{PSI} = \log_{10}(\tau_{cr}) \tag{2.127}$$

where τ_{cr} is the smallest critical time delay (msec) obtained by solving a delay differential equation model. A relationship between different values of predictive stability measures and the stability of the RTHS experimental designs is illustrated in Figure 2.8.

Both PPI and PSI assessment measures with different substructuring combinations were validated experimentally by Lin et al. (2015) for a SDOF RTHS test. Regardless, these measures were originally formulated for RTHS tests of SDOF systems. Therefore, direct application for MDOF problems is not straightforward and have not studied in detail yet. Although, Maghareh et al. (2016a) carried out a study to extend the notion of PSI for MDOF reference structures, where the delay differential equation is converted into a generalized eigenvalue problem.

2.7 Summary

In this chapter, different aspects of real-time hybrid simulation (RTHS) tests were explored. First, the substructuring method was explained, which serves as the foundation for conducting a hybrid simulation. In general, both a numerical and experimental substructures are considered for this kind of experimental testing. Then, the technical difficulties to conduct real-time hybrid simulation were discussed. In particular, real-time systems are designed to conduct RTHS tests, and in general these systems are governed by timing and computational resource constraints. Afterwards, different alternatives for numerical integration of the governing equations were considered, with a special emphasis on real-time integration techniques. Moreover, the detrimental effects that systematic errors can assert into the experimental test are well recognized. Therefore, different algorithms for compensation of actuator dynamics and coupling force measurements were presented. In addition, the design of any RTHS experimental setup should be assessed in terms of its local and/or global performance. For that matter, assessment measures were portrayed, such that error quantification and propagation of an RTHS test is better understood. Finally, this body of literature provides necessary and sufficient knowledge about RTHS testing, which is essential to continue with the development of the proposed framework for multi-axial real-time hybrid simulation (maRTHS).

Chapter 3

FRAMEWORK FOR MULTI-AXIAL REAL-TIME HYBRID SIMULATION TESTING

3.1 Problem statement

The purpose of this research is to develop and validate a framework to conduct three-dimensional, multi-axial, real-time hybrid simulation (maRTHS) for structural testing of complex large-scale systems subjected to seismic loading. Consequently, if real-time hybrid simulation (RTHS) is considered for multi-axial experimental testing, a control system must be designed such that the loading assembly can reproduce the boundary conditions in three-dimensional Cartesian space as accurately as possible within a fixed simulation time step.

Therefore, the maRTHS framework includes kinematic transformations for accurate motion tracking in global Cartesian coordinates of the loading platform, development of multi-actuator and multi-sensor calibration procedures, the design of model-based compensators with explicit consideration of multi-actuator dynamic coupling effects for improved accuracy and robustness, and practical implementation and validation guidelines for proof-of-concept representative examples .

3.2 Methodology

The maRTHS framework is based on hybrid and feedback control loops developed in three-dimensional Cartesian space, i.e., vector signals will carry out the information of multi-degree-of-freedom boundary conditions at the interface between numerical and experimental substructures. Thus, a boundary condition vector signal is defined as $\mathbf{u} = \{u_x, u_y, u_z, \theta_x, \theta_y, \theta_z\}^T$, which are the translational and rotational DOFs at the interface (Fermandois and Spencer, Jr., 2017).

This proposed framework includes both multi-actuator and multi-sensor modules that must be properly calibrated to allow for accurate motion tracking in 3D Cartesian space. In addition, the outer-loop controller is required for compensation of the servo-hydraulic dynamics, with explicit consideration of dynamic coupling effects between multiple actuators and the potential interaction with the experimental specimen that is attached to the loading platform.

A sketch of the overall framework is presented in Figure 3.1, where it can be seen

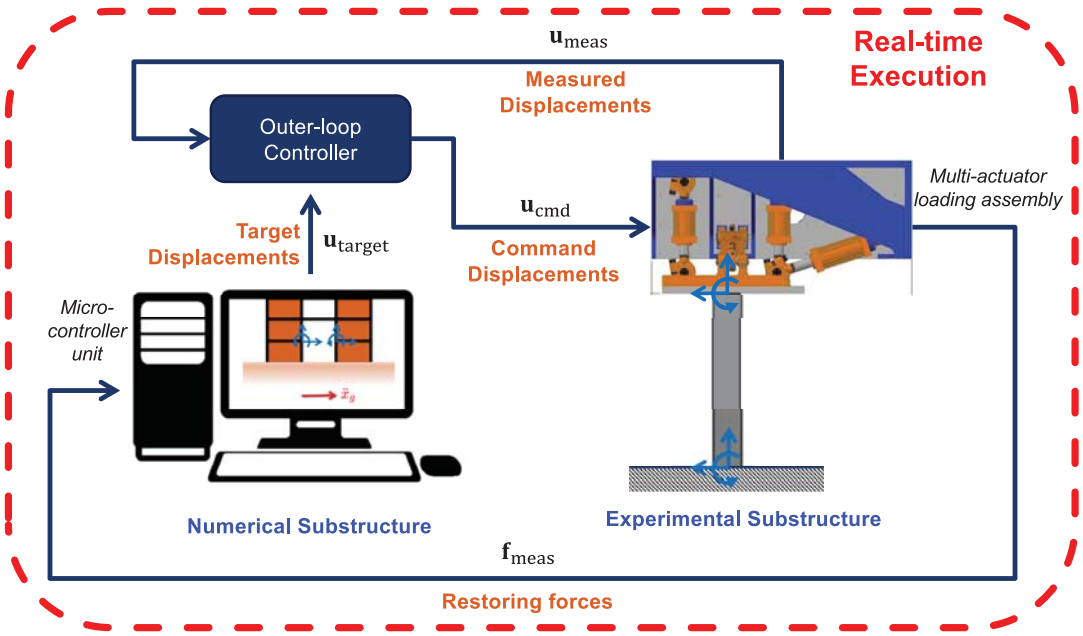


Figure 3.1: Framework for multi-axial real-time hybrid simulation (maRTHS) testing

that multiple-degree-of-freedom (MDOF) boundary conditions are imposed to the physical substructure by a modular multi-actuator loading assembly. The maRTHS test starts with the time-stepping algorithm to solve the numerical substructure must be considered first, where the target Cartesian displacements $\mathbf{u}^{\text{target}} \in \mathbb{R}^6$ at the interface between numerical and experimental substructures are obtained. Then, target Cartesian coordinates are passed through an outer-loop controller, which is the component responsible for minimizing the tracking error $\mathbf{e} \in \mathbb{R}^6$ between measured and target Cartesian displacements (i.e., $\mathbf{e} = \mathbf{u}^{\text{meas}} - \mathbf{u}^{\text{target}}$).

A command signal in Cartesian coordinates, $\mathbf{u}^{\text{cmd}} \in \mathbb{R}^6$, is provided from the outer-loop controller. This vector signal needs to be transformed to a command in actuator coordinates through a kinematic transformation. Then, each single actuator command stroke is processed by the inner-loop servo-controllers, and the resulting motion of the loading platform is obtained through an external motion measurement system, which provides the measured Cartesian coordinates, $\mathbf{u}^{\text{meas}} \in \mathbb{R}^6$, that are required for feedback control purposes.

Subsequently, after reading individual actuator forces, the restoring Cartesian forces, $\mathbf{f}^{\text{meas}} \in \mathbb{R}^6$, are estimated and applied to the numerical substructure at the interface degrees-of-freedom (DOF). Finally, the numerical substructure is ready to solve the equations for the next time step, and the procedure is repeated until the simulation ends.

Then, a real-time system is designed for the maRTHS framework, which is shown in Figure 3.2. This system is composed of three subsystems: (i) *numerical subsystem* (Figure 3.3), where the numerical substructure model, external loading, and numerical integration scheme are declared; (ii) *model-based control subsystem* (Figure 3.4), where the model-based compensation for servo-hydraulic system dynamics are defined; and (iii) *physical subsystem* (Figure 3.5), where kinematic and force transformations, calibration corrections, and

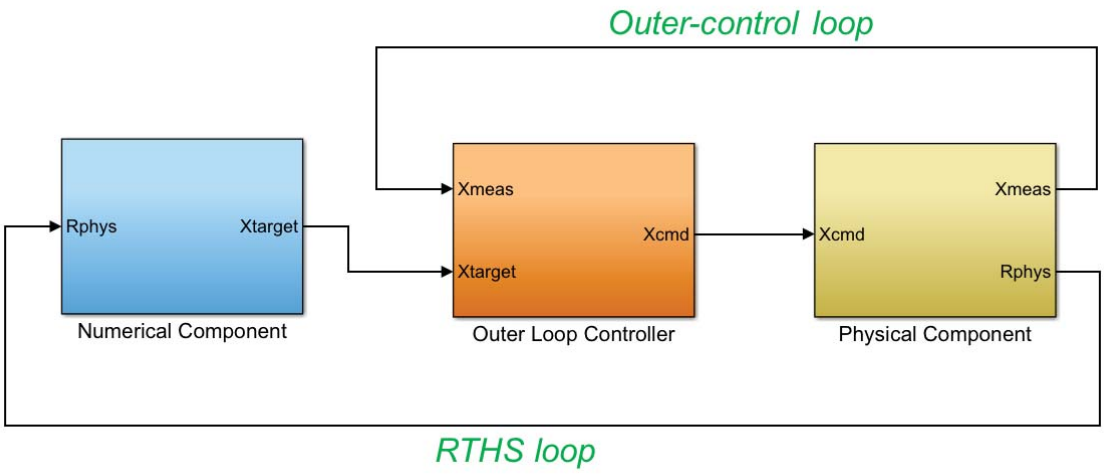


Figure 3.2: Block diagram of maRTHS code developed for real-time micro-controller execution

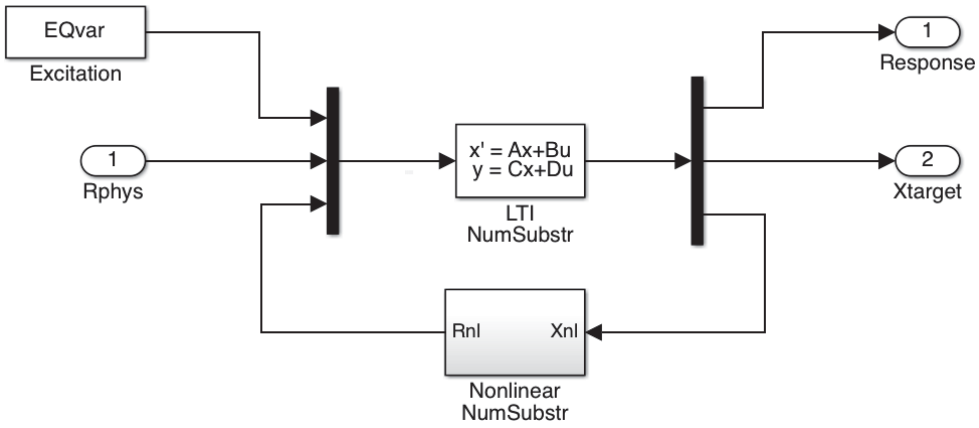


Figure 3.3: Block diagram of numerical substructure implementation

digital-analog conversions are provided to communicate with external actuators and sensors connected to the experimental substructure in real-time.

In addition, the maRTHS real-time system is designed over a micro-controller unit for real-time execution. A good practice in RTHS testing consists in embedding the code for numerical and model-based controller subsystems in the same micro-controller unit for fast calculations and seamless integration with the other experimental modules. Thus, the hybrid system can provide necessary guarantees for the real-time system to be executed in a timely manner under all expected operating conditions (i.e., avoid task overruns).

3.3 Multi-actuator loading equipment

The maRTHS framework was developed using the available experimental resources from the Newmark Civil Engineering Laboratory, located at University of Illinois at Urbana-

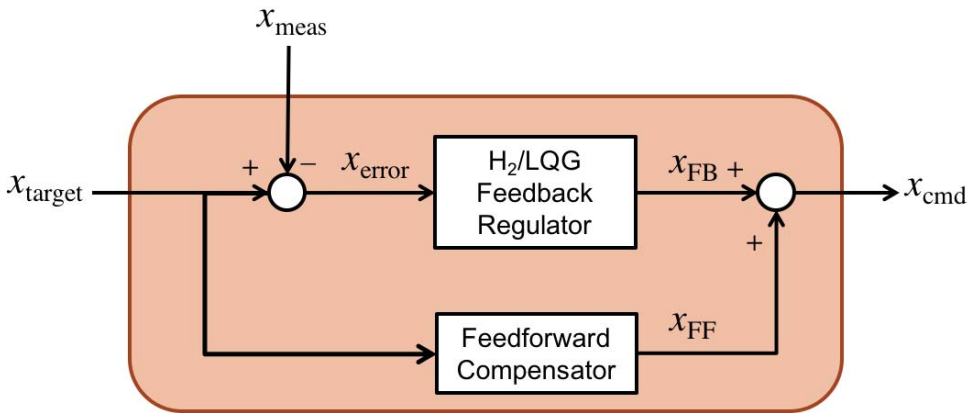


Figure 3.4: Block diagram of outer-control loop algorithm

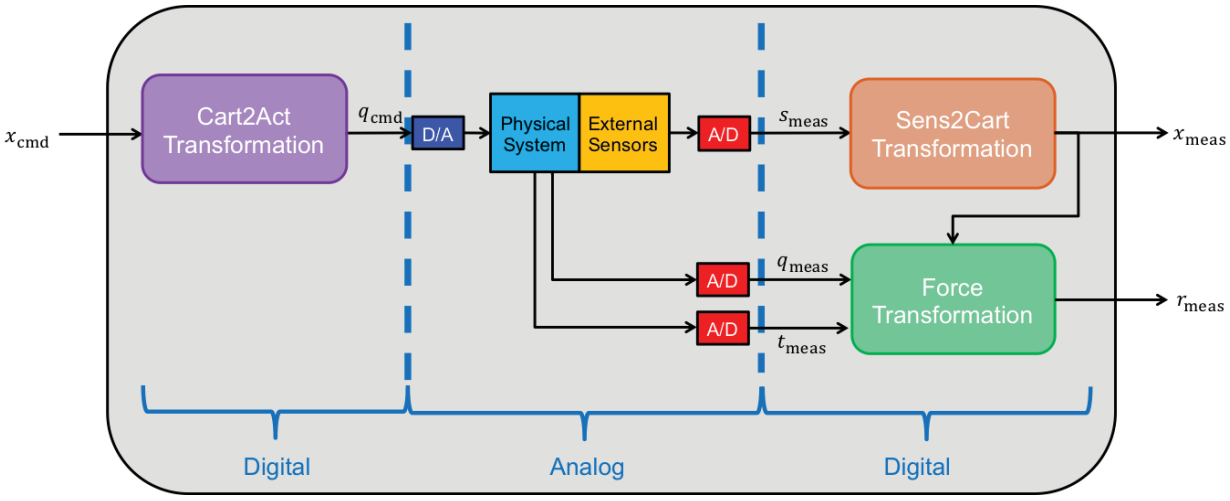
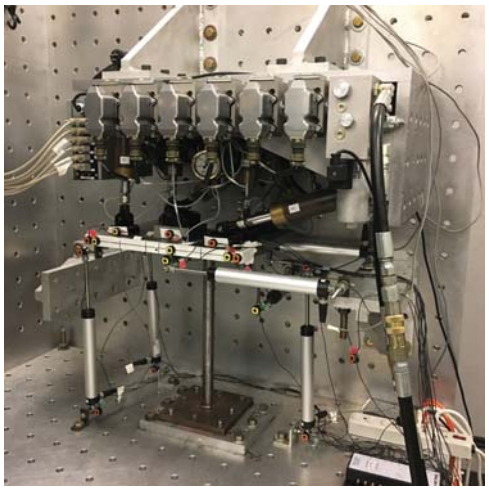
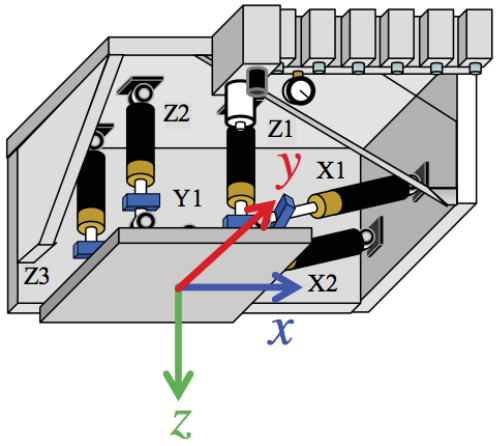


Figure 3.5: Block diagram of Cartesian-domain physical subsystem



(a) 1/5th scale LBCB



(b) Actuator labels and Cartesian coordinates

Figure 3.6: Load and boundary condition box (LBCB)

Table 3.1: Capacity specifications for small-scale LBCB (Shore Western, 2014)

	x	y	z
Displacement (mm)	±53.0	±25.4	±25.4
Rotation (deg)	±11.6	±9.4	±20.4
Force, extending (kN)	+31.14	+15.57	+46.71
Force, retracting (kN)	−18.68	−9.34	−28.02
Moment (kN-m)	±2.28	±2.66	±2.28

Champaign. In particular, a small-scale load and boundary condition box (LBCB) equipment, as shown in Figure 3.6, was chosen to conduct the multi-axial experimental tests. This equipment is a 1/5th scale version of the large-scale LBCB available at the same facility for full-scale experimental tests. The small-scale LBCB was manufactured by Shore Western Manufacturing, and it has been extensively used for academic and training purposes. For the proposed research, this device will become the testbed for development and debugging of control algorithms and testing procedures for the maRTHS framework.

The small-scale LBCB consists of six servo-hydraulic actuators mounted to a boxed frame, and connected in parallel to the loading platform. This configuration allows for controlled six-degree-of-freedom (6DOF) rigid body motion. The specifications for the small-scale LBCB are presented in Table 3.1.

In particular, the actuators are distributed such that two long actuators, with a maximum stroke of 101.6 mm (4 in), are primarily oriented with the x global coordinate; while, four shorter actuators, with a maximum stroke of 50.8 mm (2 in), are oriented primarily along the y and z global coordinates. The actuators were designed with Moog G631-3002B two-stage electro-hydraulic servo-valves, with rated flow and pressure of 10 lpm (2.5 gpm) at 1,000 psi, respectively (Moog, 2014). Also, each actuators have installed a linear variable

differential transducer (LVDT) manufactured by Trans-Tek, Inc., for stroke measurements; and a load cell manufactured by Interface, Inc., model WMC-3000, with an axial force measurement capacity of 13.34 kN (3 kip).

Furthermore, a Shore Western servo-controller is used for analog control of the small-scale LBCB. The Shore Western servo-controller consists of three SC6000 cards. Each card allows for high-precision feedback control of two individual actuators. On top of that, actuator stroke commands can be performed from an external source through the SC6000 cards, which is a feature that is fundamental for the maRTHS framework. In addition, the Shore Western servo-controller provides signal conditioning for LVDT and load cell measurements from all actuators, and it allows for analog output signals that can be acquired directly from an external real-time hardware with data acquisition capabilities.

Finally, a dedicated hydraulic power supply (HPS) with a capacity of 37.85 lpm (10 gpm) at $3,000$ psi is provided to operate the actuators of the small-scale LBCB. The connection between HPS and LBCB is made through a Shore Western 213(3)B-4 hydraulic service manifold (HSM), which has flow capacity of (60 gpm). The HSM has three independent outlets for each small-scale LBCB, each equipped with two pressure control solenoid valves for low- and high-pressure conditions, respectively. In addition, the HSM includes pressure and return accumulators that to provide sufficient pressure supply for dynamic testing purposes, and a filter.

3.4 Real-time system platform

The real-time system implementation of the proposed framework is summarized in Figure 3.7. To control the synchronized motion of the actuators in Cartesian coordinates, a micro-controller unit (MCU) is connected to the Shore Western servo-controller.

The MCU hardware of choice is a dSpace DS1103PPC micro-controller, based on a single PPC 750GX processor running at 1 GHz. This MCU comes with 20 analog-to-digital (A/D) channels and 8 digital-to-analog (D/A) channels for external device communications, each with a 16-bit resolution. Also, a host PC is connected directly to the MCU via fiber optics. The host PC consists of an Intel Core 2 Quad Processor (Q9300) running at 2.5 GHz, with 3.25 GB RAM memory, and working with Windows XP operating system. The host PC is responsible of storing all the programming code and preferences required for maRTHS, as well as recording the test measurements data. A photo of the MCU, host PC, and connector boards, is shown in Figure 3.8.

The integrated development environment (IDE) is Matlab Release 2007b. The algorithms of the real-time system are developed using Simulink block diagrams and libraries. In addition, Simulink Coder (formerly Real Time Workshop) is considered to provide a seamless transition between Simulink models and the MCU. C source code is automatically generated and loaded into the dSpace MCU, by using dSpace's Real Time Interface (RTI) software. In addition, virtual instrument interfaces can be developed to check parameters of the simulation on the fly, by using dSpace's ControlDesk software. An example of the current development of virtual instruments for maRTHS testing is shown in Figure 3.9. Likewise, a connector board allows the communication with the multi-actuator servo-controller, in terms

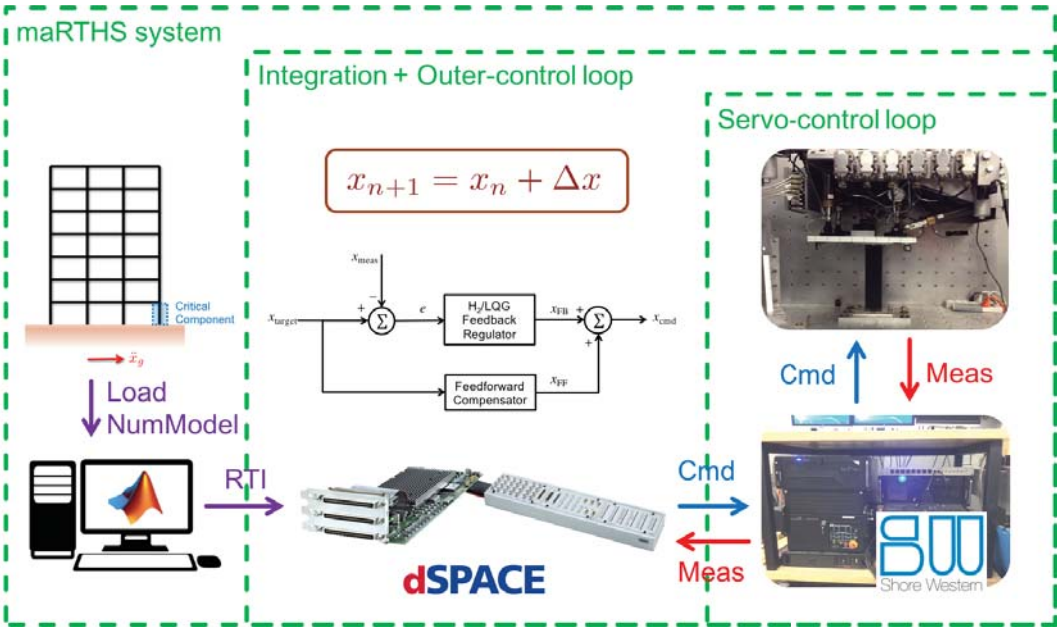


Figure 3.7: Implementation of proposed framework using small-scale LBCB equipment

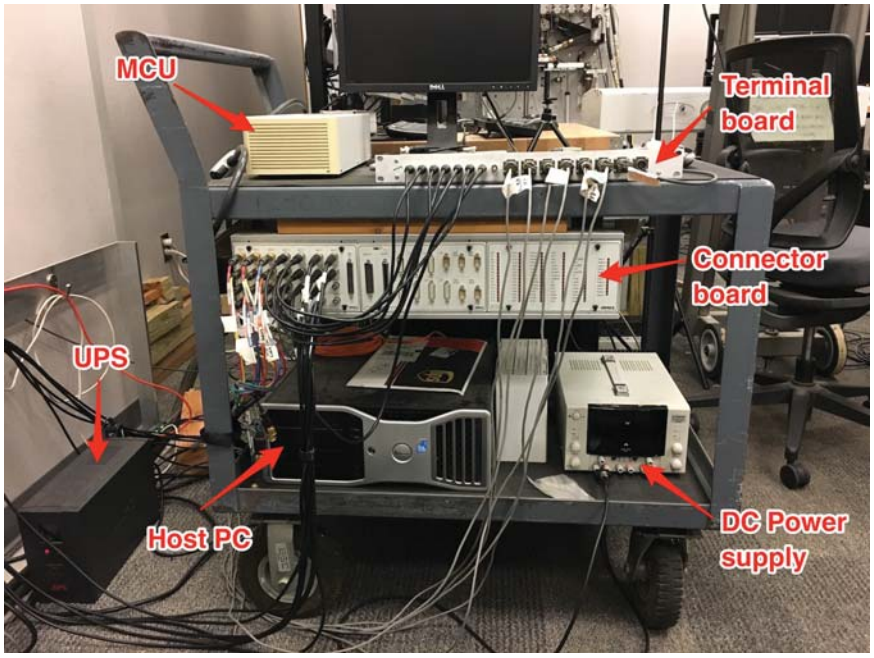


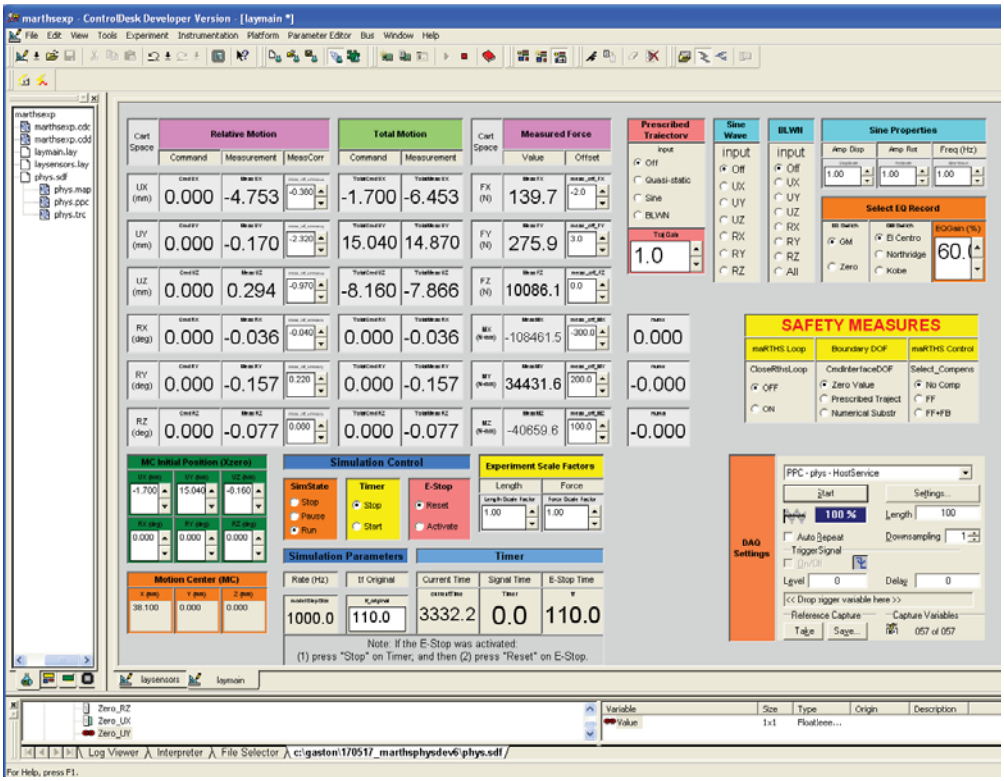
Figure 3.8: Assembled real-time hardware for maRTHS testing

of command signals (`cmd`) and measurement signals (`meas`), for every actuator and sensor installed in the LBCB.

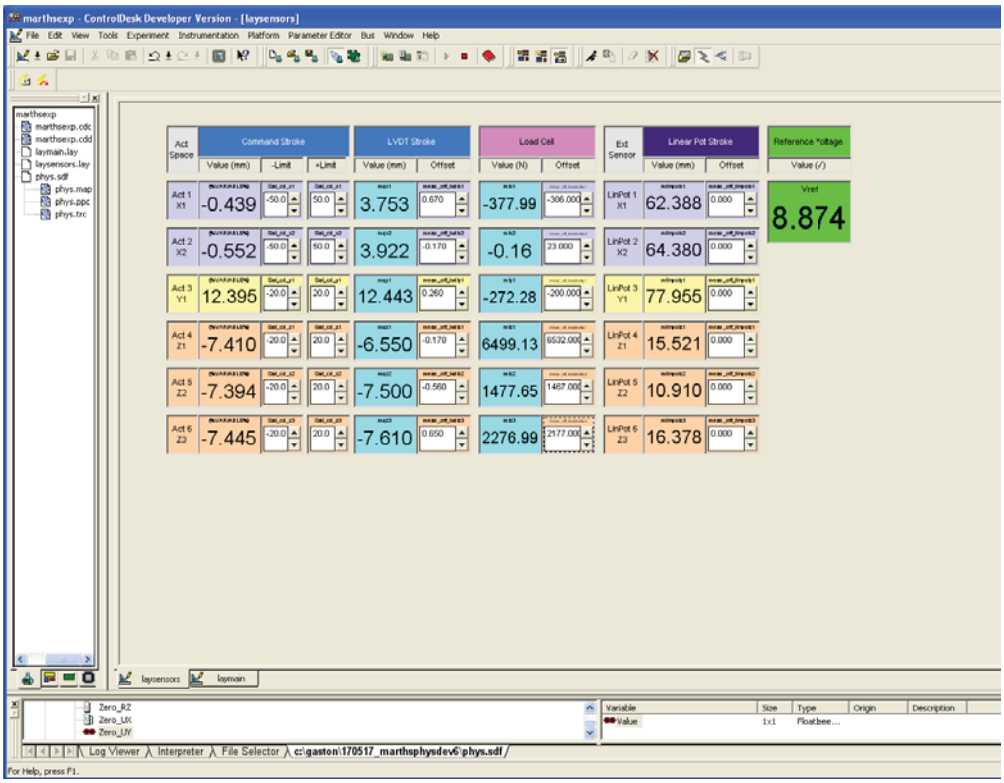
In addition, finite element analysis (FEA) of numerical substructures must be conducted using Matlab/Simulink models for compatibility with maRTHS framework. Fortunately, two standard software packages for RTHS are readily available to be used in maRTHS framework: (i) *RT-Frame2D* (Castaneda et al., 2012); and (ii) *HybridFEM* (Karavasilis et al., 2009). Both software packages allow for planar structural analysis of numerical substructures with beam-column elements, and provides a good variety of nonlinear constitutive relations and integration algorithms for RTHS tests. Nevertheless, this framework allows for the implementation of user-defined code to solve numerical structures using a state-space approach (Simeonov et al., 2000) and incorporate novel integration schemes. Both solutions may be required for optimization of MCU computational resources, and for increased reliability of real-time scheduling schemes.

3.5 Summary

In this chapter, an overall description of the multi-axial real-time hybrid simulation (maRTHS) has been presented. The framework is developed based on traditional RTHS schemes, and extended to deal with vector signals in Cartesian space for hybrid and control loops, respectively. Furthermore, details of both hardware and software required to build the maRTHS system were presented. Finally, although the current implementation of this framework considered a small-scale LBCB loading assembly, the framework scalability in terms of real-time system design should not be a problem for large-scale LBCB loading assembly and full-scale test specimens.



(a) Control instruments in Cartesian coordinates



(b) Sensor settings in actuator/transducer coordinates

Figure 3.9: Virtual instruments developed in dSpace's ControlDesk for maRTHS testing

Chapter 4

KINEMATIC TRANSFORMATIONS FOR MULTI-ACTUATOR LOADING ASSEMBLIES

4.1 Problem statement

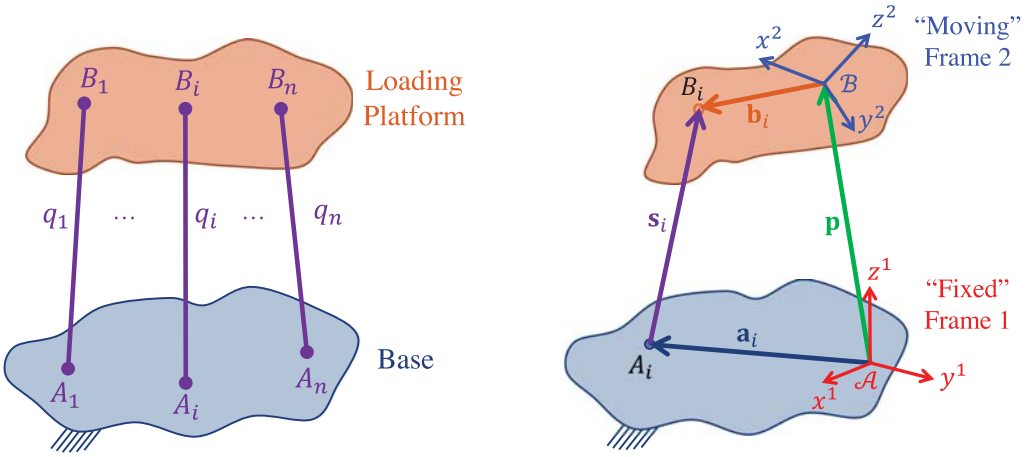
The target displacements from the numerical substructure are applied to the test specimen through multiple servo-hydraulic actuators attached to the loading platform. The actuators can only be commanded to move along its axis; therefore, a kinematic transformation between actuator and Cartesian space coordinates is required. Similarly, any displacement transducer attached to the loading platform will also measure the motion only in the direction of the sensor, which implies that kinematic transformations are also needed to acquire Cartesian coordinate measurements. Because real-time operation is essential in dynamic testing, the objective is to develop an explicit and fast solution for inverse and forward kinematic transformation problems.

4.2 Kinematics of parallel manipulators

The following provides an insight on the formulation for real-time kinematic transformations to be used on the maRTHS framework. First, we begin by writing the kinematic equations for a *parallel manipulator* (Merlet, 2006), which is the architecture chosen for the loading assembly in maRTHS (i.e. small-scale LBCB).

The parallel manipulator (see Figure 4.1a) consist of n prismatic joints (i.e. actuators or displacement transducers). The "i-th" joint is connected at one end to a fixed body at point A_i, and at the other end to a moving body at point B_i. Hence, n joints are connected in parallel to both bodies. Then, we can define two frame systems at each body; a "fixed" frame attached to the base (e.g., reaction frame), and the other "moving" frame attached to the loading platform, as shown in Figure 4.1b. This definition will be useful to represent six-degree-of-freedom (6DOF) rigid body motion of the moving loading platform in either system of coordinates.

Hence, two kinematic transformations are defined between joint coordinates (strokes from actuator or displacement transducers) and Cartesian coordinates. First, the transformation from Cartesian to joint coordinates is called *inverse kinematic transformation* (IKT), and is represented by the following mathematical relationship for three-dimensional



(a) Closed kinematic chain (b) Kinematics of "i-th" prismatic link

Figure 4.1: Parallel manipulator kinematics

Cartesian space:

$$\mathbf{s}_i = \mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i, \quad (i = \{1, \ldots, n\}) \tag{4.1}$$

where $\mathbf{s}_i \in \mathbb{R}^3$ is the vector in global coordinates that represents the position of the "i-th" joint; $\mathbf{p} \in \mathbb{R}^3$ and $\mathbf{R} \in \mathbb{R}^{3\times3}$ are the translation vector and rotational matrix that represents the position of the body frame with respect to the global frame; $\mathbf{a}_i \in \mathbb{R}^3$ are the fixed end coordinates of the "i-th"joint relative to the fixed frame; and $\mathbf{b}_i \in \mathbb{R}^3$ are the free end coordinates of the "i-th" joint relative to moving frame, and n is the total number of joints connected to the moving platform. For an over-constrained parallel manipulator, usually the number of actuators and transducers is greater or equal to the number of motion components from the loading platform. In this case, the number of actuators for three-dimensional purposes should be $n \geq 6$.

The translation vector and rotational matrix is derived from the Cartesian command signal $\mathbf{u}^{\text{cmd}} = \mathbf{u} = \{u_x, u_y, u_z, \theta_x, \theta_y, \theta_z\}^T$. The structure of the translation vector is given by $\mathbf{p} = \{u_x, u_y, u_z\}^T$, where (u_x, u_y, u_z) are the Cartesian translation coordinates; while, the rotation matrix is given by $\mathbf{R} = \mathbf{R}(\theta_x, \theta_y, \theta_z)$, where $(\theta_x, \theta_y, \theta_z)$ are the Euler angles from rigid body motion. Also, the relationship $\mathbf{R}(\theta_x, \theta_y, \theta_z)$ can be decomposed by the product of three elemental rotation matrices, $\mathbf{R}_x(\theta_x)$, $\mathbf{R}_y(\theta_y)$, and $\mathbf{R}_z(\theta_z)$, which are obtained by direct conversion from Euler angles using the yaw-pitch-roll convention:

$$\mathbf{R}(\theta_x, \theta_y, \theta_z) = \mathbf{R}_z(\theta_z)\mathbf{R}_y(\theta_y)\mathbf{R}_x(\theta_x) \tag{4.2}$$

Therefore, the formula to obtain joint strokes from IKT is the following:

$$
\begin{aligned}
q_i &= q_i(u_x, u_y, u_z, \theta_x, \theta_y, \theta_z) \\
&= \|\mathbf{s}_i\| \\
&= \|\mathbf{p}(u_x, u_y, u_z) + \mathbf{R}(\theta_x, \theta_y, \theta_z)\mathbf{b}_i - \mathbf{a}_i\|
\end{aligned} \tag{4.3}
$$

where q_i is the length of "i-th" joint, and $\|\cdot\|$ corresponds to the Euclidean norm. In particular, actuator length commands $\mathbf{q}^{\text{cmd}} = \{q_1, q_2, \ldots, q_n\}^T$ are obtained by solving the last equation for all actuator joints $i = \{1, 2, \ldots, n\}$. From this expression, the inverse kinematic transformation results in a nonlinear function of the Cartesian coordinates, with a closed-form solution that can be solved explicitly for real-time execution.

Subsequently, the transformation from joint to Cartesian coordinates is called *forward kinematic transformation* (FKT), which should corresponds to the inverse mapping of the inverse kinematic transformation described previously. The problem is stated as follows: given the stroke measurement of the "i"-th joint, q_i^{meas} , calculate the Cartesian coordinates $\mathbf{u}^{\text{meas}} = \{\hat{u}_x, \hat{u}_y, \hat{u}_z, \hat{\theta}_x, \hat{\theta}_y, \hat{\theta}_z\}^T$ of the loading platform. Thus, to obtain Cartesian coordinates from joint strokes, the previous equation (4.3) is rewritten as follows:

$$
g_i(q_i^{\text{meas}}, \mathbf{u}^{\text{meas}}) = q_i^{\text{meas}} - q_i(\mathbf{u}^{\text{meas}}) = 0, \quad (i = \{1, \ldots, n\}) \tag{4.4}
$$

Indeed, the later expression is an implicit function of Cartesian coordinate estimates $\mathbf{u}^{\text{meas}}$; hence, the problem does not have a closed-form solution. Also, the solution is determined only if the number of joint equations n is equal to the total number of rigid body Cartesian coordinates in three-dimensional space, i.e., $n = 6$. In the case of $n \geq 6$, then the problem is overdetermined, and the solution can be obtained as a least squares approximation. Moreover, if $n \leq 6$, then the problem is undetermined, and no unique solution can be found.

Clearly, the only way to solve the problem is by numerical methods, where the candidate solution needs to be updated at each iteration step until convergence is achieved. Hence, the FKT problem can be rewritten as an optimization problem where an error measurement is minimized:

$$
\begin{aligned}
\underset{\mathbf{p},\mathbf{R}}{\text{minimize}} \quad & f_0 = \frac{1}{2}\sum_{i}^{n} \mathbf{e}_i^T \mathbf{e}_i \\
\text{subject to} \quad & \mathbf{e}_i = (\mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i) - \mathbf{s}_i, \\
& \mathbf{R}^T\mathbf{R} = \mathbf{I}_{3\times 3}, \\
& \det(\mathbf{R}) = +1.
\end{aligned} \tag{4.5}
$$

where the last two constraints are needed to enforce that the resulting rotation matrix $\mathbf{R}$ does indeed belong to the special orthogonal group $SO(3)$, a necessary condition for rigid body motion.

This FKT problem can be solved iteratively until convergence is achieved with a predefined tolerance. But, iterations will impose time scheduling constraints to the real-time system which is a major issue. Another alternative is to obtain an approximation of the

measured Cartesian coordinates through linearization of the IKT relationship from (4.1). By performing a Taylor series expansion around the Cartesian coordinate $\mathbf{u}_k$ at step "k":

$$\delta\mathbf{q} \approx \mathbf{J}\delta\mathbf{u} \tag{4.6}$$

where $\delta\mathbf{q} = \mathbf{q}_{k+1} - \mathbf{q}_k$ is the joint coordinate increment, $\delta\mathbf{u} = \mathbf{u}_{k+1} - \mathbf{u}_k$ is the Cartesian coordinate increment, and $\mathbf{J} \in \mathbb{R}^{n\times 6}$ is a Jacobian matrix defined in indicial notation by:

$$J_{rs} = \frac{\partial q_r}{\partial u_s}(\mathbf{u}_k), \quad (r = \{1,\dots,n\}, s = \{1,\dots,6\}) \tag{4.7}$$

or in matrix form:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial q_1}{\partial u_1}(\mathbf{u}_k) & \cdots & \frac{\partial q_1}{\partial u_6}(\mathbf{u}_k) \\ \vdots & \ddots & \vdots \\ \frac{\partial q_n}{\partial u_1}(\mathbf{u}_k) & \cdots & \frac{\partial q_n}{\partial u_6}(\mathbf{u}_k) \end{bmatrix} \tag{4.8}$$

Then, if the Jacobian matrix is invertible (i.e., matrix is square and non-singular), a *linearized forward kinematic transformation* (LFKT) is obtained to calculate the Cartesian coordinates at next step $k+1$:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{J}^{-1}\left(\mathbf{q}_{k+1} - \mathbf{q}_k\right) \tag{4.9}$$

Each term of the Jacobian matrix is obtained analytically, thus its numerical evaluation is straightforward. More details on the analytical expression of the Jacobian matrix are provided in Appendix A.

In the particular case of over-actuated multi-actuator system, the Jacobian matrix is non-square, therefore the inverse does not exist. However, it may be possible to approximate the inverse Jacobian matrix using the Moore-Penrose pseudo-inverse, which is basically a least-squares approximation:

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \mathbf{J}^{+}\left(\mathbf{q}_{k+1} - \mathbf{q}_k\right) \tag{4.10}$$

where

$$\mathbf{J}^{+} = (\mathbf{J}^{*}\mathbf{J})^{-1}\mathbf{J}^{*} \tag{4.11}$$

Due to physical constraints, the actuators cannot faster than the flow allowed by each servo valve. Therefore, a physical constraint can be enforced by the following expression:

$$|\Delta q_k^i| \leq v_{\max}^i \Delta t, \quad \forall i = \{1,\dots,n\} \tag{4.12}$$

where $v_{\max}^i$ is the velocity capacity of the actuator "i", and Δt is the sampling time step.

In addition, two choices are available for the LFKT: (i) linearization around initial position (constant Jacobian); and (ii) linearization around current position (incremental Jacobian). In the first approach, the Jacobian is kept constant for the whole test. While, the second approach considers the update of the Jacobian matrix at each time step. The latter approach is more computationally expensive, thus timing constraints of the real-time

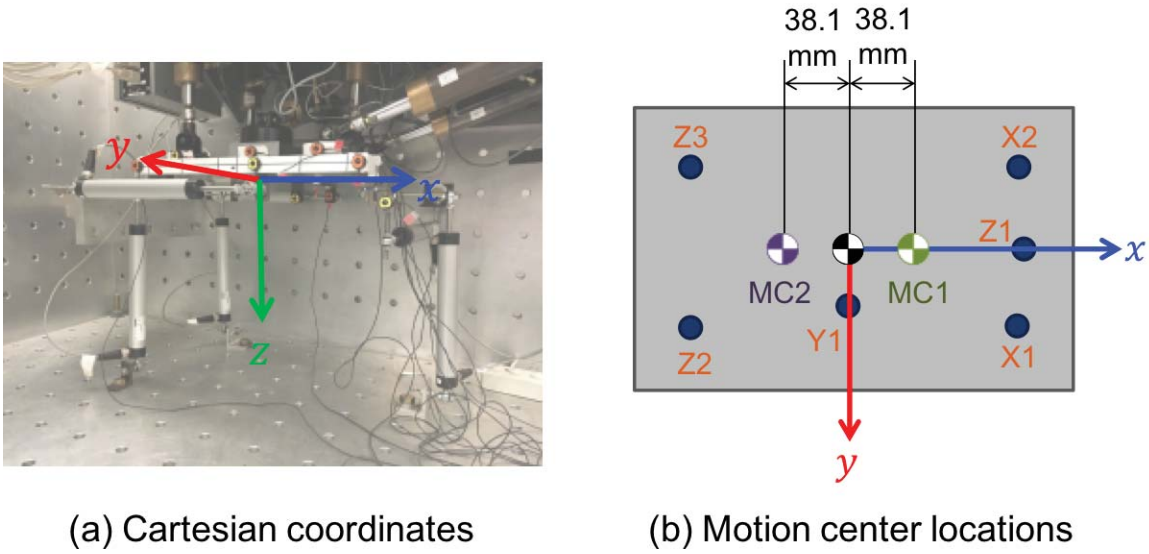


Figure 4.2: Sign convention and location of Cartesian coordinate system for small-scale LBCB

system should be a concern when implementing this approach.

Therefore, for real-time applications, the exact IKT and approximate LFKT algorithms are considered for implementation in maRTHS testing. In general, the LFKT algorithm does not provide any guarantees that it can be accurately solved for the Cartesian position of the loading platform motion center, due to the implicit nature of the FKT relationships. However, Mercan et al. (2009) found that incremental schemes to solve the FKT problem have good agreement with the true FKT results, provided that the incremental time step is sufficiently small. Usually, a sampling time step of 1/1000 seconds should be small enough to obtain good results of the measured Cartesian coordinates, while meeting the requirements from the real-time system.

Moreover, the kinematic transformations allow for the motion control of any point of interest in Cartesian space. But, usually the control point of interest in RTHS should be along the test specimen that is connected to the loading platform. For example, a choice of Cartesian coordinates for the small-LBCB case is shown in Figure 4.2, where the motion center (i.e., origin of coordinate system) is offset from the centroid of the loading platform, and located at position MC2. This choice of motion center is associated to the hole pattern on the loading platform for the connection of the test specimen.

Furthermore, when flexibility of the reaction frame should be accounted for Cartesian coordinate measurements (Chang et al., 2015), instead of using the LVDT transducers located inside each actuator, a collection of external displacement transducers are connected in parallel to the loading platform. Hence, six Celesco CLWG-150-MC4 linear potentiometers were selected for external displacement transducers. The measurement range for each linear potentiometer is 0 to 150 mm (6 in). An external power supply provides the reference voltage for stroke measurements. Also, the sign convention for actuator stroke is positive for extension (negative for retraction).

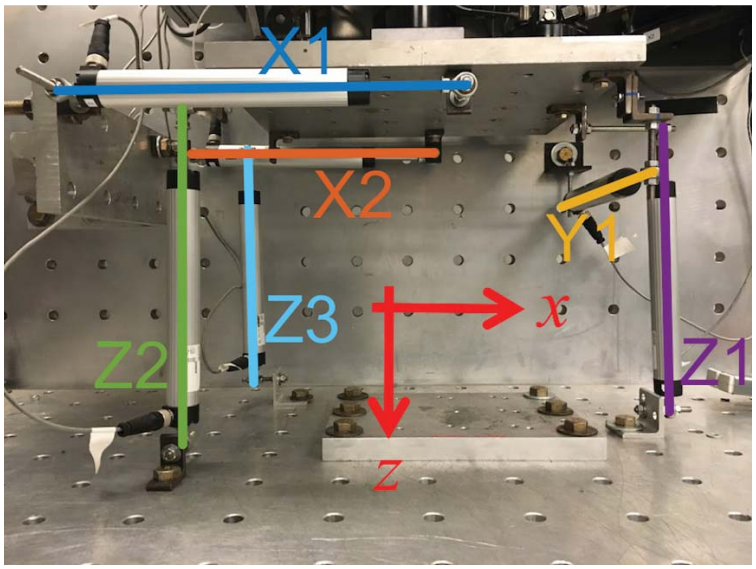


Figure 4.3: External sensor system considered for estimation of Cartesian coordinates of motion center

Hence, the external sensors were installed and conveniently oriented as close as possible to the Cartesian axes, as shown in Figure 4.3. In consideration of this sensor arrangement, the LFKT Jacobian matrix $\mathbf{J}$ of the mapping from Cartesian coordinates to external sensor strokes at the default position of the loading platform, as is obtained and presented in Figure 4.4. Each component of the LFKT Jacobian matrix $\mathbf{J}$ represents the sensitivity values of sensor strokes for a unit increment in the Cartesian motion. For instance, at the default configuration of the LBCB, a positive unit increment in Cartesian coordinate $\delta u_x = 1$ mm will correspond to a negative increment on the strokes of sensors X1 and X2 only, with values of $\delta q_{\mathrm{X1}} = \delta q_{\mathrm{X1}} = -0.95$ mm; while, the stroke of the other sensors will be small. Moreover, this results are in good agreement with previous observations by Nakata et al. (2010) on the kinematics of LBCB for classical hybrid simulation tests.

4.3 Static force analysis of loading platform

The measurement of restoring forces from the test specimen is a fundamental quantity required for hybrid simulation. Typically, multi-actuator loading assembly have inline load cells installed to each actuator. Thus, the restoring force in Cartesian coordinates can be obtained through a static analysis of the parallel manipulator (Merlet, 2006).

Let $\mathbf{f} \in \mathbb{R}^6$ be the restoring force vector in Cartesian space, and and $\boldsymbol{\tau} \in \mathbb{R}^n$ the measured forces from load cells in actuator space. To obtain an estimation of the restoring forces at the motion center in Cartesian coordinates (forces and moments), a force mapping between Cartesian and actuator spaces is required. From the principle of virtual work:

$$\delta W = \mathbf{f}^T \delta \mathbf{u} - \boldsymbol{\tau}^T \delta \mathbf{q} \tag{4.13}$$

	UX	UY	UZ	RX	RY	RZ
X1	-0.95	0.00	0.32	0.68	0.25	2.02
X2	-0.95	0.00	0.32	-0.68	0.25	-2.02
Y1	0.00	0.96	0.27	2.27	-0.17	0.60
Z1	0.00	0.00	1.00	0.00	-3.15	0.00
Z2	0.00	0.00	1.00	2.13	1.98	0.00
Z3	0.00	0.00	1.00	-2.13	1.98	0.00

Figure 4.4: Jacobian matrix $\mathbf{J}_{\mathrm{MC2}}$ considered for LFKT algorithm (translations in mm, rotations in deg, strokes in mm)

By substituting the linearized kinematics equation (4.6) into the previous equation:

$$\begin{aligned} \delta W &= \mathbf{f}^T \delta \mathbf{u} - \boldsymbol{\tau}^T \mathbf{J} \delta \mathbf{u} \\ &= \left(\mathbf{f}^T - \boldsymbol{\tau}^T \mathbf{J} \right) \delta \mathbf{u} \end{aligned} \tag{4.14}$$

From equilibrium ($\delta W = 0$):

$$\mathbf{f} = \mathbf{J}^T \boldsymbol{\tau} \tag{4.15}$$

Therefore, to obtain the restoring forces in Cartesian space, the Jacobian matrix must be solved first using the LFKT approach, and then evaluate the load cell measurements $\boldsymbol{\tau}$ in (4.15). Thus, force measurement task is highly dependent on the LFKT task in real-time execution.

4.4 Calibration of motion control system

Two calibration corrections are essential to improve the accuracy of the motion of the loading assembly: (i) *command calibration* is required in order to match both the command and real displacements measured from a standard reference; and (ii) *external sensor calibration*, where both the estimated displacements from the external sensors is calibrated according to the real measured displacements from a standard reference. In both cases, pseudo-static (very slow rate) motion is considered. Also, the standerd reference is provided by Krypton K600 DMM, a contact-less dynamic measuring machine, that provides very accurate Cartesian position measuring of up to ± 0.02 mm in three-dimensional (3D) Cartesian space.

The loading trajectory was selected as a ramp-hold sequence to achieve very slow

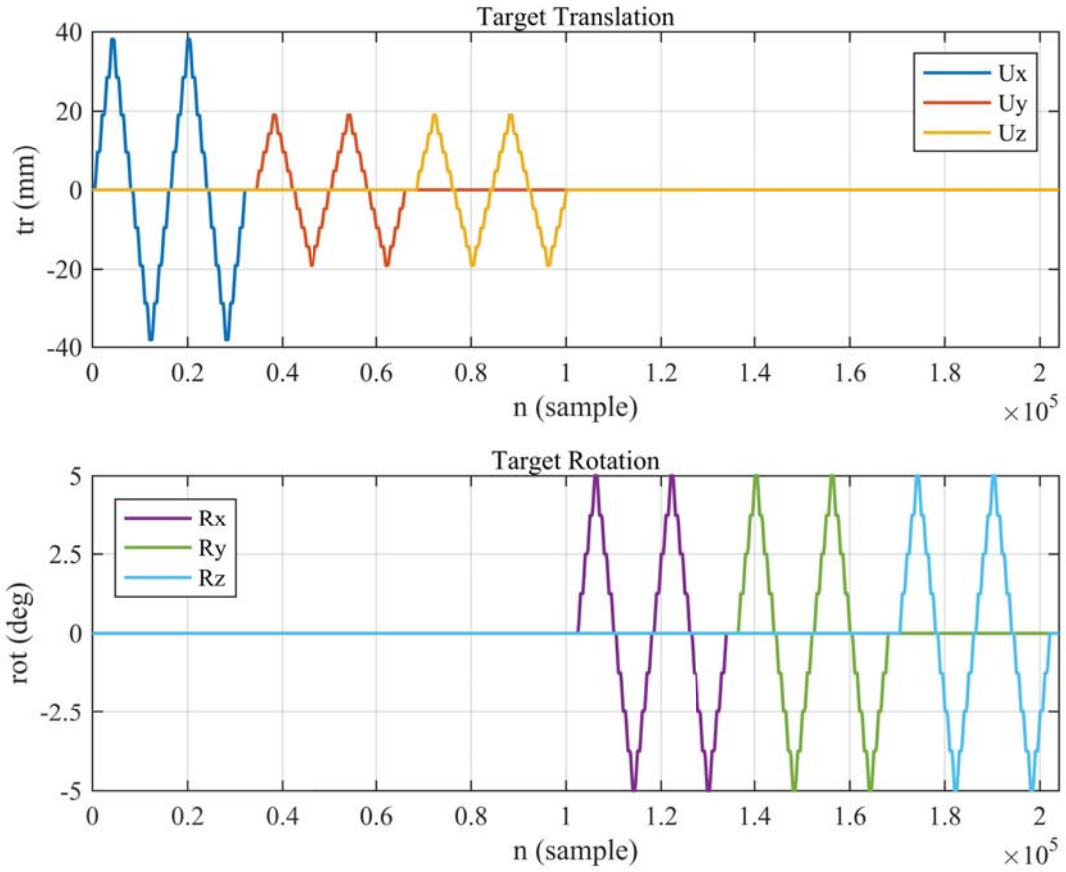


Figure 4.5: Target Cartesian coordinate trajectory for calibration purposes

motion of the small-scale LBCB, with two cycles per Cartesian coordinate, as shown in Figure 4.5. More than 200,000 data samples are collected for each channel in order to create linear regressors that will serve for calibration correction purposes.

After the calibration iteration process, the results for command errors and measurement errors are presented in Figure 4.6 and 4.7, respectively. It can be seen that the overall accuracy for command displacements is less than 0.25 mm in translation, and 0.05 deg in rotation, results that are in agreement with the calibrations performed by Nakata et al. (2010). In addition, the overall accuracy for measurement displacements is less than 0.2 mm in translation, and less than 0.03 deg in rotation. Indeed, the LFKT algorithm together with the external measuring system is able to accurately estimate Cartesian position of the motion center for the chosen loading trajectories. Although, more research is required in order to assess the accuracy for other Cartesian trajectories that are beyond the limits of this study.

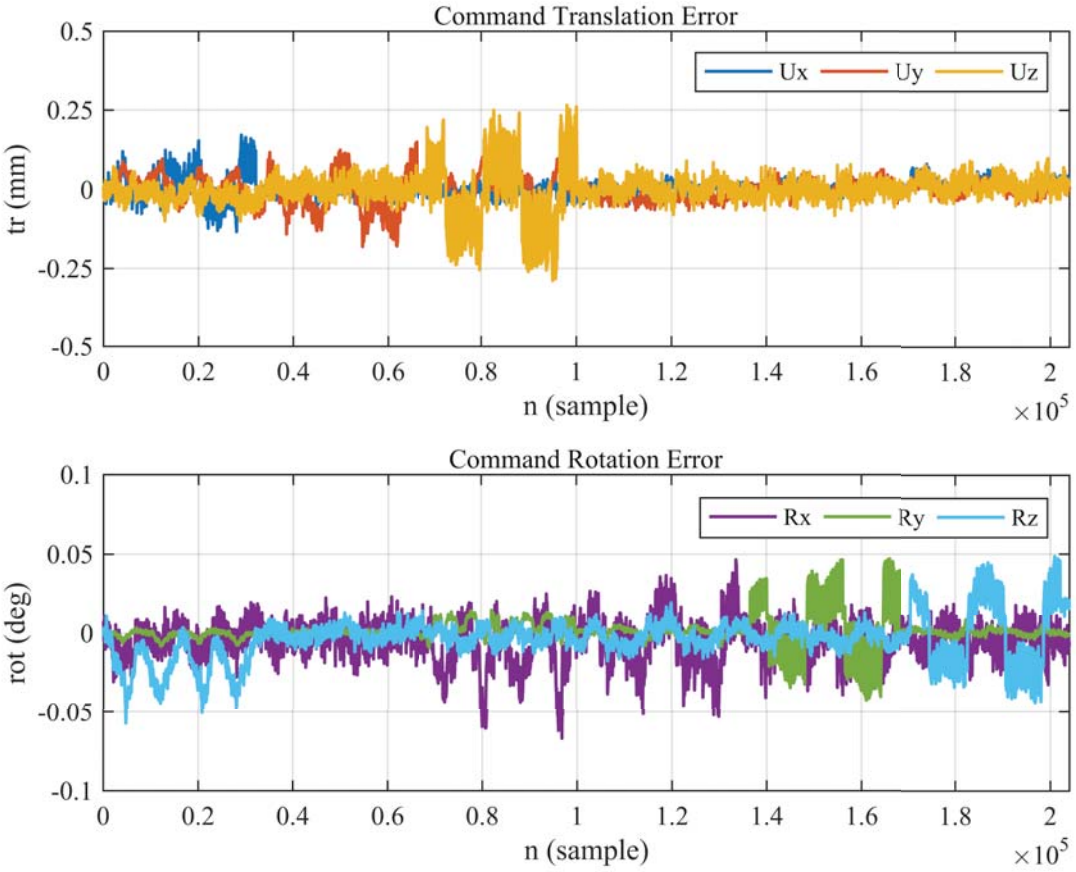


Figure 4.6: Cartesian command errors obtained after calibration procedure

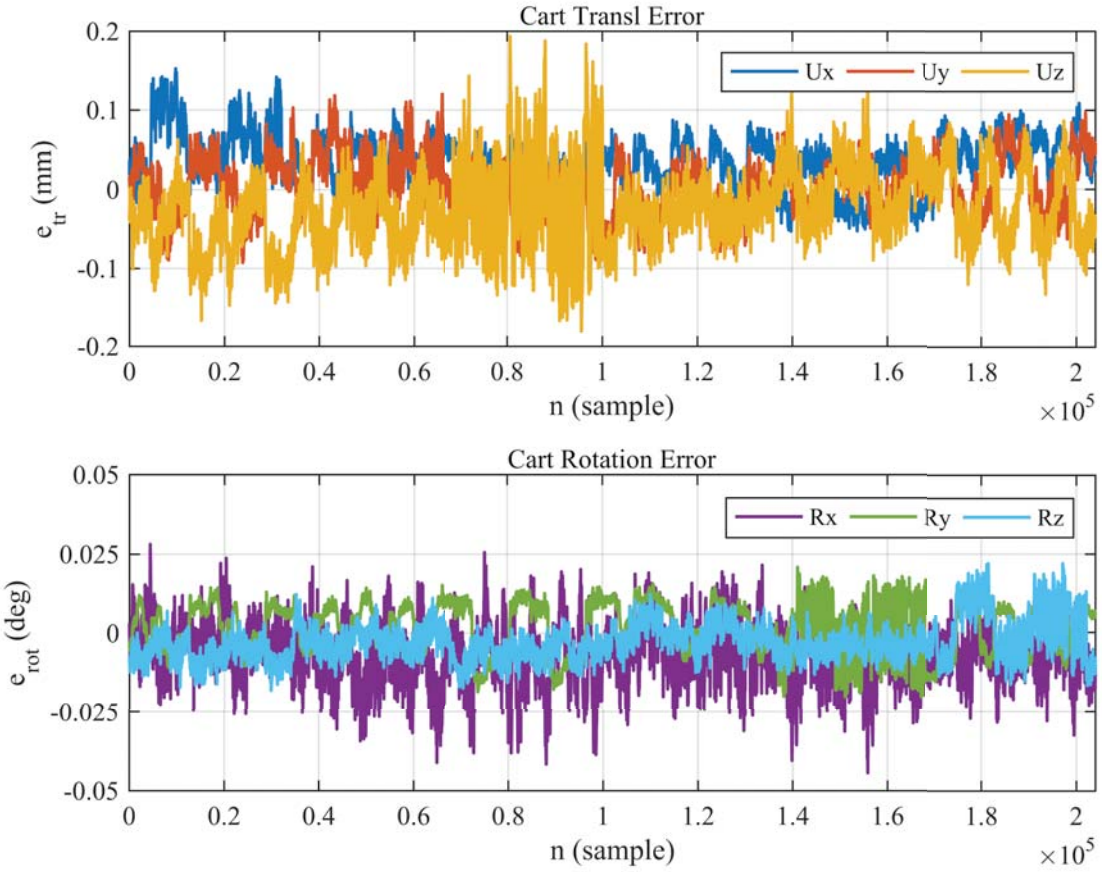


Figure 4.7: External measurement system errors obtained after calibration procedure

4.5 Summary

In this chapter, a study of the loading assembly kinematics for maRTHS testing was performed for motion control purposes. The loading assembly consists of multiple actuators connected in parallel to a rigid platform, where the kinematic relationships between rigid body motion of the loading platform in Cartesian space and the actuator strokes were derived. Then, two kinematic transformations were presented in this study: (i) inverse kinematic transformation (IKT), to convert command Cartesian position of the loading platform into individual actuator command strokes; and (ii) linearized forward kinematic transformation (LFKT), which is an incremental scheme to estimate measured Cartesian position of the loading platform from measured transducer strokes. Following these steps, a force relationship between Cartesian and actuator coordinates was also proposed, such that restoring forces of the test specimen can be directly measured using the sensors installed on the multi-actuator loading assembly. Finally, a calibration procedure for accurate position control of the loading platform was presented. The results for the calibration show very small errors in Cartesian space, that are good enough for experimental testing purposes, provided that the sampling rate will not exceed 1000 Hz.

Chapter 5

SYSTEM IDENTIFICATION OF MULTI-ACTUATOR SYSTEMS

5.1 Problem statement

To provide good reference tracking and robustness properties for multi-axial real-time hybrid simulation, an accurate representation of the dynamics of the experimental setup is needed. The goal of this chapter is to obtain a model of the multi-input, multi-output (MIMO) experimental system that incorporates all the properties from the servo-hydraulic actuators, the test specimen interaction, and any multi-actuator dynamic coupling effects, while ensuring desired characteristics such as stability and minimality of the solution.

5.2 White box model

In this section, a physically-based model for a single servo-hydraulic actuator will be provided. This model will be quite useful to capture the essential characteristics of single actuators when combined into multi-actuator loading systems, which are very complex systems that can only be be studied using a black box model approach. More information on black box modeling of single and multiple actuator systems will be presented in Section 5.3.

A typical servo-hydraulic actuator is illustrated in Figure 5.1, which is composed of an electro-hydraulic servo valve and a hydraulic actuator. The white box model of a servo-hydraulic actuator consists in the mathematical representation of three components: (i) servo valve dynamics; (ii) hydraulic actuator dynamics; and (iii) feedback control. The model of the servo-hydraulic actuator can be represented as a block diagram, which is illustrated in Figure 5.2.

In particular, the effects of pipeline and power supply dynamics are not considered in this model. Also, a further assumption is that the effects of pipeline dynamics are not significant, which is usually the case for low-frequency applications. In addition, the power supply (pumps) is given such that it provides a constant supply pressure, which is usually considered as a reasonable assumption.

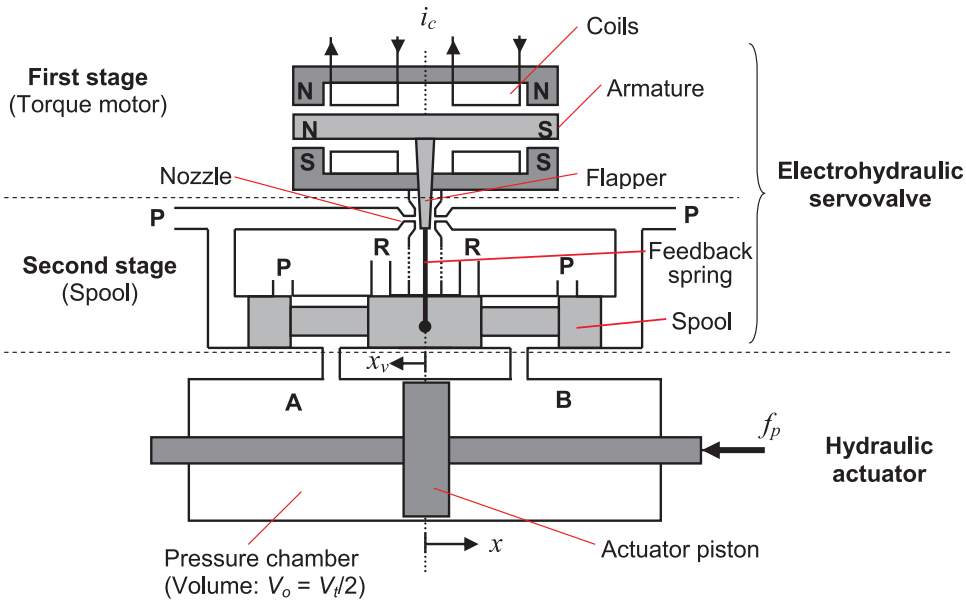


Figure 5.1: Schematic of a servo-hydraulic actuator (Carrion and Spencer, 2007)

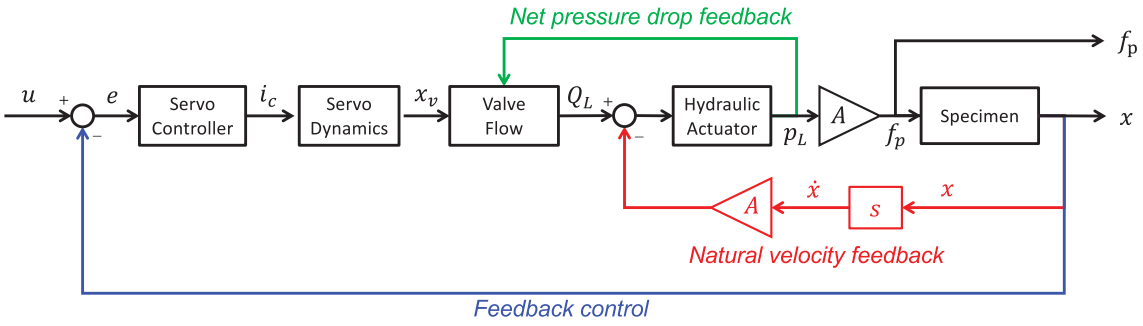


Figure 5.2: Block diagram for dynamics of servo-hydraulic actuator

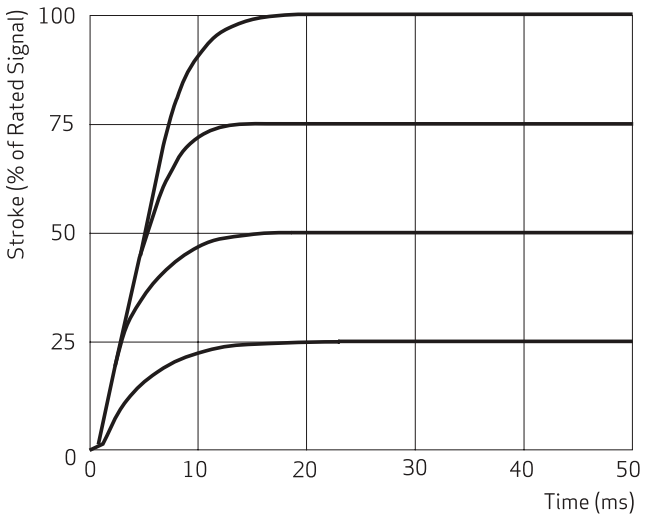


Figure 5.3: Step response of Moog G631 series servo valves (Moog, 2014)

5.2.1 Servo valve dynamics

Servo valves are very complicated devices, that are primarily used to control a source of fluid power through mechanical motion of its components. A two-stage electro-hydraulic servo valve is illustrated in Figure 5.1, where the first stage is composed of a torque motor, and the second stage consists of a spool valve. This design allows for controlled flow through the valve, by sliding the spool using a pilot valve driven by a flapper-nozzle and controlled by a feedback spring, while maintaining large hydraulic flows through the servo valve ports.

First, the dynamics associated to the torque motor and flapper-nozzle system are characterized. The torque motor is controlled by an electrical current i_c. A positive current in the motor will create a torque over the armature, which will move the flapper to the right. This will decrease the flow from the right nozzle, and increase the flow on the left nozzle. Because of differential pressure on the pilot lines, the spool will move to the left. The feedback spring will deform and will create a restoring torque over the flapper, to counteract the torque due to input current. Hence, the position of the spool is accurately controlled.

Then, the spool position response for a given input current can be approximately captured by a first-order linear model, for a frequency range from 0 to 50 Hz (Thayer, 1965):

$$x_v(s) = \frac{k_v}{1 + \tau_v s} i_c(s) \tag{5.1}$$

where x_v is the spool position; i_c is the input current; k_v is the servo gain; and τ_v is the servo rise time constant. In this study, the servo valve manufacturer (Moog Inc.) have reported a nominal rise time of $\tau_v = 18$ ms for Moog G631 series servo valves, when subjected to different step inputs, as shown in Figure 5.3. Also, the frequency response function of the Moog G631 series servo valves is shown in Figure 5.4, where a $\pm 25\%$ input amplitude was considered, with operating pressure of 3000 psi and ambient temperature of 38° C.

Subsequently, the valve flow equations are considered. As mentioned previously, the

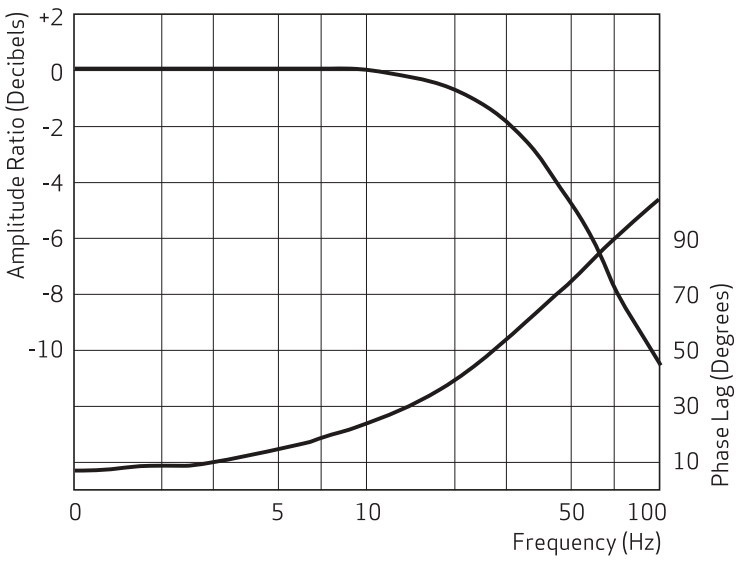


Figure 5.4: Frequency response function of Moog G631 series servo valves (Moog, 2014)

spool valve is designed to control the source of fluid power. The spool displacement will cause hydraulic fluid to flow from/to the ports that are connected to each chamber of the hydraulic actuator. Hence, the relationship between the load flow Q_L and the spool displacement x_v is expressed by the pressure-flow equation (Merritt, 1967, p.85), which is based on Bernoulli principles:

$$Q_L = C_d w x_v \sqrt{\frac{1}{\rho}\left(P_s - \frac{x_v}{|x_v|} P_L\right)} \tag{5.2}$$

where P_s is the supply (pump) pressure; P_L is the load pressure of the hydraulic actuator, which is equivalent to the pressure drop between the two actuator chambers; C_d is the coefficient of discharge of the valve orifices; w is the opening or area gradient of the valve orifices; and ρ is the fluid density. Another way to express this flow-pressure equation is in terms of the servo valve input current and rated properties:

$$Q_L = Q_R \Delta i_c \sqrt{\frac{P_S - P_L}{\Delta P_R}} \tag{5.3}$$

where Q_R is the rated flow of the servo valve for a given rated pressure drop ΔP_R, and $\Delta i_c = i_c / i_c^{\max}$ is the normalized input current. Then, the flow-pressure plot for Moog G631-3002B servo valve is shown in Figure 5.5, where $Q_R = 10$ lpm (2.5 gpm) at $\Delta P_R = 1,000$ psi.

Then, the linearized equation of the pressure-flow relationship (Merritt, 1967, p.84) for a servo valve is given by the following expression:

$$Q_L = K_q' x_v - K_c' P_L \tag{5.4}$$

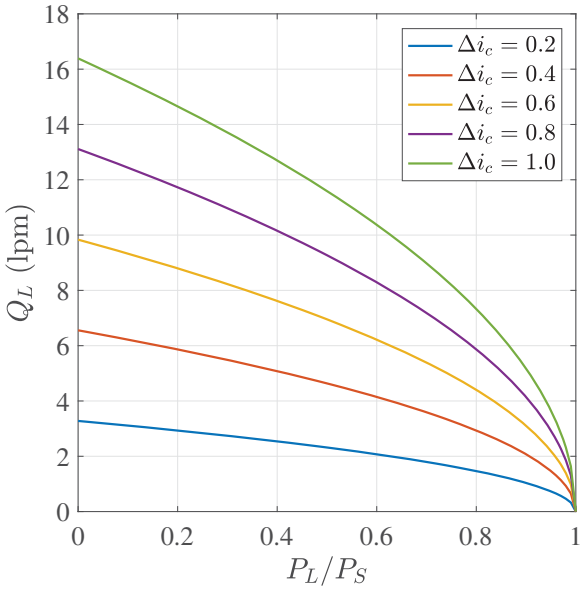


Figure 5.5: Flow-pressure plot for Moog G631-3002B servo valve

where K'_q is the valve flow gain, and K'_c is the valve flow-pressure coefficient. Both coefficients are a function of the operating point, which is the origin (i.e., $Q_L = x_v = P_L = 0$ of the pressure-flow curve.

In addition, servo valves usually present large nonlinear dynamic behavior due to hysteresis of the torque motor, frictional forces, flow saturation, and other complex flow-induced forces. Nevertheless, capturing these nonlinear effects in the servo-hydraulic actuator model is not part of the scope of this study.

5.2.2 Hydraulic actuator

The hydraulic actuator consists of hollow cylindrical tube with a piston inside that is able to slide. The piston separates the fluid inside the tube in two chambers, as shown in Figure 5.1.

Each chamber is connected to an individual port of the servo valve, thus allowing fluid pressure to be applied on the actuator chambers. The pressure of each chamber is exerted over the piston, and the net force is then applied to a test specimen connected to the piston. Hence, two governing equations are required to model a hydraulic actuator: (i) continuity (mass balance); and (ii) equilibrium (force balance).

The continuity equation is basically a flow balance inside the cylindrical tube. This equation assumes that flow has three components: (i) flow due to piston displacement; (ii) flow that escapes the actuator due to leakage; and (iii) flow stored due to fluid compressibility.

$$Q_L = A\dot{x} + C_l P_L + \frac{V_t}{4\beta_e}\dot{P}_L \tag{5.5}$$

where A is the area of the piston; C_l is the total leakage coefficient of the piston; V_t is the total volume of fluid under compression in both chambers of the actuator; and β_e is the

effective bulk modulus of the system, including hydraulic fluid, entrapped air, and mechanical compliance of the chambers (Merritt, 1967, p.148). The continuity equation can also be expressed in frequency domain as follows:

$$Q_L = Asx + C_l P_L + \frac{V_t}{4\beta_e} s P_L \tag{5.6}$$

$$P_L(s) = \frac{1}{C_l + \frac{V_t}{4\beta_e} s} \left[Q_L(s) - Asx(s) \right] \tag{5.7}$$

where the term $Asx(s)$ is commonly referred to as the "natural velocity feedback" (Dyke et al., 1995). Then, force equilibrium of the piston must be satisfied:

$$m_t \ddot{x} + c_t \dot{x} + kx + F_s = f_p \tag{5.8}$$

$$f_p = AP_L \tag{5.9}$$

where m_t is the total mass of the piston, specimen and loading attachments; c_t is the viscous damping of the actuator; k is the specimen stiffness; F_s is the force applied to the piston due to seal friction; and f_p is the net force developed by the piston.

In addition, it should be obvious that the trapped (compressed) volume of hydraulic fluid in both actuator chambers is equivalent to a "hydraulic spring". The hydraulic stiffness when the piston is centered is equal to:

$$K_h = \frac{4\beta_e A^2}{V_t} \tag{5.10}$$

Moreover, the interaction with the total inertia produces a "hydraulic natural frequency":

$$\omega_h = \sqrt{\frac{K_h}{m_t}} = \sqrt{\frac{4\beta_e A^2}{m_t V_t}} \tag{5.11}$$

This hydraulic natural frequency is an important parameter because it determines the overall speed of response of the actuator (Merritt, 1967, p.140), and the frequency response of the hydraulic actuator.

5.2.3 Feedback control

Displacement control is usually considered for servo-hydraulic actuators. Given an external command displacement u, and the actuator displacement x measured by a displacement transducer (e.g., LVDT), the servo error is defined as:

$$e(t) = u(t) - x(t) \tag{5.12}$$

The goal is to determine input current $i_c(t)$, or control signal, such that the servo error $e(t)$ is minimized. A commonly used inner-loop control algorithm for servo-hydraulic

actuators is PID control. It consists of proportional, integral, and derivative control, where each component have its own specific properties and goals. The equation for PID control is given as follows:

$$i_c(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{d}{dt} e(t) \tag{5.13}$$

where K_p, K_i, K_d are the proportional, integral, and derivative gains, respectively; and e is the servo error.

The first component in PID controllers is the proportional control, where the control signal is directly proportional to the instantaneous error $e(t)$. Large proportional gains can be used to increase the speed of transient responses, and get adequately small steady-state errors, but at the expense of large overshoots and instability. In addition, integral control can drastically reduce the steady-state error. In this case, the control signal is proportional to the accumulated servo error $\int e(t)dt$. Moreover, derivative control is used when the goal is to improve stability, speeding up transient response, and reducing overshoot; but, it has a negligible effect on steady-state errors. The control signal in this case is proportional to the rate of change of the error $\dot{e}(t)$.

Different combination of PID controllers can be considered for inner-loop feedback control of servo-hydraulic actuators. For practical applications, only proportional control is considered in this study:

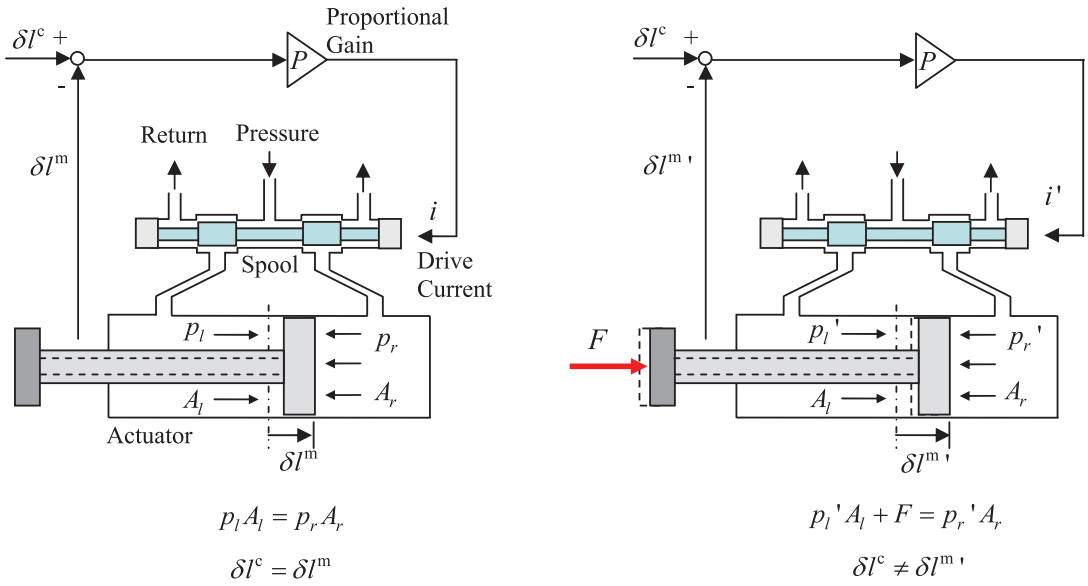
$$i_c = K_p e \tag{5.14}$$

Unfortunately, proportional control is not able to solve the problem of steady-state errors due to specimen interaction with the servo-hydraulic actuator, specially if the test specimen has a large stiffness, i.e., it experiences large reaction forces (Nakata et al., 2007). Figure 5.6 illustrates the influence of the reaction force of the test specimen on the actuator piston equilibrium. The servo-hydraulic control system consisting of the actuator and servo valve is a mechanical system by which the equilibrium of the piston is affected by the reaction force transferred through piston rod. Without the influence of the reaction force, the drive current i holds the equilibrium pressures at both sides of the piston, as shown in Figure 5.6a.

On the other hand, under the influence of relatively large reaction force F from the test specimen, the drive current i' that satisfy equilibrium of the piston will not satisfy perfect reference tracking, as shown in Figure 5.6b. Thus, a residual actuator displacement error is introduced in the servo-hydraulic system, that will be evidenced as undershoot errors. This issue can be mitigated by adding a compensation bias term, without the need of adding an integral term to the controller. More details on the issues of this steady-state bias term will be provided in Chapter 7, in particular for the experimental study of a steel column specimen with large axial stiffness.

5.2.4 Combined dynamics of servo-hydraulic actuator

By combining the servo controller equation (5.14), the first-order servo dynamics model (5.1), the linearized pressure-flow equation (5.4), the hydraulic actuator equations (5.7) and (5.8), and assumming that seal friction is minimal (i.e. $F_s \to 0$), then a fourth-order transfer



(a) Without specimen interaction (b) With specimen interaction

Figure 5.6: Effect of specimen interaction in servo error (Nakata et al., 2007)

function model can be formulated for the servo-hydraulic actuator (Carrion and Spencer, 2007):

$$G_{xu}(s) = \frac{b_0}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \tag{5.15}$$

where

$$b_0 = K_p \frac{K_q A}{K_c} \tag{5.16}$$

$$a_4 = \frac{V_t}{4\beta_e K_c} m_t \tau_v \tag{5.17}$$

$$a_3 = \frac{V_t}{4\beta_e K_c} m_t + m_t \tau_v + \frac{V_t}{4\beta_e K_c} c_t \tau_v \tag{5.18}$$

$$a_2 = m_t + \frac{V_t}{4\beta_e K_c} c_t + \frac{A^2}{K_c} \tau_v + c_t \tau_v + \frac{V_t}{4\beta_e K_c} k \tau_v \tag{5.19}$$

$$a_1 = c_t + \frac{V_t}{4\beta_e K_c} k + \frac{A^2}{K_c} + k\tau_v \tag{5.20}$$

$$a_0 = k + K_p \frac{K_q A}{K_c} \tag{5.21}$$

in which $K_q = K'_q K_v$ is the servo valve gain; and $K_c = K'_c + C_l$ is the total flow pressure coefficient.

From this white box model, a total of eleven parameters are required to estimate the servo-actuator dynamics, which are associated to properties of the servo controller, servo

valve, hydraulic actuator, and test specimen. These parameters can be experimentally identified; for example, by performing a random excitation test, and then formulating a nonlinear constrained optimization problem, the optimal parameter values that minimizes the model error can be obtained. An illustrative example of parameter estimation of a single actuator is provided by Carrion and Spencer (2007), where a nonlinear least-squares optimization was considered.

5.3 Black box model

In this section, a non-parametric frequency-domain method used to determine frequency response functions from test data will be presented for single-input-single-output (SISO) and multi-input-multi-output (MIMO) systems. The term *non-parametric* is associated to the fact that this method does not consider a finite-dimensional parameter vector for the best description of the system. In particular, SISO systems can represent the response of individual actuators; while, MIMO systems will be useful for modeling of multi-actuator loading systems.

5.3.1 Definitions

Let $x(t)$ and $y(t)$ be a pair of zero-mean, jointly wide-sense stationary (wss) stochastic processes. Then, the auto-correlation function is defined by:

$$R_{xy}(\tau) \triangleq \mathbb{E}\left[x(t)y(t+\tau)^*\right] \tag{5.22}$$

Moreover, the cross power spectral density (CPSD) is defined as the Fourier transform of the cross-correlation function:

$$S_{xy}(\omega) \triangleq \mathcal{F}\{R_{xy}\}(\omega) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} R_{xy}(\tau)e^{-i\tau\omega}d\tau \tag{5.23}$$

In practice, the discrete correlation function $R_{xy}(k)$ is approximated with a biased correlation estimate $\hat{R}_{xy}(k)$:

$$\hat{R}_x(k) = \frac{1}{N}\sum_{k=0}^{N-1-k} x(l+k)y^*(l) \quad 0 \leq k < N-1 \tag{5.24}$$

Therefore, the discrete CPSD estimate is defined as follows:

$$\hat{P}_{xy}(e^{j\omega}) = \sum_{k=-L}^{L} \hat{R}_{xy}(k)e^{-j\omega k} \quad L \leq N \tag{5.25}$$

For the case of $L \leq 10\%N$, the estimate is known as a *correlogram*. Then, for the case $L = N-1$, the estimate is known as a *periodogram*, which is primarily used for non-parametric estimation of CPSD. The periodogram is easy to compute, but it has limited ability to produce accurate estimates.

There are many methods to obtain better estimates for the periodogram; although, each method differs in its variance and resolution characteristics. The overall performance will be limited by the amount of data collected and the window chosen to reduce bias. For example, the Welch's procedure is an average modified periodogram estimate, which is asymptotically unbiased and has good performance in terms of variance and resolution. The Welch's procedure is obtained by splitting the dataset into K possibly overlapping segments of length L. Then, a window function is applied to each segment, and modified periodogram for each segment is obtained. Finally, all K periodograms are averaged. The Welch's periodogram estimate $\hat{P}_w$ is computed as follows:

$$\hat{P}_{xy}^{\text{Welch}}(e^{j\omega}) = \frac{1}{K}\sum_{k=1}^{K}\hat{P}_{xy}^{(k)}(e^{j\omega}) \tag{5.26}$$

where:

$$\hat{P}_{x}^{(k)}(e^{j\omega}) = \frac{1}{N}\sum_{n=0}^{L-1}|w(n)x^{(k)}(n)e^{-j\omega n}|^2 \tag{5.27}$$

In practical applications, the command `cpsd()` in Matlab was developed using this approach for calculations of cross-spectrums of scalar or vector signals.

5.3.2 SISO LTI systems

First, consider the single-input, single-output (SISO) relationship in time-domain for a causal, linear time-invariant (LTI) system, given by the following convolution integral:

$$y(t) = (h * u)(t) \triangleq \int_0^t h(t-\tau)u(\tau)d\tau \tag{5.28}$$

where $h(t)$ is the impulse response function of the LTI system; $u(t)$ and $y(t)$ are the input and output signals of the LTI system, respectively. The frequency-domain linear relationship can be obtained by applying the Fourier transform to the previous equation:

$$Y(\omega) = H(\omega)U(\omega) \tag{5.29}$$

where $H(\omega)$ is the frequency response function (FRF) of the LTI system. A main assumption is that the FRF is a deterministic expression for the dynamic system. Then, we can obtain relationships between the input/output (I/O) signals with the FRF. The product $Y(\omega)Y^*(\omega)$ is given by:

$$\begin{aligned} Y(\omega)Y^*(\omega) &= H(\omega)U(\omega)\left\{H(\omega)U(\omega)\right\}^* && \text{(5.30a)}\\ &= H(\omega)H^*(\omega)U(\omega)U^*(\omega) && \text{(5.30b)}\\ &= |H(\omega)|^2U(\omega)U^*(\omega) && \text{(5.30c)} \end{aligned}$$

Taking the expectation $\mathbb{E}[\cdot]$ at both sides, we obtain an expression for the output

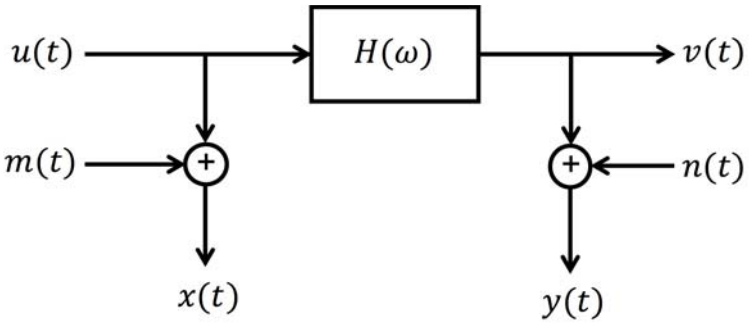


Figure 5.7: Single input, single output (SISO) dynamic system with extraneous noise

auto-spectrum $S_{yy}(\omega)$:

$$S_{yy}(\omega) = |H(\omega)|^2 S_{uu}(\omega) \tag{5.31}$$

where $S_{uu}(\omega)$ is the input auto-spectrum. Similarly, the product $Y(\omega)U(\omega)^*$ can be obtained as follows:

$$Y(\omega)U^*(\omega) = H(\omega)U(\omega)U^*(\omega) \tag{5.32}$$

Again, by taking the expectation $\mathbb{E}[\cdot]$ of the later expression, we obtain the input-output cross-spectrum $S_{yu}(\omega)$:

$$S_{yu}(\omega) = H(\omega)S_{uu}(\omega) \tag{5.33}$$

This result will be used in the following derivation of estimates for $H(\omega)$. Now, let a SISO LTI system with exogenous noise be considered for system identification, as shown in Figure 5.7, where the input and output measurements, $x(t)$ and $y(t)$, respectively, are provided by the following expressions:

$$x(t) = u(t) + m(t) \tag{5.34}$$
$$y(t) = v(t) + n(t) \tag{5.35}$$

where $u(t)$ and $v(t)$ are the I/O signals of the SISO LTI system; $m(t)$ and $n(t)$ are the input and output noises, respectively. The auto-spectrum and cross-spectrum of the I/O measurements for this problem are defined as follows:

$$S_{xx}(\omega) = S_{uu}(\omega) + S_{mm}(\omega) + S_{um}(\omega) + S_{mu}(\omega) \tag{5.36}$$
$$S_{yy}(\omega) = S_{vv}(\omega) + S_{nn}(\omega) + S_{vn}(\omega) + S_{nv}(\omega) \tag{5.37}$$
$$S_{yx}(\omega) = S_{vu}(\omega) + S_{nu}(\omega) + S_{vm}(\omega) + S_{nm}(\omega) \tag{5.38}$$

where:

$$S_{vv}(\omega) = |H(\omega)|^2 S_{uu}(\omega) \tag{5.39}$$
$$S_{vu}(\omega) = H(\omega) S_{uu}(\omega) \tag{5.40}$$

For the estimation of $H(\omega)$ using a black box approach, that is only using input/output (I/O) measurements, two main approaches are proposed: (i) assuming no input noise; and (ii) assuming no output noise. For the first approach, the main assumptions are that input noise $m(t) = 0$, and the output noise is uncorrelated (i.e., $S_{vn}(\omega) \approx 0$, $S_{nv}(\omega) \approx 0$, and $S_{nu}(\omega) \approx 0$). Then, the previous equations are simplified as follows:

$$S_{xx}(\omega) = S_{uu}(\omega) \tag{5.41}$$
$$S_{yy}(\omega) = S_{vv}(\omega) + S_{nn}(\omega) \tag{5.42}$$
$$S_{yx}(\omega) = S_{vu}(\omega) \tag{5.43}$$

Thus, the H_1 estimate for the frequency response function $H(\omega)$ of the SISO LTI system is obatined:

$$H_1(\omega) = \frac{S_{yx}(\omega)}{S_{xx}(\omega)} \tag{5.44}$$

A necessary condition for the H_1 estimate to be well-conditioned, is that the input signal $u(t)$ must be *persistently exciting*, that is:

$$S_{uu}(\omega) > 0, \qquad \forall \omega \in [0, \omega_c] \tag{5.45}$$

or in other words, the auto-spectrum $S_{uu}(\omega)$ must be invertible at least in the frequency range up to a certain cutoff frequency ω_c. Similarly, for the second approach where no output noise (i.e., $n(t) = 0$) is considered, and uncorrelated input noise (i.e., $S_{um}(\omega) \approx 0$, $S_{mu}(\omega) \approx 0$, $S_{vm}(\omega) \approx 0$) is assumed, the expressions are simplified as follows:

$$S_{xx}(\omega) = S_{uu}(\omega) + S_{mm}(\omega) \tag{5.46}$$
$$S_{yy}(\omega) = S_{vv}(\omega) \tag{5.47}$$
$$S_{yx}(\omega) = S_{vu}(\omega) \tag{5.48}$$

Finally, noting that $S_{xy}(\omega) = S^*_{yx}(\omega)$, we obtain the expression for the H_2 estimate:

$$H_2(\omega) = \frac{S_{yy}(\omega)}{S_{xy}(\omega)} \tag{5.49}$$

As explained by Rocklin et al. (1985), the estimate $H_1(\omega)$ minimizes the error due to output noise, but it could be sensitive to input noise. This situation yields an under-estimation of the true frequency response $H(\omega)$. Also, the estimate $H_2(\omega)$ minimizes the error due to input noise, but it could be sensitive to output noise; hence, the H_2 estimate

can over-estimate the true frequency response $H(\omega)$. Therefore, the relationship between these estimates and the true frequency response $H(\omega)$ is given as follows:

$$|H_1(\omega)| \leq |H(\omega)| \leq |H_2(\omega)| \tag{5.50}$$

Other estimates for $H(\omega)$ can be obtained for the general case of both correlated input and output noise acting simultaneously on the sampled data. Assuming the input and output noise sources to be uncorrelated and of equal amplitude results in the so-called H_v estimate, which is the solution of a total least-squares optimization problem (Rocklin et al., 1985), given by the following expression:

$$H_v(\omega) = \frac{S_{yx}(\omega)}{|S_{yx}(\omega)|}\sqrt{\frac{S_{yy}(\omega)}{S_{xx}(\omega)}} \tag{5.51}$$

Moreover, the coherence function is defined to measure how much of the output power is coherent (linearly related) with the input power. Then, the coherence function is given by the following expression

$$\gamma^2_{yx}(\omega) = \frac{|S_{yx}(\omega)|^2}{S_{yy}(\omega)S_{uu}(\omega)} \tag{5.52}$$

$$0 \leq \gamma^2_{yx}(\omega) \leq 1 \tag{5.53}$$

A value of $\gamma^2_{yx}(\omega) = 1$ is associated to a linear input-output relationship at the natural frequency ω, which implies a perfect correlation between random processes $x(t)$ and $y(t)$; while, a value of $\gamma^2_{yx}(\omega) = 0$ means that both signals are not correlated. Hence, if $\gamma^2_{yx}(\omega)$ is smaller than 1, it may indicate the presence of: (i) extraneous noise in the measurements, i.e., either input noise $m(t)$ or output noise $n(t)$ are not zero; (ii) leakage errors of the discrete Fourier transform (DFT) for auto and cross-spectrums; (iii) a nonlinear distortion; and/or (iv) other inputs besides $x(t)$ contributing to the output $y(t)$. Then, the coherence function could be used to determine where to trust data for curve fitting purposes (i.e. system identification). Finally, the noise spectrum for each case is given by:

$$S_{nn}(\omega) = \left[1 - \gamma^2_{yx}(\omega)\right] S_{yy}(\omega), \qquad \text{(approach 1, } H_1 \text{ estimate)} \tag{5.54}$$

$$S_{mm}(\omega) = \left[1 - \gamma^2_{yx}(\omega)\right] S_{xx}(\omega), \qquad \text{(approach 2, } H_2 \text{ estimate)} \tag{5.55}$$

5.3.3 MIMO LTI system

Consider a multi-input, multi-output (MIMO) representation of a causal, LTI dynamic system, described by the following equation:

$$\mathbf{v}(t) = (\mathbf{h} * \mathbf{u})(t) \tag{5.56}$$

where $\mathbf{u} \in \mathbb{R}^m$ is the input vector; $\mathbf{v} \in \mathbb{R}^p$ is the output vector; $\mathbf{h} \in \mathbb{R}^{p\times m}$ is the impulse response matrix. Similarly to the previous section, let the measurement signals be defined

as follows:

$$\mathbf{x}(t) = \mathbf{u}(t) + \mathbf{m}(t) \tag{5.57}$$
$$\mathbf{y}(t) = \mathbf{v}(t) + \mathbf{n}(t) \tag{5.58}$$

where $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^p$ are real-valued input and output measurement vectors, respectively; $\mathbf{m} \in \mathbb{R}^m$ and $\mathbf{n} \in \mathbb{R}^p$ are real-valued input and output noise vectors, respectively. Then, auto- and cross-spectrums of the I/O measurements for the MIMO system can be obtained as follows:

$$\mathbf{S}_{xx}(\omega) = \mathbf{S}_{uu}(\omega) + \mathbf{S}_{mm}(\omega) + \mathbf{S}_{um}(\omega) + \mathbf{S}_{mu}(\omega) \tag{5.59}$$
$$\mathbf{S}_{yy}(\omega) = \mathbf{S}_{vv}(\omega) + \mathbf{S}_{nn}(\omega) + \mathbf{S}_{vn}(\omega) + \mathbf{S}_{nv}(\omega) \tag{5.60}$$
$$\mathbf{S}_{yx}(\omega) = \mathbf{S}_{vu}(\omega) + \mathbf{S}_{nu}(\omega) + \mathbf{S}_{vm}(\omega) + \mathbf{S}_{nm}(\omega) \tag{5.61}$$

where:

$$\mathbf{S}_{vu}(\omega) = \mathbf{H}(\omega)\mathbf{S}_{uu}(\omega) \tag{5.62}$$
$$\mathbf{S}_{vv}(\omega) = \mathbf{H}(\omega)\mathbf{S}_{uu}(\omega)\mathbf{H}^*(\omega) \tag{5.63}$$

in which, $\mathbf{H}(\omega) \in \mathbb{C}^{p\times m}$ is the frequency response matrix of the MIMO system. Then, similarly to the SISO system, the H_1 and H_2 estimates are given by the following formulas (Bendat and Piersol, 2011, Chapter 7):

$$\mathbf{H}_1(\omega) = \mathbf{S}_{yx}(\omega)\mathbf{S}_{xx}^{-1}(\omega) \tag{5.64}$$
$$\mathbf{H}_2(\omega) = \mathbf{S}_{yy}(\omega)\mathbf{S}_{xy}^{+}(\omega) \tag{5.65}$$

where $\mathbf{S}_{xx} \in \mathbb{C}^{m\times m}$ and $\mathbf{S}_{yy} \in \mathbb{C}^{p\times p}$ are the input and output auto-spectrums, respectively; $\mathbf{S}_{yx} \in \mathbb{C}^{p\times m}$ is the input/output cross-spectrum; $(\cdot)^*$ is the Hermitian transpose; and $(\cdot)^+$ is the pseudo-inverse of a non-square matrix for the general case where $p \neq m$. Also, note that pseudo-inverse of a matrix $\mathbf{X}$ is defined as:

$$\mathbf{X}^+ = (\mathbf{X}^*\mathbf{X})^{-1}\mathbf{X}^* \tag{5.66}$$

Following, the multiple coherence for a multi-input, single-output (SIMO) system describes how much of the output signal power is explained by multi-input signals. The multiple coherence is defined by:

$$\gamma_{y:x}^2(\omega) = 1 - \frac{\det\left[\mathbf{S}_{yxx}(\omega)\right]}{S_{yy}(\omega)\det\left[\mathbf{S}_{xx}(\omega)\right]}, \qquad \forall\omega \tag{5.67}$$

where $\det(\mathbf{M})$ is the determinant of matrix $\mathbf{M}$, and $\mathbf{S}_{yxx} \in \mathbb{C}^{(m+1)\times(m+1)}$ is the augmented spectral matrix:

$$\mathbf{S}_{yxx}(\omega) = \begin{bmatrix} S_{yy}(\omega) & \mathbf{S}_{yx}(\omega) \\ \mathbf{S}_{xy}(\omega) & \mathbf{S}_{xx}(\omega) \end{bmatrix}, \qquad \forall\omega \tag{5.68}$$

while $\mathbf{S}_{xx} \in \mathbb{C}^{m\times m}$ is the multi-output auto-spectrum, $S_{yy} \in \mathbb{C}$ is the single-input auto-spectrum, $S_{yx} \in \mathbb{C}^{1\times m}$ and $S_{xy} \in \mathbb{C}^{m\times 1}$ are the input-output cross-spectrums.

As previously discussed for the SISO case, the multiple coherence measures the degree of linear correlation of multiple-outputs with a single-input, in the presence of output noise; a value of $\gamma^2_{y:x} = 1$ means perfect correlation, which is the case of no extraneous noise; while, $\gamma^2_{y:x} = 0$ translates to uncorrelated inputs with output. Similarly, values of multiple coherence less than unity may be associated to nonlinear behavior that is not captured by a linear representation, or additional inputs that were not considered in the experiment.

5.4 Prediction-error methods

Prediction-error methods were developed to obtain parametric models of dynamic systems, such that the error between the predictions of the model and the experimental observations are reduced. These methods usually considers a gradient-based optimization algorithm to determine the optimal parameters of the dynamic model.

Unfortunately, these methods show a tradeoff between model order and random error (variance). Higher model order (i.e., complex and very flexible model structures) can effectively reduce systematic errors (smaller bias), but at the cost of higher variance. Also, lower model order (simpler structures) can lower the model variance, but at a cost of higher systematic error (larger bias). Furthermore, likelihood functions are not always convex, i.e. there may exist several local minima. Therefore, the solution depends on good initial starting values (Ljung, 2010). In addition, the appearance of spurious modes may be associated to local minima in the optimization process. These spurious modes may produce non-minimum phase models, that could impose severe restrictions over controller classes.

The prediction-error method has been implemented in the *MFDID* toolbox for Matlab (Kim et al., 2005). The *MFDID* toolbox consists of the following procedural steps: (i) an initial estimation model is generated using a linear least-squares method; (ii) a nonlinear least-squares method (either Steiglitz-McBride and/or Gauss-Newton) is applied to improve the initial estimation model; and (iii) a maximum likelihood estimator is optimized using the Levenberg-Marquardt method.

5.4.1 Actuator-space system identification

Carrion and Spencer (2007) proposed an experimentally-based modeling approach in frequency domain for single actuators. The model structure is based on the study of white box modeling from previous sections. Thus, a fourth-order transfer function is considered:

$$G(s, \boldsymbol{\theta}) = \frac{b_0}{a_4 s^4 + a_3 s^3 + a_2 s^2 + a_1 s + a_0} \tag{5.69}$$

$$\boldsymbol{\theta} = \{a_0, a_1, a_2, a_3, a_4, b_0\}^T \tag{5.70}$$

where $s \equiv j\omega$ is the Laplace variable, ω is the natural frequency, $j = \sqrt{-1}$ is the complex number, and $\boldsymbol{\theta}$ is the model parameter vector. Also, the transfer function model is equivalent to the FRF function:

$$\hat{H}(\omega, \boldsymbol{\theta}) \equiv G(s, \boldsymbol{\theta}) \tag{5.71}$$

Then, the prediction-error method is developed such that the error between the proposed FRF model and the experimental FRF determined using black box method, is minimized:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{\omega} \left| H_{\exp}(\omega) - \hat{H}(\omega, \boldsymbol{\theta}) \right|^2 W(\omega) \tag{5.72}$$

where $\boldsymbol{\theta}^*$ is the optimal parameter vector that minimizes the cost function, and $W(\omega)$ is a weighting function. This procedure can be applied to individual actuators from a multi-actuator loading assembly, using the data collected in actuator coordinates.

5.4.2 Cartesian-space system identification

To obtain a good fit between a MIMO transfer function model and the experimental data, a good starting point is necessary, which should be sufficiently close to the global solution to avoid getting stuck in local minima during the optimization process. One strategy to choose the model order and starting guesses for all MIMO poles and zeros parameters, is to examine SISO transfer functions for each individual actuator, and then combine them into an idealized MIMO transfer function in Cartesian space by using the kinematic transformations from Chapter 4.

First, the identification of SISO transfer functions for each individual actuator is conducted, and assuming a rational polynomial function with four poles in the denominator, and no zeros in the numerator:

$$G^{(i)}(s) = \frac{b_0^{(i)}}{a_4^{(i)} s^4 + a_3^{(i)} s^3 + a_2^{(i)} s^2 + a_1^{(i)} s + a_0^{(i)}} \tag{5.73}$$

where the single "i-th" actuator transfer function $G^{(i)}(s) \in \mathbb{R}(s)$ is a rational polynomial in Laplace variable $s = j\omega$ (where ω is the natural frequency, and $j = \sqrt{-1}$ is the complex number); whereas, $b^{(i)}$ and $a_k^{(i)}$ $(i, k = \{1, 2, \ldots, 6\})$ are the scalar coefficients that are dependent on properties from the hydraulic actuator, fluid bulk modulus, servo valve, servo controller, and test specimen (Carrion and Spencer, 2007).

A main assumption is that each actuator is ideally uncoupled with the rest. Then, the MIMO transfer function in actuator space, $\mathbf{G}_{yu}^{\text{Act}}(s) \in \mathbb{R}^{6\times 6}(s)$, is chosen to be a diagonal matrix (i.e., no coupling effect is assumed in actuator space):

$$\mathbf{G}_{yu}^{\text{Act}}(s) = \text{diag}\left\{G^{(1)}(s), \ldots, G^{(6)}(s)\right\} \tag{5.74}$$

Subsequently, the initial guess for MIMO transfer function in Cartesian space, $\mathbf{G}_{yu}^{\text{Cart}}(s)$, is built by performing a similarity transformation of $\mathbf{G}_{yu}^{\text{Act}}(s)$ with the Jacobian matrix $\mathbf{J}$ from

LFKT relationships:

$$\mathbf{G}_{yu}^{\text{Cart}}(s) = \mathbf{J}^{-1}\mathbf{G}_{yu}^{\text{Act}}(s)\mathbf{J} \tag{5.75}$$

Note that transformation matrix $\mathbf{J}$ is a change of coordinates that is not necessarily an orthogonal matrix (i.e., $\mathbf{G}_{yu}^{\text{Cart}}(s)$). Also, when performing the transformation to obtain the system $\mathbf{G}_{yu}^{\text{Cart}}(s)$, the poles from the system $\mathbf{G}_{yu}^{\text{Act}}(s)$ are maintained; however, transmission zeros (i.e., roots of polynomials in the numerator) will be affected for the system in Cartesian space. Take for example the MIMO transfer matrix $\mathbf{G}_{yu}^{\text{Act}}(s) \in \mathbb{R}^{6\times 6}(s)$ in Smith form:

$$\mathbf{G}_{yu}^{\text{Act}}(s) = \frac{1}{d(s)}\mathbf{N}(s) \tag{5.76}$$

where $d(s)$ is the monic polynomial that is obtained by finding the least common denominator of all denominators in $\mathbf{G}_{yu}^{\text{Act}}(s)$; and $\mathbf{N}(s) \in \mathbb{R}^{6\times 6}(s)$ is a polynomial matrix that has the same dimensions of $\mathbf{G}_{yu}^{\text{Act}}(s)$. Further, if transfer matrix $\mathbf{G}_{yu}^{\text{Act}}(s)$ is diagonal, then $\mathbf{N}(s)$ must be diagonal as well. Then, by performing the similarity transformation:

$$\begin{aligned}\mathbf{G}_{yu}^{\text{Cart}}(s) &= \mathbf{J}^{-1}\left[\frac{1}{d(s)}\mathbf{N}(s)\right]\mathbf{J} \\ &= \frac{1}{d(s)}\left[\mathbf{J}^{-1}\mathbf{N}(s)\mathbf{J}\right]\end{aligned} \tag{5.77}$$

Therefore, the transfer matrix $\mathbf{G}_{yu}^{\text{Cart}}(s)$ will have a new assignment of transmission zeros given by the roots of numerator polynomial matrix $\mathbf{J}^{-1}\mathbf{N}(s)\mathbf{J}$, which is not a diagonal matrix anymore. Furthermore, the system in Cartesian coordinates may have a configuration with non-minimum phase (unstable) transmission zeros. This is an important detail that the control designer should take into consideration: a dynamic system with unstable transmission zeros may be difficult, or even impossible to control.

After this study, it was concluded that a sufficient structure of $\mathbf{G}_{yu}^{\text{Cart}}(s)$ for the LBCB (with six actuators) is a maximum number of 24 poles for the denominator, and 20 zeros for all numerator components of the 6×6 transfer matrix. Finally, the prediction-error method for MIMO systems is formulated:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_m \sum_n \sum_l \left| H_{mn}^{\text{exp}}(\omega_l) - \hat{H}_{mn}(\omega_l, \boldsymbol{\theta}_{mn}) \right|^2 W_{mn}(\omega_l) \tag{5.78}$$

where $\mathbf{H}^{\text{exp}}(\omega_l) = [H_{mn}^{\text{exp}}(\omega_l)]$ and $\hat{\mathbf{H}}(\omega_l, \boldsymbol{\theta}) = [\hat{H}_{mn}(\omega_l, \boldsymbol{\theta}_{mn})]$ are the experimental FRF and model FRF matrices, respectively; $\mathbf{W}(\omega_l) = [W_{mn}(\omega_l)]$ is a weighting function; and m, n, and l, are the output, input, and frequency indices, respectively. Finally, the MIMO transfer function model is given by:

$$\mathbf{G}_{yu}^{\text{Cart}}(s, \boldsymbol{\theta}^*) \equiv \hat{\mathbf{H}}(\omega, \boldsymbol{\theta}^*), \qquad (s \equiv j\omega) \tag{5.79}$$

5.5 State-space realizations

5.5.1 From state-space model to transfer function

The MIMO experimental system in Cartesian coordinates, with m inputs and p outputs, can be represented as a linear, time-invariant (LTI) state-space model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{5.80a}$$
$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \tag{5.80b}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector, $\mathbf{u} \in \mathbb{R}^m$ is the command input vector, and $\mathbf{y} \in \mathbb{R}^p$ is the response vector. Furthermore, state-space matrices have the following dimensions: $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$, and $\mathbf{D} \in \mathbb{R}^{p\times m}$.

This model can also be represented as a transfer function, which will be useful in the following sections of this paper. By performing the Laplace transform over (5.80), the MIMO transfer function becomes:

$$\mathbf{G}_{yu}(s) = \mathbf{G}_{\text{sp}}(s) + \mathbf{D} \tag{5.81}$$

where $\mathbf{G}_{yu}(s) \in \mathbb{R}^{p\times m}(s)$ is a $p \times m$ matrix of rational polynomials on $s = j\omega$, which corresponds to the Laplace variable; j is the complex number ($j = \sqrt{-1}$), and ω is the natural frequency (rad/s). The transfer function is decomposed in a strictly proper part $\mathbf{G}_{\text{sp}}(s)$ defined by:

$$\mathbf{G}_{\text{sp}}(s) = \mathbf{C}\left(s\mathbf{I} - \mathbf{A}\right)^{-1}\mathbf{B} \tag{5.82}$$

while feedforward matrix $\mathbf{D}$ is defined by:

$$\mathbf{D} = \lim_{s\to\infty} \mathbf{G}_{yu}(s) \tag{5.83}$$

On the other hand, the transfer function can be represented as a rational function matrix as follows:

$$\mathbf{G}_{yu}(s) = \frac{1}{d(s)}\mathbf{N}(s) + \mathbf{D} \tag{5.84}$$

where $d(s) \in \mathbb{R}(s)$ is a scalar monic polynomial, equal to the least common multiple of the denominators from each entry of the matrix; and $\mathbf{N}(s) \in \mathbb{R}^{p\times m}(s)$ is a matrix polynomial.

Also, recalling Cramer's rule for a matrix inverse:

$$(s\mathbf{I} - \mathbf{A})^{-1} = \frac{1}{\det(s\mathbf{I} - \mathbf{A})}\text{adj}(s\mathbf{I} - \mathbf{A}) \tag{5.85}$$

where $\det(\cdot)$ and $\text{adj}(\cdot)$ are the determinant and adjoint of a matrix, respectively. Then, the denominator and numerator of the MIMO transfer function have the following structure:

$$d(s) = \det(s\mathbf{I} - \mathbf{A}) \tag{5.86}$$

$$\mathbf{N}(s) = \mathbf{C}\left[\text{adj}(s\mathbf{I} - \mathbf{A})\right]\mathbf{B} \tag{5.87}$$

Note that the denominator $d(s)$ is the same as the characteristic polynomial of $\mathbf{A}$; thus, poles of $\mathbf{G}_{yu}(s)$ are somehow related to the eigenvalues of $\mathbf{A}$. The following theorem from Hespanha (2009) is useful to understand this relationship:

Theorem 5.1. *The poles of a real rational transfer matrix $\mathbf{G}_{yu}(s)$ is a subset of the eigenvalues of state matrix $\mathbf{A}$.*

5.5.2 From transfer function to realization

Given a proper transfer matrix $\mathbf{G}_{yu}(s) \in \mathbb{R}^{p\times m}(s)$, the realization of $\mathbf{G}_{yu}(s)$ is:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \qquad \mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \qquad \mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^m, \mathbf{y} \in \mathbb{R}^p \tag{5.88}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$, and $\mathbf{D} \in \mathbb{R}^{p\times m}$, such that the following equation holds:

$$\mathbf{G}_{yu}(s) = \mathbf{C}\left(s\mathbf{I} - \mathbf{A}\right)^{-1}\mathbf{B} + \mathbf{D} \tag{5.89}$$

Unfortunately, this realization problem does not have a unique solution, which means that a given MIMO transfer function may have multiple state-space descriptions. In addition, the number of states may significantly vary for different realizations, which is always associated to the addition of uncontrollable or unobservable dynamics, which pose a serious threat to modern feedback control design.

For example, given the following MIMO transfer function with a strictly proper, rational polynomial part:

$$\mathbf{G}_{yu}(s) = \frac{1}{d(s)}\mathbf{N}(s) + \mathbf{D} \tag{5.90}$$

where $d(s) \in \mathbb{R}(s)$ is the monic least common denominator:

$$d(s) = s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \cdots + a_1 s + a_0 \tag{5.91}$$

while $\mathbf{N}(s) \in \mathbb{R}^{p\times m}(s)$ is a polynomial matrix. Then, the strictly proper part of the MIMO system can be expanded by a linear combination of multiple SIMO systems:

$$\mathbf{G}_{\text{sp}}(s) = \frac{1}{d(s)}\left[\mathbf{N}_{n-1}s^{n-1} + \mathbf{N}_{n-2}s^{n-2} + \cdots + \mathbf{N}_1 s + \mathbf{N}_0\right] \tag{5.92}$$

where $\mathbf{N}_i \in \mathbb{R}^{p\times m}$ are constant matrices. Then, a realization can be constructed using for example a controllable canonical form (CCF):

$$\left[\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right] = \left[\begin{array}{ccccc|c} -a_{n-1}\mathbf{I}_{m\times m} & -a_{n-2}\mathbf{I}_{m\times m} & \cdots & -a_1\mathbf{I}_{m\times m} & -a_0\mathbf{I}_{m\times m} & \mathbf{I}_{m\times m} \\ \mathbf{I}_{m\times m} & \mathbf{0}_{m\times m} & \cdots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} \\ \mathbf{0}_{m\times m} & \mathbf{I}_{m\times m} & \cdots & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} & \cdots & \mathbf{I}_{m\times m} & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times m} \\ \hline \mathbf{N}_{n-1} & \mathbf{N}_{n-2} & \cdots & \mathbf{N}_1 & \mathbf{N}_0 & \mathbf{D} \end{array}\right] \tag{5.93}$$

where $\mathbf{I}_{m\times m}$ and $\mathbf{0}_{m\times m}$ are the identity and zero square matrices of order m, respectively; and the realization has state-space matrices $\mathbf{A} \in \mathbb{R}^{nm\times nm}$, $\mathbf{B} \in \mathbb{R}^{nm\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times nm}$, and $\mathbf{D} \in \mathbb{R}^{p\times m}$. In this case, the number of states increases proportional to the number of inputs; this fact is related to Theorem 5.1, where the state matrix $\mathbf{A}$ may have more eigenvalues than the poles of $\mathbf{G}_{yu}(s)$.

Then, the MIMO realization in observable canonical form (OCF) is given by:

$$\left[\begin{array}{c|c} \widetilde{\mathbf{A}} & \widetilde{\mathbf{B}} \\ \hline \widetilde{\mathbf{C}} & \widetilde{\mathbf{D}} \end{array}\right] = \left[\begin{array}{ccccc|c} -a_{n-1}\mathbf{I}_{p\times p} & \mathbf{I}_{p\times p} & \mathbf{0}_{p\times p} & \cdots & \mathbf{0}_{p\times p} & \mathbf{N}_{n-1} \\ -a_{n-2}\mathbf{I}_{p\times p} & \mathbf{0}_{p\times p} & \mathbf{I}_{p\times p} & \cdots & \mathbf{0}_{p\times p} & \mathbf{N}_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -a_1\mathbf{I}_{p\times p} & \mathbf{0}_{p\times p} & \mathbf{0}_{p\times p} & \cdots & \mathbf{I}_{p\times p} & \mathbf{N}_1 \\ -a_0\mathbf{I}_{p\times p} & \mathbf{0}_{p\times p} & \mathbf{0}_{p\times p} & \cdots & \mathbf{0}_{p\times p} & \mathbf{N}_0 \\ \hline \mathbf{I}_{p\times p} & \mathbf{0}_{p\times p} & \mathbf{0}_{p\times p} & \cdots & \mathbf{0}_{p\times p} & \mathbf{D} \end{array}\right] \tag{5.94}$$

where $\mathbf{I}_{p\times p}$ and $\mathbf{0}_{p\times p}$ are the identity and zero square matrices of order m, respectively; and the realization has state-space matrices $\widetilde{\mathbf{A}} \in \mathbb{R}^{np\times np}$, $\widetilde{\mathbf{B}} \in \mathbb{R}^{np\times m}$, $\widetilde{\mathbf{C}} \in \mathbb{R}^{p\times np}$, and $\widetilde{\mathbf{D}} \in \mathbb{R}^{p\times m}$. It can be proved that both CCF and OCF realizations are equivalent, because both realizations share the same Markov parameters:

$$\mathbf{C}\mathbf{A}^k\mathbf{B} = \widetilde{\mathbf{C}}\widetilde{\mathbf{A}}^k\widetilde{\mathbf{B}}, \qquad \forall k \geq 0 \tag{5.95}$$

But, state matrices $\mathbf{A}$ and $\widetilde{\mathbf{A}}$ from CCF and OCF realizations are not equal, and may have different number of states (when $p \neq m$). Moreover, the number of states of CCF and OCF realizations (nm and np, respectively) are clearly greater than the order of polynomial $d(s)$, which is n in this case. The growth in number of states of a realization can be attributed to spurious mode, that can be either uncontrollable or unobservable states. This spurious states could seriously hinder the design of robust controllers.

Thus, the problem of finding a realization of $\mathbf{G}_{yu}(s)$ that minimizes the number of spurious modes originated by the realization procedure is of great importance. Hence, a *minimal realization* is defined as the realization with the smaller order posible, or one with the minimum number of states possible. Moreover, the following theorem summarizes the importance of obtaining a minimal realization from a MIMO transfer function.

Theorem 5.2. *A realization of a transfer function* $\mathbf{G}_{yu}(s)$ *is minimal if and only if* $(\mathbf{A}, \mathbf{B})$ *is controlable and* $(\mathbf{A}, \mathbf{C})$ *is observable.*

A general approach to accomplish this task is through partial fractions expansion of

the strictly proper part of the MIMO transfer function:

$$\mathbf{G}_{\text{sp}}(s) = \frac{1}{\prod_{i=1}^{n}(s-p_i)}\mathbf{N}(s) = \sum_{i=1}^{n}\frac{1}{(s-p_i)}\mathbf{R}_i \tag{5.96}$$

where $\{p_i, \forall i \in [1,n]\}$ are the poles, and $\{\mathbf{R}_i, \forall i \in [1,n]\}$ are the remainder matrices. Then, the minimal realization can be obtained by using a modal form with a diagonal matrix $\mathbf{A}$, and matrices $\mathbf{B}$ and $\mathbf{C}$ obtained from factorization of $\mathbf{R}_i$. But, this approach might fail if the expansion is not possible due to its size, and if the system has repeated poles. However, another approach may still be found by general factorization of the transfer function. For example, the transfer function can be right coprime factorized as:

$$\mathbf{G}_{\text{sp}}(s) = \mathbf{N}_R(s)\mathbf{D}_R^{-1}(s) \tag{5.97}$$

where $\mathbf{N}_R(s)$ and $\mathbf{D}_R(s)$ matrices must be right coprime to ensure minimality, which means that both polynomial matrices must have no common roots. With this right-factorization, a realization can be constructed using the controllable canonical form (e.g., Varga, 1998).

5.5.3 Model reduction

An alternative way to obtain a minimal realization from a MIMO transfer function, is to reduce the order of a non-minimal realization by elimination of the uncontrollable and/or unobservable states. Hence, the proposition is as follows: given a realization $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$, if $(\mathbf{A}, \mathbf{B})$ is not controllable and/or $(\mathbf{A}, \mathbf{C})$ is not observable, then there exists a lower-order equivalent realization $(\mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r, \mathbf{D})$ for the system. In that case, if $(\mathbf{A}_r, \mathbf{B}_r)$ is controllable and $(\mathbf{A}_r, \mathbf{C}_r)$ is observable, then this lower-order realization is of minimal degree.

A general approach to obtain this lower-order realization is through balanced truncation. The problem to obtain a reduced-order model is presented as follows:

> *Given:* transfer function $\mathbf{G}(s)$ with realization $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$, where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is Hurwitz (i.e., asymptotically stable).
>
> *Find:* reduced-order transfer function $\mathbf{G}_r(s)$ with realization $(\mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r, \mathbf{D}_r)$, where $\mathbf{A}_r \in \mathbb{R}^{r\times r}$ is Hurwitz
>
> *Such that:* $\|\mathbf{G} - \mathbf{G}_r\|_\infty$ is minimized and $\dim(\mathbf{A}_r) < \dim(\mathbf{A})$

Thus, a way to obtain the reduced-order model is by defining bounds for the error norm $\|\mathbf{G} - \mathbf{G}_r\|_\infty$, such that an optimization procedure can be employed. In the following section, a methodology consisting on the inspection of singular values of the transfer function $\mathbf{G}(s)$ will be considered for balanced truncation. In essence, each singular value $\mathbf{G}(s)$ indicates how much energy is transferred from inputs to outputs of the system. Then, the states with negligible energy contributions can be removed to obtain the reduced-order model of the system.

First, the observability and controllability gramians of the system are required for the following study. Given a system $\mathbf{G}(s)$ with realization $(\mathbf{A}, \mathbf{B}, \mathbf{C})$, where $\mathbf{D} = \mathbf{0}$ is assumed

without loss of generality. The observability gramian of $(\mathbf{A}, \mathbf{C})$ is a positive semi-definite matrix defined as:

$$\mathbf{Y}_o = \int_0^\infty e^{\mathbf{A}^*\tau}\mathbf{C}^*\mathbf{C}e^{\mathbf{A}\tau}d\tau \tag{5.98}$$

If the system is observable, then $\mathbf{Y}_o$ is strictly positive definite. Also, the observability gramian can be used to estimate the amount of output energy for a given initial condition $\mathbf{x}_0$:

$$\|\mathbf{y}\|^2 = \mathbf{x}_0^*\mathbf{Y}_o\mathbf{x}_0 \tag{5.99}$$

While, the controllability gramian of $(\mathbf{A}, \mathbf{B})$ is a positive semi-definite matrix defined as:

$$\mathbf{X}_c = \int_0^\infty e^{\mathbf{A}\tau}\mathbf{B}\mathbf{B}^*e^{\mathbf{A}^*\tau}d\tau \tag{5.100}$$

Similarly, if the system is controllable, then $\mathbf{X}_c$ is strictly positive definite. In addition, if the system must be controlled to reach a given target state $\mathbf{x}_0$, the minimum control energy is given by the following expression:

$$\|\mathbf{u}_{\text{opt}}\|^2 = \mathbf{x}_0^*\mathbf{X}_c^{-1}\mathbf{x}_0 \tag{5.101}$$

On the other hand, given a non-singular matrix $\mathbf{T}$, a change of state bases is defined as

$$\mathbf{x} = \mathbf{T}\tilde{\mathbf{x}} \tag{5.102}$$

Then, the equivalent realization $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}})$ is obtained through similarity transformation

$$\tilde{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{A}\mathbf{T}, \quad \tilde{\mathbf{B}} = \mathbf{T}^{-1}\mathbf{B}, \quad \tilde{\mathbf{C}} = \mathbf{C}\mathbf{T} \tag{5.103}$$

Likewise, a similarity transformation over observability and controllability gramians is obtained:

$$\tilde{\mathbf{Y}}_o = (\mathbf{T}^{-1})^*\mathbf{Y}_o\mathbf{T}^{-1} \tag{5.104}$$

$$\tilde{\mathbf{X}}_c = \mathbf{T}\mathbf{X}_c\mathbf{T}^* \tag{5.105}$$

Furthermore, the Hankel singular values of the system $\mathbf{G}(s)$ are defined by the square roots of the eigenvalues of the Hankel matrix $\mathbf{Y}_o\mathbf{X}_c$

$$\sigma_i = \sqrt{\lambda_i\left(\mathbf{Y}_o\mathbf{X}_c\right)}, \qquad \forall i = [1, n] \tag{5.106}$$

The Hankel matrix and Hankel singular values are independent of the choice of state bases. Then, the realization $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}})$ is said to be balance if transformation matrix $\mathbf{T}$ exists such that both observability and controllability gramians are equal

$$\tilde{\mathbf{Y}}_o = \tilde{\mathbf{X}}_c = \boldsymbol{\Sigma} \tag{5.107}$$

where $\boldsymbol{\Sigma}$ is a positive definite, diagonal matrix, of Hankel singular values of transfer system $\mathbf{G}(s)$, sorted from highest to lowest values

$$\boldsymbol{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n), \quad \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n > 0 \tag{5.108}$$

Consequently, the error norm $\|\mathbf{G} - \mathbf{G}_r\|_\infty$ can be bounded from above and below to obtain a measure of accuracy of the reduced-order realization. Both upper and lower bounds are determined with information from Hankel singular values of system $\mathbf{G}(s)$:

$$\sigma_{r+1} \leq \|\mathbf{G} - \mathbf{G}_r\|_\infty \leq 2 \sum_{i=r+1}^{n} \sigma_i \tag{5.109}$$

where $r < n$ is the number of retained states from the reduced-order model, with the necessary assumption that $\sigma_{r+1} < \sigma_r$ which is true for asymptotically stable systems. Hence, the error norm is bounded by the choice of number of retained states r from the original system.

After computing a balanced realization $(\tilde{\mathbf{A}}, \tilde{\mathbf{B}}, \tilde{\mathbf{C}})$ by means of singular value decomposition, the following partition is applied over the balanced realization:

$$\tilde{\mathbf{A}} = \begin{bmatrix} \tilde{\mathbf{A}}_{11} & \tilde{\mathbf{A}}_{12} \\ \tilde{\mathbf{A}}_{21} & \tilde{\mathbf{A}}_{22} \end{bmatrix}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} \tilde{\mathbf{B}}_1 \\ \tilde{\mathbf{B}}_2 \end{bmatrix}, \quad \tilde{\mathbf{C}} = \begin{bmatrix} \tilde{\mathbf{C}}_1 & \tilde{\mathbf{C}}_2 \end{bmatrix} \tag{5.110}$$

where

$$\tilde{\mathbf{Y}}_o = \tilde{\mathbf{X}}_c = \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_2 \end{bmatrix} = \left[\begin{array}{ccc|ccc} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_r & & & \\ \hline & & & \sigma_{r+1} & & \\ & & & & \ddots & \\ & & & & & \sigma_n \end{array}\right] \tag{5.111}$$

and $\boldsymbol{\Sigma}_1$ are the retained singular values for the reduced model, which should correspond to both observable and controllable states. Then, the reduced-order balanced realization $(\mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r)$ is chosen as

$$\mathbf{A}_r = \tilde{\mathbf{A}}_{11}, \quad \mathbf{B}_r = \tilde{\mathbf{B}}_1, \quad \mathbf{C}_r = \tilde{\mathbf{C}}_1 \tag{5.112}$$

with retained Hankel singular values $\boldsymbol{\Sigma}_1 = \{\sigma_1, \ldots, \sigma_r\}$. As mentioned previously, the error norm of the balanced truncation is upper bounded by $\sum_{i=r+1}^{n} \sigma_i$. The results of this procedure are sensitive to the choice of retained states r. Therefore, multiple balanced truncations may be necessary to determine the best fit by trial and error.

5.6 Experimental results

5.6.1 Experimental FRF of the bare loading platform

First, a black-box model of the multi-actuator loading assembly without specimen interaction is obtained. This experimental FRF will serve as a baseline to compare the situation where specimen interaction affects the dynamics of the multi-actuator loading system.

An experimental modal analysis is conducted to obtain experimental frequency response functions (FRF) of the MIMO system. For this purpose, stationary, mutually uncorrelated, random signals are generated, with a cutoff frequency of 25 Hz. The command and measured Cartesian coordinates are considered as input and output signals to the system, respectively. In other words, input is defined as $\mathbf{u}(t) \triangleq \mathbf{u}^{\text{cmd}}(t)$, and output is defined as $\mathbf{y}(t) \triangleq \mathbf{u}^{\text{meas}}(t)$. Then, the H_1 estimate for the multivariate FRF from command $\mathbf{u}(t)$ to response $\mathbf{y}(t)$ is given by the following expression:

$$\mathbf{H}_{yu}(\omega_k) = \mathbf{S}_{yu}(\omega_k)\mathbf{S}_{uu}^{-1}(\omega_k), \quad k \in \{1, \ldots, N\} \tag{5.113}$$

where $\mathbf{S}_{uu}(\omega_k) \in \mathbb{C}^{6\times 6}$ is the auto-power spectral density of the commands $\mathbf{u}$; $\mathbf{S}_{yu}(\omega_k) \in \mathbb{C}^{6\times 6}$ is the cross-power spectral density of the responses $\mathbf{y}$ and commands $\mathbf{u}$; and ω_k is the discrete natural frequency. One necessary condition to obtain the experimental FRFs is that the matrix $\mathbf{S}_{uu}(\omega_k)$ should be invertible for all frequencies ω_k.

Moreover, the experimental FRF must exhibit some expected properties from the test setup, such as: (i) DC (static) response (e.g., unity gain and zero phase at $\omega_k = 0$) for diagonal components; (ii) frequency-dependent phase (i.e., measured response is indeed delayed with respect to the command signal); and (iii) dynamic coupling patterns, expressed in terms of non-zero gains for the off-diagonal components of the transfer matrix. Note that these three properties may be aggravated every time a stiffer physical specimen is connected to the test equipment.

Consequently, the experimental magnitude and phase plots for the small-scale LBCB without any test specimen attached to the loading platform are shown in Figures 5.8 and 5.9, respectively. As expected, the results for the diagonal components show unit gain, zero phase, static (DC) response, and a roll-off in magnitude with increasing frequency. Also, the phase is inversely proportional to the frequency content, meaning that the measured response is indeed delayed with respect to the command signal.

In addition, the off-diagonal components of this transfer matrix show coupling effects between Cartesian coordinates when there is no presence of a test specimen attached to the loading platform. Some coupling features are more important than others; for example, measured translation u_z^{Meas} is largely coupled with command rotation θ_y^{Cmd} for the frequency range between 10 and 20 Hz.

Furthermore, the multiple coherence is illustrated in Figure 5.10. The multiple coherence for all Cartesian outputs have values close to 1 for the frequency range between 0 to 20 Hz, which indicates that the power for each output signal is highly correlated by the multi-input signals measured in the experiment. Although, the multiple coherence of output channels u_z and θ_z have some deterioration in coherence compared to the rest output

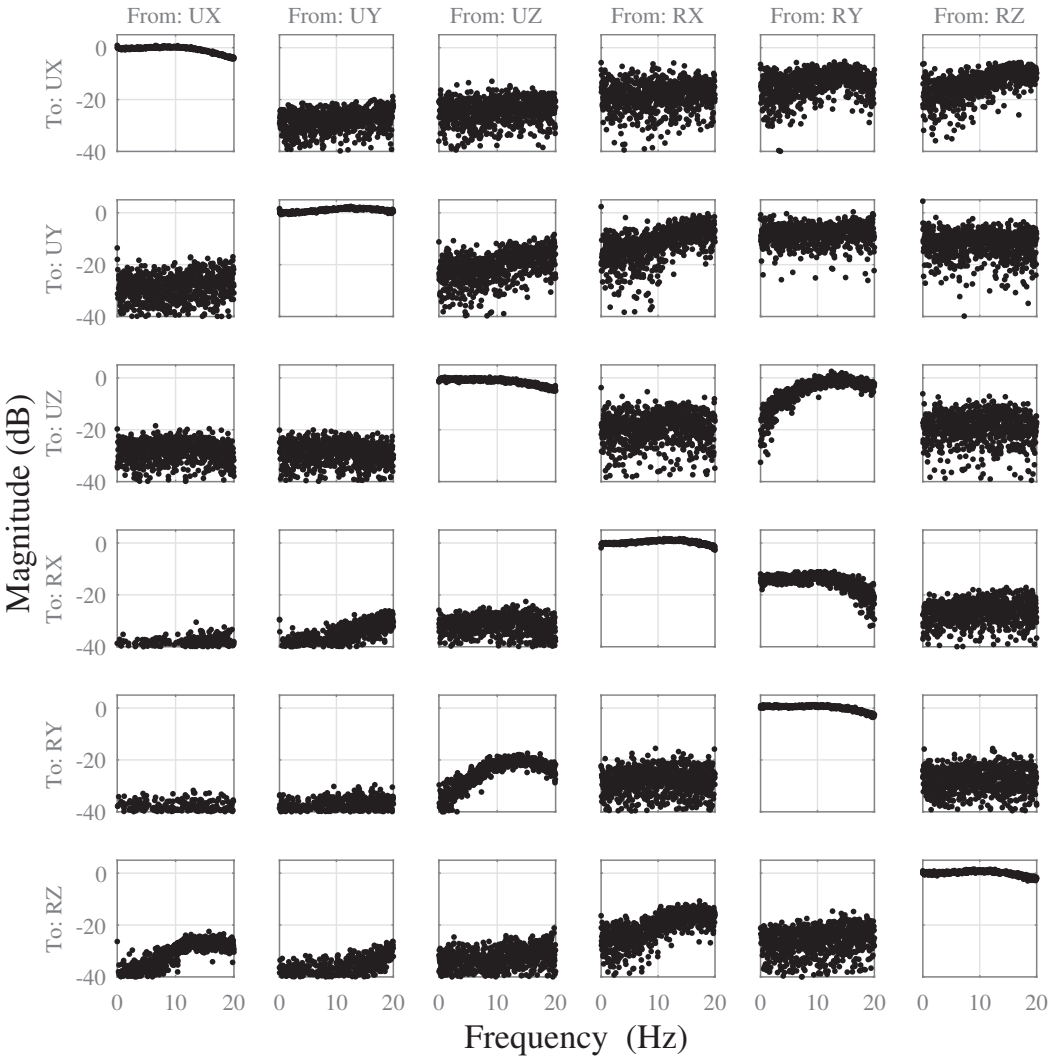


Figure 5.8: Magnitude plot of small-scale LBCB system without specimen interaction

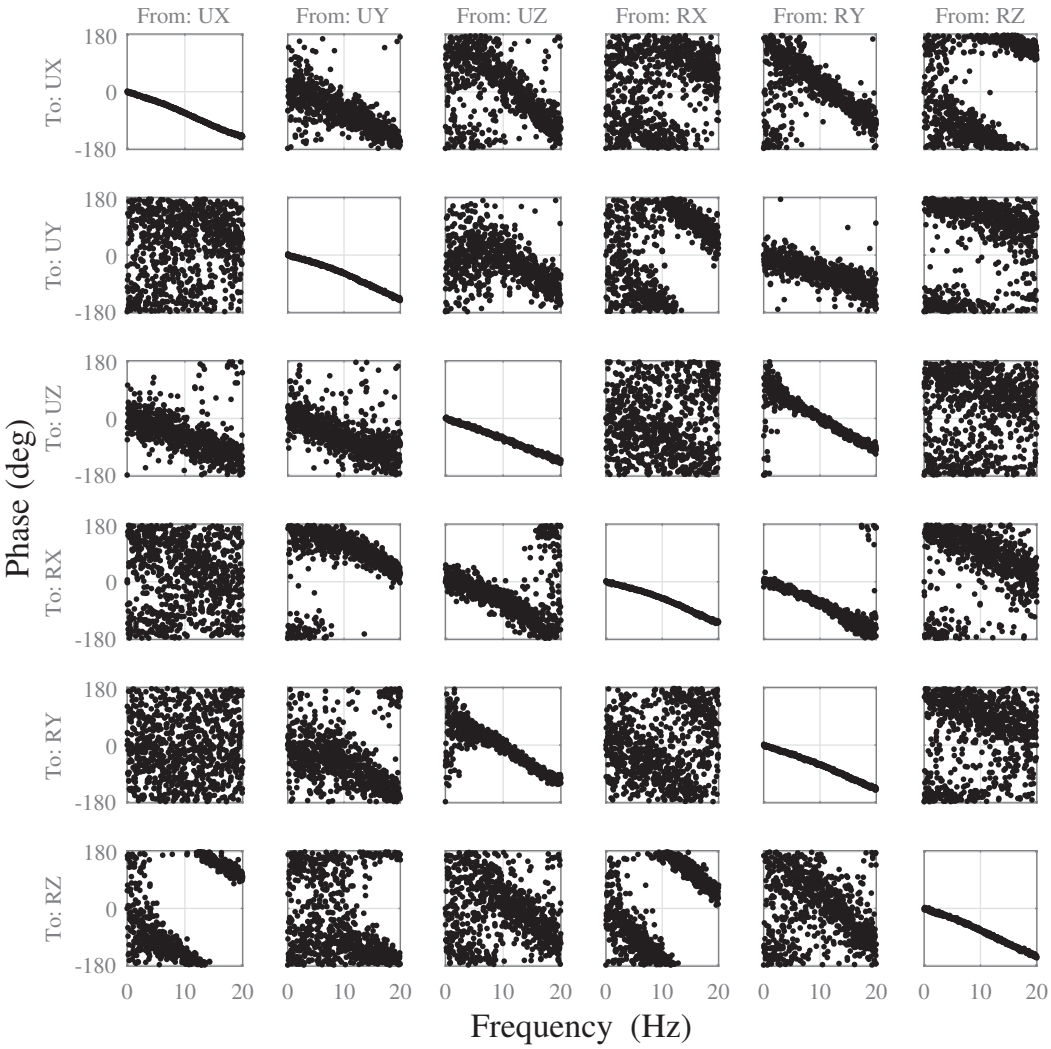


Figure 5.9: Phase plot of small-scale LBCB system without specimen interaction

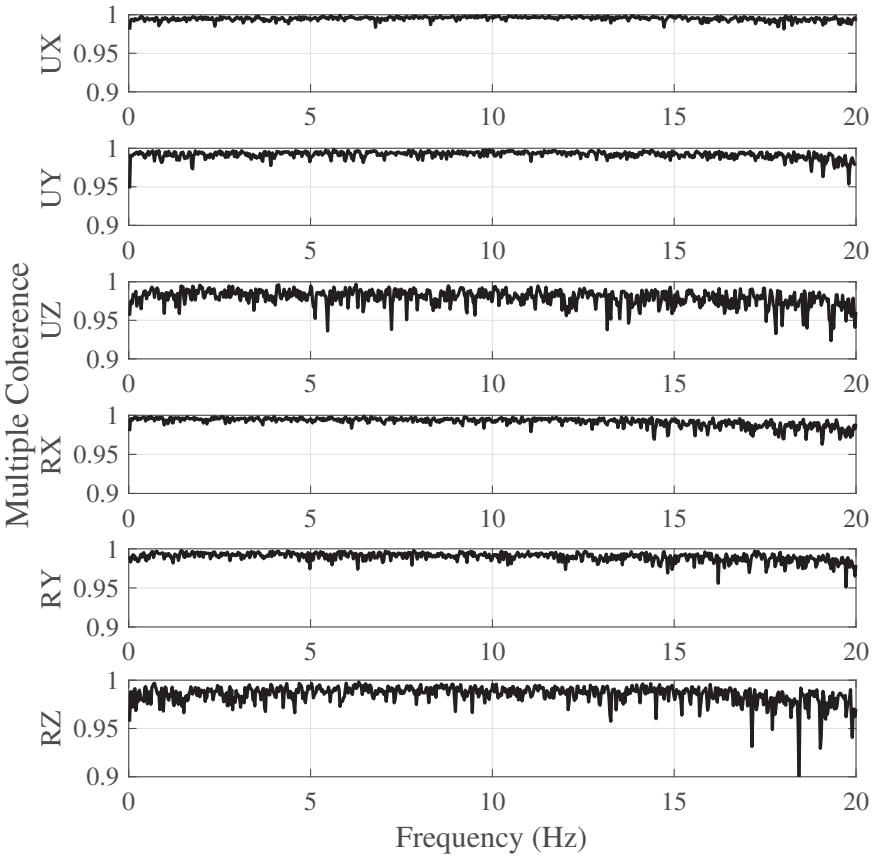


Figure 5.10: Multiple coherence plot for small-scale LBCB without specimen interaction

channels, but still within reasonable limits.

5.6.2 Experimental FRF with specimen interaction

The test specimen chosen for this experimental validation is a $\phi 1 - 1/4'' \times 18''$ ($\phi 31.75$mm$\times$ 457.2mm) steel round bar column with rigid connections at both ends. An important characteristic of the experimental specimen is the large stiffness in the axial direction compared to the actuator force capacity. Therefore, the effect of attaching the test specimen over the loading platform is equivalent to imposing a kinematic constraint on the motion of the loading assembly. Hence, the loading assembly cannot to move its motion center in the vertical direction, i.e. $u_z = 0$.

The experimental FRF of the loading assembly with specimen interaction is shown in Figures 5.11 and 5.12, and compared to the bare platform situation (i.e., no specimen interaction). Also, the effect of specimen interaction on the dynamics can be observed from the magnitude plot, specifically for the vertical translation u_z where the magnitude drops to approximately -20 dB (i.e., $\times 1/10$ reduction). Moreover, all the diagonal components have small reductions in magnitude, which are attributed to restoring forces exerted by the specimen over the loading assembly (see Section 5.2.3 for further details). Similarly, some off-diagonal components of the experimental FRF show increased degree of coupling, which was not observed for the bare loading platform case. For example, an increased coupling between command translation u_z and command rotation θ_y with measured translation u_x is recognized. Hence, any potential errors on the command rotation in Y direction could create a detrimental effect in the tracking control of the translation in the X direction.

Additionally, Figure 5.13 illustrates the effect of specimen interaction on the multiple coherence plot obtained from black-box modelling. As it was expected, the multiple coherence for output channel u_z drops significantly compared to the bare platform case. Thus, the u_z output signal cannot be explained entirely by the input signal information, and the reliability of the model to capture this output channel is deteriorated. Nevertheless, the multiple coherence for other Cartesian coordinates still remain close to 1 for the frequency range of interest.

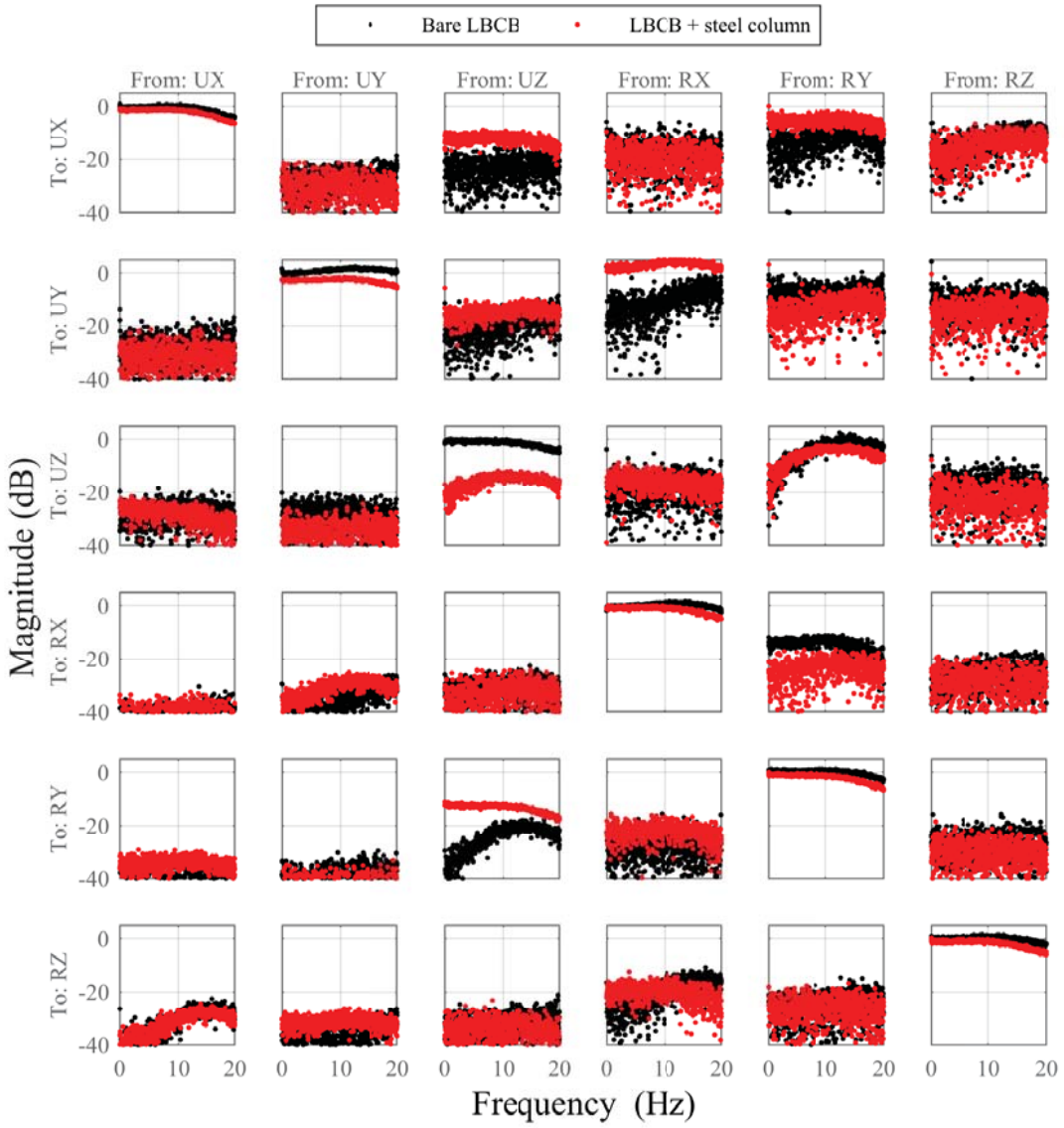


Figure 5.11: Magnitude plot for small-scale LBCB with interaction of the steel column specimen

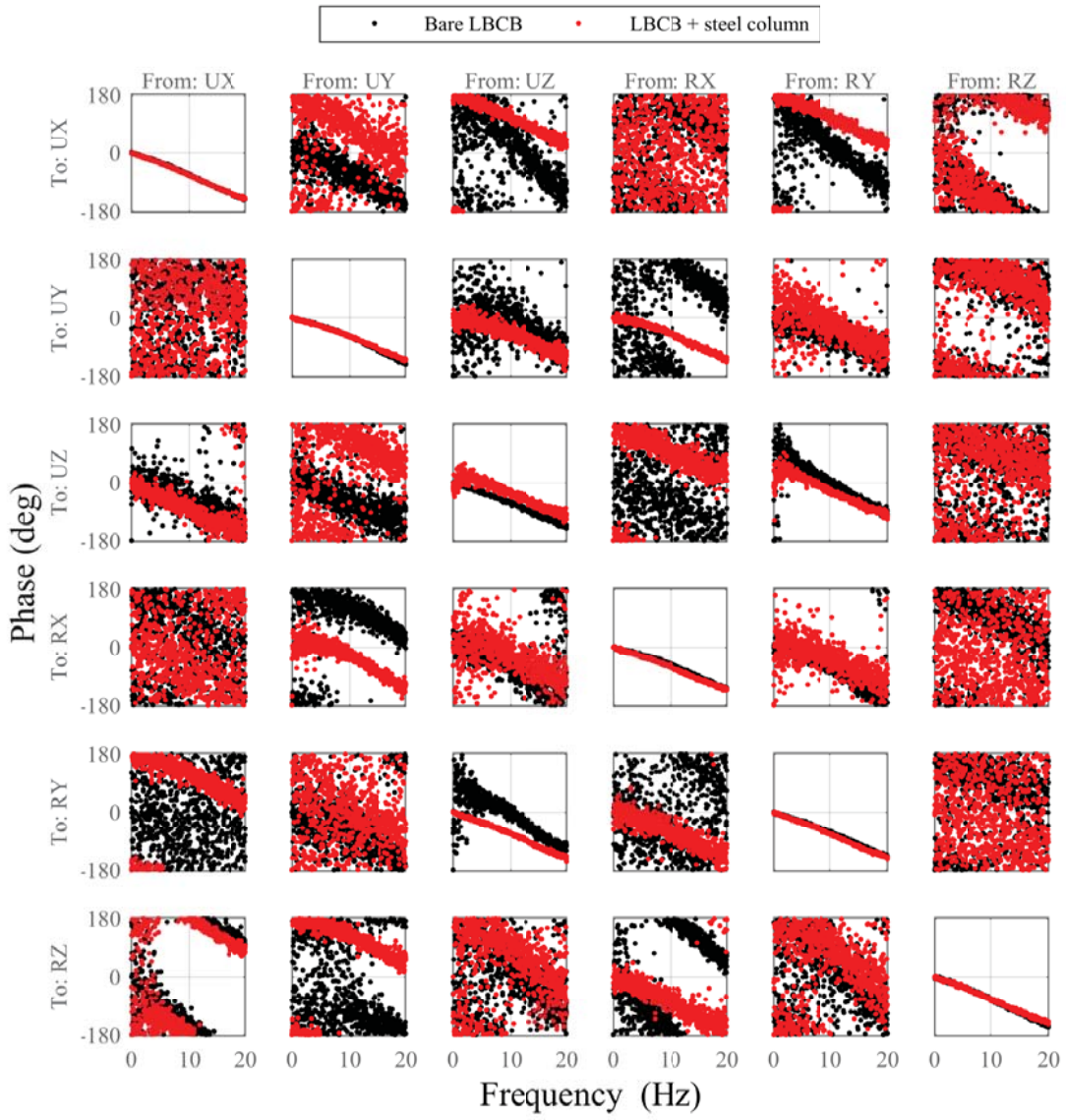


Figure 5.12: Phase plot for small-scale LBCB with interaction of the steel column specimen

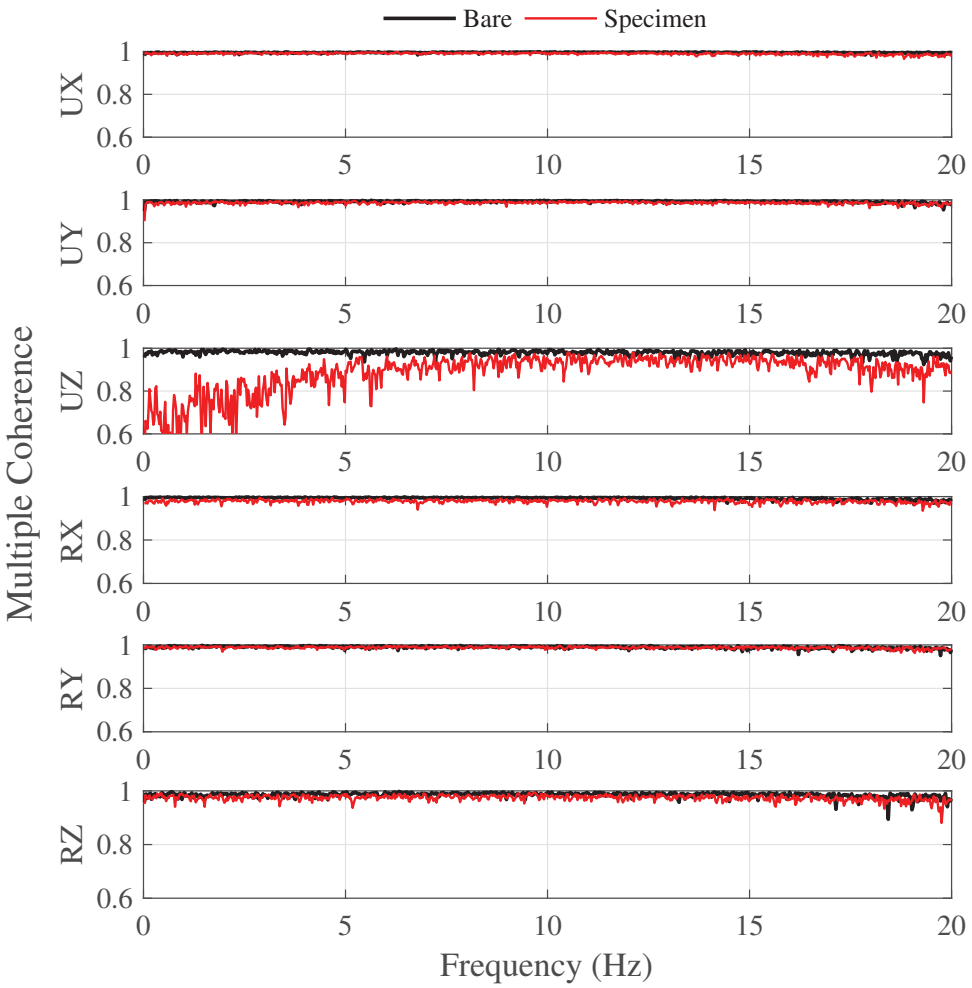


Figure 5.13: Multiple coherence plot for small-scale LBCB with interaction of the steel column specimen

5.6.3 Sensitivity of experimental FRF

The large axial stiffness of the test specimen can greatly affect the numerical precision of the models developed through system identification. For this matter, the condition number of a MIMO system is defined to measure the degree of sensitivity of the MIMO system to input perturbations:

$$\gamma_G(\omega) = \frac{\bar{\sigma}_G(\omega)}{\underline{\sigma}_G(\omega)} \tag{5.114}$$

where $\bar{\sigma}_G(\omega)$ and $\underline{\sigma}_G(j\omega)$ are the largest and smallest singular values of MIMO transfer matrix $\mathbf{G}(j\omega)$, evaluated at frequency ω.

Hence, if $\gamma_G(\omega)$ is large (e.g., $\gamma_G(\omega) > 10$), then the system is ill-conditioned, and the system may be sensitive to unstructured input uncertainty. For illustration purposes, Figure 5.14 show the singular values of the multi-actuator system for two scenarios: (i) no specimen attached (bare platform); and (ii) specimen attached. Clearly, the effect of specimen interaction will create a spread over the singular values, with a significant increase of the ratio between the largest and smallest singular value.

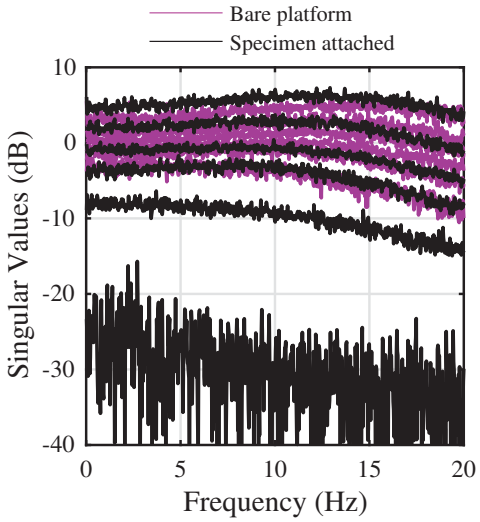


Figure 5.14: Effect of specimen interaction over singular values of multi-actuator system

The condition numbers are presented in Figure 5.15. When the system is not connected to the test specimen, the condition number is relatively small, indicating a well-conditioned problem for system identification purposes. Whereas, when the specimen is attached to the loading platform, the condition number increases drastically, and the system now is ill-conditioned.

To improve numerical conditioning, the system can be truncated by removing any weakly observable or controllable signal. In this case, the translational u_z Cartesian coordinate can be removed, which indeed is weakly observable due to the presence of the axially stiff specimen. Thus, the condition number reduces significantly for the truncation case, as seen in Figure 5.15. In addition, the effect of coordinate truncation over the singular values can be observed in Figure 5.16.

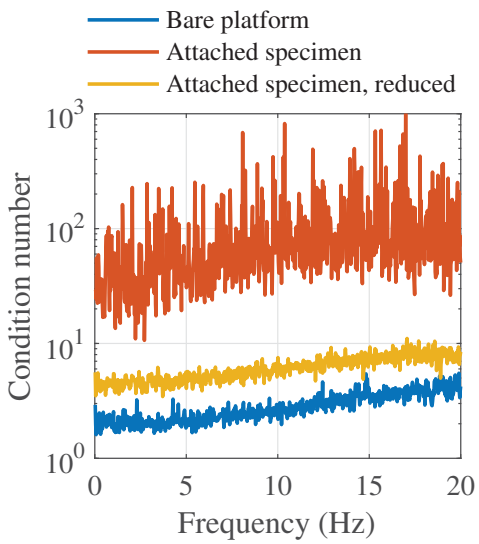


Figure 5.15: Experimental condition numbers for multi-actuator system

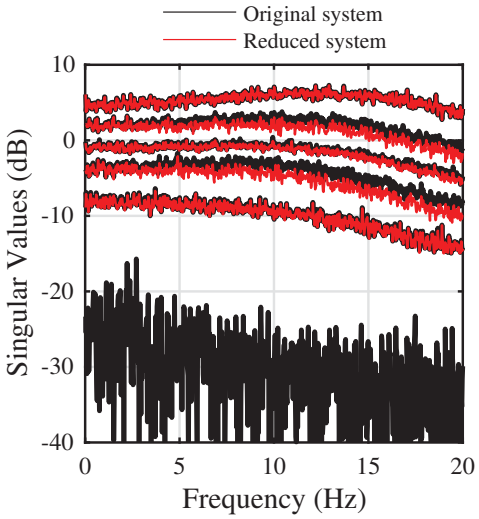


Figure 5.16: Effect of output channel removal over singular values of multi-actuator system with attached steel column specimen

5.6.4 Transfer function model in actuator space

A transfer function model for each independent actuator, including specimen interaction, was obtained by following the procedure from Section 5.4.1. For sake of simplicity, all actuator models were assumed to have a third-order structure, with three poles and no zero.

$$\mathbf{G}_{x1}(s) = \frac{1.865 \times 10^6}{s^3 + 2.860 \times 10^2 s^2 + 3.178 \times 10^4 s + 1.893 \times 10^6} \tag{5.115}$$

$$\mathbf{G}_{x2}(s) = \frac{1.495 \times 10^6}{s^3 + 2.898 \times 10^2 s^2 + 2.777 \times 10^4 s + 1.507 \times 10^6} \tag{5.116}$$

$$\mathbf{G}_{y1}(s) = \frac{2.336 \times 10^6}{s^3 + 2.928 \times 10^2 s^2 + 3.382 \times 10^4 s + 2.338 \times 10^6} \tag{5.117}$$

$$\mathbf{G}_{z1}(s) = \frac{1.864 \times 10^6}{s^3 + 3.103 \times 10^2 s^2 + 3.262 \times 10^4 s + 1.852 \times 10^6} \tag{5.118}$$

$$\mathbf{G}_{z2}(s) = \frac{2.200 \times 10^6}{s^3 + 3.156 \times 10^2 s^2 + 3.332 \times 10^4 s + 2.285 \times 10^6} \tag{5.119}$$

$$\mathbf{G}_{z3}(s) = \frac{2.226 \times 10^6}{s^3 + 2.889 \times 10^2 s^2 + 3.298 \times 10^4 s + 2.319 \times 10^6} \tag{5.120}$$

In addition, the magnitude and phase plots of single actuators are presented in Figures 5.17 and 5.18, respectively. A good fit between the transfer function model and the experimental FRF data for single actuators is obtained. Also, the results for each single actuator show static (DC) responses with almost unity gain and zero phase at $f = 0$, as expected. Likewise, the results proved that the phase angle decreases with frequency, and this relationship is not always linear for the frequency bandwidth of interest.

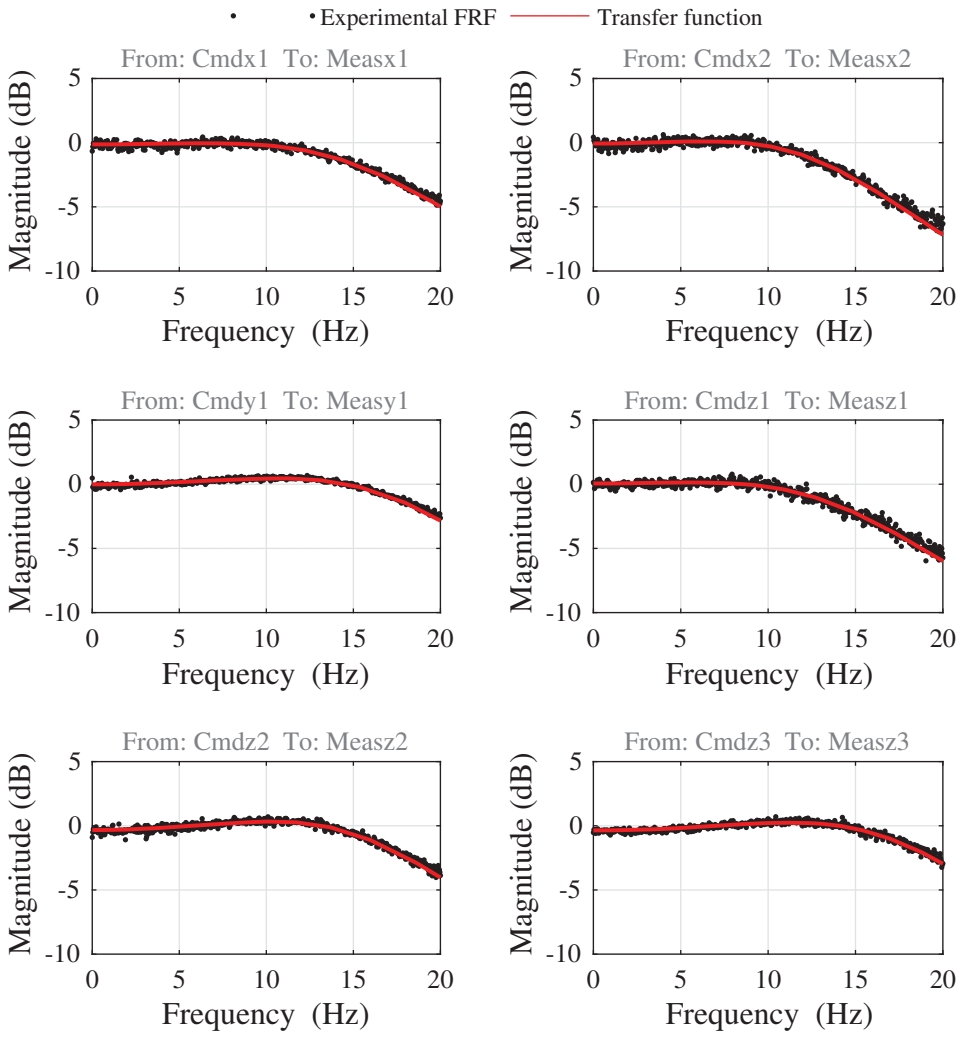


Figure 5.17: Magnitude plot of single actuators with interaction of the steel column specimen

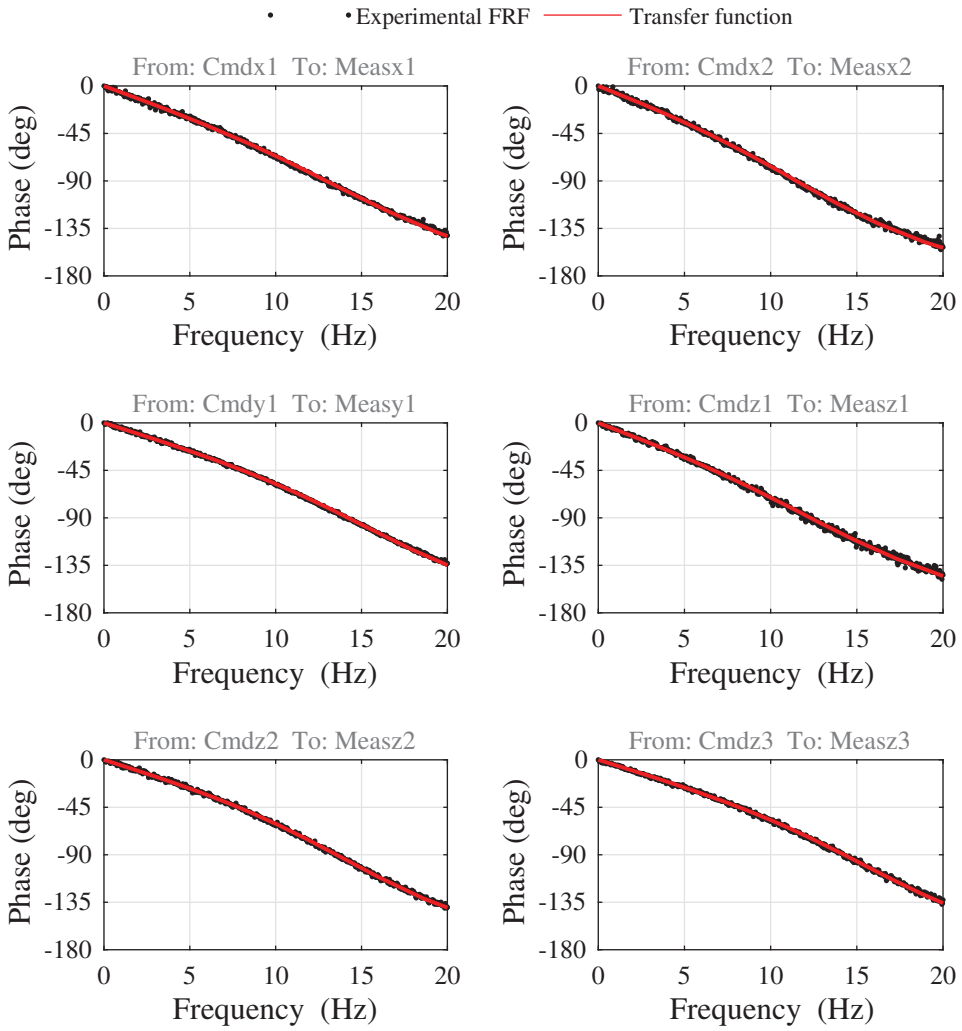


Figure 5.18: Phase plot of single actuators with interaction of the steel column specimen

5.6.5 Transfer function model in Cartesian space

Because the Cartesian coordinate u_z is weakly observable, the nominal plant model $\mathbf{G}^{\text{Cart}}(s) \in \mathbb{R}^{6\times6}(s)$ must be modified to reproduce this kinematic constraint before designing model-based controllers for RTHS testing. A practical solution that was employed was to modify the Jacobian matrix $\mathbf{J}$ required for similarity transformation from actuator to Cartesian coordinates, by including the following kinematic constraint relationship:

$$\mathbf{u} = \mathbf{T}\bar{\mathbf{u}} \tag{5.121}$$

$$\begin{Bmatrix} u_x \\ u_y \\ u_z \\ \theta_x \\ \theta_y \\ \theta_z \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} u_x \\ u_y \\ \theta_x \\ \theta_y \\ \theta_z \end{Bmatrix} \tag{5.122}$$

Then, the modified Jacobian matrix is computed as:

$$\bar{\mathbf{J}} = \mathbf{J}\mathbf{T} \tag{5.123}$$

Next, the starting guess for the transfer system in Cartesian space is obtained by similarity transformation in (5.75), using the modified Jacobian $\bar{\mathbf{J}}$. Finally, the system identification is conducted in Cartesian space as described in Section 5.4.2. The outcome is a square transfer matrix, $\bar{\mathbf{G}}^{\text{Cart}}(s) \in \mathbb{R}^{5\times5}(s)$, associated only to the controllable outputs. This modification reduces the dimension of the nominal plant and the model-based compensator, so care must be taken in the implementation of the outer-loop controller to be consistent with the target, measured, and command signal dimensions.

The magnitude and phase plots of the MIMO model in Cartesian space are presented in Figures 5.19 and 5.20, respectively. The initial Cartesian model obtained from (5.75), and the final Cartesian model after conducting the parametric optimization with the *MFDID* toolbox, are compared with the experimental FRF data. The initial model is able to represent most of the experimental FRF components of the MIMO transfer system, especially for the diagonal components. But, most off-diagonal components have large deviations that may be induced by the axial stiffness of the test specimen, which effects are not considered on the procedure to obtain the initial models from similarity transformations. Nonetheless, the system identification procedure yields a final model that matches well with the experimental FRF data, which is sufficient for controller synthesis.

The final transfer function model of the multi-actuator loading assembly in Cartesian space is included in Appendix B.1. The MIMO transfer function is expressed as a real rational matrix $\mathbf{G}_{yu}(s)$, with a 18-degree polynomial on the denominator $d(s)$, and 15-degree polynomials for each component of the numerator matrix $\mathbf{N}(s)$.

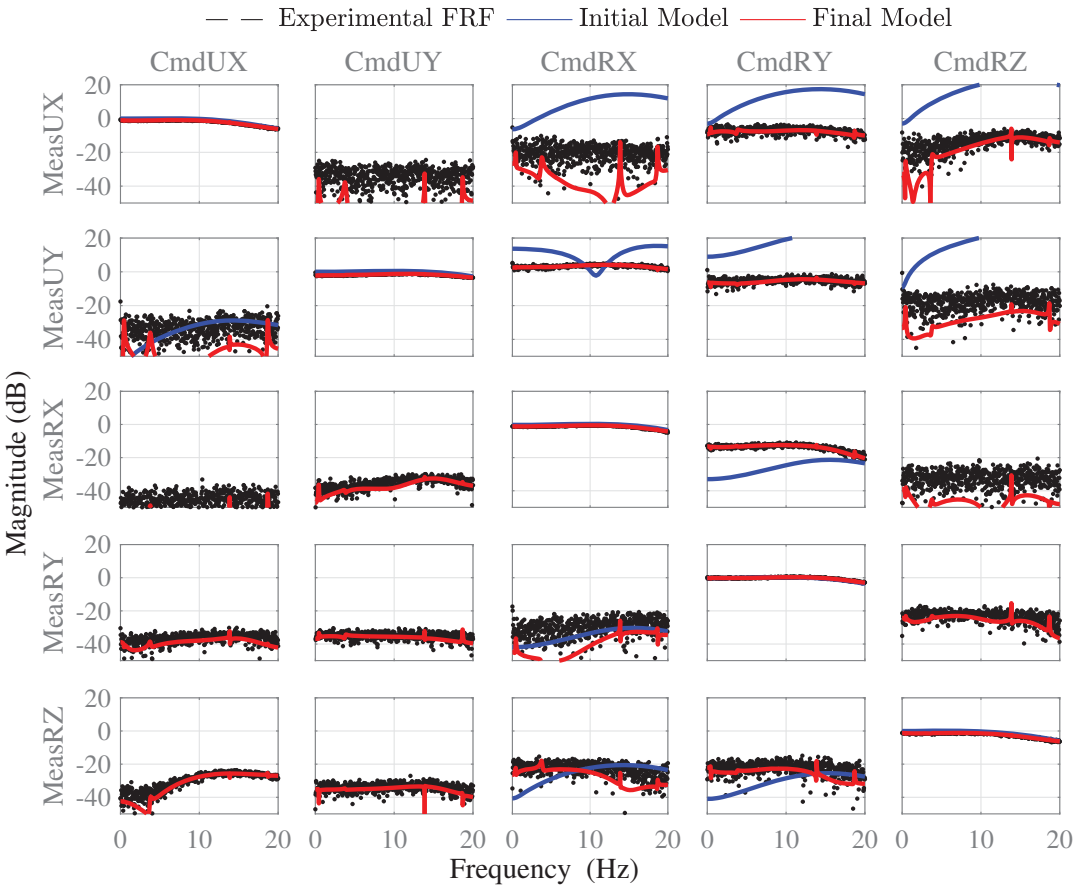


Figure 5.19: Magnitude plot of modified nominal plant with stiff axial specimen

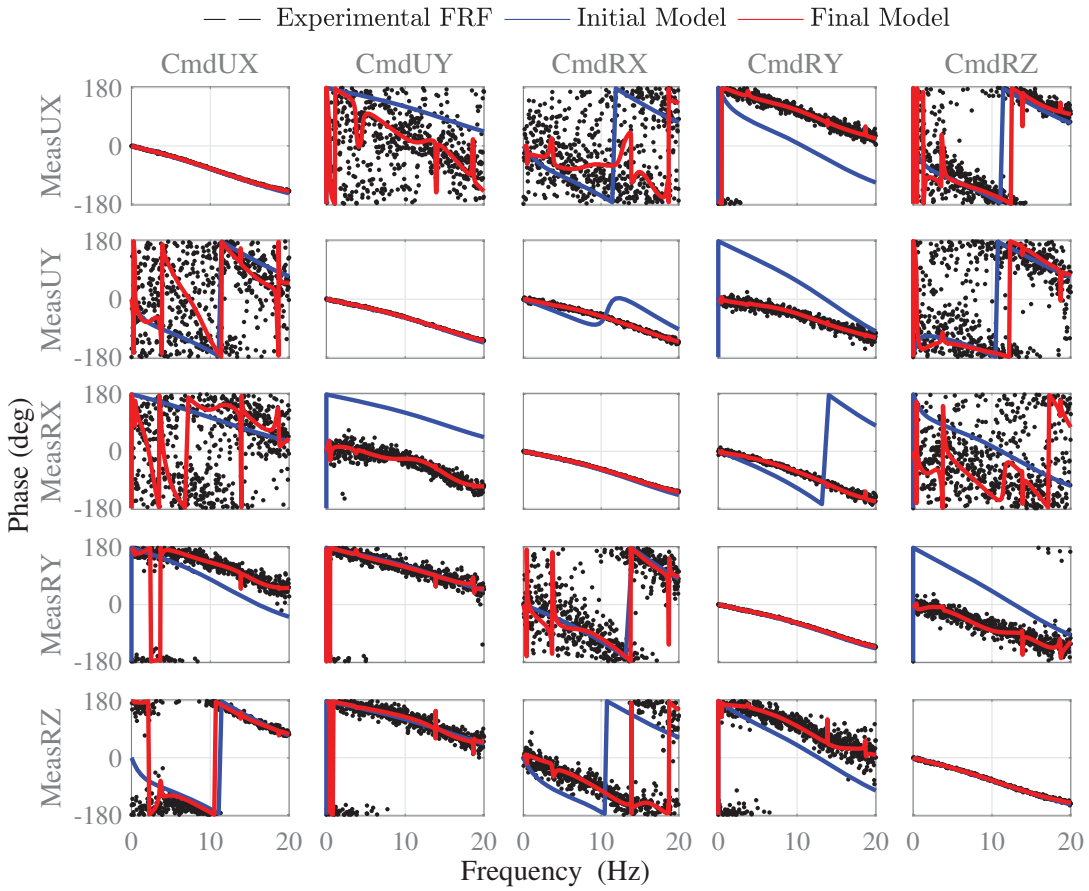
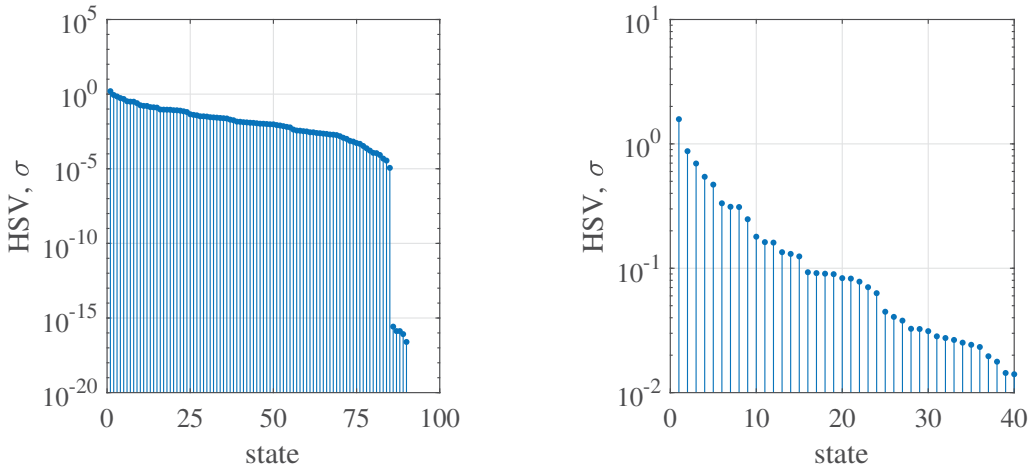


Figure 5.20: Phase plot of modified nominal plant with stiff axial specimen

5.6.6 State-space realization

A state-space realization of the identified MIMO transfer function is obtained by using the Control System Toolbox from Matlab. In particular, a controllable canonical realization $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ is obtained through the function `ss()`. But, as it was explained in Section 5.5, this realization is not necessarily minimal. The number of states of the realization is equal to 90, which is equal to the number of roots of the denominator $d(s)$ times the number of inputs of the system (i.e., $18 \times 5 = 90$). Hence, a balanced truncation is necessary to remove any unobservable or uncontrollable states from this realization.

To obtain a balanced realization, the function `balreal()` was employed for a frequency interval from 0 to 25 Hz. The Hankel singular values (HSV) of the balanced realization are presented in Figure 5.21. After a close inspection of the HSV plot, a reasonable choice of number of states to retain is exactly the number of roots from the denominator polynomial $d(s)$, i.e., choosing $r = 18$. This way the reduced-order realization will have a number of states that matches with the poles of the transfer function, in such a way the system will be close to its minimal degree.



(a) Complete view (b) Closed-up view

Figure 5.21: Hankel singular values (HSV) from balanced realization

The model reduction is performed using the `modred()` function with the `'Truncate'` option to simply remove the states and avoid the enforcement of matching DC gains. The reduced-order realization in modal canonical form is presented in Appendix B.2.

In addition, Figures 5.22 and 5.23 show the comparison between the original realization (90 states), and the reduced-order realization (18 states). The difference in terms of the diagonal components of both realizations is relatively small. Also, the off-diagonal components with predominant coupling effects were also captured well by the reduced-order realization.

Likewise, the plot of singular values of both realizations is observed in Figure 5.24. The match between realizations is very good for the first four singular values for a frequency range up to 20 Hz. While, the lowest singular value have a larger error for frequencies greater than 5 Hz.

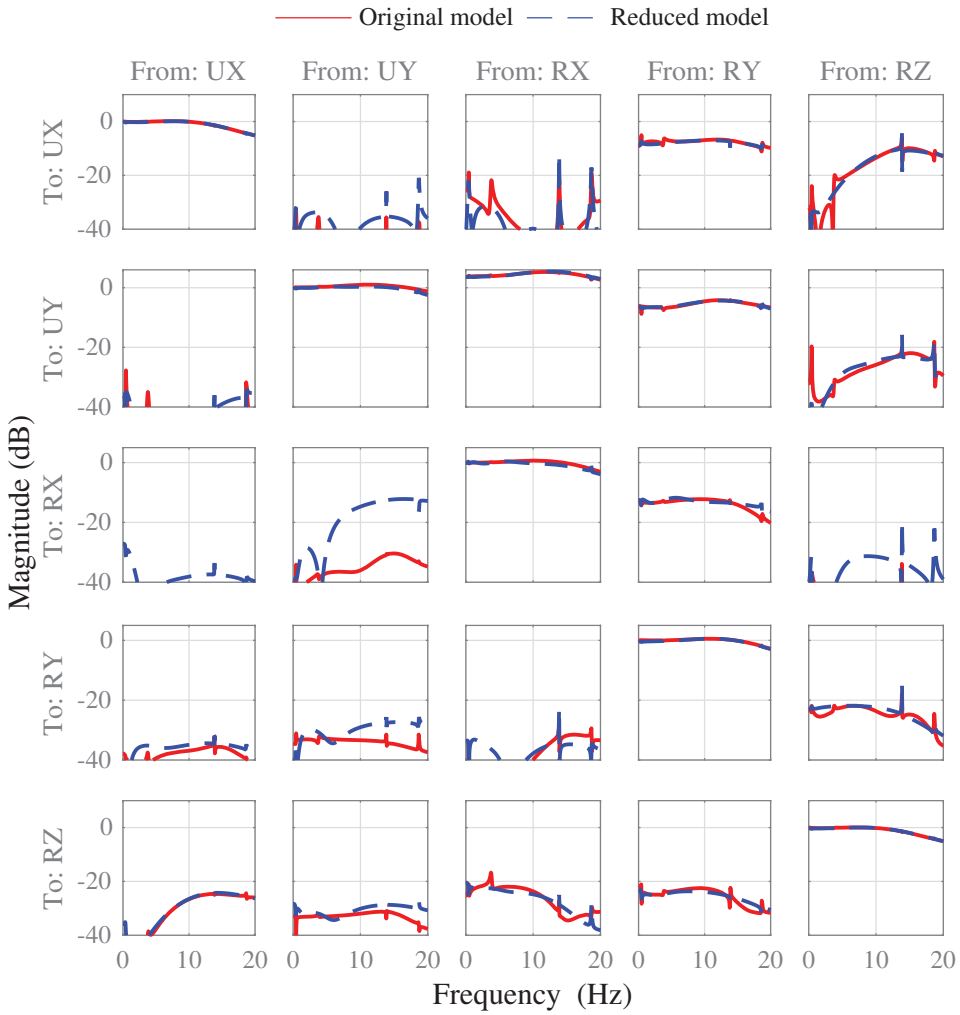


Figure 5.22: Magnitude plot of reduced-order state-space realization

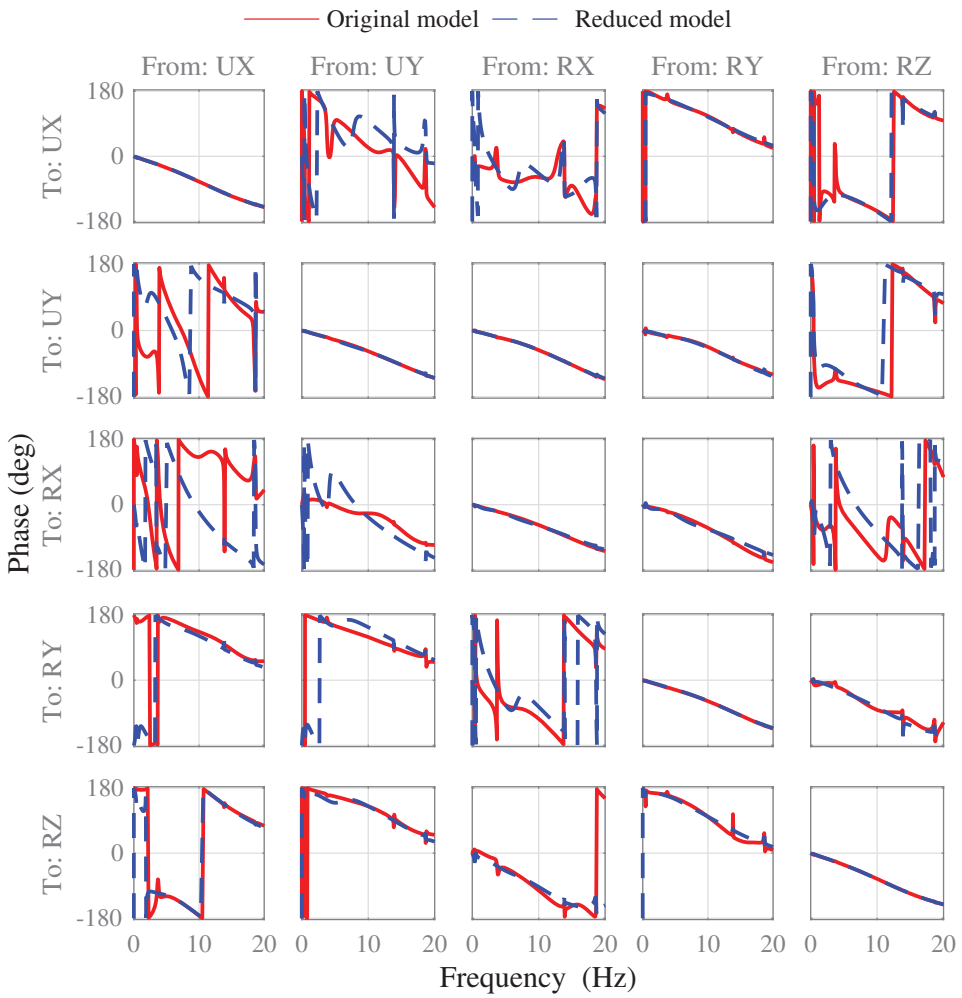


Figure 5.23: Phase plot of reduced-order state-space realization

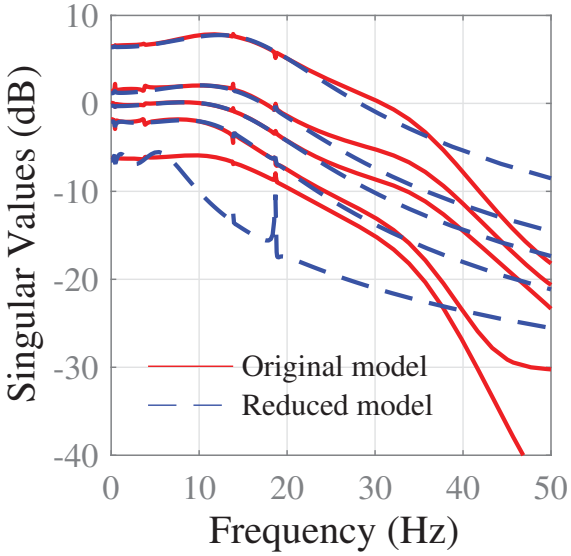


Figure 5.24: Singular values of reduced-order state-space realization

5.7 Summary

In this chapter, a linear, time-invariant, dynamical model for the multi-actuator loading assembly considered for multi-axial real-time hybrid simulation (maRTHS) testing was developed. The system is expressed in terms of a multi-input, multi-output (MIMO) model, with a structure that is based on knowledge from single servo-hydraulic dynamics. Then, a proposed methodology was to determine a black-box model of the MIMO system, by means of multivariable experimental modal analysis. Afterwards, a transfer function model is obtained through MIMO system identification.

The experimental results reveal that the task of obtaining models for individual actuators is generally straightforward; but, a MIMO model in Cartesian space is much more difficult to obtain due to the large order of the dynamical system. Nevertheless, MIMO models in transfer function and state-space formats were successfully developed for the multi-actuator loading assembly, including specimen interaction. These models can accurately capture the dynamics in Cartesian space of multi-actuator systems, something that is not possible to achieve with simpler pure time delay models commonly used in time-domain compensation techniques in RTHS testing.

Furthermore, an interesting coupling pattern is observed whenever a test specimen is connected to the loading platform, which is an effect that is very difficult to accurately predict from a pure physics-based approach. Moreover, if the coupling effects are disregarded for controller synthesis, potential systematic errors may impact the tracking performance and even stability of the maRTHS test.

Chapter 6

MODEL-BASED COMPENSATION OF MULTI-ACTUATOR SYSTEM

6.1 Problem statement

An outer-loop controller is required to reduce any reference tracking errors between target and measured displacements of the test specimen in Cartesian coordinates. The proposed design will consider a model-based compensation approach in Cartesian space, which is quite different from other RTHS solutions where single actuator feedback have been considered. The proposed Cartesian-based compensator will ensure that the correct boundary conditions are imposed into the physical specimen, because Cartesian-space control can ensure a more reliable tracking in dynamic testing (Paccot et al., 2009).

Hence, a two-stage approach is proposed for the design of the model-based controller. First, a feedforward compensator is designed, which will be the main responsible of reducing the tracking errors. Then, a feedback regulator is included to improve the stability and robustness of the controlled system. The design of both components is achieved independently to each other. Afterwards, both compensator and regulator are connected using the same architecture proposed previously in Figure 3.4. Finally, numerical simulations are required to assess the performance of the final controller design, before implementation in the real experiment.

6.2 Model-based compensation in maRTHS

To enable accurate and stable response of the RTHS test, different alternatives for time-domain or frequency-domain compensation techniques have been presented in the literature. For the purposes of this research, a model-based controller (Carrion and Spencer, 2007; Carrion et al., 2009; Phillips and Spencer, Jr., 2012, 2013a) is considered for the design of an outer-loop controller in Cartesian space.

The model-based controller is based on a feedforward-feedback control architecture, as shown in Fig 6.1. The controller provides a command signal that is determined by the following expression:

$$\mathbf{u}(t) = \mathbf{u}_{\text{ff}}(t) + \mathbf{u}_{\text{fb}}(t) \tag{6.1}$$

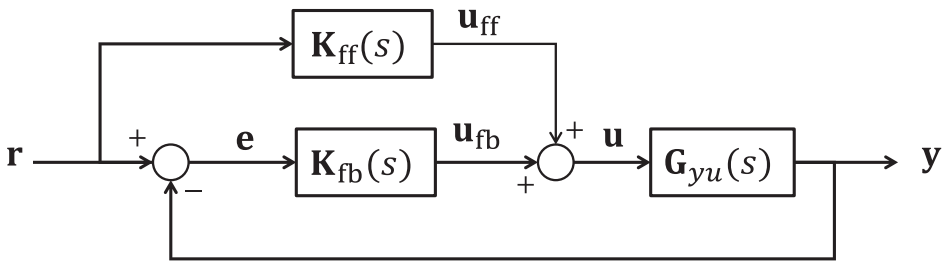


Figure 6.1: Feedforward-feedback compensator architecture for multi-axial RTHS testing

where $\mathbf{u}(t) \triangleq \mathbf{u}_{\mathrm{cmd}}(t) \in \mathbb{R}^6$ is the Cartesian command signal; $\mathbf{u}_{\mathrm{ff}}(t) \in \mathbb{R}^6$ is the feedforward command signal; and $\mathbf{u}_{\mathrm{fb}}(t) \in \mathbb{R}^6$ is the feedback command signal. Both components of the controller are calculated through a linear system with the following equations in Laplace domain:

$$\mathbf{u}_{\mathrm{ff}}(s) = \mathbf{K}_{\mathrm{ff}}(s)\mathbf{r}(s) \tag{6.2}$$

$$\mathbf{u}_{\mathrm{fb}}(s) = \mathbf{K}_{\mathrm{fb}}(s)\mathbf{e}(s) \tag{6.3}$$

where $\mathbf{r}(t) \triangleq \mathbf{u}_{\mathrm{target}}(t)$ is the target (desired) Cartesian signal; $\mathbf{y}(t) \triangleq \mathbf{u}_{\mathrm{meas}}(t)$ is the Cartesian measurement signal from the experimental setup; $\mathbf{e}(t) = \mathbf{y}(t) - \mathbf{r}(t)$ is the tracking error signal. Also, the signals in Laplace domain are given by: $\mathbf{r}(s) \triangleq \mathcal{L}\{\mathbf{r}(t)\}$, $\mathbf{y}(s) \triangleq \mathcal{L}\{\mathbf{y}(t)\}$, and $\mathbf{e}(s) = \mathbf{r}(s) - \mathbf{y}(s)$, where $\mathcal{L}\{\cdot\}$ is the Laplace transform operator.

Given this control architecture, the feedforward component of the command signal is obtained by passing the target (desired) signal originated from the numerical substructure though a linear system $\mathbf{K}_{\mathrm{ff}}(s)$ (called the feedforward controller); while, the feedback component of the command signal is obtained by passing the tracking error (i.e., the error between the target signal and measured signal from the physical plant) through a linear system $\mathbf{K}_{\mathrm{fb}}(s)$ (called the feedback controller). Hence, the feedforward compensator is the prime responsible for tracking the target displacements. On the other hand, the feedback controller provides additional robustness to the system when the feedforward is not able to perfectly track the target displacements due to potential model uncertainty.

6.3 Design of feedforward compensator

6.3.1 Inverse-based compensation

The feedforward compensator $\mathbf{K}_{\mathrm{ff}}(s)$ is designed by employing the inverse-based compensation approach. The main idea is to cancel all servo-hydraulic actuator dynamics of the system by implementing an ideal feedforward controller that is defined by the inverse model of the system.

For illustrative purposes, assuming a single-input, single-output (SISO) system with a nominal transfer function $G(j\omega)$, the ideal feedforward compensator $K_{\mathrm{ff}}^{\mathrm{ideal}}(j\omega)$ is chosen as

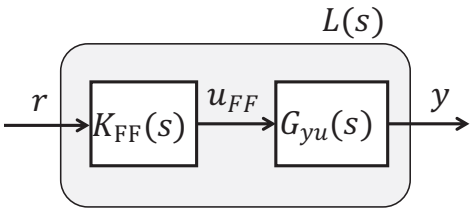


Figure 6.2: Schematic of an ideal feedforward compensator

the inverse of transfer function $G(j\omega)$:

$$K_{\text{ff}}^{\text{ideal}}(j\omega) = G^{-1}(j\omega) \tag{6.4}$$

where $j = \sqrt{-1}$ is the complex number, and ω is the natural frequency (note that $s = j\omega$ is the Laplace variable). Therefore, if we connect the ideal feedforward controller to the plant system in series, as shown in Figure 6.2, the open-loop transfer function can be written as follows:

$$\begin{aligned} L^{\text{ideal}}(j\omega) &= G(j\omega)K_{ff}^{\text{ideal}}(j\omega) \\ &= G(j\omega)G^{-1}(j\omega) \\ &= 1, \quad \forall\omega \end{aligned} \tag{6.5}$$

This result shows that in theory we could achieve perfect reference tracking using an inverse-based feedforward compensator, which yields unit-gain ($|L(j\omega)| = 1$), zero-phase ($\angle L(j\omega) = 0$), open-loop system dynamics.

Unfortunately, this ideal feedforward is quite difficult to obtain. In most cases, the plant model consists of a square, strictly proper, rational transfer matrix; therefore, the inverse is a non-proper system (i.e., it will grow unbounded as the frequency approaches infinity), and consequently the ideal feedforward cannot be implemented in real-time. Moreover, if the plant model is non-minimum phase (i.e., it has real positive zeros), then the inverse will result in real positive poles, meaning that the controller is unstable. Finally, if the system is not square (i.e., different number of inputs and outputs), then obtaining an ideal feedforward is not possible because the transfer matrix is not invertible. Some approximations for feedforward have been provided in the literature, all of them showing advantages and disadvantages (Butterworth et al., 2008; Devasia, 2002; Lee and Salapaka, 2009; Lunenburg et al., 2009; Tomizuka, 1987; Zou and Devasia, 1999). More often than not, the main disadvantage is the complexity of the different approaches that move away from the simple and intuitive idea of using feedforward in the first place.

Nevertheless, a straightforward and rational approach to approximate the inverse model for feedforward design was presented by Phillips and Spencer, Jr. (2013b). Basically, it consists in performing estimations of the experimental FRF of the inverse system:

$$\begin{aligned}\mathbf{K}_{\text{ff}}(\omega_k) &= \mathbf{G}_{uy}(\omega_k) \\ &= \mathbf{S}_{uy}(\omega_k)\mathbf{S}_{yy}^{-1}(\omega_k)\end{aligned} \tag{6.6}$$

where $\mathbf{S}_{yy} \in \mathbb{C}^{6\times6}$ is the auto-power spectral density matrix of the measurement signal $\mathbf{y}(t)$, and $\mathbf{S}_{uy} \in \mathbb{C}^{6\times6}$ is the cross-power spectral density matrix of the commands $\mathbf{u}(t)$ and measurements $\mathbf{y}(t)$, respectively. An important requirement for the output auto-spectrum $\mathbf{S}_{yy}$ is that it must be invertible to obtain the inverse FRF estimate. An example when this matrix inversion is not always possible because of numerical singularities will be discussed in Section 6.3.2

Then, a non-proper continuous transfer function can be identified from the FRF data using the *MFDID* toolbox (Kim et al., 2005). The continuous transfer function of the feedforward compensator is assumed to have only three zeros in the numerator, and no pole in the denominator:

$$\mathbf{K}_{\text{ff}}(s) = \left[k_{ij}(s)\right]_{i,j=\{1,\ldots,6\}} \tag{6.7}$$

$$k_{ij}(s) = a_{ij} + b_{ij}s + c_{ij}s^2 + d_{ij}s^3 \tag{6.8}$$

where $\mathbf{K}_{\text{ff}}(s) \in \mathbb{R}^{6\times6}(s)$ is a matrix polynomial in s, and $k_{ij}(s) \in \mathbb{R}(s)$ is a scalar polynomial in s. Then, the non-proper components of the model (i.e., time derivatives) are approximated by a backwards difference method. Hence, a discrete-time finite impulse response (FIR) filter is obtained, that will create a command signal for delay compensation purposes as a function of higher order time derivatives of the target signal (i.e., displacement, velocity, acceleration, jerk).

$$\mathbf{u}_{\text{ff}}[k] = \mathbf{K}_{\text{ff}}\mathbf{r}[k] \tag{6.9}$$

$$\mathbf{K}_{\text{ff}} = \left[k_{ij}\right]_{i,j=\{1,\ldots,6\}} \tag{6.10}$$

$$\mathbf{k}_{ij} = \begin{bmatrix} a_{ij} & b_{ij} & c_{ij} & d_{ij} \end{bmatrix} \tag{6.11}$$

$$\mathbf{r}[k] = \begin{bmatrix} \mathbf{r}_1[k] & \mathbf{r}_2[k] & \mathbf{r}_3[k] & \mathbf{r}_4[k] & \mathbf{r}_5[k] & \mathbf{r}_6[k] \end{bmatrix}^T \tag{6.12}$$

$$\mathbf{r}_j[k] = \begin{bmatrix} r_j[k] & \dot{r}_j[k] & \ddot{r}_j[k] & \dddot{r}_j[k] \end{bmatrix}, \quad j = \{1,\ldots,6\} \tag{6.13}$$

where $\mathbf{u}_{\text{ff}}[k] \in \mathbb{R}^6$ is the discrete-time feedforward control signal; $\mathbf{K}_{\text{ff}} \in \mathbb{R}^{6\times24}$ is the feedforward gain; $\mathbf{k}_{ij} \in \mathbb{R}^{1\times4}$ are the feedforward coefficients from target "j" to control "i", obtained from (6.8); $\mathbf{r}[k] \in \mathbb{R}^{24}$ is the discrete-time target signal in terms of discrete estimates at time step "k" of the displacement ($r_j[k]$), velocity ($\dot{r}_j[k]$), acceleration ($\ddot{r}_j[k]$), and jerk ($\dddot{r}_j[k]$); and $r_j[k] = r_j(kT)$ is the discrete-time target signal, with sampling period T.

In addition, the 2nd order accurate backwards difference method is used to estimate

higher-order derivatives of the target signal:

$$\dot{r}[k] \approx \frac{3r[k] - 4r[k-1] + r[k-2]}{2T} \tag{6.14}$$

$$\ddot{r}[k] \approx \frac{2r[k] - 5r[k-1] + 4r[k-2] - r[k-3]}{T^2} \tag{6.15}$$

$$\dddot{r}[k] \approx \frac{5r[k] - 18r[k-1] + 24r[k-2] - 14r[k-3] + 3r[k-4]}{2T^3} \tag{6.16}$$

6.3.2 System identification of inverse transfer function

For the particular case of specimen interaction with the multi-actuator loading assembly, it was observed that the output auto-spectrum $\mathbf{S}_{yy}(\omega_k)$ was numerically singular due to the high axial stiffness of the steel column in the vertical direction. Therefore, for the purpose of obtaining a model of the inverse MIMO system for feedforward design, the vertical translation u_z is required to be condensed and removed from the inverse model. Therefore, a transformation is applied to the original auto-spectrum, where $\mathbf{T}$ is the kinematic constraint relationship defined in (5.121).

$$\tilde{\mathbf{S}}_{yy}(\omega_k) = \mathbf{T}^T \mathbf{S}_{yy}(\omega_k) \mathbf{T} \tag{6.17}$$

Then, the modified experimental FRF of inverse system is obtained as follows:

$$\tilde{\mathbf{G}}_{uy}(\omega_k) = \mathbf{S}_{uy}(\omega_k) \tilde{\mathbf{S}}_{yy}^{-1}(\omega_k) \tag{6.18}$$

Then, the improper transfer function model for the MIMO feedforward compensator was obtained by parametric optimization using the *MFDID* toolbox. Since the transfer function is essentially a matrix polynomial, no monic denominator polynomial is required to be identified. Hence, the *MFDID* toolbox should be run with the *Structural Relationship (SR)* option disabled (Kim et al., 2005), thus allowing each component of the matrix polynomial to be optimized independently.

The final improper transfer function for MIMO feedforward compensator is included in Appendix C.1. The MIMO transfer function is expressed as a real polynomial matrix $\mathbf{K}_{\text{ff}}(s)$, where each component is a third degree polynomial. Moreover, the magnitude and phase plot of the feedforward compensator is presented in Figures 6.3 and 6.4, respectively. Both figures illustrate a good fit between the identified feedforward model and the inverse FRF estimation. In addition, the poor magnitude fit of some components (e.g., $K_{\theta_x u_x}(s)$ and $K_{\theta_x \theta_z}(s)$) is always followed by a phase without a clear trend in the inverse FRF estimation.

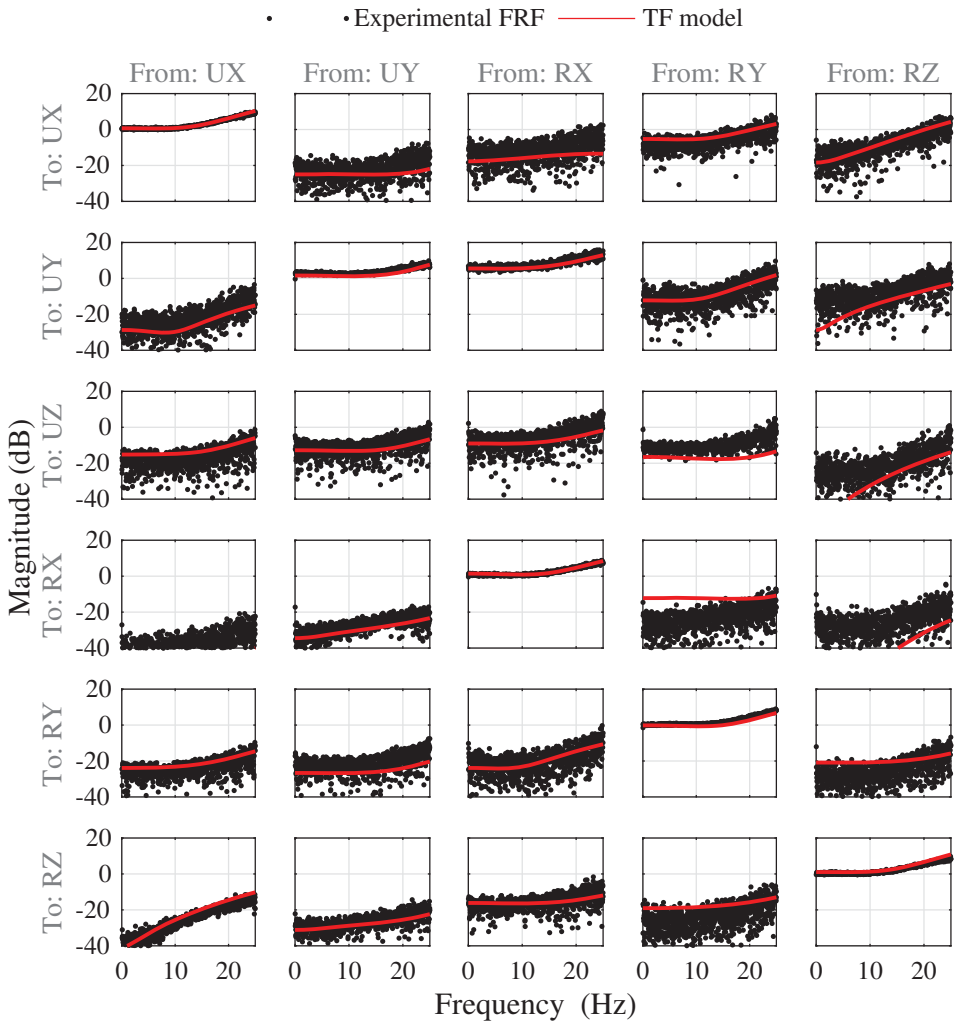


Figure 6.3: Magnitude plot of inverse MIMO system

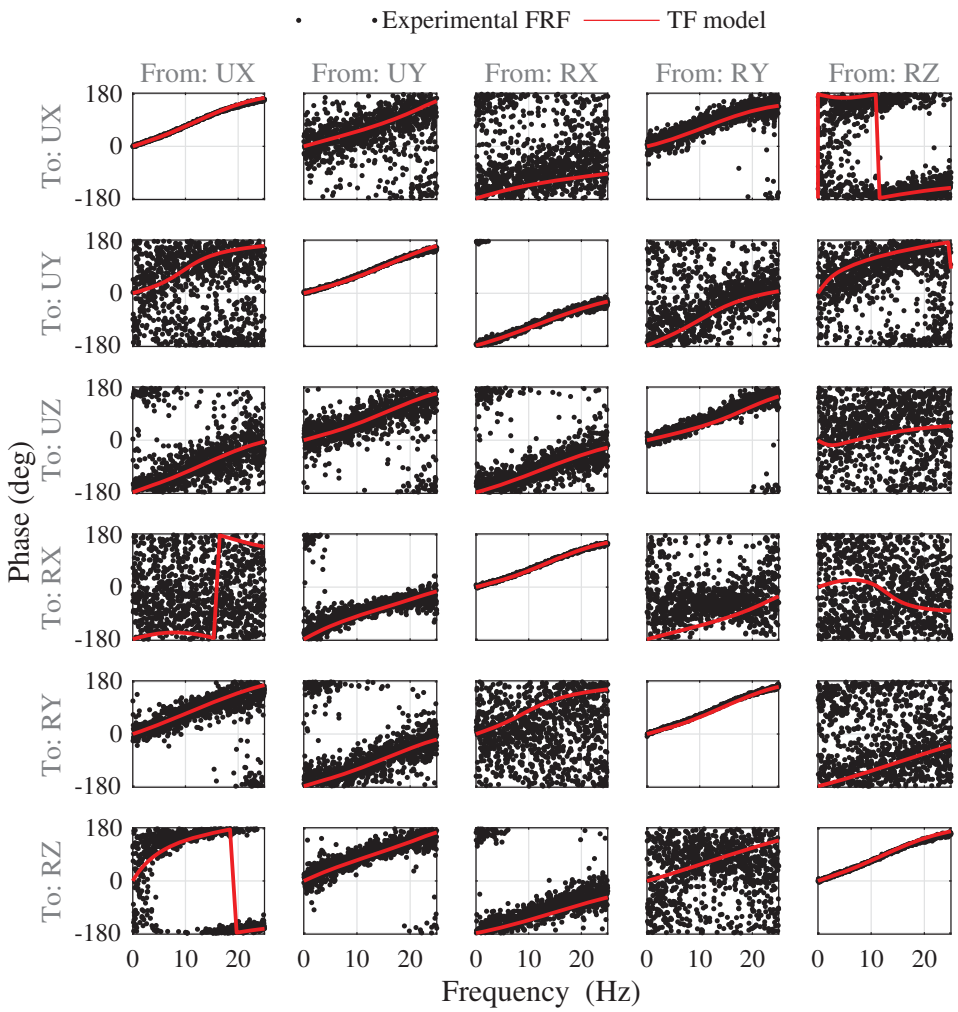


Figure 6.4: Phase plot of inverse MIMO system

6.4 Design of feedback controller

Although the feedforward compensator is the main responsible for reference tracking during the experimental test, its performance could be affected by model uncertainty. In which case, a perfect reference tracking is not achieved, especially when the model uncertainties are relatively high for a given frequency band (typically for high-frequency vibration). Therefore, a feedback regulator $\mathbf{K}_{\text{fb}}(s)$ is considered to improve the robustness of the control system.

Many approaches for feedback regulators are available in the literature. In this study, a *linear quadratic Gaussian* (LQG) regulator is chosen to design feedback controllers. The LQG regulator consists in a *linear quadratic regulator* (LQR) designed for optimal control, and a *Kalman filter* designed for optimal state estimations in the presence of disturbance and/or measurement noise. Both components are designed separately, evoking the separation principle from modern control theory. In addition, to satisfy frequency-based specifications for improved performance and robustness, a *Loop Transfer Recovery* (LTR) procedure is considered (Stein and Athans, 1987).

6.4.1 State feedback: linear quadratic regulator (LQR)

Given a continuous-time, multi-input multi-output (MIMO), linear time-invariant (LTI) system with deterministic parameters:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{6.19}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} \tag{6.20}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector; $\mathbf{u} \in \mathbb{R}^m$ is the control input vector; and $\mathbf{y} \in \mathbb{R}^p$ is the measured output vector.

The LQR problem is defined as follows (Hespanha, 2009). Find the the optimal control input $\mathbf{u}^*$ such that the following objective function is minimized

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} J(\mathbf{u}) \tag{6.21}$$

$$J(\mathbf{u}) = \int_0^\infty (\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u})dt \tag{6.22}$$

where $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{m\times m}$ are strictly positive definite matrices. Then, the solution to this optimization problem is equivalent to solving the following algebraic Ricatti equation for unknown matrix $\mathbf{P}$:

$$\mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} + \mathbf{Q} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} = \mathbf{0} \tag{6.23}$$

where $\mathbf{P}$ is a symmetric, positive definite matrix. Then, the optimal feedback gain $\mathbf{K}$ is defined as

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \tag{6.24}$$

and the optimal control input is calculated as

$$\mathbf{u}^* = -\mathbf{K}\mathbf{x} \tag{6.25}$$

In addition, since $\mathbf{y} = \mathbf{C}\mathbf{x}$, the objective criterion (6.42) can be alternatively expressed in terms of controlled output $\mathbf{y}$:

$$J(\mathbf{u}) = \int_0^\infty (\mathbf{y}^T\bar{\mathbf{Q}}\mathbf{y} + \rho\mathbf{u}^T\bar{\mathbf{R}}\mathbf{u})dt \tag{6.26}$$

where $\bar{\mathbf{Q}} \in \mathbb{R}^{p\times p}$; $\bar{\mathbf{R}} \in \mathbb{R}^{m\times m}$, and ρ a positive constant. This special criterion is related to the former criterion by

$$\mathbf{Q} = \mathbf{C}^T\bar{\mathbf{Q}}\mathbf{C}, \quad \mathbf{R} = \rho\bar{\mathbf{R}} \tag{6.27}$$

A good starting point for the choice of matrices $\bar{\mathbf{Q}}$ and $\bar{\mathbf{R}}$ is given by Bryson's rule (Franklin et al., 2015):

$$\bar{Q}_{ii} = \frac{1}{\text{maximum acceptable value of } y_i^2} \tag{6.28}$$

$$\bar{R}_{jj} = \frac{1}{\text{maximum acceptable value of } u_j^2} \tag{6.29}$$

Also, the parameter ρ is established as a trade-off between two conflicting goals: (i) for ρ very small, the controlled output energy is minimized, at the expense of large control input; and (ii) for ρ very large, the control input energy is minimized, at the expense of large controlled output signal. Thus, different values of ρ can be considered by control designer to satisfy some design specifications, as it will be explained in Section 6.4.5.

6.4.2 Kalman filter

Given a continuous-time MIMO LTI system with exogenous noise:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{B}_w\mathbf{w} \tag{6.30a}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{v} \tag{6.30b}$$

where $\mathbf{w} \in \mathbb{R}^q$ and $\mathbf{v} \in \mathbb{R}^p$ are input disturbance and output noise, respectively. Both exogenous noises are modeled as uncorrelated, zero-mean, stationary Gaussian processes:

$$\mathbb{E}[\mathbf{w}(t)] = 0, \quad \mathbb{E}[\mathbf{w}(t)\mathbf{w}^T(\tau)] = \mathbf{Q}_w\delta(t-\tau) \tag{6.31}$$

$$\mathbb{E}[\mathbf{v}(t)] = 0, \quad \mathbb{E}[\mathbf{v}(t)\mathbf{v}^T(\tau)] = \mathbf{R}_v\delta(t-\tau) \tag{6.32}$$

where $\mathbf{Q}_w \in \mathbb{R}^{q\times q}$ and $\mathbf{R}_v \in \mathbb{R}^{p\times p}$ are symmetric, positive definite matrices.

Thus, the state and output measurement vectors of the linear system can be regarded as stochastic processes. Hence, the Kalman filter problem is formulated as follows. Find the optimal state estimate $\hat{\mathbf{x}}^*$ such that the following objective function is minimized

$$\hat{\mathbf{x}}^* = \arg\min_{\hat{\mathbf{x}}} J(\hat{\mathbf{x}}) \tag{6.33}$$

$$J(\hat{\mathbf{x}}) = \lim_{t\to\infty} \mathbb{E}\left[(\mathbf{x}-\hat{\mathbf{x}})(\mathbf{x}-\hat{\mathbf{x}})^T\right] \tag{6.34}$$

The solution to this optimization problem is equivalent to solving the corresponding algebraic Ricatti equation for unknown matrix $\mathbf{P}$:

$$\mathbf{AP} + \mathbf{PA}^T + \mathbf{B}_w\mathbf{Q}_w\mathbf{B}_w^T - \mathbf{PC}^T\mathbf{R}_v^{-1}\mathbf{CP} = \mathbf{0} \tag{6.35}$$

where $\mathbf{P}$ is a symmetric, positive definite matrix. Then, the optimal state estimate is obtained by solving the following differential equation

$$\dot{\hat{\mathbf{x}}} = (\mathbf{A}-\mathbf{LC})\hat{\mathbf{x}} + \mathbf{Bu} + \mathbf{Ly} \tag{6.36}$$

in which $\mathbf{L}$ is the Kalman gain

$$\mathbf{L} = \mathbf{PC}^T\mathbf{R}_v^{-1} \tag{6.37}$$

Consequently, the choices of covariance matrices $\mathbf{Q}_w$ and $\mathbf{R}_v$ can be done using the statistics from experimental data.

6.4.3 Output feedback: linear quadratic Gaussian (LQG) regulator

Given a continuous-time MIMO LTI system with exogenous white noise:

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} + \mathbf{B}_w\mathbf{w} \tag{6.38a}$$
$$\mathbf{y} = \mathbf{Cx} + \mathbf{v} \tag{6.38b}$$

where

$$\mathbb{E}[\mathbf{w}(t)] = 0, \quad \mathbb{E}[\mathbf{w}(t)\mathbf{w}^T(\tau)] = \mathbf{Q}_w\delta(t-\tau) \tag{6.39}$$

$$\mathbb{E}[\mathbf{v}(t)] = 0, \quad \mathbb{E}[\mathbf{v}(t)\mathbf{v}^T(\tau)] = \mathbf{R}_v\delta(t-\tau) \tag{6.40}$$

The LQG regulator problem is defined as follows. Find the optimal control input $\mathbf{u}^*$ in a stochastic sense, such that the following objective function is minimized

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} J(\mathbf{u}) \tag{6.41}$$

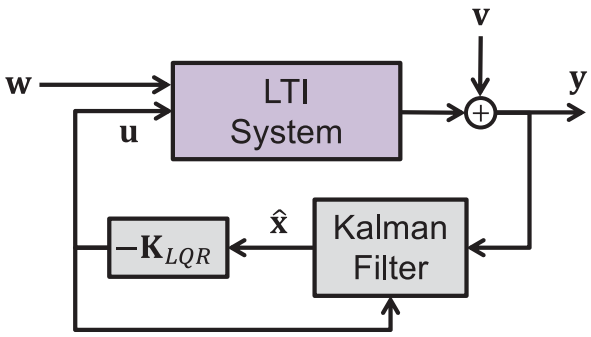


Figure 6.5: Block diagram of closed-loop system with LQG regulator

$$J(\mathbf{u}) = \lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\int_0^T (\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u})dt\right] \tag{6.42}$$

By the separation principle, the LQG controller can be designed as a combination of state-feedback LQR controller with a Kalman filter estimator. Then, the optimal control signal is defined as

$$\mathbf{u}^* = -\mathbf{K}\hat{\mathbf{x}} \tag{6.43}$$

where $\mathbf{K}$ is the LQR feedback gain calculated in (6.24), and $\hat{\mathbf{x}}$ is the state estimate obtained from the Kalman filter in (6.36). The interconnection of Kalman filter and LQR gain for feedback control purposes is illustrated in Figure 6.5 as a block diagram.

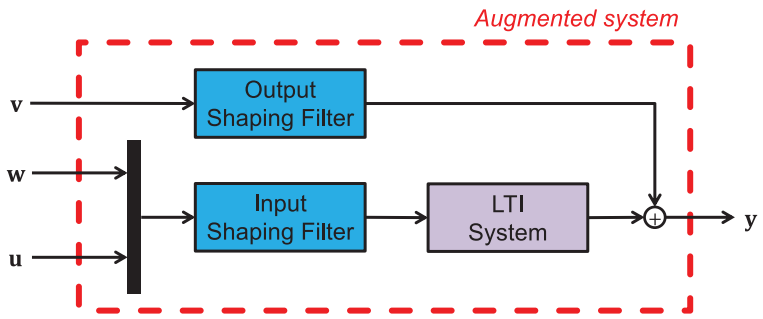


Figure 6.6: Augmented system including input and output shaping filters

6.4.4 Augmented system

If the disturbances and measurement noises are expected to have a particular frequency spectrum, then both signals can be modeled as a colored noise, i.e., a white noise signal passed through a shaping filter with matching power spectral density. Then, these shaping filters can be incorporated into the LTI system to form an augmented system for purposes of feedback control design. The shaping filters are especially important to attenuate the responses of the feedback controller at high frequencies.

In particular, a shaping filter can be designed as a low-pass filter for a frequency range $f \in [0, f_c]$. Therefore, the white-noise disturbance and noise, $\mathbf{w}$ and $\mathbf{v}$, are passed through their respective shaping filters such that the inputs to the system are attenuated for frequencies above the cutoff frequency f_c.

Consequently, the states of the LTI system, along with the states from input and output shaping filters, are incorporated into the augmented system as follows:

$$\dot{\mathbf{x}}_a = \mathbf{A}_a \mathbf{x}_a + \mathbf{B}_a \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \\ \mathbf{v} \end{bmatrix} \tag{6.44a}$$

$$\mathbf{y} = \mathbf{C}_a \mathbf{x}_a + \mathbf{D}_a \begin{bmatrix} \mathbf{u} \\ \mathbf{w} \\ \mathbf{v} \end{bmatrix} \tag{6.44b}$$

with

$$\mathbf{x}_a = \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_{\text{in}} \\ \mathbf{x}_{\text{out}} \end{bmatrix} \tag{6.45}$$

where $\mathbf{x}$ are the states of the LTI system; and $\mathbf{x}_{\text{in}}$ and $\mathbf{x}_{\text{out}}$ are the states of the input and output shaping filters, respectively. An illustration of the augmented system with its components is provided in Figure 6.6.

6.4.5 Loop transfer recovery

If any frequency-domain specifications are meant to be satisfied during feedback control design (e.g., reference tracking, noise and disturbance rejection), a procedure called *loop transfer recovery* (LTR) can be introduced (Stein and Athans, 1987). The objective of LTR is to "recover" the stability robustness properties from state-feedback LQR design, by modifying the Kalman filter design.

Consider the LQG problem formulated in Section 6.4.3, and assume that the system is minimum phase (i.e., all transmission zeros of the MIMO system have strictly negative real parts). Then, the transfer function of the LQG compensator follows the relationship

$$\mathbf{K}_{\mathrm{LQG}}(s) = -\mathbf{K}(s\mathbf{I} - \mathbf{A} + \mathbf{B}\mathbf{K} + \mathbf{L}\mathbf{C})^{-1}\mathbf{L} \tag{6.46}$$

Now, instead of treating the noise covariance matrices $\mathbf{Q}_w$ and $\mathbf{R}_v$ as fixed parameters, these can be "tuned" such that the design specifications are met. Then, the design parameters $\mathbf{Q}_w$ and $\mathbf{R}_v$ are redefined as follows:

$$\mathbf{Q}_w = \mathbf{\Gamma}^T\mathbf{\Gamma}, \quad \mathbf{R}_v = \mathbf{I} \tag{6.47}$$

$$\mathbf{\Gamma} = q\mathbf{B}_w \tag{6.48}$$

where q is a scalar design parameter. Then, in the limit as $q \to \infty$, the open-loop transfer function recovers the original properties of the state-feedback LQR controller:

$$\lim_{q\to\infty} \mathbf{K}_{\mathrm{LQG}}(s)\mathbf{G}(s) = \mathbf{K}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \tag{6.49}$$

Hence, frequency-domain loop-shaping of the multi-input, multi-output (MIMO) system can be performed by adjusting the parameter q. More details on the LTR design procedure can be found in Franklin et al. (2015). Finally, is still possible to conduct the LTR procedure in practice for non-minimum phase systems, but the performance of the feedback system is greatly limited.

6.4.6 Design considerations

The feedback regulator considered for model-based compensation in maRTHS testing was designed with using the Control Design Toolbox in *Matlab*, with the following considerations:

- *LTI system:* The state-space model developed in Section 5.6.6 was considered for optimal control purposes. The state-space matrices are presented in Appendix B.2.
- *Input shaping filter:* A Kanai-Tajimi filter is considered to attenuate the high frequency energy contained in the control signal. The Kanai-Tajimi filter was designed with characteristic frequency of $f_f = 10$ Hz (natural frequency $\omega_f = 2\pi f_f$), and a

characteristic damping of $\zeta_f = 0.70$.

$$\dot{\mathbf{x}}_{\text{in}} = \begin{bmatrix} 0 & 1 \\ -\omega_f^2 & -2\zeta_f\omega_f \end{bmatrix} \mathbf{x}_{\text{in}} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \mathbf{u} \tag{6.50a}$$

$$\mathbf{y}_{\text{in}} = \begin{bmatrix} \omega_f^2 & 2\zeta_f\omega_f \end{bmatrix} \mathbf{x}_{\text{in}} \tag{6.50b}$$

- *Output shaping filter:* A third-order Butterworth filter with a cutoff frequency $f_c = 15$ Hz is considered to reduce high-frequency noise in output response channels.

- *LQR controller:* The LQR controller was design using the `lqry()` function in *Matlab*, with the following parameters:

$$\bar{\mathbf{Q}} = \text{diag}\left(\frac{1}{u_o^2}, \frac{1}{u_o^2}, \frac{1}{u_o^2}, \frac{1}{\theta_o^2}, \frac{1}{\theta_o^2}, \frac{1}{\theta_o^2}\right), \quad \bar{\mathbf{R}} = \rho\bar{\mathbf{Q}}, \quad \rho = 1\text{e}-4$$

$$u_o = 2.58 \text{ [mm]}, \quad \sigma_\theta = 4.47 \text{ [deg]} \qquad \text{(maximum acceptable values)}$$

- *Kalman filter:* The Kalman filter was design using the `kalman()` function in *Matlab*, with the following parameters:

$$\mathbf{Q}_w = \text{diag}\left(\sigma_u^2, \sigma_u^2, \sigma_u^2, \sigma_\theta^2, \sigma_\theta^2, \sigma_\theta^2\right), \quad \mathbf{R}_v = q\mathbf{Q}_w, \quad q = 10$$

$$\sigma_u^2 = 7 \times 10^{-5} \text{ [mm}^2\text{]}, \quad \sigma_\theta^2 = 2 \times 10^{-6} \text{ [deg}^2\text{]} \qquad \text{(variance)}$$

Finally, the state-space realization in modal canonical form of the LQG controller is presented in Appendix C.2.

6.5 Performance analysis

In order to complete the design of the feedforward-feedback controller, an augmented system is specified to obtain measures of performance and robustness of the closed-loop system. The augmented system has three inputs (reference $\mathbf{r}$, disturbance $\mathbf{d}$, and noise $\mathbf{n}$), and two outputs ($\mathbf{y}$ and error $\mathbf{e}$), as illustrated in Figure 6.7.

Then, the input-output relationships of the augmented system are given by the following expressions:

$$\begin{Bmatrix} \mathbf{y} \\ \mathbf{e} \end{Bmatrix} = \begin{bmatrix} \mathbf{T}_0 + \mathbf{S}_0\mathbf{T}_{\text{ff}} & \mathbf{S}_0\mathbf{G} & -\mathbf{T}_0 \\ \mathbf{S}_0\mathbf{S}_{\text{ff}} & -\mathbf{S}_0\mathbf{G} & -\mathbf{S}_0 \end{bmatrix} \begin{Bmatrix} \mathbf{r} \\ \mathbf{d} \\ \mathbf{n} \end{Bmatrix} \tag{6.51}$$

where $\mathbf{S}_{\text{ff}}(s)$ and $\mathbf{T}_{\text{ff}}(s)$ are the feedforward sensitivity and complementary sensitivity functions, respectively:

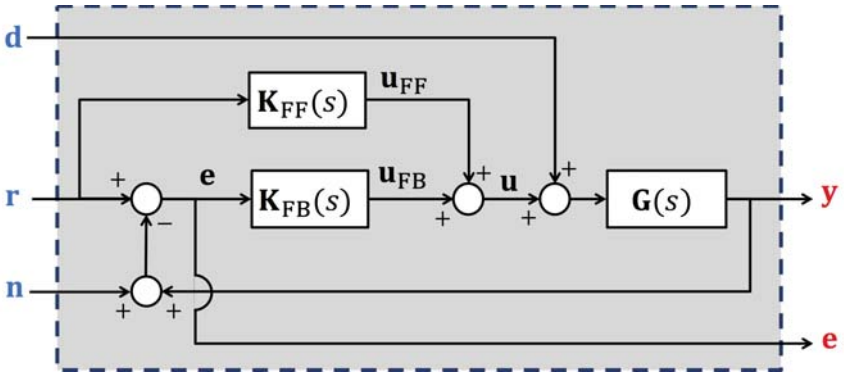


Figure 6.7: Augmented system for loop shaping purposes of model-based compensator for maRTHS

$$\mathbf{S}_{\text{ff}}(s) = \mathbf{I} + \mathbf{G}(s)\mathbf{K}_{\text{ff}}(s) \tag{6.52}$$
$$\mathbf{T}_{\text{ff}}(s) = \mathbf{I} - \mathbf{S}_{\text{ff}}(s) = \mathbf{G}(s)\mathbf{K}_{\text{ff}}(s) \tag{6.53}$$

while $\mathbf{S}_0(s)$ and $\mathbf{T}_0(s)$ are the output sensitivity and complementary sensitivity functions, respectively:

$$\mathbf{S}_0(s) = [\mathbf{I} + \mathbf{G}(s)\mathbf{K}_{\text{fb}}(s)]^{-1} \tag{6.54}$$
$$\mathbf{T}_0(s) = \mathbf{I} - \mathbf{S}_0(s) \tag{6.55}$$

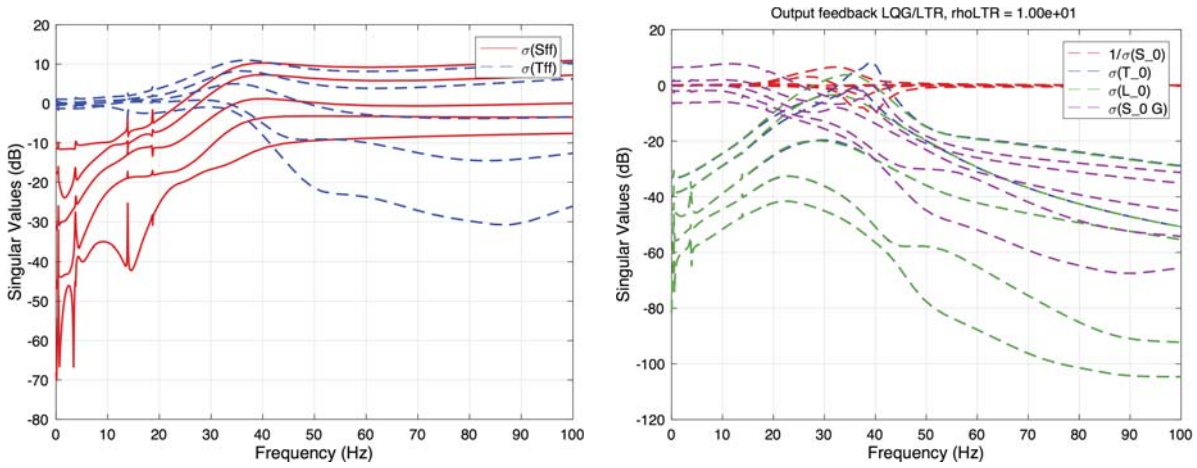
From this relationship, the frequency-domain specifications can be selected such that multiple performance objectives, in terms of matrix norms of the sensitivity transfer functions, are satisfied simultaneously:

$$\|\mathbf{T}_0 + \mathbf{S}_0\mathbf{T}_{\text{ff}}\| \to 1, \quad \omega < \omega_c \quad \text{(reference tracking)} \tag{6.56}$$
$$\|\mathbf{S}_0\mathbf{S}_{\text{ff}}\| \to 0, \quad \omega < \omega_c \quad \text{(reference tracking)} \tag{6.57}$$
$$\|\mathbf{S}_0\mathbf{G}\| \to 0, \quad \omega < \omega_c \quad \text{(disturbance rejection)} \tag{6.58}$$
$$\|\mathbf{T}_0\| \to 0, \quad \omega > \omega_c \quad \text{(noise rejection)} \tag{6.59}$$
$$\|\mathbf{S}_0\| \to 0, \quad \omega > \omega_c \quad \text{(noise rejection)} \tag{6.60}$$

where ω_c is the crossover natural frequency, which determines a tradeoff between performance and robustness bounds for the closed-loop system. Also, $\|\cdot\|$ is the matrix norm of a linear and stable MIMO system. For example, the Euclidean matrix norm can be considered for this purpose:

$$\|\mathbf{A}(j\omega)\|_2 \triangleq \bar{\sigma}(\mathbf{A}(j\omega)), \quad \forall\omega \tag{6.61}$$

where $\bar{\sigma}(\cdot)$ is the largest singular value of a matrix.



(a) Feedforward sensitivity functions (b) Feedback sensitivity functions

Figure 6.8: Loop shaping results for feedforward/LQG controller

From the set of frequency-domain design specifications, the dependency between feedback and feedforward design includes additional constraints to the compensator design. Since a closed-form solution to this optimization problem is not possible, primarily because of an intrinsic trade-off between performance and robustness (sometimes called waterbed effect), an experienced control designer should be able to find a solution that could satisfy the performance objectives for specific applications.

Consequently, the results of loop shaping design of the feedforward and feedback controllers in terms of maximum singular values of the sensitivity and complementary sensitivity functions are shown in Figure 6.8a and 6.8b, respectively. First, the feedforward controller satisfy the reference tracking objective for a frequency bandwidth up to 15 Hz. Beyond that frequency, the tracking results are deteriorated due to model uncertainty. On the other hand, good disturbance rejection is obtained with the feedback controller for frequencies up to 25 Hz, and reasonable noise rejection is obtained for high-frequency content beyond this crossover frequency. But, the LQG regulator does not offer any reference tracking guarantees, and the feedforward is the only instrument responsible for reference tracking in the context of maRTHS testing.

6.6 Summary

In this chapter, a model-based compensator was proposed for synchronization of target and measured signals in multi-axial real-time hybrid simulation (maRTHS). The model-based compensator is designed such that it offers improved performance and robustness when controlling a multi-input multi-output (MIMO) system. For that matter, a feedforward compensator with an LQG/LTR feedback regulator were designed separately to satisfy frequency-domain specifications.

The feedforward compensator was developed using an inverse-based model approach. Then, the transfer function model was a real polynomial matrix. Also, since the transfer function was improper, a finite difference discretization was employed to estimate higher-order derivatives of the target signals. Moreover, optimal control theory was considered to design the feedback regulator. An output feedback LQG regulator was proposed, and the LTR procedure was required in order to satisfy performance and robustness specifications in frequency domain.

Finally, the feedforward compensator is considered the main responsible for reference tracking in maRTHS testing, but with clear bandwidth limitations imposed by the accuracy of the inverse-based models for high frequencies. Moreover, the feedback regulator performs in terms of disturbance rejection in the low frequency range, and measurement noise rejection in the high frequency range. But, there is no evidence of improved reference tracking performance when the feedback regulator is implemented.

Chapter 7

FRAMEWORK VALIDATION: SMALL-SCALE EXPERIMENTAL TEST

7.1 Problem statement

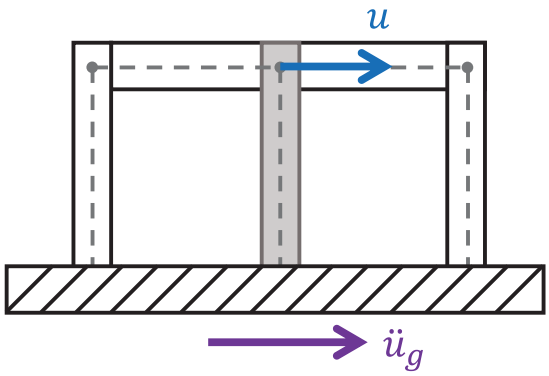
This chapter presents a series of experiments to demonstrate the effectiveness of the proposed framework. In particular, a prototype of a single-story, two bay frame structure is considered for this study, as shown in Figure 7.1a. The frame consists of three columns with fixed ends, connected through two rigid beams. Then, the center column is chosen as the experimental substructure, which exhibits a nonlinear inelastic response. Both numerical and experimental substructures are assumed to be connected through degree-of-freedom u, as shown in Figure 7.1b. The fundamental frequency of the reference prototype structure is chosen as $f_n = 2$ [Hz], with an intrinsic damping ratio of $\zeta = 5\%$, and subjected to ground motion $\ddot{u}_g$.

7.2 Description of test specimen

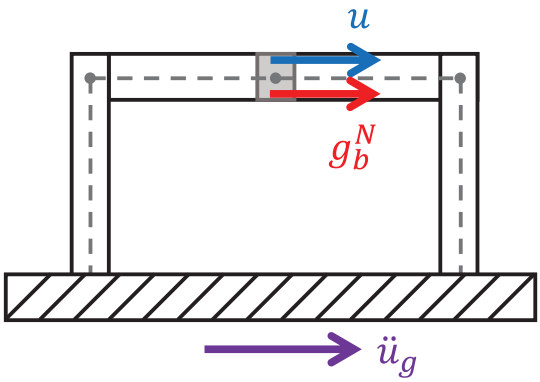
The test specimen selected for this validation study is a mild steel column with rigid connections at both ends, as shown in Figure 7.1c. The column has a uniform round bar cross-section with a diameter of 31.75 [mm] (1.25 [in]), and a total length of 457.2 [mm] (18 [in]). The steel column can undergo plastic deformations with sufficient ductility and toughness to experience multiple cycles of inelastic loading without failure. Also, the specimen is sufficiently stiff in the axial direction, imposing severe constraints over the dynamics of the multi-actuator servo-hydraulic system. Therefore, this specimen is an ideal subject to test the limits of the proposed methodology, and serve as a testbed for maRTHS development.

Before conducting the dynamic test to evaluate the performance of the proposed experimental methodology, a cyclic test was conducted to obtain the empirical stiffness matrix of the test specimen for further reference and comparison purposes. The specimen was subjected to pure translation u_x and pure rotation θ_y, in Cartesian directions x and y, respectively. Subsequently, the force f_x and moment m_y measured at the end of the test specimen are obtained.

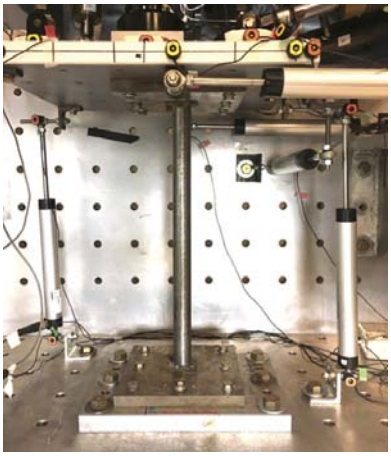
The relationship between forces and displacements is given by the following linear equation:



(a) Reference structure



(b) Numerical substructure



(c) Experimental substructure

Figure 7.1: Substructuring of maRTHS validation test

$$\mathbf{f} = \mathbf{K}_{\text{specimen}}\mathbf{u} \tag{7.1}$$

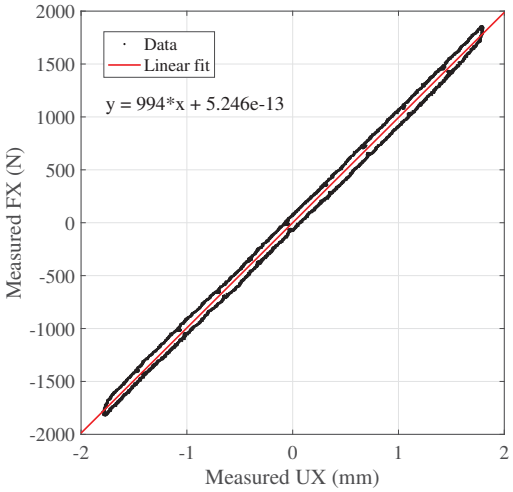
$$\begin{Bmatrix} f_x \\ m_y \end{Bmatrix} = \begin{bmatrix} K_{xx} & K_{x\theta} \\ K_{\theta x} & K_{\theta\theta} \end{bmatrix} \begin{Bmatrix} u_x \\ \theta_y \end{Bmatrix} \tag{7.2}$$

where the units are: f_x [kN], m_y [kN-mm], u_x [mm], θ_y [rad], K_{xx} [kN/mm], $K_{x\theta}$ [kN], $K_{\theta x}$ [kN], and $K_{\theta\theta}$ [kN-mm]. Figure 7.2 shows the relationship between measured forces and displacements. Even though the relationship is not perfectly linear elastic, due to friction of bolted connections, or seal friction of hydraulic actuators, the data can be approximated using a linear fit, for the purpose of obtaining an empirical stiffness matrix for reference. Then, the empirical stiffness matrix is obtained through linear regression of the experimental data:

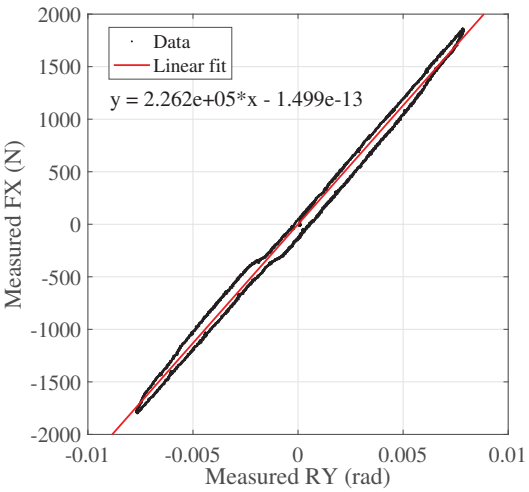
$$\mathbf{K}_{\text{specimen}} = \begin{bmatrix} 0.994 & 226.2 \\ 229.6 & 75540 \end{bmatrix} \tag{7.3}$$

Moreover, the empirical stiffness matrix is not symmetric due to either systematic and/or random errors. Obviously, this result violates Maxwell-Betti's reciprocal theorem, which is a fundamental principle of linear elastic structures. However, the differences between components $K_{x\theta}$ and $K_{\theta x}$ of the empirical stiffness matrix are considered to be small, with a normalized error of approximately 1.5%. Therefore, a way to circumvent this issue is to average components $K_{x\theta}$ and $K_{\theta x}$ to obtain a symmetric empirical stiffness matrix:

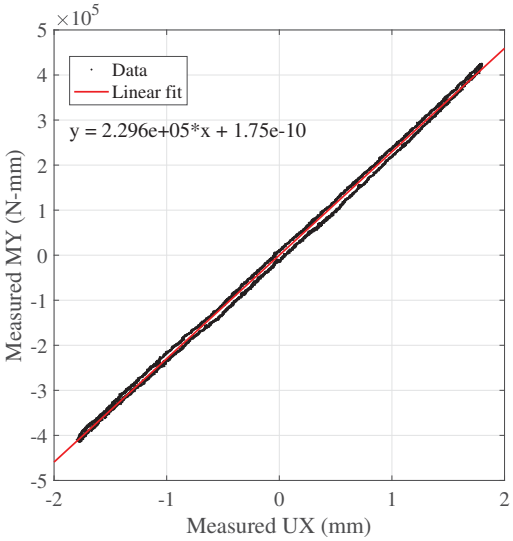
$$\mathbf{K}_{\text{specimen}}^{\text{corrected}} = \begin{bmatrix} 0.994 & 227.9 \\ 227.9 & 75540 \end{bmatrix} \tag{7.4}$$



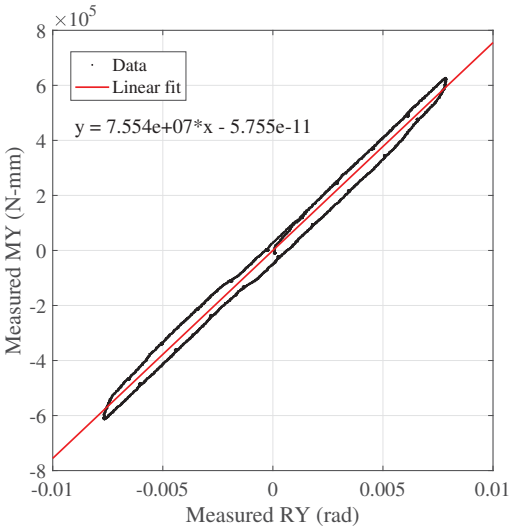
(a) Force f_x vs. translation u_x



(b) Force f_x vs. rotation θ_y



(c) Moment m_y vs. translation u_x



(d) Moment m_y vs. rotation θ_y

Figure 7.2: Elastic stiffness of steel rod column specimen

7.3 Substructuring method

The equation of motion of the reference structure is given as follows:

$$m\ddot{u} + c\dot{u} + r(u) = -m\ddot{u}_g \tag{7.5}$$

where mass m and damping c coefficients are determined using the traditional relationships from structural dynamics:

$$m = \frac{k_0}{\omega_n^2} \tag{7.6}$$

$$c = 2\zeta\omega_n m \tag{7.7}$$

while $\omega_n = 2\pi f_n$ is the natural fundamental frequency, and k_0 is the initial elastic stiffness defined by:

$$k_0 = \frac{\partial r}{\partial u}(0) \tag{7.8}$$

Subsequently, the substructuring is performed using the following definitions:

$$u = u^N \tag{7.9}$$

$$m = m^N \tag{7.10}$$

$$c = c^N \tag{7.11}$$

$$r(u) = k^N u^N + r^E(u^E) \tag{7.12}$$

$$k_0 = k^N + k^E \tag{7.13}$$

Then, the equations of motion for both the numerical and experimental substructures are given as follows:

$$m^N\ddot{u}^N + c^N\dot{u}^N + k^N u^N = -m^N\ddot{u}_g + g_b^N \qquad \text{(numerical)} \tag{7.14}$$

$$r^E(u^E) = g_b^E \qquad \text{(experimental)} \tag{7.15}$$

Henceforth, the interface compatibility and equilibrium are satisfied by:

$$u^N = u^E = u_x^{\text{target}} \qquad \text{(interface compatibility)} \tag{7.16}$$

$$g_b^N = -g_b^E = -f_x^{\text{meas}} \qquad \text{(interface equilibrium)} \tag{7.17}$$

For this study, the degree-of-freedom u_x in Cartesian coordinates is chosen as the interface lateral degree-of-freedom of the experimental substructure (i.e., $u_x = u^N = u^E$), while all the other Cartesian degrees-of-freedom are kept to zero values (i.e., $u_y = u_z =$

$\theta_x = \theta_y = \theta_z = 0$). Thus, all Cartesian degrees-of-freedom are controlled simultaneously to satisfy the prescribed boundary conditions at the interface of the hybrid system, even if one or more Cartesian cartesians have prescribed zero values.

The numerical substructure is modeled using an LTI state-space form:

$$\dot{\mathbf{x}}^N = \mathbf{A}^N \mathbf{x}^N + \mathbf{B}^N \mathbf{p}^N \tag{7.18}$$

$$\mathbf{y}^N = \mathbf{C}^N \mathbf{x}^N + \mathbf{D}^N \mathbf{p}^N \tag{7.19}$$

where $\mathbf{x}^N = \{u^N, \dot{u}^N\}^T$ is the state vector; $\mathbf{p}^N = \{\ddot{u}_g, f_x^{\text{meas}}\}^T$ is the input vector; and $\mathbf{y}^N = \{u^N, \dot{u}^N, \ddot{u}^N_{\text{abs}}\}^T$ is the output (measurement) vector. The state-space matrices are given as follows:

$$\mathbf{A}^N = \begin{bmatrix} 0 & 1 \\ -\frac{k^N}{m^N} & -\frac{c^N}{m^N} \end{bmatrix} \qquad \mathbf{B}^N = \begin{bmatrix} 0 & 0 \\ -1 & -\frac{1}{m^N} \end{bmatrix} \tag{7.20}$$

$$\mathbf{C}^N = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -\frac{k^N}{m^N} & -\frac{c^N}{m^N} \end{bmatrix} \qquad \mathbf{D}^N = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & -\frac{1}{m^N} \end{bmatrix} \tag{7.21}$$

For this study, the effective numerical column is assumed to be have a linear-elastic constitutive law with a lateral stiffness of $k_{\text{num}} = 2k_{\text{exp}}$, while the initial lateral stiffness of the test specimen is approximated to $k^E = 1$ [kN/mm]. Therefore, the initial lateral stiffness is equal to $k_0 = 3$ [kN/mm], and given a fundamental frequency of $f_n = 2$ [Hz] and damping ratio $\zeta = 5\%$, the values for mass and damping coefficients are equal to $m = m^N = 1.90\times10^{-2}$ [kN-s^2/mm] and $c = c^N = 2.30 \times 10^{-2}$ [kN-s/mm], respectively.

To solve the equations of motion, a 4th order Runge-Kutta is chosen as the numerical integration algorithm, with a fixed integration time step of $\Delta t = 1/1000$ sec. The sampling time of the controller and data acquisition system is also set to the same time step of $\Delta t = 1/1000$ sec, hence the real-time system works in single tasking mode.

7.4 Selection of ground motions

In general, the selection criteria for ground motion records in seismic performance evaluation should be associated to a specific hazard scenario for the structural system of interest. In particular, researchers and practitioners must consider the variability in earthquake ground motion, which is usually a function of site-specific conditions.

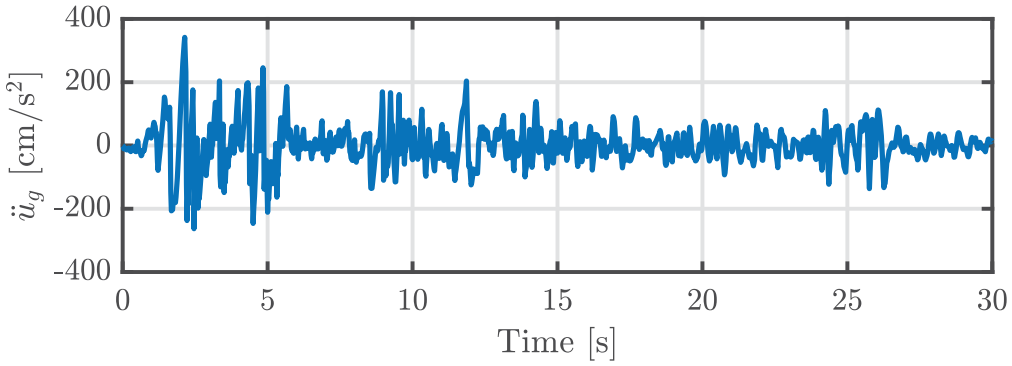
For this study, three ground motion records were chosen according to their importance and destructive characteristics, without taking into account the variability of strong ground motion and specific site conditions. The selected grounds motions are: (i) El Centro earthquake of May 18th, 1940, NS component of Imperial Valley Irrigation District substation in Imperial Valley, California; (ii) 1994 Northridge earthquake of January 17th, 1994, NS component of Sylmar County Hospital parking lot station in Sylmar, California; and (iii) Kobe earthquake of January 16th, 1995, NS component of the Japanese Meteorological Agency station in Kobe, Japan. Herein, the ground motion records are labeled ELC, NOR, and KOB, respectively. Both NOR and KOB records are classified as near-fault ground motions, which have impulsive accelerations with large damaging potential. Whereas, the ELC record exhibits characteristics of a typical far-field ground motion, even though it was recorded by a near-fault station.

The characteristics of each seismic record are provided in Table 7.1, and the first 30 seconds of each ground motion record are presented in Figure 7.3. In addition, elastic response spectra for each ground motion are calculated for reference purposes, and shown in Figure 7.4.

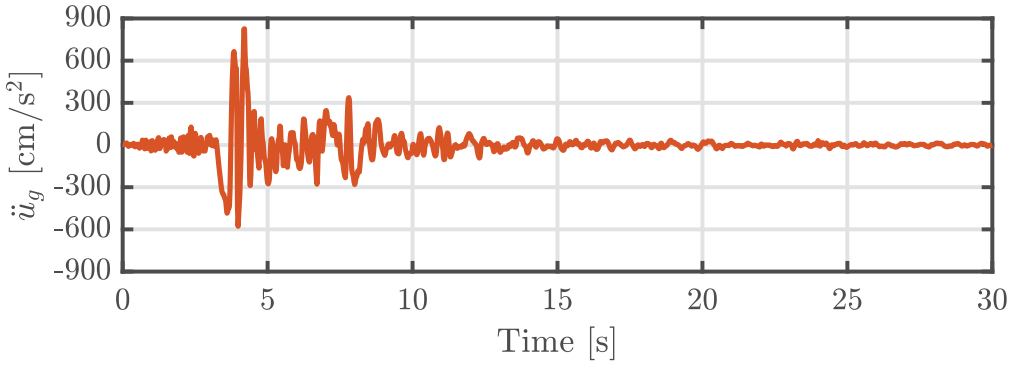
Table 7.1: Ground motion records selected for validation tests

Label	Earthquake	Year	M_w	Station name	Component	D [km]	PGA [cm/s^2]	PGV [cm/s]	PGD [cm]
ELC	El Centro	1940	6.9	Imperial Valley	NS	12.2	341.69	33.45	10.87
NOR	Northridge	1994	6.7	Sylmar	NS	15.8	826.80	128.90	32.55
KOB	Kobe	1995	6.9	KJMA	NS	1.0	805.45	81.3	17.7

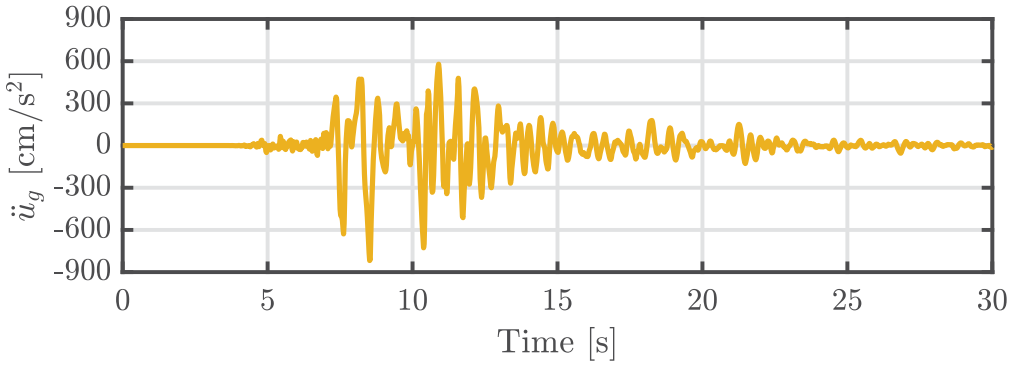
M_w: moment magnitude; D: fault distance; PGA: peak ground acceleration; PGV: peak ground velocity; PGD: peak ground displacement



(a) 1940 El Centro earthquake, Imperial Valley station, NS component

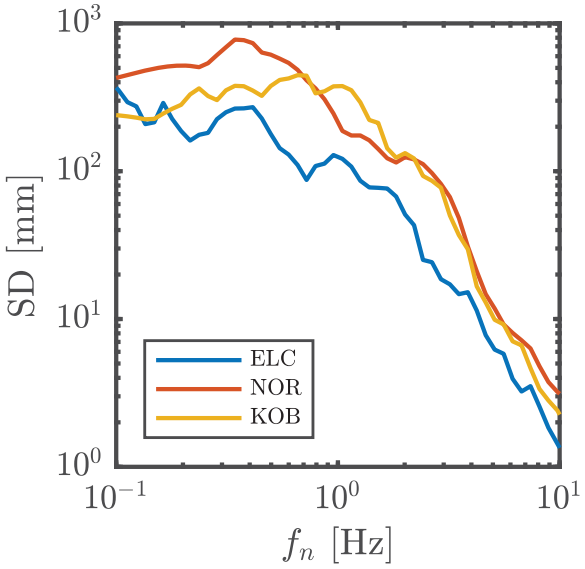


(b) 1994 Northridge earthquake, Sylmar station, NS component

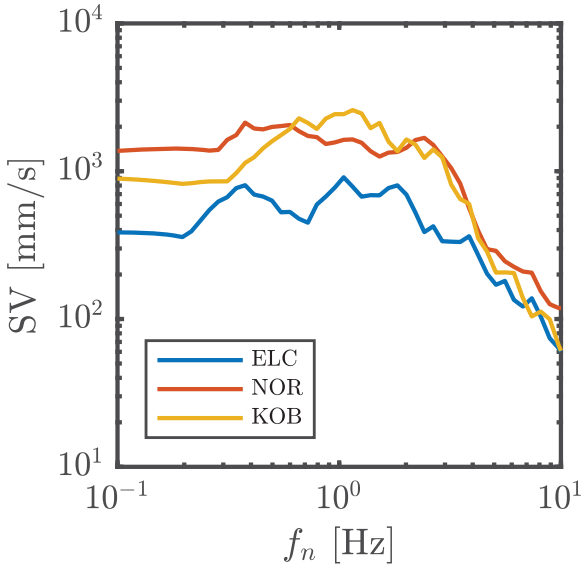


(c) 1995 Kobe earthquake, KJMA station, NS component

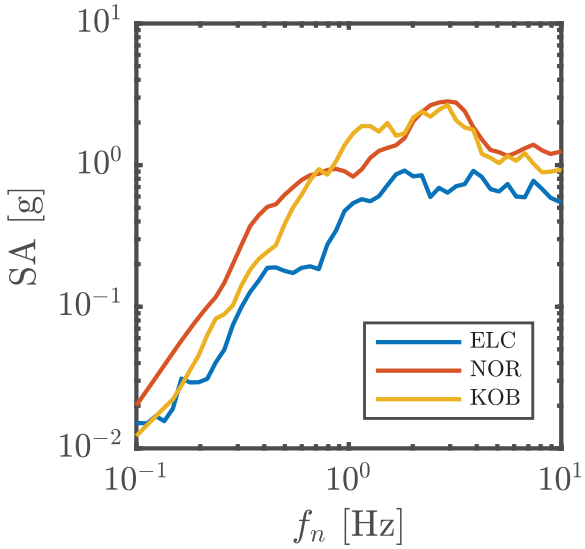
Figure 7.3: Selected ground motion records



(a) Spectral displacement



(b) Spectral velocity



(c) Spectral acceleration

Figure 7.4: Elastic response spectra for 5% damping ratio

7.5 Evaluation of model-based compensator for linear response

The model-based compensation method is evaluated through a series of experimental tests. For this purpose, only linear elastic response of the test specimen is examined. Therefore, the scaling of ground motion records is set to small values to avoid yielding of the specimen. Specifically, the ELC ground motion is chosen for this study, with scaling factors of 3% and 10% of peak ground acceleration (PGA).

The tracking performance in the time-domain is evaluated for three different control scenarios: (i) no compensation (i.e. model-based compensation was disabled from the system); (ii) only feedforward (FF) compensation; and (iii) feedforward-feedback (FF+FB) compensation. First, the tracking performance is assessed by graphic inspection of the synchronization subspace plot (SSP), where the measured displacement is plotted against the target displacement. Perfect tracking would be the case where all points lay in a perfectly straight diagonal line with 1:1 slope, i.e. measured and target signals are identical.

The SSP results for 3% and 10% scaled ELC ground motion are shown in Figures 7.5 and Figure 7.6, respectively. Thus, the model-based compensation can effectively reduce experimental errors due to multi-actuator coupling dynamics, because of a significant reduction of the counter-clockwise elliptical loops associated to phase lag in the measured Cartesian response. Furthermore, the model-based compensation is able to correct to some extent the undershoot errors at the peak responses, a phenomenon that is associated to large specimen force reactions, as previously discussed in Section 5.2.3. Still the undershoot error is not perfectly compensated, as illustrated at the peak values from Figures 7.6b and 7.6c. Better undershoot compensation may be obtained for another choice of DC gains for the feedforward compensator, which was obtained through system identification of the inverse model as explained in Section 6.3.2.

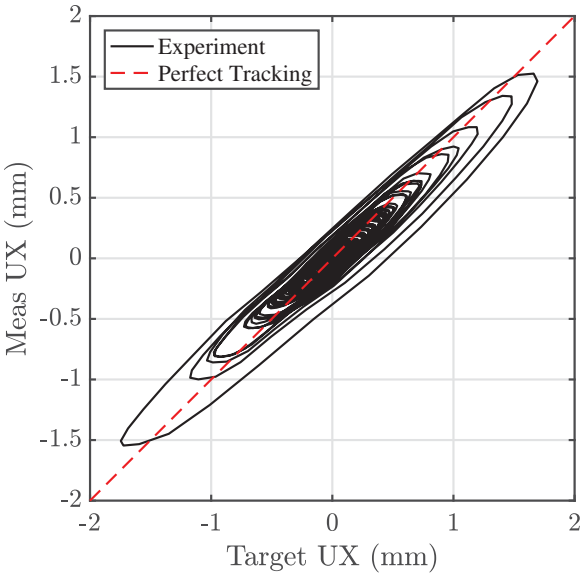
Subsequently, two quantitative error measures are considered to compare the different compensation strategies: (i) normalized peak absolute error (NPAE); and (ii) normalized root-mean-square error (NRMSE). Both error measures are defined as follows:

$$\text{NPAE} = \frac{\max\left|u_x^{\text{meas}}[k] - u_x^{\text{target}}[k]\right|}{\max\left|u_x^{\text{target}}[k]\right|} \tag{7.22}$$

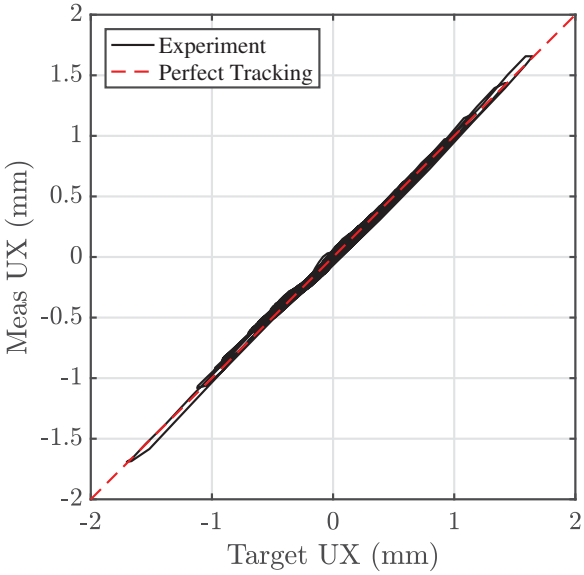
$$\text{NRMSE} = \frac{\sqrt{\frac{1}{n}\sum_{k=1}^{n}\left(u_x^{\text{meas}}[k] - u_x^{\text{target}}[k]\right)^2}}{\left|\max(u_x^{\text{target}}[k]) - \min(u_x^{\text{target}}[k])\right|} \tag{7.23}$$

where $u_x^{\text{meas}}[k]$ and $u_x^{\text{target}}[k]$ are the measured and target Cartesian displacement in x direction at time step "k", respectively.

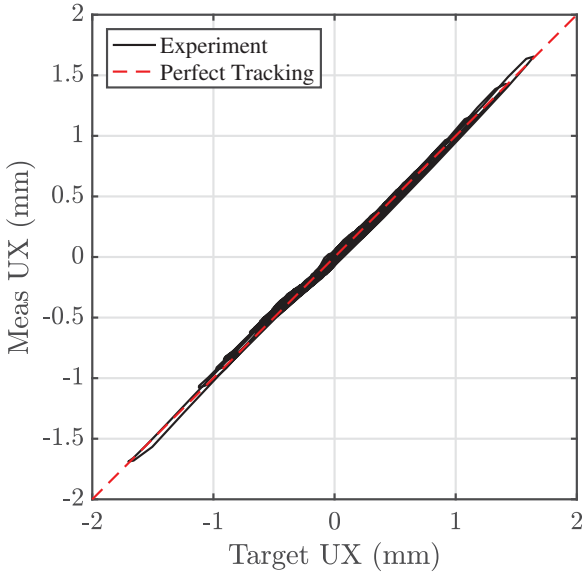
The error indices are presented in Tables 7.2 and 7.3 for tests subjected to the ELC ground motion with 3% and 10% scaling, respectively. Both compensation methods (FF and FF+FB) yield better tracking performance compared to the case without compensation, both in terms of normalized peak absolute error (NPAE) and normalized root-mean-square error (NRMSE). However, the improvement of performance given by the addition of feedback control is not evident compared to feedforward control. Indeed, model-based compensation



(a) No compensation

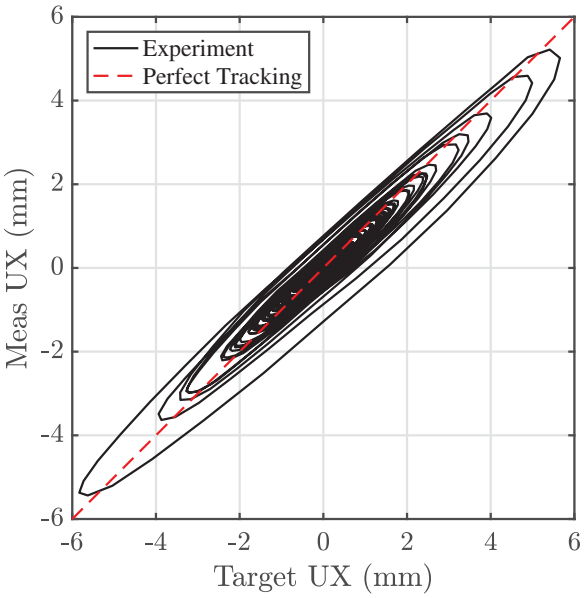


(b) Feedforward compensation

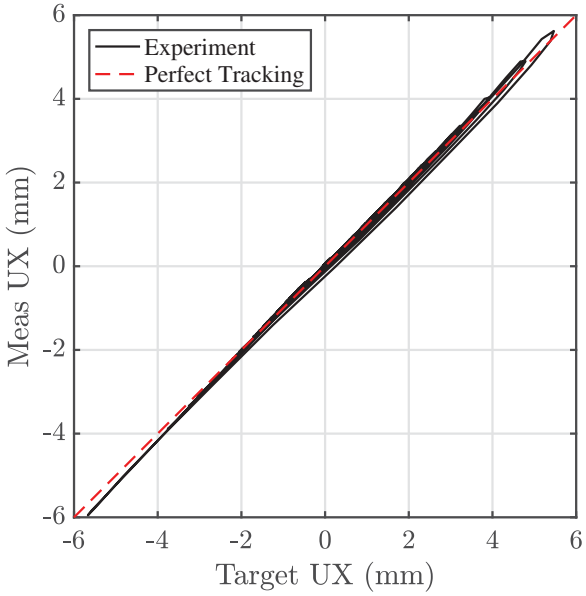


(c) Feedforward-feedback compensation

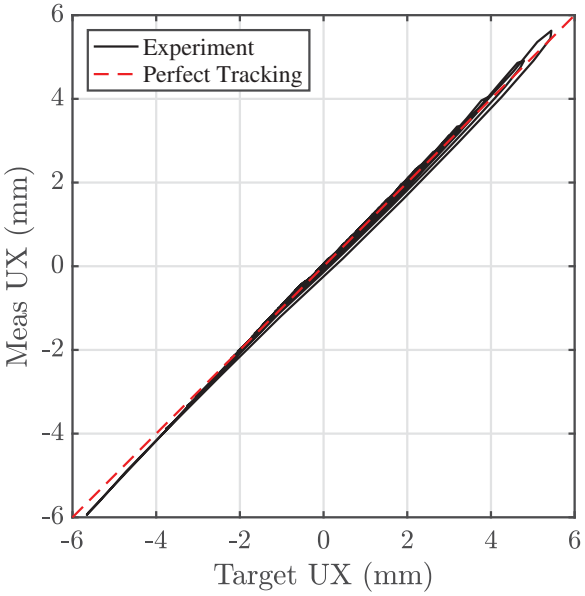
Figure 7.5: Synchronization subspace plots (SSPs) for different compensation scenarios, 3% scaled ELC ground motion



(a) No compensation



(b) Feedforward compensation



(c) Feedforward-feedback compensation

Figure 7.6: Synchronization subspace plots (SSPs) for different compensation scenarios, 10% scaled ELC ground motion

successfully reduces the tracking errors during the experimental test; nevertheless, more studies on the model-based compensation strategies should be conducted in the future.

Table 7.2: Error indices for u_x reference tracking, 3% scaled ELC ground motion

Control strategy	NPAE [%]	NRMSE [%]
No Compensation	27.13	2.49
Feedforward	5.25	1.00
Feedforward-Feedback	5.23	1.01

Table 7.3: Error indices for u_x reference tracking, 10% scaled ELC ground motion

Control strategy	NPAE [%]	NRMSE [%]
No Compensation	27.18	1.94
Feedforward	5.28	0.41
Feedforward-Feedback	5.31	0.41

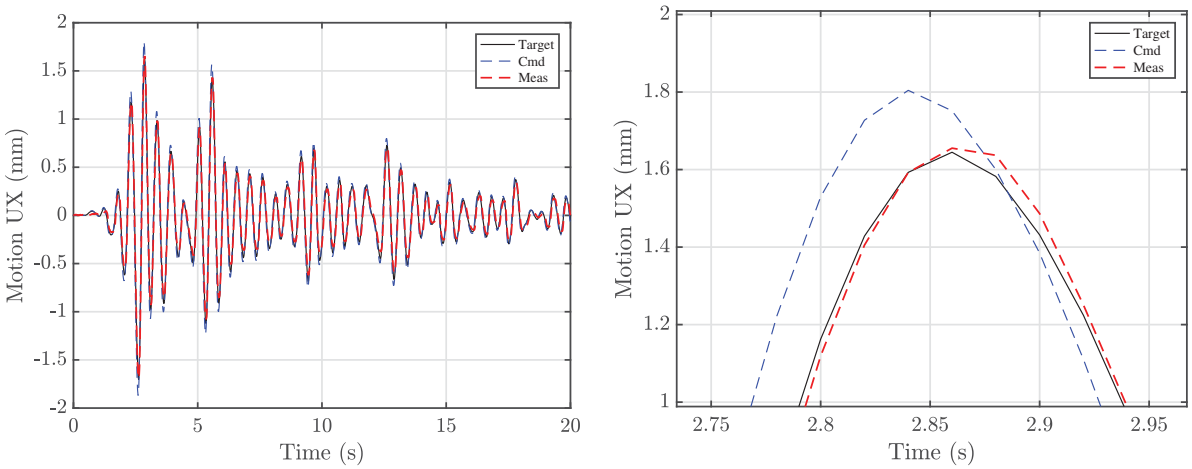
In addition, the time-domain experimental results for 3% and 10% scaled ELC ground motion are presented in Figures 7.7 and 7.9, respectively. These figures show a comparison between the target, command, and measured signals of the u_x coordinate at the interface with the test specimen, when the feedforward-feedback model-based compensator is accounted. Two close-ups of a single peak response is shown, where the measured signal shows good synchronization with the target signal. Also, the command signal is always leading the target signal, which is an expected outcome of the feedforward compensator.

Likewise, and examination of the measured Cartesian coordinates u_y, u_z, θ_x, θ_y, and θ_z, that are kept to zero values during the test, provides evidence on the cross-talk effects that were not effectively compensated. For example, Figure 7.8 shows the results for 3% scaled ELC ground motion, where the lateral and vertical translation coordinates, u_y and u_z, shows cross-talk response of less than 0.04 mm and 0.1 mm, respectively; similarly, all cross-talk responses from rotation DOFs are bounded to less than 0.04 deg. These results are considered to be small and within the tolerance for the Cartesian measurement system as explained in Section 4.4.

Likewise, Figure 7.10 shows the results for 10% scaled ELC ground motion. Thus, an increase in displacement demands on the hybrid system will slightly impact the reference tracking of the other Cartesian coordinates that are supposed to remain static. The lateral and vertical translation coordinates, u_y and u_z, shows cross-talk response of less than 0.1 mm and 0.3 mm, respectively; while, all cross-talk responses from rotation DOFs are bounded to less than 0.1 deg. This results are sufficient evidence that the maRTHS procedure can enforce the prescribed boundary conditions at the interface between substructures for three-dimensional Cartesian space problems.

Indeed, the feedback controller has the main responsibility for ensuring zeroed Cartesian coordinates. The reason being that for a zeroed target signal, the feedforward command

signal should be equal to zero. Therefore, the feedback command signal is the only responsible of reducing the tracking error.



(a) Time-history plot (b) Peak response

Figure 7.7: Structural response of maRTHS test for 3% scaled ELC ground motion (Cartesian u_x direction), feedforward-feedback compensation case

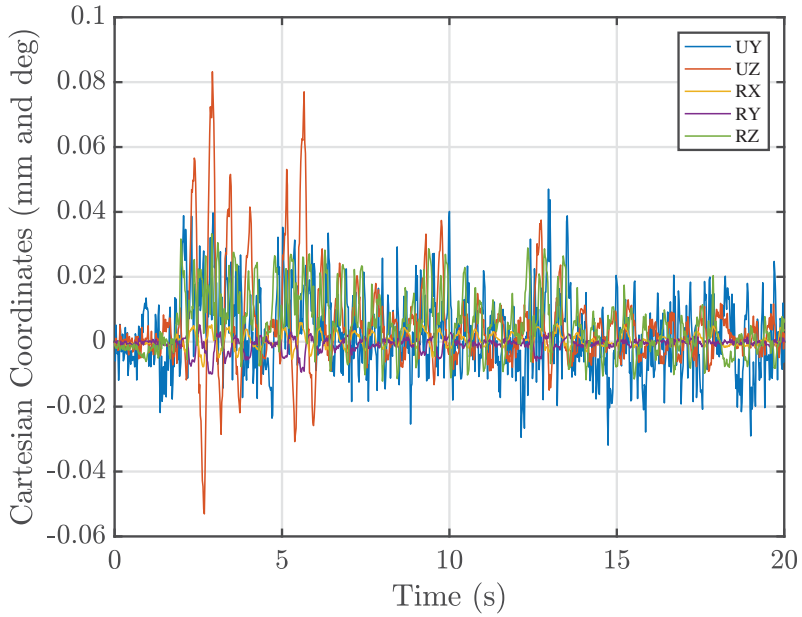
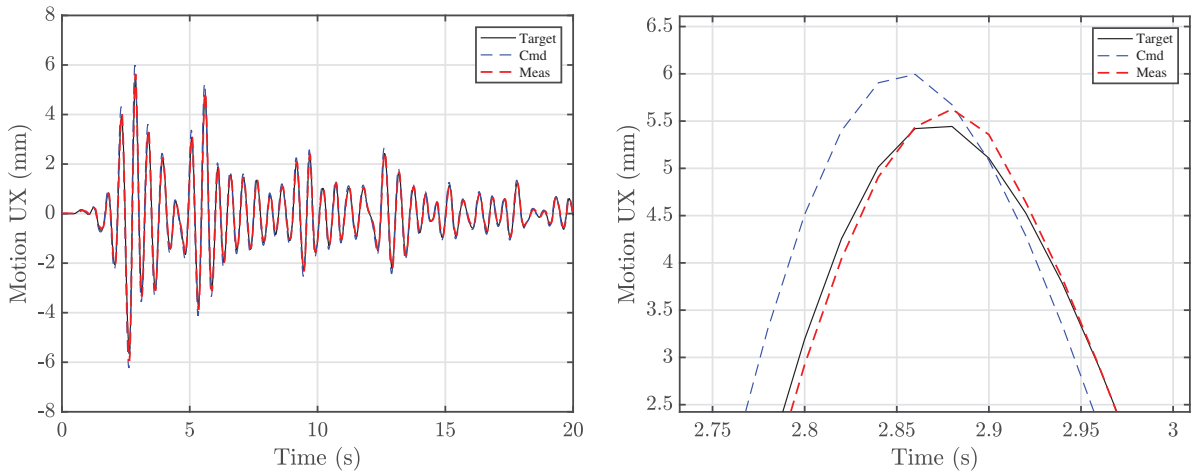


Figure 7.8: Cross-talk responses of Cartesian coordinates, 3% scaled ELC ground motion, feedforward-feedback compensation



(a) Time-history plot (b) Peak response

Figure 7.9: Structural response of maRTHS test for 10% scaled ELC ground motion (Cartesian u_x direction), feedforward-feedback compensation case

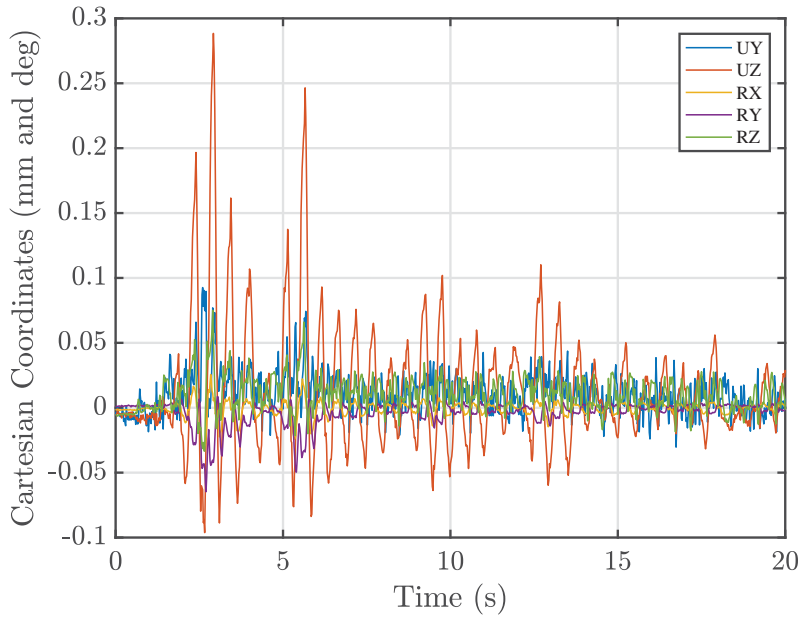


Figure 7.10: Cross-talk responses of Cartesian coordinates, 10% scaled ELC ground motion, feedforward-feedback compensation

7.6 Structural responses

7.6.1 Linear response of test specimen

As discussed previously, the performance of model-based compensation for the maRTHS test in the linear regime is within the expectations for dynamic testing. Then, an inspection of the structural responses of the hybrid system is performed. The force-displacement relationship of the test specimen when subjected to 5% and 10% scaled ELC ground motion are presented in Figures 7.11 and 7.12, respectively. These tests showed promising results for a test specimen with predominant linear response.

An almost linear elastic response is observed for measured lateral force f_x and bending moment m_y, with some dissipation that could be associated to hysteresis of the metallic specimen, friction at the joint connections, and/or seal friction inside the hydraulic actuators. Moreover, the response of vertical force f_z is attributed to large displacements.

Moreover, the results of measured lateral force f_y, bending moment m_x, and torsion moment m_z, are strictly related to tracking error of the corresponding Cartesian coordinates, and other effects such as residual forces from the loading assembly; although, these measured forces are considered to be small compared to f_x and m_y. In addition, the torsion moment m_z is another good indicator of the synchronization between actuators X1 and X2 when the system is controlled for $\theta_z = 0$. Indeed, both actuators are not perfectly equal, and will eventually oppose each others motion, which could raise an increase of torsion moment that was not intended for this study.

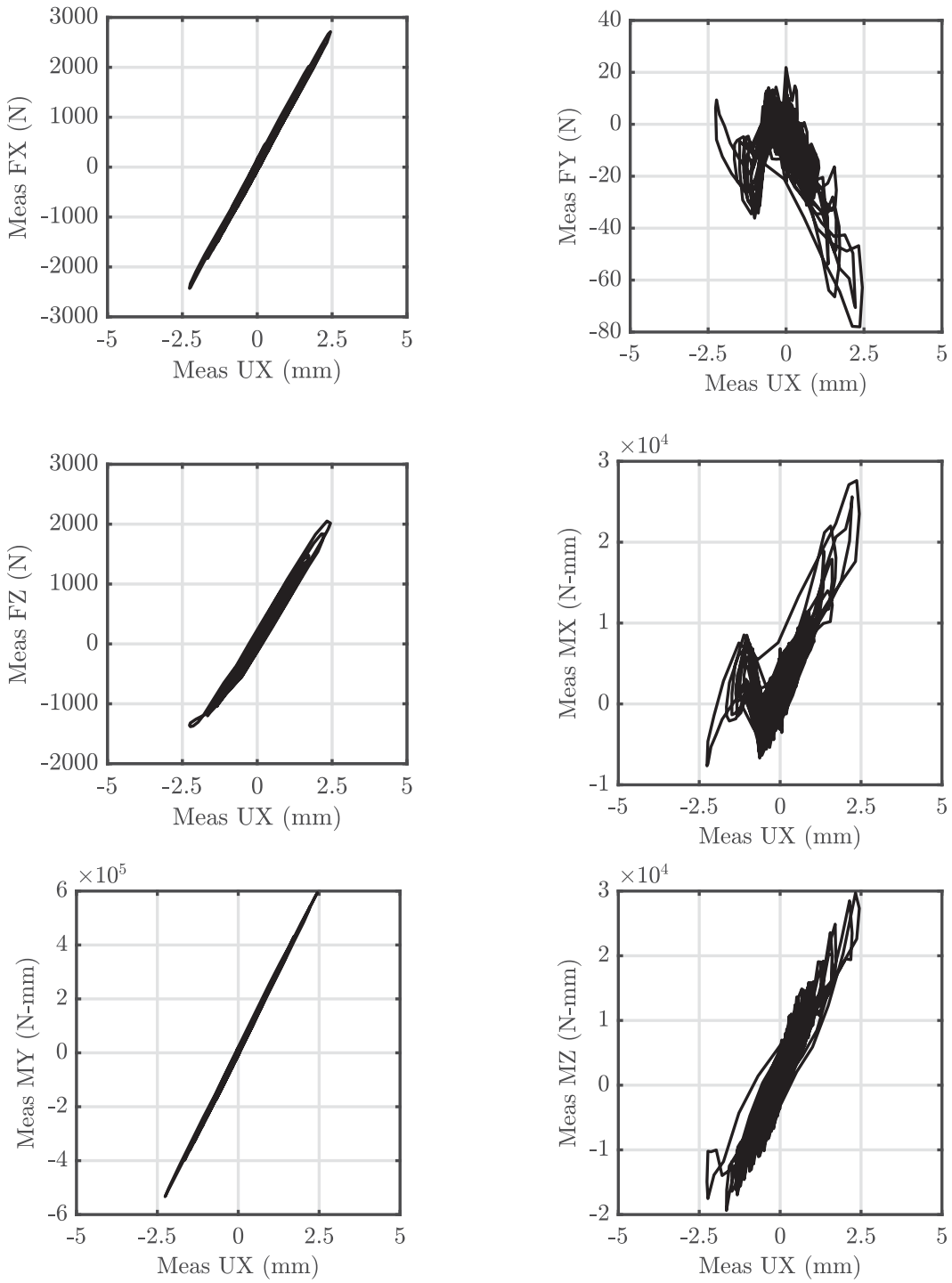


Figure 7.11: Force-displacement relations of test specimen in Cartesian coordinates, subject to 5% scaled ELC ground motion, with feedforward-feedback compensation

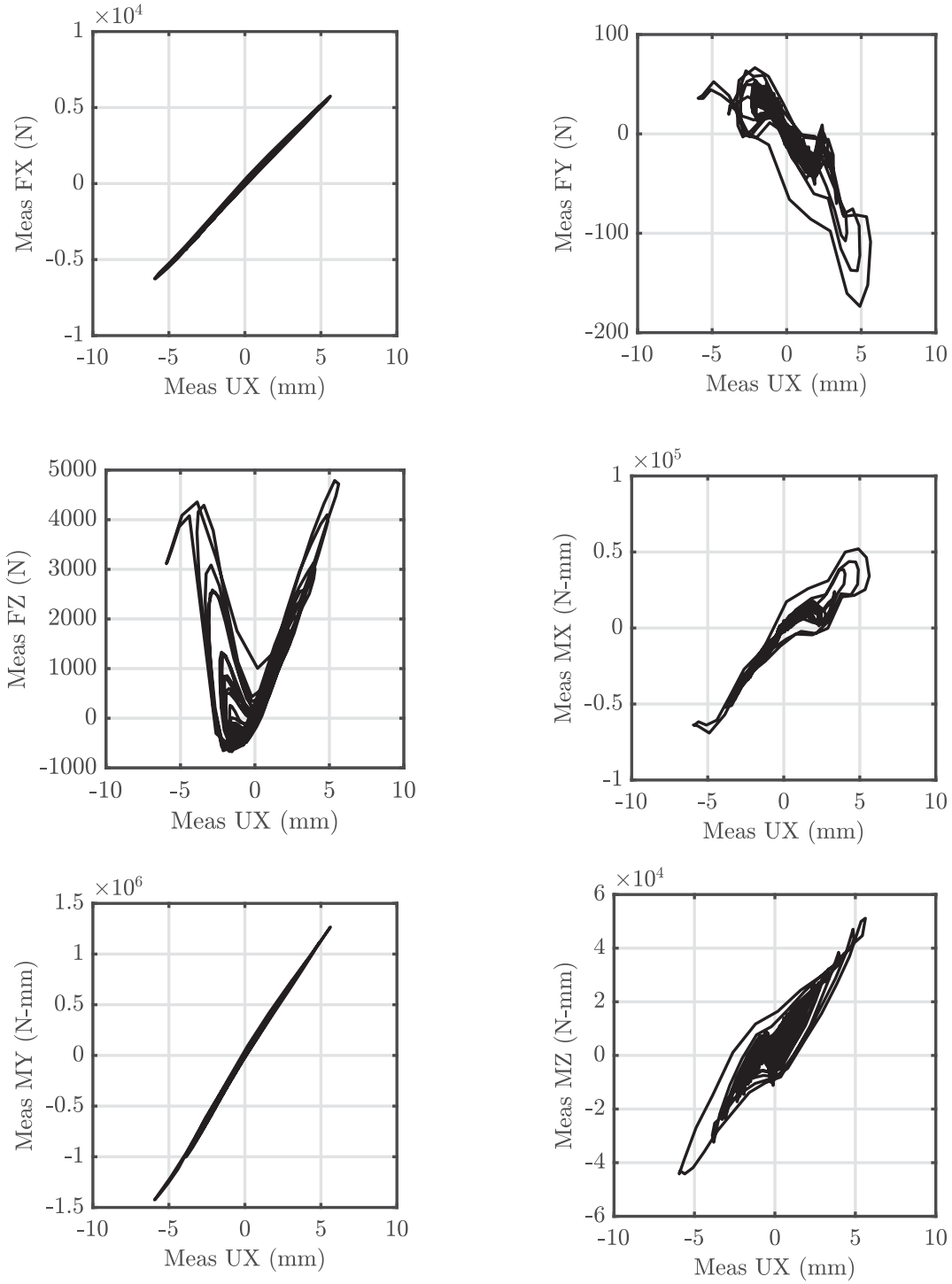


Figure 7.12: Force-displacement relations of test specimen in Cartesian coordinates, subject to 10% scaled ELC ground motion, with feedforward-feedback compensation

7.6.2 Nonlinear response of test specimen

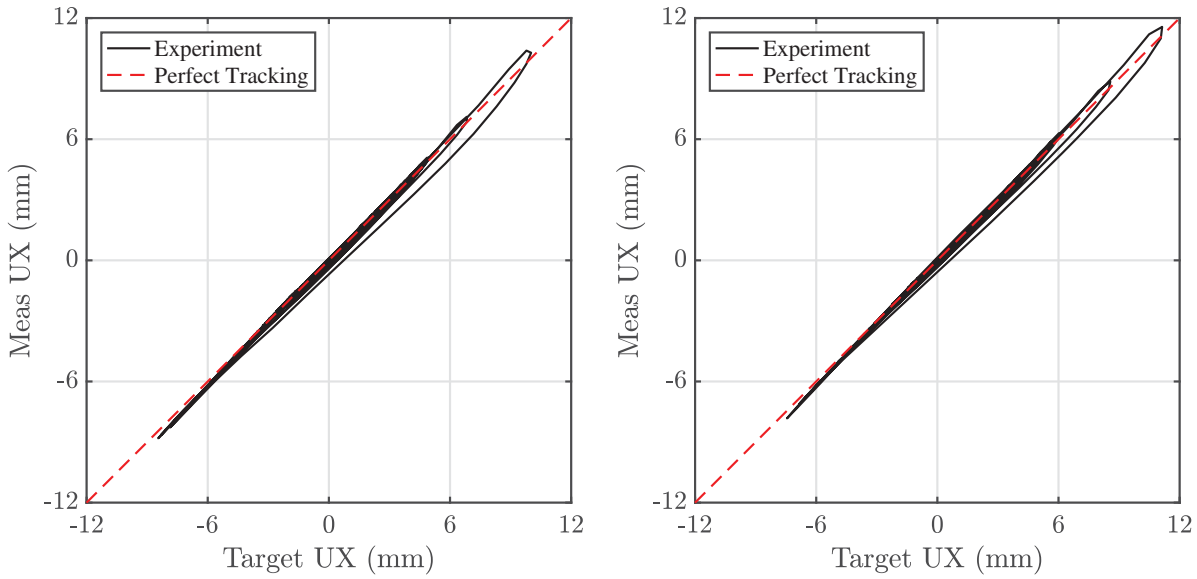
Although the tests for linear response were successful, the performance assessment of maRTHS tests with nonlinear response of the specimen is fundamental to validate the proposed framework. Indeed, the combined action of the nonlinear specimen with the loading assembly may impact the stability, performance and/or robustness of the model-based compensator.

Therefore, a series of experimental tests are conducted for larger ground acceleration, where the displacement demands of the test specimen exceeds the yield displacement. The purpose for these tests are to check the performance of the model-based compensator when the loading assembly experiences nonlinear dynamics due to plastic deformations.

First, an inspection of the reference tracking of nonlinear testing is conducted, to analyze the robustness of the model-based compensation in the presence of disturbances associated to nonlinear dynamics. Figure 7.13 presents the synchronization results of tests conducted for three ground excitations, with scaling factors chosen to yield the test specimen: (i) 20% scaled ELC ground motion; (ii) 10% scaled NOR ground motion; and (iii) 10% KOB ground motion. These tests show very good reference tracking, with some slight deterioration for large load reversals, which are typical examples of impulsive loading in near-fault ground motion records.

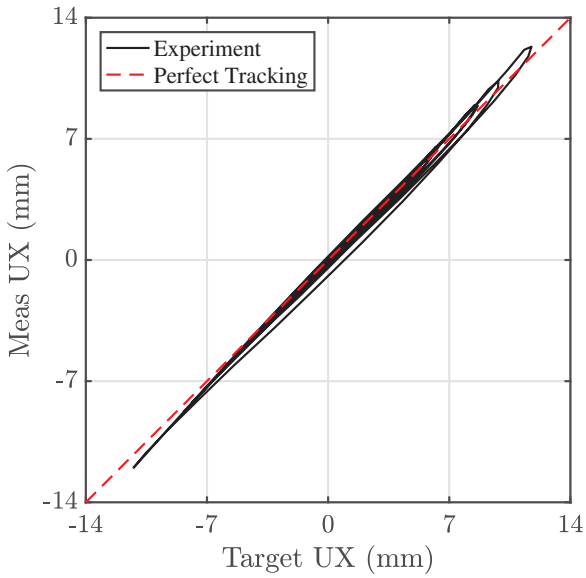
Next, the time history responses of the hybrid system are presented in Figures 7.14 through 7.19. Both measured and target signals in x Cartesian coordinate show very good agreement overall, even for pulse loading, with slight undershoot errors at the peak responses. Furthermore, the cross-talk responses are limited to relatively small values, which is translated in an adequate reference tracking for all Cartesian coordinates.

Finally, the hysteretic responses of the test specimen are provided in Figures 7.20 through 7.22. As mentioned in the previous section, the specimen exhibits a primary action of shear f_x and bending m_y, with significant vertical reaction f_z due to geometrical nonlinearities. Plastic deformations are observed, with symmetric and stable hysteretic loops that are an important instrument for energy dissipation.



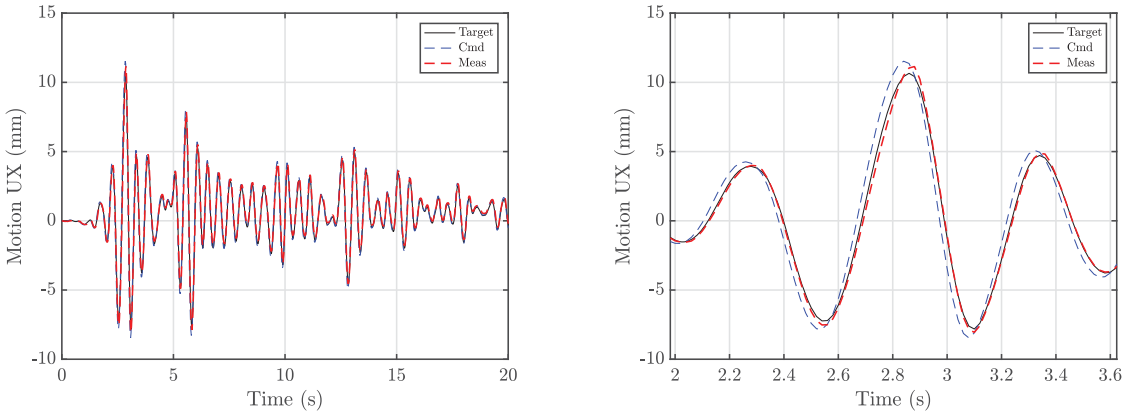
(a) 20% scaled ELC ground motion

(b) 10% scaled NOR ground motion



(c) 10% scaled KOB ground motion

Figure 7.13: Synchronization subspace plot (SSP) for nonlinear response of test specimen, feedforward-feedback compensation



(a) Time-history plot (b) Peak response

Figure 7.14: Structural response of maRTHS test for 20% scaled ELC ground motion, feedforward-feedback compensation case

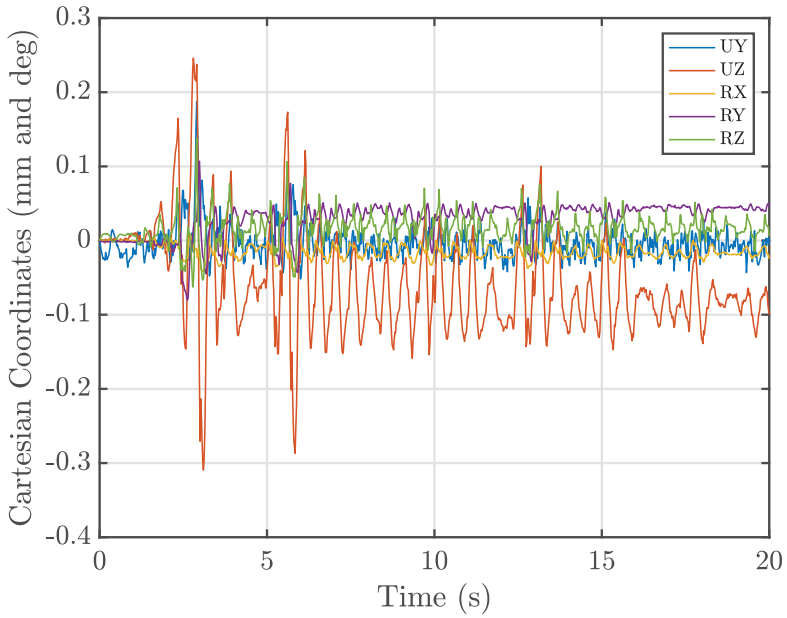
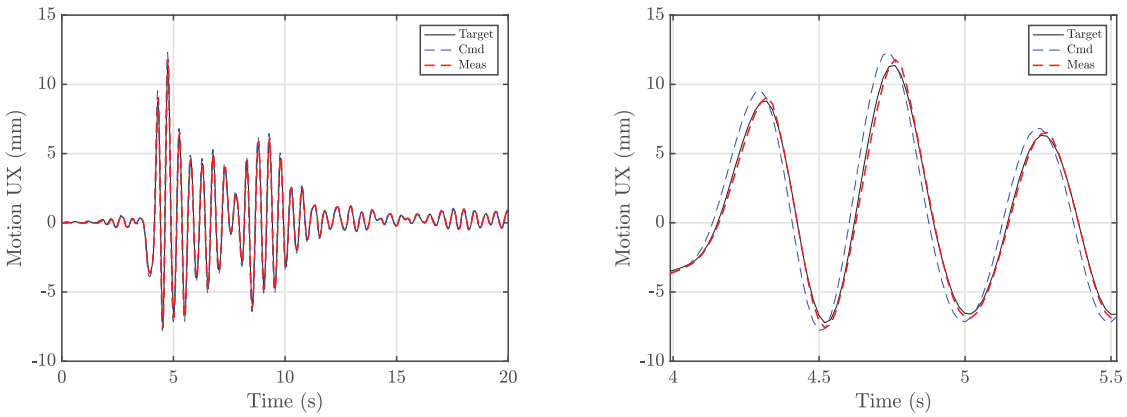


Figure 7.15: Cross-talk responses of Cartesian coordinates, 20% scaled ELC ground motion, feedforward-feedback compensation



(a) Time-history plot (b) Peak response

Figure 7.16: Structural response of maRTHS test for 10% scaled NOR ground motion, feedforward-feedback compensation case

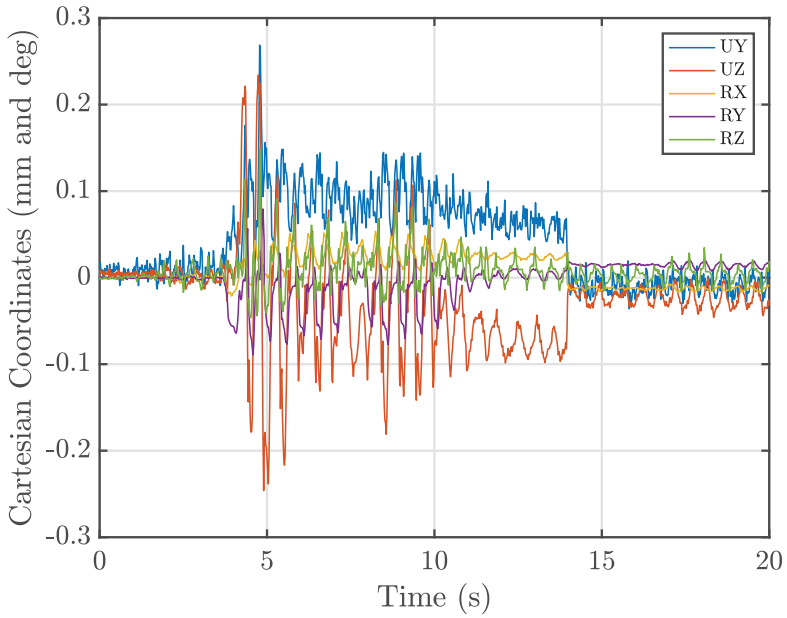
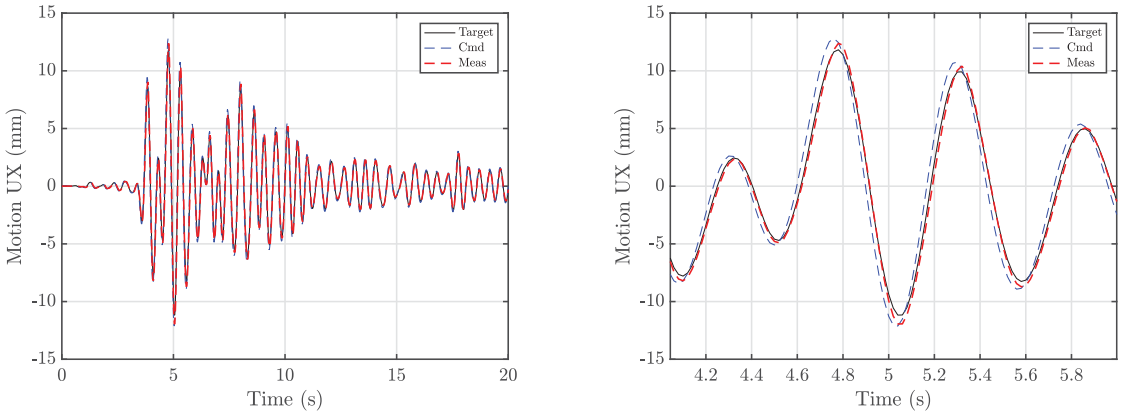


Figure 7.17: Cross-talk responses of Cartesian coordinates, 10% scaled NOR ground motion, feedforward-feedback compensation



(a) Time-history plot

(b) Peak response

Figure 7.18: Structural response of maRTHS test for 10% scaled KOB ground motion, feedforward-feedback compensation case

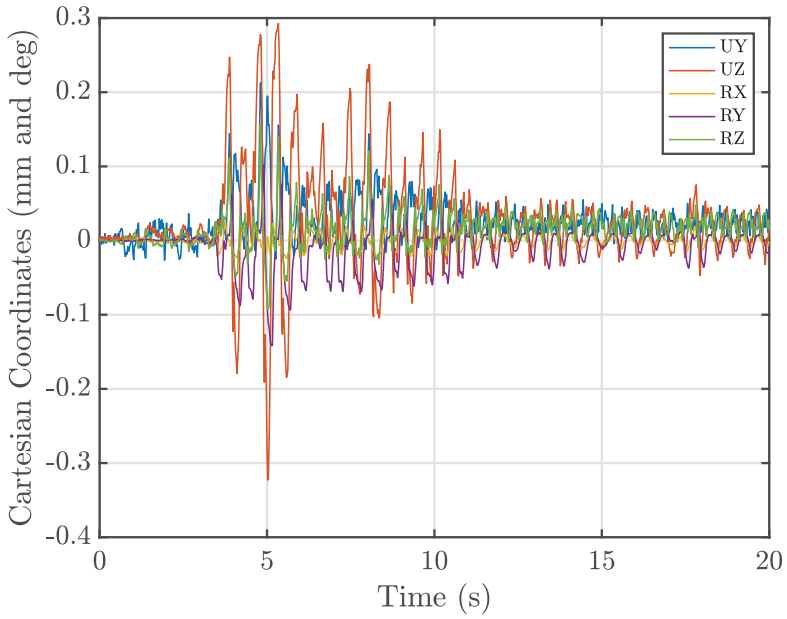


Figure 7.19: Cross-talk responses of Cartesian coordinates, 10% scaled KOB ground motion, feedforward-feedback compensation

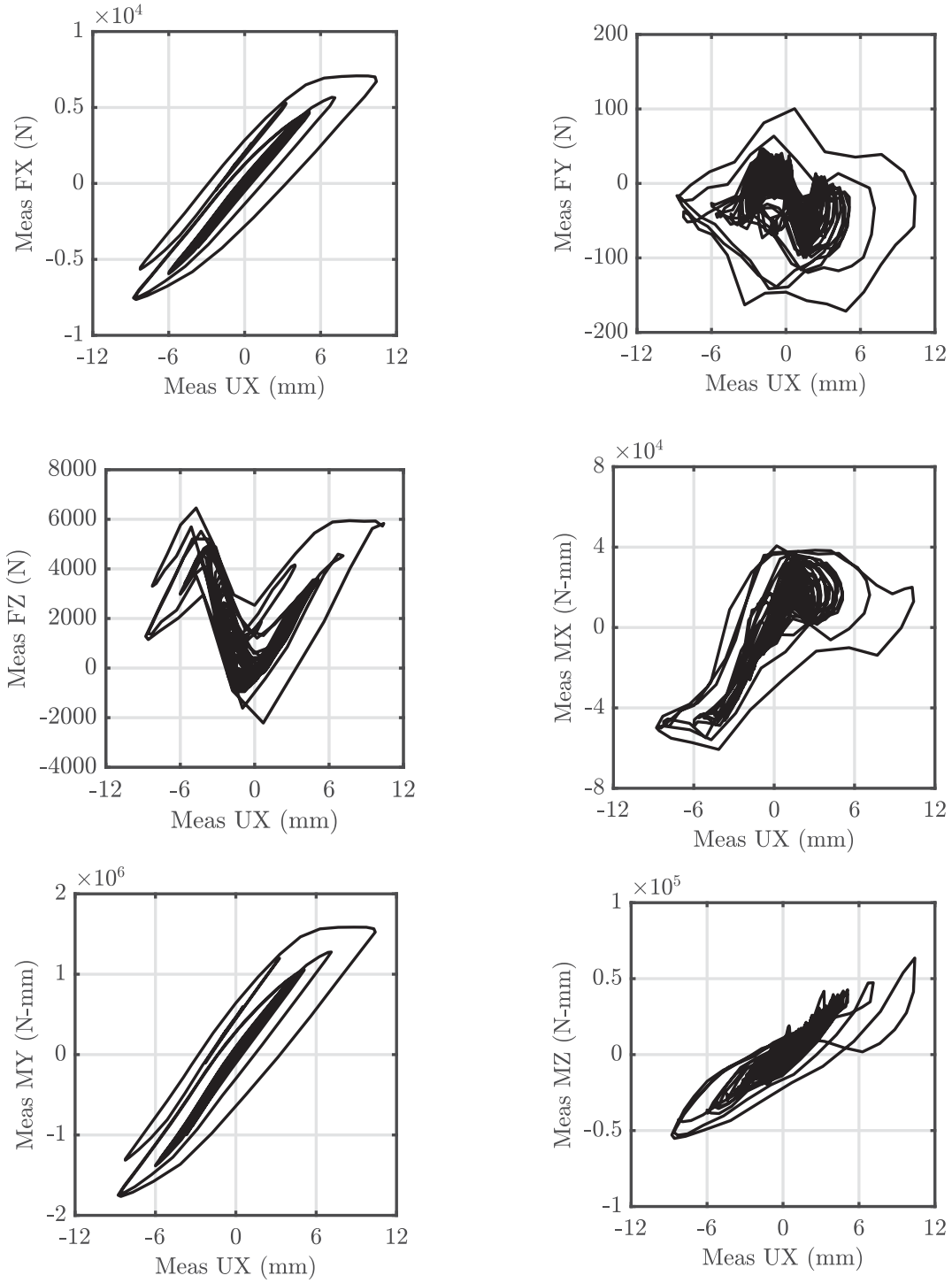


Figure 7.20: Force-displacement relations of test specimen in Cartesian coordinates, subject to 20% scaled ELC ground motion, with feedforward-feedback compensation

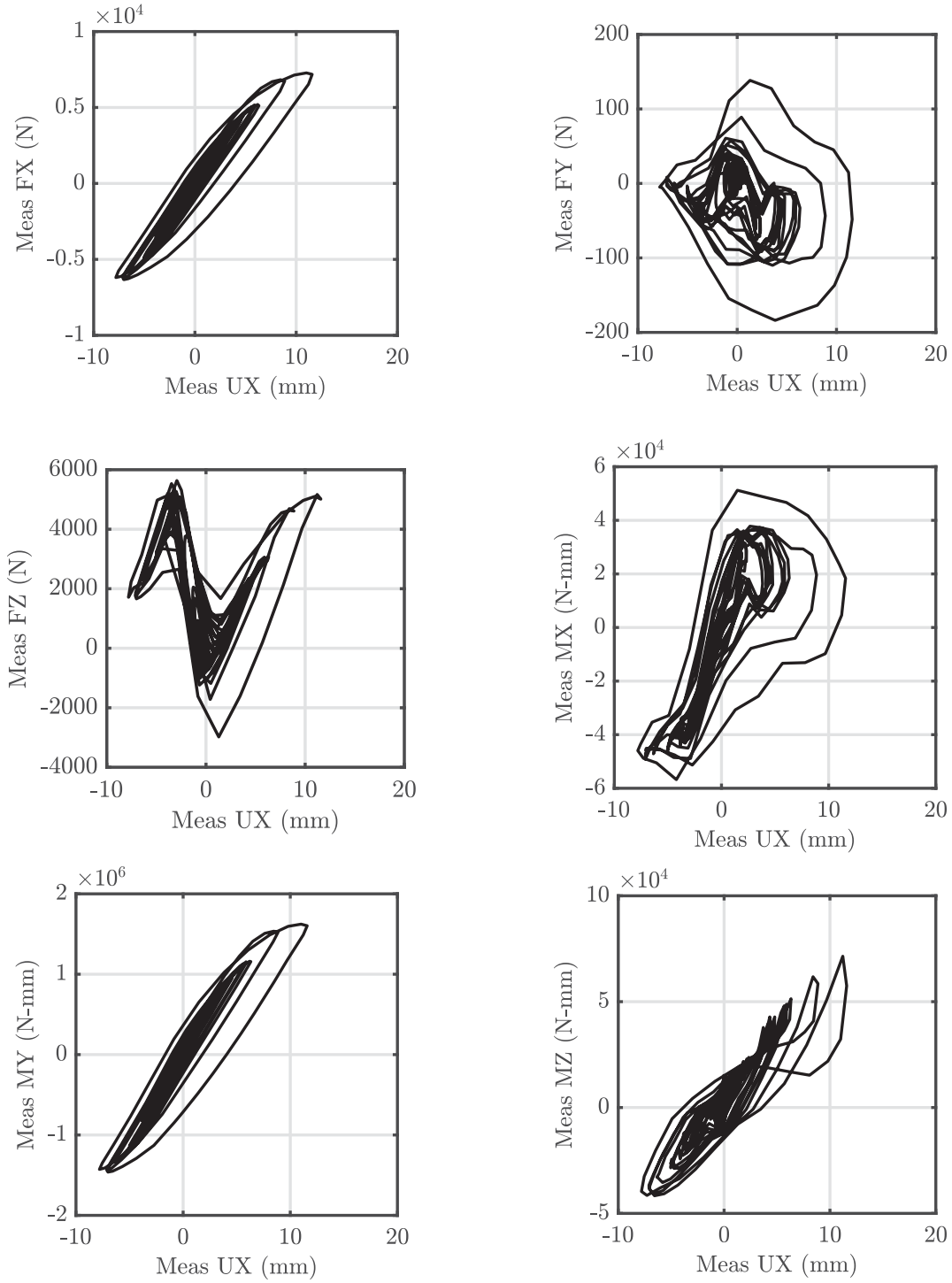


Figure 7.21: Force-displacement relations of test specimen in Cartesian coordinates, subject to 10% scaled NOR ground motion, with feedforward-feedback compensation

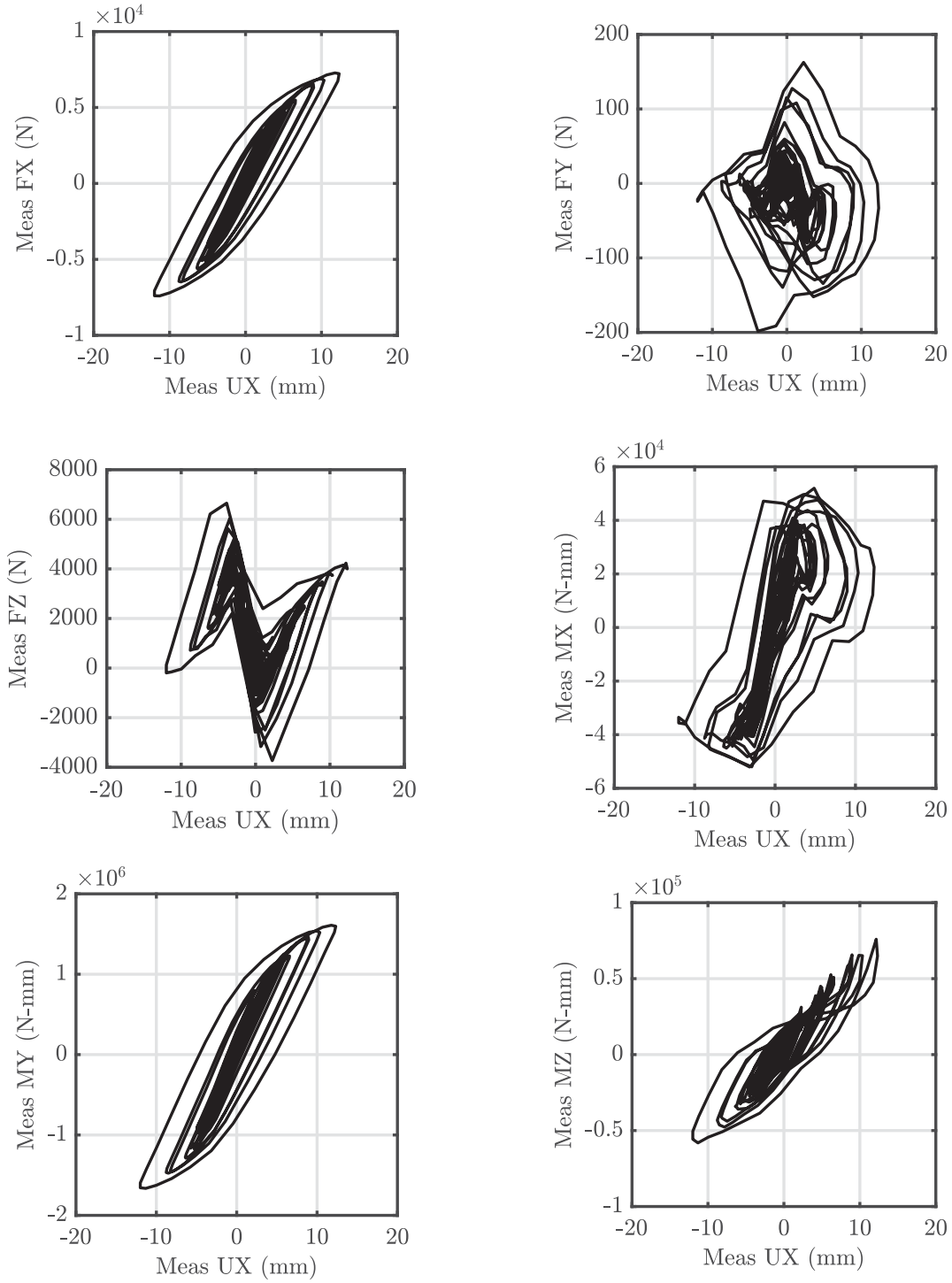
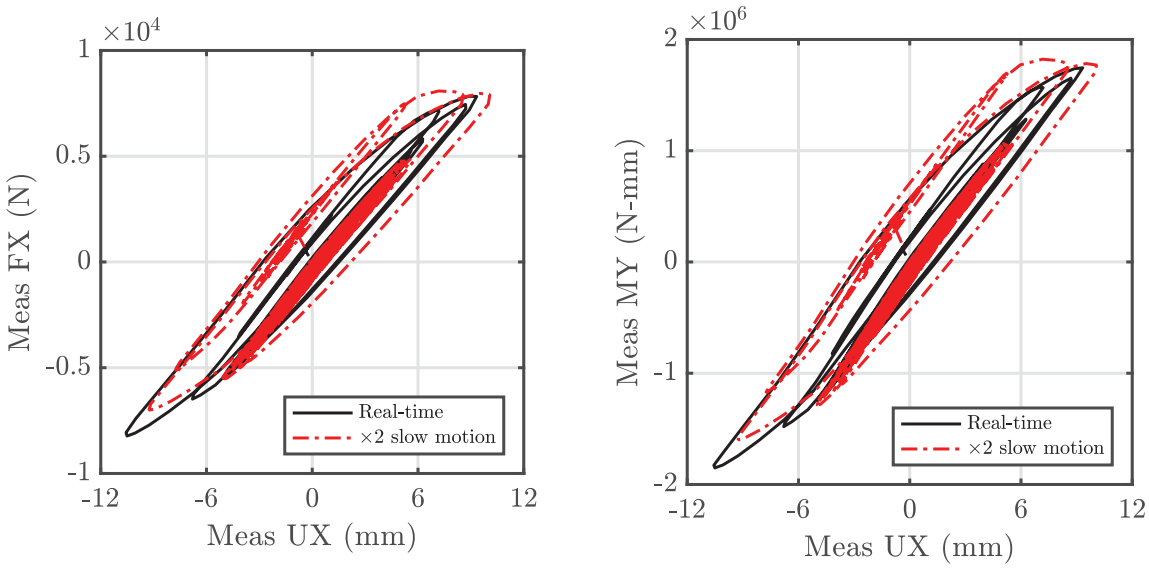


Figure 7.22: Force-displacement relations of test specimen in Cartesian coordinates, subject to 10% scaled KOB ground motion, with feedforward-feedback compensation

7.6.3 Comparisons between real-time and "fast" (continuous) hybrid simulation

In addition, a brief study on the rate effects on the structural response of steel specimens was conducted through the proposed framework. The hybrid system was subjected to 20% scaled ELC ground motion with model-based compensation, and the speed of loading was modified to evaluate the strain rate dependency of the test specimen.

The comparison between real-time test and $\times 2$ slow motion test (i.e., the time axis of the ground motion record was scaled by two times) is presented in Figure 7.23. Clearly, the rate of loading has a significant impact on the structural response that will have an important effect both locally and in the global response of the structural system. Therefore, this evidence is useful to support the needs for maRTHS testing development.



(a) Shear force vs. lateral displacement

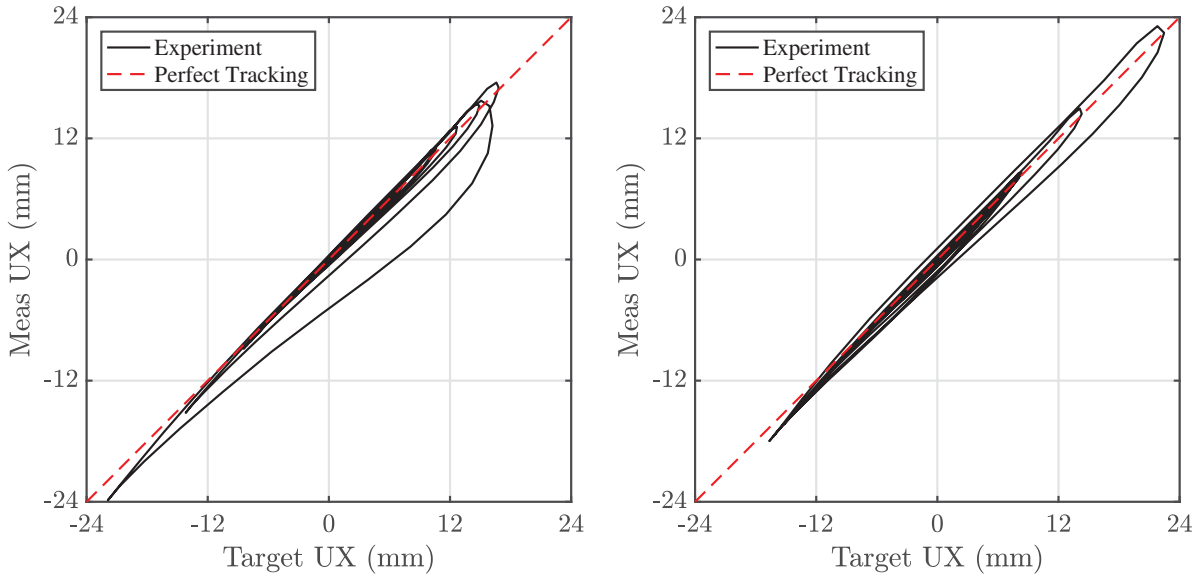
(b) Bending moment vs. lateral displacement

Figure 7.23: Comparison of hysteretic response of test specimen for different rates of deformation, subject to 20% scaled ELC ground motion, with feedforward-feedback compensation

7.7 Limitations of loading assembly

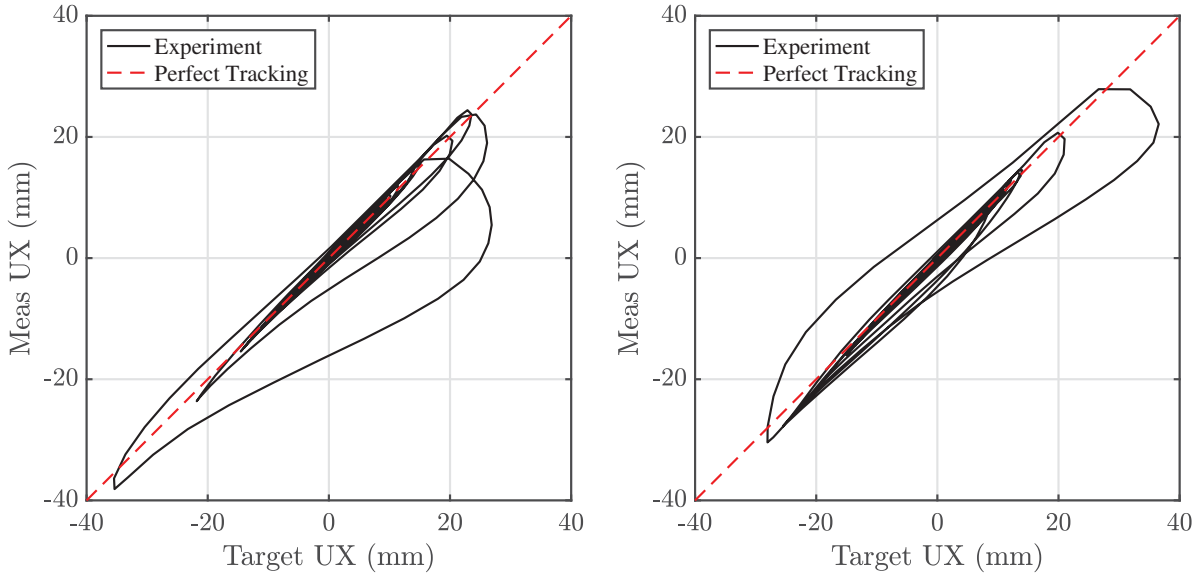
For the case of severe loading, synchronization subspace plots shows a decrease on reference tracking, a phenomenon that is primarily associated to the limits of the loading assembly to satisfy both velocity and force demands over the test specimen.

First, for the case of 40% ELC ground motion, the SSP shown in Figure 7.24 demonstrates that the direction of the ground motion changes the synchronization pattern of the test; the tracking error is smaller when the hybrid system is subjected to $-\ddot{u}_g$ ground acceleration, compared to $+\ddot{u}_g$ ground acceleration. This phenomenon is explained due to the impulsive nature of the selected ground motion, where the larger pulse has a predominant direction, as observed in Figure 7.3a, and the fact that all double-ended actuators have different capacities in extension and retraction. This observation is exacerbated for the case of 60% ELC ground motion, as shown in Figure 7.25. However, if the loading is applied slowly (e.g., $\times 3$ slow motion), then the synchronization errors are significantly reduced as observed in Figure 7.26. Thus, the loading assembly in its current state has particular limitations that confine the practical applications of maRTHS testing for the simulation of extreme events.



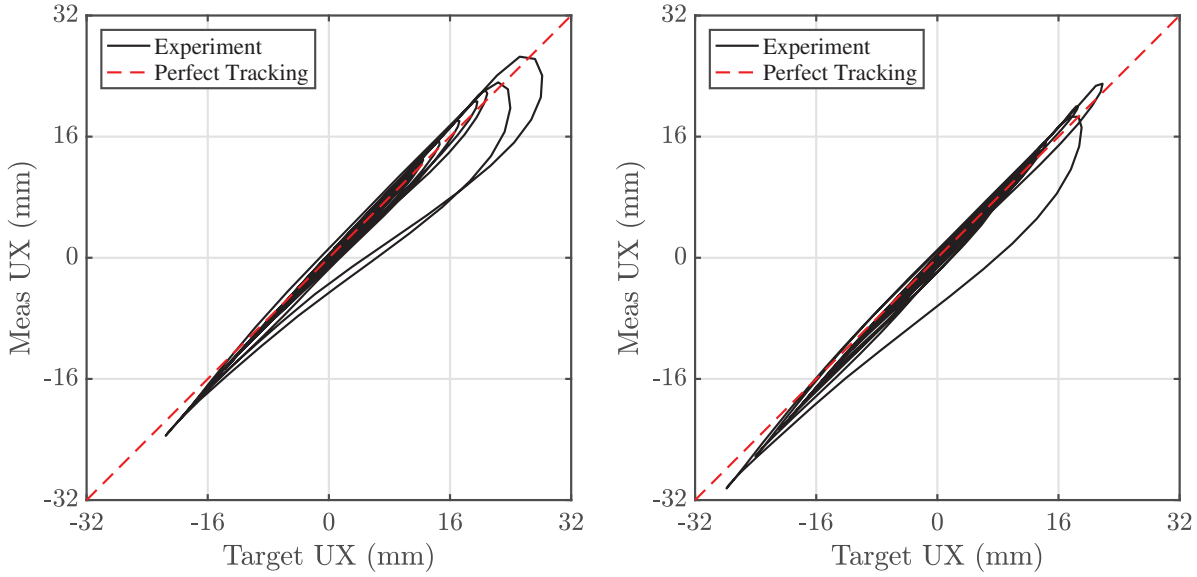
(a) $+\ddot{u}_g$ direction (b) $-\ddot{u}_g$ direction

Figure 7.24: Synchronization subspace plot, 40% ELC ground motion



(a) $+\ddot{u}_g$ direction (b) $-\ddot{u}_g$ direction

Figure 7.25: Synchronization subspace plot, 60% ELC ground motion



(a) $+u_x$ direction (b) $-u_x$ direction

Figure 7.26: Synchronization subspace plot, 60% ELC ground motion, with $\times 3$ slow motion

Recall that all actuators have a flow-pressure relationship associated to the servo-valve dynamics. Therefore, the velocity capacity of the actuator is limited by the pump supply (flow and pressure), and the load force over the piston. If a specimen is attached to the actuator piston, the reaction force of the specimen will limit how fast the piston can move.

To better understand the reasons for this limitations, an estimation of the load flow demand is required. For this purpose, an estimation is obtained from the actuator stroke measurements. First, the "i-th" piston velocity $\dot{q}_i[k]$ is calculated from LVDT stroke measurements $q_i[k]$ using a central difference approach:

$$\dot{q}_i[k] = \frac{q_i[k+1] - q_i[k-1]}{2\Delta t} \qquad \forall i \in [1,6], k \in [0,N] \tag{7.24}$$

To estimate the "i-th" actuator load flow demand $Q_{L,i}^{\text{demand}}[k]$, the following calculation is performed:

$$Q_{L,i}^{\text{demand}}[k] = \begin{cases} A_{\text{ext}}\dot{q}_i[k] & \text{if } \dot{q}_i[k] \geq 0 \\ A_{\text{ret}}\dot{q}_i[k] & \text{if } \dot{q}_i[k] < 0 \end{cases} \qquad \forall i \in [1,6], k \in [0,N] \tag{7.25}$$

where A_{ext} and A_{ret} is the effective piston area of a double-ended actuator when the piston is extending and retracting, respectively:

$$A_{\text{ext}} = \frac{\pi}{4}D^2 \tag{7.26}$$

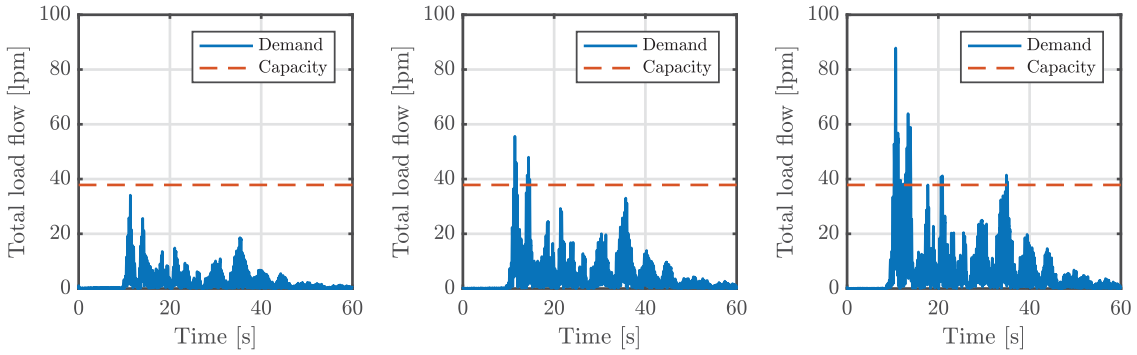
$$A_{\text{ret}} = \frac{\pi}{4}(D^2 - d^2) \tag{7.27}$$

while D and d are the piston bore diameter and piston rod diameter, respectively. Similarly, the total flow demand from the LBCB loading assembly is the summation of the actuator flow demands, such that continuity is satisfied:

$$Q_{\text{LBCB}}^{\text{demand}}[k] = \sum_{i=1}^{6} Q_{L,i}^{\text{demand}}[k] \tag{7.28}$$

Then, a comparison between loading assembly flow demands and pump flow capacity can be obtained. In this study, the pump connected to the LBCB system is working at 37.85 [lpm] at 3,000 [psi]. Figure 7.27 shows the demand vs. capacity relation of total load flow, when the hybrid test is performed in real-time with increased scaling for ELC ground motion. Indeed, the case of 20% ground motion scaling yields a good theoretical balance between total flow demand and pump capacity to run the maRTHS test without prominent limitations. However, when 40% and 60% scaling are considered, the transient flow demands exceeds the available flow capacity of the pump. In these situations, the LBCB does not have enough power to move at the target rate with the prescribed reaction forces due to the presence of the test specimen.

On the other hand, the estimation of pressure drop is required to determine the capacity load flow of each servo valve. The pressure drop is a function of the force sustained by the actuator. Then, the pressure drop of the "i-th" servo valve $\Delta P_i[k]$ is calculated with the



(a) 20% scaled (b) 40% scaled (c) 60% scaled

Figure 7.27: Total flow demand from LBCB vs. pump flow supply, ELC ground motion

following equation:

$$\Delta P_i[k] = \begin{cases} P_S - \frac{\tau_i[k]}{A_{\text{ext}}} & \text{if } \tau_i[k] \geq 0 \\ P_S - \frac{\tau_i[k]}{A_{\text{ret}}} & \text{if } \tau_i[k] < 0 \end{cases} \qquad \forall i \in [1,6], k \in [0,N] \tag{7.29}$$

while P_S is the supply pressure, adjusted to $2,500$ [psi] through the service manifold of the LBCB system; and $\tau_i[k]$ is the "i-th" load cell measurement. In addition, the "i-th" servo pressure drop $\Delta P_i[k]$ ranges between 0 (when actuator reaches the maximum load force) and P_s (when load force is zero).

Subsequently, the "i-th" capacity load flow $Q_{L,i}^{\text{capacity}}[k]$ is determined as follows:

$$Q_{L,i}^{\text{capacity}}[k] = Q_R \sqrt{\frac{\Delta P_i[k]}{\Delta P_R}} \qquad \forall i \in [1,6], k \in [0,N] \tag{7.30}$$

in which Q_R and ΔP_R are the rated flow and pressure drop of the servo valve, with values $Q_R = 10$ [lpm] and $\Delta P_R = 1,000$ [psi]. Clearly, a decrease in pressure drop due to an increase in load force can significantly reduce the load flow capacity for the actuators. Thus, this phenomenon creates a limit over the velocity capacity, with severe limitations over the multi-actuator loading assembly considered for real-time hybrid simulation.

Hence, the empirical flow-pressure relations for each actuator of the LBCB loading assembly when the hybrid system is subjected to 60% scaled ELC ground motion are illustrated in Figures 7.28 through 7.33. Thus, the LBCB system has reached its limit to offer sufficient power to move at the prescribed rate with the sustained load force, as observed by the load flow capacity vs. demand relation for each single actuator. This phenomenon is appreciated at the peak response in actuators X1 and X2 (Figures 7.28 and 7.29, respectively).

Furthermore, the large axial stiffness of the test specimen, and the proximity of the specimen axis to the vertical actuator Z1, produces an extremely large pressure drop, where basically the Z1 actuator reaches $Q_L = 0$ and the actuator is not able to move at all (i.e., the actuator is locked). Indeed, this may be a possible explanation of the reduced synchronization results on the SSP shown in Figures 7.25.

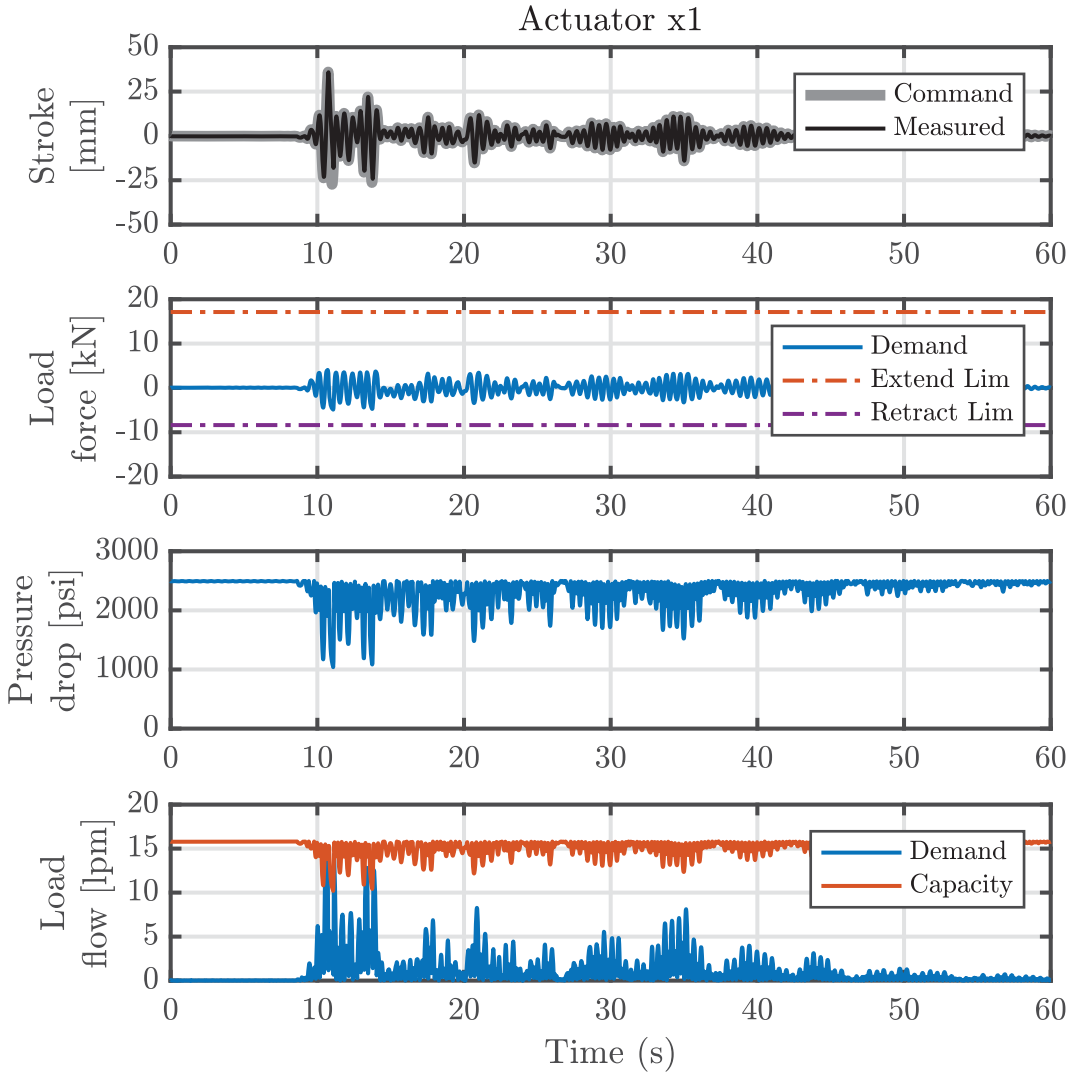


Figure 7.28: Actuator X1 flow-pressure relations, subject to 60% scaled ELC ground motion

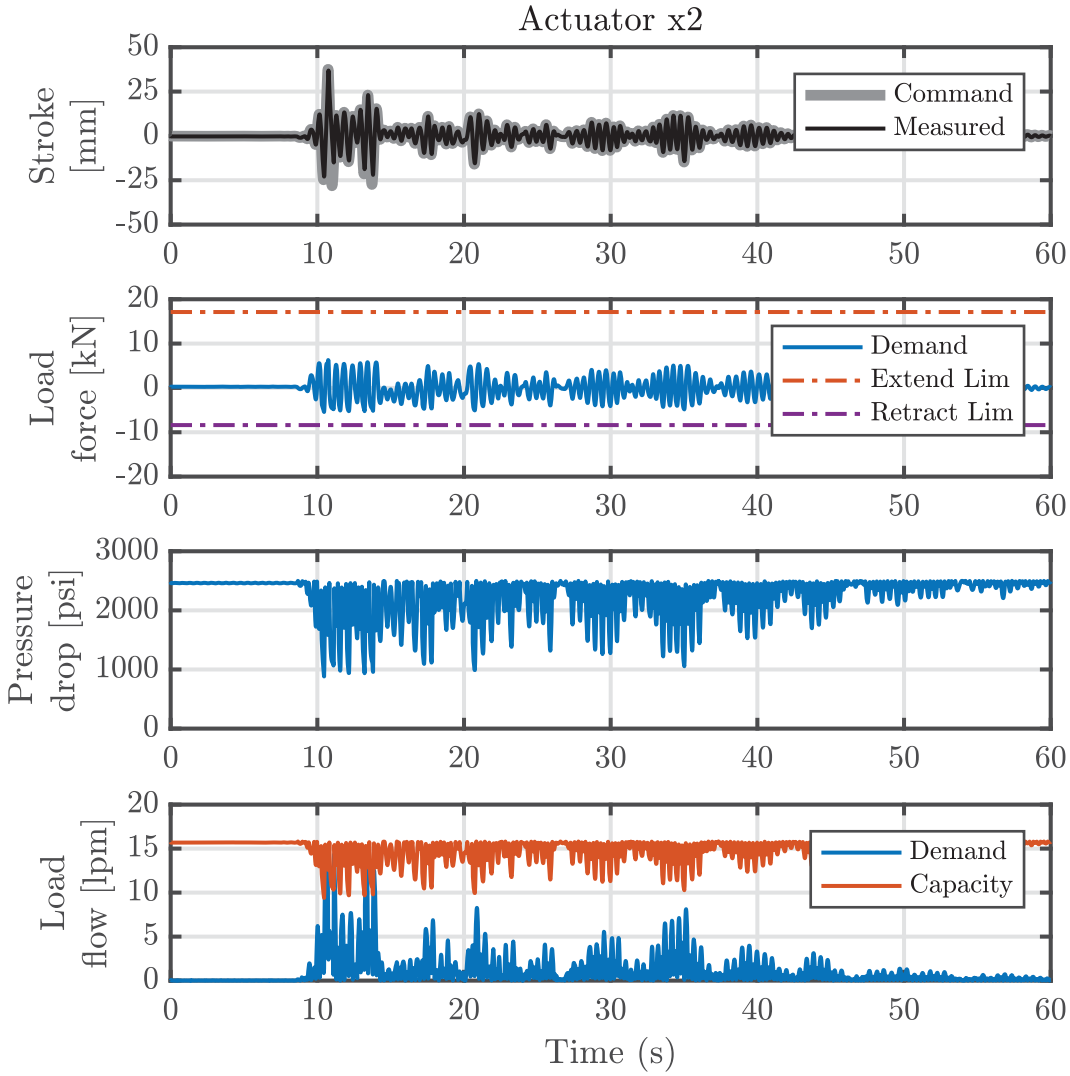


Figure 7.29: Actuator X2 flow-pressure relations, subject to 60% scaled ELC ground motion

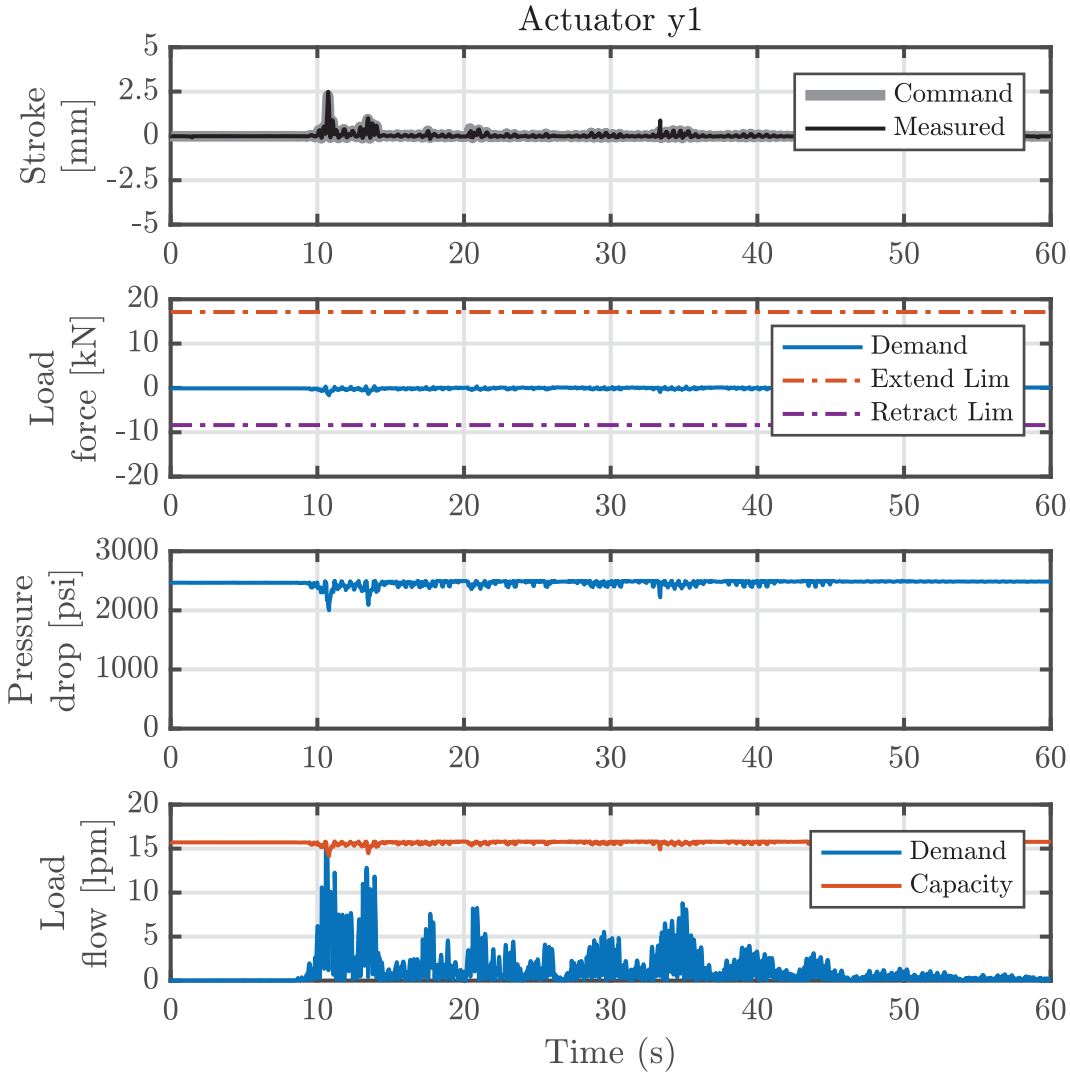


Figure 7.30: Actuator Y1 flow-pressure relations, subject to 60% scaled ELC ground motion

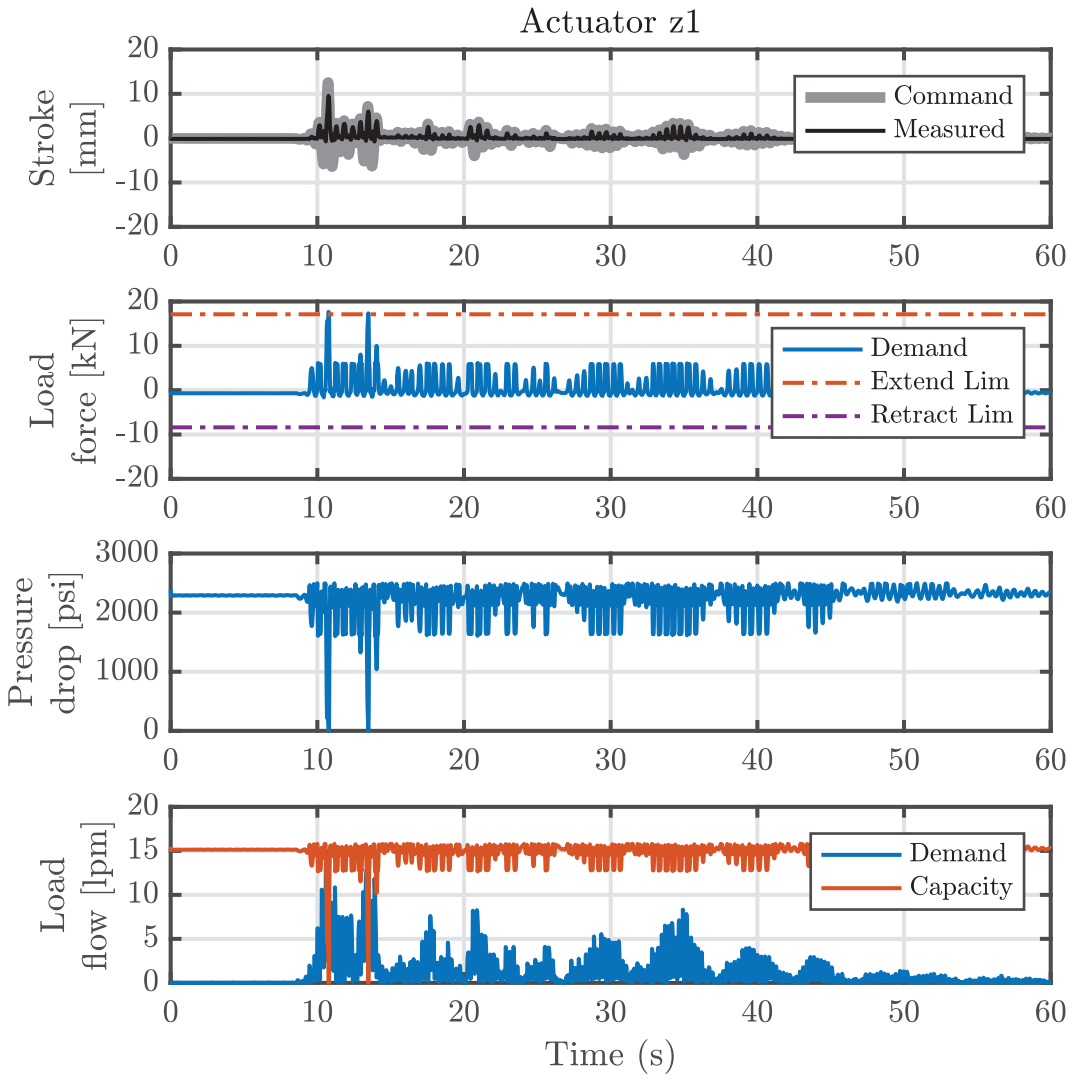


Figure 7.31: Actuator Z1 flow-pressure relations, subject to 60% scaled ELC ground motion

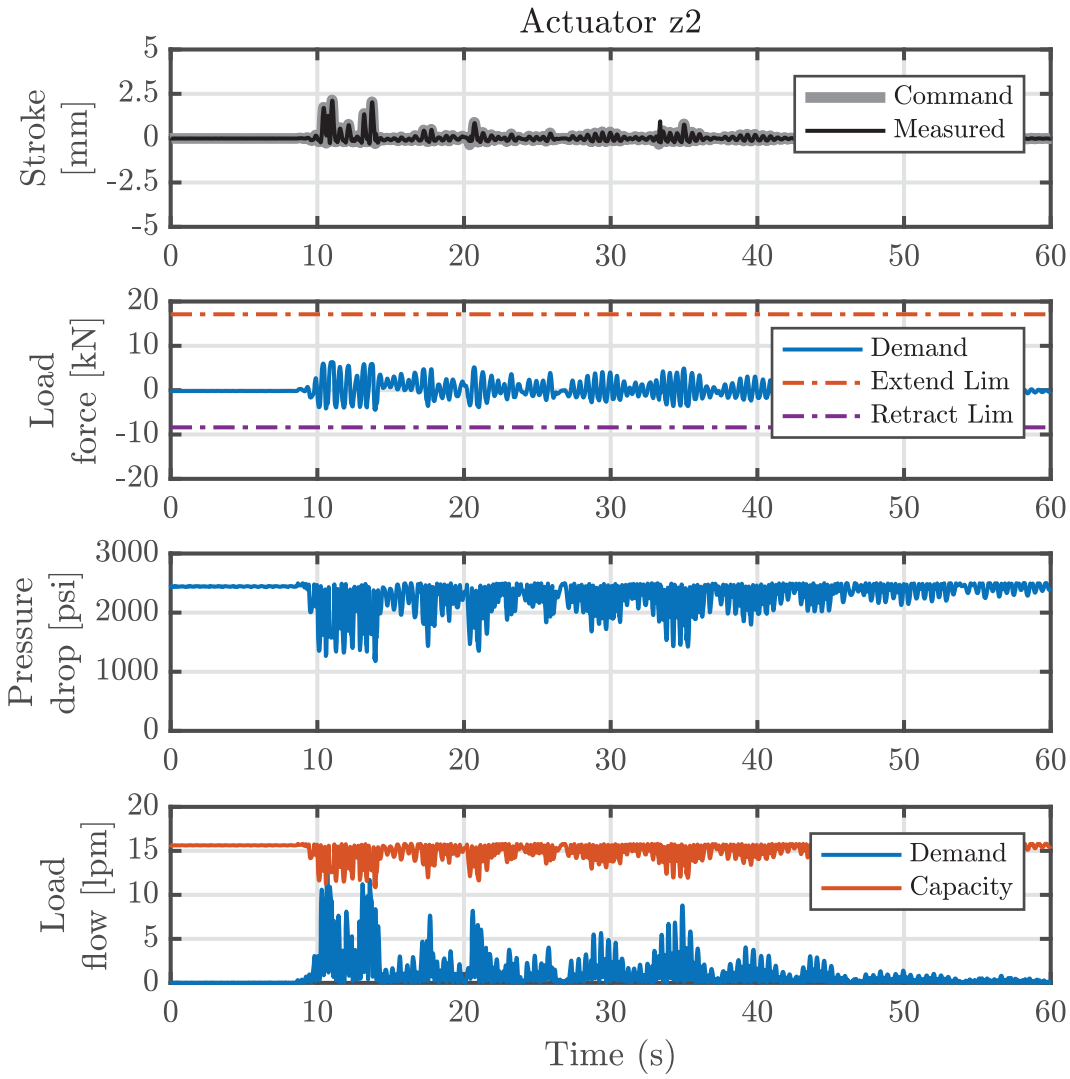


Figure 7.32: Actuator Z2 flow-pressure relations, subject to 60% scaled ELC ground motion

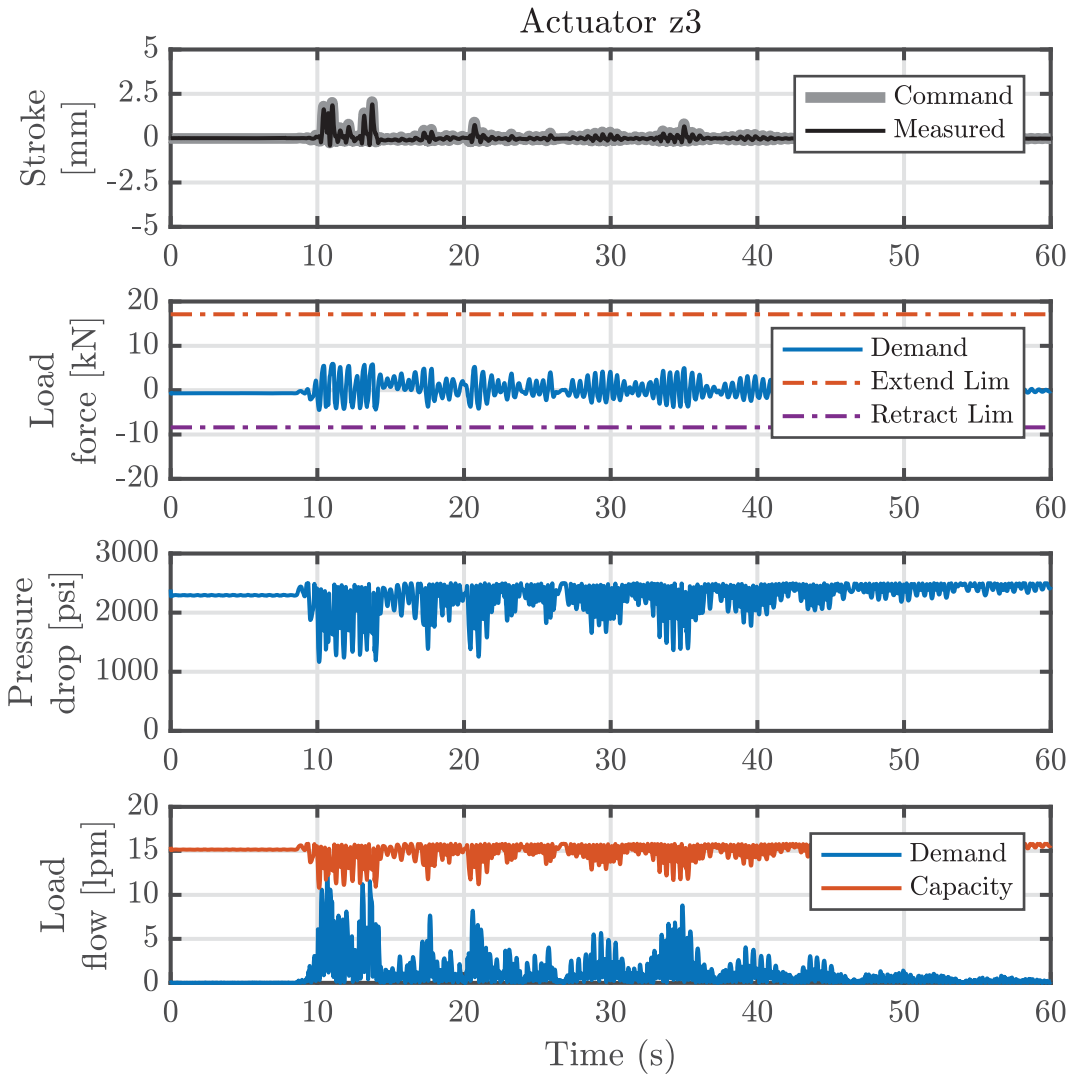


Figure 7.33: Actuator Z3 flow-pressure relations, subject to 60% scaled ELC ground motion

7.8 Summary

In this chapter, the experimental study of a steel column specimen was considered to validate the proposed maRTHS methodology. The reference structure was chosen as a one-story frame building with a single degree-of-freedom associated to its lateral displacement. For substructuring purposes, only the u_x coordinate was considered as the interface degree-of-freedom of the hybrid system, while the other Cartesian coordinates of the loading assembly were commanded a zero signal.

Good reference tracking is achieved when model-based compensation is considered for the dynamic test. Also, the feedforward controller is responsible for the outstanding reference tracking capabilities of the model-based compensator, whereas the performance of the feedback controller did not offer a substantial improvement. In addition, the proposed methodology was able to successfully track all the controlled Cartesian coordinates simultaneously, with tracking errors that are considered be sufficiently small. Moreover, the resulting nonlinear response of the test specimen due to increased seismic excitation did not degrade heavily the tracking performance of the hybrid system. But, for experiments with severe loading, the hybrid system reached its maximum dynamic capacity, which is mainly associated to the hydraulic power supply.

Chapter 8

CONCLUSIONS AND FUTURE STUDIES

8.1 Conclusions

This study presents a framework for multi-axial real-time hybrid simulation (maRTHS). This framework offers the opportunity to increase the class of structures that can be experimentally tested using the hybrid simulation technique. The framework employs a multi-actuator loading assembly to prescribe three-dimensional loads and boundary conditions at the interface between numerical and experimental substructures. The complexities of the implementation are presented, and the methodology for kinematic transformations, equipment calibration, system identification, and control design are discussed in detail.

The implementation was performed in the Newmark Civil Engineering Laboratory at the University of Illinois at Urbana-Champaign. A small-scale Load and Boundary Condition Box (LBCB) was chosen as the multi-actuator loading assembly, controlled in real-time by a dSpace micro-controller unit through an analog interface with the LBCB servo-controller unit. The control algorithms were developed using Matlab/Simulink program, which provides great flexibility for code development and deployment.

Kinematic transformations between Cartesian coordinates and actuator and sensor coordinates were developed. These relationships are fundamental to control the loading platform in Cartesian space using commands to the individual actuators through the servo-controller unit. Two kinematic transformations were introduced: (i) inverse kinematics, from Cartesian to actuator coordinates; and (ii) forward kinematics, from actuator/sensor to Cartesian coordinates. Although both transformations are nonlinear, the former is an explicit function, and it can be easily implemented in this framework. However, the forward kinematic transformation is a nonlinear implicit relationship, without a closed-form solution. Thus, an approximation to this transformation was required for real-time implementation. A linearized forward kinematic transformation was presented, that exhibits good accuracy for fast transformations of the measured Cartesian coordinates of the controlled motion center. The kinematic transformation components for maRTHS testing were validated experimentally using a precise Cartesian contact-less dynamic measuring machine.

Following, a study on the dynamic modeling of the multi-actuator loading assembly in Cartesian space was presented. The models obtained in this chapter were required to develop the model-based compensators that are fundamental for synchronization purposes between hybrid substructures. To achieve a good representation of the physics of the loading

system, system identification for multi-input, multi-output (MIMO) systems was developed. The procedure consisted in the experimental estimation of the frequency response functions (FRF) for the MIMO system. Then, a MIMO transfer function model was identified such that the error between the experimental FRF data and the model was minimized. Although, a good fit for the off-diagonal components of the MIMO system was hard to achieve using the system identification tools. In addition, different challenges were introduced regarding the correct way to obtain state-space models with minimal realizations, which is a requirement for optimal control design. Nevertheless, the models developed in this section matched well with the experimental data for frequency bandwidth of interest, and satisfied all the necessary conditions for control design.

Subsequently, model-based compensation for maRTHS testing was developed. The method consists of a feedforward-feedback architecture for compensation of multi-actuator dynamics for real-time dynamic testing. The feedforward compensator had the role of synchronizing the target (reference) signals from the numerical substructure with the measured signals from the experimental substructure. To this end, the feedforward compensator was designed using an inverse-based model approach, where an experimental FRF was obtained for the inverse system, and then system identification was performed to create an improper transfer function model. The improper terms of this transfer function were approximated a finite differences approach for real-time implementation. In addition, a feedback regulator was introduced to increase the robustness of the system for high-frequency noises and disturbances that can be associated with model uncertainty. Both feedforward and feedback components are designed to meet frequency-domain specifications for improved reference tracking and disturbance/noise rejection for a given operation bandwidth.

Finally, validation of the maRTHS framework was carried out for a single-story building structure. Good reference tracking performance of the target boundary conditions was obtained in an accurate, reliable and stable manner. Although the boundary conditions of the example were rather simple, the potential of the maRTHS framework to control multiple-degree-of-freedom motion in real-time was demonstrated. Moreover, this small-scale implementation can potentially provide a test-bed for future research applications to verify and evaluate rate-dependent materials and components that can be used for the design of structural systems subjected to multi-axial dynamic loading.

8.2 Future Studies

8.2.1 Complex boundary conditions in maRTHS tests

The validation of the maRTHS framework, as described in Chapter 7, considered a structural system with a simple choice of substructuring, where only translational motion was enabled at the interface between substructures. While the framework was developed, validation using a more general substructuring choice would involve the incorporation of all six degrees-of-freedom (DOF) at the end of a beam-column element. Thus, further experimental tests are required to assess the performance of the model-based compensator when imposing all DOFs in Cartesian coordinates using the multi-actuator loading assembly to enhance the capabilities of this framework.

8.2.2 Mixed-mode control in maRTHS testing

The model-based compensation algorithm described in Section 6.2 considered only the reference tracking problem for motion in Cartesian coordinates. But, if other physical quantities are prescribed at the interface between substructures, such as Cartesian forces, then the proposed technique is not suitable for tracking of multi-metric variables.

Thus, by combining mixed-mode control with maRTHS testing, it would be possible to dynamically controlling both displacement and force DOFs simultaneously is possible. This situation would be ideal to impose gravity loads and overturning moments due to seismic actions over three-dimensional test specimens.

To achieve this purpose, a similar procedure described by Nakata et al. (2007) could be employed; however, additional studies are required on the necessary provisions for stable and robust performance in real-time dynamic tests. Alternatively, the additional compliance technique for force-controlled RTHS could be explored (Chae et al., 2018; Shao and Reinhorn, 2012).

8.2.3 Model-based adaptive compensation

The proposed compensation algorithm for multi-actuator loading assemblies was based on a model previously derived using system identification techniques, as described in Chapter 5. The derived models are obtained considering the interaction with the test specimen attached to the loading assembly, by performing a random excitation test. However, this test is usually performed using the test specimen before the maRTHS test takes place. Although the magnitude of random excitation required to complete this task is considered to be small, there is a chance that the specimen could be damaged after completing the system identification procedure, and its structural properties will not be necessarily representative of the undamaged scenario for structural performance assessment purposes. Moreover, the model-based compensator is designed with a fixed linearized model of the experimental substructure. Thus, any change of its structural properties while conducting the maRTHS test could impact the stability and accuracy of the experimental test, especially for nonlinear specimens with strength and stiffness degradation or hardening effects.

Therefore, more studies should be conducted on the improvement of feedforward compensation for nonlinear systems by employing adaptive control techniques (e.g., Chen et al., 2015). This feature could be extremely useful to provide a simpler, rational, and flexible approach for maRTHS testing. For example, system identification and model-based compensator design could be performed once for the bare LBCB system, i.e., without specimen interaction. Then, when the specimen is connected to the LBCB, the adaptation of the feedforward parameters during the real-time test would allow for improved compensation that captures specimen interaction with the multi-actuator system, together with better stability and robustness guarantees for safe execution.

8.2.4 Large-scale implementation of maRTHS framework

In Chapter 7, the validation of the maRTHS framework was performed over a small-scale experimental substructure. Although the tests were satisfactory, further study is re-

quired regarding any potential problems that this framework could experience when large servo-hydraulic actuators are utilized on the multi-actuator loading assembly.

One of the large-scale Load and Boundary Condition Boxes (LBCB) in the Newmark Civil Engineering Laboratory (NCEL) at the University of Illinois at Urbana-Champaign has been upgraded to accommodate dynamic loading, where new servo-valves, manifolds, and accumulators were installed by the manufacturer to allow for the increased dynamical capabilities of the multi-actuator system. Further contributions are anticipated to upgrade the full-scale LBCB dynamic testing equipment at Newmark Civil Engineering Laboratory to implement the maRTHS framework developed throughout this study. Also, continued collaborations with Newmark Civil Engineering Laboratory is expected for the development of user-friendly and flexible interfaces of this proposed framework to allow for practical applications on future experimental testing research. Finally, the development of guidelines to help end-users design and conduct maRTHS experiments will be required as well.

8.2.5 Multi-point interface boundary conditions

The Load and Boundary Condition Boxes (LBCB) available at Newmark Civil Engineering Laboratory (NCEL) are a modular and flexible solution to design hybrid simulation tests for large and complex experimental substructures. In particular, this capability has been considered to impose multi-point interface boundary conditions over large test specimens, by using multiple LBCB systems connected with the test specimen at different locations (e.g., Mahmoud et al., 2013).

Although, the capability of including multiple LBCBs for maRTHS testing have not been considered in this study, leaving an opportunity to extend this framework for the coordination of multiple dynamic LBCBs in real-time. For this matter, is essential to study the associated kinematic constraints of loading assemblies with several actuators coupled through the respective loading platforms as well thought the interactions with the test specimen. In addition, the simultaneous coordination of multiple actuators will impose a heavy burden over the computational resources of the micro-controller if a centralized approach for model-based compensation and integration of numerical substructures is considered. Further studies will be required to understand the restrictions in hardware and software, and find solutions that could accommodate for reliable testing.

8.2.6 Inertial effects in test specimens

The stability and accuracy of the multi-actuator model-based compensator for maRTHS testing was verified both numerically and experimentally in Chapters 6 and 7 respectively. However, this validation was performed under the assumption that the experimental substructure consisted of a test specimen without significant inertial effects, such as large masses physically build and installed over the loading platform.

When a sufficiently large mass is installed on the multi-actuator loading assembly, achieving system stability and robustness could be much harder. Thus, further studies are required to inspect whether the model-based compensation proposed in this framework will be sufficient to compensate for inertial effects when conducting maRTHS testing.

REFERENCES

Abdelnaby, A. E., Frankie, T. M., Elnashai, A. S., Spencer, B. F., Kuchma, D. A., Silva, P., and Chang, C.-M. (2014). Numerical and hybrid analysis of a curved bridge and methods of numerical model calibration. *Engineering Structures*, 70:234–245.

Ahmadizadeh, M. and Mosqueda, G. (2009). Online energy-based error indicator for the assessment of numerical and experimental errors in a hybrid simulation. *Engineering Structures*, 31(9):1987–1996.

Ahmadizadeh, M., Mosqueda, G., and Reinhorn, A. M. (2008). Compensation of actuator delay and dynamics for real-time hybrid structural simulation. *Earthquake Engineering & Structural Dynamics*, 37(1):21–42.

Anastasopoulos, I., Loli, M., Georgarakos, T., and Drosos, V. (2013). Shaking Table Testing of Rocking – Isolated Bridge Pier on Sand. *Journal of Earthquake Engineering*, 17(1):1–32.

Asai, T., Chang, C.-M., Phillips, B. M., and Spencer Jr., B. F. (2013). Real-time hybrid simulation of a smart outrigger damping system for high-rise buildings. *Engineering Structures*, 57:177–188.

Ashwin, P. (1998). Non-linear dynamics, loss of synchronization and symmetry breaking. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 212(3):183–187.

Bendat, J. S. and Piersol, A. G. (2011). *Random data: analysis and measurement procedures*, volume 729. John Wiley & Sons, fourth edition.

Blakeborough, A., Williams, M. S., Darby, A. P., and Williams, D. M. (2001). The development of real–time substructure testing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 359(1786):1869–1891.

Bonnet, P. A., Lim, C. N., Williams, M. S., Blakeborough, A., Neild, S. A., Stoten, D. P., and Taylor, C. A. (2007). Real-time hybrid experiments with Newmark integration, MC-Smd outer-loop control and multi-tasking strategies. *Earthquake Engineering & Structural Dynamics*, 36(1):119–141.

Bonnet, P. A., Williams, M. S., and Blakeborough, A. (2008). Evaluation of numerical time-integration schemes for real-time hybrid testing. *Earthquake Engineering & Structural Dynamics*, 37(13):1467–1490.

Brodersen, M. L., Ou, G., Høgsberg, J., and Dyke, S. J. (2016). Analysis of hybrid viscous damper by real time hybrid simulations. *Engineering Structures*, 126:675–688.

Bunting, G. B. (2016). *Parallel Real-Time Hybrid Simulation of structures using multi-scale models*. PhD thesis, Purdue University, West Lafayette, IN.

Bursi, O. S., Gonzalez-Buelga, A., Vulcan, L., Neild, S. A., and Wagg, D. J. (2008). Novel coupling Rosenbrock-based algorithms for real-time dynamic substructure testing. *Earthquake Engineering & Structural Dynamics*, 37(3):339–360.

Bursi, O. S., He, L., Lamarche, C.-P., and Bonelli, A. (2010). Linearly Implicit Time Integration Methods for Real-Time Dynamic Substructure Testing. *Journal of Engineering Mechanics*, 136(11):1380–1389.

Bursi, O. S., Jia, C., Vulcan, L., Neild, S. A., and Wagg, D. J. (2011). Rosenbrock-based algorithms and subcycling strategies for real-time nonlinear substructure testing. *Earthquake Engineering & Structural Dynamics*, 40(1):1–19.

Buttazzo, G. C. (2011). *Hard Real-Time Computing Systems*, volume 24 of *Real-Time Systems Series*. Springer Science & Business Media, third edition.

Butterworth, J. A., Pao, L. Y., and Abramovitch, D. Y. (2008). The effect of nonminimum-phase zero locations on the performance of feedforward model-inverse control techniques in discrete-time systems. In *Proceedings of the American Control Conference*, pages 2696–2702.

Carrion, J. E. and Spencer, B. F. (2007). Model-based Strategies for Real-time Hybrid Testing. Technical Report NSEL-006, Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL.

Carrion, J. E., Spencer Jr., B. F., and Phillips, B. M. (2009). Real-time hybrid simulation for structural control performance assessment. *Earthquake Engineering & Engineering Vibration*, 8(4):481–492.

Castaneda, N. E., Gao, X., and Dyke, S. J. (2012). RT-Frame2D: A Computational Platform for the Real-Time Hybrid Simulation of Dynamically-excited Steel Frame Structures (Version 1.0)[Computer software].

Cha, Y.-J., Agrawal, A. K., Friedman, A., Phillips, B. M., Ahn, R., Dong, B., Dyke, S. J., Spencer, B. F., Ricles, J., and Christenson, R. E. (2014). Performance Validations of Semiactive Controllers on Large-Scale Moment-Resisting Frame Equipped with 200-kN MR Damper Using Real-Time Hybrid Simulations. *Journal of Structural Engineering*, 140(10):04014066.

Chae, Y., Kazemibidokhti, K., and Ricles, J. M. (2013a). Adaptive time series compensator for delay compensation of servo-hydraulic actuator systems for real-time hybrid simulation. *Earthquake Engineering & Structural Dynamics*, 42(11):1697–1715.

Chae, Y., Park, M., Kim, C.-Y., and Park, Y. S. (2017). Experimental study on the rate-dependency of reinforced concrete structures using slow and real-time hybrid simulations. *Engineering Structures*, 132:648–658.

Chae, Y., Rabiee, R., Dursun, A., and Kim, C.-Y. (2018). Real-time force control for servo-hydraulic actuator systems using adaptive time series compensator and compliance springs. *Earthquake Engineering & Structural Dynamics*, 47(4):854–871.

Chae, Y., Ricles, J. M., and Sause, R. (2013b). Modeling of a large-scale magneto-rheological damper for seismic hazard mitigation. Part II: Semi-active mode. *Earthquake Engineering & Structural Dynamics*, 42(5):669–685.

Chae, Y., Ricles, J. M., and Sause, R. (2014). Large-scale real-time hybrid simulation of a three-story steel frame building with magneto-rheological dampers. *Earthquake Engineering & Structural Dynamics*, 139(7):1215–1226.

Chang, C.-M., Frankie, T. M., Spencer, B. F., and Kuchma, D. A. (2015). Multiple Degrees of Freedom Positioning Correction for Hybrid Simulation. *Journal of Earthquake Engineering*, 19(2):277–296.

Chang, S.-Y. (2002). Explicit Pseudodynamic Algorithm with Unconditional Stability. *Journal of Engineering Mechanics*, 128(9):935–947.

Chen, C. and Ricles, J. M. (2008). Development of Direct Integration Algorithms for Structural Dynamics Using Discrete Control Theory. *Journal of Engineering Mechanics*, 134(8):676–683.

Chen, C. and Ricles, J. M. (2009). Analysis of actuator delay compensation methods for real-time testing. *Engineering Structures*, 31(11):2643–2655.

Chen, C. and Ricles, J. M. (2010). Tracking Error-Based Servohydraulic Actuator Adaptive Compensation for Real-Time Hybrid Simulation. *Journal of Structural Engineering*, 136(4):432–440.

Chen, P.-C., Chang, C.-M., Spencer, Jr., B. F., and Tsai, K.-C. (2015). Adaptive model-based tracking control for real-time hybrid simulation. *Bulletin of Earthquake Engineering*, 13(6):1633–1653.

Chen, P.-C. and Tsai, K.-C. (2013). Dual compensation strategy for real-time hybrid testing. *Earthquake Engineering & Structural Dynamics*, 42(1):1–23.

Chen, P.-C., Tsai, K.-C., and Lin, P.-Y. (2014). Real-time hybrid testing of a smart base isolation system. *Earthquake Engineering & Structural Dynamics*, 43(1):139–158.

Christenson, R. E., Dyke, S. J., Zhang, J., Mosqueda, G., Chen, C., Nakata, N., Laplace, P., Song, W., Chae, Y., Marshall, G., Ou, G., Riascos Gonzales, C. A., and Song, C. (2014). Hybrid simulation: a discussion of current assessment measures. Technical report, George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES), West Lafayette, IN.

Christopoulos, C. and Montgomery, M. (2013). Viscoelastic coupling dampers (VCDs) for enhanced wind and seismic performance of high-rise buildings. *Earthquake Engineering & Structural Dynamics*, 42(15):2217–2233.

Darby, A. P., Blakeborough, A., and Williams, M. (1999). Real-Time Substructure Tests Using Hydraulic Actuator. *Journal of Engineering Mechanics*, 125(10):1133–1139.

Darby, A. P., Williams, M. S., and Blakeborough, A. (2002). Stability and Delay Compensation for Real-Time Substructure Testing. *Journal of Engineering Mechanics*, 128(12):1276–1284.

de Klerk, D., Rixen, D. J., and Voormeeren, S. (2008). General Framework for Dynamic Substructuring: History, Review and Classification of Techniques. *AIAA Journal*, 46(5):1169–1181.

Dermitzakis, S. N. and Mahin, S. A. (1985). Development of Substructuring Techniques for On-Line Computer Controlled Seismic Performance Testing. Technical Report UCB/EERC-85/04, Earthquake Engineering Research Center, Berkeley, CA.

Devasia, S. (2002). Should model-based inverse inputs be used as feedforward under plant uncertainty? *IEEE Transactions on Automatic Control*, 47(11):1865–1871.

Dimig, J., Shield, C. K., French, C. W., Bailey, F., and Clark, A. (1999). Effective Force Testing: A Method of Seismic Simulation for Structural Testing. *Journal of Structural Engineering*, 125(9):1028–1037.

Dong, B., Sause, R., and Ricles, J. M. (2015). Accurate real-time hybrid earthquake simulations on large-scale MDOF steel structure with nonlinear viscous dampers. *Earthquake Engineering & Structural Dynamics*, 44(12):2035–2055.

Dyke, S. J., Spencer Jr., B. F., Quast, P., and Sain, M. (1995). Role of Control-Structure Interaction in Protective System Design. *Journal of Engineering Mechanics*, 121(2):322–338.

Elanwar, H. H. and Elnashai, A. S. (2016a). Application of In-Test Model Updating to Earthquake Structural Assessment. *Journal of Earthquake Engineering*, 20(1):62–79.

Elanwar, H. H. and Elnashai, A. S. (2016b). Framework for Online Model Updating in Earthquake Hybrid Simulations. *Journal of Earthquake Engineering*, 20(1):80–100.

Elnashai, A. S., Spencer, Jr., B. F., Kuchma, D., Ghaboussi, J., Hashash, Y., and Gan, Q. (2004). Multi-axial full-scale sub-structured testing and simulation (MUST-SIM) facility at the University of Illinois at Urbana-Champaign. In *13th World Conference on Earthquake Engineering*, number 1756, Vancouver, Canada.

Fermandois, G. A. and Spencer, Jr., B. F. (2017). Model-based framework for multi-axial real-time hybrid simulation testing. *Earthquake Engineering and Engineering Vibration*, 16(4):671–691.

Frankie, T. M., Abdelnaby, A. E., Silva, P., Sanders, D., Elnashai, A. S., Spencer, Jr., B. F., Kuchma, D., and Chang, C.-M. (2013). Hybrid Simulation of Curved Four-Span Bridge: Comparison of Numerical and Hybrid Experimental/Analytical Results and Methods of Numerical Model Calibration. In *ASCE Structures Congress 2013*, pages 721–732, Reston, VA.

Franklin, G. F., Powell, J. D., and Emami-Naeini, A. (2015). *Feedback control of dynamic systems*. Pearson, seventh edition.

French, C. W., Schultz, A. E., Hajjar, J. F., Shield, C. K., Ernie, D. W., Dexter, R. J., Du, D. H.-C., Olson, S. A., Daugherty, D. J., and Wan, C. P. (2004). Multi-axial Subassemblage Testing (MAST) System: Description and Capabilities. In *13th World Conference on Earthquake Engineering*, Vancouver, Canada.

Friedman, A. J., Dyke, S. J., Phillips, B. M., Ahn, R., Dong, B., Chae, Y., Castaneda, N., Jiang, Z., Zhang, J., Cha, Y.-J., Ozdagli, A. I., Spencer, B. F., Ricles, J., Christenson, R. E., Agrawal, A., and Sause, R. (2015). Large-Scale Real-Time Hybrid Simulation for Evaluation of Advanced Damping System Performance. *Journal of Structural Engineering*, 141(6):04014150.

Gao, X., Castaneda, N., and Dyke, S. J. (2013). Real time hybrid simulation: from dynamic system, motion control to experimental error. *Earthquake Engineering & Structural Dynamics*, 42(6):815–832.

Gao, X., Castaneda, N., and Dyke, S. J. (2014). Experimental Validation of a Generalized Procedure for MDOF Real-Time Hybrid Simulation. *Journal of Engineering Mechanics*, 140(4):04013006.

Guo, T., Chen, C., Xu, W., and Sanchez, F. (2014). A frequency response analysis approach for quantitative assessment of actuator tracking for real-time hybrid simulation. *Smart Materials and Structures*, 23(4):045042.

Hakuno, M., Shidawara, M., and Hara, T. (1969). Dynamic destructive test of a cantilever beam, controlled by an analog-computer. *Proceedings of the Japan Society of Civil Engineers*, 1969(171):1–9.

Hashemi, M. J., Al-Mahaidi, R., Kalfat, R., and Burnett, G. (2015). Development and validation of multi-axis substructure testing system for full-scale experiments. *Australian Journal of Structural Engineering*, 16(4):302–315.

Hashemi, M. J., Masroor, A., and Mosqueda, G. (2014). Implementation of online model updating in hybrid simulation. *Earthquake Engineering & Structural Dynamics*, 43(3):395–412.

Hashemi, M. J. and Mosqueda, G. (2014). Innovative substructuring technique for hybrid simulation of multistory buildings through collapse. *Earthquake Engineering & Structural Dynamics*, 43(14):2059–2074.

Hashemi, M. J., Tsang, H.-H., Al-Ogaidi, Y., Wilson, J. L., and Al-Mahaidi, R. (2017). Collapse Assessment of Reinforced Concrete Building Columns through Multi-Axis Hybrid Simulation. *ACI Structural Journal*, 114(2):437–449.

Hespanha, J. P. (2009). *Linear systems theory*. Princeton University Press.

Hessabi, R. M. and Mercan, O. (2012). Phase and amplitude error indices for error quantification in pseudodynamic testing. *Earthquake Engineering & Structural Dynamics*, 41(10):1347–1364.

Horiuchi, T., Inoue, M., Konno, T., and Namita, Y. (1999). Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber. *Earthquake Engineering & Structural Dynamics*, 28(10):1121–1141.

Horiuchi, T. and Konno, T. (2001). A new method for compensating actuator delay in real-time hybrid experiments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 359(1786):1893–1909.

Horiuchi, T., Nakagawa, M., Sugano, M., and Konno, T. (1996). Development of a Real-time Hybrid Experimental System with Actuator Delay Compensation. In *Proceedings of 11th World Conference in Earthquake Engineering*, number 660, Acapulco, Mexico.

Huang, H.-M., Tidwell, T., Gill, C., Lu, C., Gao, X., and Dyke, S. J. (2010). Cyber-Physical Systems for Real-Time Hybrid Structural Testing: A Case Study. In *1st ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS'10)*, Stockholm, Sweden.

Hung, C.-C. and El-Tawil, S. (2009). A method for estimating specimen tangent stiffness for hybrid simulation. *Earthquake Engineering & Structural Dynamics*, 38(1):115–134.

ICDO (2016). Natural Disasters. Retrieved from http://www.icdo.org/en/disasters/natural-disasters.

Jiang, Z., Kim, S. J., Plude, S., and Christenson, R. E. (2013). Real-time hybrid simulation of a complex bridge model with MR dampers using the convolution integral method. *Smart Materials and Structures*, 22(10):105008.

Jung, R.-Y., Shing, P. B., Stauffer, E., and Thoen, B. (2007). Performance of a real-time pseudodynamic test system considering nonlinear structural response. *Earthquake Engineering & Structural Dynamics*, 36(12):1785–1809.

Karavasilis, T. L., Ricles, J. M., Marullo, T., and Chen, C. (2009). HybridFEM. A program for nonlinear dynamic time history analysis and real-time hybrid simulation of structures. Technical Report ATLSS-09-08, Lehigh University.

Kim, H.-J. and Christopoulos, C. (2008). Friction Damped Posttensioned Self-Centering Steel Moment-Resisting Frames. *Journal of Structural Engineering*, 134(11):1768–1779.

Kim, S. B., Spencer, Jr., B. F., and Yun, C.-B. (2005). Frequency Domain Identification of Multi-Input, Multi-Output Systems Considering Physical Relationships between Measured Variables. *Journal of Engineering Mechanics*, 131(5):461–472.

Kim, S. J., Christenson, R. E., Phillips, B. M., and Spencer, Jr., B. F. (2012). Geographically Distributed Real-Time Hybrid Simulation of MR Dampers for Seismic Hazard Mitigation. In *Proceedings of the 20th Analysis and Computation Specialty Conference*, pages 382–393, Reston, VA.

Kim, S. J., Christenson, R. E., Wojtkiewicz, S. F., and Johnson, E. A. (2011a). Real-time hybrid simulation using the convolution integral method. *Smart Materials and Structures*, 20(2):025024.

Kim, S. J., Holub, C. J., and Elnashai, A. S. (2011b). Experimental investigation of the behavior of RC bridge piers subjected to horizontal and vertical earthquake motion. *Engineering Structures*, 33(7):2221–2235.

Kolay, C. and Ricles, J. M. (2014). Development of a family of unconditionally stable explicit direct integration algorithms with controllable numerical energy dissipation. *Earthquake Engineering & Structural Dynamics*, 43(9):1361–1380.

Kolay, C. and Ricles, J. M. (2017). Improved Explicit Integration Algorithms for Structural Dynamic Analysis with Unconditional Stability and Controllable Numerical Dissipation. *Journal of Earthquake Engineering*, pages 1–22.

Kolay, C., Ricles, J. M., Marullo, T. M., Mahvashmohammadi, A., and Sause, R. (2014). Implementation and application of the unconditionally stable explicit parametrically dissipative KR-α method for real-time hybrid simulation. *Earthquake Engineering & Structural Dynamics*.

Kwon, O.-S., Nakata, N., Park, K.-S., Elnashai, A. S., and Spencer, Jr., B. F. (2007). UI-SIMCOR: User manual and examples (Version 2.6)[Computer software].

Lamarche, C.-P., Bonelli, A., Bursi, O. S., and Tremblay, R. (2009). A Rosenbrock-W method for real-time dynamic substructuring and pseudo-dynamic testing. *Earthquake Engineering & Structural Dynamics*, 38(9):1071–1092.

Lamarche, C.-P., Tremblay, R., Léger, P., Leclerc, M., and Bursi, O. S. (2010). Comparison between real-time dynamic substructuring and shake table testing techniques for nonlinear seismic applications. *Earthquake Engineering & Structural Dynamics*, 39(12):1299–1320.

Lee, C. and Salapaka, S. M. (2009). Robust broadband nanopositioning: fundamental trade-offs, analysis, and design in a two-degree-of-freedom control framework. *Nanotechnology*, 20(3):035501.

Li, X., Ozdagli, A. I., Dyke, S. J., Lu, X., and Christenson, R. E. (2017). Development and Verification of Distributed Real-Time Hybrid Simulation Methods. *Journal of Computing in Civil Engineering*, 31(4):04017014.

Li, Y., Li, J., Li, W., and Samali, B. (2013). Development and characterization of a magnetorheological elastomer based adaptive seismic isolator. *Smart Materials and Structures*, 22(3):035005.

Lim, C. N., Neild, S. A., Stoten, D. P., Drury, D., and Taylor, C. A. (2007). Adaptive Control Strategy for Dynamic Substructuring Tests. *Journal of Engineering Mechanics*, 133(8):864–873.

Lin, F., Maghareh, A., Dyke, S. J., and Lu, X. (2015). Experimental implementation of predictive indicators for configuring a real-time hybrid simulation. *Engineering Structures*, 101:427–438.

Liu, Y., Goorts, K., Ashasi-Sorkhabi, A., Mercan, O., and Narasimhan, S. (2016). A state space-based explicit integration method for real-time hybrid simulation. *Structural Control and Health Monitoring*, 23(4):641–658.

Ljung, L. (2010). Perspectives on system identification. *Annual Reviews in Control*, 34(1):1–12.

Lombardi, D., Bhattacharya, S., and Muir Wood, D. (2013). Dynamic soil–structure interaction of monopile supported wind turbines in cohesive soil. *Soil Dynamics and Earthquake Engineering*, 49:165–180.

Lunenburg, J. J. M., van de Wal, M. M. J., Bosgra, O. H., and Oomen, T. A. E. (2009). Inversion-Based Feedforward Design for Beyond Rigid Body Systems: A Literature Survey. Technical Report DCT 2009.105, Eindhoven University of Technology.

Maghareh, A., Dyke, S. J., Prakash, A., and Bunting, G. B. (2014). Establishing a predictive performance indicator for real-time hybrid simulation. *Earthquake Engineering & Structural Dynamics*, 43(15):2299–2318.

Maghareh, A., Dyke, S. J., Rabieniaharatbar, S., and Prakash, A. (2016a). Predictive stability indicator: a novel approach to configuring a real-time hybrid simulation. *Earthquake Engineering & Structural Dynamics*.

Maghareh, A., Waldbjørn, J. P., Dyke, S. J., Prakash, A., and Ozdagli, A. I. (2016b). Adaptive multi-rate interface: development and experimental verification for real-time hybrid simulation. *Earthquake Engineering & Structural Dynamics*, 45(9):1411–1425.

Mahin, S. A. and Shing, P. B. (1985). Pseudodynamic Method for Seismic Testing. *Journal of Structural Engineering*, 111(7):1482–1503.

Mahin, S. A., Shing, P. B., Thewalt, C. R., and Hanson, R. D. (1989). Pseudodynamic Test Method – Current Status and Future Directions. *Journal of Structural Engineering*, 115(8):2113–2128.

Mahmoud, H. N., Elnashai, A. S., Spencer, Jr., B. F., Kwon, O.-S., and Bennier, D. J. (2013). Hybrid Simulation for Earthquake Response of Semirigid Partial-Strength Steel Frames. *Journal of Structural Engineering*, 139(7):1134–1148.

McCrum, D. and Williams, M. (2016). An overview of seismic hybrid testing of engineering structures. *Engineering Structures*, 118:240–261.

Mercan, O. and Ricles, J. M. (2009). Experimental Studies on Real-Time Testing of Structures with Elastomeric Dampers. *Journal of Structural Engineering*, 135(9):1124–1133.

Mercan, O., Ricles, J. M., Sause, R., and Marullo, T. (2009). Kinematic transformations for planar multi-directional pseudodynamic testing. *Earthquake Engineering & Structural Dynamics*, 38(9):1093–1119.

Merlet, J.-P. (2006). *Parallel robots*, volume 128 of *Solid Mechanics and its Applications*. Springer Science & Business Media, second edition.

Merritt, H. E. (1967). *Hydraulic control systems*. John Wiley and Sons.

Moog (2014). Moog G631/631 Series Flow Control Servo Valves. Technical Report Rev. L, January 2014.

Mosqueda, G., Stojadinovic, B., and Mahin, S. A. (2004). Geographically distributed continuous hybrid simulation. In *13th World Conference on Earthquake Engineering*, number 0959, Vancouver, Canada.

Mosqueda, G., Stojadinovic, B., and Mahin, S. A. (2007a). Real-Time Error Monitoring for Hybrid Simulation. Part I: Methodology and Experimental Verification. *Journal of Structural Engineering*, 133(8):1100–1108.

Mosqueda, G., Stojadinovic, B., and Mahin, S. A. (2007b). Real-Time Error Monitoring for Hybrid Simulation. Part II: Structural Response Modification due to Errors. *Journal of Structural Engineering*, 133(8):1109–1117.

Murray, J. A. and Sasani, M. (2016). Near-collapse response of existing RC building under severe pulse-type ground motion using hybrid simulation. *Earthquake Engineering & Structural Dynamics*, 45(7):1109–1127.

Murray, J. A. and Sasani, M. (2017). Collapse Resistance of a Seven-Story Structure with Multiple Shear-Axial Column Failures Using Hybrid Simulation. *Journal of Structural Engineering*, 143(5):04017012.

Nakashima, M. (2001). Development, potential, and limitations of real-time online (pseudo-dynamic) testing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 359(1786):1851–1867.

Nakashima, M., Kaminosono, T., Ishida, M., and Ando, K. (1990). Integration techniques for substructure pseudo dynamic test. In *4th U.S. National Conference on Earthquake Engineering*, volume 2, pages 515–524.

Nakashima, M., Kato, H., and Takaoka, E. (1992). Development of real-time pseudo dynamic testing. *Earthquake Engineering & Structural Dynamics*, 21(1):79–92.

Nakashima, M. and Masaoka, N. (1999). Real-time on-line test for MDOF systems. *Earthquake Engineering & Structural Dynamics*, 28(4):393–420.

Nakata, N. (2013). Effective force testing using a robust loop shaping controller. *Earthquake Engineering & Structural Dynamics*, 42(2):261–275.

Nakata, N. and Krug, E. (2013). Multidegrees-of-freedom effective force testing: a feasibility study and robust stability assessment. *Earthquake Engineering & Structural Dynamics*, 42(13):1985–2002.

Nakata, N., Krug, E., and King, A. (2014). Experimental implementation and verification of multi-degrees-of-freedom effective force testing. *Earthquake Engineering & Structural Dynamics*, 43(3):413–428.

Nakata, N., Spencer, Jr., B. F., and Elnashai, A. S. (2007). Multi-dimensional Mixed-mode Hybrid Simulation Control and Applications. Technical Report NSEL-005, Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL.

Nakata, N., Spencer, Jr., B. F., and Elnashai, A. S. (2010). Sensitivity-Based External Calibration of Multiaxial Loading System. *Journal of Engineering Mechanics*, 136(2):189–198.

Nojavan, A., Schultz, A. E., Haselton, C., Simathathien, S., Liu, X., and Chao, S.-H. (2015). A New Data Set for Full-Scale Reinforced Concrete Columns under Collapse-Consistent Loading Protocols. *Earthquake Spectra*, 31(2):1211–1231.

Ou, G., Dyke, S. J., and Prakash, A. (2017). Real time hybrid simulation with online model updating: An analysis of accuracy. *Mechanical Systems and Signal Processing*, 84:223–240.

Ou, G., Ozdagli, A. I., Dyke, S. J., and Wu, B. (2015a). Robust integrated actuator control: experimental verification and real-time hybrid-simulation implementation. *Earthquake Engineering & Structural Dynamics*, 44(3):441–460.

Ou, G., Prakash, A., and Dyke, S. J. (2015b). Modified Runge-Kutta Integration Algorithm for Improved Stability and Accuracy in Real Time Hybrid Simulation. *Journal of Earthquake Engineering*, 19(7):1112–1139.

Paccot, F., Andreff, N., and Martinet, P. (2009). A Review on the Dynamic Control of Parallel Kinematic Machines: Theory and Experiments. *The International Journal of Robotics Research*, 28(3):395–416.

Phillips, B. M., Chae, Y., Jiang, Z., Spencer Jr., B. F., Ricles, J. M., Christenson, R. E., Dyke, S. J., and Agrawal, A. K. (2010). Real-time hybrid simulation benchmark study with a large-scale MR damper. In *5th World Conference on Structural Control and Monitoring*, pages 12–14, Tokyo, Japan.

Phillips, B. M. and Spencer, Jr., B. F. (2012). Model-Based Framework for Real-Time Dynamic Structural Performance Evaluation. Technical Report NSEL-031, Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL.

Phillips, B. M. and Spencer, Jr., B. F. (2013a). Model-Based Feedforward-Feedback Actuator Control for Real-Time Hybrid Simulation. *Journal of Structural Engineering*, 139(7):1205–1214.

Phillips, B. M. and Spencer, Jr., B. F. (2013b). Model-Based Multiactuator Control for Real-Time Hybrid Simulation. *Journal of Engineering Mechanics*, 139(2):219–228.

Reinhorn, A. M., Sivaselvan, M., Weinreber, S., and Shao, X. (2004). Real-time dynamic hybrid testing of structural systems. In *13th World Conference on Earthquake Engineering*, number 1644, Vancouver, Canada.

Reinhorn, A. M., Sivaselvan, M. V., and Liang, Z. (2005). Large scale real time dynamic hybrid testing technique – shake tables substructure testing. In Itoh, Y. and Aoki, T., editors, *The First International Conference on Advances in Experimental Structural Engineering*, Nagoya, Japan.

Rocklin, G., Crowley, J., and Vold, H. (1985). A comparison of H1, H2, and Hv frequency response functions. In *Proceedings of the 3rd international Modal Analysis*, pages 272–278, Orlando, FL.

Ryan, K. L. and Dao, N. D. (2016). Influence of Vertical Ground Shaking on Horizontal Response of Seismically Isolated Buildings with Friction Bearings. *Journal of Structural Engineering*, 142(1):04015089.

Sanchez, J., Masroor, A., Mosqueda, G., and Ryan, K. (2013). Static and Dynamic Stability of Elastomeric Bearings for Seismic Protection of Structures. *Journal of Structural Engineering*, 139(7):1149–1159.

Saouma, V., Haussmann, G., Kang, D.-H., and Ghannoum, W. (2014). Real-Time Hybrid Simulation of a Nonductile Reinforced Concrete Frame. *Journal of Structural Engineering*, 140(2):1–12.

Schellenberg, A., Kim, H. K., Takahashi, Y., Fenves, G. L., and Mahin, S. A. (2009). OpenFresco: command language manual (Version 2.6)[Computer software].

Shao, X., Mueller, A., and Mohammed, B. A. (2015). Real-Time Hybrid Simulation with Online Model Updating: Methodology and Implementation. *Journal of Engineering Mechanics*, page 04015074.

Shao, X. and Reinhorn, A. M. (2012). Development of a Controller Platform for Force-Based Real-Time Hybrid Simulation. *Journal of Earthquake Engineering*, 16(2):274–295.

Shao, X., Reinhorn, A. M., and Sivaselvan, M. V. (2011). Real-Time Hybrid Simulation Using Shake Tables and Dynamic Actuators. *Journal of Structural Engineering*, 137(7):748–760.

Shing, P. B. (2008). Integration schemes for real-time hybrid testing. In Saouma, V. and Sivaselvan, M., editors, *Hybrid Simulation: Theory, Implementation and Applications*. Taylor & Francis.

Shing, P. B., Nakashima, M., and Bursi, O. S. (1996). Application of Pseudodynamic Test Method to Structural Research. *Earthquake Spectra*, 12(1):29–56.

Shing, P.-S. B. and Mahin, S. A. (1988). Rate-of-loading effects on pseudodynamic tests. *Journal of Structural Engineering*, 114(11):2403–2420.

Shore Western (2014). Load and boundary condition box (LBCB). Retrieved from http://www.shorewestern.com.

Simeonov, V. K., Sivaselvan, M. V., and Reinhorn, A. M. (2000). Nonlinear Analysis of Structural Frame Systems by the State-Space Approach. *Computer-Aided Civil and Infrastructure Engineering*, 15(2):76–89.

Sivaselvan, M. V., Reinhorn, A. M., Shao, X., and Weinreber, S. (2008). Dynamic force control with hydraulic actuators using added compliance and displacement compensation. *Earthquake Engineering & Structural Dynamics*, 37(15):1785–1800.

Stein, G. and Athans, M. (1987). The LQG/LTR procedure for multivariable feedback control design. *IEEE Transactions on Automatic Control*, 32(2):105–114.

Stojadinovic, B., Mosqueda, G., and Mahin, S. A. (2006). Event-Driven Control System for Geographically Distributed Hybrid Simulation. *Journal of Structural Engineering*, 132(1):68–77.

Takanashi, K. and Nakashima, M. (1987). Japanese Activities on On-Line Testing. *Journal of Engineering Mechanics*, 113(7):1014–1032.

Takanashi, K. and Ohi, K. (1983). Earthquake response analysis of steel structures by rapid computer-actuator on-line system: a progress report, trial system and dynamic response of steel beams. *Bulletin of Earthquake Resistant Structure Research Center*, (16):103–109.

Takanashi, K., Udagawa, K., Seki, M., Okada, T., and Tanaka, H. (1975). Nonlinear Earthquake Response Analysis of Structures by a Computer-Actuator On-Line System. *Bulletin of Earthquake Resistant Structure Research Center*, (8):1–17.

Thayer, W. (1965). Transfer Functions for Moog Servovalves. Technical Report #103, Moog Inc. Controls Division, East Aurora, NY.

Thewalt, C. R. and Mahin, S. A. (1987). Hybrid solution techniques for generalized pseudodynamic testing. Technical Report UCB/EERC-87/09, Earthquake Engineering Research Center, University of California, Berkeley, CA.

Thewalt, C. R. and Roman, M. (1994). Performance Parameters for Pseudodynamic Tests. *Journal of Structural Engineering*, 120(9):2768–2781.

Tomizuka, M. (1987). Zero Phase Error Tracking Algorithm for Digital Control. *Journal of Dynamic Systems, Measurement, and Control*, 109(1):65–68.

Varga, A. (1998). Computation of coprime factorizations of rational matrices. *Linear Algebra and its Applications*, 271(1-3):83–115.

Wallace, M., Wagg, D., and Neild, S. (2005). An adaptive polynomial based forward prediction algorithm for multi-actuator real-time dynamic substructuring. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 461(2064):3807–3826.

Wu, B., Bao, H., Ou, J., and Tian, S. (2005). Stability and accuracy analysis of the central difference method for real-time substructure testing. *Earthquake Engineering & Structural Dynamics*, 34(7):705–718.

Wu, B., Xu, G., Wang, Q., and Williams, M. S. (2006). Operator-splitting method for real-time substructure testing. *Earthquake Engineering & Structural Dynamics*, 35(3):293–314.

Yang, T., Tung, D. P., Li, Y., Lin, J. Y., Li, K., and Guo, W. (2017). Theory and implementation of switch-based hybrid simulation technology for earthquake engineering applications. *Earthquake Engineering & Structural Dynamics*, 46(14):2603–2617.

Yang, Y. S., Tsai, K.-C., Elnashai, A. S., Kwon, O.-S., and Lin, S. L. (2009). Preliminary Study on Online Updating Hybrid Simulation. Technical Report NCREE-09-001, National Center for Research on Earthquake Engineering, Taipei, Taiwan.

Zhao, J., French, C. W., Shield, C. K., and Posbergh, T. (2003). Considerations for the development of real-time dynamic testing using servo-hydraulic actuation. *Earthquake Engineering & Structural Dynamics*, 32(11):1773–1794.

Zou, Q. and Devasia, S. (1999). Preview-Based Stable-Inversion for Output Tracking of Linear Systems. *Journal of Dynamic Systems, Measurement, and Control*, 121(4):625.

Appendix A

CALCULATION OF JACOBIAN MATRIX FOR KINEMATIC TRANSFORMATIONS

Let $\mathbf{s}_i \in \mathbb{R}^3$ be the Cartesian coordinates of the "i-th" joint of a parallel manipulator, as presented in Chapter 4. The equation for this vector is given by:

$$\mathbf{s}_i = \mathbf{p} + \mathbf{R}\mathbf{b}_i - \mathbf{a}_i, \quad (i = \{1, \ldots, n\}) \tag{A.1}$$

where $\mathbf{p} \in \mathbb{R}^3$ and $\mathbf{R} \in \mathbb{R}^{3\times3}$ are the translation vector and rotational matrix that represents the position of the body frame with respect to the global frame; $\mathbf{a}_i = \{a_{xi}, a_{yi}, a_{zi}\}^T$ are the fixed end coordinates of the "i-th"joint relative to the fixed frame, and $\mathbf{b}_i = \{b_{xi}, b_{yi}, b_{zi}\}^T$ are the free end coordinates of the "i-th" joint relative to moving frame; and n is the total number of joints connected to the moving platform. Then, given the inverse kinematic transformation (IKT) formula:

$$\begin{aligned} q_i &= q_i(u_x, u_y, u_z, \theta_x, \theta_y, \theta_z) \\ &= \|\mathbf{s}_i\| \\ &= \|\mathbf{p}(u_x, u_y, u_z) + \mathbf{R}(\theta_x, \theta_y, \theta_z)\mathbf{b}_i - \mathbf{a}_i\| \end{aligned} \tag{A.2}$$

$$\mathbf{q} = \mathbf{q}(\mathbf{u}) \tag{A.3}$$

where $\mathbf{q} = \{q_1, q_2, \ldots, q_n\}^T$ is the vector of joint strokes of the parallel manipulator, and $\mathbf{u} = \{u_x, u_y, u_z, \theta_x, \theta_y, \theta_z\}^T$ is the Cartesian coordinates of the manipulator's loading platform.

By performing a Taylor series expansion around the Cartesian coordinate $\mathbf{u}_k$ at step "k":

$$\delta\mathbf{q} \approx \mathbf{J}\delta\mathbf{u} \tag{A.4}$$

where $\delta\mathbf{q} = \mathbf{q}_{k+1} - \mathbf{q}_k$ is the joint coordinate increment, $\delta\mathbf{u} = \mathbf{u}_{k+1} - \mathbf{u}_k$ is the Cartesian coordinate increment, and $\mathbf{J} \in \mathbb{R}^{n\times6}$ is a Jacobian matrix defined in indicial notation by:

$$J_{rs} = \frac{\partial q_r}{\partial u_s}(\mathbf{u}_k), \quad \forall r = \{1, \ldots, n\}, s = \{1, \ldots, 6\} \tag{A.5}$$

or in matrix form:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial q_1}{\partial u_1}(\mathbf{u}_k) & \cdots & \frac{\partial q_1}{\partial u_6}(\mathbf{u}_k) \\ \vdots & \ddots & \vdots \\ \frac{\partial q_n}{\partial u_1}(\mathbf{u}_k) & \cdots & \frac{\partial q_n}{\partial u_6}(\mathbf{u}_k) \end{bmatrix} \tag{A.6}$$

$$s_x = \sin\theta_x \qquad c_x = \cos\theta_x \tag{A.7}$$
$$s_y = \sin\theta_y \qquad c_y = \cos\theta_y \tag{A.8}$$
$$s_z = \sin\theta_z \qquad c_z = \cos\theta_z \tag{A.9}$$

Then, the analytic expressions to obtain the i-th row of the Jacobian matrix are obtained as follows:

$$\frac{\partial q_i}{\partial u_x} = \frac{1}{2q_i}\left\{2u_x - 2a_{xi} - 2b_{yi}(c_x s_z - c_z s_x s_y) + 2b_{zi}(s_x s_z + c_x c_z s_y) + 2b_{xi}c_y c_z\right\} \tag{A.10}$$
$$\frac{\partial q_i}{\partial u_y} = \frac{1}{2q_i}\left\{2u_y - 2a_{yi} + 2b_{yi}(c_x c_z + s_x s_y s_z) - 2b_{zi}(c_z s_x - c_x s_y s_z) + 2b_{xi}c_y s_z\right\} \tag{A.11}$$
$$\frac{\partial q_i}{\partial u_z} = \frac{1}{2q_i}\left\{2u_z - 2a_{zi} - 2b_{xi}s_y + 2b_{zi}c_x c_y + 2b_{yi}c_y s_x\right\} \tag{A.12}$$

$$\begin{aligned}\frac{\partial q_i}{\partial \theta_x} = \frac{1}{2q_i}\{&2\left[b_{yi}(s_x s_z + c_x c_z s_y) + b_{zi}(c_x s_z - c_z s_x s_y)\right] \\ &\times \left[u_x - a_{xi} - b_{yi}(c_x s_z - c_z s_x s_y) + b_{zi}(s_x s_z + c_x c_z s_y) + b_{xi}c_y c_z\right] \\ &- 2\left[b_{yi}(c_z s_x - c_x s_y s_z) + b_{zi}(c_x c_z + s_x s_y s_z)\right] \\ &\times \left[u_y - a_{yi} + b_{yi}(c_x c_z + s_x s_y s_z) - b_{zi}(c_z s_x - c_x s_y s_z) + b_{xi}c_y s_z\right] \\ &+2(b_{yi}c_x c_y - b_{zi}c_y s_x)(u_z - a_{zi} - b_{xi}s_y + b_{zi}c_x c_y + b_{yi}c_y s_x)\}\end{aligned} \tag{A.13}$$

$$\begin{aligned}\frac{\partial q_i}{\partial \theta_y} = \frac{1}{2q_i}\{&2(b_{zi}c_x c_y c_z - b_{xi}c_z s_y + b_{yi}c_y c_z s_x) \\ &\times \left[u_x - a_{xi} - b_{yi}(c_x s_z - c_z s_x s_y) + b_{zi}(s_x s_z + c_x c_z s_y) + b_{xi}c_y c_z\right] \\ &- 2(b_{xi}c_y + b_{zi}c_x s_y + b_{yi}s_x s_y)(u_z - a_{zi} - b_{xi}s_y + b_{zi}c_x c_y + b_{yi}c_y s_x) \\ &+ 2(b_{zi}c_x c_y s_z - b_{xi}s_y s_z + b_{yi}c_y s_x s_z) \\ &\times \left[u_y - a_{yi} + b_{yi}(c_x c_z + s_x s_y s_z) - b_{zi}(c_z s_x - c_x s_y s_z) + b_{xi}c_y s_z\right]\}\end{aligned} \tag{A.14}$$

$$\begin{aligned}\frac{\partial q_i}{\partial \theta_z} = \frac{1}{2q_i}\{&2\left[b_{zi}(s_x s_z + c_x c_z s_y) - b_{yi}(c_x s_z - c_z s_x s_y) + b_{xi}c_y c_z\right] \\ &\times \left[u_y - a_{yi} + b_{yi}(c_x c_z + s_x s_y s_z) - b_{zi}(c_z s_x - c_x s_y s_z) + b_{xi}c_y s_z\right] \\ &- 2\left[b_{yi}(c_x c_z + s_x s_y s_z) - b_{zi}(c_z s_x - c_x s_y s_z) + b_{xi}c_y s_z)\right] \\ &\times \left[u_x - a_{xi} - b_{yi}(c_x s_z - c_z s_x s_y) + b_{zi}(s_x s_z + c_x c_z s_y) + b_{xi}c_y c_z\right]\}\end{aligned} \tag{A.15}$$

Appendix B

DYNAMIC MODEL OF MULTI-AXIAL LOADING ASSEMBLY

B.1 Transfer function

The multi-input, multi-output transfer function can be represented as a rational function matrix:

$$\mathbf{G}_{yu}(s) = \frac{1}{d(s)}\mathbf{N}(s) \tag{B.1}$$

where $d(s) \in \mathbb{R}(s)$ is a scalar monic polynomial, equal to the least common multiple of the denominators from each entry of the matrix; and $\mathbf{N}(s) \in \mathbb{R}^{p\times m}(s)$ is a matrix polynomial.

For the case of the small-scale LBCB loading assembly with a steel column specimen attached to its loading platform, the transfer function in Cartesian coordinates was reduced to remove the weakly observable output u_z. Therefore, the transfer matrix is 5×5 square matrix, with $p = m = 5$. The structure of the identified transfer system is the following:

$$\mathbf{G}_{yu}(s) = \frac{1}{d(s)} \begin{bmatrix} n_{11}(s) & n_{12}(s) & n_{13}(s) & n_{14}(s) & n_{15}(s) \\ n_{21}(s) & n_{22}(s) & n_{23}(s) & n_{24}(s) & n_{25}(s) \\ n_{31}(s) & n_{32}(s) & n_{33}(s) & n_{34}(s) & n_{35}(s) \\ n_{41}(s) & n_{42}(s) & n_{43}(s) & n_{44}(s) & n_{45}(s) \\ n_{51}(s) & n_{52}(s) & n_{53}(s) & n_{54}(s) & n_{55}(s) \end{bmatrix} \tag{B.2}$$

$$\mathbf{y} = \left\{u_x^{\text{Meas}}, u_y^{\text{Meas}}, \theta_x^{\text{Meas}}, \theta_y^{\text{Meas}}, \theta_z^{\text{Meas}}\right\}^T \tag{B.3}$$

$$\mathbf{u} = \left\{u_x^{\text{Cmd}}, u_y^{\text{Cmd}}, \theta_x^{\text{Cmd}}, \theta_y^{\text{Cmd}}, \theta_z^{\text{Cmd}}\right\}^T \tag{B.4}$$

The polynomial coefficients for both denominator and numerator array are presented as source code for Matlab as follows.

Denominator

$$d(s) = \sum_{k=1}^{N} d_k s^{N-k}, \quad N = 19 \tag{B.5}$$

Listing B.1: Denominator polynomial

```
den = [1, 2.306e+06, 9.087e+08, 3.533e+11, 7.795e+13, 1.488e+16, 2.139e+18, 2.619e+20,
    2.575e+22, 2.133e+24,  1.42e+26, 7.689e+27,  3.15e+29, 9.447e+30, 1.658e+32,  3.18e+33,
     1.45e+34,  2.82e+34,  9.33e+34];
```

Numerator

$$n_{ij}(s) = \sum_{k=1}^{N} n_{ij,k} s^{N-k}, \quad N = 19 \tag{B.6}$$

Listing B.2: Numerator polynomial

```
num{1,1} = [0,          0,          0, 3.392e+09,   1.2e+12, 9.441e+14, 1.759e+17, 3.665e+19,
     4.628e+21,  5.22e+23, 4.422e+25, 3.013e+27, 1.549e+29,  6.11e+30, 1.003e+32, 2.561e+33,
     1.241e+34, 2.322e+34, 8.299e+34];

num{1,2} = [0,          0,          0, 5.664e+08,-3.595e+10, 2.382e+13,-1.045e+15, 3.632e
    +17,-1.098e+19, 2.351e+21,-5.763e+22, 5.173e+24, -1.62e+26,-2.664e+27,-1.458e+29,-4.299e
    +30, -2.34e+31, 5.298e+31,-1.633e+32];

num{1,3} = [0,          0,          0, 7.662e+09, 2.319e+11, 3.139e+14, 9.791e+15, 4.982e+18,
     1.538e+20, 3.841e+22, 1.083e+24, 1.447e+26, 3.225e+27, 2.272e+29, 2.806e+30, 8.369e+31,
     5.826e+32,  1.15e+33, 4.269e+33]

num{1,4} = [0,          0,          0,-7.026e+09, 1.181e+12,-5.747e+14,-1.845e+16,-1.848e
    +19,-1.338e+21,-2.555e+23,-1.622e+25,-1.474e+27, -6.34e+28,-2.981e+30,-4.144e+31,-1.237e
    +33, -5.05e+33,-1.172e+34,-3.191e+34]

num{1,5} = [0,          0,          0, 3.487e+09,-1.447e+12,-1.824e+14,-7.639e+16,-8.176e+18,
     -1.25e+21,-9.389e+22,-7.537e+24, -3.59e+26,-1.462e+28,-2.004e+29,-6.455e+30,-2.088e
    +31,-4.185e+32,-5.219e+31,-2.146e+33]

num{2,1} = [0,          0,          0, 1.065e+09, 6.606e+10, 4.309e+13, 2.168e+15, 6.034e+17,
     2.699e+19, 3.393e+21, 1.641e+23, 6.743e+24, 4.722e+26, 3.703e+27, 4.057e+29, 2.622e+30,
     1.024e+32,-2.673e+31, 2.481e+32]

num{2,2} = [0,          0,          0, 2.371e+09, 4.644e+11, 1.133e+15, 1.692e+17, 4.317e+19,
     4.726e+21, 5.869e+23, 4.476e+25, 3.181e+27, 1.523e+29,  5.82e+30, 9.553e+31, 2.347e+33,
     1.099e+34, 2.114e+34, 7.232e+34]

num{2,3} = [0,          0,          0, 1.914e+10, 1.404e+12, 2.615e+15,  3.23e+17, 8.597e+19,
     8.693e+21, 1.108e+24, 8.104e+25, 5.816e+27, 2.728e+29, 1.027e+31, 1.696e+32, 4.093e+33,
     1.896e+34, 3.692e+34, 1.242e+35]

num{2,4} = [0,          0,          0, 3.011e+09, 6.928e+11, 9.367e+14, 1.251e+17, 3.295e+19,
      3.24e+21, 4.212e+23, 2.939e+25,  2.14e+27, 9.655e+28, 3.536e+30, 5.939e+31, 1.374e+33,
     6.734e+33,  1.19e+34, 4.534e+34]

num{2,5} = [0,          0,          0,-4.953e+09,-5.596e+11,-2.697e+14, -2.73e+16,-5.282e
    +18,-4.409e+20,-4.529e+22,-2.792e+24, -1.58e+26,-6.245e+27,-1.338e+29,-2.893e+30,-3.027e
    +31,-1.481e+32, 6.706e+31,-2.014e+33]

```

```
num{3,1} = [0,          0,          0, 2.264e+08,-1.221e+10, 9.514e+12,-6.241e+14, 1.438e
    +17,-1.091e+19, 9.604e+20,-7.972e+22, 2.936e+24,-2.314e+26, 4.133e+27,-1.751e+29, 1.262e
    +30,-2.944e+31, 7.465e+30,-2.266e+32]

num{3,2} = [0,          0,          0,-1.283e+09, 9.361e+10,-3.229e+13, 6.339e+15,-2.484e+16,
     1.246e+20, 4.538e+21, 9.394e+23, 3.302e+25, 2.588e+27, 5.608e+28, 1.358e+30, 2.142e+31,
     7.882e+31, 2.282e+32, 4.026e+32]

num{3,3} = [0,          0,          0, 1.027e+10, 2.501e+12, 1.576e+15, 2.495e+17, 5.246e+19,
     5.904e+21, 6.771e+23, 5.253e+25,  3.58e+27, 1.734e+29, 6.475e+30,  1.08e+32, 2.602e+33,
     1.234e+34, 2.342e+34,  8.17e+34]

num{3,4} = [0,          0,          0, 5.111e+09, 9.436e+11, 4.044e+14, 6.553e+16, 1.171e+19,
     1.421e+21, 1.481e+23, 1.252e+25, 7.963e+26, 4.215e+28, 1.502e+30, 2.646e+31, 6.124e+32,
     3.044e+33, 5.455e+33, 2.036e+34]

num{3,5} = [0,          0,          0,-1.507e+09, 1.224e+11,-6.636e+13, 4.694e+15, -1.03e+18,
     6.625e+19,-6.645e+21, 4.169e+23, -1.54e+25, 1.086e+27,-1.954e+27, 7.339e+29,  2.97e+30,
     1.551e+32,-2.146e+31, 1.071e+33]

num{4,1} = [0,          0,          0, 6.639e+08, -5.03e+10, 1.442e+13,-3.911e+15, -1.13e
    +17,-8.513e+19, -4.43e+21,-7.211e+23,-2.947e+25,-2.293e+27,-5.153e+28,-1.401e+30,-1.981e
    +31,-1.542e+32,  -1.9e+32,-1.068e+33]

num{4,2} = [0,          0,          0,-1.189e+08,-4.065e+10,-2.563e+13,-4.232e+15,-9.091e
    +17,-1.018e+20,-1.212e+22,-9.182e+23,-6.611e+25,-3.066e+27,-1.256e+29,-1.911e+30,-5.098e
    +31,-2.198e+32,-4.551e+32,-1.341e+33]

num{4,3} = [0,          0,          0, 1.298e+09, 2.511e+10, 4.091e+13,-9.303e+14,  4.58e
    +17,-3.022e+19,  2.42e+21, -1.83e+23, 7.984e+24,-1.251e+26, 1.946e+28, 9.148e+28, 8.769e
    +30, 6.942e+31,-1.141e+31, 5.291e+32]

num{4,4} = [0,          0,          0, 1.009e+10, 1.648e+12, 1.586e+15, 2.336e+17, 5.429e+19,
     5.965e+21, 7.174e+23, 5.499e+25, 3.869e+27, 1.856e+29, 7.147e+30, 1.171e+32, 2.899e+33,
     1.379e+34, 2.604e+34, 9.119e+34]

num{4,5} = [0,          0,          0,-5.928e+09, 4.881e+11,-2.006e+14, 2.627e+16,-1.782e+18,
     4.952e+20, 3.779e+21, 3.985e+24, 1.005e+26, 1.258e+28, 2.996e+29, 7.756e+30, 1.336e+32,
     8.816e+32, 1.292e+33, 5.735e+33]

num{5,1} = [0,          0,          0, 3.919e+08, 1.088e+11, -4.96e+13,-1.647e+15,-2.009e
    +18,-9.198e+19,-2.419e+22,-8.206e+23,-1.023e+26,-2.153e+27,-9.005e+28,-1.112e+30,-2.205e
    +31,-1.072e+32,-1.981e+32,-7.035e+32]

num{5,2} = [0,          0,          0, 7.021e+07,-7.732e+10,-2.108e+13,-5.778e+15,-9.215e
    +17,-1.253e+20,-1.301e+22,-1.072e+24,-7.091e+25,-3.466e+27,-1.267e+29,-2.128e+30, -5.06e
    +31,-2.296e+32,-4.227e+32,-1.634e+33]

num{5,3} = [0,          0,          0,  3.95e+09, 4.259e+11, 1.743e+14,  2.18e+16, 3.328e+18,
     4.076e+20, 3.447e+22, 3.352e+24, 1.878e+26, 1.101e+28, 4.413e+29, 7.117e+30, 1.971e+32,
     6.836e+32, 1.847e+33, 4.429e+33]

num{5,4} = [0,          0,       0,-2.887e+09,-1.067e+11,-1.539e+14,-1.175e+16,-3.459e
    +18,-3.081e+20,-3.848e+22,-3.077e+24,-2.001e+26,-1.135e+28,-3.984e+29,-7.457e+30,-1.647e
    +32,-9.806e+32,-1.558e+33,-7.336e+33]

num{5,5} = [0,          0,          0, 5.644e+08, 6.767e+11, 7.686e+14, 1.516e+17, 3.279e+19,
     4.226e+21, 4.842e+23, 4.137e+25, 2.851e+27,  1.47e+29, 5.871e+30, 9.594e+31, 2.469e+33,
     1.209e+34, 2.234e+34, 8.091e+34]
```

B.2 State-space model

The reduced-order state-space realization is obtained using the Control Systems Toolbox in Matlab, and stored as a continuous-time state-space system on variable `sysred`. Then, the state-space matrices in compressed sparse column format are presented as follows.

Listing B.3: Reduced-order state-space model

```
>> sparse(sysred.A)

ans =

   (1,1)       -0.3500
   (2,1)     -117.2957
   (1,2)      117.2957
   (2,2)       -0.3500
   (3,3)       -0.0807
   (4,3)      -87.1080
   (3,4)       87.1080
   (4,4)       -0.0807
   (5,5)      -55.5823
   (6,5)      -92.0661
   (5,6)       92.0661
   (6,6)      -55.5823
   (7,7)      -55.1178
   (8,7)      -86.2675
   (7,8)       86.2675
   (8,8)      -55.1178
   (9,9)      -62.5264
  (10,9)      -78.5177
   (9,10)      78.5177
  (10,10)     -62.5264
  (11,11)     -49.7527
  (12,11)     -67.4119
  (11,12)      67.4119
  (12,12)     -49.7527
  (13,13)     -18.2806
  (14,13)     -32.2685
  (13,14)      32.2685
  (14,14)     -18.2806
  (15,15)      -6.6601
  (16,15)      -4.9226
  (15,16)       4.9226
  (16,16)      -6.6601
  (17,17)      -0.5251
  (18,17)      -2.5055
  (17,18)       2.5055
  (18,18)      -0.5251

>> sparse(sysred.B)

ans =

   (1,1)        0.0429
   (2,1)       -0.0163
   (3,1)        0.0091
   (4,1)       -0.0162
   (5,1)       -0.0328
   (6,1)       -0.4234
   (7,1)        0.4438
   (8,1)        0.0266
   (9,1)        5.2815
  (10,1)        1.0500
  (11,1)       -1.6147
```

```
(12,1)       6.8055
(13,1)       0.0315
(14,1)      -0.6698
(15,1)      -0.0940
(16,1)      -1.3703
(17,1)       0.0120
(18,1)       0.0785
 (1,2)       0.0755
 (2,2)      -0.2140
 (3,2)       0.0165
 (4,2)       0.0347
 (5,2)      -7.5481
 (6,2)      -0.6386
 (7,2)       0.8663
 (8,2)       0.4348
 (9,2)      -0.8748
(10,2)      -0.3693
(11,2)      -0.3950
(12,2)      -0.3234
(13,2)       2.0663
(14,2)      -3.0582
(15,2)       1.0189
(16,2)      -2.0266
(17,2)      -0.1243
(18,2)       0.0785
 (1,3)      -0.2388
 (2,3)       0.2934
 (3,3)       0.0961
 (4,3)      -0.1101
 (5,3)     -14.9745
 (6,3)      -1.4931
 (7,3)       1.6836
 (8,3)       2.5473
 (9,3)      -2.6731
(10,3)      -0.6487
(11,3)      -0.6011
(12,3)       2.4353
(13,3)      -1.5822
(14,3)       3.4167
(15,3)      -1.1774
(16,3)       2.7194
(17,3)       0.2842
(18,3)      -0.2647
 (1,4)      -0.1630
 (2,4)      -0.1157
 (3,4)       0.0573
 (4,4)       0.0088
 (5,4)      -5.0050
 (6,4)      -3.6994
 (7,4)      -0.9270
 (8,4)      15.6338
 (9,4)       0.1778
(10,4)      -1.3451
(11,4)      -0.0399
(12,4)      -1.6046
(13,4)      -0.9270
(14,4)       1.3462
(15,4)      -0.5745
(16,4)      -0.1499
(17,4)      -0.0136
(18,4)       0.2444
 (1,5)      -0.1568
 (2,5)       0.0838
 (3,5)      -0.2018
 (4,5)      -0.1776
 (5,5)       1.6704
 (6,5)       0.3700
```

```
   (7,5)       -1.8294
   (8,5)       -1.2419
   (9,5)      -12.5135
  (10,5)        0.9031
  (11,5)       -1.4442
  (12,5)       14.6551
  (13,5)        0.6250
  (14,5)        0.1654
  (15,5)       -0.1034
  (16,5)        0.3437
  (17,5)        0.0710
  (18,5)       -0.0873

>> sparse(sysred.C)

ans =

   (1,1)        0.2524
   (2,1)       -0.3696
   (3,1)        0.2814
   (4,1)        0.0180
   (5,1)        0.0419
   (1,2)       -0.0606
   (2,2)       -0.0401
   (3,2)       -0.0695
   (4,2)        0.0507
   (5,2)       -0.0132
   (1,3)        0.0589
   (2,3)        0.0604
   (3,3)       -0.0646
   (4,3)        0.0006
   (5,3)       -0.0234
   (1,4)       -0.2882
   (2,4)       -0.0016
   (3,4)       -0.0296
   (4,4)        0.0752
   (5,4)       -0.0011
   (1,5)       -0.2000
   (2,5)        3.0400
   (3,5)        1.1083
   (4,5)        1.0467
   (5,5)       -0.6071
   (1,6)        1.4241
   (2,6)       17.2503
   (3,6)        6.8390
   (4,6)       -1.8725
   (5,6)       -0.9999
   (1,7)       -3.6309
   (2,7)       -0.4326
   (3,7)       -0.6729
   (4,7)        9.5545
   (5,7)        0.4230
   (1,8)        0.5549
   (2,8)        4.3598
   (3,8)        1.2792
   (4,8)       -1.3315
   (5,8)        0.1299
   (1,9)       -0.7075
   (2,9)        1.1600
   (3,9)        0.6235
   (4,9)       -1.0348
   (5,9)        1.3893
   (1,10)     -13.3875
   (2,10)       1.3091
   (3,10)       0.0826
   (4,10)       1.0171
   (5,10)       6.2863
```

```
   (1,11)       8.7176
   (2,11)       0.1492
   (3,11)       0.5312
   (4,11)       0.3770
   (5,11)       3.8431
   (1,12)       0.9705
   (2,12)       0.3080
   (3,12)       0.3052
   (4,12)       0.1032
   (5,12)      -0.7928
   (1,13)       0.4336
   (2,13)       0.1066
   (3,13)      -0.8062
   (4,13)       0.2695
   (5,13)       0.1862
   (1,14)      -0.2926
   (2,14)      -1.2558
   (3,14)       2.4092
   (4,14)      -0.2131
   (5,14)      -0.1176
   (1,15)       0.0356
   (2,15)       0.2591
   (3,15)      -0.7688
   (4,15)       0.0219
   (5,15)       0.1115
   (1,16)      -0.2032
   (2,16)      -0.1929
   (3,16)       0.5802
   (4,16)      -0.1712
   (5,16)       0.0000
   (1,17)       0.1787
   (2,17)       0.1455
   (3,17)      -0.0799
   (4,17)       0.0573
   (5,17)      -0.0892
   (1,18)      -0.1891
   (2,18)       0.0284
   (3,18)      -0.2225
   (4,18)      -0.0471
   (5,18)       0.0427

>> sparse(sysred.D)

ans =

   All zero sparse: 5x5
```

Appendix C

DYNAMIC MODELS FOR MIMO FEEDFORWARD-FEEDBACK COMPENSATION

C.1 Feedforward compensator

For the case of the small-scale LBCB loading assembly with a steel column specimen attached to its loading platform, the model of the MIMO inverse system was identified using an improper transfer function with three zeroes and no poles for each component of the real polynomial matrix.

$$\mathbf{K}_{\text{ff}}(s) = \begin{bmatrix} k_{11}(s) & k_{12}(s) & k_{13}(s) & k_{14}(s) & k_{15}(s) & k_{16}(s) \\ k_{21}(s) & k_{22}(s) & k_{23}(s) & k_{24}(s) & k_{25}(s) & k_{16}(s) \\ k_{31}(s) & k_{32}(s) & k_{33}(s) & k_{34}(s) & k_{35}(s) & k_{36}(s) \\ k_{41}(s) & k_{42}(s) & k_{43}(s) & k_{44}(s) & k_{45}(s) & k_{46}(s) \\ k_{51}(s) & k_{52}(s) & k_{53}(s) & k_{54}(s) & k_{55}(s) & k_{56}(s) \\ k_{61}(s) & k_{62}(s) & k_{63}(s) & k_{64}(s) & k_{65}(s) & k_{66}(s) \end{bmatrix} \tag{C.1}$$

where $k_{ij}(s)$ is a third degree polynomial.

$$k_{ij}(s) = \sum_{k=1}^{N} k_{ij,k} s^{N-k}, \quad N = 4 \tag{C.2}$$

The polynomial coefficients for the polynomial array are presented as Matlab source code as follows.

Listing C.1: Feedforward polynomial coefficients

```
k{1,1} = [5.713e-07, 0.0001742,   0.01855,     1.081];
k{1,2} = [2.414e-08, 5.277e-06, 0.0008014,   0.05637];
k{1,3} = [0];
k{1,4} = [-2.42e-08,-4.651e-06, -0.001943,    -0.1301];
k{1,5} = [7.407e-08, 6.664e-05,  0.007643,     0.5448];
k{1,6} = [3.146e-07, 4.805e-05,  0.001507,    -0.1203];
k{2,1} = [9.461e-09, 8.369e-06, 0.0005533,    0.03675];
k{2,2} = [5.264e-07, 0.0001423,   0.01739,      1.202];
k{2,3} = [0];
k{2,4} = [-6.2e-07,-0.0002351,  -0.02874,     -1.891];
k{2,5} = [-2.479e-07,-5.979e-05,  -0.00513,    -0.2445];
k{2,6} = [1.03e-07, 2.988e-05,  0.002762,    0.03685];
k{3,1} = [-1.18e-07, -2.74e-05, -0.003177,    -0.1744];
k{3,2} = [9.629e-08, 2.725e-05,  0.003301,     0.2295];
```

```
k{3,3} = [0];
k{3,4} = [-1.301e-07,-4.446e-05, -0.005431,   -0.3544];
k{3,5} = [4.374e-08, 1.359e-05,  0.001731,    0.1483];
k{3,6} = [-4.342e-08, -5.23e-06,-8.864e-05,  0.002719];
k{4,1} = [-2.935e-09, 2.235e-07,-3.067e-05,  -0.00215];
k{4,2} = [-1.638e-08,-3.439e-06,-0.0005143,  -0.01886];
k{4,3} = [0];
k{4,4} = [3.504e-07, 0.0001437,   0.01609,     1.178];
k{4,5} = [-9.123e-08,-1.988e-05, -0.003222,   -0.2452];
k{4,6} = [1.913e-08,-1.202e-07, 9.746e-05,  0.005654];
k{5,1} = [3.878e-08,     1e-05,  0.001188,   0.06405];
k{5,2} = [-2.042e-08,-5.575e-06,-0.0007046,  -0.04647];
k{5,3} = [0];
k{5,4} = [1.368e-08,  1.33e-05,  0.001143,   0.06414];
k{5,5} = [4.084e-07, 0.0001239,   0.01409,    0.9799];
k{5,6} = [-2.313e-08,-8.685e-06,  -0.00124,  -0.09044];
k{6,1} = [5.108e-08, 1.248e-05, 0.0007437,  0.009343];
k{6,2} = [2.139e-08, 4.139e-06, 0.0006322,   0.02771];
k{6,3} = [0];
k{6,4} = [-1.599e-08, -1.19e-05, -0.001737,   -0.1559];
k{6,5} = [3.091e-08, 1.146e-05,   0.00168,     0.112];
k{6,6} = [6.766e-07,  0.000186,   0.01999,     1.131];
```

C.2 Feedback regulator

The LQG controller was developed using the Control Systems Toolbox in Matlab, and stored as a continuous-time state-space system on variable `Kfb`. Then, the state-space matrices in compressed sparse column format are presented as follows.

Listing C.2: LQG state-space model

```
>> sparse(Kfb.A)

ans =

   (1,1)     -280.3596
   (2,2)     -194.9325
   (3,3)     -168.4553
   (4,3)     -176.7843
   (3,4)      176.7843
   (4,4)     -168.4553
   (5,5)     -195.1131
   (6,5)     -135.9876
   (5,6)      135.9876
   (6,6)     -195.1131
   (7,7)     -179.5517
   (8,7)     -134.2377
   (7,8)      134.2377
   (8,8)     -179.5517
   (9,9)       -0.6983
  (10,10)     -52.3513
  (11,10)    -216.8903
  (10,11)     216.8903
  (11,11)     -52.3513
  (12,12)     -46.4842
  (13,12)    -185.6338
  (12,13)     185.6338
  (13,13)     -46.4842
  (14,14)     -43.6515
  (15,14)    -166.3473
  (14,15)     166.3473
  (15,15)     -43.6515
  (16,16)     -34.5484
  (17,16)    -150.4171
  (16,17)     150.4171
  (17,17)     -34.5484
  (18,18)     -39.8315
  (19,18)    -118.3478
  (18,19)     118.3478
  (19,19)     -39.8315
  (20,20)       -0.5004
  (21,20)    -117.3994
  (20,21)     117.3994
  (21,21)       -0.5004
  (22,22)       -0.1163
  (23,22)     -87.1010
  (22,23)      87.1010
  (23,23)       -0.1163

>> sparse(Kfb.B)

ans =

   (1,1)        1.7228
   (2,1)       -2.6176
   (3,1)        0.2538
   (4,1)       -0.1520
```

```
   (5,1)       7.6212
   (6,1)     -17.7861
   (7,1)      -1.9802
   (8,1)      13.2886
   (9,1)       0.0001
  (10,1)       0.3070
  (11,1)      -0.1425
  (12,1)      -3.8670
  (13,1)      -3.7970
  (14,1)       0.1212
  (15,1)       4.4950
  (16,1)       2.3746
  (17,1)       2.9223
  (18,1)      -0.0227
  (19,1)      -0.0164
  (20,1)       0.0338
  (21,1)       0.0159
  (22,1)       0.0191
  (23,1)      -0.0049
   (1,2)      -8.9674
   (2,2)      -2.2550
   (3,2)      -3.0488
   (4,2)     -11.9181
   (5,2)      -0.5773
   (6,2)      -4.6952
   (7,2)       7.9723
   (8,2)       0.6262
   (9,2)       0.0534
  (10,2)      -3.9720
  (11,2)       3.2917
  (12,2)       0.0287
  (13,2)       0.3620
  (14,2)       1.0310
  (15,2)       1.7814
  (16,2)       0.1774
  (17,2)       0.7755
  (18,2)       1.4804
  (19,2)      -0.6450
  (20,2)      -0.0618
  (21,2)       0.0675
  (22,2)      -0.0165
  (23,2)       0.0098
   (1,3)     -29.0741
   (2,3)     -12.4093
   (3,3)     -12.7978
   (4,3)     -41.3267
   (5,3)      -4.9272
   (6,3)     -14.7017
   (7,3)      29.8664
   (8,3)      -2.3994
   (9,3)      -0.0251
  (10,3)     -12.4376
  (11,3)      12.0530
  (12,3)       0.0721
  (13,3)       0.7966
  (14,3)       3.0662
  (15,3)       5.7679
  (16,3)       1.6137
  (17,3)       2.3131
  (18,3)       1.6569
  (19,3)      -2.5226
  (20,3)      -0.0796
  (21,3)       0.1206
  (22,3)       0.0067
  (23,3)       0.0012
   (1,4)       5.1982
   (2,4)       0.7528
```

```
   (3,4)       -0.1294
   (4,4)        6.6094
   (5,4)       -9.9192
   (6,4)       24.6526
   (7,4)       -2.0587
   (8,4)      -12.2104
   (9,4)       -0.0341
  (10,4)        2.2021
  (11,4)       -1.5543
  (12,4)        4.3461
  (13,4)        4.3751
  (14,4)       -2.9195
  (15,4)       -4.4708
  (16,4)       -2.1966
  (17,4)       -2.3245
  (18,4)       -1.5857
  (19,4)        0.7519
  (20,4)        0.0174
  (21,4)       -0.0897
  (22,4)        0.0031
  (23,4)       -0.0020
   (1,5)        5.4584
   (2,5)        1.8252
   (3,5)        2.6265
   (4,5)        7.4102
   (5,5)      -11.4650
   (6,5)       20.8479
   (7,5)        0.5891
   (8,5)      -18.4145
   (9,5)       -0.0113
  (10,5)        2.1764
  (11,5)       -2.1999
  (12,5)        3.6313
  (13,5)        4.3101
  (14,5)       -0.8198
  (15,5)       -7.7957
  (16,5)       -1.5230
  (17,5)       -4.6326
  (18,5)       -0.9855
  (19,5)        0.8281
  (20,5)       -0.0266
  (21,5)       -0.0852
  (22,5)       -0.0488
  (23,5)       -0.0203

>> sparse(Kfb.C)

ans =

   (1,1)        2.1548
   (2,1)       -3.6499
   (3,1)      -33.4944
   (4,1)      -15.8750
   (5,1)        1.8363
   (1,2)       -4.0682
   (2,2)        0.9599
   (3,2)      -17.0821
   (4,2)       -7.3672
   (5,2)        4.5828
   (1,3)       -0.4066
   (2,3)        1.4331
   (3,3)       29.4731
   (4,3)        9.0964
   (5,3)        2.1621
   (1,4)       -0.0946
   (2,4)       -1.7552
   (3,4)      -18.4109
```

192	(4,4)	-11.8466
193	(5,4)	3.5758
194	(1,5)	3.1778
195	(2,5)	0.3911
196	(3,5)	2.4791
197	(4,5)	-13.2796
198	(5,5)	-3.6041
199	(1,6)	-1.9224
200	(2,6)	-0.2011
201	(3,6)	3.4016
202	(4,6)	10.2908
203	(5,6)	3.8140
204	(1,7)	0.3927
205	(2,7)	0.2612
206	(3,7)	1.2926
207	(4,7)	-5.2556
208	(5,7)	6.0809
209	(1,8)	0.2001
210	(2,8)	0.1077
211	(3,8)	4.6423
212	(4,8)	11.0188
213	(5,8)	-3.4499
214	(1,9)	0.2199
215	(2,9)	0.1224
216	(3,9)	-0.0026
217	(4,9)	0.0230
218	(5,9)	-0.1086
219	(1,10)	-0.1635
220	(2,10)	1.1893
221	(3,10)	22.6593
222	(4,10)	5.6480
223	(5,10)	0.5704
224	(1,11)	0.3744
225	(2,11)	-1.6226
226	(3,11)	-26.2111
227	(4,11)	-8.8953
228	(5,11)	0.4343
229	(1,12)	2.1889
230	(2,12)	0.1261
231	(3,12)	4.6263
232	(4,12)	-13.4021
233	(5,12)	-8.4024
234	(1,13)	0.0401
235	(2,13)	0.1939
236	(3,13)	3.4274
237	(4,13)	-6.9055
238	(5,13)	4.7504
239	(1,14)	-0.2595
240	(2,14)	-0.3943
241	(3,14)	-0.9839
242	(4,14)	1.0123
243	(5,14)	5.7159
244	(1,15)	-1.7512
245	(2,15)	0.0540
246	(3,15)	2.3901
247	(4,15)	-5.8050
248	(5,15)	1.1850
249	(1,16)	-3.3123
250	(2,16)	0.5085
251	(3,16)	0.6141
252	(4,16)	-2.5589
253	(5,16)	-2.2286
254	(1,17)	3.4734
255	(2,17)	-0.1579
256	(3,17)	-0.1828
257	(4,17)	0.6625
258	(5,17)	6.3050

```
   (1,18)      -0.3796
   (2,18)       4.0502
   (3,18)      -3.3286
   (4,18)      -0.4247
   (5,18)      -0.1929
   (1,19)      -0.3616
   (2,19)       5.0541
   (3,19)      -1.9681
   (4,19)       0.1930
   (5,19)      -0.2526
   (1,20)      -0.2680
   (2,20)       0.5518
   (3,20)      -0.1618
   (4,20)       0.0117
   (5,20)      -0.0756
   (1,21)      -0.0837
   (2,21)      -0.4271
   (3,21)       0.2940
   (4,21)      -0.1415
   (5,21)      -0.1074
   (1,22)       0.1591
   (2,22)       0.0922
   (3,22)      -0.0602
   (4,22)      -0.0073
   (5,22)       0.0415
   (1,23)      -0.0313
   (2,23)       0.1135
   (3,23)      -0.0588
   (4,23)       0.0061
   (5,23)      -0.0315

>> sparse(Kfb.D)

ans =

   All zero sparse: 5x5
```

List of Recent NSEL Reports

No.	*Authors*	*Title*	*Date*
033	Linderman, L.E., Spencer, B.F.	Smart Wireless Control of Civil Structures	January 2014
034	Denavit, M.D. and Hajjar, J.F.	Characterization of Behavior of Steel-Concrete Composite Members and Frames with Applications for Design	July 2014
035	Jang, S. and Spencer, B.F.	Structural Health Monitoring for Bridge Structures using Wireless Smart Sensors	May 2015
036	Jo, H. and Spencer, B.F.	Multi-scale Structural Health Monitoring using Wireless Smart Sensors	May 2015
037	Li, J. and Spencer, B.F.	Monitoring, Modeling, and Hybrid Simulation: An Integrated Bayesian-based Approach to High-fidelity Fragility Analysis	May 2015
038	Sim, S-H. and Spencer, B.F.	Decentralized Identification and Multimetric Monitoring of Civil Infrastructure using Smart Sensors	June 2015
039	Giles, R.K. and Spencer, B.F.	Development of a Long-term, Multimetric Structural Health Monitoring System for a Historic Steel Truss Swing Bridge	June 2015
040	Spencer, B.F., Moreu, F., and Kim, R.E.	Campaign Monitoring of Railroad Bridges in High-Speed Rail Shared Corridors using Wireless Smart Sensors	June 2015
041	Moreu, F. and Spencer, B.F.	Framework for Consequence-based Management and Safety of Railroad Bridge Infrastructure Using Wireless Smart Sensors (WSS)	June 2015
042	Spencer, B.F. and Gardoni, P. (Eds.)	Innovations and Advances in Structural Engineering: Honoring the Career of Yozo Fujino	August 2015
043	Asai, T. and Spencer, B.F.	Structural Control Strategies for Earthquake Response Reduction of Buildings	August 2015
044	Kim, R. and Spencer, B.F.	Modeling and Monitoring of the Dynamic Response of Railroad Bridges using Wireless Smart Sensors	September 2015
045	Wierschem, N. and Spencer, B.F.	Targeted Energy Transfer using Nonlinear Energy Sinks for the Attenuation of Transient Loads on Building Structures	September 2015
046	Fahnestock, L.A. and Hashash, Y.M.A.	Structural and Geotechnical Observations after the April 25, 2015 M7.8 Gorkha, Nepal Earthquake and its Aftershocks	December 2016
047	Yoon, H. and Spencer, B.F.	Enabling Smart City Resilience: Post-disaster Response and Structural Health Monitoring	December 2016
048	Kozak, D.L., Luo, J., Olson, S.M., LaFave, J.M., and Fahnestock, L.A.	Modification of Ground Motions for use in Central North America: Southern Illinois Surface Ground Motions for Structural Analysis	September 2017
049	Fermandois, G.A. and Spencer, B.F.	Development and Implementation of a Multi-axial Real-time Hybrid Simulation Framework	September 2018