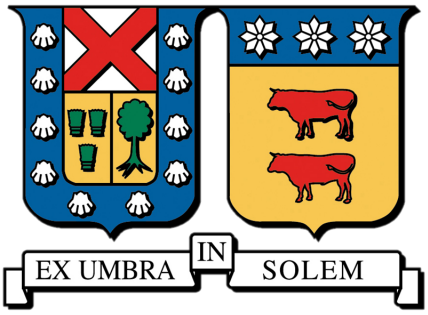
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE OBRAS CIVILES
SANTIAGO - CHILE



ROBUST ADAPTIVE COMPENSATION OF FORCE-BASED REAL-TIME HYBRID SIMULATION TESTING WITH UNCERTAIN COMPLIANCE SPRING AND FORCE MEASUREMENT NOISE

Thesis by

Diego Ignacio Araya Iglesias

Submitted to the Civil Engineering Department
in partial fulfillment of the requirements for the title of
Civil Engineer
and the degree of
Master of Science in Civil Engineering

Advisor
Gastón Andrés Fermandois Cornejo

July 28th, 2023

ABSTRACT

Real-time Hybrid Simulation (RTHS) is an effective method for estimating the dynamic response of structural systems. This framework has been demonstrated to work very well in many research applications of structural engineering. In an RTHS experiment, the structure is divided into a numerical part and an experimental part (specimen), coupled through a transfer system (actuator) that imposes conditions from the numerical structure onto the specimen.

The typical framework involves imposing displacements on the specimen. However, it is limited when the displacement is imposed over a high-stiffness specimen (e.g., axial deformation of a column or lateral displacement of a shear wall). These values are of a very small order of magnitude, which can potentially cause numerical instability in the simulation. The solution to this issue is to employ a force-based framework, where forces are commanded from the numerical structure to the actuator. However, it is well-known that the actuator is designed to operate based on displacements since it intrinsically possesses a highly rigid oil column and highly nonlinear behavior. This makes force control very sensitive to controller parameters, leading to instability in the simulation.

This study proposes a force-based real-time hybrid simulation (RTHS) framework with robust adaptive model-based compensation. Adding a compliance spring between the loading actuator and a rigid specimen is an alternative to measure restoring forces through load cells with significant noise. However, we considered compliance spring and load cell properties to be uncertainties. Robust adaptive model-based compensation will be employed to overcome force-tracking errors between substructures. The proposed methodology will be verified in a virtual RTHS environment, where parametric studies will be considered to check the system's robustness over uncertain compliance and specimen properties.

Keywords: *force-based real-time hybrid simulation, compliance spring, adaptive compensation, model-based controller, robust compensation.*

In memoriam of

J.H.I.D. $\sim$ D.S.A.C. $\sim$ C.I.T.A.C.

Regards

En primer lugar agradezco a mis padres, Carlos Araya y María Loreto Iglesias por su amor, preocupación, motivación y apoyo que siempre han estado presente en mi vida. A mis hermanos Constanza y Joaquín quienes siempre están para apoyarme. A mi amada compañera de vida Michelle, quien fué un pilar fundamental en todo este proceso, y lo es también en mi vida. A María Eugenia del Carmen por su preocupación y apoyo.

Agradezo profundamente el apoyo y la paciencia del Profesor Gastón Fermandois en este largo proceso formativo. A mis compañeros de equipo Cristóbal Galmez, María Quiroz, Diego Mera y Daniel Maurel por brindarme su apoyo cuando solicitaba su ayuda, y por hacer este proceso sea divertido y de crecimiento tanto personal como de trabajo en equipo.

Contents

List of Tables

List of Figures

1 Introduction

1.1. Motivation

Real-time hybrid simulation (RTHS) (Nakashima et al., 1992) is a powerful tool that combines a numerical computer simulation of a structure subjected to recorded seismic accelerations with an experimental test in the laboratory. This form of structural analysis has proven to be a very effective and low-cost technique compared to other proposed methodologies: only one physical component is required as a test specimen (e.g., columns or MR dampers) instead of building a complete structure in the laboratory.

This technique has made significant progress in recent years (Bousias, 2014, Najafi et al., 2023). To carry out the simulation, the structure under study is separated into a numerical substructure (NS), which will be evaluated using a numerical simulation on a computer, and an experimental substructure (ES), which is tested in the laboratory using a transfer system (e.g., shake tables or servo-hydraulic actuators). The most used methodology for an RTHS test is based on command displacements from the NS to the actuator and measuring forces from the specimen that return to the NS as inputs to complete the hybrid loop. The boundary conditions of displacement compatibility and force equilibrium must be satisfied to validate the simulation. On the other hand, force-based hybrid testing consists of commanding forces from the NS to the actuator, which applies them on the specimen, from where the measured states (position, velocity, and acceleration) return as input to the NS.

One of the first experimental force-based tests was the *Effective Force Testing* (EFT) proposed by Dimig et al. (1999). In this study, the forces that are applied by the actuators can be calculated *offline* since the seismic record is known before the test, and the entire structure is experimentally tested. EFT checks the possibility of commanding forces by an actuator. Still, problems with the natural velocity feedback studied by Dyke et al. (1995) and limitations with the servo-valve and actuator capacities were evidenced.

Moreover, in a force-based test, low-magnitude displacement errors (compatibility) in the simulation could have the effect of significant errors in measured forces (equilibrium), leading the simulation to numerical and experimental instabilities. Shao and Reinhorn (2012) proposed the *Real-time Dynamic Hybrid Testing scheme* (RTDHT) with a novel solution developed by Sivaselvan et al. (2008), consisting of introducing a *compliance spring* between the piston and the test specimen. This idea is initially proposed by Pratt et al. (2002) for a robotic force control. The main goal of the compliance spring is to add flexibility to the system. Then, the controller can be formulated in a displacement-based environment while forces are still commanded from the NS as input to the controller. In addition, the force can be measured by load cells or using Hooke's Law (i.e., measuring the elongation of the spring with a displacement transducer and multiplying by its elastic stiffness).

The literature provides some examples of hybrid testing with compliance springs. Chae et al.

(2018) compare simulations with force-based and displacement-based compensation using his designed Adaptive Time Series Compensation (ATS) (Chae et al., 2013). In this study, a proposal made by Chae is to use a beam as a compliance element when axial loads are applied on large-scale test structures. The beam must be designed to ensure its linear elastic behavior works as a spring. Later, Chae and Rabiee (2018) implement EFT with ATS compensation and a compliance frame attached to non-linear structures and verify the advantages of the EFT for large-scale systems with a force control scheme.

On the other hand, different dynamic compensation methods have been developed for RTHS tests and are available to reduce the synchronization error between commanded and measured signals. Most methods assume a constant time delay of the transfer system (Horiuchi et al., 1999). However, the time delay is affected by the interaction between the actuator and the specimen. This issue is known as control-structure interaction (Dyke et al., 1995). Later, novel compensation schemes were developed. Carrion and Spencer Jr (2007) proposes a method based on the servo-hydraulic and the physical specimen model, known as model-based compensation.

One of the main objectives of the study and development of experimental methodologies in civil engineering is to represent a seismic event as realistically as possible while obtaining a reliable response from the system to analyze its material's behavior or weaknesses. An example of interest, among all the possible existing challenges, is the development and application of hybrid simulation tests using multiple actuators (Elnashai et al., 2006), which allows for the imposition of displacements, rotations, and forces in more than one direction on the degree of freedom of interest of the specimen.

In this study, a force-based RTHS will be performed considering a high-stiffness specimen and a compliance spring, which adds flexibility to the test setup. To allow good synchronization between substructures, a robust adaptive model-based compensator, based on the study done by Gálmez and Fermandois (2022), will be formulated to work in a force-based environment. This formulation will consider uncertainty in the dynamic properties of the specimen and compliance spring. Also, the noise of load cell measurement will be incorporated into the simulation. The performance of the robust compensator will be compared with the traditional model-based compensator through a series of virtual tests using a modified version of the RTHS benchmark problem.

1.2. Objectives

The main objective of this work is to develop and virtually validate an adaptive model-based dynamic compensation algorithm with robust calibration for a force-based real-time hybrid simulation test, considering a rigid specimen and uncertainty in the physical parameters of the structure.

The specific objectives are the following:

1. Implement the designed algorithm of the adaptive model-based compensator to work in a force-controlled environment using a compliance spring.
2. Study the robustness of the adaptive compensator considering different levels of uncertainty in the physical parameters and signal measurement noise, comparing it with an inverse feed-forward controller.
3. Validate the proposed methodology considering different seismic records, levels of structural parameters uncertainty, and measurement noise.

1.3. Document organization

- **Chapter 2**: A literature review of the state-of-the-art methods for experimental testing, real-time hybrid simulation, and dynamic compensation is presented, focusing on real-time hybrid simulation based on forces and dynamic compensation using an Inverse Feedforward controller and model-based adaptive compensation.
- **Chapter 3**: A summary of the theoretical framework that will serve as the basis for the research is presented: dynamic systems, substructuring method, transfer system model, and dynamic compensation.
- **Chapter 4**: The formulation of the problem to be addressed in this work is described: the reference structure and its partitioning, which will be used to develop this study. Furthermore, the considerations taken into account during the definition of the parameters are described.
- **Chapter 5**: The theoretical framework, methodology, and design of the Inverse Feedforward and Adaptive model-based compensation controllers are presented.
- **Chapter 6**: All the cases considered and developed in this study are presented, along with their respective analyses and descriptions of the obtained results.
- **Chapter 7**: The conclusions of the study are presented, along with possible future work horizons to improve the proposed methodology.

2 Literature review

2.1. Experimental testing methods

In civil engineering, structural design has been fundamental for the development of different societies around the world, delivering as a final product the construction of homes and roads that have been able to withstand natural phenomena such as earthquakes or hurricanes, ensuring the safety of people and the continuous development of countries. Representing a dynamic system in a finite element mathematical model to study its response corresponds to the final result of various research that has aimed at creating and validating models that realistically represent these structural systems. Experimental tests have played a fundamental role in understanding, developing, and validating models that represent the behavior of structures or constitutive laws of materials (Fermandois and Spencer Jr, 2018) that will be used in structural design.

The *quasi-static test* is an experimental test in which a predefined cyclic load of increasing amplitude is applied to the specimen over an extended time. This test has allowed for a slower and more detailed observation of damage progression (e.g., cracks) and failure mechanisms in the response obtained from the model (Nakashima, 2020), as well as identifying the nonlinear hysteresis behavior of materials (Najafi et al., 2023). This method has been the basis for codes and design guidelines for structural elements (McCrum and Williams, 2016) due to its low cost, allowing structures to be tested on a more realistic scale. However, because the loads are predefined, i.e., the current load does not affect the load of the next step, this test does not allow for the representation of the specimen's dynamic behavior. In summary, the quasi-static test provides a reliable result of the specimen's capacity, but not its performance (Phillips and Spencer Jr, 2012).

On the other hand, the *shaking table test* is the best method to realistically represent the inertial response of the structure to seismic solicitations in real-time. However, the limitations of using this method are the high cost of the equipment needed in the laboratory and building the structures that will be tested. Additionally, it only allows for the application of seismic loads. A solution used to reduce time and cost is to use scaled-down models. However, local phenomena such as fatigue, local buckling, or crack propagation (Phillips and Spencer Jr, 2012) are not well represented when tiny scales are used.

New alternatives, such as *Pseudo-dynamic testing* or *Hybrid testing*, are intended to reduce performance limitations, simulation time, and computational cost.

2.1.1. PsD test and Hybrid Simulation

The concept of *Pseudo-dynamic testing* was proposed by Hakuno et al. (1969) and later formalized by Takanashi et al. (1975). Dermitzakis and Mahin (1985) conducted the first test, including the concept of substructuring, which involves partitioning the reference structure into an experimental and a numerically modeled part. This type of testing is later known as hybrid simulation. Nakashima (2020) summarizes the staged hybrid simulation's history and current state.

A *Hybrid Simulation test* is a method that combines numerical simulation with experimental laboratory testing. This means that it is not necessary to construct the entire structure to be tested. This method allows testing critical elements of the reference structure that are expected to have a nonlinear response, which is difficult to obtain in computer simulations. In contrast, the rest of the structure is modeled on the computer and solved numerically simultaneously with the experimental test. Loads are imposed over an extended time, allowing signal delays and actuator limitations to be disregarded. On the other hand, when tests with substructuring are performed, special care must be taken to ensure that the boundary conditions at the interface between the numerical and experimental substructure are realistically represented. Otherwise, unexpected results may be obtained.

Although the Hybrid Simulation test is an effective and low-cost alternative for testing, due to the extended time scale in which loads are applied, the rate-dependent effects of the experimental substructure are not captured. This led to the concept of Hybrid Simulation on a real-time scale.

2.1.2. Real-time Hybrid Simulation

The first *Real-time Hybrid Simulation* (RTHS) test was published by Nakashima et al. (1992). This test is excellent for testing rate-dependent elements such as passive, semi-active, and active control devices.

Commonly used RTHS test, the numerical substructure is subjected to ground motion caused by an earthquake. The term *Real-time* refers to the synchronization of the time step of the seismic record with that of the simulation, meaning that the equation of motion must be solved at the same time step as the earthquake. This process involves a higher computational demand and demands on the transfer system. The equation of motion of the numerical substructure is solved using a numerical integration method per step to obtain the displacements imposed on the experimental substructure by actuators. The force between the piston and the specimen is measured using sensors, which enter as input to the numerical equation. In this way, the displacement of the next step is calculated. The scheme of a typical RTHS test is shown in Figure 2.1.

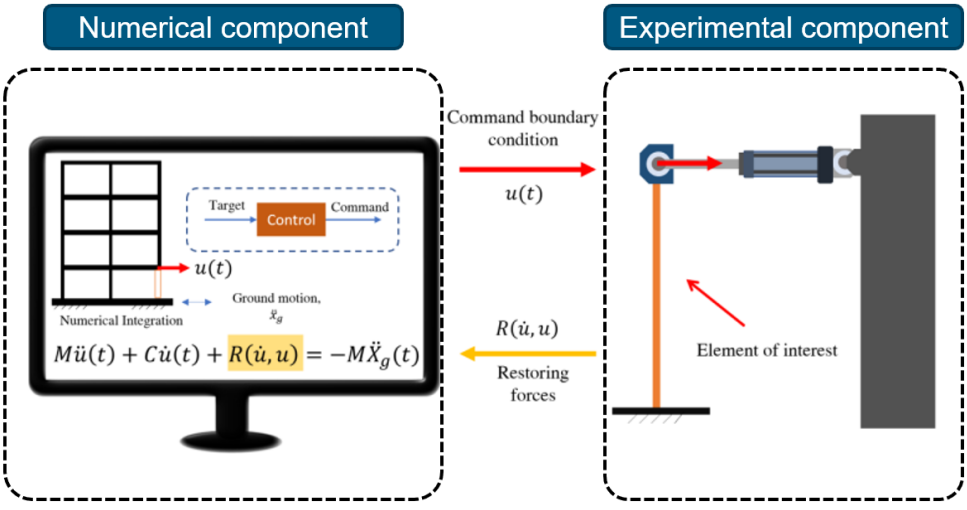


Figure 2.1: Real-time hybrid simulation scheme (Najafi and Spencer Jr, 2021).

The study of hybrid simulation encompasses several lines of research that are currently under

development. McCrum and Williams (2016) conduct a review of the state of the art in several research lines. An example is the development of new integration methods that are stable, converge to a solution, and are capable of integrating within a defined time step (Carrion and Spencer Jr, 2007, Dyke et al., 1995). The *Explicit* methods are integration schemes that compute the response of the next time step based on the previous one. They are widely used in HS due to their ease of implementation and low computational cost. However, most of these methods are *conditionally stable*, limiting the maximum time step that can be used. The *Central Difference Method* (CMD) is the scheme that has been most commonly used in PsD and Real-time Hybrid Testing (RTHT) (Benson Shing, 2008, Bonnet et al., 2008). An example is the *Chang Method* (CM) (Chang, 2002, Chang et al., 2011), which is unconditionally stable depending on the material behavior. It also exhibits better error propagation properties than the CMB and explicit Newmark methods. Chen and Ricles (2008) developed the *CR Method* for RTHT using linear systems, including a stability criterion related to damping. The *Implicit* methods calculate the value of the response at the next time step using one or more values of the present and past response. Unlike explicit methods, implicit methods are mostly unconditionally stable. The Newmark method (Newmark, 1959) is one of the most widely used implicit methods. However, the method introduces numerical damping for values of $\gamma > 0.5$. The *Operator-splitting* (OS) methods, initially proposed by Hughes et al. (1979), is suggested as an improvement over implicit methods as they do not require an iterative process for the entire procedure. The method is implicit for the linear part of the problem and explicit for the nonlinear part. Other methods have been developed based on OS, such as the alpha OS (Combescure and Pegon, 1997). Furthermore, improvements have been made to the method to expand its implementation in various types of simulations (Bonelli and Bursi, 2005, Ghaboussi et al., 2006).

On the other hand, several studies have been conducted on RTHS tests using different integration methods to compare their performances, results, strengths, and weaknesses. Bonnet et al. (2008) conducts several tests considering different integration methods using various multi-degree-of-freedom structures. Bas and Moustafa (2020) performs real-time hybrid simulation tests on a moment-resisting frame considering nonlinear behavior, aiming to employ and evaluate the performance of different integration methods to identify their limitations.

The current challenges of RTHS related to this work include force-based real-time hybrid simulation and adaptive dynamic compensation.

2.1.3. Force-based RTHS

The real-time hybrid simulation based on forces (FBRTHS) test has the main characteristic of commanding forces to the transfer system to be applied to the specimen. The first test conducted in this format corresponds to the *Effective Force Testing* (EFT) proposed and performed by Dimig et al. (1999) for a single DOF system showing issues on the actuator at the time to impose forces on the specimen at its natural frequency due to the control-structure interaction (CSI) (Dyke et al., 1995) and the light damp of the structure. Later the same study was conducted by Shield et al. (2001) with the velocity feedback correction applied. Nakata (2013) develops a controller to implement EFT in multi-degree-of-freedom tests, where initially it is tested with a two-degree-of-freedom structure (Nakata and Krug, 2013) and then expands the study to MDOF (Nakata et al., 2014).

Reinhorn et al. (2004a) propose an innovative method for conducting real-time hybrid simulation using force-based control, later formalized as *Real-time Dynamic Hybrid Simulation* (RTDHS) (Shao and Reinhorn, 2007). This method is developed for MDOF structures and can be imple-

mented to utilize either an actuator, a shaking table, or both within the same simulation. The study is demonstrated for a 3-story structure, with the second story representing the specimen. Substructuring is developed based on the approach proposed by Shing et al. (1994). The substructuring flowchart, illustrating the signals composing the commanded force to the specimen, is presented in Figure 2.2. The comprehensive workflow, including design, analysis, implementation, and results, up to the year 2007, is detailed in the document by Shao (2007). (Shao and Reinhorn, 2012) illustrates the RTDHT scheme for various types of existing dynamic tests, including quasi-static tests, PsD tests, and EFT.

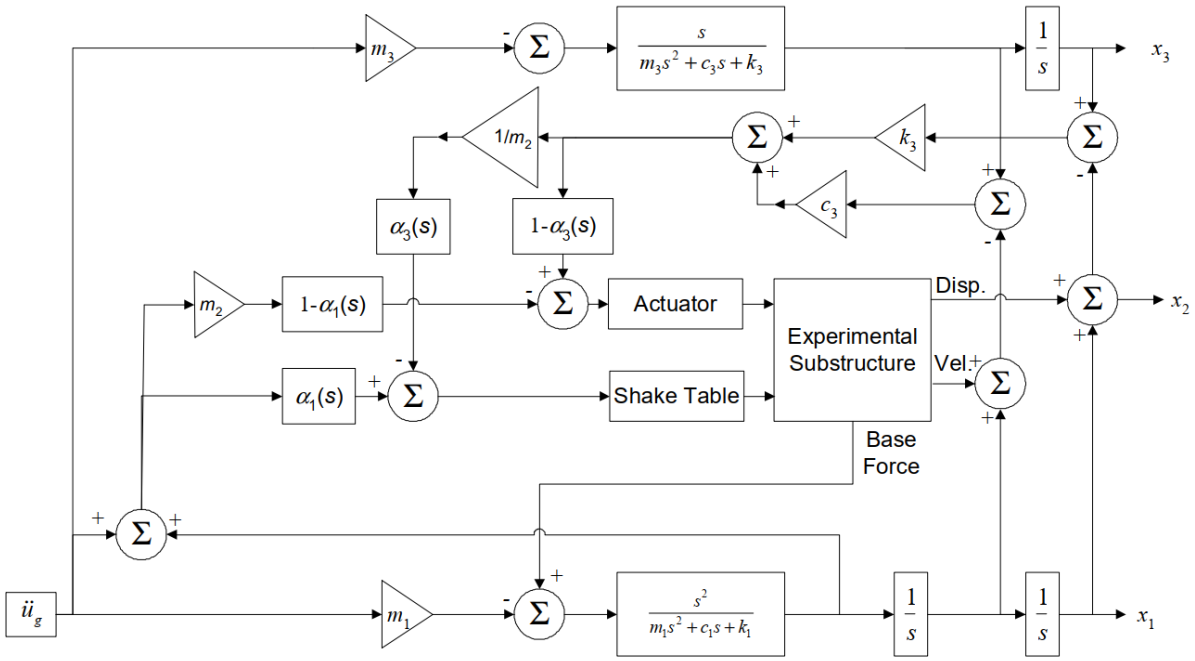


Figure 2.2: Force-based substructuring flowchart (Reinhorn et al., 2004b).

The particular interest in studying this simulation methodology for the present research arises from testing highly rigid specimens. Displacement control on a specimen with high rigidity can cause instability in the simulation, making force control methodology the solution to this problem. Additionally, the implementation of this methodology is limited by the oil flow capacity of the actuator, which is mechanically rigid, as higher forces require a higher flow rate. Sivaselvan et al. (2008) proposes the use of a compliance spring placed between the specimen and the actuator shown in Figure 2.3, which adds flexibility to the system, improving the force control. Due to the use of the compliance spring, it is possible to compensate for displacements within a force-based test setup (Sivaselvan et al., 2008). Subsequently, Chae et al. (2013) demonstrates that within the same force-based test framework, it is possible to apply either force-based or displacement-based controllers. Bousias (2014) provides an overview of state of the art in hybrid simulation for rigid specimens up to the year 2014.

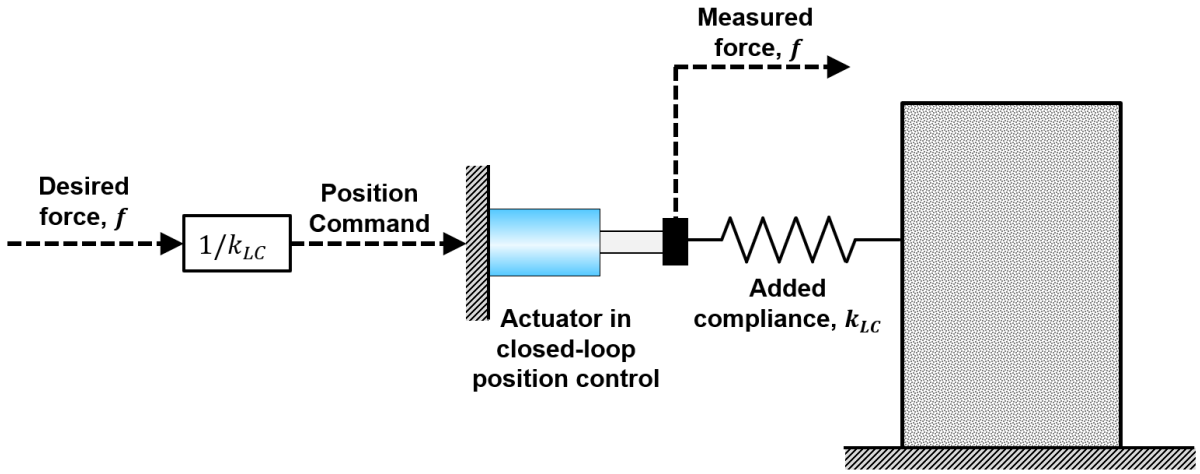


Figure 2.3: Force-based implementation for a rigid specimen using a compliance spring (Sivaselvan et al., 2008).

2.2. Dynamic Compensation

The actuator is a system with its own dynamics of nonlinear nature, in which each main component is modeled as a transfer function (Merritt, 1967). These functions are the result of linearizing the equations that form the dynamics of each component of the system (Carrion and Spencer Jr, 2007). Due to its dynamics, a delay occurs between the commanded and measured signals, a situation that, if not controlled, can lead to numerical instability in the simulation. Additionally, the interaction between the actuator and the specimen also contributes to the delay of the signals (Dyke et al., 1995).

Initially, the dynamics of the actuator are modeled as a time delay between the commanded and measured signals. The study conducted by Horiuchi et al. (1999) estimates the signal delay and performs tests considering the actuator as a pure delay by sweeping values close to the estimated delay to evaluate its effect on the simulation. They also propose an N^{th}-order polynomial adjustment, which is successfully implemented in subsequent studies (Blakeborough et al., 2001). Other authors also study time delay and develop their own compensation methods. Zhao et al. (2003) studies the phase shift between signals to develop a phase lead compensator. Wallace et al. (2005b) develops a method in which they calculate the critical time delay of a linear system. Shao et al. (2006) uses a Smith Predictor to compensate for the time delay between displacements. Chen (2007) proposes a compensation method using the inverse of the simplified transfer function of the actuator in the discrete domain to cancel its dynamics.

The mentioned compensations consider the actuator as a constant time delay throughout the simulation, whereas in reality, it is not the case. Uncertainty in physical parameters is also not included. Therefore, there is a need to study more sophisticated compensation methods that can account for changes in the actuator. This leads to the emergence of model-based adaptive compensations, which have parameters that adapt within the simulation and are constructed based on the parametric model of the actuator. Phillips and Spencer Jr (2011) proposes a model-based controller consisting of a feedforward controller and a feedback controller, as shown in Figure 2.4.

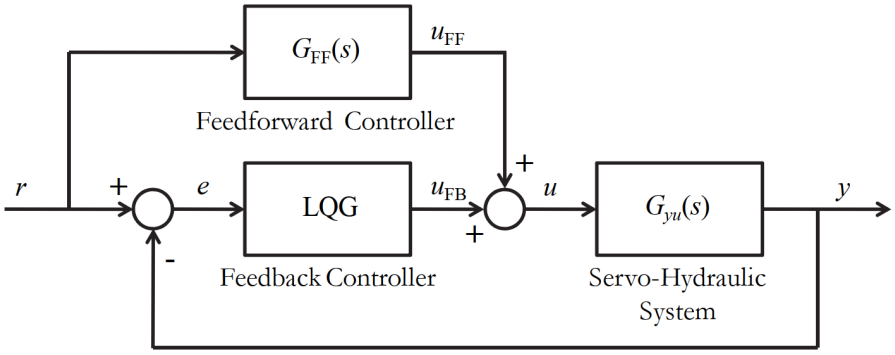


Figure 2.4: Block diagram of model-based controller implementation (Phillips and Spencer Jr, 2011).

Later, Chen and Ricles (2009) performs a comparison of different compensation methods that use the inverse of the actuator to cancel its dynamics, which then leads to the proposal of an adaptive feedforward compensation using the inverse of the control plant (Chen and Tsai, 2013) which then is implemented in the framework proposed by Phillips and Spencer Jr (2011) creating the *Adaptive Model-based Compensator* (AMB). This compensation method utilizes an adaptation law based on a gradient algorithm to estimate the delay between signals (Ioannou and Fidan, 2006). Its implementation is shown in Figure 2.5.

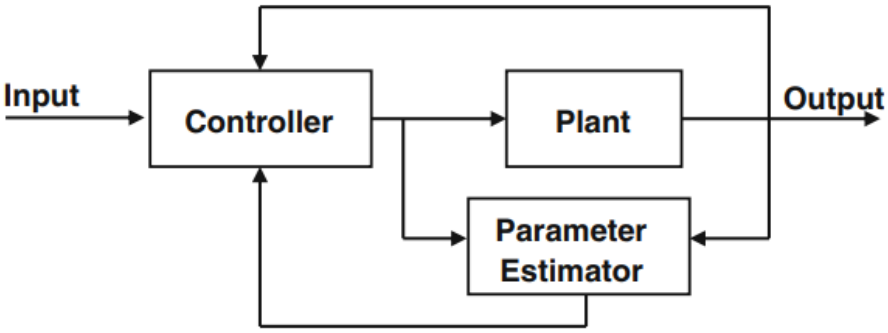


Figure 2.5: Block diagram of adaptive control implementation (Chen et al., 2015).

The study by Chae et al. (2013) introduces an adaptive compensation method known as "Adaptive Time Series." The adaptive process relies on the "Least-square Method," which utilizes both the commanded and measured signals for the adaptation law. Other forms of adaptive compensation have also been proposed, such as the "Conditional Adaptive Time Series" (Palacio-Betancur and Soto, 2019), which builds upon ATS but employs the "Recursive Least-square Method." This method has lower computational costs but requires design parameters that are not as intuitively obtained.

Gálmez and Fermandois (2022) proposes an *Adaptive Model-based Compensator* based on the AMB by Chen et al. (2015), with the difference that it uses feedforward compensation without the LQG feedback regulator, as the latter requires more knowledge for proper design. Additionally, the goal is to demonstrate the adaptive capability of the feedforward compensator without the regulator (Fermandois et al., 2020). Thus, only the parameters of the actuator are required as prior knowledge before carrying out the optimization process.

3 Background

3.1. Dynamic model representation

A *dynamical system* refers to a system that changes its behavior over time. The most commonly used approach is studying the system's response to external stimulations (i.e., forces). This document analyzes a structure's (building) response to ground accelerations (earthquakes). To do this, a simplified discrete analytical model of the structure is first generated, where the building is modeled as a dynamic system composed of masses and springs. This model is constructed considering the following:

1. The mass is concentrated on each floor. These masses can only move horizontally along a single axis. No displacements in other directions or rotations are considered.
2. Columns only provide stiffness to the system and connect the concentrated masses to the ground.
3. Energy dissipation in the structure occurs in reality in various ways, such as friction between steel connections or between concrete crack surfaces. This dissipation is represented as intrinsic viscous damping of the structure in the model.
4. The materials that make up the structure behave only in the linear elastic range.
5. The system parameters are considered linear-time-invariant (LTI).

Figure 3.1 shows a diagram illustrating how a simplified model represents an actual building. As an illustrative example, a multi-degree of freedom (MDOF) system of n floors is considered. Based on the simplified model shown in figure 3.1, the differential equation of motion for an MDOF structure defined in a domain $\mathbf{\Omega}$ is described as follows:

$$\mathbf{\Omega}: \quad \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{p}(t) \tag{3.1}$$

Where $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ are the mass, damping, and stiffness matrices, respectively. $\mathbf{x}(t) = \{x_1(t), x_2(t), x_3(t)\}^T \in \mathbb{R}^n$ is the displacement vector relative to the ground. $\dot{\mathbf{x}}(t) \in \mathbb{R}^n$ and $\ddot{\mathbf{x}}(t) \in \mathbb{R}^n$ are the velocity and acceleration vectors obtained from the respective time derivatives from $\mathbf{x}(t)$. $\mathbf{p}(t)$ is the external load vector. In structural engineering, the external forces considered correspond to the inertial forces induced by the earthquake. Thus, the vector $\mathbf{p}(t)$ can be replaced by $\mathbf{M\Gamma}\ddot{x}_g(t)$, where $\ddot{x}_g(t)$ is the vector of the seismic record and $\mathbf{\Gamma}$ is the seismic participation vector. Considering the above, equation (3.1) is written as shown in equation (3.2). Initial conditions for equation (3.2) are $\mathbf{x}(0) = \dot{\mathbf{x}}(0) = \mathbf{0_{n\times 1}}$.

$$\mathbf{\Omega}: \quad \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{M\Gamma}\ddot{x}_g(t) \tag{3.2}$$

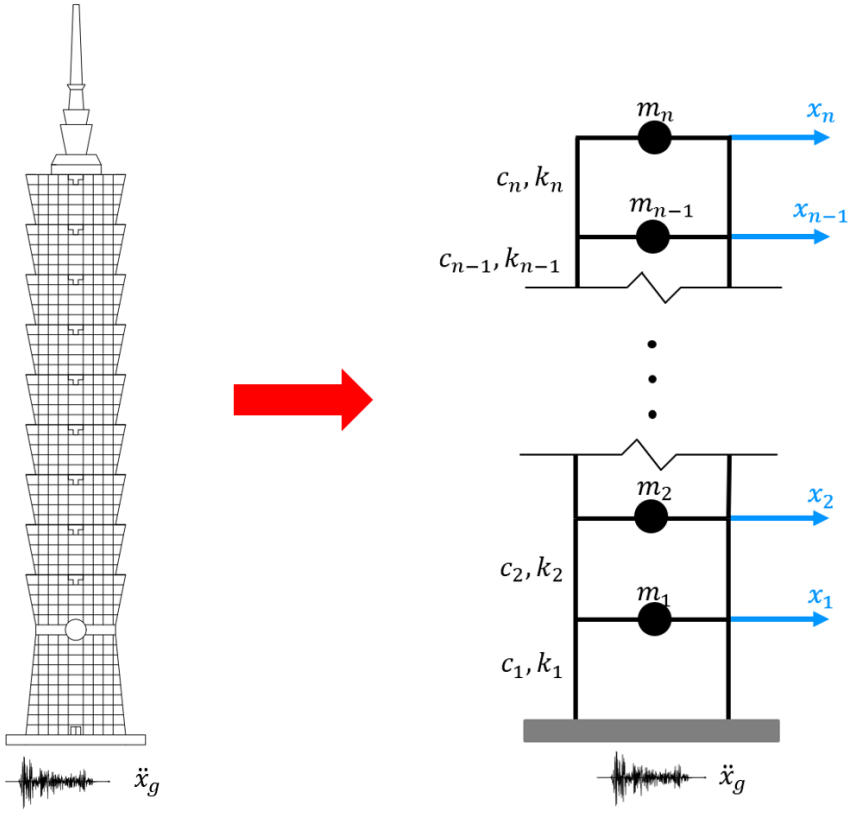


Figure 3.1: Real building (left) and a simplified model (right).

3.1.1. State-Space representation

The second-order equation of an MDOF system (3.1) can be rewritten as a first-order differential equation regarding time. The state vector is defined as $\mathbf{z}(t) = [\mathbf{x}(t), \dot{\mathbf{x}}(t)]^T$, and the rate of change of state vector is defined as $\dot{\mathbf{z}}(t) = [\dot{\mathbf{x}}(t), \ddot{\mathbf{x}}(t)]^T$. The general form of the sate-space equation is shown as follows:

$$\dot{\mathbf{z}}(t) = \mathbf{A}\mathbf{z}(t) + \mathbf{B}\mathbf{p}(t) \tag{3.3}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{z}(t) + \mathbf{D}\mathbf{p}(t) \tag{3.4}$$

Where $\mathbf{y}(t)$ is the vector of responses or outputs, $\mathbf{p}(t)$ is the input vector, $\mathbf{A}$ is the state matrix, $\mathbf{B}$ is the input matrix, $\mathbf{C}$ is the output matrix, and $\mathbf{D}$ is the feedthrough matrix. Since the system is considered time-invariant, the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ do not depend on time. The input equation of the system represented as state-space is shown in equation (3.11), where $\mathbf{p}(t) = \mathbf{M^{-1}\Gamma}$ is substituted.

$$\dot{\mathbf{z}}(t) = \begin{Bmatrix} \dot{\mathbf{x}}(t) \\ \ddot{\mathbf{x}}(t) \end{Bmatrix} = \begin{bmatrix} \mathbf{0}_{n\times n} & \mathbf{I}_{n\times n} \\ \mathbf{M^{-1}K} & \mathbf{M^{-1}C} \end{bmatrix} \begin{Bmatrix} \mathbf{x}(t) \\ \dot{\mathbf{x}}(t) \end{Bmatrix} + \begin{bmatrix} \mathbf{0}_{n\times 1} \\ \mathbf{M^{-1}\Gamma} \end{bmatrix} \ddot{x}_g(t) \tag{3.5}$$

The output vector $\mathbf{y}(t)$ is obtained through a linear combination of the states and the input signal, as shown in equation (3.10). Illustratively, equation (3.12) shows the following outputs: Displacements $\mathbf{x}(t)$, velocities $\dot{\mathbf{x}}(t)$, accelerations relative to the ground $\ddot{\mathbf{x}}(t)$, absolute acceler-

ations $\ddot{\mathbf{x}}_{abs}(t)$ and the story forces ($\mathbf{V}(t)$), for the n degrees of freedom considered. $\mathbf{I}_{p\times q}$ and $\mathbf{0}_{p\times q}$ correspond to the identity and zero matrices, respectively, with dimensions of p rows and q columns.

$$\mathbf{y}(t) = \begin{Bmatrix} \mathbf{x}(t) \\ \dot{\mathbf{x}}(t) \\ \ddot{\mathbf{x}}(t) \\ \ddot{\mathbf{x}}_{abs}(t) \\ \mathbf{V}(t) \end{Bmatrix} = \begin{bmatrix} \mathbf{I}_{n\times n} & \mathbf{0}_{n\times n} \\ \mathbf{0}_{n\times n} & \mathbf{I}_{n\times n} \\ \mathbf{M^{-1}K} & \mathbf{M^{-1}C} \\ \mathbf{M^{-1}K} & \mathbf{M^{-1}C} \\ \mathbf{K} & \mathbf{C} \end{bmatrix} \begin{Bmatrix} \mathbf{x}(t) \\ \dot{\mathbf{x}}(t) \end{Bmatrix} + \begin{bmatrix} \mathbf{0}_{n\times 1} \\ \mathbf{0}_{n\times 1} \\ \mathbf{M^{-1}\Gamma} \\ \mathbf{0}_{n\times 1} \\ \mathbf{0}_{n\times 1} \end{bmatrix} \ddot{x}_g(t) \tag{3.6}$$

Schematically, figure 3.2 shows the block diagram of the system represented in its state-space form.

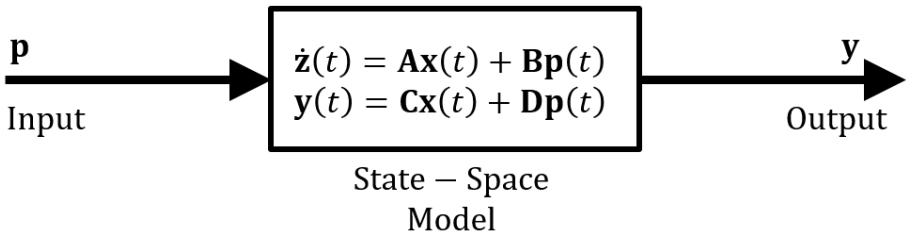


Figure 3.2: State-Space representation in block diagram.

3.1.2. Transfer function representation

This study will use transfer functions to represent single-degree-of-freedom (SDOF) systems. Thus, equation (3.13) for an SDOF system can be written as follows:

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = p(t) \tag{3.7}$$

Where m, c, and k are the mass, damping, and stiffness of the SDOF system, $x(t)$, $\dot{x}(t)$, and $\ddot{x}(t)$ as the position, velocity, and acceleration as a function of time, and $p(t)$ as the external force vector. The Laplace transform $\mathcal{L}\{\cdot\}$ allows the equation (3.13) to change from time to complex domain. Thus, the transfer function associated with the system described by equation (3.13) is given as follows:

$$G_{xp}(s) = \frac{X(s)}{P(s)} = \frac{1}{ms^2 + cs + k} \tag{3.8}$$

With "s" as the Laplace variable. Furthermore, if the external force $f(t)$ is replaced by the earthquake $-m\ddot{x}_g(t)$, equation (3.14) can be rewritten in terms of damping ratio and natural frequency as shown in equation (3.15), where $\omega_n = \sqrt{k/m}$ is the fundamental circular frequency with units in $[rad/s]$, and ζ is the damping ratio.

$$G_{x\ddot{x}_g}(s) = \frac{X(s)}{\ddot{X}_g(s)} = \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{3.9}$$

Schematically, figure 3.3 shows the block diagram of the system in equation 3.14 represented as a transfer function.

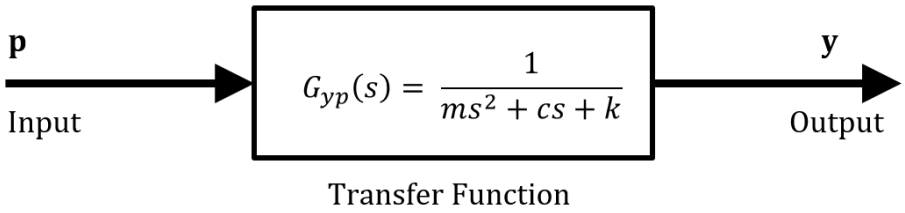


Figure 3.3: Transfer function representation in block diagram.

3.2. Substructuring method

A method known as substructuring is used to divide the domain $\mathbf{\Omega}$ of the reference structure into smaller partitions that are coupled and can be solved independently. However, the coupling between the divisions must be included in the motion equations as boundary conditions. Illustratively, the initial domain (reference structure) will be partitioned into two parts with their respective substructures, denominated *Numerical Domain* ($\mathbf{\Omega}^N$) and *Experimental Domain* ($\mathbf{\Omega}^E$), *Numerical Substructure* (NS) and *Experimental Substructure* (ES) respectively, where superscripts N and E denotes numerical and experimental, respectively. Figure 3.4 shows on the left side of the system domain $\mathbf{\Omega}$. On the right side, the initial part is partitioned into two, where the boundary zone $\mathbf{\Gamma}_b$ is marked in green. Thus, the reference structure can be defined now as the union of the subdomains generated $\mathbf{\Omega} = \mathbf{\Omega}^N \cup \mathbf{\Omega}^E$. Furthermore, the response of the reference structure can be obtained by combining the response from solving the coupled motion equations of each substructure. The coupled numerical and experimental equations of motion are shown in equation (3.3) and equation (3.4), respectively. For simplicity, from now on, (t) will not be written for time-dependent variables since the following equations are obtained from equation (3.1).

$$\mathbf{\Omega}^N : \mathbf{M}^N\ddot{\mathbf{x}}^N + \mathbf{C}^N\dot{\mathbf{x}}^N + \mathbf{K}^N\mathbf{x}^N = \mathbf{p}^N + \mathbf{g}^N \tag{3.10}$$

$$\mathbf{\Omega}^E : \quad \mathbf{M}^E\ddot{\mathbf{x}}^E + \mathbf{C}^E\dot{\mathbf{x}}^E + \mathbf{K}^E\mathbf{x}^E = \mathbf{p}^E + \mathbf{g}^E \tag{3.11}$$

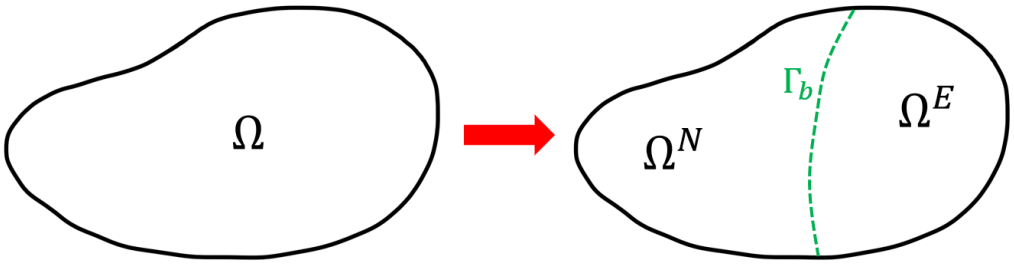


Figure 3.4: Initial domain (left) and partitioned domain (right).

The response of the entire system is obtained by combining the responses of the defined subdomains (numerical and experimental), which are coupled and must satisfy boundary conditions (BC) for displacements and forces, expressed in equations (3.5) and (3.6). Figure 3.5 shows schematics with the BCs of the subdomains, where in figure 3.5.a the displacement boundary conditions are shown, while in figure 3.5.b, the force boundary conditions between both subdomains are depicted.

$$\mathbf{x}^N = \begin{Bmatrix} \mathbf{x}_i^N \\ \mathbf{x}_b^N \end{Bmatrix}, \quad \mathbf{x}^E = \begin{Bmatrix} \mathbf{x}_i^E \\ \mathbf{x}_b^E \end{Bmatrix} \tag{3.12}$$

$$\mathbf{g}^N = \begin{Bmatrix} \mathbf{g}_i^N \\ \mathbf{g}_b^N \end{Bmatrix}, \quad \mathbf{g}^E = \begin{Bmatrix} \mathbf{g}_i^E \\ \mathbf{g}_b^E \end{Bmatrix} \tag{3.13}$$

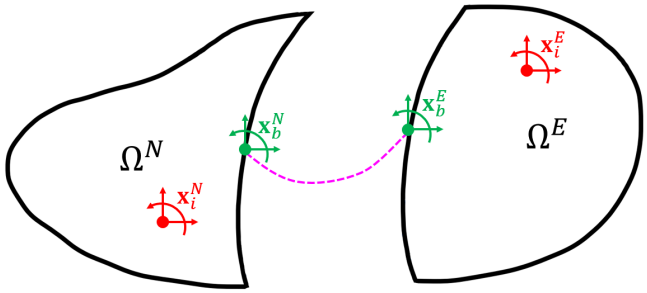
Where $\mathbf{x}^N = \{\mathbf{x}_i^N, \mathbf{x}_b^N\}^T$ and $\mathbf{x}^E = \{\mathbf{x}_i^E, \mathbf{x}_b^E\}^T$ are the coupling displacement vectors. $\mathbf{g}^N = \{\mathbf{g}_i^N, \mathbf{g}_b^N\}^T$ and $\mathbf{g}^E = \{\mathbf{g}_i^E, \mathbf{g}_b^E\}^T$ are the coupling force vectors between both domains, applied to their respective substructure, where the subscripts i and b indicate interior and boundary DOF, respectively. In this type of substructuring, it is assumed that the subdomains are coupled only at the boundary ($\mathbf{\Gamma}_b$) that separates them. Thus, for the displacement boundary conditions in equation (3.12), the following must be satisfied:

$$\mathbf{x}_b^N = \mathbf{x}_b^E \tag{3.14}$$

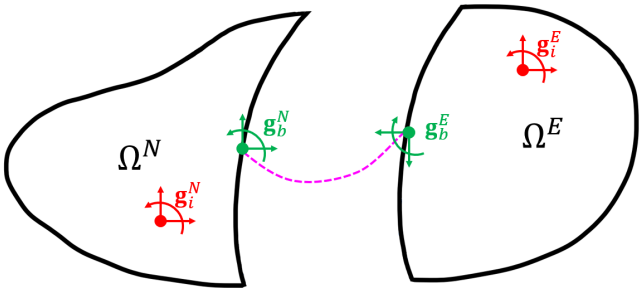
Similarly, the force boundary conditions shown in (3.13) must satisfy the following:

$$\mathbf{g}_b^N + \mathbf{g}_b^E = \mathbf{0}_b \tag{3.15}$$

Due to the aforementioned assumption, the internal BC forces are given by $\mathbf{g}_i^N = \mathbf{0}_i^N$ and $\mathbf{g}_i^E = \mathbf{0}_i^E$. Moreover, it should be noted that in an RTHS test, the BCs mentioned in equations (3.14) and (3.15) must be satisfied at each integration step of the simulation.



(a) Displacement.



(b) Force.

Figure 3.5: Boundary conditions.

3.3. Transfer system model

In the context of an RTHS test, the transfer system typically used is the *Servo-hydraulic Actuator*, which consists of a set of mechanical devices that impose displacements or forces on the Specimen. Figure 3.6 shows the model proposed by Carrion and Spencer Jr (2007), where the input is the target signal u and the output is the displacement of the Specimen x.

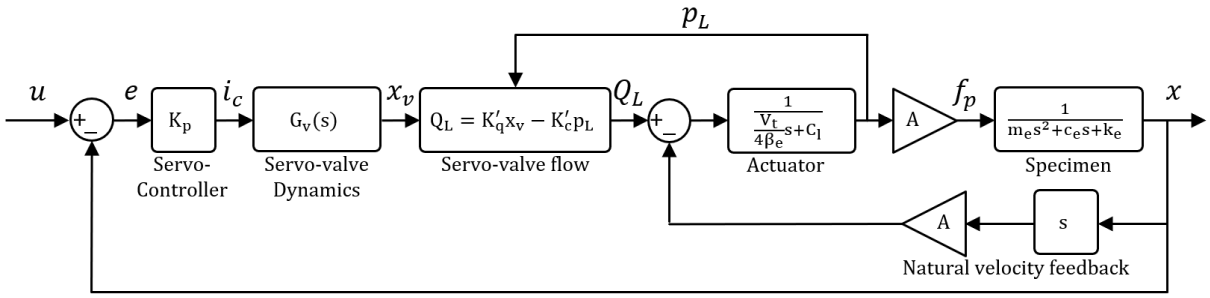


Figure 3.6: Transfer system model.

The main components modeled in the block diagram of Figure 3.6 correspond to the *Servo-controller*, *Servo-valve*, *Servo-valve flow*, *Actuator*, and the *Specimen*. The mathematical equations representing the physics of these components are presented below. Some of these equations are nonlinear; however, it is possible to linearize them around an operating point of the device (Carrion and Spencer Jr, 2007). This allows the modeling of each component as a transfer function in the Laplace domain. Thus, the modeled devices are connected in series to form the transfer system.

3.3.1. Servo-controller

The servo-controller is used to provide stability to the Actuator by using the measured signal as feedback to generate the error signal, as shown below:

$$e = u - x \tag{3.16}$$

Thus, in conjunction with a *PID* controller, from which only the proportional component K_p is used, the commanded signal to the servo-valve i_c is obtained as shown in Equation (3.17).

$$i_c = K_p e \quad \Rightarrow \quad i_c = K_p(u - x) \tag{3.17}$$

3.3.2. Servo-valve dynamics

The servo-valve corresponds to the interface between the electrical and mechanical components of the transfer system. As an input, it receives the electrical signal i_c, which controls the oil inside the chamber to move the piston. For low-frequency tests, the dynamics of this device can be linearly approximated as a first-order transfer function, with a valve gain k_v and a time delay constant τ_v, as shown in Equation (3.18). This approximation is proposed in the literature by Merritt (1967), which is also used by Carrion and Spencer Jr (2007) in their model.

$$G_v(s) \approx \frac{k_v}{s + t_v} \tag{3.18}$$

3.3.3. Servo-valve flow

For the flow of oil through the chambers of the actuator, the expression to calculate the flow rate Q_L is linearly approximated (Merritt, 1967), as shown below:

$$Q_L \approx K'_q x_v - K'_c p_L \tag{3.19}$$

Where K'_q is the flow valve gain, x_v is the position of the valve spool, K'_c is the gain of the pressurized flow of the valve, and p_L is the oil pressure that pushes the actuator. The system is linearized at the operating point when there is no flow, that is, $Q_L = x_v = p_L = 0$.

3.3.4. Actuator dynamics

The actuator is modeled as a transfer function generated from the equilibrium of forces related to the oil flow within the actuator chambers (Merritt, 1967), as shown in Equation (3.20).

$$Q_L = A\dot{x} + C_l p_L + \frac{V_t}{4\beta_e}\dot{p}_L \quad \Rightarrow \quad \frac{p_L(s)}{Q_L(s) - Asx(s)} = \frac{1}{\frac{V_t}{4\beta_e}s + C_l} \tag{3.20}$$

Where C_l is the leakage coefficient of the piston, V_t is the volume of the compressed fluid within the actuator chamber, β_e is the effective bulk modulus of the system, and A is the piston area.

3.3.5. Specimen

The specimen is represented as a transfer function based on the balance of forces generated from the interaction with the actuator piston. For a single degree of freedom (SDOF) system, the force balance is given by:

$$m_e\ddot{x}_e + c_E\dot{x}_e + k_e x_e + f_s = f^p \tag{3.21}$$

Where m_e, c_e, and k_e represent the mass, damping, and stiffness of the specimen, respectively, and f_s means the friction that occurs in the piston, which is neglected as modern actuators use low-friction seals, reducing its magnitude (Phillips and Spencer Jr, 2012).

3.3.6. Simplified model

The model proposed by Carrion and Spencer Jr (2007) in the previous section is simplified and rearranged equivalently by (Phillips and Spencer Jr, 2012), as shown in Figure 3.7. The simplified model reduces the servo-valve blocks to a single first-order transfer function called $G_s(s)$.

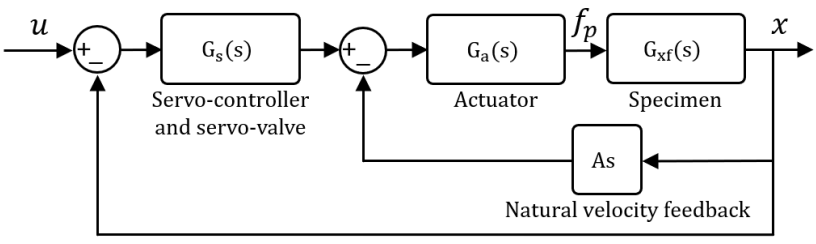


Figure 3.7: Transfer system model.

The transfer functions are as follows:

$$G_s(s) = \frac{\frac{K_p K_q}{\tau_v}}{s + \frac{1}{\tau_v}} \tag{3.22}$$

$$G_a(s) = \frac{\frac{4\beta_e A}{V_t}}{s + \frac{4\beta_e}{V_t K_c}} \tag{3.23}$$

$$G_{xf}(s) = \frac{1}{m_e s^2 + c_e s + k_e} \tag{3.24}$$

This simplified model is also used in the RTHS benchmark problem presented by Silva et al. (2020) and subsequently by Fermandois (2019) and Gálmez and Fermandois (2022).

3.4. Dynamic Compensation

A fundamental requirement when conducting real-time experimental tests is to meet the boundary conditions at each integration step. However, the transfer system used, whether it is an actuator or a shaker table, introduces errors between the target and measured signals, with the most common ones being time delays and amplitude differences. To overcome this problem, control system theory is used, which aims to reduce the error between the target and measured signals. The controller is also a system with physical behavior and implementation. In this study, two controllers will be used:

- Inverse feedforward (IFF).
- Adaptive model-based compensation (AMBC).

3.4.1. Inverse feedforward

The main objective of an *Inverse feedforward* controller (IFF) is to cancel the dynamics of the control plant using its inverse to reduce the tracking error between the target and measured signals, as shown in figure 3.8. The design used in this section is based on the work done by Fermandois (2019) and Phillips and Spencer Jr (2012).

Illustratively, let $G_p(s)$ be the control plant, a proper, stable, and non-minimum-phase transfer function, with n poles and no zeros, written as indicated in equation (3.25), Where k corresponds to the static gain and p_j are the poles of the transfer function.

$$G_p(s) = \frac{k}{\prod_{j=1}^{n}(s - p_j)} \tag{3.25}$$

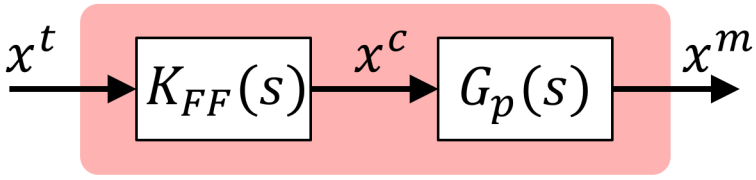


Figure 3.8: Block diagram of feedforward implementation.

From figure 3.8, x^t, x^c, and x^m correspond to the target, commanded, and measured signals, respectively. To design the inverse feedforward compensator, it is necessary to calculate the inverse of the control plant. Since the control plant is a rational transfer function, its inverse is obtained by interchanging the numerator and denominator, i.e., the poles are moved to the numerator, and the static gain is placed in the denominator. Then, the inverse feedforward compensator, denoted as $K_{\text{FF}}(s)$, is defined as the inverse of the control plant $G_p^{-1}(s)$, and its expression is given by:

$$K_{\text{FF}}(s) = G_p^{-1}(s) = \frac{\prod_{j=1}^{n}(s - p_j)}{k} \tag{3.26}$$

By expanding expression (3.26), a polynomial of degree n is obtained. Thus, $K_{\text{FF}}(s)$ can be alternatively written as follows:

$$K_{\text{FF}}(s) = c_0 + c_1 s + c_2 s^2 + c_3 s^3 + \ldots + c_n s^n$$

$$K_{\text{FF}}(s) = \sum_{i=0}^{n} c_i s^i \tag{3.27}$$

Where c_i are the coefficients obtained by expanding the polynomial of degree n and grouping constants. Following the implementation scheme of the inverse feedforward compensator shown in Figure 3.8, the commanded signal x^c is defined as follows.

$$x^c = K_{\text{FF}}(s)\, x^t \tag{3.28}$$

In equation (3.26), it can be observed that the expression corresponding to the inverse of the control plant is an improper transfer function, therefore, is not physically realizable. To solve this problem, the IFF controller will be implemented in the discrete-time domain, allowing for its realization. Moreover, this approach helps to better represent an RTHS experiment in the laboratory, as controllers must be expressed in discrete time to be used in a *digital signal processor* (DSP) (Phillips and Spencer Jr, 2012). Before transitioning to the discrete-time domain, it is convenient to express equation (3.28) in its parametric form, as shown below:

$$\begin{aligned} x^c &= \begin{bmatrix} c_0 & c_1 & c_2 & \ldots & c_n \end{bmatrix} \begin{bmatrix} 1 & s & s^2 & \ldots & s^n \end{bmatrix}^T x^t \\ x^c &= \begin{bmatrix} c_0 & c_1 & c_2 & \ldots & c_n \end{bmatrix} \begin{bmatrix} x^t & sx^t & s^2x^t & \ldots & s^nx^t \end{bmatrix}^T \end{aligned} \tag{3.29}$$

The terms involving the Laplace variable s correspond to derivatives in the time domain. Thus, equation (3.29) can now be expressed in terms of the signal x^t and its time derivatives as follows.

$$x^c = \begin{bmatrix} c_0 & c_1 & c_2 & \ldots & c_n \end{bmatrix} \begin{bmatrix} x^t & \dot{x}^t & \ddot{x}^t & \ldots & (x^t)^{(n)} \end{bmatrix}^T \tag{3.30}$$

Where $(x^t)^{(n)}$ denotes the n-th order time derivative of the signal x^t. Therefore, equation (3.30) written in the discrete-time domain is expressed as follows.

$$x^c[k] = \begin{bmatrix} c_0 & c_1 & c_2 & \ldots & c_n \end{bmatrix} \begin{bmatrix} x^t[k] & \dot{x}^t[k] & \ddot{x}^t[k] & \ldots & (x^t)^{(n)}[k] \end{bmatrix}^T \tag{3.31}$$

Or equivalently as follows:

$$x^c[k] = \mathbf{K}_{\text{FF}}\ \mathbf{r}[k] \tag{3.32}$$

Where $\mathbf{K}_{\text{FF}} = \begin{bmatrix} c_0 & c_1 & c_2 & \dots & c_n \end{bmatrix}$ is the vector of gains obtained from the control plant $G_p(s)$, and $\mathbf{r}[k] = \begin{bmatrix} x^t[k] & \dot{x}^t[k] & \ddot{x}^t[k] & \dots & (x^t)^{(n)}[k] \end{bmatrix}^T$ is a vector that contains the signal x^t and its temporal derivatives. In the discrete-time domain, the target signal is $x^t[k] = x^t(k\Delta t)$, with Δt being the sampling period. The backward difference method (BDM) will be employed to estimate higher-order time derivatives, considering an accuracy order of 1. Thus, the approximate time derivatives are obtained as follows:

$$\begin{bmatrix} x^t[k] \\ \dot{x}^t[k] \\ \ddot{x}^t[k] \\ \vdots \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots \\ \frac{1}{\Delta t} & \frac{-1}{\Delta t} & 0 & \\ \frac{1}{\Delta t^2} & \frac{-2}{\Delta t^2} & \frac{1}{\Delta t^2} & \\ \vdots & & & \ddots \end{bmatrix} \begin{bmatrix} x^t[k] \\ x^t[k-1] \\ x^t[k-2] \\ \vdots \end{bmatrix} \tag{3.33}$$

In an equivalent and compact form, equation (3.33) can be expressed as follows:

$$(x^t)^{(n)}[k] \approx \sum_{i=0}^{n} \frac{a_i x^t[k-i]}{b_0} \tag{3.34}$$

Where a_i and b_0 are numerical coefficients. Finally, by combining the gains from $\mathbf{K}_{\text{FF}}$ and the coefficients obtained from the estimation of the time derivatives using the BDM with equation (3.34), a *Finite Impulse Response* (FIR) filter of order n is formed. Thus, equation (3.32) is reformulated by considering the FIR filter for the discrete feedforward compensator as:

$$x^c[k] = k_0 x^t[k] + k_1 x^t[k-1] + k_2 x^t[k-2] + \ldots + k_n x^t[k-n] \tag{3.35}$$

Or equivalently as:

$$x^c[k] = \sum_{i=0}^{n} k_i x^t[k-i] \tag{3.36}$$

3.4.2. Adaptive model-based compensation

3.4.2.1. Methodology

The Adaptive model-based compensation (AMBC) design will follow the guidelines proposed in the study by Gálmez and Fermandois (2022). Figure 3.9 shows the block diagram implementation of AMBC, where x^t, x^c, and x^m correspond to the target, commanded and measured signals, respectively. In the scheme presented in Figure 3.9, it can be observed that the AMBC is a feedforward controller, with the detail that its coefficients are variable, unlike the case of the IFF controller mentioned in section 3.4.1 where its coefficients are constant throughout the simulation. In the adaptation process block, the coefficients are generated, which, together with the target signal x^t, are taken by the feedforward block to generate the commanded signal x^c. This compensation aims to reduce the synchronization error $e = x^m - x^t \to 0$.

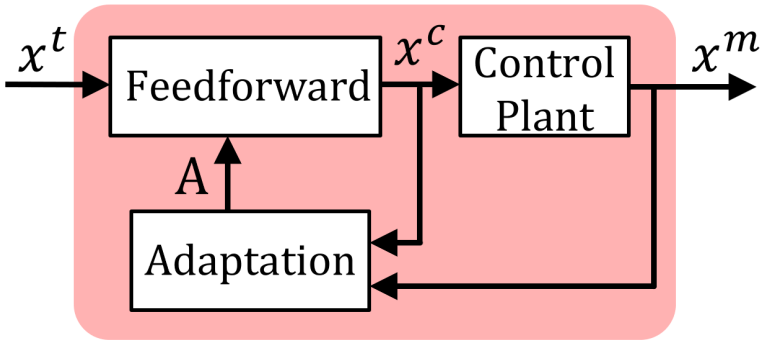


Figure 3.9: Block diagram of adaptive model-based compensation implementation.

The design of the AMBC is based on an adaptive control law based on the study done by Ioannou and Fidan (2006), which is also referenced by Chen et al. (2015) for his controller called *Adaptive Model-Based Tracking Control* (AMB). Ioannou defines the adaptive law as a differential equation of the form:

$$\dot{A} = H(t)\varepsilon \tag{3.37}$$

Where ε corresponds to the estimated error, which depends on the difference between the estimated coefficients A^* and the unknown coefficients A, $H(t)$ is a time-varying gain vector that depends on the measured signal x^m. The unknown coefficients A are obtained using the gradient method, where the goal is to minimize a cost function.

The design is performed for a control plant $G_p(s)$ with three poles and no zeros, as shown in equation (3.37), where a_i are the coefficients of the denominator of the transfer function, with $i \in [1, n]$, and s corresponds to the Laplace variable in the complex space. The coefficient a_0 is known as the DC Gain. Considering the form in which the denominator of the transfer function in equation (3.37) is written, the coefficient a_0 takes values close to 1, which means that for low frequencies, the amplitudes between the commanded and measured signal will be similar, $x^m \approx x^c$. The coefficient a_1 is related to the time delay at low frequencies. On the other hand, coefficients a_2 and a_3 are related to the system dynamics for high frequencies.

$$G_p(s) = \frac{x^m}{x^c} = \frac{1}{a_3 s^3 + a_2 s^2 + a_1 s + a_0} \tag{3.38}$$

Since the AMBC is a feedforward controller, similar to section 3.4.1, the inverse of the control plant is sought to generate the commanded control signal from the target signal, as indicated below:

$$x^c = G_p^{-1}(s)x^t = (a_3 s^3 + a_2 s^2 + a_1 s + a_0)x^t \tag{3.39}$$

The terms s^n in equation (3.39) correspond to temporal derivatives of the input signal x^t. Thus, equation (3.39) can be written in the following parametric form shown in equation (3.40)

$$x^c = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 \end{bmatrix} \begin{bmatrix} x^t & \dot{x}^t & \ddot{x}^t & \dddot{x}^t \end{bmatrix}^T \tag{3.40}$$

The higher time derivatives of x^t are approximately estimated using the backward finite difference method. Furthermore, the transfer function (3.40) must be proper for its implementation

to be realizable. To achieve the previous statement, the signals x^t and x^c are pre-filtered with a Butterworth filter $\Lambda(s)$ of order $n = 4$, as shown in the following expression:

$$\frac{x^c}{\Lambda(s)} = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 \end{bmatrix} \begin{bmatrix} \frac{x^t}{\Lambda(s)} & \frac{\dot{x}^t}{\Lambda(s)} & \frac{\ddot{x}^t}{\Lambda(s)} & \frac{\dddot{x}^t}{\Lambda(s)} \end{bmatrix}^T \tag{3.41}$$

Thus, the parametric model with the filtered signals is defined as follows:

$$z = A^T X_t \tag{3.42}$$

Where z is the filtered signal x^c, $A^T = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 \end{bmatrix}$ is defined as the vector of unknown coefficients and $X_t = \begin{bmatrix} x^t & \dot{x}^t & \ddot{x}^t & \dddot{x}^t \end{bmatrix}^T$ is a vector that contains the signal x^t and its estimated temporal derivatives.

On the other hand, the estimation of z is defined as $\hat{z}$ obtained from the measured signal x^m and the estimated coefficients A^*, as shown below:

$$\hat{z} = A^{*T} X_m \tag{3.43}$$

With the filtered signals z and $\hat{z}$, the estimation error ε is defined as:

$$\varepsilon = \frac{z - \hat{z}}{m_s^2} = \frac{z - A^{*T} X_m}{m_s^2} \tag{3.44}$$

Where $m_s^2 = 1 + X_m^T X_m$ is a signal that normalizes the estimated error, in this way, it is possible to define the instantaneous cost function as:

$$J(A) = \frac{\varepsilon^2 m_s^2}{2} = \frac{(z - A^{*T} X_m)^2}{2m_s^2} \tag{3.45}$$

The algorithm minimizes the cost function $J(A)$. Thus the gradient algorithm is defined as follows:

$$\dot{A} = -\Gamma \nabla J(A) \tag{3.46}$$

Where Γ is a diagonal matrix called the *Adaptive Gain Matrix*, which is associated with the adaptation rate of the parameters a_i. Finally, the gradient of the cost function calculation results in $\nabla J(A) = -\varepsilon X_m$, which is substituted into equation (3.46), yielding the adaptive law shown in equation (3.7), which has a consistent form with equation (3.35).

$$\dot{A} = \Gamma \varepsilon X_m \tag{3.47}$$

Figure 3.10 illustrates the block diagram of the AMBC implementation, providing a more detailed view of the internal process carried out by the *Adaptation* block depicted in Figure 3.9.

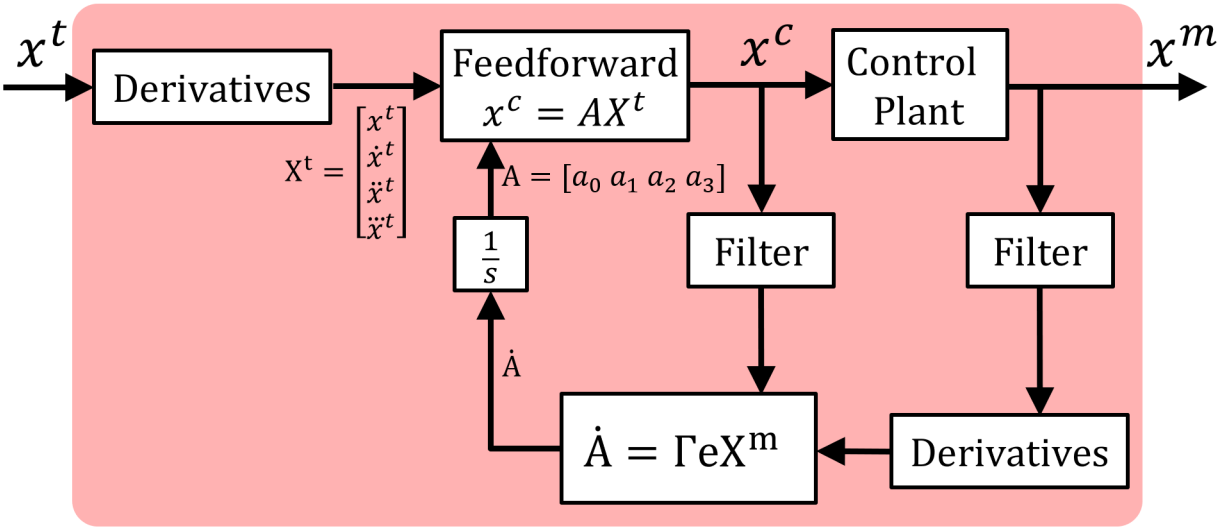


Figure 3.10: Internal processes of adaptive model-based compensation.

3.4.2.2. Design and calibration process

The optimization and calibration of the adaptive gains Γ^* involve a random and iterative process, as shown in the scheme presented in Figure 3.11. The optimization process used is *Particle Swarm Optimization* (Kennedy and Eberhart, 1995).

Before that, the inputs required for the iterative process must be defined, which are detailed below:

1. Obtain the transfer function of the control plant model without the specimen, expressed in the same form as equation (3.29).
2. Invert the transfer function obtained from step 1. The obtained coefficients are used for the integration step where A is obtained from $\dot{A}$.
3. Define the number of generations (N_g), particles (N_p), and simulations (N_s).
4. Define minimum and maximum values for each gain of Γ_s.
5. Define a set of gains for Γ_s for each particle for the first generation of the optimization process.
6. Define the level of uncertainty for the parameters of the RS, compliance, and specimen.

In the iterative process, N_s simulations will be performed for each particle and each generation, considering a set of adaptive gains for each particle. Additionally, each simulation includes uncertainty in the parameters of the RS, compliance, and specimen, as well as noise in the load cells and LVDT. When the N_s simulations of each particle are completed, the performance indicator J_2 is calculated for each simulation, as indicated in equation (3.48).

$$J_{2,i}(\Gamma_s) = \frac{\mathbb{E}[(x_i^t - x_i^m)^2]}{\mathbb{E}[(x_i^t)^2]} \tag{3.48}$$

Where $\mathbb{E}[\cdot]$ corresponds to the expected value operator in a random process. After obtaining the J_2 indicators for each simulation, the average of the N_s simulations carried out for the particle

is calculated, denoted as $R_2(\Gamma_k)$, as indicated in equation 3.49. At the same time, this indicator corresponds to the objective function of the optimization process.

$$R_2(\Gamma_s) = \frac{1}{N}\sum_{i=1}^{N_s} J_{2,i}(\Gamma_k) \tag{3.49}$$

This process is carried out until the values of the gains Γ^* converge to an optimal solution. It is essential to mention that the goal of this optimization process is not to obtain the parameters a_i corresponding to A. This process aims to get the gains of Γ^*, which, as mentioned in the previous section, are related to the adaptation rate of the parameters a_i. In this way, it is ensured that the adaptation of the parameters a_i will maintain good performance in the face of the uncertainties established for the problem. Finally, when the final gains Γ^* for adapting the $AMBC$ from the initial model to different virtual control plants are obtained, the controller is ready for implementation.

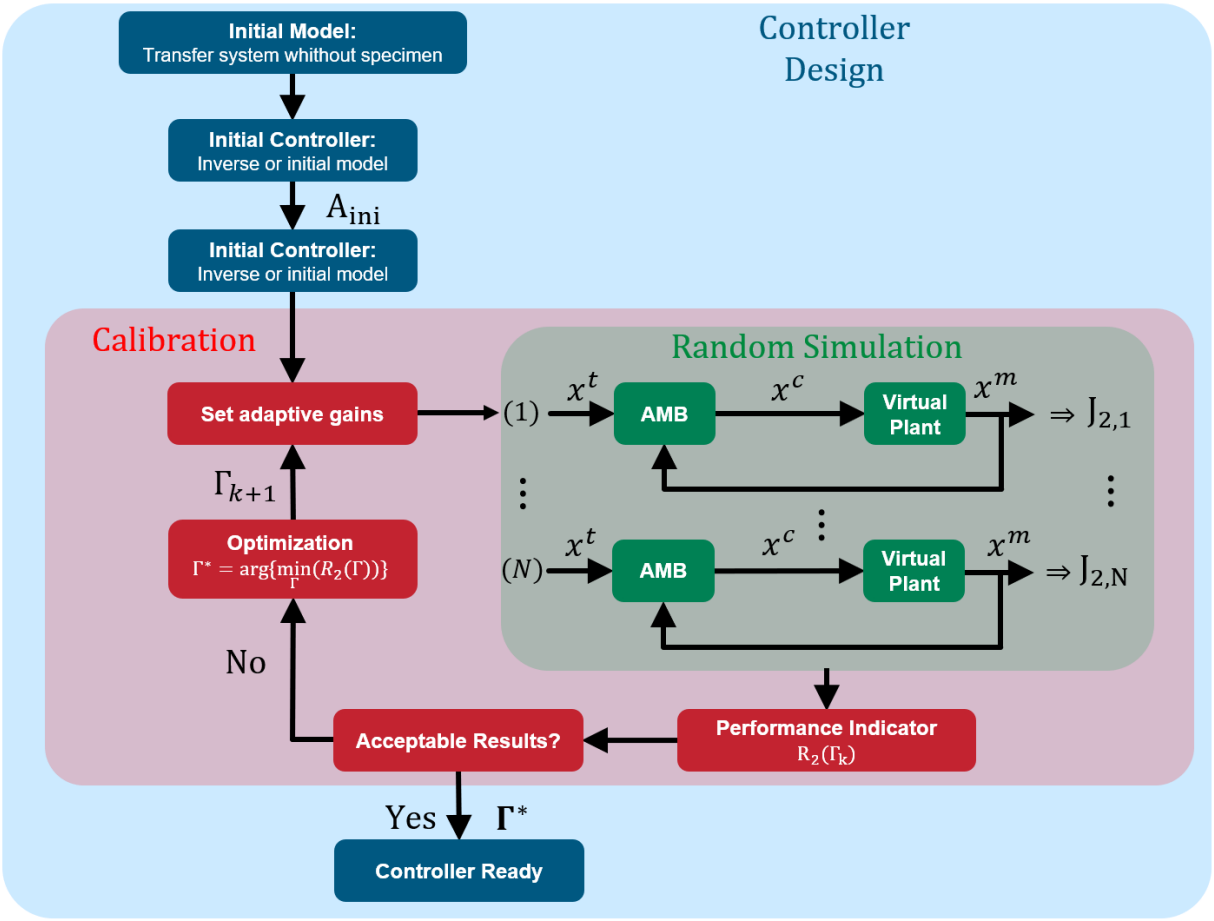


Figure 3.11: Block diagram of calibration and optimization of AMBC controller.

4 Problem formulation

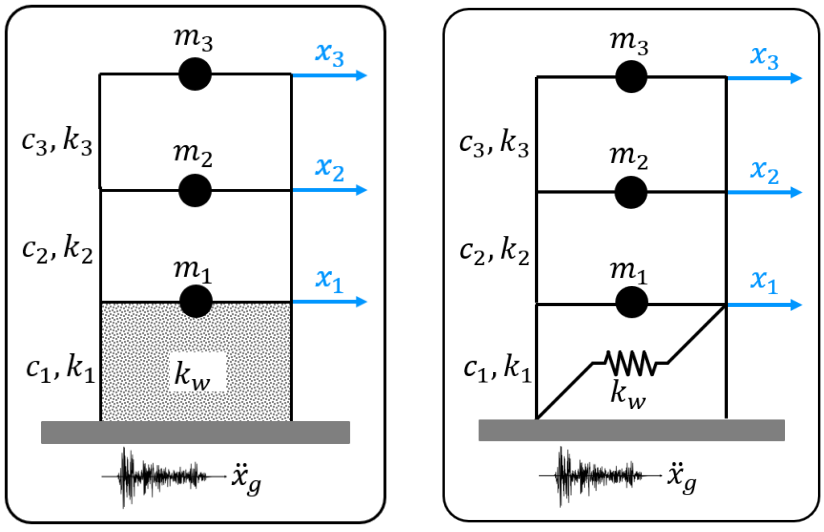
4.1. Reference structure

4.1.1. Model and description

In this study, the reference structure (RS) is a three-story one-bay lumped-mass shear building shown in figure 4.1. It is modeled as a linear time-invariant system, with three lateral degrees of freedom (DOF) on each floor level. A shear wall is mounted between the ground and the first story to add stiffness to the first story DOF and, consequently, to the entire system. The equation of motion (EOM) that represents the dynamics of the RS, defined in a domain $\mathbf{\Omega}$, is shown in equation (4.1). The simplified model of the reference structure is constructed following the guidelines detailed in section 3.1.

$$\mathbf{\Omega}: \quad \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{C}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) = \mathbf{p}(t) \tag{4.1}$$

Where $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$ are the mass, damping, and stiffness matrices, respectively. $\mathbf{x}(t) = \{x_1(t), x_2(t), x_3(t)\}^T \in \mathbb{R}^n$ is the displacement vector relative to the ground. $\dot{\mathbf{x}}(t) \in \mathbb{R}^n$ and $\ddot{\mathbf{x}}(t) \in \mathbb{R}^n$ are the velocity and acceleration vectors obtained from the respective time derivatives from $\mathbf{x}(t)$. $\mathbf{p}(t)$ is the external load vector. For this problem, $\mathbf{p}(t) = -\mathbf{M}\mathbf{\Gamma}\ddot{x}_g$, where $\mathbf{\Gamma} = \{1, 1, 1\}^T$ is the seismic participation vector, and $\ddot{x}_g(t)$ is the ground acceleration. The initial conditions for this problem are $\mathbf{x}(0) = \dot{\mathbf{x}}(0) = \{0, 0, 0\}^T$.



(a) Reference structure simplified model.

(b) Wall representated as an inner spring.

Figure 4.1: Reference structure schemes.

The chosen RS is a simple model of the scaled three-story test structure used in several RTHS previous studies (Dyke et al., 1996, Phillips and Spencer Jr, 2012). The mass matrix $\mathbf{M}$ and stiffness matrix $\mathbf{K}$ are symmetric and have the following form:

$$\mathbf{M} = \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix}; \ \mathbf{K} = \begin{bmatrix} k_1 + k_2 + k_w & -k_2 & 0 \\ -k_2 & k_2 + k_3 & -k_3 \\ 0 & -k_3 & k_3 \end{bmatrix} \tag{4.2}$$

Where m_i and k_i are the mass and stiffness of each story, and k_w is the wall stiffness. Subscript $i = \{1, 2, 3\}$ denotes the story number. The damping matrix $\mathbf{C}$ is obtained employing modal damping with the following damping ratios $\zeta_i = \{5, 5, 5\}[\%]$ for the three modes. Thus, $\mathbf{C}$ matrix is obtained from the calculation shown in equation (4.3), where m_i^* and ω_{ni} the modal mass and natural frequencies, respectively. $\mathbf{\Phi}$ is the mode shape matrix.

$$\mathbf{C} = (\mathbf{\Phi}^T)^{-1} \begin{bmatrix} 2\zeta\omega_{n1}m_1^* & 0 & 0 \\ 0 & 2\zeta\omega_{n2}m_2^* & 0 \\ 0 & 0 & 2\zeta\omega_{n3}m_3^* \end{bmatrix} (\mathbf{\Phi})^{-1} \tag{4.3}$$

The RS is modeled in its *State-space* format to be implemented in Simulink (MATLAB). Following the form described in section 3.1.1, the model is described in the same way as equations (3.11) and (3.12). Mass and stiffness values for each story are shown in table 4.1.

Table 4.1: Reference structure parameters.

Story (DOF)	**Mass** [kg]	**Stiffness** [kN/mm]
1	98.3	11.52
2	98.3	0.684
3	98.3	0.684

4.1.2. Wall considerations

In the model, it is considered that the wall only contributes stiffness k_w to the system. The wall's mass is considered negligible compared to the mass of each floor, and the wall's dimensions are not considered. Moreover, the wall is connected to the ground and the lower zone of the first-floor slab. Thus, it is established that the wall has fixed ends, which implies that deformations (due to bending and shear) will be purely horizontal and are in line with the shear deformation scheme of the building. Additionally, although the detailed design of the wall is beyond the scope of this study, it is important to mention that the wall's stiffness depends on its resistance to bending and shear and its dimensions. The aspect ratio between its height and width, $\frac{Hw}{Bw}$, determines whether the wall deformation is governed by bending or shear. The same ratio also determines whether the wall stiffness is predominantly influenced by its bending stiffness or its shear stiffness (Fares, 2018). Figure 4.2 illustrates the deformation of the wall.

These previous statements are made because this research aims to implement the AMBC in an RTHS test with a rigid specimen, introducing uncertainty in the experimental parameters. Thus, the sole impact of adding the wall is to increase the stiffness of the first degree of freedom of the building. Moreover, a ratio between the stiffness of the wall and the stiffness k_1 of $\frac{k_w}{k_1} = 20$

is considered for the design of the wall stiffness, which is added to the DOF 1 in the stiffness matrix **K**. The value of the stiffness ratio mentioned is chosen based on the actuator's physical limitations and the compliance spring's stiffness constant defined in section 4.3.6. Thus, the value of the stiffness contributed by the wall to DOF 1 of the RS is shown in Table 4.2.

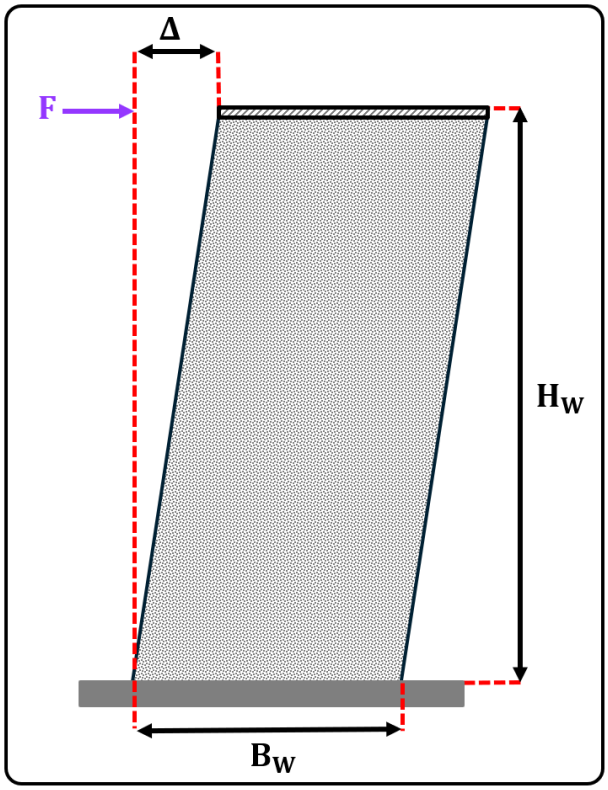


Figure 4.2: Wall deformation (*illustrative scheme*).

Table 4.2: Contribution of wall stiffness.

Stiffness	**Value** [kN/mm]
DOF 1 (before adding the wall)	1.2
Wall	10.32
DOF 1 (after adding the wall)	11.52

4.1.3. Wall effects

To observe, before the experiments, the impact of adding the wall to the RS, the vibration modes are calculated for both cases. The obtained fundamental periods and frequencies are shown in Tables 4.3 and 4.4. An increase in all the fundamental frequencies is observed when the wall is added, indicating that the increase in stiffness in the DOF affects all the vibration modes. Although the frequencies of the third mode exceed the typical frequencies of RTHS tests, it does not prevent displacements from being obtained in the DOFs of the RS.

Table 4.3: $\mathbf{T}_n$ and $\mathbf{f}_n$ without the wall.

Mode	$\mathbf{T}_n$ $[s]$	$\mathbf{f}_n$ $[Hz]$
1	0.183	5.457
2	0.063	15.81
3	0.042	23.63

Table 4.4: $\mathbf{T}_n$ and $\mathbf{f}_n$ with the wall.

Mode	$\mathbf{T}_n$ $[s]$	$\mathbf{f}_n$ $[Hz]$
1	0.125	8.019
2	0.047	21.27
3	0.018	54.59

Figure 4.3 shows the vibration modes for the two cases described. It can be observed that the first floor with the wall is hardly excited by the frequencies of the first two modes, unlike the case without the wall, where the first floor is excited at its three fundamental frequencies. Additionally, for the third mode in the case with the wall, it can be observed that the second and third floors are very lightly excited at the frequency of $\mathbf{f}_3 = 54.59$ $[Hz]$, due to the significant difference in stiffness between the first floor and the other two floors.

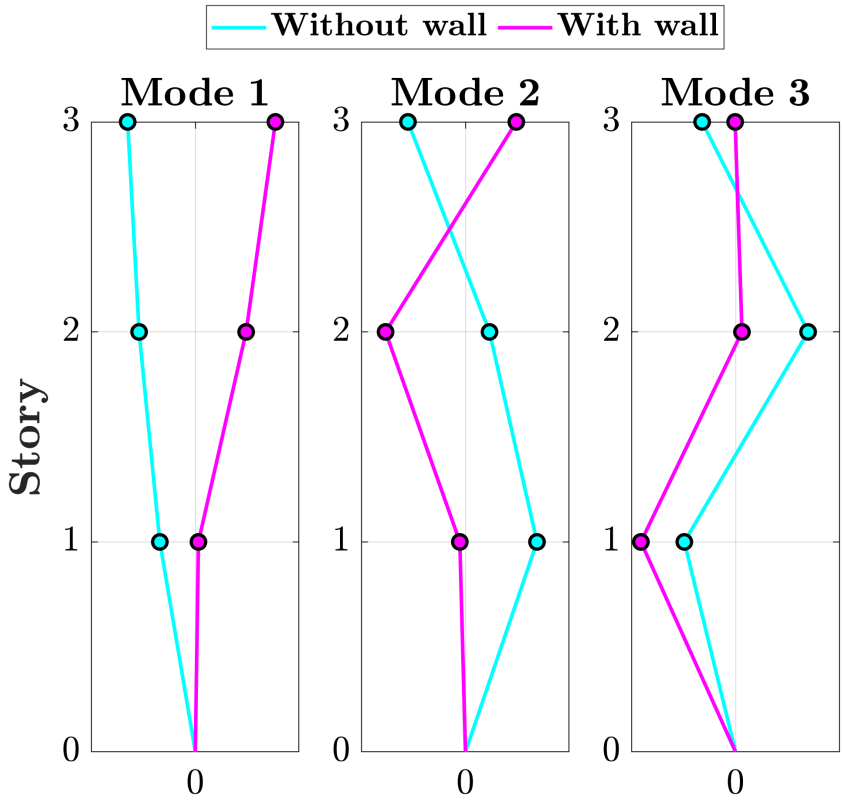


Figure 4.3: Vibration modes.

4.2. Force-based substructure

The method known as substructuring, shown in section 3.2, is used to divide the domain $\mathbf{\Omega}$ of the reference structure into smaller partitions that are coupled and can be solved independently. However, the coupling between the divisions must be included in the motion equations as boundary conditions. In this case, the RS is divided into two substructures, denominated Numerical (NS)

and Experimental (ES) substructures, with domains $\mathbf{\Omega}^N$ and $\mathbf{\Omega}^E$, respectively. Thus, the RS can be defined now as the union of the subdomains generated $\mathbf{\Omega} = \mathbf{\Omega}^N \cup \mathbf{\Omega}^E$. Furthermore, the response of the reference structure can be obtained by combining the responses obtained from solving the coupled motion equations of each substructure. The coupled numerical and experimental equations of motion are shown in equations (4.4) and (4.5), respectively. Additionally, figure 4.4 shows the closed hybrid loop used in this study, where the coupling force $\mathbf{g}_b^N$ at the boundary DOF is obtained from the numerical substructure, which is imposed on the specimen through a transfer system. The position and velocity of the specimen are measured and fed back as input to the numerical substructure, closing the hybrid loop.

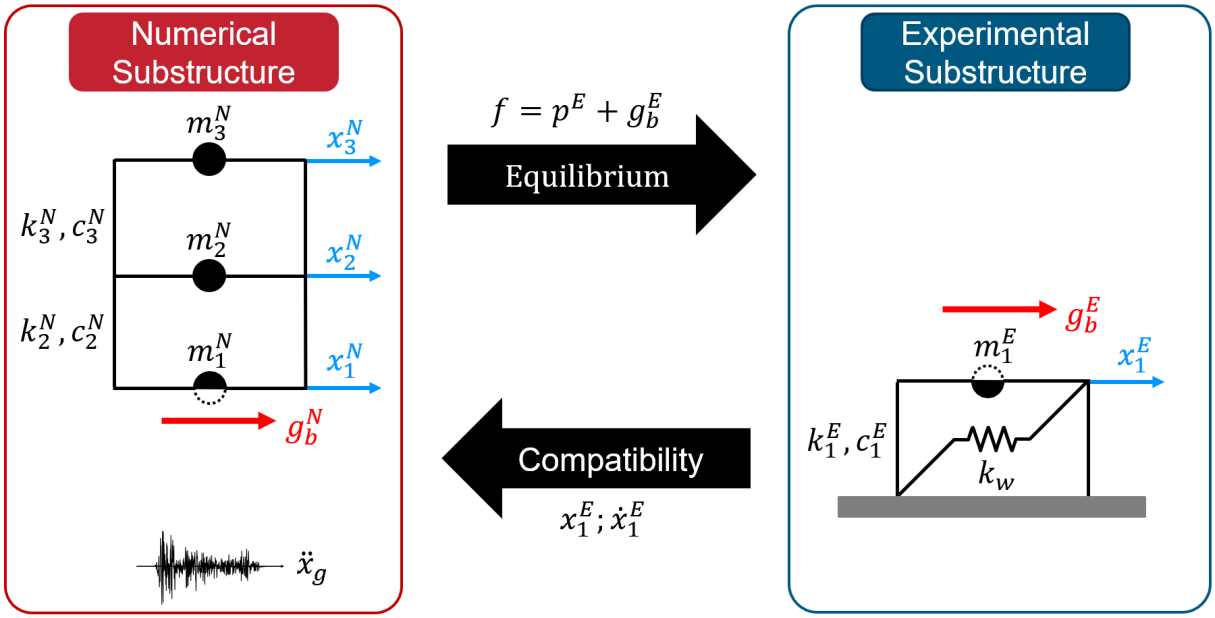


Figure 4.4: Hybrid loop for a force-based RTHS test.

$$\mathbf{\Omega}^N : \mathbf{M}^N\ddot{\mathbf{x}}^N + \mathbf{C}^N\dot{\mathbf{x}}^N + \mathbf{K}^N\mathbf{x}^N = \mathbf{p}^N + \mathbf{g}^N \tag{4.4}$$

$$\mathbf{\Omega}^E : \mathbf{M}^E\ddot{\mathbf{x}}^E + \mathbf{C}^E\dot{\mathbf{x}}^E + \mathbf{K}^E\mathbf{x}^E = \mathbf{p}^E + \mathbf{g}^E \tag{4.5}$$

Superscripts N and E are used to identify the numerical and experimental substructures. The vectors $\mathbf{g}^N = \{\mathbf{g}_i^N, \mathbf{g}_b^N\}^T$ and $\mathbf{g}^E = \{\mathbf{g}_i^E, \mathbf{g}_b^E\}^T$ are the coupling force vectors between both domains, applied to their respective substructure, where the subscripts "i" and "b" indicate interior and boundary DOF, respectively.

Assuming that the subdomains are only coupled on the boundary DOF 1, then the internal coupling forces are equaled to zero $\mathbf{g}_i^N = \mathbf{0}_i^N$ and $\mathbf{g}_i^E = \mathbf{0}_i^E$. With the previous assumption, the resulting boundary conditions for this problem are displacement compatibility and force equilibrium, shown in Eq.(4.6).

$$\mathbf{x}_b^N = \mathbf{x}_b^E \; ; \quad \mathbf{g}_b^N + \mathbf{g}_b^E = \mathbf{0}_b \tag{4.6}$$

The reference EOM in Eq.(4.1) is separated into numerical and experimental components as in Eq.(4.7). The parameters and the DOF are associated with each partition using superscript N and E, and story subscript i.

$$\begin{bmatrix} m_1^E & 0 & 0 \\ 0 & m_2^N & 0 \\ 0 & 0 & m_3^N \end{bmatrix} \begin{Bmatrix} \ddot{x}_1^E \\ \ddot{x}_2^N \\ \ddot{x}_3^N \end{Bmatrix} + \begin{bmatrix} c_{11}^E & c_{12}^N & c_{13}^N \\ c_{21}^N & c_{22}^N & c_{23}^N \\ c_{31}^N & c_{22}^N & c_{33}^N \end{bmatrix} \begin{Bmatrix} \dot{x}_1^E \\ \dot{x}_2^N \\ \dot{x}_3^N \end{Bmatrix} + \\ \begin{bmatrix} k_1^E + k_2^N & -k_2^N & 0 \\ -k_2^N & k_2^N + k_3^N & -k_3^N \\ 0 & -k_3^N & k_3^N \end{bmatrix} \begin{Bmatrix} x_1^E \\ x_2^N \\ x_3^N \end{Bmatrix} = - \begin{bmatrix} m_1^E & 0 & 0 \\ 0 & m_2^N & 0 \\ 0 & 0 & m_3^N \end{bmatrix} \begin{Bmatrix} 1 \\ 1 \\ 1 \end{Bmatrix} \ddot{x}_g \tag{4.7}$$

From Eq.(4.7), the numerical substructure is separated from the experimental one since they will be solved separately, resulting in the following two EOM:

$$\begin{bmatrix} m_2^N & 0 \\ 0 & m_3^N \end{bmatrix} \begin{Bmatrix} \ddot{x}_2^N \\ \ddot{x}_3^N \end{Bmatrix} + \begin{bmatrix} c_{22}^N & c_{23}^N \\ c_{22}^N & c_{33}^N \end{bmatrix} \begin{Bmatrix} \dot{x}_2^N \\ \dot{x}_3^N \end{Bmatrix} + \begin{bmatrix} k_2^N + k_3^N & -k_3^N \\ -k_3^N & k_3^N \end{bmatrix} \begin{Bmatrix} x_2^N \\ x_3^N \end{Bmatrix} = \\ \underbrace{- \begin{bmatrix} m_2^N & 0 \\ 0 & m_3^N \end{bmatrix} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \ddot{x}_g}_{\mathbf{p}^N} \underbrace{- \begin{Bmatrix} c_{21}^N \\ c_{31}^N \end{Bmatrix} \dot{x}_1^E - \begin{Bmatrix} -k_2^N \\ 0 \end{Bmatrix} x_1^E}_{\mathbf{g}_b^N} \tag{4.8}$$

$$m_1^E \ddot{x}_1^E + c_1^E \dot{x}_1^E + k_1^E x_1^E = \underbrace{-m_1^E \ddot{x}_g}_{\mathbf{p}^E} + \underbrace{c_2^N (\dot{x}_2^N - \dot{x}_1^E) + k_2^N (x_2^N - x_1^E)}_{\mathbf{g}_b^E} \tag{4.9}$$

The motion equations shown are consistent with the force-based substructuring performed by Shao and Reinhorn (2012). However, only actuators will be used as the transfer system in this case. Additionally, each equation denotes contributions from external forces $\mathbf{p}$ and from coupling forces $\mathbf{g_b}$.

Figure 4.5 shows the block diagram of the substructuring implementation in Simulink, following the scheme shown in Figure 4.4, where it can be observed that the target force f^t is sent from the numerical substructure to the experimental substructure. It also shows that the measured displacement and velocity from the experimental substructure are sent to the numerical substructure to close the hybrid simulation loop.

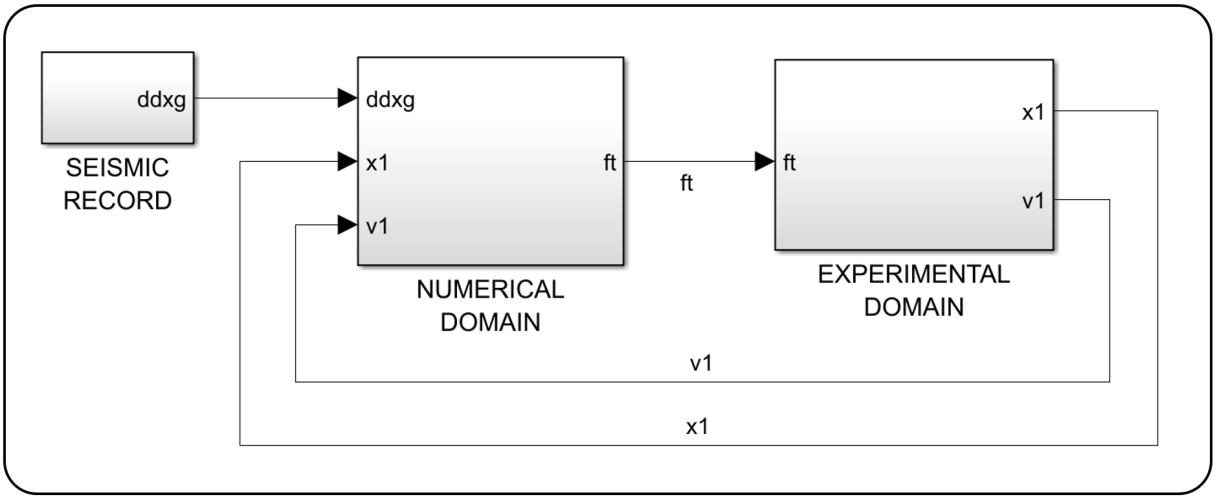


Figure 4.5: Simulink block diagram of substructure implementation

4.3. Experimental Domain

The *Experimental Domain* represents the components of the experimental part in an RTHS test. It consists of a transfer system, the compliance spring, and the specimen, as shown in Figure 4.6. LVDT and load measurement sensors also belong to the Experimental Domain, but they will be included in the scheme starting from Chapter 5.

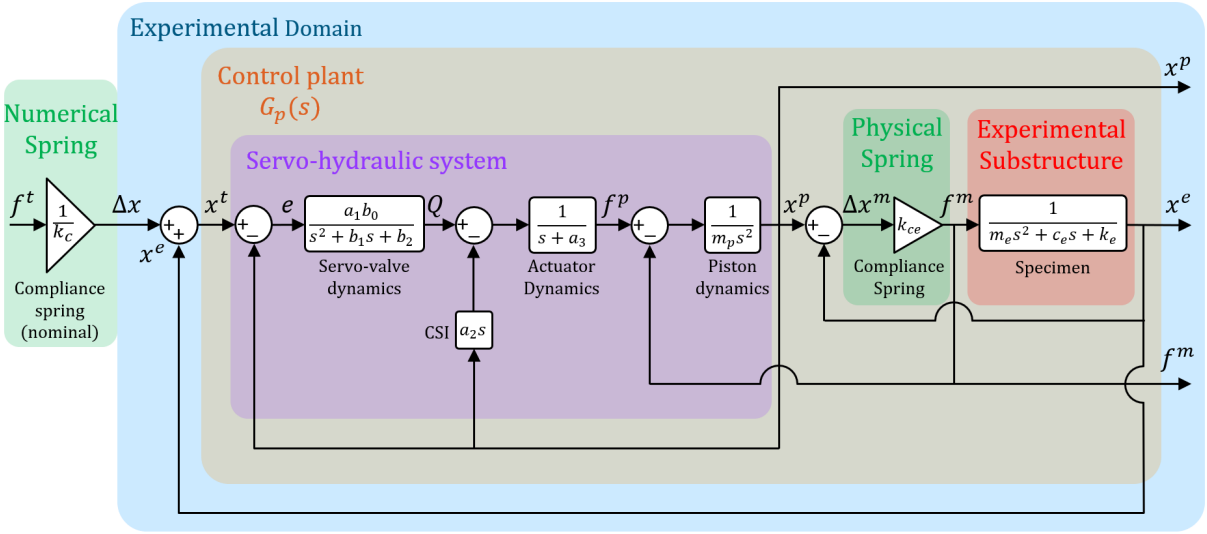


Figure 4.6: Block diagram of the control plant.

Where $a_1 b_0$, a_2, a_3, b_1, and b_2 are parameters of the servo-valve and the actuator. These parameters are obtained after simplifying the actuator model proposed by Carrion and Spencer Jr (2007), as explained in Chapter 3.3. m_p is the mass of the piston, k_c and k_{ce} correspond to the nominal and uncertain stiffness of the compliance spring, respectively. m_e, c_e, and k_e represent the specimen's mass, damping, and stiffness.

In the following sections, the modeling, transfer functions, and implementation of each component of the Experimental Domain are detailed.

4.3.1. Transfer system

A transfer system is modeled and implemented into the experimental domain as shown in Figure 4.6 to impose the target force from the numerical substructure onto the specimen. The model focuses on three stages of the system: the servo valve, the actuator, and the interaction with the specimen. These components are modeled as linear time-invariant systems (LTI), which allows them to be represented as transfer functions connected in series in the Laplace domain. The transfer system presented corresponds to the simplified model proposed by Phillips and Spencer Jr (2012) with some modifications implemented to adapt it to the force-based scheme.

From Figure 4.7, the input corresponds to the target displacement x^t, and the output is the measured displacement of the piston x^p. The input f^m corresponds to the measured force of the compliance spring, which is considered a disturbance signal in the model and represents the force equilibrium between the piston and the compliance spring.

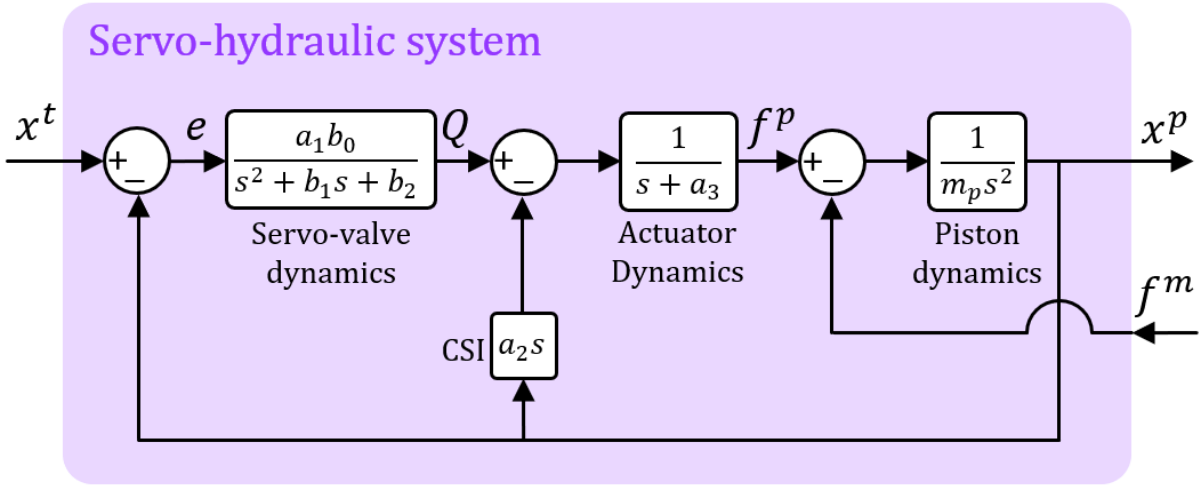


Figure 4.7: Block diagram of the control plant.

The parameter values of the hydraulic servo system are obtained from a *Genetic Algorithm* optimization process conducted in the study by Qian et al. (2014) and are presented in Table 4.6.

Table 4.5: Transfer system parameter values.

Parameter	Value	Unit	Parameter	Value	Unit
$a_1\beta_0$	$2.13 \cdot 10^{13}$	$\frac{\text{Pa·m}}{\text{s}}$	β_2	10^5	$\frac{1}{\text{s}}$
a_2	$4.23 \cdot 10^6$	$\text{Pa} \cdot \text{m}$	m_p	1	kg
a_3	3.3	$\frac{1}{\text{s}}$	c_p	0	$\frac{\text{N·s}}{\text{m}}$
β_1	425	–	k_p	0	$\frac{\text{N}}{\text{m}}$

4.3.2. Initial model

It corresponds to the transfer function of the control plant without the specimen or the compliance spring, i.e., only includes the components of the actuator. Due to this, it is considered to represent an "*initial model*" of the test. It is denoted as $G_{ini}(s)$. It includes the blocks *Servo-valve dynamics*, *Actuator dynamics*, and *CSI* (Control Structure Interaction), which takes the target displacement x^t as input and outputs the measured displacement of the piston x^p. This transfer function is primarily used for designing the AMBC controller and selecting the compliance spring for this study. Its expression is as follows:

$$G_{ini}(s) = \frac{x^m(s)}{x^t(s)} \quad \Rightarrow \quad G_{ini}(s) = \frac{B_0}{A_3 \cdot s^3 + A_2 \cdot s^2 + A_1 \cdot s + A_0} \tag{4.10}$$

Where:

$$\begin{aligned} B_0 &= a_1 b_0 \\ A_3 &= a_2 + k_e \\ A_2 &= a_2 b_1 + k_e(a_3 + b_1) \\ A_1 &= a_2 b_2 + k_e(a_3 b_1 + b_2) \\ A_0 &= a_1 b_0 + a_3 b_2 k_e \end{aligned}$$

This transfer function can be obtained experimentally by performing system identification of the actuator without the specimen. However, to maintain the context of the proposed virtual RTHS tests in this study, the parametric transfer function from the benchmark problem (Silva et al., 2020) will be used, but considering a dummy specimen with low-magnitude physical properties, where $m_e = c_e = 0$ and $k_e = 1\frac{N}{m}$ (Gálmez and Fermandois, 2022). Finally, by substituting the values from Table 4.5, we obtain the transfer function of the initial model:

$$G_{ini}(s) = \frac{5.035 \cdot 10^6}{s^3 + 425s^2 + 10^5 s + 5.035 \cdot 10^6} \tag{4.11}$$

4.3.3. Actuator with specimen

It corresponds to the transfer function of the system composed of the servo-hydraulic system with the specimen attached to the piston. It is denoted as $G_e(s)$. This model represents the laboratory test, as it serves as the basis for designing controllers to reduce the delays between the commanded and measured signals. The transfer function is as follows:

$$G_e(s) = \frac{x^m(s)}{x^t(s)} \quad \Rightarrow \quad G_e(s) = \frac{B_0}{A_5 \cdot s^5 + A_4 \cdot s^4 + A_3 \cdot s^3 + A_2 \cdot s^2 + A_1 \cdot s + A_0} \tag{4.12}$$

Where:

$$
\begin{aligned}
B_0 &= a_1 b_0 \\
A_5 &= m_p \\
A_4 &= m_p(a_3 + b_1) \\
A_3 &= a_2 \cdot b_1 + a_3(b_2 \cdot m_p + k_e) + b_1 \cdot k_e \\
A_2 &= a_2 \cdot b_1 + k_e(a_3 + b_1) \\
A_1 &= a_2 \cdot b_2 + k_e(a_3 \cdot b_1 + b_2) \\
A_0 &= a_1 b_0 + a_3 \cdot b_2 \cdot k_e
\end{aligned}
$$

Experimental parameter values are shown in Table 4.6.

Table 4.6: Experimental parameters values.

Parameter	Value	Unit
m_e	98.3	kg
c_e	3004.8	$\frac{\text{N·s}}{\text{m}}$
k_e	$10.836 \cdot 10^6$	$\frac{\text{N}}{\text{m}}$

Finally, the transfer function of the transfer system combined with the specimen is obtained, as shown below:

$$
G_e(s) = \frac{2.167 \cdot 10^{11}}{s^5 + 458.9s^4 + 2.678 \cdot 10^5 s^3 + 6.893 \cdot 10^7 s^2 + 1.549 \cdot 10^{10} s + 2.531 \cdot 10^{11}} \tag{4.13}
$$

4.3.4. Control plant

The control plant considers the blocks of the servo-hydraulic system, the specimen, and the compliance spring, as shown in Figure 4.6. This representation is referred to as $G_p(s)$. Due to the compliance spring placed between the piston and the specimen, a block with the piston dynamics is added to obtain the displacement of the piston connected to the compliance spring, modifying the control plants proposed by Silva et al. (2020) and Sivaselvan et al. (2008).

Algebraically reducing the blocks in the Laplace domain, a transfer function with seven poles and two zeros is obtained, as shown in Equation (4.14).

$$
G_p(s) = \frac{x^m(s)}{x^t(s)} \quad \Rightarrow \quad G_p(s) = \frac{B_2 \cdot s^2 + B_1 \cdot s + B_0}{A_7 \cdot s^7 + A_6 \cdot s^6 + A_5 \cdot s^5 + A_4 \cdot s^4 + A_3 \cdot s^3 + A_2 \cdot s^2 + A_1 \cdot s + A_0} \tag{4.14}
$$

Where:

$$
\begin{aligned}
B_2 &= a_1 b_0 \cdot m_e \\
B_1 &= a_1 b_0 \cdot c_e \\
B_0 &= a_1 b_0 \cdot k_e \\
A_7 &= m_e \cdot m_p \\
A_6 &= (a_3 \cdot m_e + b_1 \cdot m_e + c_e) m_p \\
A_5 &= a_2 \cdot m_e + a_3 \cdot m_p (b_1 \cdot m_e + c_e) + b_1 \cdot c_e \cdot m_p + b_2 \cdot m_e \cdot m_p + k_c (m_e + m_p) + k_e \cdot m_p \\
A_4 &= a_2 (b_1 \cdot m_e + c_e) + a_3 (b_1 \cdot c_e \cdot m_p + b_2 \cdot m_e \cdot m_p + k_c (m_e + m_p) + k_e \cdot m_p) + \\
&\quad b_1 (k_c (m_e + m_p) + k_e \cdot m_p) + (b_2 \cdot m_p + k_c) c_e \\
A_3 &= a_2 (b_1 \cdot c_e + b_2 \cdot m_e + k_c + k_e) + a_3 (b_1 (k_c (m_e + m_p) + k_e \cdot m_p) + (b_2 \cdot m_p + k_c) c_e) + \\
&\quad b_1 \cdot c_e \cdot k_c + b_2 (k_c (m_e + m_p) + k_e m_p) + k_c k_e \\
A_2 &= a_1 b_0 \cdot m_e + a_2 (b_1 (k_c + k_e) + b_2 \cdot c_e) + a_3 (b_1 \cdot c_e \cdot k_c + b_2 (k_c (m_e + m_p) + k_e \cdot m_p) + k_c \cdot k_e) + \\
&\quad (b_1 \cdot k_e + b_2 \cdot c_e) k_c \\
A_1 &= a_1 b_0 \cdot c_e + a_2 \cdot b_2 (k_c + k_e) + (a_3 (b_1 \cdot k_e + b_2 \cdot c_e) + b_2 \cdot k_e) k_c \\
A_0 &= a_1 b_0 (k_c + k_e) + a_3 \cdot b_2 \cdot k_c \cdot k_e
\end{aligned}
$$

Upon substituting the experimental values, the following $G_p(s)$ expression is obtained:

$$
G_p(s) = \frac{2.13 \cdot 10^{13} s^2 + 6.511 \cdot 10^{14} s + 2.465 \cdot 10^{18}}{s^7 + 458.9 s^6 + 5.002 \cdot 10^6 s^5 + 2.229 \cdot 10^9 s^4 + 1.101 \cdot 10^{12} s^3 + 2.698 \cdot 10^{14} s^2 + 5.567 \cdot 10^{16} s + 2.458 \cdot 10^{18}} \tag{4.15}
$$

Figure 4.8 depicts the pole-zero diagram of $G_p(s)$ in the complex plane. It can be observed that two poles and zeros are cancellable. Thus, it is possible to reduce Equation (4.15) to a transfer function with five poles and no zeros, as illustrated in equation (4.16).

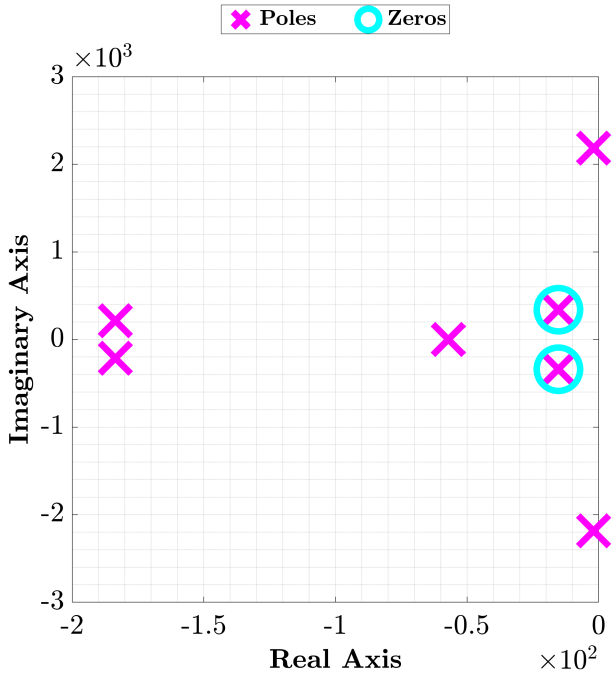


Figure 4.8: Poles and zeros of $G_p(s)$.

$$G_p(s) = \frac{2.13 \cdot 10^{13}}{s^5 + 428.2s^4 + 4.874 \cdot 10^6 s^3 + 2.03 \cdot 10^9 s^2 + 4.782 \cdot 10^{11} s + 2.16 \cdot 10^{13}} \tag{4.16}$$

To verify that the cancellation of poles and zeros does not affect the dynamics of the control plant, the Bode diagram is constructed for both the original and reduced plants, as shown in Figure 4.9. In both cases, there are no differences in the frequency range of interest, $[0, 20]$ Hz. Therefore, it is possible to use $G_p(s)$ in its reduced form going forward.

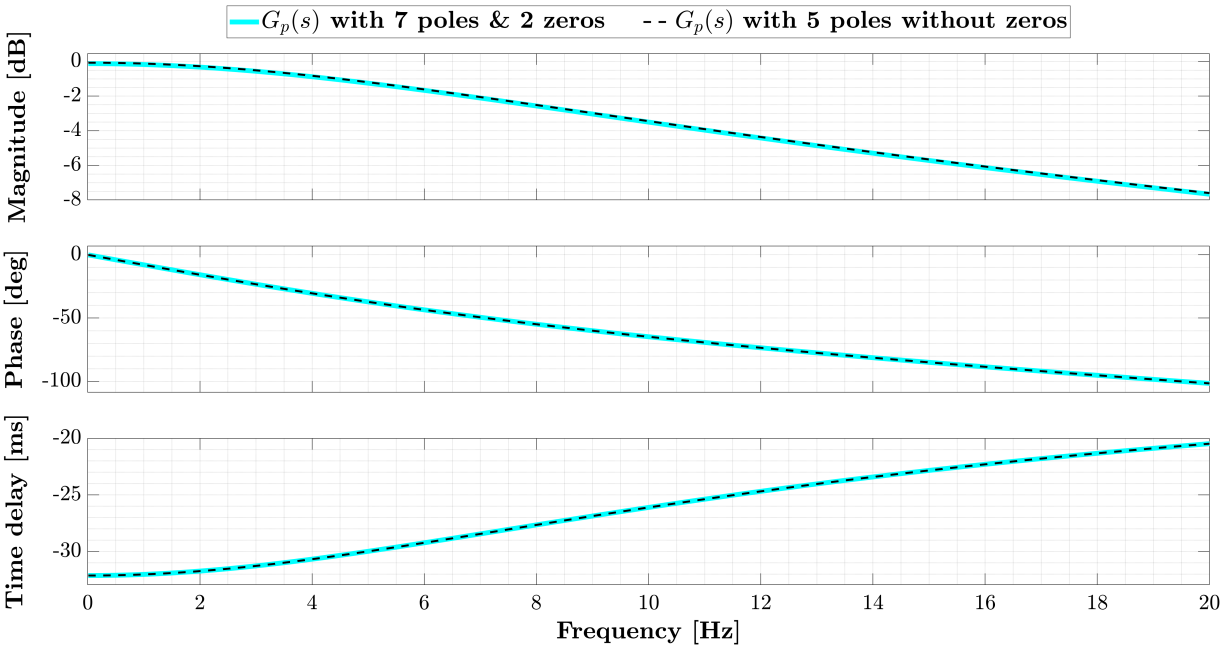


Figure 4.9: Bode diagram of $G_p(s)$ original and reduced.

4.3.5. Estimated control plant

Virtual system identification is performed since the AMBC controller originates from a design considering a third-order transfer function with no zeros (Gálmez and Fermandois, 2022). This identification process is carried out in Matlab 2022b to reduce the order of the control plant transfer function, $G_p(s)$, to a third-order function with no zeros. Equation (4.17) shows the resulting transfer function.

$$G_{est}(s) = \frac{x^m(s)}{x^t(s)} \quad \Rightarrow \quad G_{est}(s) = \frac{B_0}{A_3 \cdot s^3 + A_2 \cdot s^2 + A_1 \cdot s + A_0} \tag{4.17}$$

The identification is carried out using the *tfest*$(\cdot)$ function, which requires the input and response signals of the system. The input signal is generated using the *chirp*$(\cdot)$ function in Matlab, which produces a cosine signal with a frequency sweep shown in Figure 4.10 considering the input parameters from Table 4.7.

Table 4.7: System identification inputs.

Parameter	Value	Unit	Parameter	Value	Unit
Initial frequency	0.1	Hz	Sampling frequency	8,192	Hz
Final frequency	20	Hz	Simulation time	5	s

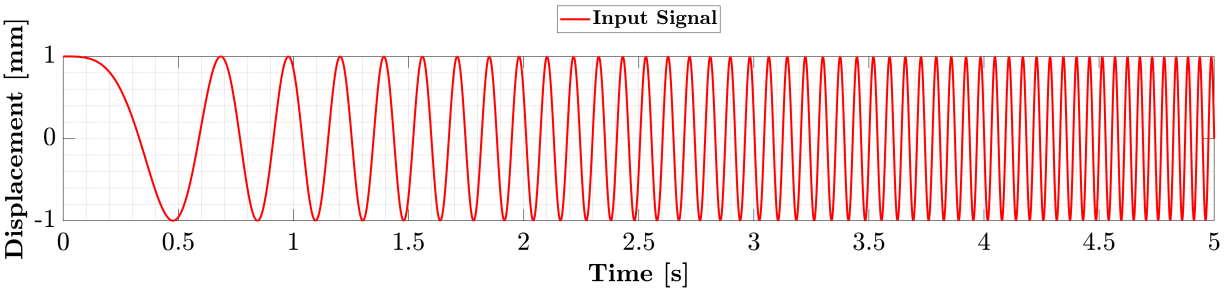


Figure 4.10: Chirp input signal.

Using the *lsim*(·) command, the response of the system $G_p(s)$ considering the *chirp*(·) as the input is obtained. Finally, the *tfest*(·) command is utilized with the input-output data of the system, along with the number of poles (3) and zeros (0), to obtain the estimated transfer function of the control plant:

$$G_{est}(s) = \frac{4.497 \cdot 10^6}{s^3 + 422.2s^2 + 1.007 \cdot 10^5 s + 4.561 \cdot 10^6} \tag{4.18}$$

The fit of the estimated control plant $G_{est}(s)$ concerning $G_p(s)$ is 99.98%, with a normalized root mean square error (NRMSE) of 0.02% and a mean square error (MSE) of 6.22×10^{-9}. To verify the obtained estimation compared to the other plants, the Bode diagram and pole-zero diagram are presented in Figures 4.11 and 4.12, respectively.

In the Bode diagram shown in Figure 4.11, the initial plant $G_{ini}(s)$ is plotted first as the starting point for the analysis, corresponding to the actuator dynamics without the specimen or compliance spring. Secondly, $G_e(s)$ is plotted, representing the actuator system combined with the specimen. It can be observed that the graph deviates from the initial plant. The magnitude decreases, indicating that the system is less sensitive in the frequency range under study. It is also noticeable that the time delay increases for low frequencies. Then, the complete control plant $G_p(s)$ is plotted, showing an improvement in the response, which is very close to the initial plant. Finally, the estimation with three poles and no zeros $G_{est}(s)$ exhibits approximately the same dynamics as the complete control plant $G_p(s)$, validated for the frequency range under study, the estimation $G_{est}(s)$ can be used as the control plant.

The same analysis as the previous paragraph is depicted in the pole-zero diagram shown in Figure 4.12, where, in the end, the zeros of the complete control plant $G_p(s)$ and the estimated plant $G_{est}(s)$ are approximately overlapping, verifying the information provided by the Bode diagrams in Figure 4.11.

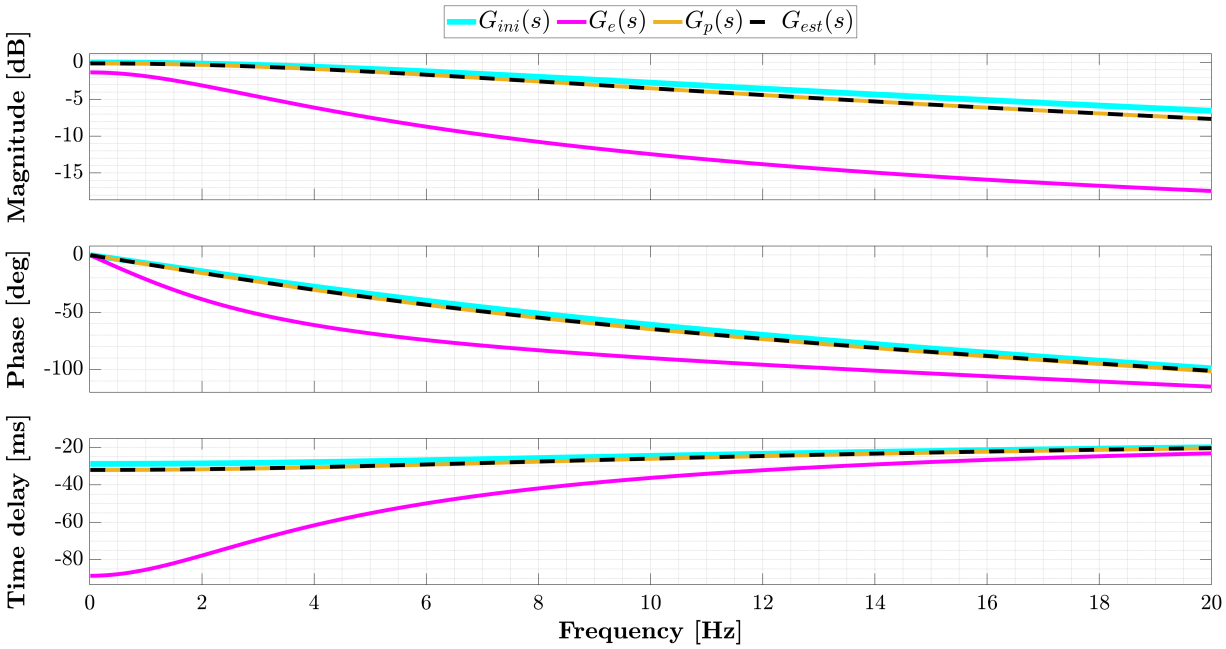


Figure 4.11: Bode plot of described plants.

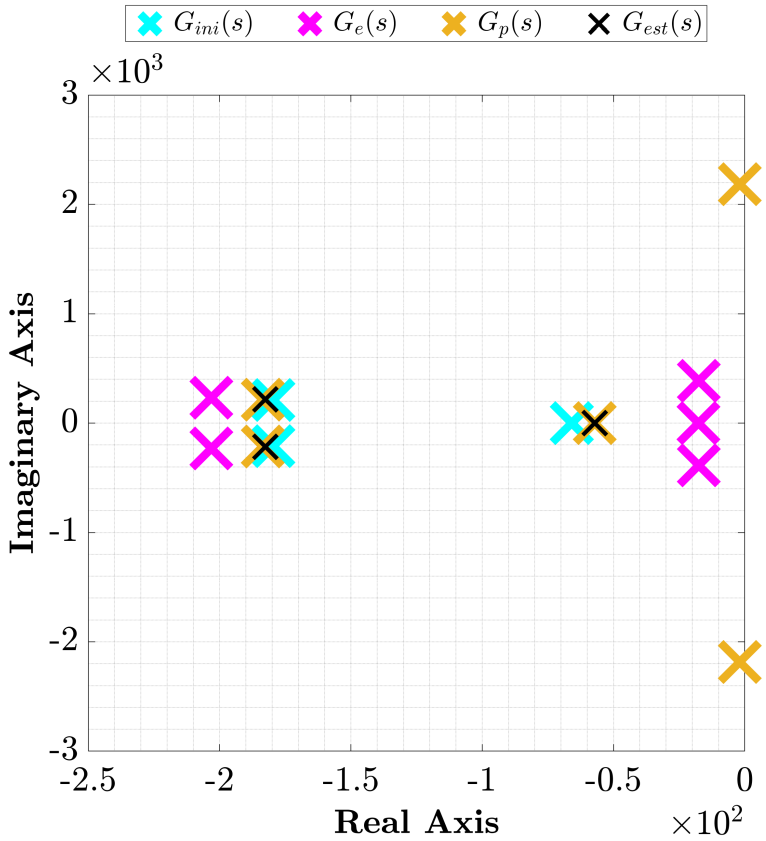


Figure 4.12: Poles and zeros of described plants.

4.3.6. Compliance spring implementation

As previously mentioned, the actuator is a nonlinear system with a stiff oil column designed to be operated based on displacement control (Zhao et al., 2003). However, since the specimen corresponding to the first floor has high rigidity due to the wall, the measured displacements are expected to have a short order of magnitude. Noise could contaminate the measured displacement signal, causing errors that generate large values and variations in the restorative force, causing the boundary conditions not to be satisfied in the integration time, and producing drag errors that can lead the test to numerical instability. In addition, given the rigid nature of the specimen, there are problems in the interaction between the piston and the specimen due to the connection and friction between the elements that compose it. One option for the aforementioned difficulties is to implement a force control methodology. However, similar problems arise as those already mentioned.

As a solution, it is chosen to incorporate a compliance spring between the piston and the specimen (Sivaselvan et al., 2008) shown in figure 4.13, which primarily adds flexibility to the system and allows the actuator to be controlled based on displacements. At the same time, the simulation is resolved based on a force scheme.

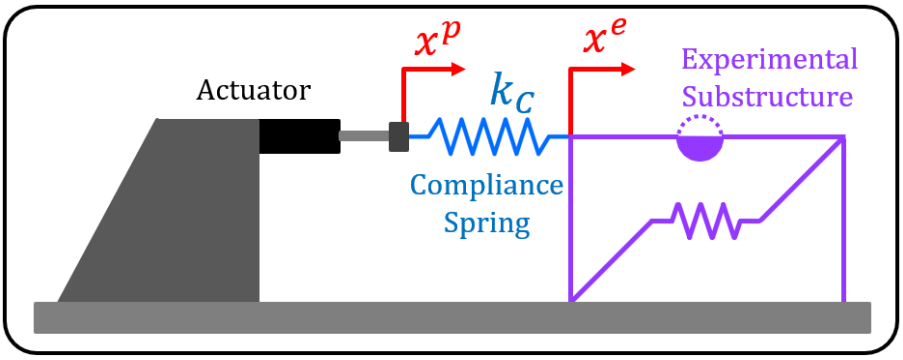


Figure 4.13: Implementation of the compliance spring.

Figure 4.14 presents the block diagram of the control plant focused on incorporating the compliance spring. The equations framed with a dashed line and red arrows represent the theory mentioned in this paragraph. The input signal corresponds to the target force f^t, which is divided by the nominal value of the compliance spring to convert the force signal into displacement. This is achieved by using Hooke's Law to calculate the spring force based on its deformation, as shown in Equation (4.19), where F_s is the restitutive force, K is the spring stiffness constant, and $\Delta x(t) = x^t - x^e$ is the relative length between the ends of the spring for a given time. The subtraction of x^t with the measured displacement of the piston x^p represents the internal feedback of the actuator. After passing through the actuator, the displacements of the piston x^p and the experimental substructure x^e are measured, which represent the ends of the spring. By subtracting both signals, Δx^m is obtained, which is then multiplied by the experimental value of the compliance spring after applying uncertainty to its value, meaning that the nominal and experimental values of the compliance spring will not be equal, i.e., $k_c \neq k_{ce}$.

$$F_s = K\Delta x(t) \quad \Rightarrow \quad f^t = k_c(x^t - x^e) \quad \Rightarrow \quad x^t = \frac{f^t}{k_c} + x^e \tag{4.19}$$

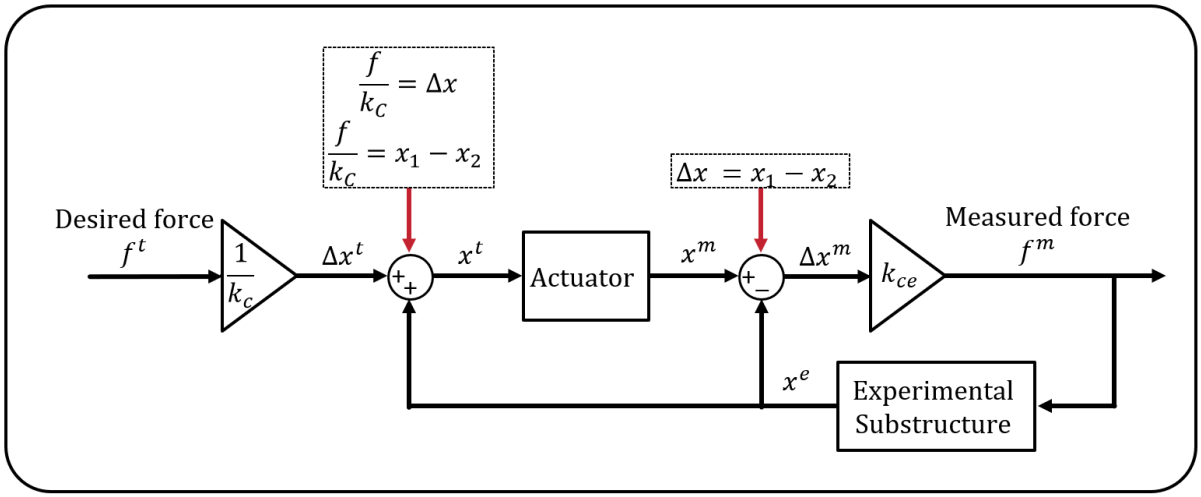


Figure 4.14: Block diagram of compliance spring incorporation effect (Sivaselvan et al., 2008).

The spring chosen for this study has mechanical properties that allow it to function in the linear elastic range throughout the test. In addition, the spring cannot be too flexible (very low stiffness) compared to the specimen since, to achieve the commanded displacement, the piston must perform a more significant displacement with a higher velocity, leading to instability in the simulation. It is also possible for the actuator to break down by commanding a displacement for which it was not designed since it exceeds its limit. On the contrary, adding a spring with a stiffness higher than the specimen can also lead to numerical instability since the piston displacement tends to be null. Thus, the spring stiffness of choice allows the actuator not to exceed the benchmark problem's limitations (Silva et al., 2020) and ensures simulation stability. The physical limitations considered for the servo-hydraulic actuator in this study are shown in Table 4.8.

Table 4.8: Actuator maximum physical capacity.

Parameter	Value	Unit
Stroke	±63	mm
Piston velocity	300	$\frac{\text{mm}}{\text{s}}$
Force	8.9	kN

Thus, the chosen compliance stiffness corresponds to 5% of the stiffness of DOF 1, as indicated in equation (4.20). Table 4.9 presents the nominal parameters of the specimen and the compliance spring that will be used in this study.

$$k_c = \frac{5}{100} \cdot k_e \tag{4.20}$$

Table 4.9: Specimen and compliance spring nominal values.

Parameter	Value	Unit	Parameter	Value	Unit
m_e	$98.3 \cdot 10^{-6}$	$\frac{\text{kN}\cdot\text{s}^2}{\text{mm}}$	k_e	10.836	$\frac{\text{kN}}{\text{mm}}$
c_e	$3004.8 \cdot 10^{-6}$	$\frac{\text{kN}\cdot\text{s}}{\text{mm}}$	k_c	0.5418	$\frac{\text{kN}}{\text{mm}}$

In addition to the aforementioned conditions for selecting the compliance spring stiffness, tests are conducted considering a range of compliance spring stiffness values. These values are compared with the maximum piston velocity for each test. Figure 4.15 shows the obtained results, where the piston velocity increases as the compliance stiffness decreases. Conversely, if the compliance stiffness increases, the piston velocity decreases. The figure also indicates the proposed stiffness point. The reason for not demanding a higher piston velocity is that when using the controllers later on, the speed will increase to reach the target displacements. Therefore, it is chosen to maintain this stiffness ratio without compensation and with nominal values for the physical parameters.

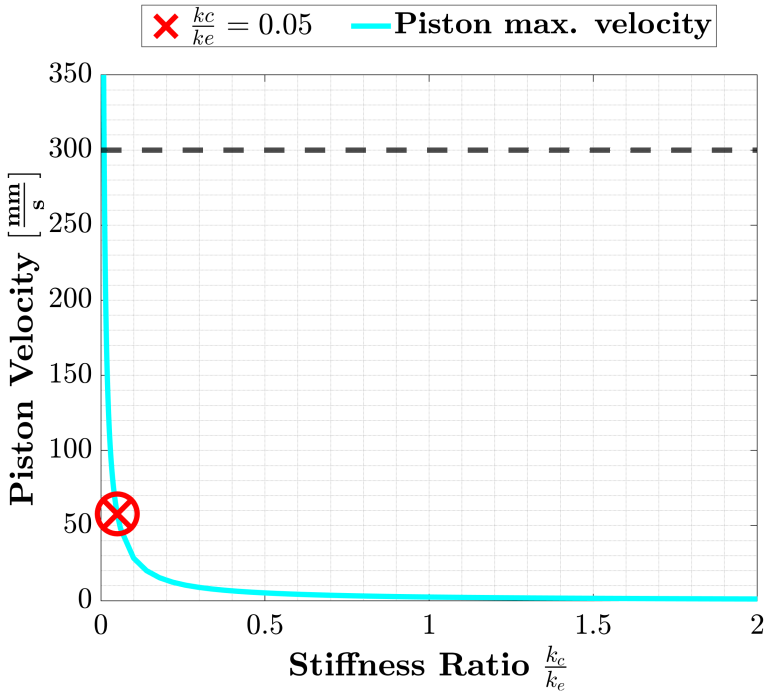


Figure 4.15: Piston velocity for different stiffness ratio values.

Finally, Figure 4.16 presents a Bode diagram featuring the initial plant $G_{ini}(s)$ and the plant with the specimen $G_e(s)$. Following the guidelines proposed by Sivaselvan et al. (2008), the compliance spring is incorporated in series with the specimen, resulting in a lower equivalent stiffness or, in other words, adding flexibility to the system. Therefore, incorporating the compliance spring is expected to initially improve the system response. Three cases are shown: stiffness lower than, equal to, and more significant than the specimen. Bode diagrams are calculated for different control plants $G_p(s)$ considering a range of compliance spring stiffness values $k_c = [0.01k_e \sim 10k_e]$, where it can be observed that even by using a spring with a higher stiffness than that of the specimen, the result improves. Theoretically, if the compliance spring is highly flexible, the piston will move freely, thus returning to the initial plant without the specimen $G_{ini}(s)$. On the other hand, if the compliance spring is too rigid, the actuator cannot impose displacements, leading to simulation failure due to numerical instability caused by the lack of convergence in each time step.

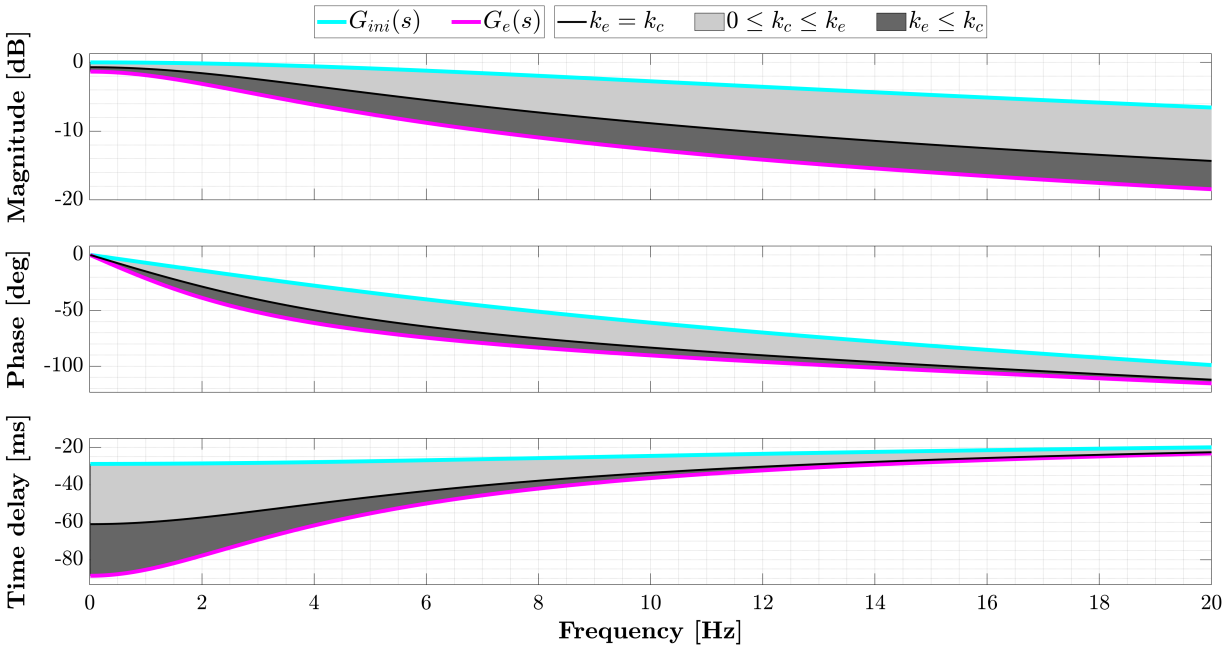


Figure 4.16: Magnitude, phase and time delay for different values of k_c.

4.4. Seismic Records

The seismic records that will be used in this study are widely employed in hybrid simulation research. Table 4.10 displays the names of the selected records along with relevant data that characterize them. Figure 4.17 shows a graph depicting the unscaled acceleration records for each seismic event. The CEN record is chosen as the primary one, with which the verifications and adaptive gain optimization process are performed.

Table 4.10: Ground motion records parameters.

Record	Label	M_w	Comp.	PGA [g]
El Centro 1940	CEN	6.9	NS	0.3197
El Maule 2010	MAU	6.9	NS	0.4009
Kobe 1995	KOB	6.9	NS	0.6455

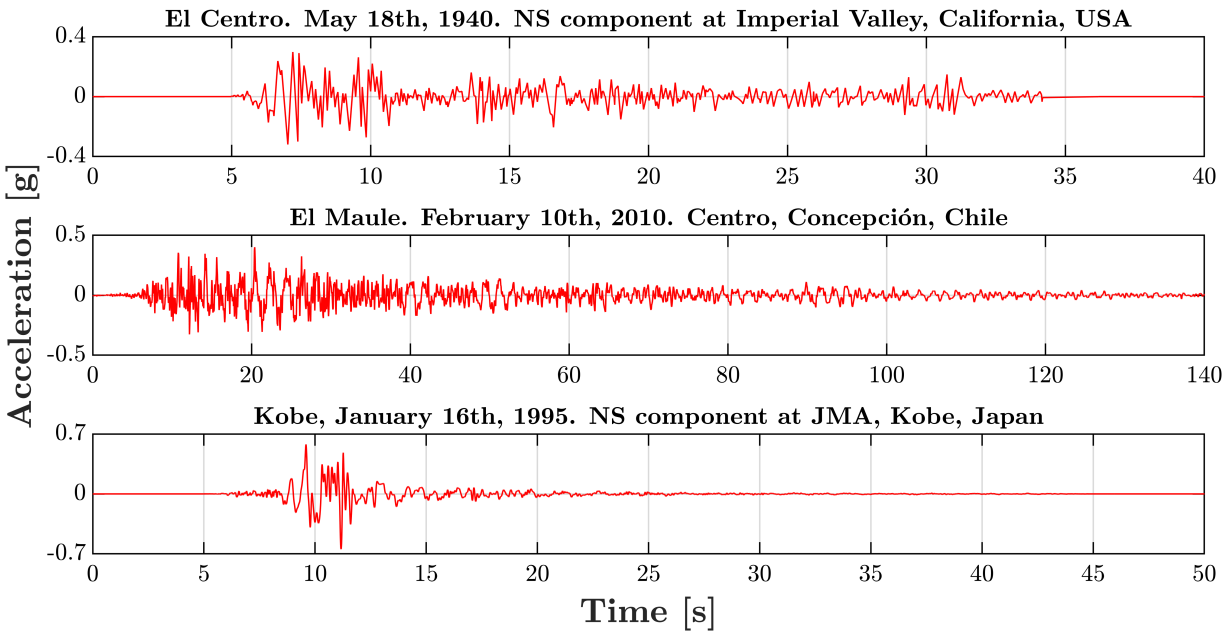


Figure 4.17: Acceleration records for chosen earthquakes.

4.5. Performance Evaluation Criteria

The performance evaluation of the adaptive controller will be carried out by calculating performance estimators from the synchronization errors between the commanded and measured signals obtained from each simulation. In certain cases, synchronization with the reference structure will also be evaluated. These estimators are widely used in real-time experimental testing research. In this study, estimators that are already defined in the benchmark problem (Silva et al., 2020) will be used, along with estimators specifically defined for force signals.

The following list of estimators is expressed in the discrete-time domain, where l represents the discrete time index, and L represents the total number of elements in the vector, except for time delay, which is defined in the frequency domain.

1. $\mathbf{J_2}$: The normalized root-mean-square error (NRMSE) between the target displacement x^t and the measured displacement x^m is used to measure the synchronization error between the displacements caused by the transfer system.

$$J_2 = \sqrt{\frac{\sum_{l=1}^{L}(x^m[l] - x^t[l])^2}{\sum_{l=1}^{L}(x^t[l])^2}} \cdot 100\ [\%] \tag{4.21}$$

2. $\mathbf{A_2}$: The normalized root-mean-square error (NRMSE) between the target force signal f^t and the measured force signal f^m measure the synchronization error between the mentioned signals. It is equivalent to the J_2 criterion for displacements.

$$A_2 = \sqrt{\frac{\sum_{l=1}^{L}(f^m[l] - f^t[l])^2}{\sum_{l=1}^{L}(f^t[l])^2}} \cdot 100\ [\%] \tag{4.22}$$

3. $\mathbf{J_4}$: The normalized root-mean-square error (NRMSE) between the reference displacement x_i^r and the measured displacement x_i^m at the DOF of interest i. It measures the synchronization error between the simulation substructured and the reference displacement signal.

$$J_4 = \sqrt{\frac{\sum_{l=1}^{L}(x_i^r[l] - x_i^m[l])^2}{\sum_{l=1}^{L}(x_i^r[l])^2}} \cdot 100\ [\%] \tag{4.23}$$

4. $\mathbf{A_4}$: The normalized root-mean-square error (NRMSE) between the reference force f_i^r and the measured force f_i^m at the DOF of interest i. It measures the synchronization error between the simulation substructured and the reference force signal.

$$A_4 = \sqrt{\frac{\sum_{l=1}^{L}(f_i^r[l] - f_i^m[l])^2}{\sum_{l=1}^{L}(f_i^r[l])^2}} \cdot 100\ [\%] \tag{4.24}$$

5. $\mathbf{R_2}$: The mean of the normalized root-mean-square error (NRMSE) between the reference displacement x_i^r and the measured displacement x_i^m at the DOF of interest i.

$$R_2 = \frac{1}{L}\sum_{l=1}^{L}(J_{2,n}) \tag{4.25}$$

6. $\mathbf{R_{2A}}$: The mean of the normalized root-mean-square error (NRMSE) between the reference force f_i^r and the measured force f_i^m at the DOF of interest i.

$$R_{2A} = \frac{1}{L}\sum_{l=1}^{L}(A_{2,n}) \tag{4.26}$$

7. $\tau_\mathbf{d}$ is the time delay between target and measured signals, obtained in the frequency domain using the guidelines proposed in the study of (Guo et al., 2014). Its unit is *miliseconds* [ms].

5 Compensation design

The next chapter presents the procedure for designing the compensation for the problem formulated in Chapter 4, based on the theory shown in Section 3.4. Although the AMBC is the only method required for the study's objective, an IFF will also be used for comparative purposes, as its coefficients remain constant over time.

5.1. Inverse feedforward compensation

Following the guidelines outlined in Section 3.4.1, the first step is obtaining the inverse of the control plant $G_p(s)$, defined in the previous chapter. The estimated control plant $G_{est}(s)$ will be used as the control plant, as it has the same dynamics as $G_p(s)$ within the frequency range of interest. Thus, considering the format of equation (3.26), the inverse of the control plant, denoted as $G_p^{-1}(s)$, is written as follows:

$$K_{\text{FF}}(s) = G_p^{-1}(s) = \frac{x^m}{x^c} = \frac{s^3 + 422.2s^2 + 1.007 \cdot 10^5 s + 4.561 \cdot 10^6}{4.497 \cdot 10^6} \tag{5.1}$$

By reducing the terms of equation (5.1) and rearranging them for later implementation in the simulation, the following expression is obtained:

$$\begin{aligned} &K_{\text{FF}}(s) = 1.0142 + 0.022387s + 9.3874 \cdot 10^{-5}s^2 + 2.2235 \cdot 10^{-7}s^3 \\ \Rightarrow \quad &x^c = \begin{bmatrix} 1.0141 & 0.022378 & 9.3837 \cdot 10^{-5} & 2.2226 \cdot 10^{-7} \end{bmatrix} \begin{bmatrix} x^t & \dot{x}^t & \ddot{x}^t & \dddot{x}^t \end{bmatrix}^T \end{aligned} \tag{5.2}$$

The IFF controller must be expressed in the discrete-time domain to be implemented as an FIR filter in the simulation. The temporal derivatives of equation (5.2) are calculated using the BDM to achieve this. Since the control plant has three poles, equation (3.33) is then written as follows:

$$\begin{bmatrix} x^t[k] \\ \dot{x}^t[k] \\ \ddot{x}^t[k] \\ \dddot{x}^t[k] \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{\Delta t} & \frac{-1}{\Delta t} & 0 & 0 \\ \frac{1}{\Delta t^2} & \frac{-2}{\Delta t^2} & \frac{1}{\Delta t^2} & 0 \\ \frac{1}{\Delta t^2} & \frac{-3}{\Delta t^2} & \frac{3}{\Delta t^2} & \frac{-1}{\Delta t^2} \end{bmatrix} \begin{bmatrix} x^t[k] \\ x^t[k-1] \\ x^t[k-2] \\ x^t[k-3] \end{bmatrix} \tag{5.3}$$

In a summarized form, the coefficients for calculating the approximations of the temporal derivatives that will be replaced in equation (3.34) are shown in the following table:

Table 5.1: Finite difference coefficients of higher-order time derivatives approximations.

Signal	b_0	a_0	a_1	a_2	a_3
$x^t[k]$	1	1			
$\dot{x}^t[k]$	Δt	1	-1		
$\ddot{x}^t[k]$	Δt^2	1	-2	1	
$\dddot{x}^t[k]$	Δt^3	1	-3	3	-1

The coefficients of the temporal derivatives are obtained considering a time step of $\Delta t = \frac{1}{4096}[s]$. By combining these coefficients with those obtained in equation (5.2), the values of the coefficients k_i in equation (3.36) are obtained, which are described below in a vector defined as K_{ff} as shown in equation 5.4.

$$\mathbf{K}_{\text{ff}} = 10^4 \cdot \begin{bmatrix} 1.694 & -4.981 & 4.741 & -1.528 \end{bmatrix} \tag{5.4}$$

Thus, the *IFF* controller implemented as an FIR filter is expressed as follows:

$$x^c[k] = 1.694 \cdot 10^4 x^t[k] - 4.981 \cdot 10^4 x^t[k-1] + 4.741 \cdot 10^4 x^t[k-2] - 1.528 \cdot 10^4 x^t[k-3] \tag{5.5}$$

The figure below shows the implementation of the FIR filter in MATLAB, where it receives the target displacement signal x^t as input and outputs the commanded displacement signal x^c, which serves as input to the control plant $G_p(s)$.

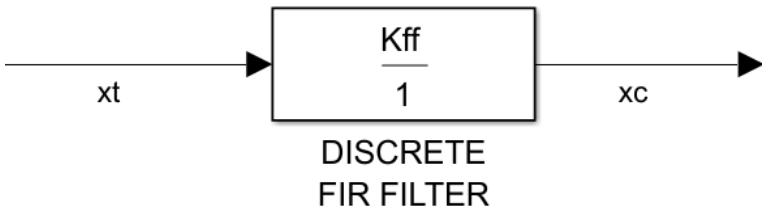


Figure 5.1: FIR filter implementation in Simulink.

To verify the proper implementation of the IFF controller, the transfer function called $L_p(s)$ is calculated, defined as follows:

$$L_p(s) = K_{\text{ff}} \cdot G_p(s) \tag{5.6}$$

where $G_p(s)$ represents the control plant, and $K_{\text{ff}}(s)$ denotes the IFF controller transfer function. This is possible because transfer functions are defined in the Laplace domain, where the resulting signal is the product of the transfer functions that precede it. Figure 5.2 illustrates a diagram showing the three mentioned transfer functions.

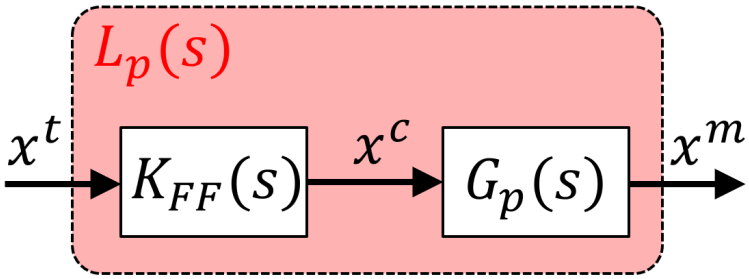


Figure 5.2: $L_p(s)$ obtained from the IFF controller implementation.

To obtain $L_p(s)$, an expression in the continuous-time domain for the designed FIR filter (5.5) must be derived. This is achieved using the Tustin method or *Bilinear Transformation*, which allows transfer functions to be written from the discrete-time domain to the continuous-time domain and vice versa. This method is directly applied using the Matlab command *d2c(·)*, where K_{FF} is defined as $d2c(K_{\text{ff}}, 'tustin')$. The transformed and rewritten transfer function of the FIR filter in the continuous-time domain is as follows:

$$K_{\text{FF}}(s) = \frac{1.287 \cdot 10^5 s^3 + 5.464 \cdot 10^7 s^2 + 1.251 \cdot 10^{10} s + 5.575 \cdot 10^{11}}{s^3 + 2.458 \cdot 10^4 s^2 + 2.013 \cdot 10^8 s + 5.498 \cdot 10^{11}} \tag{5.7}$$

Once the expression for $K_{\text{FF}}(s)$ is obtained, it is possible to calculate $L_p(s)$. The resulting expression is as follows:

$$L_p(s) = \frac{5.789 \cdot 10^{11} s^3 + 2.457 \cdot 10^{14} s^2 + 5.627 \cdot 10^{16} s + 2.507 \cdot 10^{18}}{s^6 + 2.5 \cdot 10^4 s^5 + 2.118 \cdot 10^8 s^4 + 6.372 \cdot 10^{11} s^3 + 2.525 \cdot 10^{14} s^2 + 5.627 \cdot 10^{16} s + 2.507 \cdot 10^{18}} \tag{5.8}$$

Given that $G_p(s)$, $K_{\text{FF}}(s)$, and $L_p(s)$ are known transfer functions, a Bode plot is generated to compare their respective dynamics in the frequency domain:

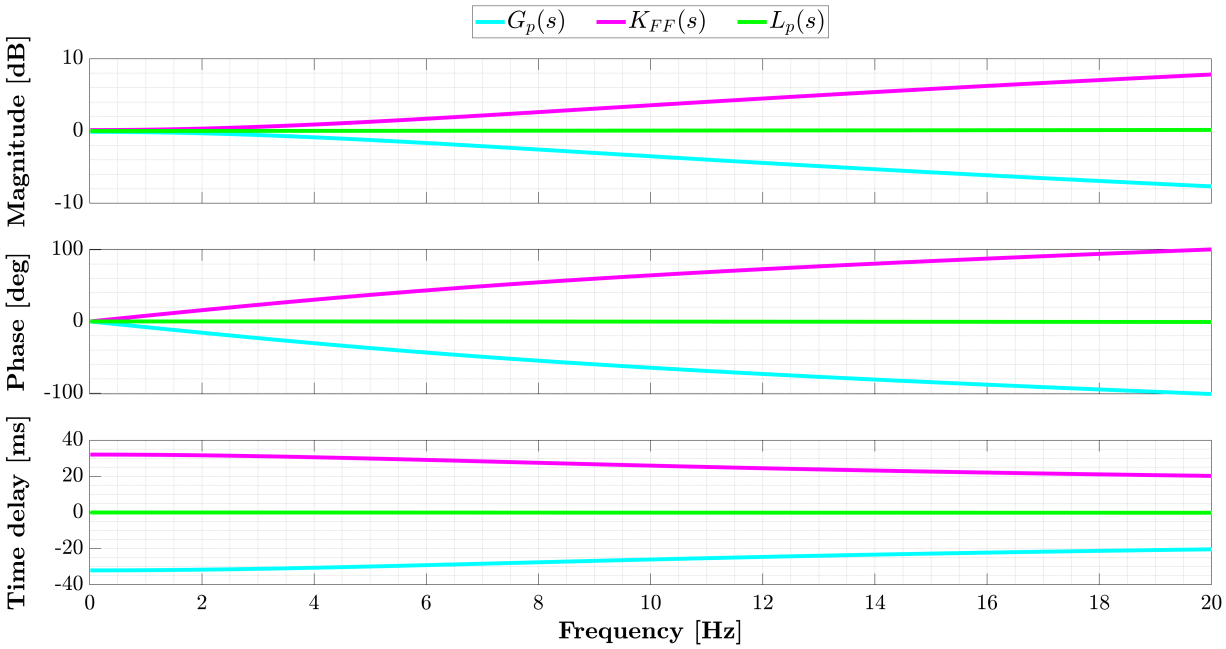


Figure 5.3: Comparative bode plot for IFF controller design.

In the magnitude, phase, and time delay plots of Figure 5.3, it can be observed that the control

plant $G_p(s)$ and the IFF controller $K_{\text{FF}}(s)$ exhibit opposite behavior. As a result, $L_p(s)$ corresponds to a continuous line with a value of 0 for magnitude, phase, and time delay, confirming the correct implementation of an IFF controller, as the dynamics of the control plant are nullified. In an ideal case, the commanded and measured displacements are approximately equal, i.e., $x^t \approx x^m$, within the range of frequencies of interest.

It is essential to mention that the Tustin method is only applied to verify the implementation. The detailed development of the domain transformation using this method is beyond the scope of this thesis.

5.2. Adaptive model-based compensation

The design of AMBC follows the guidelines defined in section 3.4.2.1, where the first step involves obtaining a Laplace domain expression for the transfer system without the compliance spring or the specimen. The transfer function obtained from the nominal parameters corresponds to $G_{ini}(s)$:

$$G_{ini}(s) = \frac{5.035 \cdot 10^6}{s^3 + 425s^2 + 10^5 s + 5.035 \cdot 10^6} \tag{5.9}$$

The initial IFF controller, written as equation (3.40), is obtained by calculating the inverse of the initial plant $G_{ini}(s)$ as shown below:

$$x^c = \begin{bmatrix} 1 & 1.986 \cdot 10^{-2} & 8.440 \cdot 10^{-5} & 1.986 \cdot 10^{-7} \end{bmatrix} \begin{bmatrix} x^t & \dot{x}^t & \ddot{x}^t & \dddot{x}^t \end{bmatrix}^T \tag{5.10}$$

As mentioned in section 3.4.2.1, a fourth-order Butterworth filter with a cutoff frequency of $20[Hz]$ is implemented, corresponding to the maximum value within the frequency range of interest in the study. This filter has a flat response (unit gain) up to values close to the cutoff frequency. This means it allows most of the low-frequency input signal to pass while attenuating high-frequency noise (low-pass filter).

Figure 5.4 shows the Bode plot of the initial plant $G_{ini}(s)$, the control plant $G_p(s)$, and the Butterworth filter. It can be observed that the frequency response is a flat graph that begins to decline as it approaches the upper limit frequency of interest. Additionally, the Butterworth filter introduces an extra time delay in the signal. However, since the filter is applied to the commanded and measured signals, it will only affect the adaptation process, indirectly impacting the compensation process.

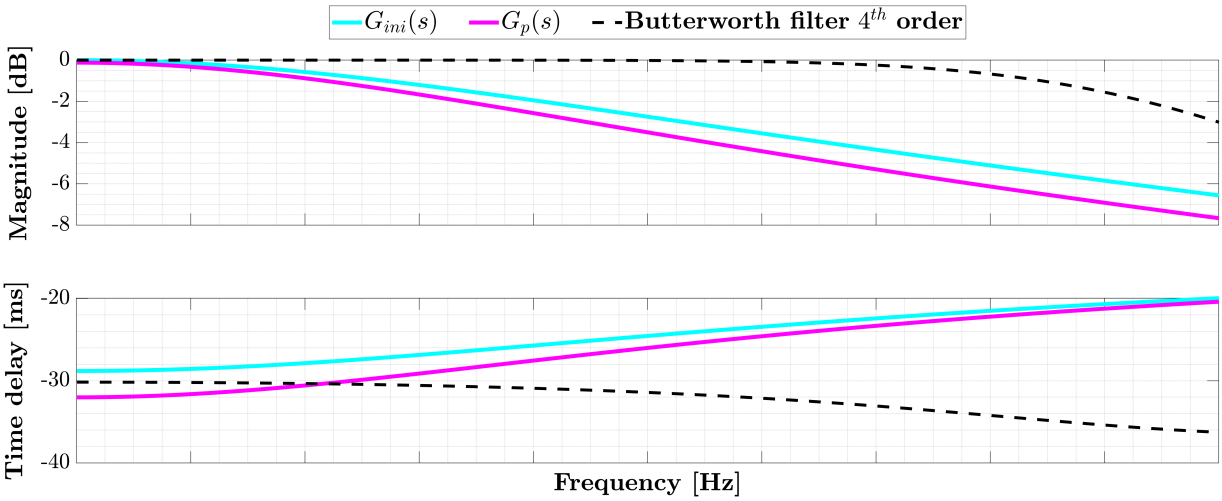


Figure 5.4: Comparative bode plot of initial plant, control plant, and Butterworth filter.

5.2.1. Adaptive gains optimization

Following the control plant configuration shown in Figure 4.6, due to the inclusion of the compliance spring, it is possible to transform the target force f^t into the target displacement x^t. In this way, it is possible to implement the same optimization algorithm used by Gálmez and Fermandois (2022). Figure 5.5 shows the block diagram implementation in Simulink, where the optimization process will occur.

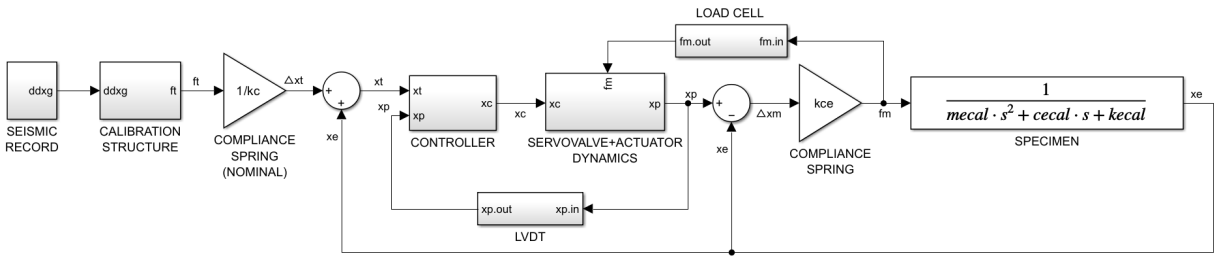


Figure 5.5: Simulink block diagram for the optimization process.

The implementation shown in Figure 5.5 is constructed based on the optimization done by Gálmez and Fermandois (2022) with modifications to adapt it to a force-based environment. As mentioned, the compliance spring allows switching domains (from force to displacement), thereby designing the controller based on displacements. Furthermore, the load cell and LVDT are incorporated, which are used to measure the force signals of the compliance spring and piston displacement, respectively. Both instruments introduce noise that contaminates their respective measured signals. The noise is modeled as white noise, where a noise power and seed value are utilized to achieve a minimum Signal-to-Noise Ratio (SNR) of 30 for both instruments. This value is configured beforehand to ensure all simulations achieve an SNR above the minimum mentioned. This value guarantees that the signal measured has a good quality despite the noise added by the instrument.

As mentioned in section 3.4.2.2, the optimization process utilized is *Particle Swarm Optimization* (Kennedy and Eberhart, 1995). The algorithm aims to minimize a cost function each

time a generation is completed. A finite number of particles are defined to achieve the previous statement, and each particle is assigned with distinct values of adaptive gains Γ_k. Furthermore, considering the presence of experimental parameter uncertainties, a certain number of simulations are performed for each particle. After each particle's simulation, the corresponding evaluation estimator J_2 is calculated, which represents the *Normalized Root Mean Square Error* (NRMSE) of synchronization as shown in equation 3.48. Subsequently, upon completing the simulations for each particle, the average of the obtained estimators $J_{2,n}$ is calculated, referred to as $R_2(\Gamma_k)$, as shown in equation (3.49). The first input to be defined corresponds to the input values for generations, particles, and simulations used in this study, shown in Table 5.2.

Table 5.2: Particle swarm optimization inputs.

Parameter	Symbol	Value
Generations	N_g	30
Particles	N_p	30
Simulations	N_s	100

The optimization process is performed considering uncertainty in the parameters of the reference structure, the seismic scale, and the stiffness of the experimental compliance spring, which is entered into the algorithm as a percentage P_{unc} to generate ranges of possible input values. This approach is chosen to incorporate uncertainty as it quantifies, in a straightforward manner, the potential measurement error due to instrument imperfections or specialist errors. To illustrate this step, consider the nominal value of the reference structure mass, $m_r = 98.3$ [kg], which, when considering a $P_{unc} = 20\%$ uncertainty, implies that the mass value for the optimization will be chosen within the range $98.3 \cdot [1 - 0.2, 1 + 0.2] \Rightarrow [78.64, 117.96]$ [kg]. In this way, it is established that for all inputs where uncertainty is considered, their ranges will be calculated as follows:

$$\text{input}^* = \text{input} \cdot \left[1 - \tfrac{P_{inc}}{100} \quad 1 + \tfrac{P_{inc}}{100}\right] \tag{5.11}$$

The uncertainty is added to the parameters of the reference structure since they correspond to the values measured by the experimenter. In this way, the uncertainty will propagate to the parameters after the substructuring has been performed. Although the inputs will be chosen from a range of values, the Matlab function *lhsdesign(·)*, created based on Latin Hypercube Sampling (LHS) (McKay et al., 2000), is used. To explain the operation of LHS, let's consider $N_p = 10$ particles as input to the function *lhsdesign(10)*, from which a column vector with 10 values chosen from the interval $\left[(0, \frac{1}{N_p}) \quad (\frac{1}{N_p}, \frac{2}{N_p}) \quad \dots \quad (\frac{N_p-1}{N_p}, 1)\right]$ will be obtained. These values are then randomly permuted. This ensures that is obtained a set of random values representing the actual variability of the parameters, which will incorporate the uncertainty.

For the adaptive gains, the diagonal matrix Γ^* is defined. Conveniently, the vector γ is defined as:

$$\gamma := \log_{10}\left(\text{diag}(\Gamma^*)\right) = \begin{bmatrix}\gamma_0 & \gamma_1 & \gamma_2 & \gamma_3\end{bmatrix} \tag{5.12}$$

Where the values γ_j correspond to the exponent of each base ten coefficient, thus, a vector γ is assigned with initial values for each particle randomly chosen before optimization. These values are bounded from below and above. Table 5.3 displays the gains values limits. The same range chosen by Gálmez and Fermandois (2022) is selected to compare the results obtained based on the same established limits.

Table 5.3: Initital values limits.

Coefficient	Min. value	Max. value
γ_0	2	10
γ_1	0	8
γ_2	-2	6
γ_3	-4	4

Thus, at the end of each generation, it is possible to obtain the minimum value of $R_2(\Gamma_k)$ among all particles. Therefore, the velocity $v_p[g]$ and position $p_p[g]$ of each particle are updated according to the following expressions, respectively:

$$v_p[g+1] \coloneqq \omega \cdot v_p[g] + c_1 \cdot r_1[g] \cdot (PB_p[g] - p_p[g]) + c_2 \cdot r_2[g] \cdot (GB[g] - p_p[g]) \tag{5.13}$$

$$p_p[g+1] \coloneqq p_p[g] + v_p[g+1] \tag{5.14}$$

where the subscript p refers to each particle $p \in [1, N_p]$, $g \in [1, N_g]$ corresponds to the generation, c_1 and c_2 represent the pre-defined weights for velocity update, *PB* stands for *Personal Best Result* of each particle, i.e., the lowest value of R_2 obtained by each particle, and *GB* represents the *Global Best Result*, i.e., the lowest value of R_2 obtained among all particles. ω corresponds to the weight of the inertia of the previous velocity, which is considered constant in this process. The coefficients $r_1[g]; \wedge; r_2[g] \in [0, 1]$ are randomly chosen for each generation. The predefined values of the mentioned parameters are shown in Table 5.4.

Table 5.4: Weight predefined values.

Parameter	Symbol	Value
Inertia weight	ω	0.5
Weight for personal best result	c_1	0.5
Weight for global best result	c_2	0.5

The uncertainty value for the numerical and experimental parameters is defined, expressed as a percentage, as indicated below:

$$P_{unc} = 20\% \tag{5.15}$$

Table 5.5 presents a list of the input parameters, where their nominal, minimum, and maximum values formed with the level of uncertainty are displayed.

Table 5.5: Input parameters values and uncertainties for the optimization process.

Parameter	Nominal value	Range	Unit
m_r	98.3	[78.64 , 117.96]	kg
$k_{r,1}$	10.794	[8.635 , 12.953]	$\frac{\text{kN}}{\text{mm}}$
$k_{r,2}$	0.686	[0.548 , 0.823]	$\frac{\text{kN}}{\text{mm}}$
$k_{r,3}$	0.684	[0.547 , 0.820]	$\frac{\text{kN}}{\text{mm}}$
Scale	80	[64 , 96]	%

The modal damping matrix $\mathbf{C}$ with uncertainty is obtained from the mass and stiffness matrices with uncertainty. No uncertainty is added to the critical damping ratios ζ_i since, for this study, it is assumed that the uncertainty arises from the combination of mass and stiffness for each particle. Furthermore, due to the permutation of the uncertainty values generated by the *lhsdesign(·)* function, it is ensured that the modal damping matrix is generated from the mass and stiffness matrix with different uncertainties for each particle within the range defined in Table 5.5. For instance, the steps performed by the algorithm to obtain the modal damping matrix with uncertainty for one particle and $N_s = 5$ simulations are summarized below:

1. The nominal mass values $m_{r,1}$, $m_{r,2}$, and $m_{r,3}$, and stiffness values $k_{r,1}$, $k_{r,2}$, and $k_{r,3}$ are given for each floor.

2. A matrix of 5 rows and 6 columns is generated using the command *lhsdesign(5,6)*. The obtained result in Matlab is as follows:

$$\text{lhsdesign}(5,6) = \begin{bmatrix} 0.76 & 0.74 & 0.81 & 0.73 & 0.12 & 0.79 \\ 0.51 & 0.98 & 0.31 & 0.24 & 0.31 & 0.09 \\ 0.05 & 0.36 & 0.58 & 0.06 & 0.91 & 0.99 \\ 0.27 & 0.45 & 0.68 & 0.88 & 0.75 & 0.23 \\ 0.93 & 0.18 & 0.03 & 0.52 & 0.50 & 0.43 \end{bmatrix} \tag{5.16}$$

3. The mass with uncertainty for floor 1 is calculated as $m_{r,1}^{unc} = m_{r,1} \cdot [0.8 + (1.2 - 0.8) \cdot \text{lhsdesign}(:, 1)]$, resulting in a vector of 5 mass values randomly selected within the range of interest. The obtained result for this example is $m_{r,1}^{unc} = [108.52\ ,\ 98.69\ ,\ 80.61\ ,\ 89.26\ ,\ 115.21]$ [kg]. The process is repeated for the remaining parameters $m_{r,i}$ y $k_{r,i}$.

4. Finally, for each particle, a modal damping matrix $\mathbf{C}$ is calculated.

In this way, uncertain values are generated for each one of the particles mentioned in the input table 5.2. Then, the optimization process is carried out.

5.2.2. Optimization results

The optimization yields the following values for γ^*:

$$\gamma^* = \begin{bmatrix} 9.2424 & 7.0077 & 3.0903 & -0.0707 \end{bmatrix} \tag{5.17}$$

Therefore, the resulting diagonal matrix Γ^* from the optimization process is as follows:

$$\Gamma^* = \text{diag}\left(\begin{bmatrix} 10^{9.24} & 10^{7.01} & 10^{3.09} & 10^{-0.07} \end{bmatrix}\right) \tag{5.18}$$

The resulting value of R_2 from the optimization process is:

$$R_2 = 1.4857\% \tag{5.19}$$

The resulting value of R_2 from equation (5.19) is higher than the one obtained in the study conducted by Gálmez and Fermandois (2022). Although the numerical and experimental properties and the problem formulation are different, an optimization process is performed considering the same described parameters without considering the load cell and LVDT noise to observe the impact of its incorporation in the control plant of the optimization. This instrument typically adds a considerable level of noise to the measured signal, directly affecting the convergence error value R_2.

Table 5.6 shows a comparative table of the obtained R_2 values for three cases: Optimization with included load cell, optimization without load cell, and the result obtained by Gálmez and Fermandois (2022).

Table 5.6: Numerical and experimental parameters values and uncertainties for the optimization process.

Case	R_2 [%]	γ^*
Force-based without load cell	0.42	$\begin{bmatrix} 9.1 & 7.8 & 3.4 & -0.7 \end{bmatrix}$
Force-based with load cell	1.48	$\begin{bmatrix} 9.2 & 7.0 & 3.1 & -0.1 \end{bmatrix}$
Gálmez and Fermandois (2022)	0.85	$\begin{bmatrix} 8.4 & 6.2 & 2.1 & 0.8 \end{bmatrix}$

Figure 5.6 displays a graph showing the evolution of each particle's γ_i parameters along with the optimized value obtained. It can be observed that from generation 15 onwards, approximately, the γ_i values tend to stabilize around the optimum. However, the convergence is not absolute due to the parameter's uncertainties and the added load cell noise. The adaptive gain γ_3 is the gain that shows the least convergence compared to the other values of γ^*.

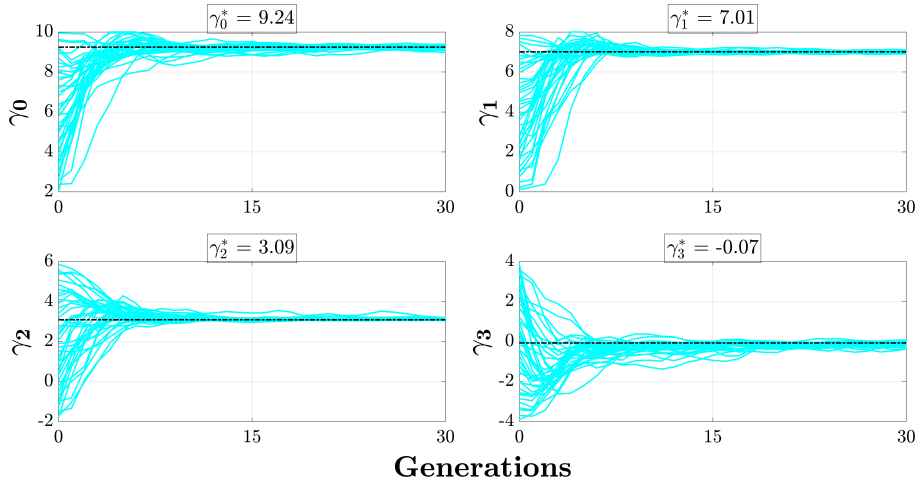


Figure 5.6: γ_i evolution though generations.

Figure 5.7 shows the evolution of the R_2 estimator throughout each generation for each particle. It can be observed that the R_2 values of each particle tend to converge to the optimal R_2 value. A graph of the last three iterations is generated on the right side of Figure 5.7 to watch what happens in the vicinity of the R_2 values, where it is evident that the R_2 values are slightly unstable for values close to the optimum. This is mainly due to three reasons:

1. The iterative process and convergence criteria utilized in the particle swarm optimization process employ weight coefficients, which, for this study, were kept constant and without damping.
2. The noise from the load cell and the LVDT.
3. The uncertainty in the design parameters.

In Figure 5.8, the evolution of the estimator R_{2A} is shown, which has a higher optimal value than R_2 ($R_{2A} > 10\%$). However, it is observed that the values of R_{2A} also tend to converge. The discrepancy between the nominal and uncertain values of the compliance spring stiffness primarily generates this difference. In Section 6.2.1.1, further details regarding this variation in the compliance spring stiffness are discussed.

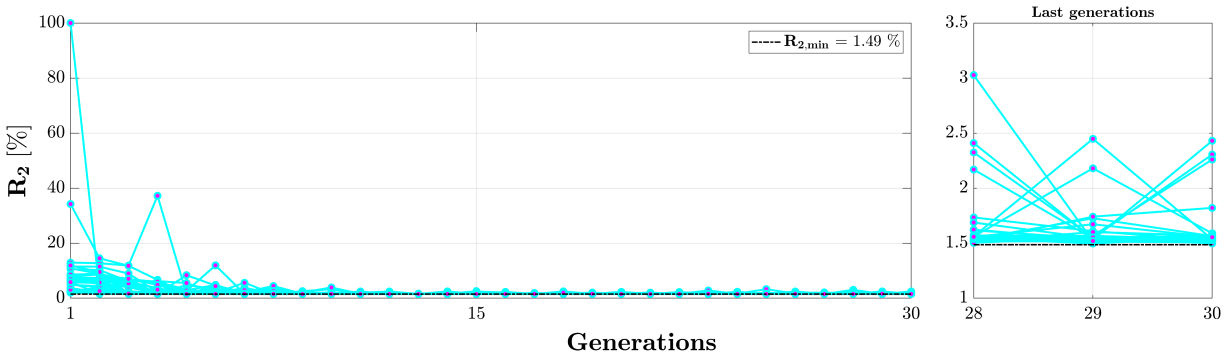


Figure 5.7: R_2 of each particle evolution though generations.

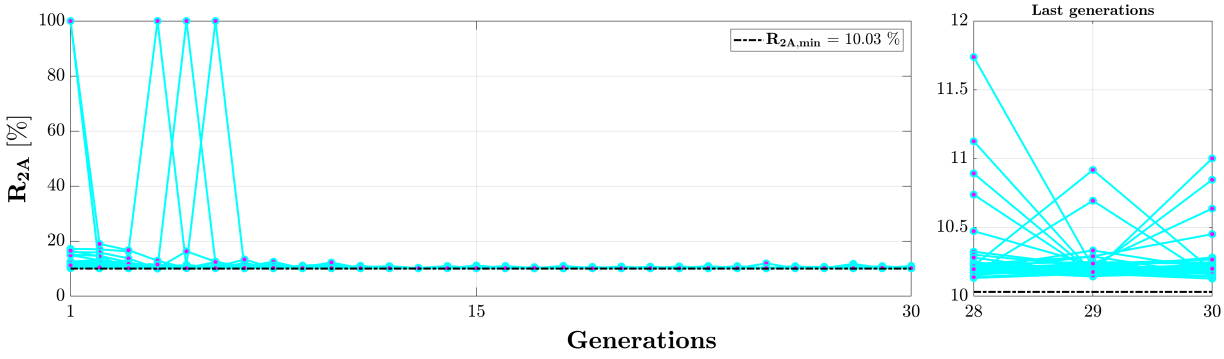


Figure 5.8: R_{2A} of each particle evolution though generations.

Another way to evaluate the result is to map values near the optima. This process consists of scanning values for two coefficients of γ^* while keeping the other two fixed. Since there are four parameters of γ^*, the analysis has six possible combinations. For the mapping, a grid of 20×20 is defined, resulting in 400 points per graph. The X and Y axes represent the values of the adaptive coefficients of γ^*, considering the defined limits from the optimization process. The value on the Z axis corresponds to the R_2 estimator obtained from performing 100 simulations for each point

of the grid.

Figure 5.9 shows the obtained graph, where the X and Y axes represent the variable coefficients, while the constant coefficients are indicated on each chart. The R_2 values are represented with a color gradient, as shown in the color bar on the right side of Figure 5.9. Additionally, the optimal values of the variable coefficients are marked in red. The optimal points for the graphs are located in the region where $R_2 < 2\%$. In all graphs of Figure 5.9, it is observed that the R_2 estimator values increase abruptly for high adaptive gains due to the overshooting that occurs in the adaptive process (Gálmez and Fermandois, 2022). Furthermore, it is observed that for most of the different combinations of γ_0 and γ_3, R_2 values below 2% are obtained, except for gain values close to their respective predefined limits.

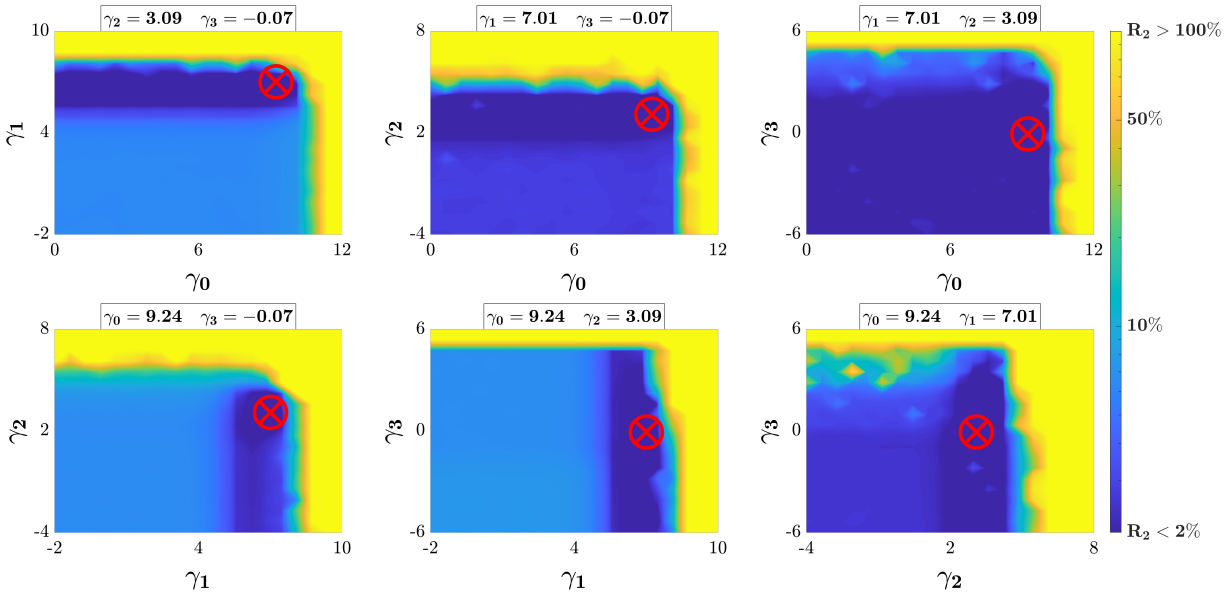


Figure 5.9: R_2 map for different adaptive gains around the optimal considering an uncertainty on the parameters of 20%.

Figure 5.10 shows the mapping results of the gains for the estimator R_{2A}, where all the obtained values are approximately between 10% and 20%. This is in contrast to the case of R_2, where distinct optimal zones can be identified in blue. In other words, regardless of the combination of adaptive gains γ_j^*, the value of R_{2A} will exhibit a constant trend. However, for high values of gains γ_j^*, the values of the estimator R_{2A} are greater than or equal to 100%, coinciding with the values greater than or equal to 100% obtained for R_2.

It is essential to mention that the obtained results from the optimization process are based on the nominal reference parameter values with uncertainty added, so any modification will yield different results.

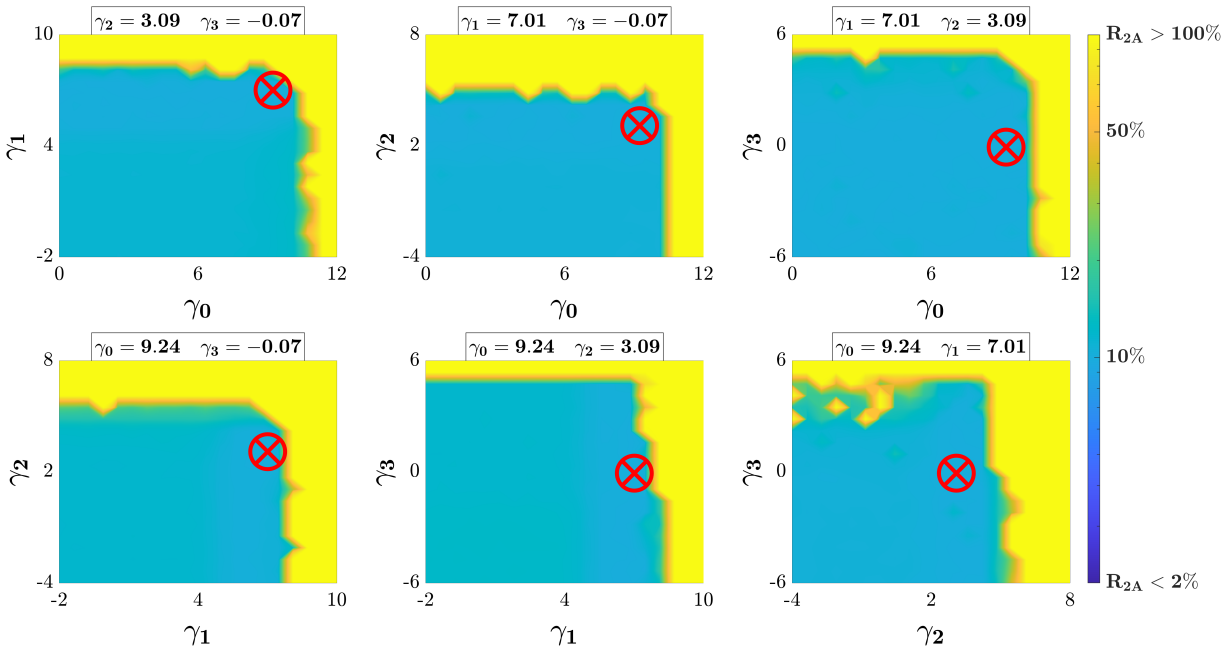


Figure 5.10: R_{2A} map for different adaptive gains around the optimal considering an uncertainty on the parameters of 20%.

5.2.3. Adaptive gains sensibility

A sensitivity analysis is performed to assess the impact of varying the coefficients γ_j on the estimators R_2 and R_{2A}. A set of 100 values is constructed for each γ_j within the limits defined in the optimization process shown in Table 5.3. For each value of γ_j, 100 simulations are conducted, and the estimators R_2 and R_{2A} are calculated. The results are shown in Figure 5.11 for R_2 and in Figure 5.12 for R_{2A}.

The results obtained for R_2 in Figure 5.11, where the values of γ_j^* obtained from the optimization process are also plotted in red. It reveals that the most sensitive gain is γ_1, as it exhibits a decrease in the value of R_2 for values between 5 and 7, with with a subsequent abrupt increase until reaching values of $R_2 \geq 100\%$. The parameters γ_0 and γ_2 show a similar R_2 behavior than γ_1 but on a smaller scale. Finally, γ_3 is the optimal gain that shows the highest dispersion in the values of R_2. However, the overall behavior of γ_3 tends to be the same as the rest of the optimal gains.

In Figure 5.12, the results of R_{2A} are presented for the same analysis described in the previous paragraph. It can be observed that the adaptive gains exhibit similar behavior to those obtained for the estimator R_2 but with values starting from approximately 10%. Moreover, for gain values close to the upper limit, the estimator R_{2A} reaches values of 100% or higher.

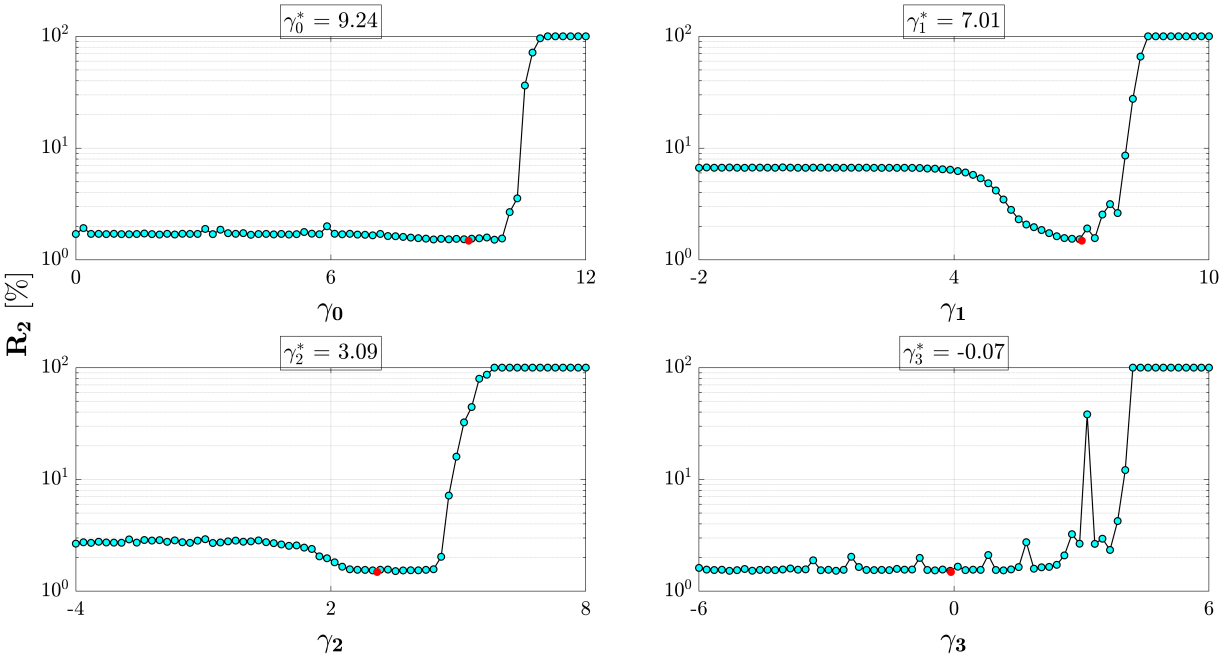


Figure 5.11: R_2 for variable values of γ_j.

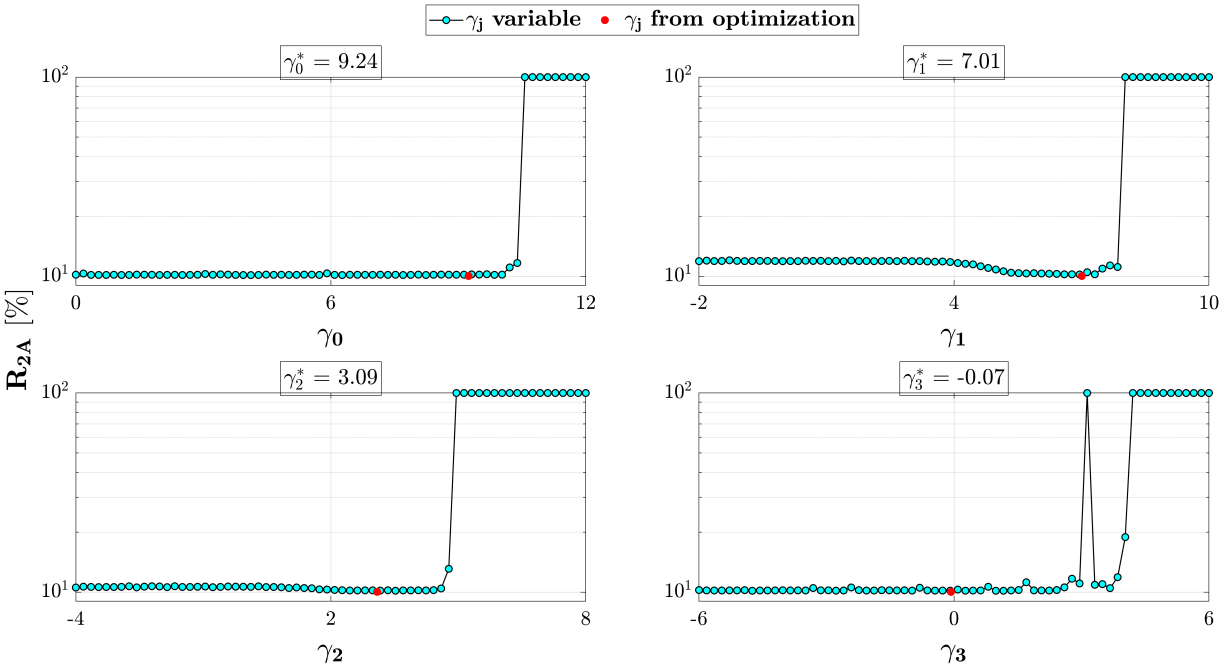


Figure 5.12: R_{2A} for variable values of γ_j.

6 Virtual simulations

In the following chapter, the computational implementation of the proposed methodology described in Chapter 4 is presented. Subsequently, the obtained results will be shown and analyzed. First, simulations are conducted using nominal parameters, where the results are analyzed with the overall objective of observing the impact on the variability of the parameters defined in each section. Then, simulations are performed considering uncertainty in the physical parameters, considering different cases of noise and seismic records.

6.1. Implementation

The problem formulated in chapter 4 is implemented in Simulink v10.7 of Matlab 2023a. The numerical integration method to be employed in the simulations is the fourth-order Runge-Kutta method (ode4) (Rosenbrock, 1963), which is suitable for solving differential equations of rigid systems, using a sampling frequency of 4096 [Hz]. Figure 6.1 shows the blocks assembled in Simulink, where the implementation of the substructuring is shown along with the transfer system and the LVDT. Figure 6.2 displays the inside of the control panel block from figure 6.1, where the actuator, compliance, specimen, and load cell are observed. The simulations to be performed will be conducted in the scheme depicted in these two figures.

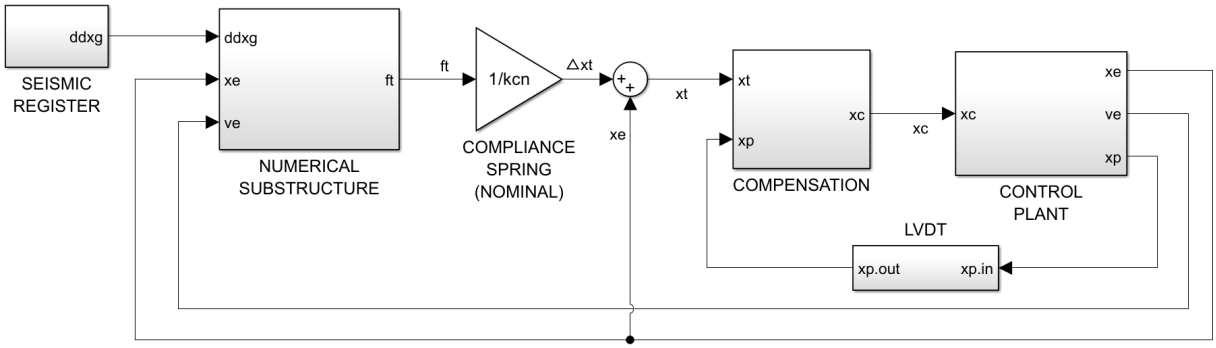


Figure 6.1: Simulink block diagram of the FBRTHS implementation.

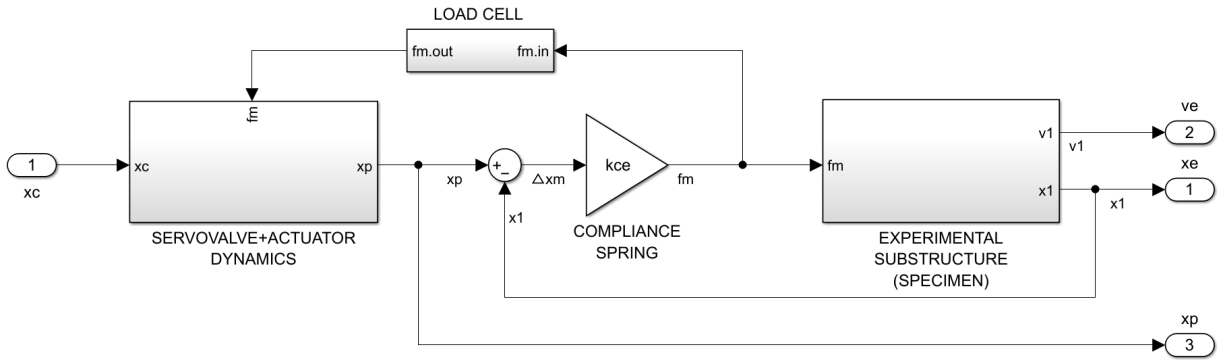


Figure 6.2: Simulink block diagram of the control plant.

6.2. Results

The results presented in the following sections are obtained from conducting n simulations described in each case. The obtained results are analyzed according to the criteria defined in section 4.5. Additionally, different implementation cases are considered to analyze the impact on the simulation results.

6.2.1. Simulations with nominal parameters

The first analysis considers the parameters with their nominal values and without measurement noise. A simulation is performed for each seismic record shown in section 4.4. The results are shown in table 6.1, where it can be observed that the obtained values of J_2 and A_2 for the AMBC are below 1% for all three earthquakes, surpassing the performance of the IFF compensator. The time delay is also reduced to less than 1 [ms] for the AMBC. The obtained results describe the excellent performance of the AMBC, as its objective in RTHS experiments is to reduce tracking errors. Finally, for the three earthquakes, it can be observed that the compensators require a higher piston velocity to achieve the commanded displacement within the simulation time step, with both results being below the limit of 300 $\frac{mm}{s}$.

Table 6.1: Tracking evaluation, time delay, and maximum piston's velocity for nominal cases for El Centro earthquake.

EQ	Case	$\mathbf{J_2}$ [%]	$\mathbf{A_2}$ [%]	$\tau_{\mathbf{d}}(\mathbf{x^t x^m})$ [ms]	$\tau_{\mathbf{d}}(\mathbf{f^t - f^m})$ [ms]	max $\mathbf{v_p}$ [$\frac{\mathbf{mm}}{\mathbf{s}}$]
	UNC	58.74	60.79	18.956	19.551	67.08
CEN	**IFF**	1.24	1.30	0.226	0.237	113.3
	AMBC	0.42	0.44	0.001	0.001	107.98
	UNC	47.23	49.18	18.306	18.815	57.07
MAU	**IFF**	1.05	1.11	0.224	0.236	72.20
	AMBC	0.07	0.07	$5.43 \cdot 10^{-4}$	$5.74 \cdot 10^{-4}$	72.09
	UNC	54.72	56.74	18.921	19.511	110.89
KOB	**IFF**	1.13	1.19	0.224	0.236	148.15
	AMBC	0.24	0.25	0.011	0.011	147.43

Table 6.2 presents the tracking errors concerning the reference structure, where it can be observed that excellent results are also obtained with the AMBC, as the values of A_4 and J_4 are approximated ten times lower than the IFF controller.

Table 6.2: Tracking results with the reference structure for El Centro earthquake.

Case	DOF	$\mathbf{J_4}$ [%]	$\mathbf{A_4}$ [%]
	1	60.34	13.57
UNC	2	16.84	16.16
	3	16.93	17.41
	1	1.32	0.55
IFF	2	0.39	0.37
	3	0.39	0.40
	1	0.44	0.53
AMBC	2	0.08	0.09
	3	0.08	0.09

Complementarily to the analysis of the estimators, the *synchronization subspace plot method* (SSP) (Wallace et al., 2005a) is used, which consists of plotting the target and measured signals. In an ideal scenario, these signals form a straight line with a slope equal to 1. A slope under 1 indicates undershoot, while a greater than 1 indicates overshoot. If the graph deviates from the straight line, it indicates the presence of a time delay between the signals. Thus, the graphs generated with the obtained results are shown in Figure 6.3 for displacements and Figure 6.4 for forces. Both cases were obtained using the El Centro (CEN) seismic record. For the uncompensated cases, the graph deviates from the straight line, and there is a tendency to generate a slope less than 1, as expected for an uncompensated test. On the contrary, excellent results are obtained

for the compensated cases, where a slightly better performance can be observed for the AMBC.

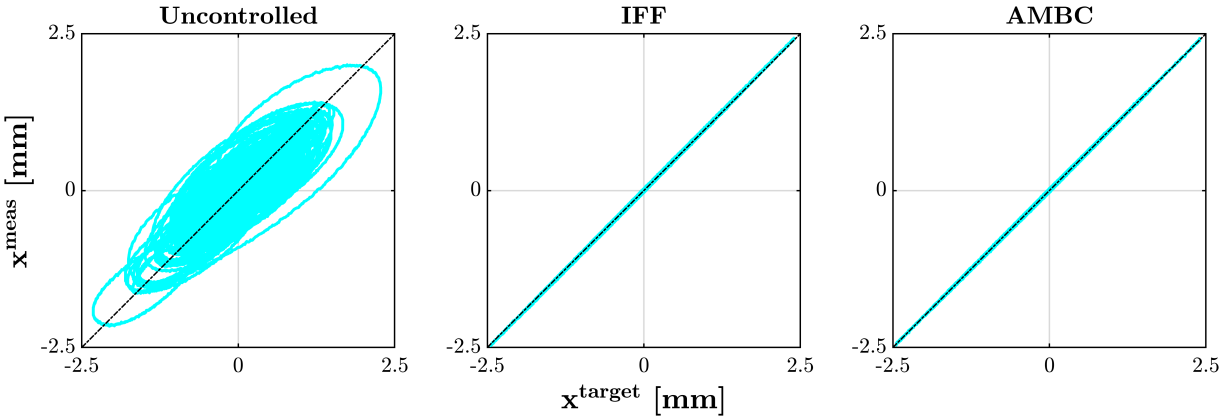


Figure 6.3: SSP plot of displacement tracking for El Centro earthquake.

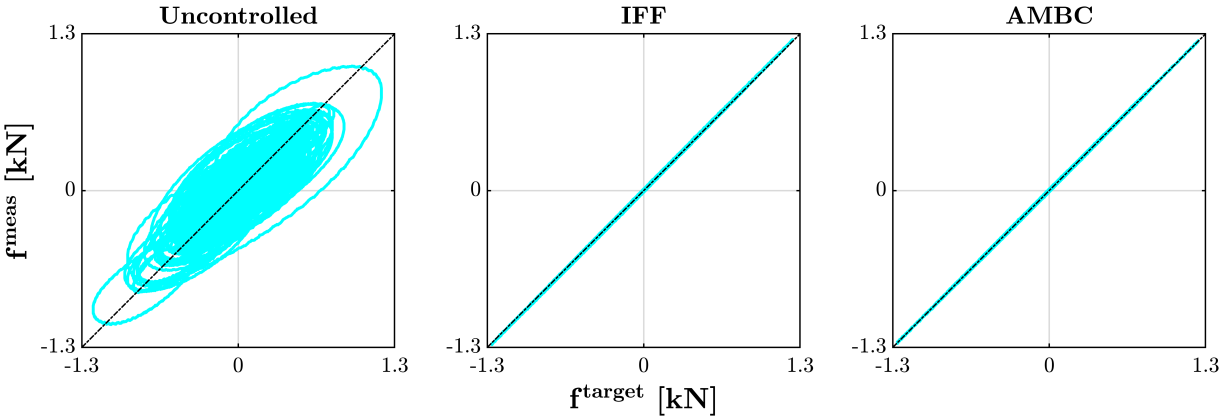


Figure 6.4: SSP plot of force tracking for El Centro earthquake.

Figure 6.5 shows the evolution of the adaptive gains in the simulation for the AMBC case. At approximately 5 [s], the gains undergo a variation in their values due to the beginning of the earthquake, where the gain a_0 exhibits the greatest variation in the adaptive compensation process in comparison with the other gains.

The gains show rapid adaptation as they adjust and tend to remain constant from the beginning of the seismic event and throughout its development. Then, approximately from 35 seconds onwards, when the seismic motion diminishes, a slight change in the gains is observed once again.

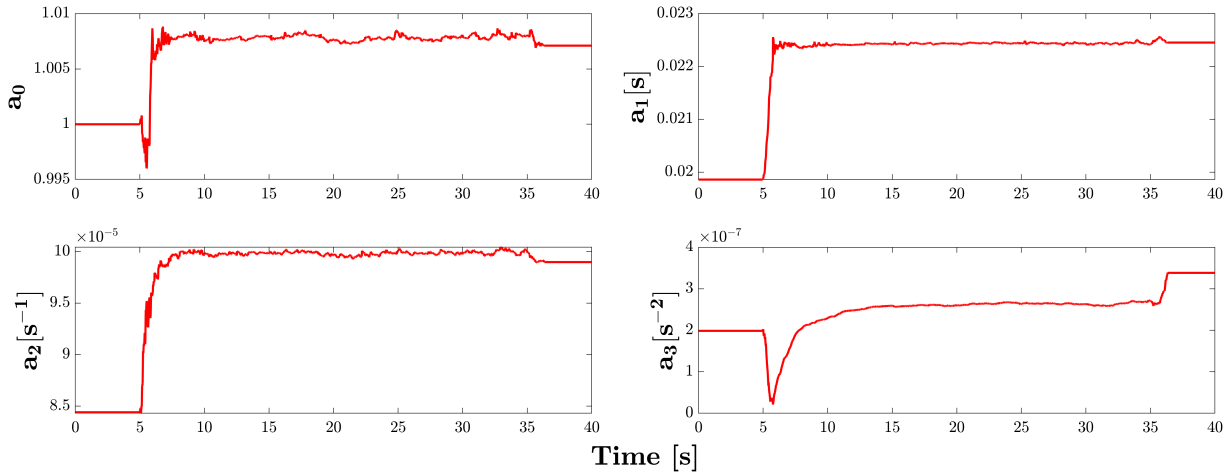


Figure 6.5: Gains adaptation results for El Centro earthquake.

On the other hand, the study conducted by Gálmez and Fermandois (2022) mentions that the adaptive process of the gains can be interpreted as online system identification, as the initial conditions of the compensator correspond to the initial model of the actuator without the specimen or the compliance spring, as shown in equation (5.10). It is observed that before the seismic event, the adaptive gains have the same value as the respective coefficients of the initial control plant. However, they vary at the onset of the seismic motion and tend to converge to another value during the duration of the movement. This difference is mainly due to the use of a control plant estimated with fewer poles and no zeros compared to the complete control plant that has 7 poles and 2 zeros. This results in the estimated transfer function not capturing the entire dynamics of the control plant. Additionally, another possible cause is that, due to uncertainty, the nominal and experimental values of the compliance spring are different. The considerations made for the design of the controller are shown in the development of Chapter 4.3.

6.2.1.1. Compliance spring variability

Before applying uncertainty to the numerical and experimental parameters, 100 simulations are performed using different values for the experimental compliance spring stiffness, using the following settings: nominal values for all physical parameters, IFF and AMBC controllers, and seismic records: CEN, MAU, and KOB. Experimental stiffness values ranging from 0.005% to 200% of the nominal stiffness k_{cn} are considered, evenly distributed, i.e., $k_{ce} \in [0.005\% k_{cn} \sim 200\% k_{cn}]$. In each simulation, the performance estimators J_2 and A_2 are calculated to explicitly observe what happens when the experimental compliance stiffness differs from the defined nominal value.

Figure 6.6 and 6.7 show the obtained values of the performance estimator J_2 and A_2, respectively, where the X axis represents the stiffness ratios used. It is observed that, for all three seismic records, the results of the J_2 estimator with the AMBC controller are lower than those obtained with the IFF. For the seismic record of El Centro, results are observed to have no clear trend as seen for the other two earthquakes, and a minimum value is evident for a stiffness ratio close to 1. This may be attributed to the controller being designed with the El Centro earthquake, making it more sensitive when used with this particular seismic record. In summary, better results of the J_2 estimator are obtained using the AMBC for the defined stiffness ratios.

The results obtained for the estimator A_2 only show differences in the minimum values obtained for the 3 seismic records presented. This means that in the case of the estimator A_2, good results will only be obtained when there is less uncertainty between the nominal and experimental values.

It is important to mention that from the results obtained and shown in Figures 6.6 and 6.7, it can be inferred that despite using large uncertainty values, the AMBC controller will be capable of maintaining good tracking between the displacements. This is because the target displacement is obtained with the nominal stiffness of the compliance, while the measured displacement is directly obtained from the actuator piston, i.e., the experimental stiffness is not used to obtain the measured displacement.

On the other hand, the increase in the value of the estimator A_2 as the stiffness ratio deviates from unity is due to the fact that the force measured in the compliance spring is different from the commanded force, as they are calculated with the nominal and experimental stiffness, respectively.

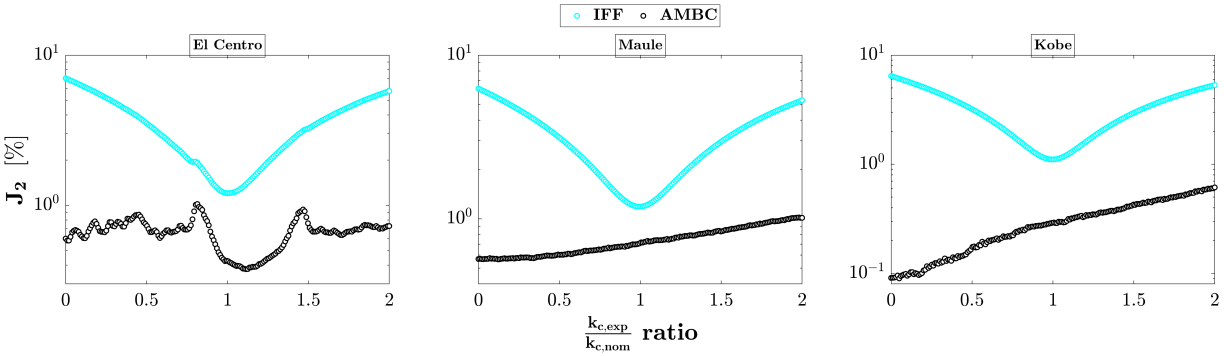


Figure 6.6: J_2 for different k_c ratio values with IFF and AMBC compensation.

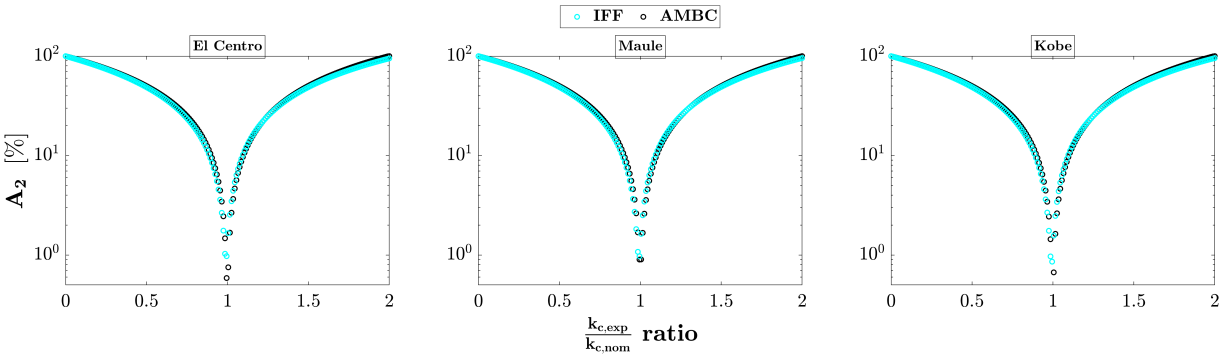


Figure 6.7: A_2 for different k_c ratio values with IFF and AMBC compensation.

6.2.1.2. Measurement noise impact

To observe the impact of the LVDT and load cell noise before the simulations with uncertainty, a single simulation is performed for each case of *Noise source* shown in Table 6.3. Nominal parameters are considered with El Centro seismic record. The results of the estimators J_2 and A_2, along with their respective time delays, are shown in Table 6.3.

Both sensors are modeled as white noise, for which their parameters are defined to obtain a signal-to-noise ratio (SNR) greater than 30 in each simulation. This value indicates that the measured signal has good quality despite being contaminated with noise.

The results show that sensor noise directly impacts the controller's performance, with J_2 being the most affected in the case of both sensors having measurement noise. However, considering the results obtained in the previous section, it is expected that adding uncertainty in the experimental compliance stiffness will cause an increase in the value of A_2, as the ratio between the experimental stiffness and the nominal stiffness deviates from unity.

Table 6.3: Tracking results with different noise sources

Noise source	Case	$\mathbf{J_2}$ [%]	$\mathbf{A_2}$ [%]	Force $\tau_{\mathbf{d}}(\mathbf{x^t} - \mathbf{x^m})$ [ms]	$\tau_{\mathbf{d}}(\mathbf{f^t} - \mathbf{f^m})$ [ms]
	UNC	58.74	60.79	18.96	19.55
Noisless	**IFF**	1.24	1.30	0.225	0.237
	AMBC	0.42	0.44	$8.26 \cdot 10^{-3}$	$8.65 \cdot 10^{-3}$
	UNC	58.77	60.79	18.95	19.55
LVDT	**IFF**	2.29	1.30	0.228	0.237
	AMBC	1.99	0.49	$2.07 \cdot 10^{-3}$	$3.61 \cdot 10^{-3}$
	UNC	58.76	60.83	19.96	19.55
Load cell	**IFF**	2.00	2.45	0.226	0.236
	AMBC	1.49	2.00	$1.18 \cdot 10^{-2}$	$1.20 \cdot 10^{-2}$
LVDT	**UNC**	58.79	60.82	18.96	19.55
&	**IFF**	2.78	2.44	0.226	0.237
Load cell	**AMBC**	2.49	2.07	$2.59 \cdot 10^{-3}$	$1.13 \cdot 10^{-3}$

6.2.2. Simulations with uncertain parameters

For this section, 100 simulations are performed for each of the three cases of uncertainty in the numerical and experimental parameters $[10\%, 20\%, 50\%]$, implemented as detailed in section 5.2.1. Additionally, three cases are studied: without control, with IFF compensation, and with AMBC controller. Performance estimators J_2, A_2, R_2, R_{2A} and time delays of displacement and force signals are calculated. The analysis of the results will focus on the estimator J_2 and R_2 since the IFF and AMBC compensators were designed in the displacement domain, while the analysis of the estimator A_2 and R_{A2} aims to review the behavior of force signal tracking given the substructuring scheme used.

The first result corresponds to the estimators J_2 and A_2 for each of the simulations performed, considering three different levels of uncertainty in the parameters and various seismic records. Figures 6.8, 6.9, and 6.10 show the results for the El Centro, El Maule, and Kobe earthquakes, respectively, differentiated by color for control cases. The graphs are presented with the Y axis in a logarithmic scale.

For all three seismic records, the obtained results for J_2 and A_2 tend to exhibit the same behavior and order of magnitude, respectively. Firstly, for all three seismic records, the estimator J_2 shows a notable low dispersion as the level of uncertainty increases, in comparison to the results of the estimator A_2, which exhibits a highly variable behavior even with the lowest level of uncertainty considered. On the other hand, for all three seismic records under study, the J_2 obtained

from the AMBC controller has lower values than those obtained from the IFF controller, and this behavior is more pronounced for lower levels of uncertainty. Furthermore, the good performance of both controllers is observed, as the values of J_2 for all levels of uncertainty considered yield results below 10% in all simulations.

On the contrary, the performance of force tracking signals evaluated with the estimator A_2 shows more scattered results than those of the estimator J_2 because since the measured force signal is obtained from the actuator displacement and the experimental structure using the experimental compliance spring stiffness (including uncertainty) after the target force has been transformed into target displacement using the nominal compliance spring stiffness ($k_{c,nominal} \neq k_{c,experimental}$). Therefore, the larger the uncertainty, the worse the results will be, i.e., higher values of the estimator A_2 will be obtained.

To summarize the results shown in Figures 6.8, 6.9, and 6.10, the estimators R_2 and R_{2A} defined in Section 4.5 are calculated, which correspond to the averages of the values J_2 and A_2, respectively. Additionally, the standard deviation, denoted as $s(R_2)$ and $s(R_{2A})$, is calculated for each estimator. The results of R_2 and R_{2A} are presented in Tables 6.4 and 6.5, respectively. Despite the poor results obtained from the estimator A_2, it is observed that in none of the cases does it exceed 100%.

Table 6.4 shows that the values of R_2 with the AMBC controller are lower than those obtained by the IFF controller for all seismic events and uncertainties. Additionally, it is noted that the standard deviation for the controlled cases, considering all seismic events, is less than one, which means that the dispersion is very low and all the results of J_2 are close to R_2, even for high uncertainty values.

Table 6.5 presents the results of the estimator R_{2A}, where it can be observed that none of the presented cases yield a value lower than 1%. Additionally, R_{2A} values increase as the uncertainty grows, unlike the case of displacements presented in Table 6.4. The compensated results exhibit even greater dispersion than those obtained from the uncontrolled simulations. The cause of these results is the difference between the nominal and experimental stiffness values caused by the uncertainty, as explained in the preceding paragraphs.

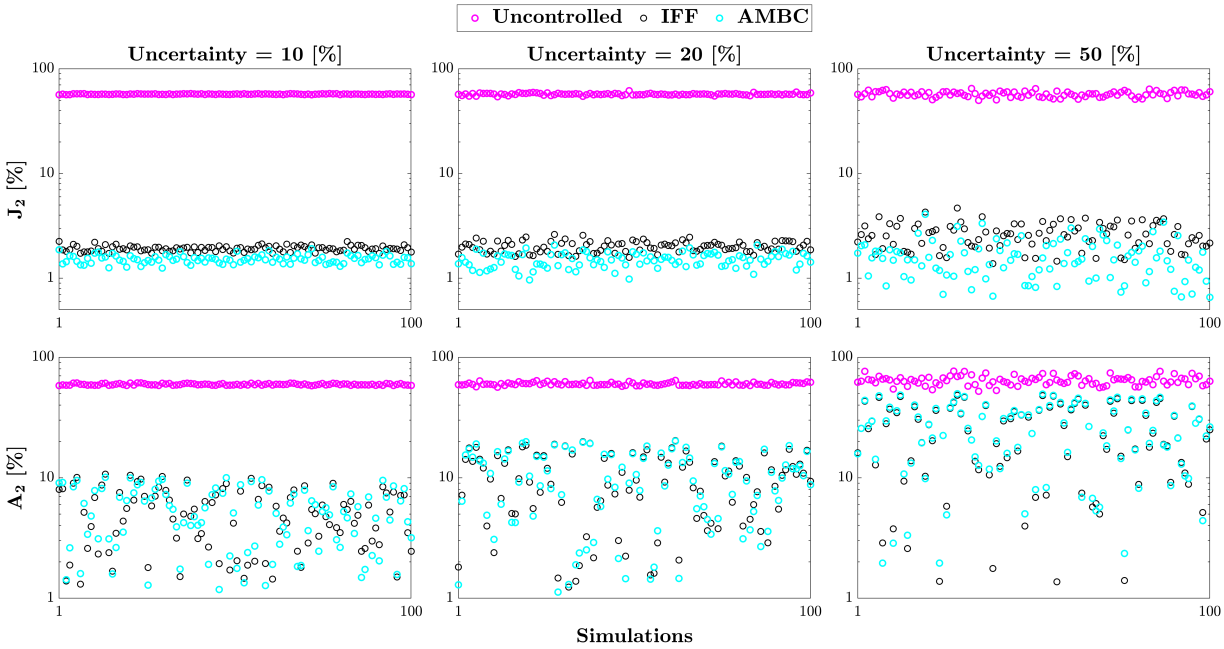


Figure 6.8: J_2 & A_2 values from 100 simulations for each control case and El Centro earthquake.

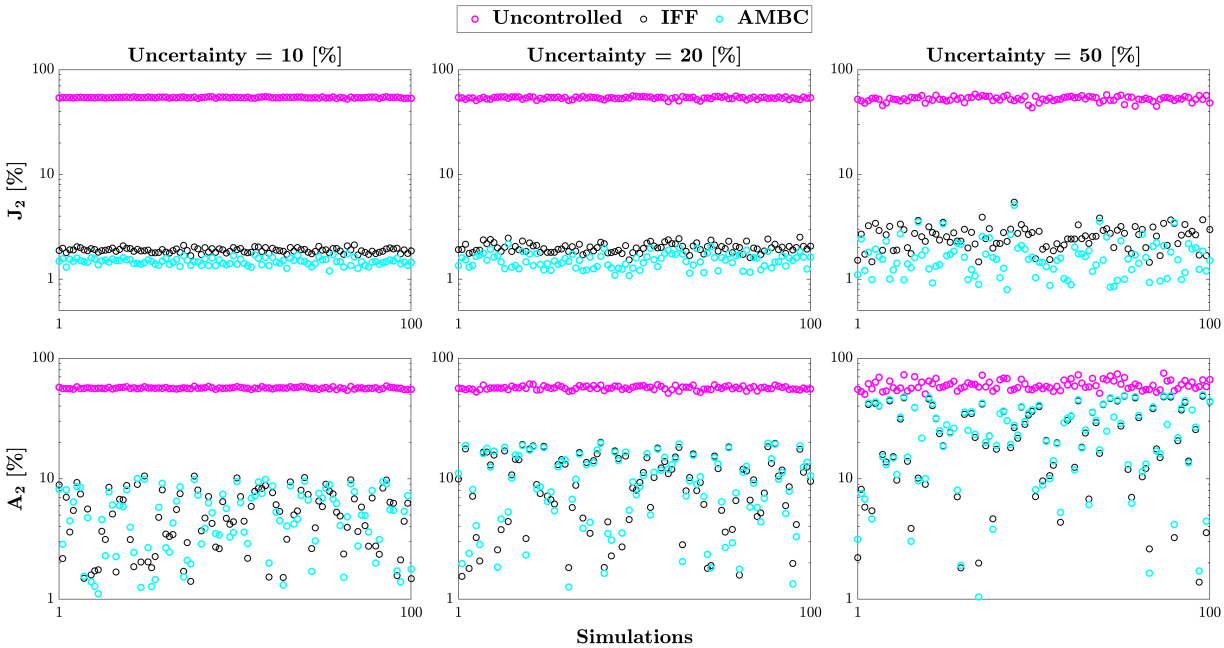


Figure 6.9: J_2 & A_2 values from 100 simulations for each control case and El Maule earthquake.

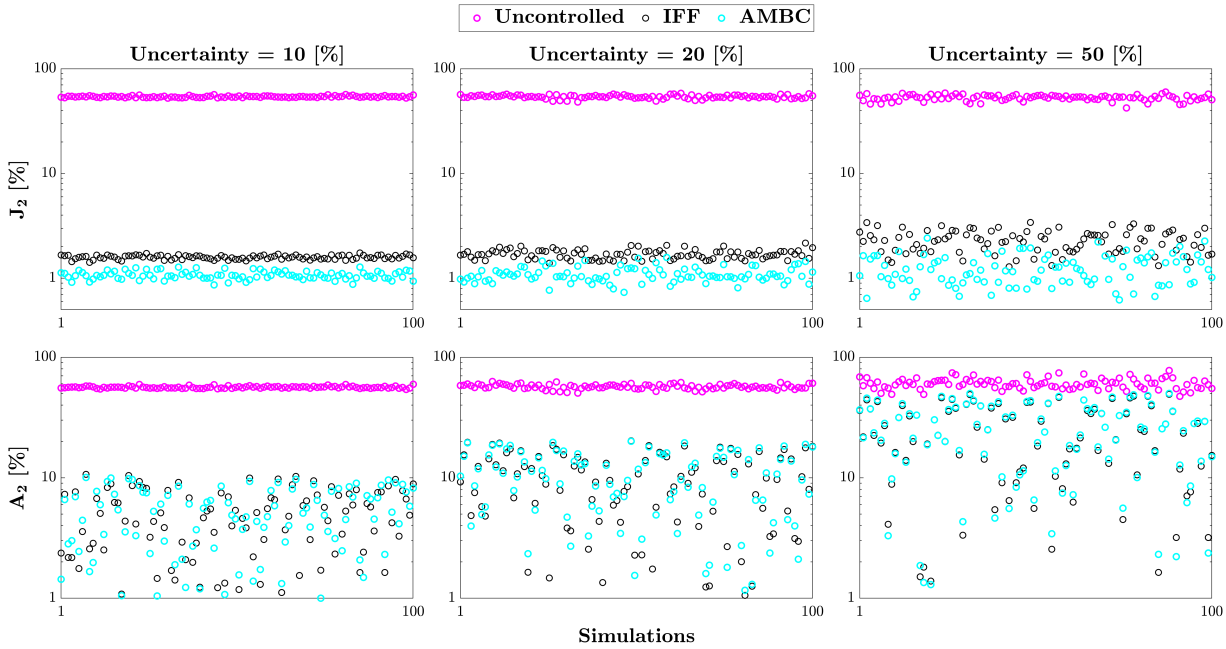


Figure 6.10: J_2 & A_2 values from 100 simulations for each control case and Kobe earthquake.

Table 6.4: R_2 and standard deviation (s) for each seismic record, control, and uncertainty cases.

Seismic Record	Uncertainty	Uncontrolled		IFF		AMBC	
		$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$	$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$	$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$ [%]
	10%	57.18	0.36	1.92	0.12	1.52	0.14
CEN	20%	57.13	1.17	2.02	0.24	1.50	0.24
	50%	57.13	3.44	2.61	0.72	1.61	0.68
	10%	54.26	0.52	1.89	0.10	1.47	0.11
MAU	20%	53.75	1.32	2.01	0.21	1.52	0.24
	50%	52.53	3.04	2.59	0.66	1.66	0.70
	10%	54.06	0.93	1.59	0.07	1.07	0.09
KOB	20%	54.04	2.04	1.70	0.17	1.09	0.18
	50%	52.82	3.33	2.28	0.55	1.23	0.41

Table 6.5: R_{2A} and standard deviation (s) for each seismic record, control, and uncertainty case.

Seismic Record	Uncertainty	Uncontrolled		IFF		AMBC	
		$\mathbf{R_{2A}}$ [%]	$\mathbf{s(R_{2A})}$	$\mathbf{R_{2A}}$ [%]	$\mathbf{s(R_{2A})}$	$\mathbf{R_{2A}}$ [%]	$\mathbf{s(R_{2A})}$ [%]
	10%	59.43	0.87	5.27	2.66	5.30	2.71
CEN	20%	60.00	1.65	9.96	5.51	10.17	5.68
	50%	64.23	5.57	24.46	14.01	25.10	14.40
	10%	56.49	0.88	5.24	2.66	5.24	2.71
MAU	20%	56.66	2.02	9.98	5.49	10.16	5.63
	50%	60.17	5.86	24.57	14.10	25.11	14.44
	10%	56.29	1.09	5.15	2.73	5.15	2.79
KOB	20%	56.96	2.60	9.90	5.57	10.07	5.74
	50%	60.43	6.09	24.52	14.15	25.06	14.48

From the plots in Figures 6.8, 6.9, and 6.10, boxplots are constructed to observe the dispersion of the results from a different perspective and make comparisons. Figures 6.11, 6.12, and 6.13 correspond to the results of the estimator J_2 for the three seismic records and control cases, while Figures 6.14, 6.15, and 6.16 correspond to the results of the estimator A_2. for the same cases than J_2.

The boxplots for the estimator J_2 show excellent results for the AMBC controller in comparison with the IFF controller. The AMBC boxplots for the three seismic records show that both the box and its whiskers expand towards higher and lower values of J_2 as the uncertainty increases, while the median remains almost constant. On the other hand, the boxplots for the IFF controller show that the boxes only grow towards higher values of J_2, and the medians also shift upwards. The boxplots demonstrate the robustness of the AMBC controller over the IFF controller, as the AMBC results tend to remain constant regardless of increasing uncertainty.

The boxplots of the estimator A_2 demonstrate that force tracking worsens as uncertainty increases for the three seismic records and control cases. These results allow us to establish that the factor causing poor performance in force signal tracking is the added uncertainty in the compliance spring.

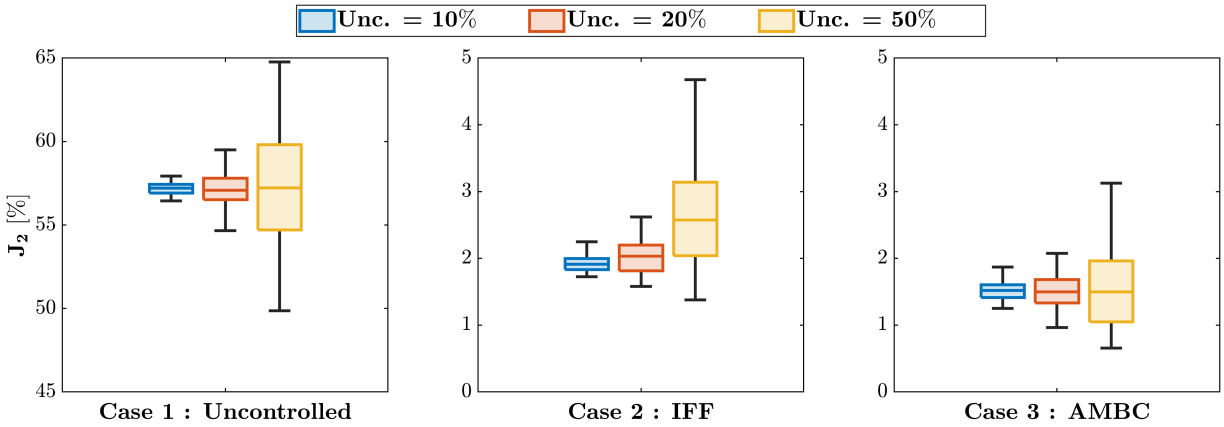


Figure 6.11: J_2 Boxplot results for 100 simulations using El Centro seismic record.

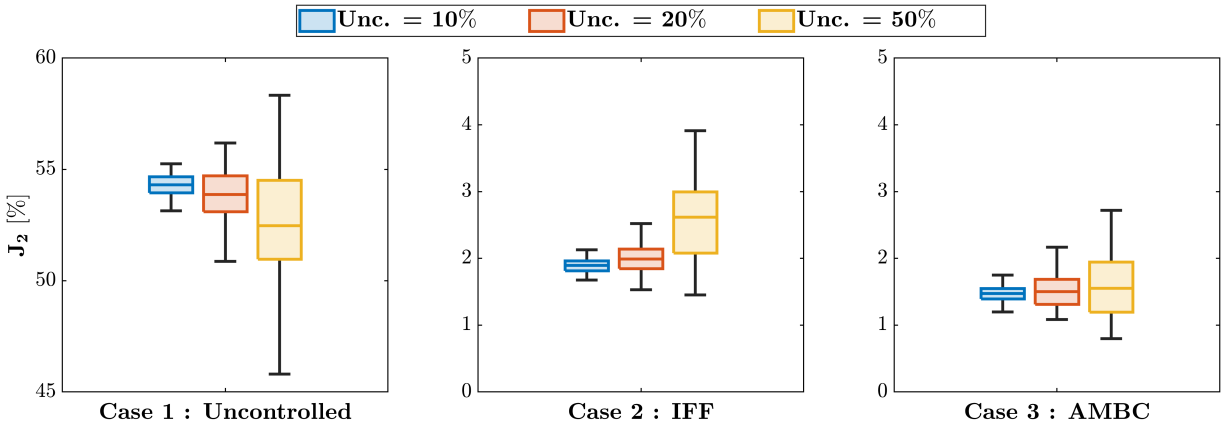


Figure 6.12: J_2 Boxplot results for 100 simulations using El Maule seismic record.

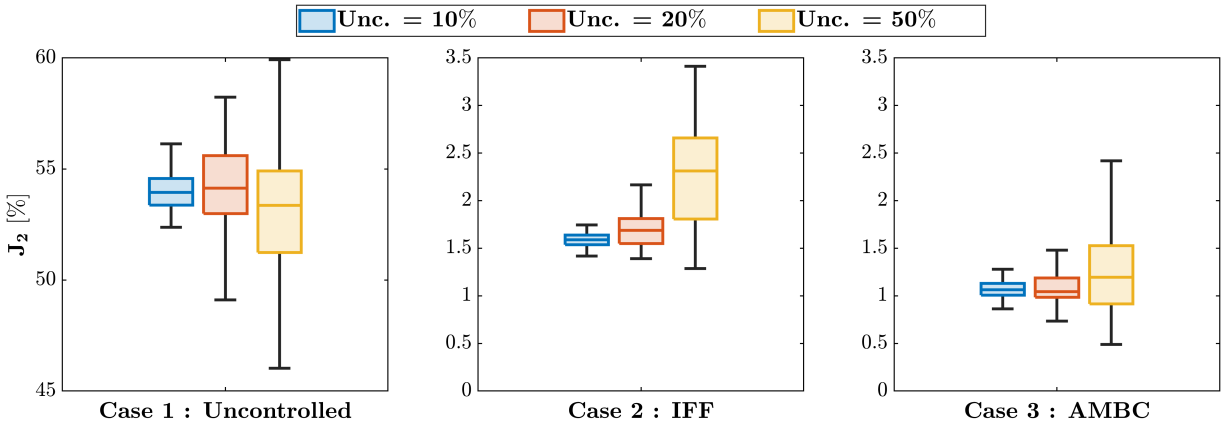


Figure 6.13: J_2 Boxplot results for 100 simulations using Kobe seismic record.

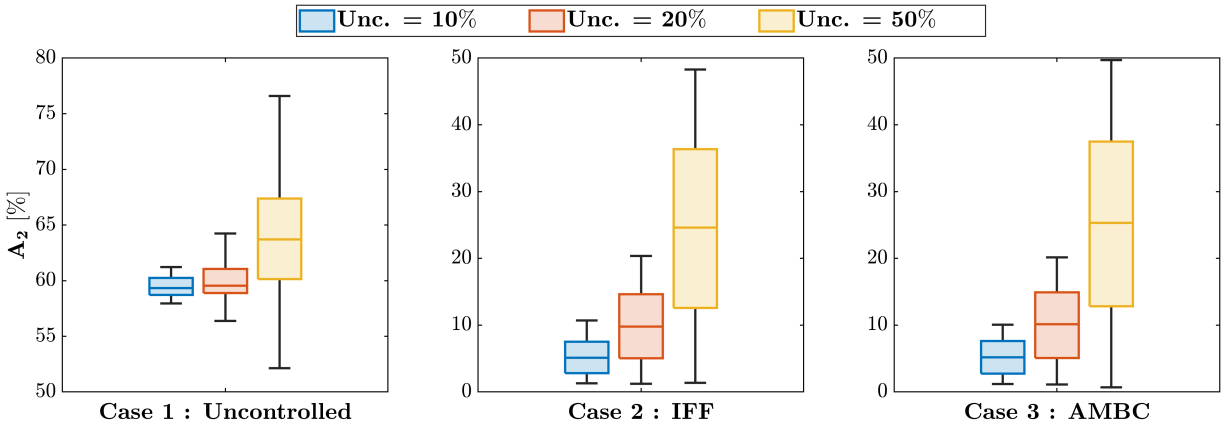


Figure 6.14: A_2 Boxplot results for 100 simulations using El Centro seismic record.

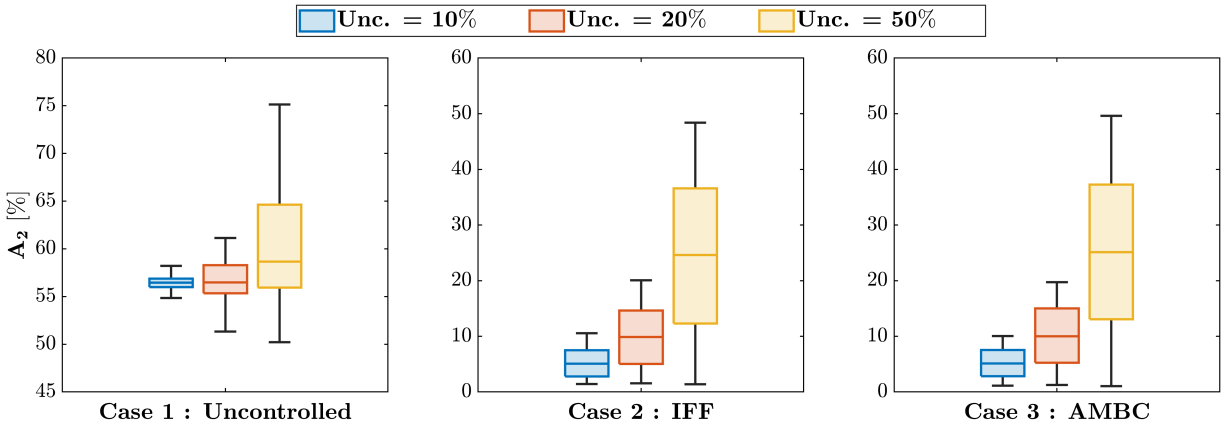


Figure 6.15: A_2 Boxplot results for 100 simulations using El Maule seismic record.

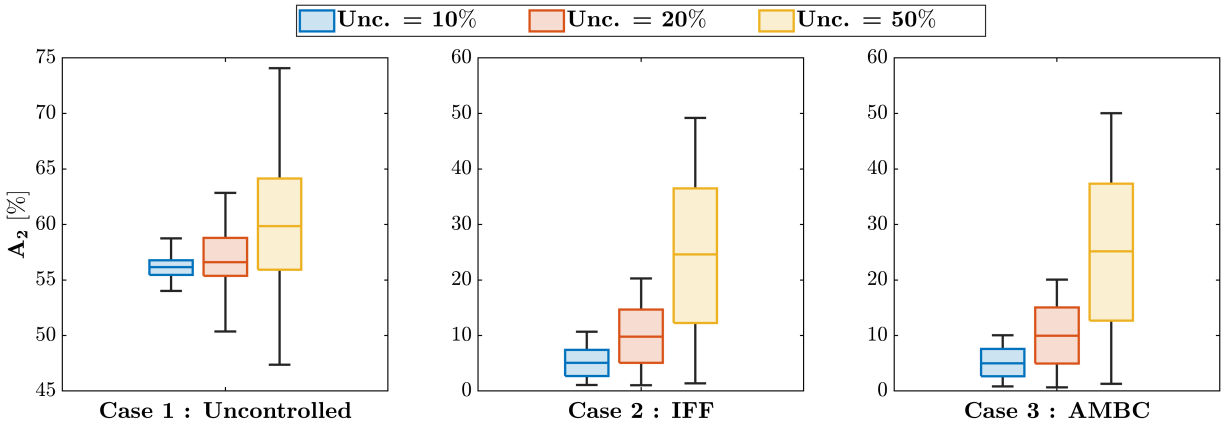


Figure 6.16: A_2 Boxplot results for 100 simulations using Kobe seismic record.

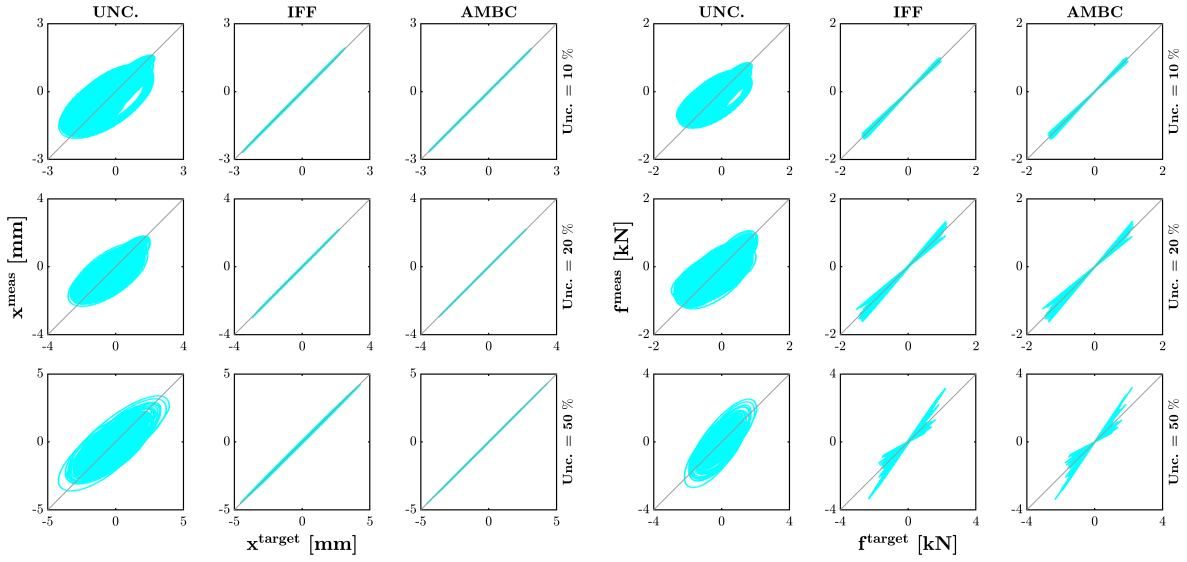
In Figures 6.17, 6.18, and 6.19, the Synchronization Subspace Plots (SSP) (Ashwin, 1998, Wallace et al., 2005a) are shown to analyze the tracking of displacements and forces, respectively. The graphs present the results of 100 simulations for each control and uncertainty case. In summary,

the analysis consists of assessing whether the displayed graph closely resembles the identity function, which indicates good signal tracking. Conversely, if the graph deviates from the identity function, it indicates poor signal tracking. Additionally, the following aspects are analyzed:

- If the graphs deviate from the identity function, forming an ellipse, then there are issues with time delay.
- If the slope of the graph is less than 1, then there is an undershoot, as the measured signal does not reach the target signal.
- If the slope of the graph is greater than 1, then there is an overshoot, as the measured signal exceeds the target signal.

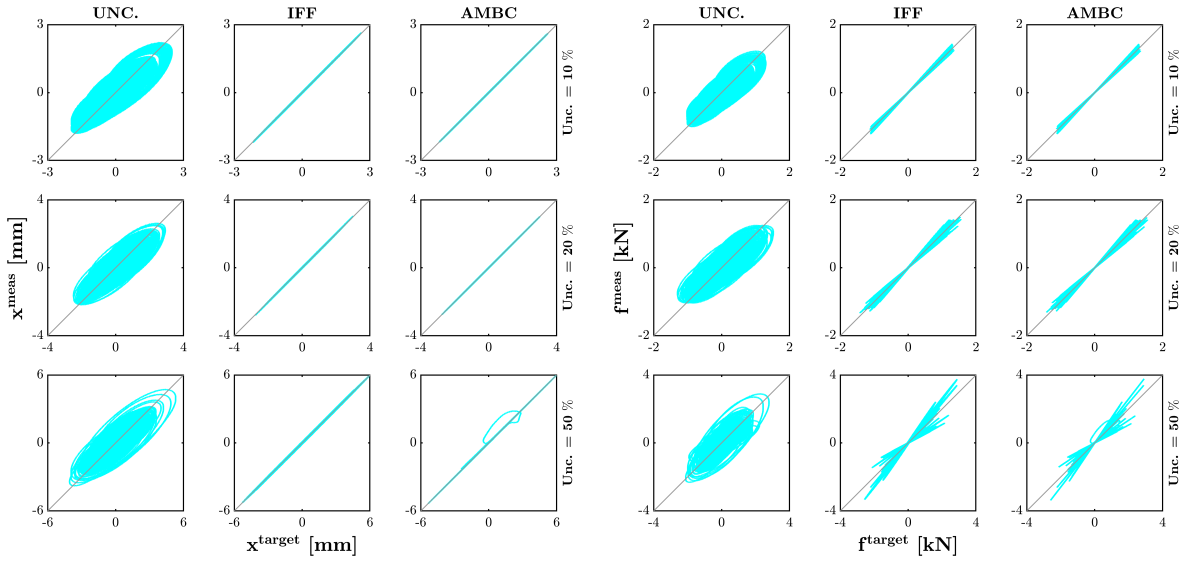
The results for the displacement tracking show that both IFF and AMBC controllers exhibit excellent performance for all three seismic records, even in the case with 50% uncertainty.

On the other hand, the force tracking graphs for the target and measured signals show differences as they deviate from the identity function. Since the graphs share a common point (0,0), it can be determined that there is almost no time delay between the signals. Furthermore, there are slopes both less and greater than 1 due to the uncertainty causing the experimental compliance spring stiffness values to differ from the nominal value for each simulation. Thus, it is deduced that in cases where the nominal stiffness is lower than the experimental stiffness ($k_{c,nom} < k_{c,exp}$), when the piston imposes displacement on the compliance spring, the measured force will be lower than the target force, resulting in undershoot. Conversely, it is extended that in situations where $k_{c,nom} > k_{c,exp}$, overshoot occurs. The aforementioned explanation justifies the results obtained for force tracking.



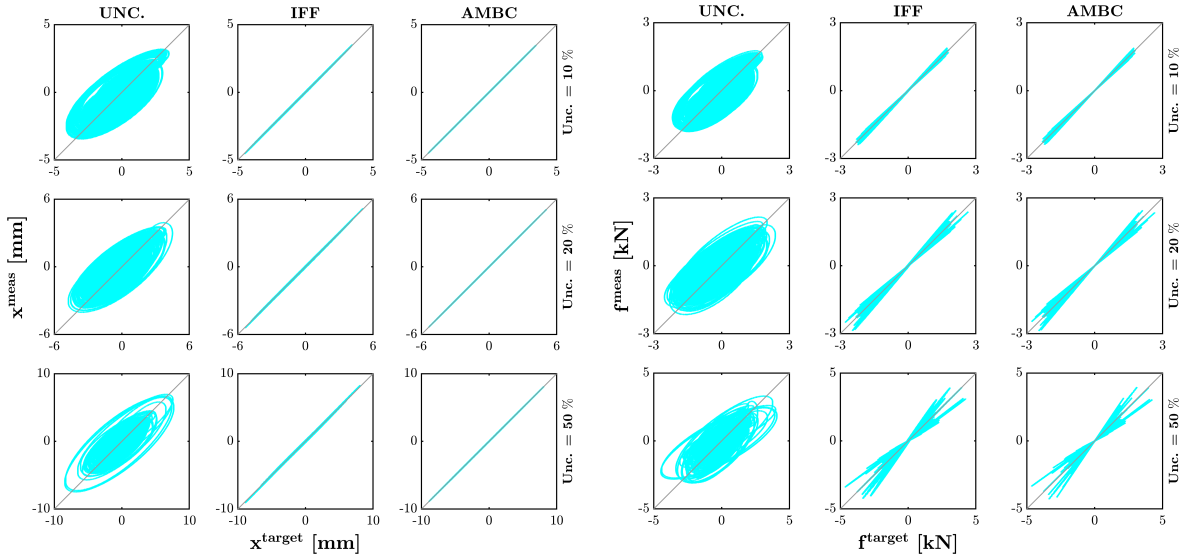
(a) Displacement tracking. **(b)** Force tracking.

Figure 6.17: SSP plots for El Centro earthquake with different control and uncertainty cases.



(a) Displacement tracking. **(b)** Force tracking.

Figure 6.18: SSP plots for El Maule earthquake with different control and uncertainty cases.



(a) Displacement tracking. **(b)** Force tracking.

Figure 6.19: SSP plots for Kobe earthquake with different control and uncertainty cases.

In Figures 6.20, 6.21, and 6.22, the evolution of the adaptive gains from conducting 100 simulations are shown in gray for each earthquake and uncertainty case. The median and the average are also plotted to observe the trend of the result dispersion.

An important aspect to highlight in the analysis of the adaptive gains is the fact that at the beginning of the seismic event, the variation of each gain, referred to as "adaptation", occurs very rapidly, around 1 or 2 seconds approximately, and then remains nearly constant until the end of the seismic motion. This behavior is clearly observed in the graphs shown for the three seismic cases. It is also observed that with higher uncertainty, the variation of the adaptive gains is also greater.

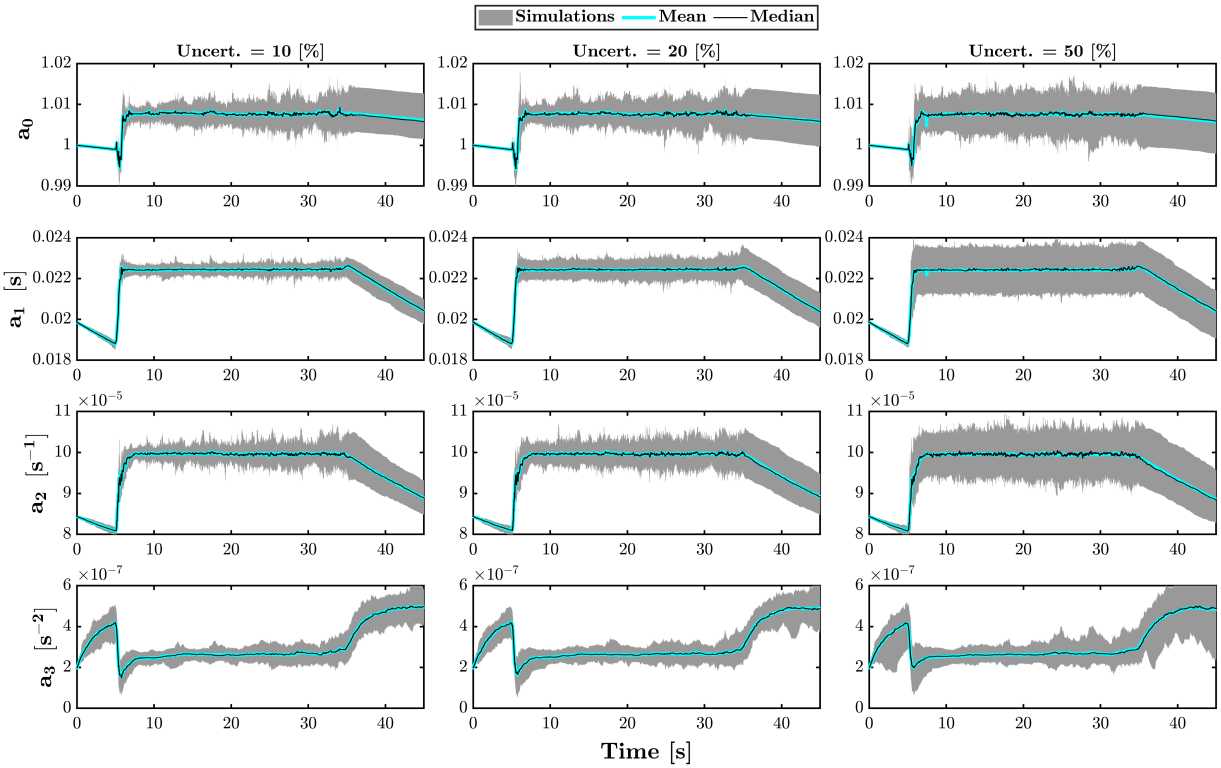


Figure 6.20: AMB gains evolution for El Centro earthquake.

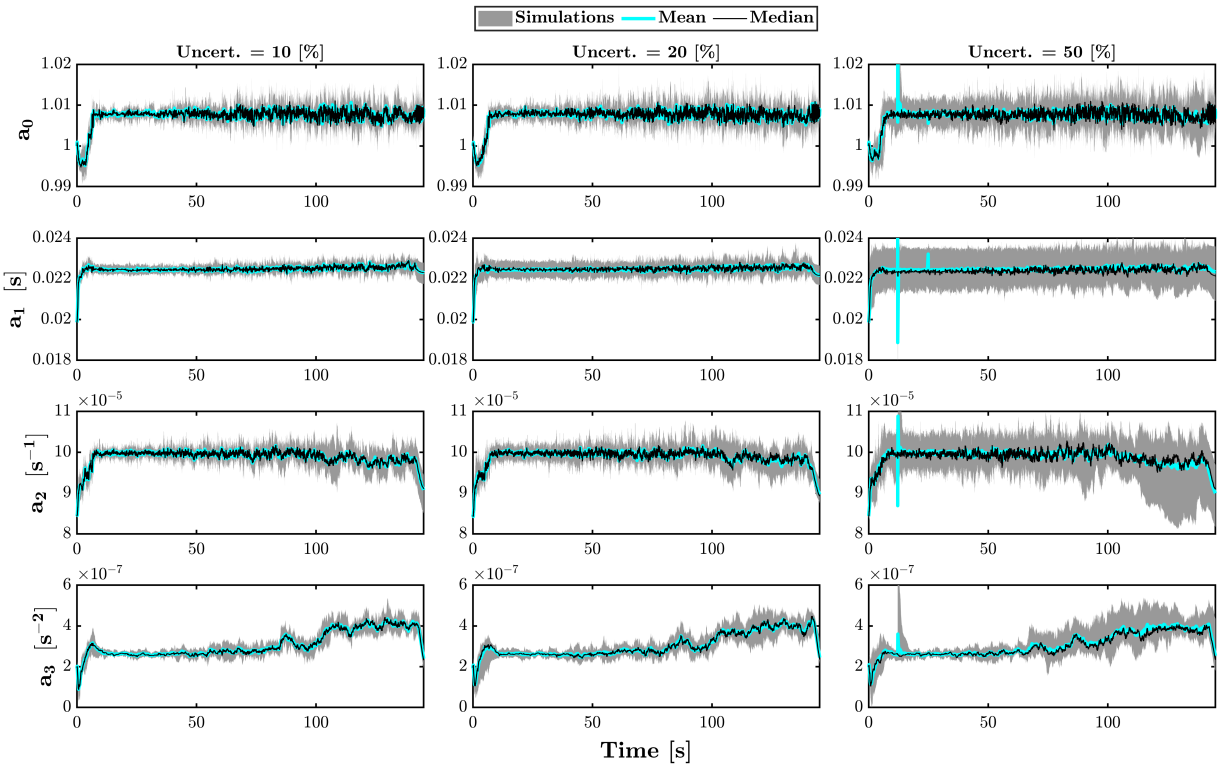


Figure 6.21: AMB gains evolution for El Maule earthquake.

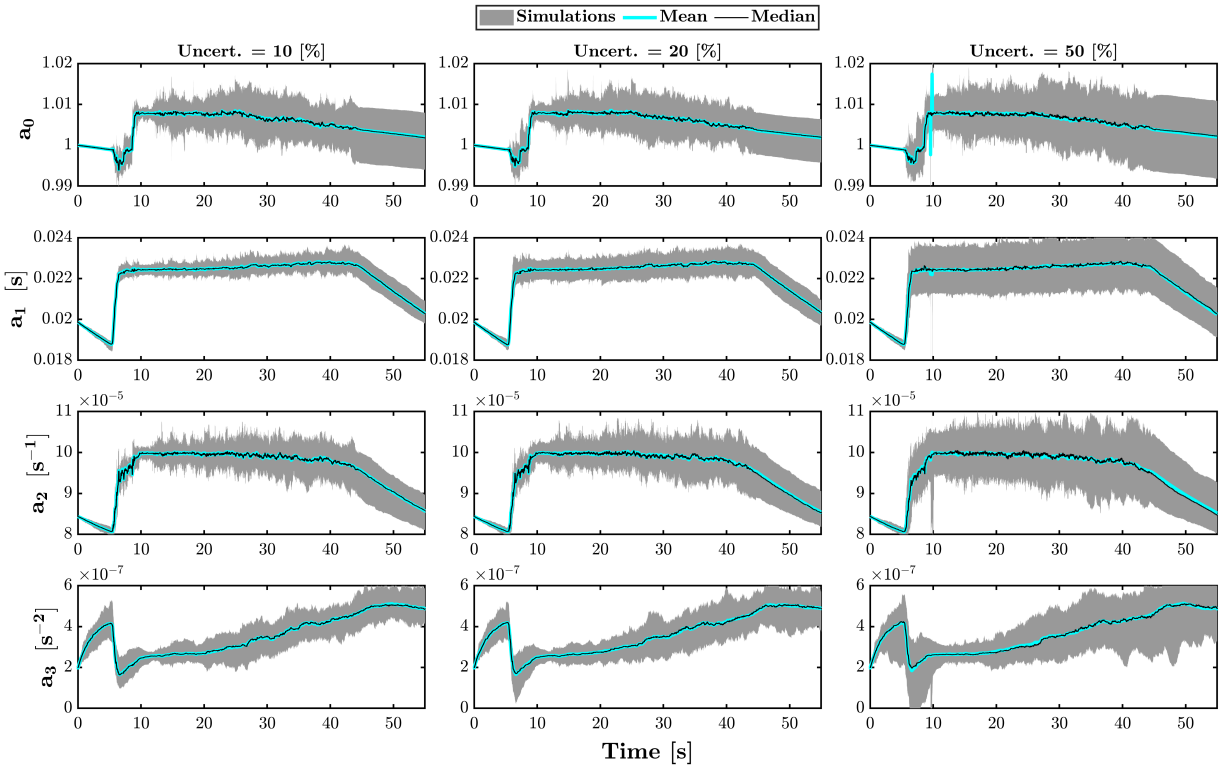


Figure 6.22: AMB gains evolution for Kobe earthquake.

6.2.2.1. Compliance spring without uncertainty

To verify that the uncertainty in the compliance spring stiffness is the main cause of the high values of the estimator A_2, 100 simulations are performed for each control case without adding uncertainty to the compliance spring ($P_{unc}(k_c) = 0$), while maintaining uncertainty in the rest of the parameters. This case will only be analyzed for the El Centro seismic record, as the objective is to examine the impact of the uncertainty.

Figure 6.23 presents the results of the estimators J_2 and A_2 when considering $P_{unc}(k_c) = 0$. It can be observed that the values obtained for J_2 maintain the same behavior as the previous results. However, the results of the estimator A_2 have improved considerably, as even when considering 50% uncertainty in the parameters, values of A_2 between approximately 1% and 2% are obtained.

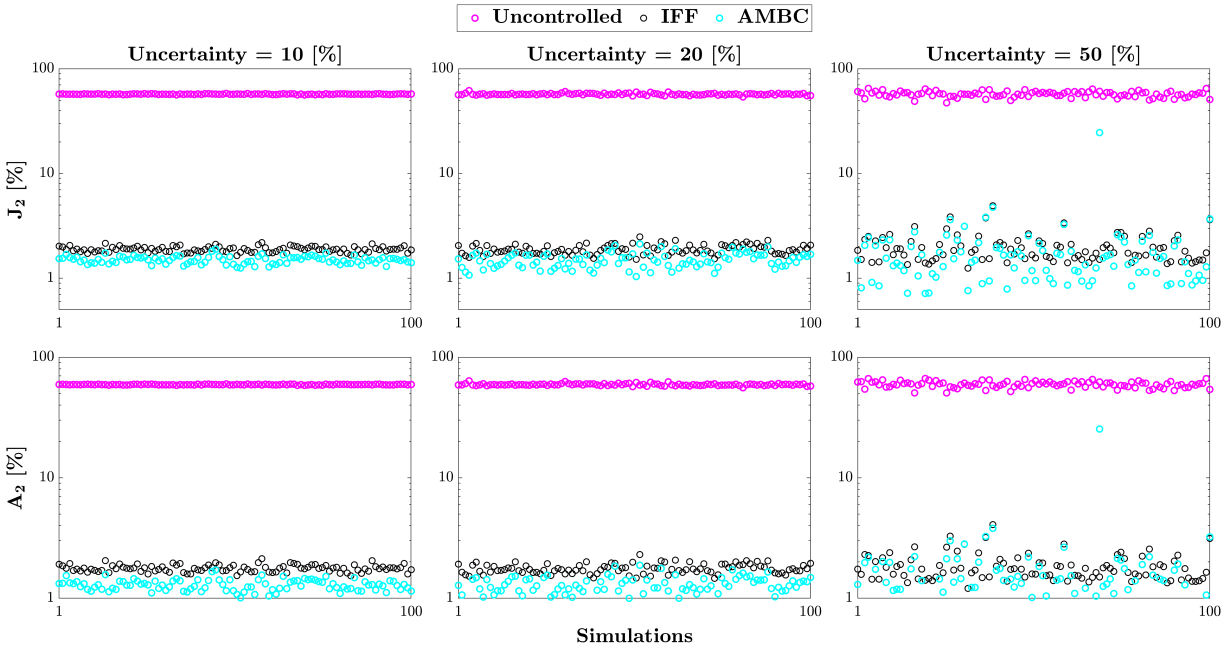


Figure 6.23: J_2 & A_2 values for El Centro earthquake with $P_{unc}(k_c) = 0$.

Tables 6.6 and 6.7 display the results of the estimators R_2 and R_{2A} obtained from the values shown in Figure 6.13. It can be observed that the results of J_2 remain within the same order of magnitude as those presented in Table 6.4. On the other hand, the values of R_{2A} decrease to the same order of magnitude as the values of R_2, decreasing in the case with 50% uncertainty from $R_{2A} = 25.1\%$ to $R_{2A} = 1.62\%$. Similarly, the dispersion decreases to the same order as those obtained for R_2 in Table 6.6.

Table 6.6: R_2 and standard deviation (s) for each control case, El Centro seismic record and $P_{unc}(k_c) = 0$.

Seismic Record	Uncertainty	Uncontrolled		IFF		AMBC	
		$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$	$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$	$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$ [%]
	10 %	57.19	0.37	1.89	0.11	1.50	0.12
CEN	20 %	57.16	1.17	1.89	0.20	1.49	0.24
	50 %	57.13	3.68	1.99	0.63	1.84	2.42

Table 6.7: R_{2A} and standard deviation (s) for each control case, El Centro seismic record and $P_{unc}(k_c) = 0$.

Seismic Record	Uncertainty	Uncontrolled		IFF		AMBC	
		$\mathbf{R_{2A}}$ [%]	$\mathbf{s(R_{2A})}$	$\mathbf{R_{2A}}$ [%]	$\mathbf{s(R_{2A})}$	$\mathbf{R_{2A}}$ [%]	$\mathbf{s(R_{2A})}$ [%]
	10 %	59.23	0.36	1.77	0.12	1.29	0.14
CEN	20 %	59.22	1.16	1.76	0.16	1.28	0.21
	50 %	59.34	3.59	1.84	0.49	1.62	2.48

Using the results shown in Figure 6.23, boxplots are constructed for J_2 and A_2, displayed in Figures 6.24 and 6.25, respectively, for each control case. The boxplots indicate that the AMBC controller maintains better results than the IFF controller. The values of A_2 decrease to the point where they are within the same order of magnitude as the results shown for R_2. Furthermore, it is observed that the results of the AMBC controller, for both J_2 and A_2, exhibit slightly higher dispersion than the results obtained using the IFF controller. This outcome is also evident in the standard deviation values displayed in Table 6.7.

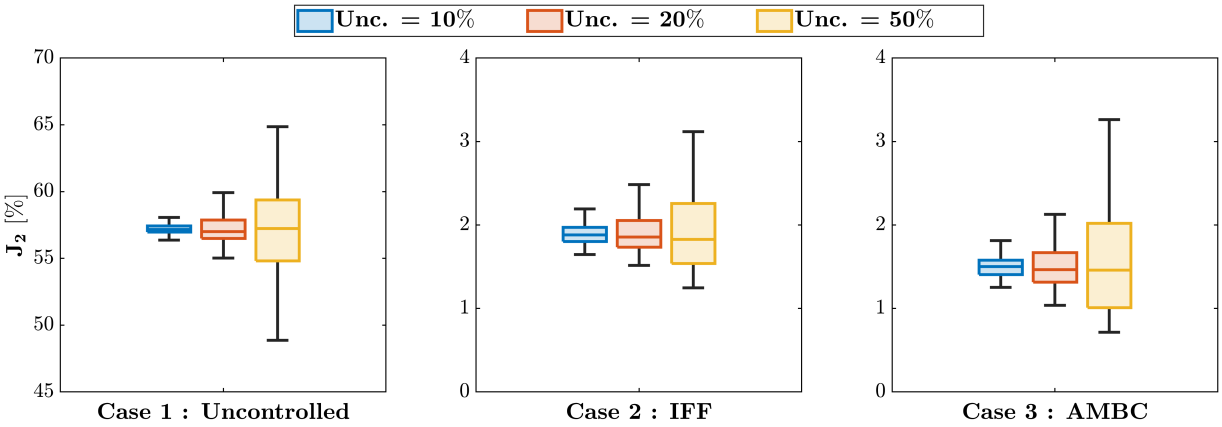


Figure 6.24: J_2 Boxplot results for 100 simulations with different control and uncertainty cases.

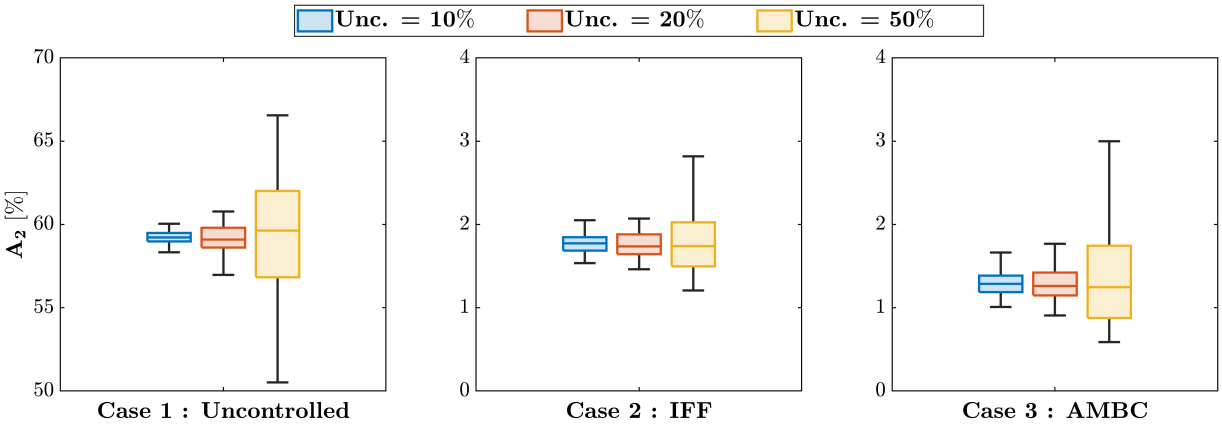


Figure 6.25: A_2 Boxplot results for 100 simulations with different control and uncertainty cases.

Thus, it is demonstrated that the factor that most influence the growth of the A_2 estimator is the uncertainty in the compliance spring.

6.2.2.2. Uncertainty applied to actuator parameters

As a final analysis, the effect of adding uncertainty in the parameters of the actuator shown in Table 4.5 is studied. 100 simulations are conducted considering three cases of uncertainty in the parameters and three seismic records. Two cases of uncertainty in the actuator parameters are considered: 5% and 10%. The results are shown from Figure 6.26 to Figure 6.28.

In Figures 6.26 and 6.27, the obtained J_2 and A_2 estimators are shown. It is important to highlight that with only a 5% uncertainty, the results obtained for the IFF controller worsen

compared to those obtained in the previous sections, deviating from those obtained by the AMBC controller.

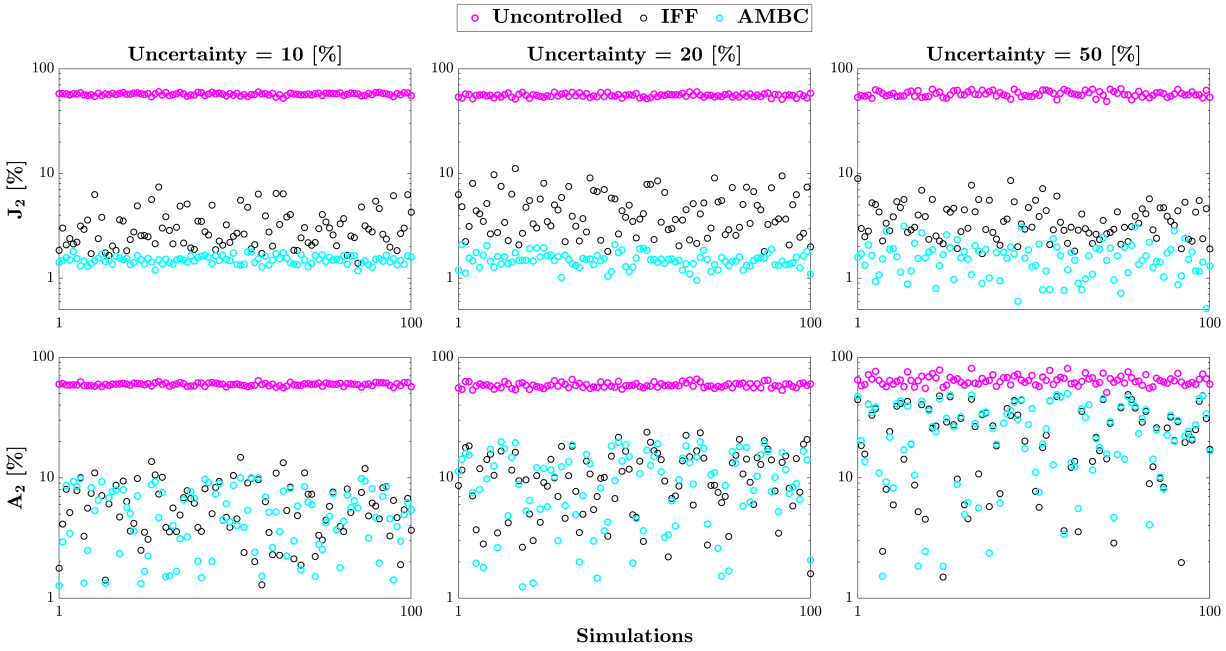


Figure 6.26: J_2 and A_2 for 100 simulations with $P_{unc} = 5\%$ uncertainty applied to the actuator parameters.

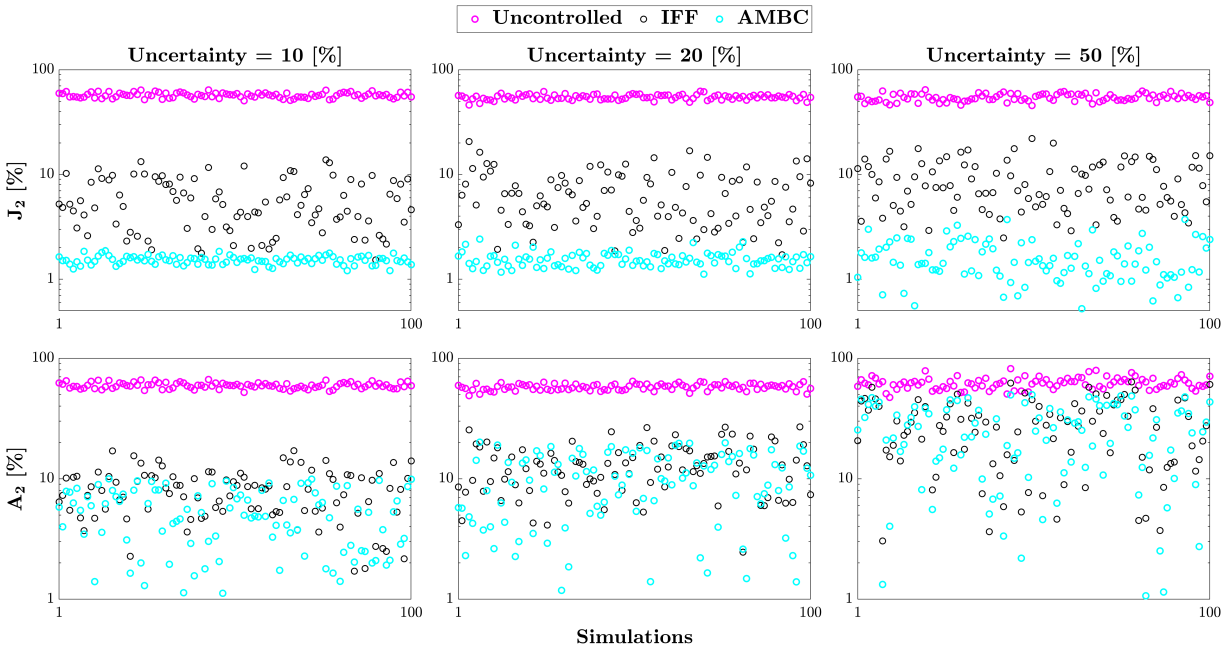


Figure 6.27: J_2 and A_2 for 100 simulations with $P_{unc} = 10\%$ uncertainty applied to the actuator parameters.

In Table 6.8, the R_2 estimator and the standard deviation are calculated for the cases shown in Figures 6.26 and 6.27. It is observed that the results for R_2 remain within the same range as the previous results. It is important to mention that the values of R_2 obtained for the AMBC

controller tend to be in the same order of magnitude as the results shown in the previous sections. In contrast, the results for R_2 obtained from the IFF controller are all greater than 3%.

Table 6.8: R_2 and standard deviation (s) for each control case and El Centro seismic record.

Actuator's Uncertainty	Uncertainty	Uncontrolled		IFF		AMBC	
		$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$	$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$	$\mathbf{R_2}$ [%]	$\mathbf{s(R_2)}$ [%]
5 %	10 %	57.15	1.51	3.13	1.34	1.49	0.13
	20 %	55.60	2.10	4.47	2.14	1.49	0.24
	50 %	57.46	3.61	3.70	1.49	1.58	0.58
10 %	10 %	57.13	3.21	5.78	3.14	1.52	0.14
	20 %	54.74	3.08	6.74	3.89	1.54	0.26
	50 %	54.37	4.37	8.72	4.43	1.60	0.69

Boxplots are constructed for each case of uncertainty in the actuator parameters and shown in Figures 6.28 and 6.29. It is observed that the results obtained with the IFF controller worsen and deviate from the good performance shown in the results of the previous sections, reaching data ranges greater than 10% in the case of uncertainty in the actuator parameters of $P_{unc} = 10\%$. On the contrary, it is observed that despite the added uncertainty in the actuator, the AMBC controller maintains excellent performance.

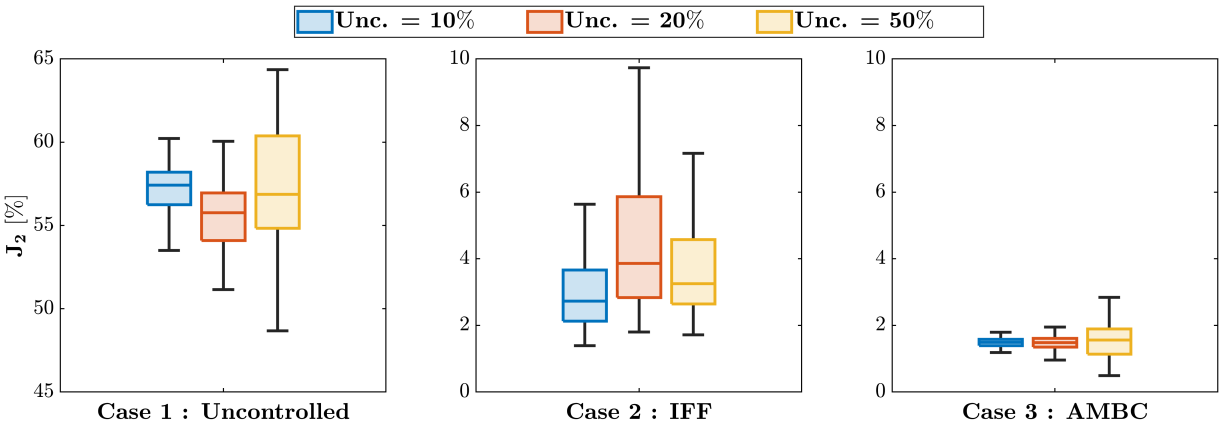


Figure 6.28: J_2 with actuator's uncertainty $P_{unc} = 5\%$.

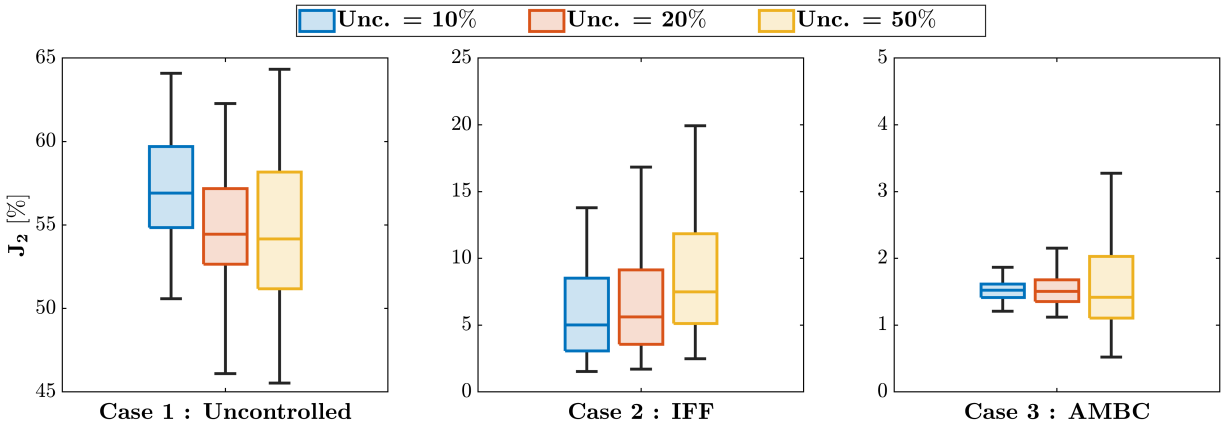


Figure 6.29: J_2 with actuator's uncertainty $P_{unc} = 10\%$.

The IFF controller is designed with the nominal values of the control plant, which includes the actuator, compliance spring, and specimen. Given the results reviewed in this chapter, it is observed that the IFF controller is highly sensitive to the variation of the parameters of the transfer system (actuator). However, a sensitivity analysis of the actuator parameters and their relation with the IFF controller is not conducted as it falls outside the scope of this study.

7 Conclusions

7.1. Summary

In this study, the implementation of the adaptive model-based compensator proposed by Gálmez and Fermandois (2022) for a force-based real-time hybrid simulation scheme using a compliance spring between the actuator and the specimen, and considering uncertainty in the numerical and experimental parameters, was presented.

Firstly, a substructuring approach compatible with the AMBC and using the compliance spring is studied and implemented. Then, an optimization process is conducted to obtain the adaptive parameters of the AMBC considering uncertainty in the specimen, the compliance spring, and the numerical parameters. Subsequently, simulations are performed to study the sensitivity of the obtained adaptive parameters.

Finally, simulations are carried out considering multiple cases, such as different seismic records, uncertainty levels, and compensation types. A sensitivity analysis of parameters of interest within the simulation scheme is conducted.

7.2. Conclusions

Based on the research conducted and the objectives outlined in Section 1.2, as well as the results presented in Chapter 6.2, the following conclusions are derived:

- The computational implementation of a force-based real-time hybrid simulation experiment is carried out virtually. It utilizes adaptive model-based compensation and considers uncertainties in the structure and compliance spring parameters.
- The optimization process algorithm is adapted and implemented to work in a displacement-based scheme within a force-based real-time hybrid simulation due to the utilization of the compliance spring. This allows for obtaining the adaptive gains of the AMBC. The sensitivity analyses performed on the optimal adaptive gains confirm that the results reduce the displacement tracking error J_2 and force error A_2 when considering a 20% uncertainty in the structure parameters used for optimization.
- The evidence indicates that the AMBC controller exhibits better performance than the IFF controller for simulations without uncertainty, as the results of J_2 and A_2 obtained with the AMBC controller are approximately 10 times lower than those obtained with the IFF controller. The difference between time delays shows similar behavior for AMBC compared with the IFF controller. For AMBC, values of J_2 and A_2 below 1% are achieved, and the time delays between target and measured signals are in the order of 10^{-2} [ms].
- One of the most important elements of the proposed simulation scheme is the compliance spring and its design process. Even a small difference between its nominal and experimental

stiffness values is sufficient to cause an increase in the parameter A_2, leading to a discrepancy between the measured force and the target force. In other words, the desired force is not being imposed on the specimen, resulting in an imbalance of forces.

- The results shown throughout section 6.2 demonstrate the robustness of the AMBC controller over the IFF controller, as the values obtained for the estimators J_2 and R_2 remained approximately within the same order of magnitude for all the studied cases, which include uncertainty in the structure's parameters, compliance spring, actuator, noise in the load cell and LVDT, and different seismic records. A different situation is observed for the IFF controller, which deteriorates considerably when uncertainty is added to the actuator's parameters.
- It is evident that the IFF controller is highly sensitive to variations in the actuator parameters. However, this analysis is not conducted as it falls outside the scope of the present study.
- It is concluded that the element that generates the most error in the simulation when considering uncertainty within the proposed framework is the transfer system (servohydraulic actuator), which with only 5% uncertainty, significantly worsens the performance of the IFF controller.

7.3. Future Work

The following are potential future studies that can be conducted based on the present research:

- This study is conducted considering the linear behavior of the materials in the reference structure and its substructures. A potential future study involves considering nonlinear behavior in the specimen.
- In the control plant, a compliance spring is used between the actuator and the specimen. However, using a spring in a large-scale experiment presents implementation challenges. Future work corresponds to the implementation of other elements that serve the same function as the compliance spring, such as a beam, which needs to be designed to remain within the elastic range at all times under the requested displacements.
- The design of the AMBC is performed within a displacement-based scheme due to the compliance spring, despite the general simulation scheme being force-based. Future work involves implementing the AMBC design using force signals, which would change the control plant and the location of the controller.
- The overall study is conducted for a simulation of a shear building where substructuring is performed in the first-floor degree of freedom, corresponding to horizontal displacement. A possible future study involves implementing this study considering the vertical degree of freedom of a column, with the objective of controlling the axial load, which varies during a seismic event.
- Following the previous point, if the implementation in the vertical degree of freedom is achieved, a possible advancement would be the implementation of this scheme in real-time hybrid simulation tests, considering multiple actuators.

References

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.

Bas, E. and Moustafa, M. (2020). Performance and limitations of real-time hybrid simulation with nonlinear computational substructures. *Experimental techniques*, 44:715–734.

Benson Shing, P. (2008). Real-time hybrid testing techniques. *Modern testing techniques for structural systems: Dynamics and control*, pages 259–292.

Blakeborough, A., Williams, M., Darby, A., and Williams, D. (2001). The development of real–time substructure testing. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 359(1786):1869–1891.

Bonelli, A. and Bursi, O. S. (2005). Predictor-corrector procedures for pseudo-dynamic tests. *Engineering computations*.

Bonnet, P., Williams, M., and Blakeborough, A. (2008). Evaluation of numerical time-integration schemes for real-time hybrid testing. *Earthquake engineering & structural dynamics*, 37(13):1467–1490.

Bousias, S. N. (2014). Seismic hybrid simulation of stiff structures: overview and current advances. *Journal of Structures*, 2014.

Carrion, J. E. and Spencer Jr, B. (2007). Model-based strategies for real-time hybrid testing. Technical report, Newmark Structural Engineering Laboratory. University of Illinois at Urbana-Champaign.

Chae, Y., Kazemibidokhti, K., and Ricles, J. M. (2013). 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. and Rabiee, R. (2018). Implementation of effective force testing for nonlinear structures. *Journal of Structural Engineering*, 144(10).

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.

Chang, S.-Y. (2002). Explicit pseudodynamic algorithm with unconditional stability. *Journal of Engineering Mechanics*, 128(9):935–947.

Chang, S.-Y., Yang, Y.-S., and Hsu, C.-W. (2011). A family of explicit algorithms for general pseudodynamic testing. *Earthquake Engineering and Engineering Vibration*, 10:51–64.

Chen, C. (2007). *Development and numerical simulation of hybrid effective force testing method*. Lehigh University.

Chen, C. and Ricles, J. M. (2008). Stability analysis of sdof real-time hybrid testing systems with explicit integration algorithms and actuator delay. *Earthquake Engineering & Structural*

Dynamics, 37(4):597–613.

Chen, C. and Ricles, J. M. (2009). Analysis of actuator delay compensation methods for real-time testing. *Engineering Structures*, 31(11):2643–2655.

Chen, P.-C., Chang, C.-M., Spencer, B. F., and Tsai, K.-C. (2015). Adaptive model-based tracking control for real-time hybrid simulation. *Bulletin of Earthquake Engineering*, 13:1633–1653.

Chen, P.-C. and Tsai, K.-C. (2013). Dual compensation strategy for real-time hybrid testing. *Earthquake Engineering & Structural Dynamics*, 42(1):1–23.

Combescure, D. and Pegon, P. (1997). α-operator splitting time integration technique for pseudodynamic testing error propagation analysis. *Soil Dynamics and Earthquake Engineering*, 16(7-8):427–443.

Dermitzakis, S. N. and Mahin, S. A. (1985). *Development of substructuring techniques for on-line computer controlled seismic performance testing*. PhD thesis, University of California, Berkeley.

Dimig, J., Shield, C., French, C., 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.

Dyke, S., 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.

Dyke, S. J., Spencer Jr, B., Sain, M., and Carlson, J. (1996). Modeling and control of magnetorheological dampers for seismic response reduction. *Smart materials and structures*, 5(5):565.

Elnashai, A. S., Spencer, B. F., Kuchma, D. A., Yang, G., Carrion, J., Gan, Q., and Jig Kim, S. (2006). The multi-axial full-scale sub-structured testing and simulation (must-sim) facility at the university of illinois at urbana-champaign. In *Advances in earthquake engineering for urban risk reduction*, pages 245–260. Springer.

Fares, A. M. (2018). Simplified equations for rigidity and lateral deflection for reinforced concrete cantilever shear walls. *International Journal of Civil and Environmental Engineering*, 12(10):1051–1055.

Fermandois, G., Galmez, C., and Valdebenito, M. (2020). Optimal gain calibration of adaptive model-based compensation for real-time hybrid simulation testing. In *17th World Conference on Earthquake Engineering (17WCEE)(Sendai)*.

Fermandois, G. and Spencer Jr, B. F. (2018). Development and implementation of a multi-axial real-time hybrid simulation framework. Technical report, Newmark Structural Engineering Laboratory. University of Illinois at Urbana-Champaign.

Fermandois, G. A. (2019). Application of model-based compensation methods to real-time hybrid simulation benchmark. *Mechanical Systems and Signal Processing*, 131:394–416.

Gálmez, C. and Fermandois, G. (2022). Robust adaptive model-based compensator for the real-time hybrid simulation benchmark. *Structural Control and Health Monitoring*, 29(7):e2962.

Ghaboussi, J., Yun, G., and Hashash, Y. (2006). A novel predictor-corrector algorithm for substructure pseudo-dynamic testing. *Earthquake engineering & structural dynamics*, 35(4):453–476.

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. In *Proceedings of the Japan society of civil engineers*, pages

1–9. Japan Society of Civil Engineers.

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.

Hughes, T. J., Pister, K. S., and Taylor, R. L. (1979). Implicit-explicit finite elements in nonlinear transient analysis. *Computer Methods in Applied Mechanics and Engineering*, 17:159–182.

Ioannou, P. and Fidan, B. (2006). *Adaptive control tutorial*. SIAM.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pages 1942–1948. IEEE.

McCrum, D. and Williams, M. (2016). An overview of seismic hybrid testing of engineering structures. *Engineering Structures*, 118:240–261.

McKay, M. D., Beckman, R. J., and Conover, W. J. (2000). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61.

Merritt, H. E. (1967). Hydraulic control systems, john wiley & sons. *J. Wiley*.

Najafi, A., Fermandois, G. A., Dyke, S. J., and Spencer Jr, B. F. (2023). Hybrid simulation with multiple actuators: A state-of-the-art review. *Engineering Structures*, 276:115284.

Najafi, A. and Spencer Jr, B. F. (2021). Multi-axial real-time hybrid simulation framework for testing nonlinear structural systems with multiple boundary interfaces. *Newmark Structural Engineering Laboratory Report Series {add number}*.

Nakashima, M. (2020). Hybrid simulation: An early history. *Earthquake Engineering and Structural Dynamics*, 49(10):949–962.

Nakashima, M., Kato, H., and Takaoka, E. (1992). Development of real-time pseudo-dynamic testing. *Earthquake engineering & structural dynamics*, 21(1):79–92.

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.

Newmark, N. M. (1959). A method of computation for structural dynamics. *Journal of the engineering mechanics division*, 85(3):67–94.

Palacio-Betancur, A. and Soto, M. G. (2019). Adaptive tracking control for real-time hybrid simulation of structures subjected to seismic loading. *Mechanical Systems and Signal Processing*, 134:106345.

Phillips, B. M. and Spencer Jr, B. F. (2011). Model-based feedforward-feedback tracking control for real-time hybrid simulation. *Newmark Structural Engineering Laboratory Report Series 028*.

Phillips, B. M. and Spencer Jr, B. F. (2012). Model-based framework for real-time dynamic structural performance evaluation. Technical report, Newmark Structural Engineering Laboratory. University of Illinois at Urbana-Champaign.

Pratt, J., Krupp, B., and Morse, C. (2002). Series elastic actuators for high fidelity force control. *Industrial Robot: An International Journal*, 29(3):234–241.

Qian, Y., Ou, G., Maghareh, A., and Dyke, S. J. (2014). Parametric identification of a servo-hydraulic actuator for real-time hybrid simulation. *Mechanical Systems and Signal Processing*, 48(1-2):260–273.

Reinhorn, A., Sivaselvan, M., Weinreber, S., and Shao, X. (2004a). A novel approach to dynamic force control. In *Third European Conference on Structural Control*.

Reinhorn, A., Sivaselvan, M., Weinreber, S., and Shao, X. (2004b). Real-time dynamic hybrid testing of structural systems. *13th World Conference on Earthquake Engineering*.

Rosenbrock, H. (1963). Some general implicit processes for the numerical solution of differential equations. *The Computer Journal*, 5(4):329–330.

Shao, X. (2007). *Unified control platform for real time dynamic hybrid simulation*. PhD thesis, State University of New York at Buffalo.

Shao, X. and Reinhorn, A. (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., and Sivaselvan, M. (2006). Real-time dynamic hybrid testing using force-based substructuring. In *100th Anniversary Earthquake Conference, San Francisco, CA*.

Shao, X. and Reinhorn, A. M. (2007). Real-time hybrid dynamic simulation with substructure techniques. In *Structural Engineering Research Frontiers*, pages 1–16. ASCE 7 Library.

Shield, C. K., French, C. W., and Timm, J. (2001). Development and implementation of the effective force testing method for seismic simulation of large–scale structures. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 359(1786):1911–1929.

Shing, P. B., Bursi, O. S., and Vannan, M. T. (1994). Pseudodynamic tests of a concentrically braced frame using substructuring techniques. *Journal of Constructional Steel Research*, 29(1-3):121–148.

Silva, C. E., Gomez, D., Maghareh, A., Dyke, S. J., and Spencer Jr, B. F. (2020). Benchmark control problem for real-time hybrid simulation. *Mechanical Systems and Signal Processing*, 135:106381.

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 and Structural Dynamics*, 37(15):1785–1800.

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.

Wallace, M., Sieber, J., Neild, S. A., Wagg, D. J., and Krauskopf, B. (2005a). Stability analysis of real-time dynamic substructuring using delay differential equation models. *Earthquake engineering & structural dynamics*, 34(15):1817–1832.

Wallace, M., Wagg, D., and Neild, S. (2005b). 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.

Zhao, J., French, C., Shield, C., 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.

Appendix

Appendix A. Matlab Code

The source code for the work conducted in this study is hosted in the author's personal repository at the following link: https://github.com/diego-araya-iglesias/vFBRTHS_AMBC.git