University of Alberta

Nonlinear State-Space Control Design for Displacement-Based Real-Time Testing of Structural Systems

by

Seyed Abdol Hadi Moosavi Nanehkaran

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Structural Engineering

Department of Civil and Environmental Engineering

Spring 2011
Edmonton, Alberta

Examining Committee

Dr. Oya Mercan, Department of Civil Engineering and Environmental

Dr. Alan Lynch, Department of Electrical and Computer Engineering

Dr. Samer Adeeb, Department of Civil Engineering and Environmental

Abstract

This study presents the nonlinear design of a state space controller to control hydraulic actuators under displacement control, specifically for real-time pseudo-dynamic testing applications. The proposed control design process uses the nonlinear state space model of the dynamics of the system to be controlled; and utilizes state feedback linearization through a transformation of the state variables. Comparisons of numerical simulation results for linear state-space and nonlinear state-space controllers are given. Also robustness of the control design with respect to identified parameters is investigated. It is shown that a controller with improved performance can be designed using nonlinear state space control design techniques, provided that a representative model of the system is available.

Preface

There are several dynamic testing methods that have been introduced and used to examine and/or verify the dynamic performance of conventional and new structural systems. Some of these methods themselves are still under research in an attempt to make them more efficient and accurate. Among these, pseudo-dynamic (PSD) testing method and as its extension hybrid PSD testing method offer economical and practical ways to assess the dynamic behaviour of structural systems. If executed at fast rates (ideally in real-time) these testing methods can handle load-rate dependent structures (such as the ones that have dampers installed for seismic hazard mitigation purposes) appropriately.

In a PSD test the equation of motion of the test structure is solved by a direct step-by step integration algorithm where the inertial and damping force characteristics are kept analytical. These displacements are imposed on the test structure by hydraulic actuators; the resulting restoring forces from the deformed structure are measured and fed back to the integration algorithm for the computation of next step displacements. The method is called hybrid PSD when the whole structure is split into experimental and analytical substructures to avoid fabricating a big structure in the laboratory. During a hybrid PSD test the command displacements generated by the

integration algorithm are imposed on the experimental and analytical substructures and the results from both are combined and fed back.

One of the main challenges in using PSD method is the sensitivity of the results to the experimental errors. This is because of the closed loop nature of the method; i.e. in each time step of the test procedure, the integration algorithm uses the measured information (which may be contaminated by errors) from the previous step to generate the new displacements to be applied. This fact implies that the errors in each step are affected from the errors of the previous steps which may be cumulatively added together until the end of the test. As a result PSD method suffers from propagation of the error which at best causes accuracy problems or at worst renders the whole test unstable (i.e., the test needs to be aborted as the displacements grow unboundedly).

From a stability (and also accuracy) point of view, the delay in the measured signals (as opposed to lead, or amplitude errors) has been found critical in real-time testing (Mercan and Ricles 2008). The delay mainly arises due to the time it takes the actuators to reach the command displacements issued by the integration algorithm.

In general a PID controller (or a modified version) is employed to control the actuator displacements which may not provide acceptably accurate tracking especially when large displacements need to be imposed at fast rates under considerable load. This may be due to the fact that under these

types of conditions, the servo-hydraulic system nonlinearities are invoked or the test structure presents highly nonlinear behaviour. Hence, when nonlinearities in the servo-hydraulic system are invoked, there might be a need to use nonlinear state-space controller to account for them.

Due to its matrix form, a state-space control is preferable to a PID controller in cases when physically coupled multiple degrees of freedom need to be controlled. The main effort of the research presented is to develop a control algorithm based on advanced control theories (e.g. nonlinear state-space) and assess its performance in comparison with a PID controller. Simulations are done using a model of the servo-hydraulic system that accounts for nonlinearities. This study is intended to lead the application of nonlinear state-space control strategy in PSD testing method.

Acknowledgement

Foremost, I would like to express my sincere gratitude to my advisor Dr. Oya Mercan for the continuous support of my M.Sc. study and research, for her persistence, interest, and help. Her guidance helped me in all the time of research and writing of this thesis. Besides my advisor, I would like to thank the rest of my thesis committee: Dr. Alan Lynch and Dr. Samer Adeeb for their insightful comments, and hard questions.

Last but not the least I would like to thank my family for their life-time support.

Contents

List of Tables

List of Figures

1 Introduction

1.1 General

This chapter contains background information about seismic testing methods. The motivation and objective of the research and organization of the dissertation are also described.

1.2 Seismic Testing Methods of Structures

One of the main goals in seismic performance testing is to impose loading conditions on a test specimen that are representative of those that might happen during a real earthquake. To achieve this goal, various forms of earthquake and structural dynamic testing methods have been the subject of research.

Four experimental laboratory techniques are typically used in seismic performance testing of structures: quasi-static testing, shaking table testing, effective force testing (EFT) and pseudo-dynamic (PSD) method

In a quasi-static test, a predefined cyclic displacement history is applied to the structure or structural component under study and the behaviour is observed and analyzed. The predefined displacement history, if not selected

from some typically used displacement histories, is based on a response computed from a dynamic time history analysis prior to testing. This method is commonly used and economical. However, it is limited in terms of delivering the true earthquake response. This is because the model with which the pre-test dynamic analysis is performed may not accurately predict the behaviour and in turn, the resulting displacement time history may not correspond to the real earthquake response.

Placing a structure on a shaking table and exerting a properly scaled ground motion may be the most realistic method. In spite of this, due to payload restrictions of shake tables, shaking table tests are implemented on small-scale test structures. This implies that the ground acceleration needs to be scaled (compressed) accordingly. As a result the available time for observing the behaviour will be very little during the test. Generally speaking, despite the fact that the shaking table may be representative of the actual seismic behaviour, the combined effects of the need to construct the complete structure, small scale test specimens, short observation time and finally the cost limit the use of shaking table.

Effective force testing (EFT) is a real-time testing method. This method uses a force control approach and can be employed in real-time earthquake simulation of large scale structures. Knowing the structural mass and ground acceleration history, the complete force history that should be applied to the structure is calculated beforehand. Despite the conceptual simplicity, the

implementation of this method has been observed to be challenging due to actuator-structure interaction (Dimig et al. 1999). To overcome this problem, Zhao (2003) proposed a nonlinear velocity compensation scheme and verified it through simulations and experimental studies under limited conditions.

In the late 1970s and early 1980s PSD method was initiated as an experimental technique in which the displacement response of structure to a given ground acceleration is numerically calculated and quasi-statically imposed on the structure (Takanashi et al. 1975; Okada et al. 1980; Mahin and Williams 1981; Shing and and Mahin 1983; Mahin and Shing 1985). This computed response is based on analytically predefined inertia and viscous damping as well as the experimentally measured structural resisting force. Details for the procedure are given below.

1.3 PSD and Hybrid PSD testing method

In a PSD test, the test structure is first idealized as a discrete parameter system. Thereby for a structure subjected to ground acceleration (Figure 1-1) the governing equations of motion can be expressed as a system of second order ordinary differential equations with respect to time.

$$\boldsymbol{M}\,\boldsymbol{a}(t) + \boldsymbol{C}\,\boldsymbol{v}(t) + \boldsymbol{r}(t) = \boldsymbol{P}(\boldsymbol{M}, a_g, t) \qquad \text{Eq. 1-1}$$

$$\boldsymbol{M}\,\boldsymbol{a}(t) + \boldsymbol{C}\,\boldsymbol{v}(t) + \boldsymbol{r}(t) = \boldsymbol{P}(\boldsymbol{M}, a_g, t)$$

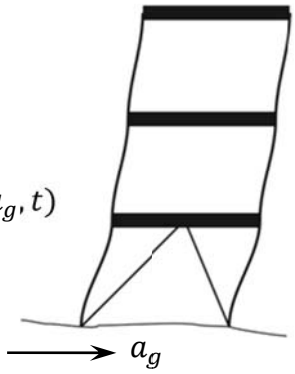


Figure 1-1 Structure subject to ground acceleration in a real earthquake

In the above equation $\boldsymbol{M}$ and $\boldsymbol{C}$ are the mass and damping matrices, respectively; $\boldsymbol{a}(t)$ and $\boldsymbol{v}(t)$ are the acceleration and velocity vectors respectively; $\boldsymbol{r}(t)$ is the resisting force vector in the structure and $\boldsymbol{P}(t)$ is the external (effective) load vector that is obtained using the mass matrix and the given ground acceleration. All of the matrices and vectors defined above are associated with the degrees of freedom (DOFs) of the idealized structure. For a step-by-step simulation or testing (e.g. PSD) a discretized form of Eq. 1-1 is used

$$\boldsymbol{M}\,\boldsymbol{a_i} + \boldsymbol{C}\,\boldsymbol{v_i} + \boldsymbol{r_i} = \boldsymbol{P_i} \qquad \text{Eq. 1-2}$$

At each step of a conventional (slow and in extended time scale) PSD test (see Figure 1-2) the above equation of motion is solved using a direct step-by-step integration algorithm to generate the command displacements.

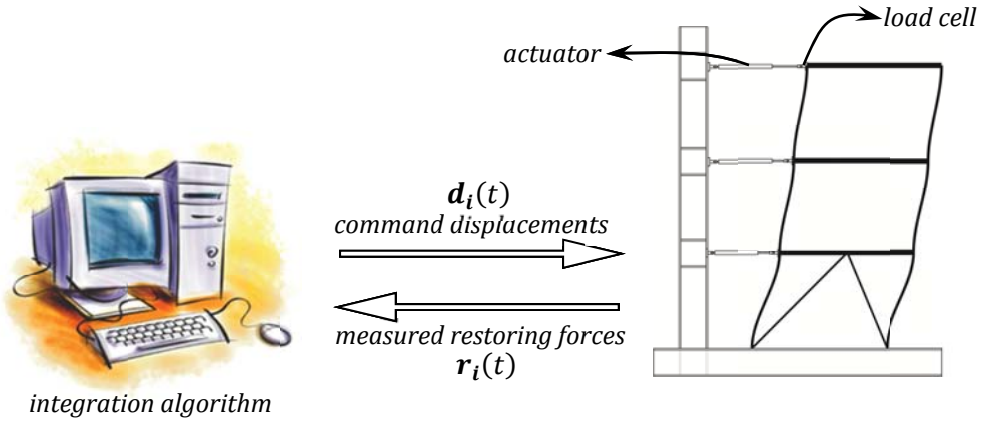


Figure 1-2 PSD testing method

These target displacements are imposed by hydraulic actuators under a feedback control quasi-statically. Then the resisting forces from the structure are measured and fed back to the integration algorithm to be used to generate target displacements for the next step. It should be noted that depending on the experiment rate and dynamic characteristics of the test structure, the measured restoring force may include stiffness (strain-dependent), damping or inertial forces. Capturing inertial force contribution in the measured restoring force is not desirable, that is why special attention is paid to minimize the moving mass in the experimental test set-up.

The main feature of PSD testing method is that the resisting forces are measured experimentally from the deformed test structure and are used by the integration algorithm in command generation. This assures that any nonlinear characteristics associated with the resisting forces are properly accounted for.

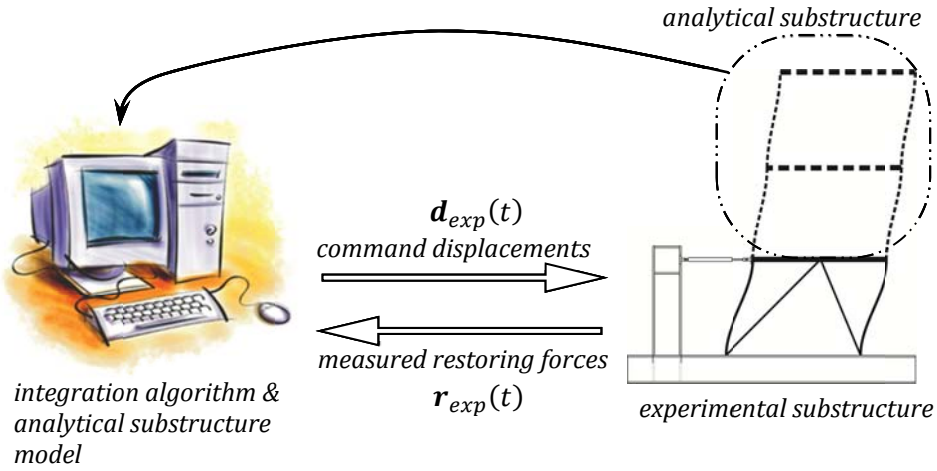


Figure 1-3 Hybrid PSD testing method

Hybrid PSD test method (Figure 1-3) is an economical and practical extension of PSD testing. In this method only a portion of the test structure, for which a reliable analytical model may not exist is tested physically (which is referred to as the experimental substructure), and the remaining part is modeled in a computer (and referred to as the analytical substructure). In a hybrid PSD test the two substructures are coupled together; the command displacements generated by the integration algorithm are imposed to both substructures simultaneously, and the restoring forces coming from both are combined and fed back to the integration algorithm.

In a conventional PSD test, since the target displacements are imposed quasi-statically (i.e. in an extended time scale), the behaviour of structures with material properties or components that are load-rate dependent may not be accurately captured. Examples of these load-rate dependent structural

components and devices are metallic, friction, visco-elastic, viscous fluid, tuned mass, tuned liquid, elastomeric dampers and lead rubber bearings which are introduced into structures for vibration mitigation purposes. For assessing the capabilities of structures equipped with these devices, it is necessary to perform the PSD test dynamically at rates approaching to real-time.

In PSD testing method the results are sensitive to experimental errors. This is because the method has a closed loop algorithm; i.e. in each time step of the test procedure, the integration algorithm uses the measured information from the previous step (which may be contaminated by errors) to generate the new displacements to be applied. This means that the errors in each step are affected from the errors of the previous steps which are cumulatively added together until the end of the test. Hence PSD method is prone to propagation of the error, which at best causes accuracy problems or at worst renders the whole test unstable (i.e., the test needs to be aborted as the displacements grow unboundedly). Mercan (2007) showed that actuator delay in following the command displacements in experimental setup may impair the dynamic stability of the test setup in a real-time PSD test. This is also true for real-time hybrid PSD method.

One of the main sources of time delay is the time it takes the actuator to reach the command displacement calculated by the integration algorithm which cannot be done instantaneously. This is because it takes time to

convert energies (electrical to mechanical) in an experimental setup. Therefore it is critical to use the available equipment efficiently so that the corresponding time delay is minimum. This is done by using the control theory to design servo-hydraulic controllers. The commonly used controller in PSD tests is the well-known PID (proportional-Integral-Derivative) controller. PID controllers are popular because they can be adjusted (tuned) to control complex systems without the need for complete knowledge over the dominating dynamics.

Conducting a fast (ideally real-time) PSD test requires accurate actuator control by means of sophisticated servo-hydraulic strategies and reliable computation scheme through efficient integration algorithms.

The dynamics of an electrohydraulic servo system is highly nonlinear and involves sign and square root functions. However in most industrial contexts linear control theory is used. Although linearization about an operating point decreases design effort, it degrades the performance at regions off the operating point. There have been several works on controlling electrohydraulic servo systems using advanced control methods.

Lim (2002) applied linearization and pole placement. Yanada and Furuta (2007) combined linear theory with an adaptive approach. Kwon et al. (2007) applied full-state feedback linearization. Seo et al. (2007) used feedback linearization to design controllers for displacement, velocity and differential pressure control of a rotational hydraulic drive and Ayalew and

Jablokow (2007) used partial-input feedback linearization for the control of electrohydraulic servo systems. Mintsa et al. (2009) used feedback linearization to design an adaptive control for electrohydraulic position servo system with the objective of enhancing robustness with respect to variations of supply pressure.

1.4 Research goals and thesis organisation

In order to improve the tracking capability, and in turn the accuracy of the overall real-time PSD test results, this thesis investigates the design and implementation of advanced control algorithms.

As stated before, currently the majority of the controllers used in PSD testing are PID based controllers. They are tuned according to a linearized model of the system dynamics (for single input single output systems), or by ad-hoc tuning; where the accurate window of operation is limited to the linear range of the system dynamics. In the case of a multi-degree of freedom test structure, state-space design offers considerable reduction in control design effort (Mercan et al. 2006). The above mentioned methods, as they are designed based on linear models of the system, may not provide acceptably accurate tracking when the servo-hydraulic system nonlinearities are invoked or the test structure presents highly nonlinear behaviour.

In this study the design of a nonlinear state-space controller for real time pseudo-dynamic testing of structural systems is presented based on advanced control theories using a nonlinear model of the system.

After the introduction in this chapter, Chapter 2 considers the governing dynamics and equations of a servo-hydraulic system. Chapter 3 begins with basics of control theory and afterwards linear and nonlinear control design methods are introduced and discussed. Chapter 4 applies the control design methods introduced in Chapter 3 on the dynamics discussed in Chapter 2. Finally, Chapter 5 illustrates the implementation of the control methods by computer simulation and compares the behaviour of different controllers.

Chapter 6 discusses the design of a single degree of freedom test setup to investigate different aspects of a PSD test method (e.g. applying different controllers). A conference paper titled “Modification of Integration Algorithm to Account for Load Discontinuity in Pseudo-Dynamic Testing” that was done along with this research is presented as an appendix.

2 Modelling of the Servo-System and Identification of system parameters (system ID)

To work out a new control approach with an improved performance for a servo-hydraulic system, the behaviour of the system has to be well understood through simulation of representative models. These models are based on the important dynamics that govern the behaviour and include physical parameters some of which may already be known (e.g. piston area) while others need to be identified through system identification. While deciding for the level of accuracy (complexity) of the model, one needs also to consider if the parameters that appear in the model are easily obtainable through system identification or not. In the next section different parts of a servo hydraulic system and the governing dynamics are elaborated.

2.1 Components of a Servo-Hydraulic System

A position controlled servo-hydraulic system used in a PSD test typically consists of a hydraulic power supply, a flow control servo-valve, a linear actuator, a displacement transducer, and an electronic servo-controller. Figure 2-1 shows a block diagram that represents how the above mentioned parts are interconnected. The controller compares the command displacement with the measured displacement coming from the displacement transducer (e.g. an LVDT) to determine the position error and

then sends out a command signal to drive the flow control servo-valve. In fact the command signal introduces a spool displacement in the servo-valve to adjust the flow of pressurized oil from the hydraulic power supply to the linear actuator chambers in order to move the actuator piston to the desired position.

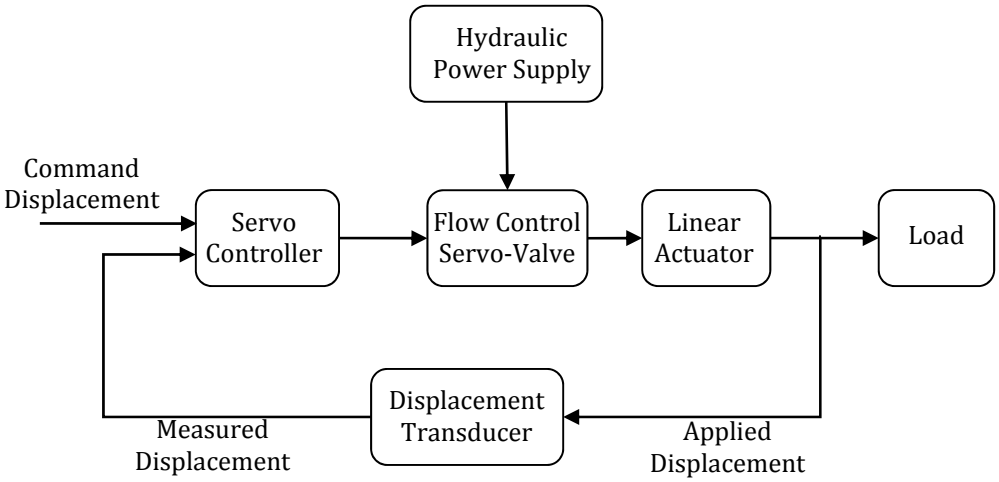


Figure 2-1 Block diagram of inner loop in PSD test method

The following gives a concise explanation of the parts shown in Figure 2-1.

- Hydraulic Power supply

A hydraulic power supply provides the pressurized fluid (oil) for the hydraulic system. The level of oil pressure in the power supply is selected considering several factors. Low pressure systems have less leakage but physically larger components are needed to provide a specified force. On the other hand, high pressure systems experience more leakage but have better

dynamic performance and have smaller (lighter) components. In many high performance systems 3000 psi (210 bar) is selected for the system pressure.

2.1.1 Flow Control Servo-Valve

An electro hydraulic flow control servo valve is a servo valve which is designed to produce hydraulic flow output proportional to electrical current input. The mechanism and dynamics are discussed later.

2.1.2 Linear Hydraulic Actuator

A hydraulic actuator converts hydraulic energy to mechanical force or motion. They are implemented where large actuation forces and fast motion are required. Governing dynamics are given later.

2.1.3 Displacement Transducer

They generally come built-in with actuators and are often attached directly to the piston rod. There are various types of feedback transducers in use including inductive linear variable differential transformer (LVDT). It is a common practice to include external displacement transducers in the test setup to check the measurements of the internal transducers and to exclude the actuator support motion.

2.1.4 Servo Controller

A controller continuously compares the command displacement against the actuator position that is measured by a displacement transducer. The result of this comparison is displacement error which is then manipulated by

a control law in order to generate and send a command signal to the servo valve.

2.2 Dynamics of a Servo-Hydraulic System

2.2.1 Servo-valve dynamics

Servo-valves are used in servo-hydraulic systems to convert the electrical command signal, coming from the controller, to a spool displacement. This displacement along with the differential pressure between the servo-valve ports results in oil flow through valve control ports into and out of the actuator chamber enabling the motion of the hydraulic piston.

Among several types of servo-valves are the flow control servo-valves in which the control flow at constant load pressure is proportional to the electrical input current (Thayer 1958, revised in 1965).

Figure 2-2 shows a two stage flow control servo valve. It is called two-stage as it has two portions containing a hydraulic amplifier. A hydraulic amplifier is a fluid valving device which acts as a power amplifier, such as a sliding spool and a nozzle flapper and will be elaborated in this chapter.

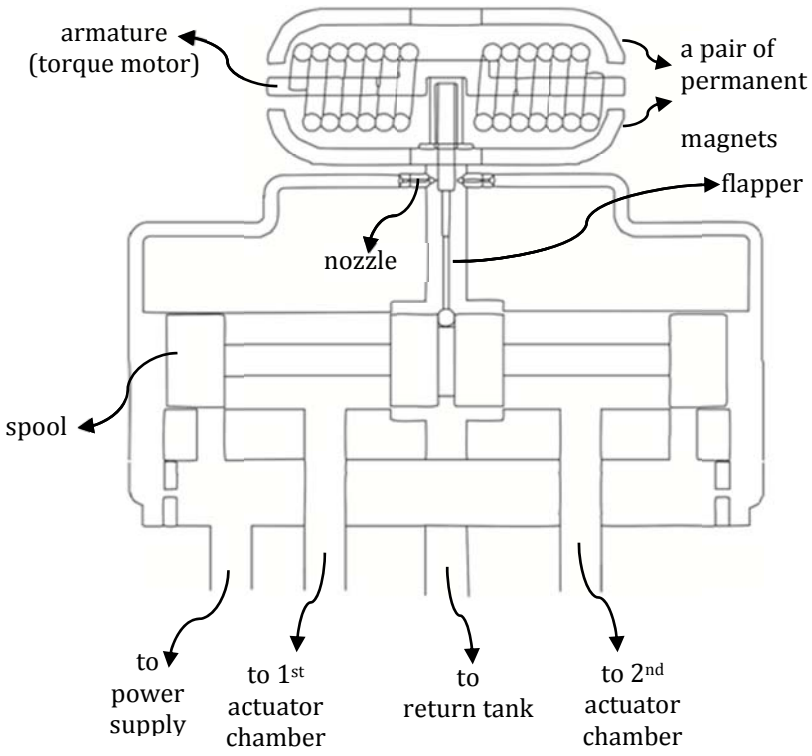


Figure 2-2 Cross section of a two-stage flow control valve

In a two-stage servo-valve the electrical command signal applied to the torque motor coils creates a magnetic force. This magnetic force causes a deflection of armature-flapper assembly where the resulting deflection restricts fluid flow through one of the two nozzles and redirects the flow to displace the spool.

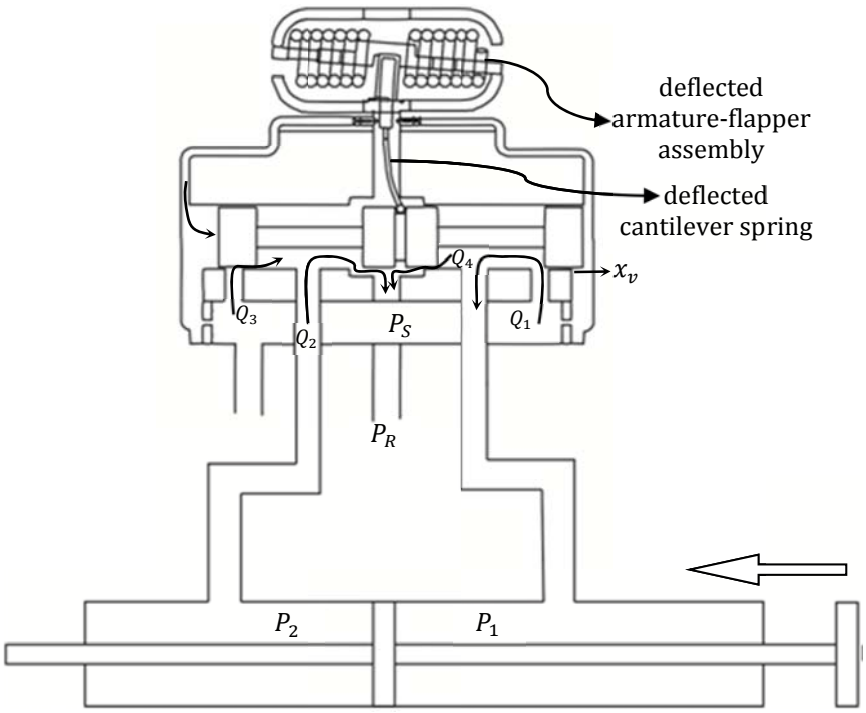


Figure 2-3 Cross section of a two-stage flow control valve at operation

Spool displacement opens the supply pressure port (P_S) to one of the control ports and at the same time opens the return tank port (P_R) to the other control port. The displaced spool applies a force to the cantilever spring shown in Figure 2-3 that creates a restoring torque on the armature-flapper assembly. Spool movement continues until the restoring force becomes equal to the torque from the magnetic force and that is when the assembly moves back to the neutral position and the spool is held open in a state of equilibrium until the electrical command signal changes to a new level (Moog 2010).

In summary, as pointed out before, the spool position is proportional to the input current. Also with constant pressure drop across the valve

(constant load pressure) the load flow is proportional to the spool position. The governing dynamics of the servo valve will be discussed in two parts; valve spool dynamics which includes the relationship between the input current and the spool displacement and valve flow dynamics which explains how the spool displacement relates to the flow from the valve to the actuator chambers.

2.2.1.1 Valve Spool Dynamics

Servo-valves are complicated devices. Experience has shown that their nonlinear and non-ideal characteristics make it hard to theoretically analyze servo-valve dynamics in systems design. Instead, it is more convenient but also acceptably accurate to approximate servo-valve dynamics with suitable empirical transfer functions by using measured servo-valve response (Thayer 1958, revised in 1965). Depending on the frequency range of interest the servo-valve dynamics can be represented by a first order transfer function.

$$\frac{X_v(s)}{I(s)} = \frac{k_v}{1 + \tau\, s} \qquad \text{Eq. 2-1}$$

Where x_v, i, k_v and τ are servo-valve spool opening (see Figure 2-3), differential current input to servo-valve, servo-valve static flow gain at zero load pressure drop and τ apparent servo-valve time constant. It should be noted that on the left hand side of Eq. 2-1 the Laplace transforms of spool opening and input current are used.

2.2.1.2 Valve Flow Dynamics

As can be seen in the Figure 2-3, four flows can be recognized between the servo-valve and the actuator ports.

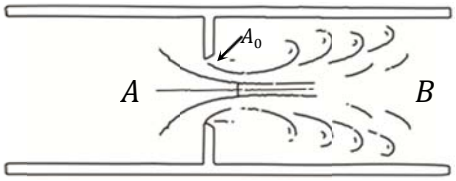


Figure 2-4 Turbulent flow through an orifice

These flows are classified as turbulent flows. Eq. 2-2 represents the turbulent flow through an orifice (see Figure 2-4)

$$Q = C_d\, A_0 \sqrt{\frac{2}{\rho}(P_A - P_B)} \qquad \text{Eq. 2-2}$$

C_d, A_0, ρ and $P_A - P_B$ are discharge coefficient, orifice area, density of the flowing liquid and differential pressure between point A and B respectively.

Rewriting the above turbulent orifice flow for Figure 2-3 assuming for now that C_d is the same for all orifices gives

$$Q_1 = C_d\, A_1 \sqrt{\frac{2}{\rho}(P_S - P_1)} \qquad \text{Eq. 2-3}$$

$$Q_2 = C_d\, A_2 \sqrt{\frac{2}{\rho}\, P_2} \qquad \text{Eq. 2-4}$$

$$Q_3 = C_d\, A_3 \sqrt{\frac{2}{\rho}(P_S - P_2)} \qquad \text{Eq. 2-5}$$

$$Q_4 = C_d\, A_4 \sqrt{\frac{2}{\rho}\, P_1} \qquad \text{Eq. 2-6}$$

In the above equations, $A_1(A_2,\ A_3\ or\ A_4)$ is the orifice area associated with flow $Q_1(Q_2, Q_3\ or\ Q_4)$, and P_S is the supply pressure. Also P_1 and P_2 are the pressures at each one of the actuator chambers.

The return pressure (P_R) is assumed to be zero as it is usually much smaller than the other pressures involved. If the return pressure is not negligible, the supply pressure in the above expressions can be interpreted as supply pressure minus return pressure.

The areas of the orifices are functions of the spool opening x_v. Because the valve orifices are matched,

$$A_1(x_v) = A_2(x_v) \text{ and } A_3(x_v) = A_4(x_v) \qquad \text{Eq. 2-7}$$

And because they are symmetrical,

$$A_1(x_v) = A_3(-x_v) \text{ and } A_2(x_v) = A_4(-x_v) \qquad \text{Eq. 2-8}$$

It can be shown that when the orifice areas are matched and symmetrical

$$Q_1 = Q_2 \text{ and } Q_3 = Q_4 \text{ (Merritt 1967)} \qquad \text{Eq. 2-9}$$

Combining Eq. 2-3, Eq. 2-4, Eq. 2-7 and Eq. 2-9 pressures at two chambers can be related to supply pressure by,

$$P_S = P_1 + P_2 \qquad \text{Eq. 2-10}$$

By definition the load pressure is the pressure difference between the two actuator chambers and is expressed as

$$P_L = P_1 - P_2 \qquad \text{Eq. 2-11}$$

Using Eq. 2-10 and Eq. 2-11, P_1 and P_2 may be written as

$$P_1 = \frac{(P_S + P_L)}{2} \qquad \text{Eq. 2-12}$$

$$P_2 = \frac{(P_S - P_L)}{2} \qquad \text{Eq. 2-13}$$

Load flow, which is the flow from the valve to one of the actuator chambers, can be expressed as

$$Q_L = Q_1 - Q_4 = Q_2 - Q_3 \qquad \text{Eq. 2-14}$$

Finally using the derived equations for a matched and symmetrical valve, the load flow may be written as

$$Q_L = C_d\, A_1 \sqrt{\frac{1}{\rho}(P_S - P_L)} - C_d\, A_4 \sqrt{\frac{1}{\rho}(P_S + P_L)} \qquad \text{Eq. 2-15}$$

For an ideal critical center valve with matched and symmetrical orifices the leakage flows (Q_4 and Q_3 when the spool displacement is positive) are zero because the valve geometry is assumed ideal. On the other hand when the spool displacement is negative Q_1 and Q_2 will be the leakage flows. Considering the fact that the orifice areas are linear functions of the spool opening as the product of the spool opening and full periphery of the spool (πd), load flow can be expressed as (Merritt 1967)

$$Q_L = \eta\, x_v \sqrt{P_s - sign(x_v) P_L} \qquad \text{Eq. 2-16}$$

where η is defined as

$$\eta = C_d\, \pi\, d \sqrt{\frac{1}{\rho}} \qquad \text{Eq. 2-17}$$

Eq 2-16 is not differentiable due to sign function. Sign function may be approximated by some smooth function and then used in the control design (Chapter 4)

2.2.2 Actuator Chamber Pressure Dynamics

The oil flow between valve and actuator chambers causes the actuators piston to move. Using conservation of mass principle on both sides of the actuator chambers, the actuator pressure dynamics can be expressed as

$$Q_L = A_p\, \dot{x}_p + C_{tp}\, P_L + \frac{V_t}{4\beta_e} \dot{P}_L \qquad \text{Eq. 2-18}$$

In this equation A_p is the actuator piston cross-sectional area; $\dot{x}_p$ is piston velocity or the time derivative of piston displacement (x_p); C_{tp} is the leakage coefficient of piston; P_L and $\dot{P}_L$ are load pressure as defined in Eq. 2-11 and rate of load pressure respectively; V_t is actuator chamber volume and β_e is oil modulus.

2.2.3 Piston Dynamics

Writing the force equation of motion and considering a static friction and the external force from a test specimen gives

$$m_p\,\ddot{x}_p + b_p\,\dot{x}_p + T_f + F_{ext} = A_p\,P_L \qquad \text{Eq. 2-19}$$

m_p, b_p and T_f are piston mass, viscous damping coefficient of actuator piston and static friction respectively and F_{ext} represents the effect of stiffness, damping and inertial forces from a test specimen.

2.2.4 Linear Approximation of the Dynamics

For controller design purposes a linearized, simplified model is necessary especially when the controller design will be based on linear control theory.

Note that the valve spool dynamics (Eq. 2-1) and actuator chamber pressure dynamics (Eq. 2-18) are already linear. Using Taylor series expansion, Eq. 2-15 (that represents flow-pressure relationship) can be linearized about an operating point ($Q_L = P_L = x_v = 0$) while assuming zero leakage flow and ideal geometry (Merritt 1967) to be

$$Q_L = K_q\, x_v - K_c\, P_L \qquad \text{Eq. 2-20}$$

K_q and K_c are called the flow gain coefficient and the flow pressure coefficient respectively. And for the piston dynamics, in case the friction is negligible Eq. 2-19 can be rewritten as

$$m_p\, \ddot{x}_p + b_p\ \dot{x}_p + F_{ext} = A_p\, P_L \qquad \text{Eq. 2-21}$$

Table 2-1 gives a summary of both nonlinear and linear equations discussed so far.

Table 2-1 Nonlinear dynamics of a servo-hydraulic system and its linear approximation

Dynamics	Nonlinear	Linear
Valve spool	$\frac{X_v(s)}{I(s)} = \frac{k_v}{1+\tau\, s}$ (Eq. 2-1)	$\frac{X_v(s)}{I(s)} = \frac{k_v}{1+\tau\, s}$ (Eq. 2-1)
Valve flow	$Q_L = \eta\, x_v\sqrt{P_s - sign(x_v)P_L}$ (Eq. 2-16)	$Q_L = K_q\, x_v - K_c\, P_L$ (Eq. 2-20)
Actuator chamber pressure	$Q_L = A_p\, \dot{x}_p + C_{tp}\, P_L + \frac{V_t}{4\beta_e}\dot{P}_L$ (Eq. 2-18)	$Q_L = A_p\, \dot{x}_p + C_{tp}\, P_L + \frac{V_t}{4\beta_e}\dot{P}_L$ (Eq. 2-18)
Actuator motion	$m_p\, \ddot{x}_p + b_p\ \dot{x}_p + T_f + F_{ext} = A_p\, P_L$ (Eq. 2-19)	$m_p\, \ddot{x}_p + b_p\ \dot{x}_p + F_{ext} = A_p\, P_L$ (Eq. 2-21)

2.3 System Identification

It is a good idea to identify the system starting from the simplified (linearized) model and finalize it by adding relevant nonlinearities and additional dynamics as required by the measured response.

Grey-box modeling option of the system ID toolbox of MATLAB is ideal when the model to be identified is known (e.g. derived from the first principles) and the numerical values of the parameters that appear in this model need to be estimated from measured data.

In identifying the servo-hydraulic system mentioned above the transfer function relating the input current and the spool opening is known to be

$$\frac{X_v(s)}{I(s)} = \frac{k_v}{1 + \tau\, s} \qquad \text{Eq. 2-22}$$

For a free standing actuator that have F_{ext} equal to zero, a transfer function can be established using Eq. 2-18, Eq. 2-20, and Eq. 2-21 which in turn when is combined with Eq. 2-22 will give a direct transfer function from input current to actuator piston displacement.

$$\frac{X_p(s)}{I(s)} = \frac{4A_p\, \beta_e\, K_q\, k_v\, \omega_v}{s\left[(m_p\, V_t)s^2 + (4\beta_e\, k_{ce}\, m_p + b_p\, V_t)s + 4\beta_e(A_p{}^2 + b_p k_{ce})\right](s + \omega_v)} \qquad \text{Eq. 2-23}$$

where $k_{ce} = C_{tp} + K_c$ and $\omega_v = 1 \,/\, \tau$.

MATLAB identification toolbox has a general built-in system model that can be adjusted to have four poles and no zeroes as

$$\frac{k\, e^{-T_d\, s}}{s[1 + 2\zeta T_w\, s + (T_w\, s)^2](1 + T_{p_3} s)} \qquad \text{Eq. 2-24}$$

The term $e^{-T_d\, s}$ is considered to take care of any time delay that may exist in the system.

The identification procedure basically starts with exciting the system with some predefined inputs and logging the response. There is no restriction for selecting the inputs but usually step and sinusoid inputs with different frequencies are used. Then the logged data is analyzed by MATLAB identification toolbox and values for transfer function parameters are found.

Comparing Eq. 2-23 and Eq. 2-24 and considering the fact that $\frac{b_p\, k_{ce}}{{A_p}^2}$ has a very small value and can be neglected (Zhang et al. 2005), it can be shown

$$T_w = \sqrt{\frac{m_p\, V_t}{4\beta_e\, {A_p}^2}} \qquad \text{Eq. 2-25}$$

$$\zeta = \frac{b_p}{4\, A_p}\sqrt{\frac{V_t}{m_p\, \beta_e}} + \frac{k_{ce}}{A_p}\sqrt{\frac{m_p\, \beta_e}{V_t}} \qquad \text{Eq. 2-26}$$

$$T_{p_3} = \frac{1}{\omega_v} \qquad \text{Eq. 2-27}$$

$$k = \frac{K_q\, k_v}{A_p} \qquad \text{Eq. 2-28}$$

3 Control Theory

General

In a servo-hydraulic system the controller calculates the displacement error and uses it as an input to a control law. The output of the control law is the command signal to the servo valve. In this chapter some principles of feedback control theory is given. Then classic control design, linear state-space control design and nonlinear state-space control design are introduced.

3.1 Feedback Control of Dynamic Systems

In the context of this study a controller is designed to give the following characteristics to the system; (1) the ability to follow command displacements (tracking); (2) the ability to maintain the system stability; (3) the ability to reduce the sensitivity of the system to external disturbances (disturbance rejection). Some criteria need to be defined in order to have a basis of comparison between the performances of different controllers. For this purpose specific input signals like step functions and sinusoidal functions are used. Figure 3-1 shows a typical response of a system to a unit

step input. A physical system stores energy therefore when it is subjected to an input it cannot instantly follow it. Instead a steady state condition is reached after exhibiting a transient response.

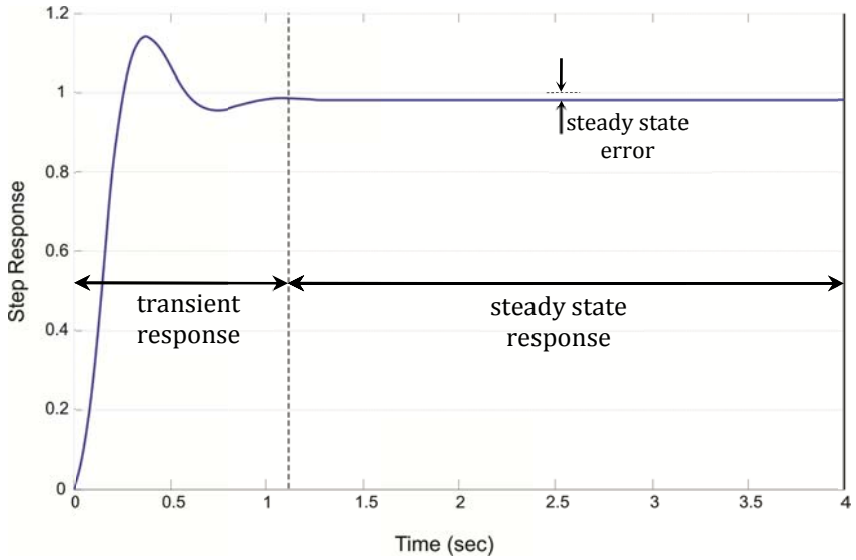


Figure 3-1 an example for a unit step response of a system

For the case of a PSD test, other than stability requirement, which is an essential factor for the system to be operational, transient response characteristics and steady-state error are also considered as controller performance criteria. Mercan (2007) showed that phase error (i.e. the time delay taking the system to apply the command displacement and to measure the resisting force) has much more of a detrimental effect compared to an amplitude error. Phase error introduces an equivalent negative damping that causes inaccuracy and if it goes beyond a critical value, dynamic instability will occur.

Figure 3-2 gives a block diagram of the system including the possible disturbances and noises. It should be noticed that external disturbances may also be present in individual components (e.g., the servo valve) but in order to be concise in Figure 3-2, only the resultant of the disturbances is shown.

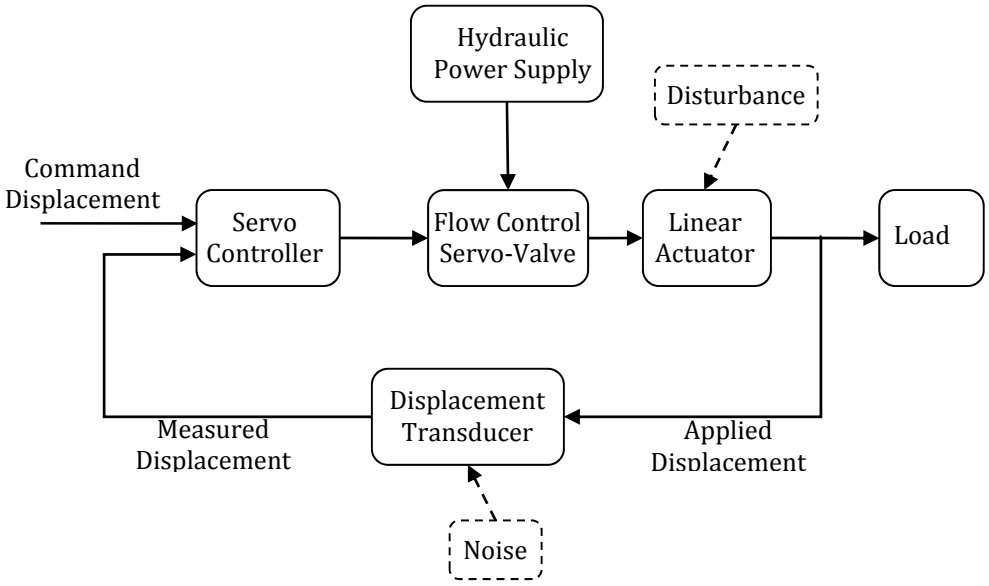


Figure 3-2 Block diagram of a servo hydraulic model including disturbance and noise

The dynamics of a system can be defined by a set (system) of differential equations which are obtained from principles of physics. For a quick approximate analysis and for controller design purposes using linear control theory, linear approximation of the differential equations (whenever they entail nonlinear terms) is used. On the other hand, to represent the system dynamics more realistically computer simulations including system nonlinearities can be performed. Moreover, in the event that the linear controllers that are based on linear models are not efficient, nonlinear design of feedback control may be used.

3.2 Linear Control Design (Basics)

One of the attributes of a linear time-invariant system is that it obeys the principle of superposition. This principle states that if the system has an input that can be expressed as a sum of signals then the response of the system can be expressed as the sum of the individual responses to each signal. In control engineering the dynamics of the system are typically studied using root locus (in s-plane), frequency response or state-space methods. The first two methods are based on Laplace transform and are mainly used in classical control analysis or design. State-space based methods are used in modern control design.

3.2.1 Laplace Transform and Transfer Function

The Laplace transform is the mathematical tool that transforms differential equations into an algebraic form which are easier to manipulate. Compared to the Fourier transform which is informative about the steady state response, the Laplace transform yields complete response characteristics (both transient and steady state response) (Franklin et al. 2010).

The unilateral (one sided) Laplace transform for a time domain function like $f(t)$ is

$$\mathcal{L}\big(f(t)\big) = F(s) = \int_0^{\infty} f(t)e^{-s\,t}\,dt \qquad \text{Eq. 3-1}$$

Where s is a complex variable of the form $s = \sigma + i\,\omega$. σ is the real part and ω is the imaginary part of the complex variable. For elementary functions in time domain, their Laplace transforms are tabulated as provided in Appendix A. therefore there is no need to perform the integration in Eq. 3-1.

The interpretation of the dynamic behaviour in s-domain (upon performing Laplace transformation) is provided using the following exemplar where a mass-damper-spring system is considered (see Figure 5-2).

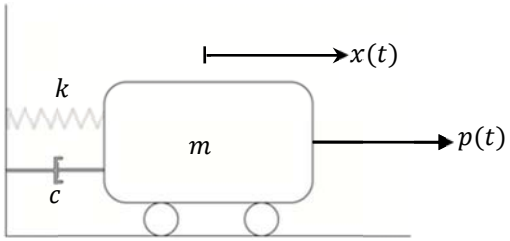


Figure 3-3 a mass-damper-spring system

The equation of motion for the above oscillator in time domain is

$$m\,\ddot{x}(t) + c\,\dot{x}(t) + k\,x(t) = p(t) \qquad \text{Eq. 3-2}$$

Assuming zero initial conditions ($x(0) = 0$, $\dot{x}(0) = 0$) and taking the Laplace transform of both sides of Eq. 3-2

$$\mathcal{L}\big(m\,\ddot{x}(t) + c\,\dot{x}(t) + k\,x(t)\big) = \mathcal{L}(p(t)) \qquad \text{Eq. 3-3}$$

$$\Rightarrow m\,s^2\,X(s)\; + c\,s\,X(s) + k\,X(s) = P(s)$$

From Eq. 3-3 $X(s)$ can be solved as

$$X(s) = \frac{P(s)}{m\,s^2 + c\,s + k} \qquad \text{Eq. 3-4}$$

If, for example, $P(t)$ is a given force time function like a unit-step function $(1(t))$, which is defined as

$$1(t) = \begin{cases} 0, & t < 0 \\ 1, & t \geq 0. \end{cases} \qquad \text{Eq. 3-5}$$

Again looking at the table in Appendix A, it is known that

$$\mathcal{L}\big(1(t)\big) = \frac{1}{s} \qquad \text{Eq. 3-6}$$

Therefore the response of the system to a unit step input in Laplace domain

$$X(s) = \frac{1}{s(m\,s^2 + c\,s + k)} \qquad \text{Eq. 3-7}$$

To get the response in time domain, *inverse Laplace transform* can be used and it is defined as

$$\mathcal{L}^{-1}\big(F(s)\big) = f(t) = \frac{1}{2\pi\, i}\int_{\sigma_c+i\infty}^{\sigma_c+i\infty} F(s)e^{s\,t}\,ds \qquad \text{Eq. 3-8}$$

Where σ_c is a chosen value to the right of all the singularities of $F(s)$ in the s-plane. s-plane is the plane on which the s-type variables ($\sigma + i\,\omega$) can be shown. In fact, Eq. 3-8 is seldom used. Instead, complex Laplace transforms are broken down into simpler expressions that are listed in the tables along with their corresponding time responses (Appendix A). For example, if the numerical values of the physical properties are such that Eq. 3-7 can be written as

$$X(s) = \frac{1}{s(\,s^2 + 3\,s + 2)} \qquad \text{Eq. 3-9}$$

Using the partial-fraction expansion technique Eq. 3-10 can be broken down into simpler expressions

$$X(s) = \frac{\frac{1}{2}}{s} - \frac{1}{s+1} + \frac{\frac{1}{2}}{s+2} \qquad \text{Eq. 3-10}$$

Using the matching time functions from Appendix A, the corresponding time function of each component can be found and the total time response for $x(t)$ will be the sum of these time functions. Hence

$$x(t) = \frac{1}{2} - e^{-t} + \frac{1}{2}e^{-2t} \quad for\ t \geq 0 \qquad \text{Eq. 3-11}$$

3.2.2 The Block Diagram

A transfer function is defined as the ratio of the Laplace transform of the output to the Laplace transform of the input. In many control systems the dynamic equations can be written so that their components do not interact except by having the input of one transfer function as the output of another one. The dynamics of a system having multiple components are easier to represent in a block diagram form where each block represents the transfer function of one component (e.g. G1(s), G2(s) in Figure 3-4) and the input-output relationships between the blocks are shown by lines and arrows. The resulting transfer function for the whole can be obtained by block diagram algebra. This method is often easier and more informative than algebraic manipulation. Some examples for block diagrams and their equivalent algebraic input-output relationships are shown in Figure 3-4.

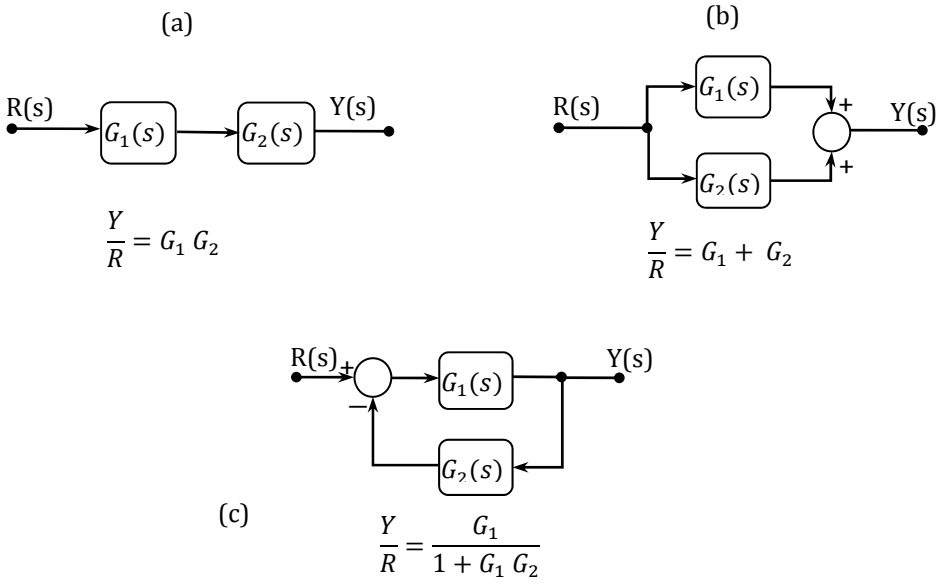


Figure 3-4 Three examples of elementary block diagrams

Figure 3-4(c) illustrates a negative feedback arrangement that is used to compare the output of a system with the command input to perform a tracking task. This is also referred to as closed-loop control as opposed to open-loop control. Open-loop control is generally simpler and does not introduce stability problems. Although feedback control is more complicated and may have stability issues, it has the potential to achieve a much better performance. Moreover, if the process is naturally (in open-loop) unstable, feedback control is the only possibility to attain a stable system that meets any performance criteria (Franklin et al. 2010).

3.2.3 S-plane, Poles and Zeros

In the design and analysis of a control system, the transfer function of the system gives useful information about the system characteristics including its frequency response.

The roots of the numerator of the system transfer function are called *zeros* of the system which correspond to the locations in the *s*-plane where the transfer function is zero. The roots of the denominator of the system transfer function are called the *poles* of the system. Apparently, the poles are the locations in the *s*-plane where the magnitude of the transfer function becomes infinite. For example, in the previous example the transfer function had no zero and three poles. The zeros and poles may be complex quantities and their location can be displayed in a complex plane, which is referred to as the *s*-plane.

The poles of the system determine its stability properties and also the natural or unforced behaviour of the system. Basically the shape of the response is determined by the poles. Figure 3-5 shows the responses in time domain associated with poles at different locations in the s-plane. As a general rule poles farther to the left in the s-plane (LHP in Figure 3-5) are associated with natural signals that decay faster than those associated with poles closer to the imaginary axis. Also as indicated in the figure, the poles with positive real values (in right half-plane, i.e. RHP) correspond to growing exponential functions which are unstable.

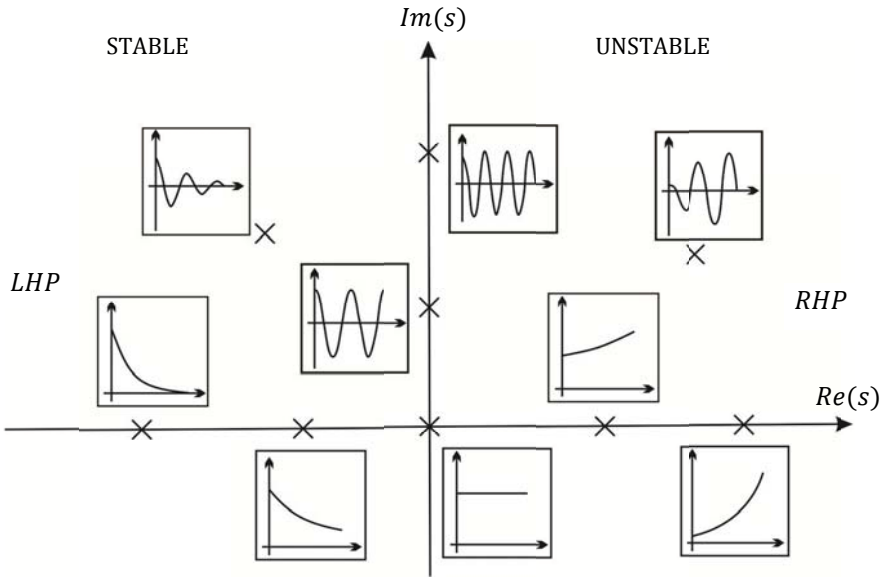


Figure 3-5 Time function associated with points in the s-plane (Franklin et al. 2010).

A feedback controller improves the dynamic response by mainly modifying the system's poles locations.

3.3 PID Controller Design

The fact that PID controllers are able to control complex systems without the need for precise identification of their dynamics has made them popular in control applications. A PID controller controls the dynamic error based on the magnitude, history and rate of the calculated error. The corresponding transfer function of a PID controller has three terms.

$$u(t) = K_p\, e(t) + K_i \int_0^t e(\tau)\; d\tau + K_d \frac{d(e(t))}{dt} \qquad \text{Eq. 3-12}$$

In the above equation $e(t)$ is the error signal and $u(t)$ is controller output. K_p, K_i and K_d are the gains (coefficients) corresponding to each of the terms. In fact designing a PID controller is to decide on a combination of these three gains to get the desired system behaviour. Figure 3-6 shows the block diagram of a PID controller.

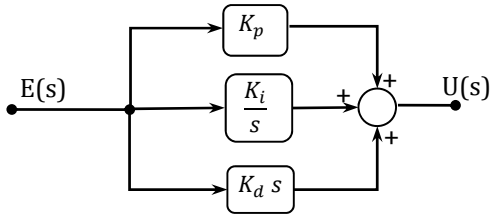


Figure 3-6 Block diagram of a PID controller

Therefore the transfer function for a PID controller is as below, which introduces an extra pole to the system.

$$\frac{K_D\, s^2 + K_P\, s + K_I}{s}$$ Eq. 3-13

A PID controller may be designed using the *root locus* method, *frequency response* method or can be tuned experimentally via an in-situ approach such as using Ziegler Nichols tuning rules(Franklin et al. 2010).

Root locus method studies the effect of any one parameter that enters the equation linearly to modify the location of system's poles in the s-plane. Typically that one parameter is chosen from one of the PID gains (where the others are expressed in relation to that one).

The use of frequency response methods is more common in the design of feedback control systems for industrial applications. One of the reasons is that with frequency response method it is easy to use experimental information for design purposes. Frequency response methods utilize *Bode* plots that portray the steady state response of a system to sinusoidal input.

When a representative analytical model for the system is not available, a PID controller can still be used by using experimental tuning approaches like Ziegler Nichols (Franklin et al. 2010).

3.4 State-Space Controller Design

Studying the system dynamic in state space form has the following main advantages (Franklin et al. 2010):

- Having the differential equations in state-variable form gives a compact standard form where multi-input multi-output systems can be studied easily even in the presence of nonlinearities.
- In contrast to transfer function which relates only the input to the output and does not show the internal behaviour of the system, the state form connects the internal variables to external inputs and outputs. This keeps the internal information at hand, which at times is important.

The state-variable representation of a continuous linear time-invariant open-loop dynamic system can be expressed as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\,\boldsymbol{x}(t) + \boldsymbol{B}\,\boldsymbol{u}(t) \qquad \text{Eq. 3-14}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\,\boldsymbol{x}(t) + \boldsymbol{D}\,\boldsymbol{u}(t)$$

where the $n \times 1$ column vector $\boldsymbol{x}(t)$ is called the state (vector) of the system, $\boldsymbol{u}(t)$ is the $m \times 1$ input vector, $\boldsymbol{y}(t)$ is the $p \times 1$ output vector, $\boldsymbol{A}$ is the $n \times n$ system matrix, $\boldsymbol{B}$ is the $n \times m$ input matrix, $\boldsymbol{C}$ is the $p \times n$ the output matrix, $\boldsymbol{D}$ is the $p \times m$ direct transmittance matrix; and, as can be understood from above, n, m, and p are the dimensions of state, input and output vectors, respectively.

As a result of the freedom in choosing the state vector, state space representation of a system is not unique. However, for a given system they are equivalent in terms of the input-output relationship.

The eigenvalues of matrix $\boldsymbol{A}$ are the roots of the characteristic equation (i.e., the roots of the denominator polynomial (poles) of the open-loop transfer function for a single-input single-output system).

In state-space method moving the closed-loop pole locations to desired locations is accomplished through a full state feedback. For the state-variable system described above, with full state feedback, the input vector becomes

$$\boldsymbol{u}(t) = \boldsymbol{r}(t) - \boldsymbol{K}\,\boldsymbol{x}(t) \qquad \text{Eq. 3-15}$$

Where $\boldsymbol{r}(t)$ is the $m \times 1$ reference input vector and $\boldsymbol{K}$ is an $m \times n$ gain matrix. For example if the reference input $\boldsymbol{r}(t)$ is zero (such a controller is called a regulator) for the closed loop system dynamics $\dot{\boldsymbol{x}}(t)$ becomes

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\,\boldsymbol{x}(t) + \boldsymbol{B}\left(-\boldsymbol{K}\,\boldsymbol{x}(t)\right) = (\boldsymbol{A} - \boldsymbol{B}\,\boldsymbol{K})\,\boldsymbol{x}(t) = \boldsymbol{A}_{cl}\,\boldsymbol{x}(t) \qquad \text{Eq. 3-16}$$

In this case, the eigenvalues of matrix $\boldsymbol{A}_{cl}$ (roots of $det(s\,\boldsymbol{I} - \boldsymbol{A}_{cl}) = 0$) are the closed-loop poles. It can be shown that the closed-loop poles of the system can be placed anywhere in the complex plane as long as all the states are controllable (How 2007). This is called pole-placement method. There is a function called *place* in the commercially available software package MATLAB which calculates matrix $\boldsymbol{K}$ to have the closed loop poles of the system move to the desired locations.

In the case of tracking a reference input ($\boldsymbol{r}(t) \neq 0$), the nonzero reference input needs to be introduced properly for a good performance in tracking.

This is done by scaling the reference input and then combining it with full state feedback to get the proper input vector (How 2007).

$$\boldsymbol{u}(t) = \bar{\boldsymbol{N}}\,\boldsymbol{r}(t) - \boldsymbol{K}\,\boldsymbol{x}(t) \qquad \text{Eq. 3-17}$$

where

$$\bar{\boldsymbol{N}} = -(\boldsymbol{C}(\boldsymbol{A} - \boldsymbol{B}\,\boldsymbol{K})^{-1}\boldsymbol{B})^{-1} \qquad \text{Eq. 3-18}$$

Eq. 3-17 ensures that for a step input there will be no steady state error after transient behaviour.

One way to select the location of closed loop poles is to consider treating the system as a second order system by selecting a pair of dominant poles, with the remaining poles having a real part corresponding to sufficiently damped modes. This will result in a system which is similar to a second order system (How 2007).

3.5 Nonlinear State Space Controller Design

In all of the aforementioned methods the dynamic system to be controlled was assumed to be linear. Most dynamic systems have some sort of nonlinearity. In some cases the nonlinearities may safely be ignored or linearized about an operating point. However there are systems where the nonlinearities cannot be ignored or the range of operation is beyond the limits where linear approximations are valid. In order to design a controller for such systems nonlinear techniques need to be used.

Instead of using linear approximations of the dynamics as done in Jacobian linearization, feedback linearization is a nonlinear control design approach which algebraically transforms nonlinear system dynamics into linear ones and permits the subsequent application of linear control techniques.

These nonlinear techniques make use of differential geometry concepts. In the following sections wherever a new differential geometric concept is used it is defined briefly, and also a more detailed summary of differential geometry is provided in Appendix B.

A single input nonlinear system in the neighbourhood of an equilibrium point, $\boldsymbol{x_e}$, corresponding to $u = 0$ i.e. $\boldsymbol{f}(\boldsymbol{x_e}) = \boldsymbol{0}$. can be expressed in state-variable form as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}\big(\boldsymbol{x}(t)\big) + \boldsymbol{g}\big(\boldsymbol{x}(t)\big)\, u(t) \qquad \text{Eq. 3-19}$$

or simply

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\, u \qquad \text{Eq. 3-20}$$

In Eq. 3-20 $\boldsymbol{f}$ and $\boldsymbol{g}$ are assumed to be smooth vector fields and $\boldsymbol{g}(\boldsymbol{x_e}) \neq \boldsymbol{0}$. A vector field is a map that assigns each $\boldsymbol{x}$ a vector $\boldsymbol{f}\ (\boldsymbol{x})$ of the same size. So for example, if $\boldsymbol{x}$ is a state vector of size $n \times 1$,

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} f_1(\boldsymbol{x}) \\ f_2(\boldsymbol{x}) \\ \vdots \\ f_n(\boldsymbol{x}) \end{pmatrix} \qquad \text{Eq. 3-21}$$

As will be elaborated later there are necessary and sufficient conditions under which the system defined by equation Eq. 3-20 is transformable into a linear controllable system by nonlinear feedback and coordinate transformation. This problem is called *feedback linearization*. Feedback linearization is viewed as a generalization of pole placement for linear systems.

The nonlinear single input system in Eq. 3-20 is said to be locally state feedback linearizable if it is locally feedback equivalent to a linear system in Brunovsky controller form (Marino and Tomei 1995) which is

$$\dot{\boldsymbol{z}} = \boldsymbol{A_c}\, \boldsymbol{z} + \boldsymbol{b_c}\, v \qquad \text{Eq. 3-22}$$

where the state vector and input in the new coordinate are $\boldsymbol{z}$ and v respectively and

$$\boldsymbol{A_c} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \qquad \text{Eq. 3-23}$$

$$\boldsymbol{b_c} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \qquad \text{Eq. 3-24}$$

where the state vector and input in the new coordinate are $\boldsymbol{z}$ and v respectively.

In order to be able to cast a set of equations in Brunovsky controller form, the theorem of feedback linearization needs to be satisfied. This theorem indicates that the single input system in Eq. 3-20 with n states is locally state feedback linearizable if and only if in a neighbourhood of origin:

(i) the distribution span$\{\boldsymbol{g}, \dots, ad_{\boldsymbol{f}}^{n-1}\boldsymbol{g}\}$ is of rank n, and

(ii) the distribution span$\{\boldsymbol{g}, \dots, ad_{\boldsymbol{f}}^{n-2}\boldsymbol{g}\}$ is *involutive* and of constant rank $n-1$.

Expressions like $ad_{\boldsymbol{f}}^{n-1}\boldsymbol{g}$ are iterative forms of Lie bracket which is a function in differential geometry acting on two vector fields like $\boldsymbol{f}$ and $\boldsymbol{g}$. Lie bracket function is illustrated in Appendix B. Also an exact mathematical explanation of distribution and its rank is given in Appendix B. However as a simple explanation, condition (i) requires that the space generated from the indicated vector fields has a dimension of n which means the vector fields have to be linearly independent. Condition (ii) requires that the space generated from the indicated vector fields has a dimension of $n-1$. Moreover a distribution is involutive if, given any two vector fields $\boldsymbol{f}$ and $\boldsymbol{g}$

belonging to that distribution, their *Lie bracket,* $[\boldsymbol{f}, \boldsymbol{g}]$, also belongs to the distribution.

The above two conditions guarantee the existence of a function $h: R^n \to R$ such that in the neighbourhood of origin, the following conditions are satisfied.

$$\langle dh\,, ad^{n-1}_{(-\boldsymbol{f})}\boldsymbol{g}\rangle \neq 0 \qquad \text{Eq. 3-25}$$

$$\langle dh\,, ad^{i}_{(-\boldsymbol{f})}\boldsymbol{g}\rangle = 0, \qquad 0 \leq i \leq n-2$$

In the above expressions dh is called *gradient* of h and is defined as

$$dh = \frac{\partial h}{\partial \boldsymbol{x}} = \left(\frac{\partial h}{\partial x_1}, \frac{\partial h}{\partial x_2}, \dots, \frac{\partial h}{\partial x_n}\right) \qquad \text{Eq. 3-26}$$

and *inner product* is defined as

$$\langle dh\,, \boldsymbol{f}\rangle = \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \left(\frac{\partial h}{\partial x_i}.\, f_i\right) \qquad \text{Eq. 3-27}$$

Having solved the conditions in Eq. 3-25 for h, the transformation from $\boldsymbol{x}$ to $\boldsymbol{z}$ will be

$$\boldsymbol{z} = (z_1, z_2, \dots, z_n)^T = \left(h(x), L_f h(x), \dots, L_f^{n-1} h(x)\right)^T \qquad \text{Eq. 3-28}$$

where the expression $L_{\boldsymbol{f}} h(x)$ is called the *Lie derivative* of function $h(x)$ along the vector field $\boldsymbol{f}$. This operator is also defined in Appendix B

Hence the dynamic system in Eq. 3-20 i.e.

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\, u \qquad \text{Eq. 3-29}$$

transforms into

$$\dot{z}_i = z_{i+1}, \qquad 1 \leq i \leq n-1 \qquad \text{Eq. 3-30}$$

$$\dot{z}_n = v = L_{\boldsymbol{f}}^{n} h(\boldsymbol{x}) + L_{\boldsymbol{g}} L_{\boldsymbol{f}}^{n-1} h(\boldsymbol{x})\, u \qquad \text{Eq. 3-31}$$

The system expressed above is dynamically equivalent to the original system which means they have identical poles. v is the input of the transformed system and u (input for the original system) can be calculated from it using equation Eq. 3-31. So the state feedback is

$$u = \frac{v - L_{\boldsymbol{f}}^{n} h(\boldsymbol{x})}{L_{\boldsymbol{g}} L_{\boldsymbol{f}}^{n-1} h(\boldsymbol{x})} \qquad \text{Eq. 3-32}$$

4 Implementation of Control Methods

General

In this chapter the control techniques introduced in Chapter 3 are utilized. As discussed before an actuator delay resulting in a time delay in experimental substructure can impair the dynamic stability and accuracy of the system. Control theory is used to design servo-hydraulic controller to minimize actuator delay.

In this study a linearized model of the system dynamics was used in the design of the controller using linear control design techniques. A nonlinear model of the system whose parameters were obtained through system identification (Mercan 2007) was used in the nonlinear state space control design and also in the numerical simulations. This is because the nonlinear model is accepted to provide a more realistic representation of the system dynamics.

4.1 Dynamic Model of a Servo-Hydraulic System

The dynamic equations that govern a servo-hydraulic system were explained in Chapter 2 and two versions of the model (linear and nonlinear)

were introduced. These dynamics are modelled in commercial software package MATLAB/Simulink to simulate the inner loop of a PSD test setup.

Linear Model

Equations for the linear model presented in Chapter 2 are summarized here.

Table 4-1 Linearized dynamics of a servo-hydraulic system

Valve spool (in s domain)	$\frac{X_v(s)}{I(s)} = \frac{k_v}{1+\tau\, s}$	(Eq. 2-1)
Valve flow	$Q_L = K_q\, x_v - K_c\, P_L$	(Eq. 2-20)
Actuator chamber pressure	$Q_L = A_p\, \dot{x}_p + C_{tp}\, P_L + \frac{V_t}{4\beta_e}\dot{P}_L$	(Eq. 2-18)
Actuator motion	$m_p\, \ddot{x}_p + b_p\ \dot{x}_p + F_{ext} = A_p\, P_L$	(Eq. 2-21)

Figure 4-1 shows the Simulink model for a servo-hydraulic system with the above dynamics that is connected to a linear single-degree of freedom test structure. This structure has a mass of m, damping of c and stiffness of k. As it is shown in the figure this adds a load dynamics to the above equations as

$$F_{ext} = m\, \ddot{x}_p + c\, \dot{x}_p + k\, x_p \qquad \text{Eq. 4-1}$$

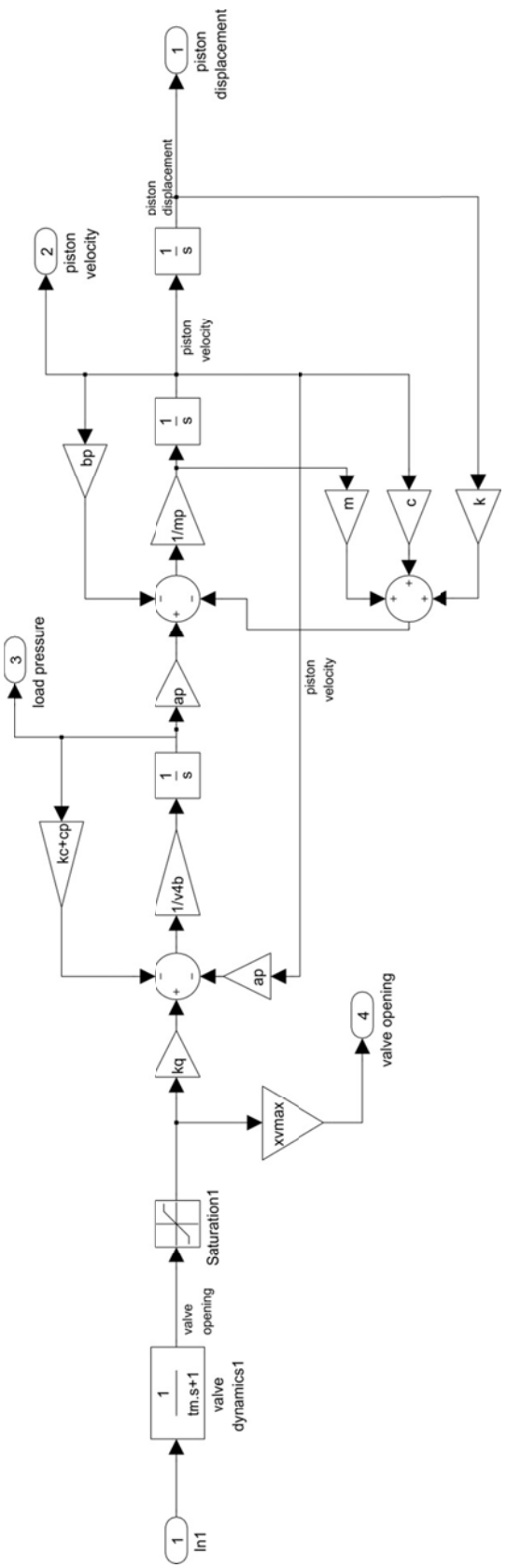


Figure 4-1 Simulink model for a servo-hydraulic system with linearized dynamics

Nonlinear Model

Equations for the nonlinear model are

Table 4-2 Nonlinear dynamics of a servo-hydraulic

Valve spool (in s domain)	$\frac{X_v(s)}{I(s)} = \frac{k_v}{1+\tau\, s}$	(Eq. 2-1)
Valve flow	$Q_L = \eta\, x_v \sqrt{P_s - sign(x_v) P_L}$	(Eq. 2-16)
Actuator chamber pressure	$Q_L = A_p\, \dot{x}_p + C_{tp}\, P_L + \frac{V_t}{4\beta_e} \dot{P}_L$	(Eq. 2-18)
Actuator motion	$m_p\, \ddot{x}_p + b_p\ \dot{x}_p + T_f + F_{ext} = A_p\, P_L$	(Eq. 2-19)

Figure 4-2 shows the Simulink model for a servo-hydraulic system of the above dynamics that is connected to a linear single-degree of freedom test structure.

The linear model will be used for designing a PID and a linear state space controller and the nonlinear model will be used to design a nonlinear state space controller.

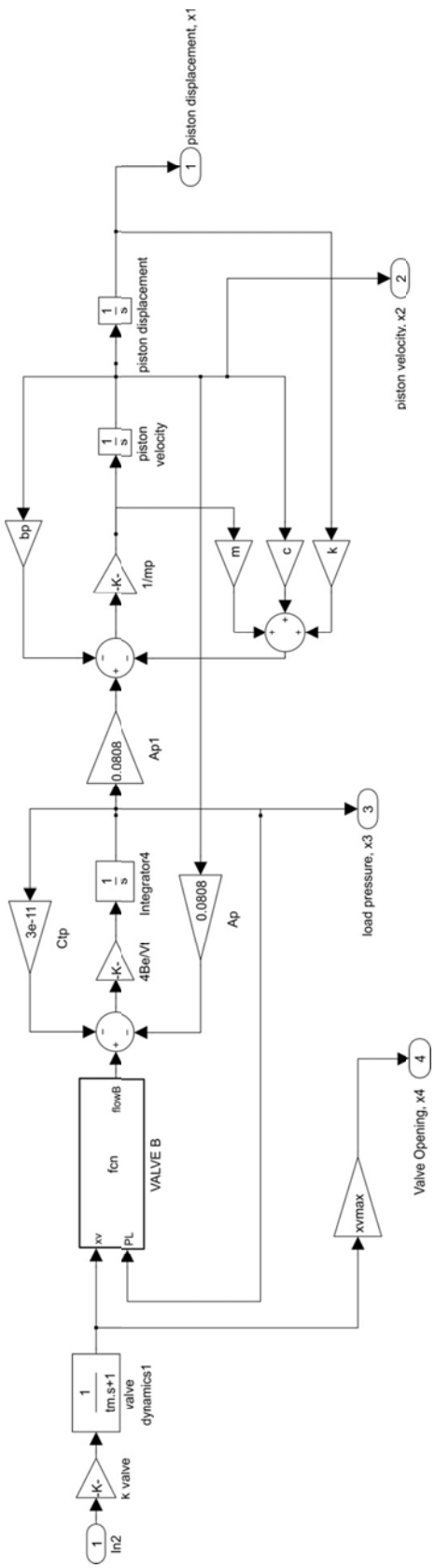


Figure 4-2 Simulink model for a servo-hydraulic system with nonlinear dynamics

4.2 PID Controller Design

Figure 4-3 shows the Simulink model for a servo-hydraulic system along with a PID controller. As it can be seen, the servo-hydraulic system is represented by a block which has an input as a command signal, which is issued by the controller, and four outputs that may be measured during a test and are namely piston displacement, piston velocity, load pressure and valve opening. In this study the nonlinear models introduced in Figure 4-2 was used and embedded to this block.

The following gives a summary of the roles of each PID gains (Ahmadizadeh 2007).

Proportional (K_p) – This is to handle the present requirements. The error is multiplied by K_p. Hence, the greater the proportional gain, the more the servo-valve opens for a given error. There is a trade-off for selecting an appropriate K_p. Although a large proportional gain may decrease the error resulting in a closer tracking of reference signal and reduced response time, it decreases the stability margin of the system and increases the frequency of the oscillation in the transient response.

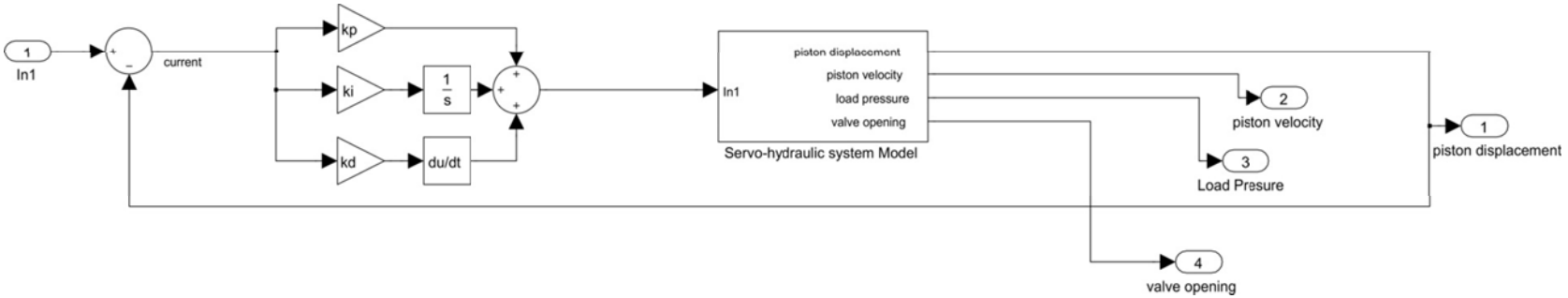


Figure 4-3 Simulink model for a servo-hydraulic system with a PID controller

Integral (K_I) – The error is integrated (added up) over a period of time, multiplied by a constant K_I and added to the control signal. A well-tuned PI controller will converge to the reference signal (zero steady-state error), leading to a reduced error between command and feedback.

Integral (K_D) – This is to handle the future requirements. The first derivative of the error over time is calculated, multiplied by K_D and added to the control signal. Basically, this term controls the response to a change in the system. In practice due to noises that enter the measured signal in a real PSD test, a controller without a derivative term may be used.

4.2.1 Controller Tuning

Tuning a controller is adjusting its parameters to get a desired system response. The goal is to have a system that has a stable and fast response (i.e., a short response time) with a small steady-state error.

One of the Ziegler-Nichols tuning methods is called ultimate sensitivity method and is used in this study. In this method the criteria for adjusting the parameters are based on evaluating the amplitude and frequency of the oscillations of the system at the limit of stability rather than taking the step response. To use the method, the proportional gain is increased until the system becomes marginally stable and continuous oscillations just begin, with amplitude limited by the saturation of the actuator. The corresponding gain is defined as K_u (known as ultimate gain) and the period of oscillation is P_u (known as ultimate period). P_u should be measured when the amplitude of

oscillation is as small as possible. Then the tuning parameters for a PI controller are selected as $k_p = 0.45\,K_u$ and $T_I = {}^{P_u}/_{1.2}$. Experience has shown that the controller settings according to Ziegler-Nichols rules provide acceptable closed-loop response for many systems. Ziegler-Nichols gives a good starting point for the controller parameters and the fine tuning of the controller is still needed to achieve a desired behaviour. Details for this method can be found in (Franklin et al. 2010).

Tuning the nonlinear model in Simulink results in $K_P = 80$, $K_I = 0.5$ and $K_D = 0$. And closed-loop poles of the system can be calculated as following.

$$p_1 = -242, p_2 = -141,\ \ p_{3,4} = -21.9 \pm 31.2\,i,\ p_5 = -0.00625$$

As shown in Equation 3-13, the transfer function of a PID controller adds a pole at origin to the open loop transfer function in Equation 2-23. Here, in the closed loop transfer function this extra pole moves a bit off the origin ($p_5 = -0.00625$).

4.3 Linear State-Space Controller Design

4.3.1 State-variable form of equations

As pointed out before, for a given system, depending on the states selected, there may be multiple state space representations. Using the linear equations introduced in Table 4-1, four states and one input can be chosen.

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_p \\ \dot{x}_p \\ P_L \\ x_v \end{pmatrix}, \quad u = i \qquad \text{Eq. 4-2}$$

Alternatively, if the servo-valve is assumed to have a fast response to the input signal compared to the rest of the system; its dynamics can be omitted without compromising the tracking capability.

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} x_p \\ \dot{x}_p \\ P_L \end{pmatrix}, \quad u = x_v \qquad \text{Eq. 4-3}$$

Considering these three states, the state-variable representation (Equation 3-14) of the system will be

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \dot{x}_p \\ \ddot{x}_p \\ \dot{P}_L \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ \dfrac{-k}{m_p + m} & \dfrac{-(b_p + c)}{m_p + m} & \dfrac{A_p}{m_p + m} \\ 0 & \dfrac{-A_p 4\,\beta_e}{V_t} & \dfrac{-(K_c + C_{tp})4\,\beta_e}{V_t} \end{pmatrix} \begin{pmatrix} x_p \\ \dot{x}_p \\ P_L \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \dfrac{K_q 4\,\beta_e}{V_t} \end{pmatrix} x_v \qquad \text{Eq. 4-4}$$

$$\boldsymbol{y} = x_p = (1 \quad 0 \quad 0) \begin{pmatrix} x_p \\ \dot{x}_p \\ P_L \end{pmatrix}$$

And if all four states are considered

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \dot{x}_p \\ \ddot{x}_p \\ \dot{P}_L \\ \dot{x}_v \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 1 & 0 & 0 \\ \dfrac{-k}{m_p + m} & \dfrac{-(b_p + c)}{m_p + m} & \dfrac{A_p}{m_p + m} & 0 \\ 0 & \dfrac{-A_p 4\,\beta_e}{V_t} & \dfrac{-(K_c + C_{tp})4\,\beta_e}{V_t} & \dfrac{K_q 4\,\beta_e}{V_t} \\ 0 & 0 & 0 & -\dfrac{1}{\tau} \end{pmatrix} \begin{pmatrix} x_p \\ \dot{x}_p \\ P_L \\ x_v \end{pmatrix} \qquad \text{Eq. 4-5}$$

$$+ \begin{pmatrix} 0 \\ 0 \\ 0 \\ \dfrac{k_v}{\tau} \end{pmatrix} i$$

It should be noticed that the valve spool dynamics (Equation 2-1) in time domain is

$$\dot{x}_v = -\frac{1}{\tau} x_v + \frac{k_v}{\tau} i \qquad \text{Eq. 4-6}$$

4.3.2 Pole Placement

Both three-state and four-state presentation of the system can be used to design the controller. For the four-state case, poles are selected equal to the poles corresponding to the previously designed system with PID controller ignoring the pole associated to PID itself, i.e.

$$p_1 = -242, p_2 = -141, \;\; p_{3,4} = -21.9 \pm 31.2\, i$$

Function *place.m* of commercial software package MATLAB is used to find matrix $\boldsymbol{K}$ and then using equation 3-18 matrix $\bar{\boldsymbol{N}}$ is calculated to introduce the signal input as

$$\boldsymbol{u}(t) = \bar{\boldsymbol{N}}\,\boldsymbol{r}(t) - \boldsymbol{K}\,\boldsymbol{x}(t) \qquad \text{Eq. 4-7}$$

Appendix C contains the MATLAB code which is used to calculate these matrices.

If the servo-valve spool dynamics is ignored (assumed to be faster than other parts of the system) considering only three states, the corresponding pole needs to be ignored too. As illustrated before, the poles far left in the complex plane are related to fast responses. Therefore pole $p = -242$ is the one associated to servo-valve spool dynamics. Consequently, the selected poles to design a three-state controller are

$$p_1 = -141, \;\; p_{2,3} = -21.9 \pm 31.2\, i$$

4.4 Nonlinear State-Space Controller Design

4.4.1 State-variable form of equations

State-variable form of a nonlinear system needs to be written in the form of

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\, u \qquad \text{Eq. 4-8}$$

It can be shown that the only way to write down the nonlinear equations in the above form is to consider all four states. So having

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_p \\ \dot{x}_p \\ P_L \\ x_v \end{pmatrix}, \quad u = i \qquad \text{Eq. 4-9}$$

state-variable form of the dynamics according to Table 4-2 will be

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \dot{x}_p \\ \ddot{x}_p \\ \dot{P}_L \\ \dot{x}_v \end{pmatrix} = \begin{pmatrix} \dot{x}_p \\ \dfrac{-k\,x_p - (b_p + c)\dot{x}_p + A_p\,P_L}{m_p + m} \\ \dfrac{4\,\beta_e}{V_t}\left(-A_p\,\dot{x}_p - C_{tp}\,P_L + \eta\,x_v\sqrt{P_s - \text{sign}(x_v)P_L}\right) \\ \dfrac{x_v}{\tau} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \dfrac{k_v}{\tau} \end{pmatrix} i \qquad \text{Eq. 4-10}$$

To follow the calculations the following notation will be used instead

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \underbrace{\begin{pmatrix} x_2 \\ \dfrac{-k\,x_1 - (b_p + c)x_2 + A_p\,x_3}{m_p + m} \\ \dfrac{4\,\beta_e}{V_t}\left(-A_p\,x_2 - C_{tp}\,x_3 + \eta\,x_4\sqrt{P_s - \text{sign}(x_4)x_3}\right) \\ -\dfrac{x_4}{\tau} \end{pmatrix}}_{\boldsymbol{f}(x)} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ \dfrac{k_v}{\tau} \end{pmatrix}}_{\boldsymbol{g}(x)} u \qquad \text{Eq. 4-11}$$

It was stated in Chapter 3 that $\boldsymbol{f}$ and $\boldsymbol{g}$ must be smooth vector fields but the sign function in the third term of $\boldsymbol{f}$ does not allow differentiation at $x_4 = 0$. Figure 4-4 shows that for a large coefficient n sign function can be approximated by using an Arc Tan. Mintsa et al. (2009) used an equation involving exponential terms to approximate the sign function.

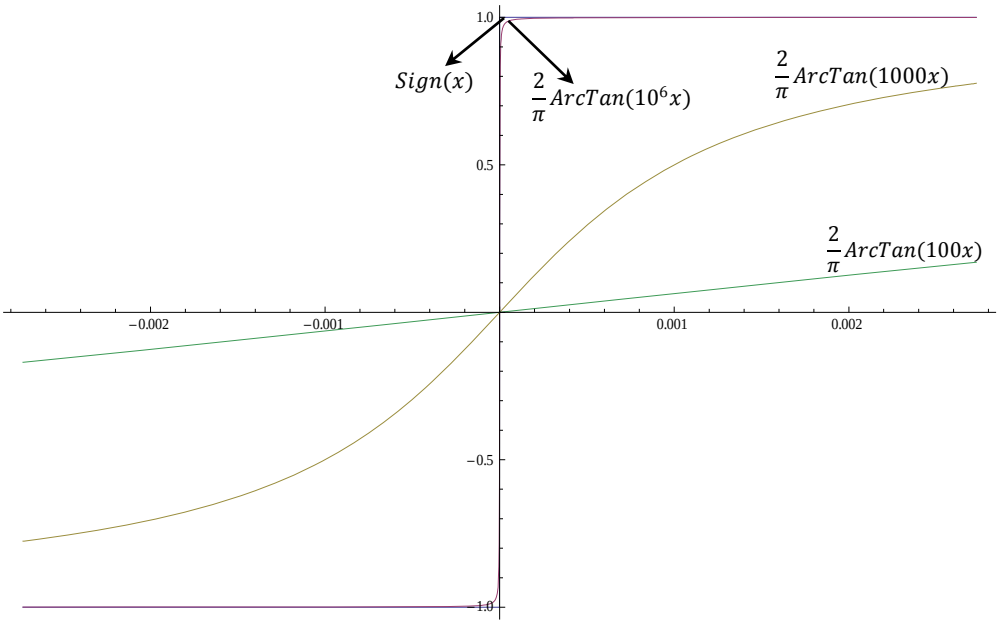


Figure 4-4 Approximation of function $Sign(x)$

Hence the dynamics in Eq. 4-11 can be approximated as

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \underbrace{\begin{pmatrix} x_2 \\ \frac{-k\,x_1 - (b_p + c)x_2 + A_p\,x_3}{m_p + m} \\ \frac{4\beta_e}{V_t}\left(-A_p x_2 - C_{tp} x_3 + \eta\, x_4 \sqrt{P_s - \frac{2}{\pi}ArcTan(n\,x_4)x_3}\right) \\ -\frac{x_4}{\tau} \end{pmatrix}}_{\boldsymbol{f}(\boldsymbol{x})} + \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{k_v}{\tau} \end{pmatrix}}_{\boldsymbol{g}(\boldsymbol{x})} u \qquad \text{Eq. 4-12}$$

$\boldsymbol{f}$ and $\boldsymbol{g}$ in Eq. 4-12 are smooth vector fields which can be used to design a nonlinear controller.

As it was stated in Chapter 3, for a system to be locally state feedback linearizable two conditions must be satisfied. These conditions are now checked for the above nonlinear system.

- Condition (i) the distribution span$\{\boldsymbol{g}, \ldots, ad_{\boldsymbol{f}}^{n-1}\boldsymbol{g}\}$ is of rank n.

$$\boldsymbol{g} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{k_v}{\tau} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \star \end{pmatrix}, \qquad \text{Eq. 4-13}$$

$$ad_{\boldsymbol{f}}^{1}\boldsymbol{g} = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\boldsymbol{g} = \begin{pmatrix} 0 \\ 0 \\ -\frac{4\,\beta_e}{V_t}\frac{k_v}{\tau}\eta\,\dfrac{\left(P_s - \dfrac{n\,x_3\,x_4}{\pi + n^2\,\pi\;{x_4}^2} - \dfrac{2x_3 ArcTan(n\,x_4)}{\pi}\right)}{\sqrt{P_s - \frac{2}{\pi}ArcTan(n\,x_4)x_3}} \\ \frac{k_v}{\tau^2} \end{pmatrix} \qquad \text{Eq. 4-14}$$

$$= \begin{pmatrix} 0 \\ 0 \\ \star \\ \star \end{pmatrix}$$

And in the same manner,

$$ad_{\boldsymbol{f}}^{2}\boldsymbol{g} = ad_{\boldsymbol{f}}^{1}\left(ad_{\boldsymbol{f}}^{1}\boldsymbol{g}\right) = \begin{pmatrix} 0 \\ \star \\ \star \\ \star \end{pmatrix} and \qquad \text{Eq. 4-15}$$

$$ad_{\boldsymbol{f}}^{3}\boldsymbol{g} = \begin{pmatrix} \star \\ \star \\ \star \\ \star \end{pmatrix} \qquad \text{Eq. 4-16}$$

In the above equations "$\star$" represents a nonzero expression. Since

$$\text{span}\left\{\begin{pmatrix} 0 \\ 0 \\ 0 \\ \star \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \star \\ \star \end{pmatrix}, \begin{pmatrix} 0 \\ \star \\ \star \\ \star \end{pmatrix}, \begin{pmatrix} \star \\ \star \\ \star \\ \star \end{pmatrix}\right\} \qquad \text{Eq. 4-17}$$

has a rank equal to 4, the first condition is satisfied.

- Condition (ii) the distribution span$\{\boldsymbol{g}, \dots, ad_{\boldsymbol{f}}^{n-2}\boldsymbol{g}\}$ is *involutive* and of constant rank $n-1$.

It is already known that

$$\text{span}\left\{\begin{pmatrix}0\\0\\0\\\star\end{pmatrix},\begin{pmatrix}0\\0\\\star\\\star\end{pmatrix},\begin{pmatrix}0\\\star\\\star\\\star\end{pmatrix}\right\} \qquad \text{Eq. 4-18}$$

has a rank of 3. However the distribution needs to be involutive.

It can be shown that

$$\left[\boldsymbol{g}, ad_{\boldsymbol{f}}^{1}\boldsymbol{g}\right] = \begin{pmatrix}0\\0\\\star\\0\end{pmatrix} \qquad \text{Eq. 4-19}$$

$$\left[\boldsymbol{g}, ad_{\boldsymbol{f}}^{2}\boldsymbol{g}\right] = \begin{pmatrix}0\\0\\0\\0\end{pmatrix} \qquad \text{Eq. 4-20}$$

$$\left[ad_{\boldsymbol{f}}^{1}\boldsymbol{g}, ad_{\boldsymbol{f}}^{2}\boldsymbol{g}\right] = \begin{pmatrix}0\\\star\\\star\\0\end{pmatrix} \qquad \text{Eq. 4-21}$$

All above vector fields belong to the distribution in Equation 4-15 which means the distribution is involutive and the second condition is also satisfied.

Therefore the nonlinear system is locally state feedback linearizable and there is a function like $h: R^4 \rightarrow R$ such that

$$\langle dh\,, ad^3_{(-\boldsymbol{f})}\boldsymbol{g}\rangle \neq 0$$

$$\langle dh\,, ad^i_{(-\boldsymbol{f})}\boldsymbol{g}\rangle = 0, \qquad 0 \leq i \leq 2$$

Eq. 4-22

The above conditions are examined in more detail in the following.

1. $\langle dh\,, \boldsymbol{g}\rangle = 0$, which gives

$$\left(\frac{\partial h}{\partial x_1}, \frac{\partial h}{\partial x_2}, \frac{\partial h}{\partial x_3}, \frac{\partial h}{\partial x_4}\right)\begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{k_v}{\tau} \end{pmatrix} = 0 \Rightarrow \frac{\partial h}{\partial x_4}\frac{k_v}{\tau} = 0 \Rightarrow \frac{\partial h}{\partial x_4} = 0 \qquad \text{Eq. 4-23}$$

2. $\langle dh\,, ad^1_{(-\boldsymbol{f})}\boldsymbol{g}\rangle = 0$, which in the same manner gives

$$\frac{\partial h}{\partial x_3} = 0 \qquad \text{Eq. 4-24}$$

3. $\langle dh\,, ad^2_{(-\boldsymbol{f})}\boldsymbol{g}\rangle = 0$, which also in the same manner gives

$$\frac{\partial h}{\partial x_2} = 0 \qquad \text{Eq. 4-25}$$

4. $\langle dh\,, ad^3_{(-\boldsymbol{f})}\boldsymbol{g}\rangle \neq 0$, which gives

$$\frac{\partial h}{\partial x_1} \neq 0 \qquad \text{Eq. 4-26}$$

It should be noted that the function h is not unique (Marino and Tomei 1995).

Function h may be selected to be

$$h = x_1 \qquad \text{Eq. 4-27}$$

As was introduced in Equation 3-25, coordinate transformation is defined as

$$\boldsymbol{z} = (z_1, z_2, z_3, z_4)^T = \left(h(\boldsymbol{x}), L_f h(\boldsymbol{x}), L_f^2 h(\boldsymbol{x}), L_f^3 h(\boldsymbol{x})\right)^T \qquad \text{Eq. 4-28}$$

So

$$z_1 = h(\boldsymbol{x}) = x_1 \qquad \text{Eq. 4-29}$$

$$z_2 = L_{\boldsymbol{f}} h(\boldsymbol{x}) = (1 \quad 0 \quad 0 \quad 0)\boldsymbol{f} = x_2 \qquad \text{Eq. 4-30}$$

$$z_3 = L_{\boldsymbol{f}}^2 h(\boldsymbol{x}) = (0 \quad 1 \quad 0 \quad 0)\boldsymbol{f} = \frac{-k\, x_1 - \left(b_p + c\right)x_2 + A_p\, x_3}{m_p + m} \qquad \text{Eq. 4-31}$$

$$z_4 = L_{\boldsymbol{f}}^3 h(\boldsymbol{x}) = \frac{1}{m_p+m}\Bigg(-k\, x_2 - \frac{b_p+c}{m_p+m}\left(-k\, x_1 - \left(b_p + c\right)x_2 + A_p\, x_3\right) + A_p \frac{4\,\beta_e}{V_t}\left(-A_p\, x_2 - C_{tp}\, x_3 + \eta\, x_4 \sqrt{P_s - \frac{2}{\pi} ArcTan(n\, x_4) x_3}\right)\Bigg) \qquad \text{Eq. 4-32}$$

This transformation of coordinates lets the system dynamics to be written in Brunovsky controller form as

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \dot{z}_3 = z_4 \\ \dot{z}_4 = v \end{cases} \qquad \text{Eq. 4-33}$$

where

$$v = L_f^4 h(\boldsymbol{x}) + L_g L_f^3 h(\boldsymbol{x})\, u \qquad \text{Eq. 4-34}$$

and in the above equation

$$L_f^4 h(\boldsymbol{x}) = \left(\frac{b_p k}{(m+m_p)^2} + \frac{c\,k}{(m+m_p)^2}\right) x_2 - \frac{k\left(-\frac{k\,x_1}{m+m_p} - \frac{(b_p+c)x_2}{m+m_p} + \frac{A_p x_3}{m+m_p}\right)}{m+m_p} +$$

$$\left(\frac{b_p(b_p+c)}{(m+m_p)^2} + \frac{c(b_p+c)}{(m+m_p)^2}\right)\left(-\frac{k\,x_1}{m+m_p} - \frac{(b_p+c)x_2}{m+m_p} + \frac{A_p x_3}{m+m_p}\right) -$$

$$\frac{{A_p}^2\left(-\frac{k\,x_1}{m+m_p} - \frac{(b_p+c)x_2}{m+m_p} + \frac{A_p x_3}{m+m_p}\right)}{(m+m_p)} \frac{4\,\beta_e}{V_t} -$$

$$\frac{1}{\tau_m}\, x_4\left(-\frac{A_p n\,\eta\, x_3\; x_4}{(m+m_p)\pi\frac{4\,\beta_e}{V_t}(1+n^2\; {x_4}^2)\sqrt{P_s - \frac{2\,x_3\,\mathrm{ArcTan}[n\,x_4]}{\pi}}} + \frac{A_p\eta\sqrt{P_s - \frac{2\,x_3\,\mathrm{ArcTan}[n\,x_4]}{\pi}}}{(m+m_p)} \frac{4\,\beta_e}{V_t}\right) + \qquad \text{Eq. 4-35}$$

$$\left(-\frac{A_p b_p}{(m+m_p)^2} - \frac{A_p c}{(m+m_p)^2}\right)\left(-A_p\, x_2\, \frac{4\,\beta_e}{V_t} - C_{tp}\, x_3\, \frac{4\,\beta_e}{V_t} +\right.$$

$$\left.\frac{4\,\beta_e}{V_t}\eta\; x_4\sqrt{P_s - \frac{2\,x_3\,\mathrm{ArcTan}[n\,x_4]}{\pi}}\right) + \left(-\frac{A_p C_{tp}}{(m+m_p)}\frac{4\,\beta_e}{V_t} -\right.$$

$$\left.\frac{4\,\beta_e}{V_t}\frac{A_p\eta\; x_4\,\mathrm{ArcTan}[n\,x_4]}{(m+m_p)\pi\sqrt{P_s - \frac{2\,x_3\,\mathrm{ArcTan}[n\,x_4]}{\pi}}}\right)\left(-\frac{4\,\beta_e}{V_t}A_p\, x_2 - \frac{4\,\beta_e}{V_t}C_{tp}x_3 +\right.$$

$$\left.\frac{4\,\beta_e}{V_t}\eta\, x_4\sqrt{P_s - \frac{2\,x_3\,\mathrm{ArcTan}[n\,x_4]}{\pi}}\right)$$

and

$$L_g L_f^3 h(\boldsymbol{x})$$

$$= \frac{k_v}{\tau_m}\left(-\frac{4\,\beta_e}{V_t}\frac{A_p\, n\ \eta\, x_3\, x_4}{(m+m_p)\pi(1+n^2 {x_4}^2)\sqrt{P_s - \frac{2x_3 \text{ArcTan}[n\, x_4]}{\pi}}}\right. \qquad \text{Eq. 4-36}$$

$$\left. + \frac{A_p \eta \sqrt{P_s - \frac{2x_3 \text{ArcTan}[n\, x_4]}{\pi}}}{(m+m_p)}\frac{4\,\beta_e}{V_t}\right)$$

Therefore the state feedback will be calculated as

$$u = \frac{v - L_f^4 h(\boldsymbol{x})}{L_g L_f^3 h(\boldsymbol{x})} \qquad \text{Eq. 4-37}$$

Chapter 5 illustrates implementation of the above coordinate transformation in a Matlab-Simulink model along with linear controllers.

5 Simulations Results

General

This chapter presents simulation results for the inner loop of PSD test models i.e. the control of servo-hydraulic-test structure system. Simulink, which is developed by MathWorks, is an interactive graphical environment for modeling, simulating and analyzing dynamic systems and it works with MATLAB. The Simulink models presented in this chapter include linearized and nonlinear model of a servo-hydraulic system discussed in Chapter 2 and the controllers discussed and implemented in Chapter 3 and Chapter 4 including PID, linear state-space and nonlinear state-space controllers. These simulations only consider the servo-hydraulic system connected to a test structure. To better understand the proposed control method, for a complete simulation of a PSD test, the outer-loop needs to be introduced where the integration algorithm and also the analytical substructure model (if it is a hybrid PSD test) reside.

5.1 Numerical Values for Servo-Hydraulic System and Test Structure in Linear and Nonlinear Models

The values of the parameters for linear and nonlinear formulation of dynamics of a servo-hydraulic system are given in Table 5-1. These numerical values are obtained through system identification for a servo-hydraulic system used in another PSD test research program (Zhang et al. 2005) and are assumed to be representative. For the structure selection, the important thing for an SDOF are the natural period (should be around 1 or 1.3 sec), damping (around 2 to 5 %). In a PSD test, it is desired to keep the inertial and damping forces analytical.

Table 5-1 Values for Parameters for Linear and Nonlinear Servo-Hydraulic System Models

parameter		value
m	mass of test structure	1025 kg
c	viscous damping coefficient of test structure	500 N − sec/m
k	stiffness test structure	$5.00E^6$ N/m
m_p	mass of actuator piston	1025 kg
b_p	viscous damping coefficient of actuator piston	356.18 E^3N − sec/m
A_p	actuator piston cross section area	0.0808 m^2
$\frac{V_t}{4\beta_e}$	V_t = actuator chamber volume β_e = oil modulus	5.65 E^{-10} m^3/Pa
P_s	supply pressure of the hydraulic system	207 E^5Pa
$\eta = C_d\Gamma\sqrt{1/\rho}$	C_d →orifice coeffiecient Γ →valve opening gradient ρ →hydraulic oil density	2.62 E^{-3} m^3/s /$Pa^{0.5}$
K_c	servo-valve flow-pressure coefficient	1.5 E^{-11} m^3/(Pa − sec)
K_q	servo-valve flow gain	0.035 m^3/sec
C_{tp}	actuator leakage coefficient	3 E^{-11} m^3/(Pa − sec)
k_v	servo-valve gain	0.9613
τ_m	servo-valve time constant	0.004 sec
$\omega_{n,exp}$	natural frequency of the test structure	69.8 1/sec
ζ_{exp}	damping ratio of the test structure	0.00349

5.2 Comparison of Controllers With and Without Saturation Limits

Numerical simulations were performed to compare the tracking capabilities of the PID, linear state-space and nonlinear state-space controllers. As explained in Chapter 4, all the controllers were designed to have the same dynamic characteristics. This was done by figuring out the pole locations from the tuned PID controller and using them for designing the linear and nonlinear state-space controllers. Linear state space controller uses the linearized state space representation (Table 4-1) in the pole placement procedure. On the other hand, the nonlinear controller design accounts for the nonlinearity in servo-valve flow-pressure relationship (Table 4-2). It utilizes a coordinate transformation to express the system in Brunovsky form which provides an equivalent linear system representation for the system dynamics. It should be noted that the calculated input to this equivalent system (v) has to be transformed back to the input of the original system (u) using the corresponding expression in Equation 4-34. While performing the numerical simulations, the dynamics of the servo-hydraulic test structure system was represented by the nonlinear model as it is presumed to be more representative of the real behaviour. Figure 5-1 shows step responses of the system with linear and nonlinear state-space controllers. It was observed that the PID response was almost identical to linear state-space controller which was expected as both were based on linearized representation of the system. Therefore only the linear state-space

response is shown. In Figure 5-1 the servo-valve is assumed to be able to open as much as needed (i.e., the system does not saturate). The nonlinear controller is observed to be faster than the linear one in the transient portion of the response. Moreover, in the steady state part of the response linear design has some steady state error while the nonlinear design has a zero steady state error.

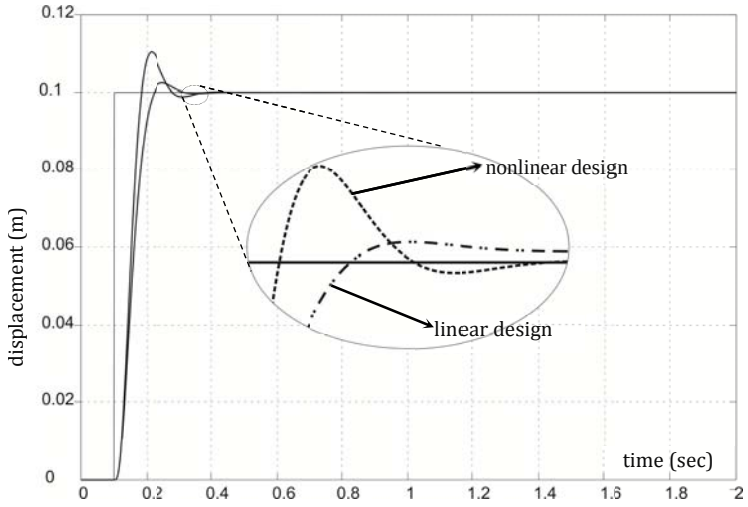


Figure 5-1 Step responses for a system without saturation

Figure 5-2 shows the step response for the same systems in Figure 5-1 with the difference that it takes saturation limits into account. As can be seen, when the servo-valve opening is limited, the nonlinear design is still performing better than the linear design during both transient and steady state response phases.

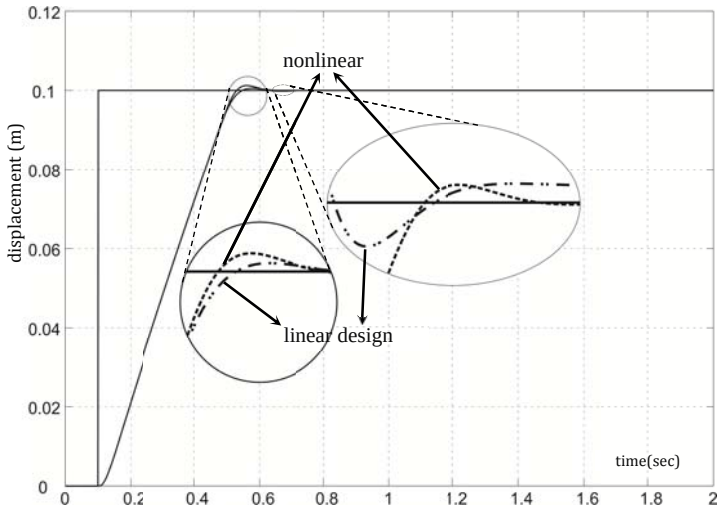


Figure 5-2 Step responses for a system with saturation

In Figure 5-3 response to a sinusoid input is shown for a system without saturation. Nonlinear state-space controller can be seen to have less time lag.

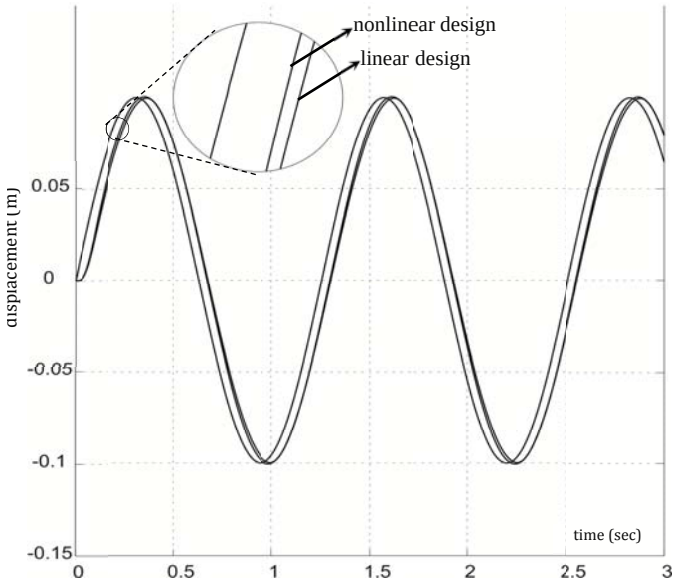


Figure 5-3 Response to sinusoid input

Figures 5-4 gives a comparison between nonlinear and linear state-space controllers looking at load pressure variation for the case when there is no

saturation in the servo-valve. Figures 5-5 gives the same comparison for the case when saturation happens in servo-valve.

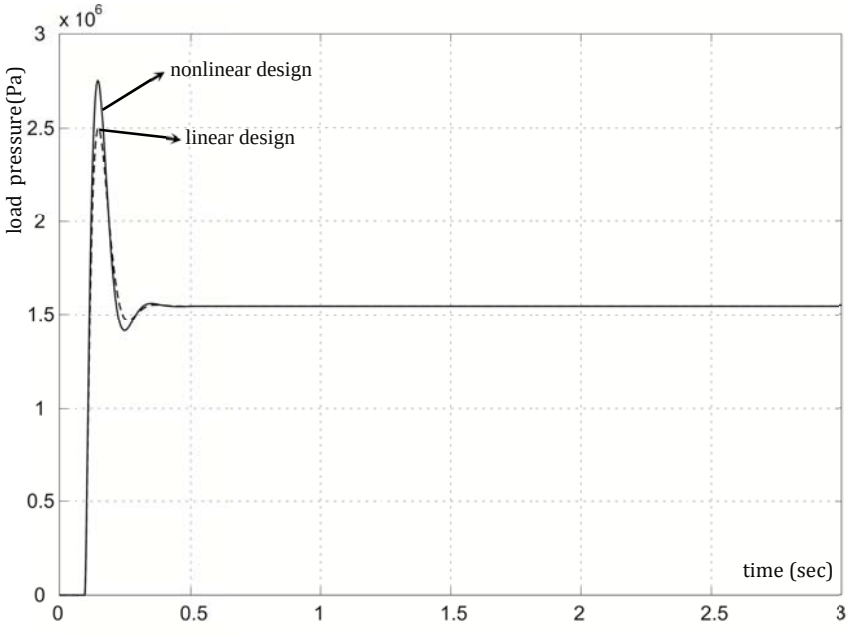


Figure 5-4 load pressure variation without servo-valve saturation

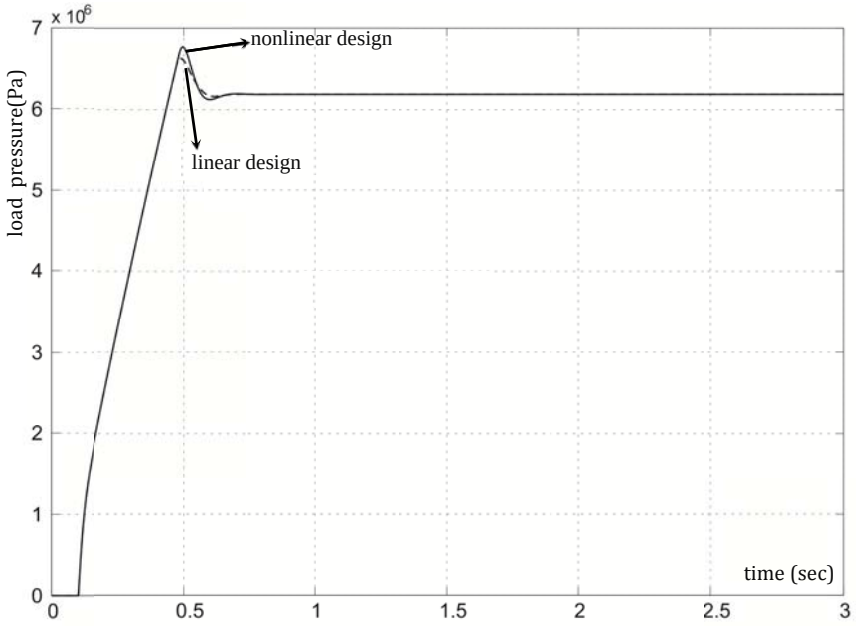


Figure 5-5 load pressure variation with servo-valve saturation

In addition, Figures 5-6 gives a comparison between nonlinear and linear state-space controllers looking at the servo-valve opening variation for the case when there is no saturation in the servo-valve. Figures 5-7 gives the same comparison for the case when saturation happens in servo-valve. It can be noticed that the better behaviour of the proposed nonlinear controller is achieved by a small change in servo-valve opening.

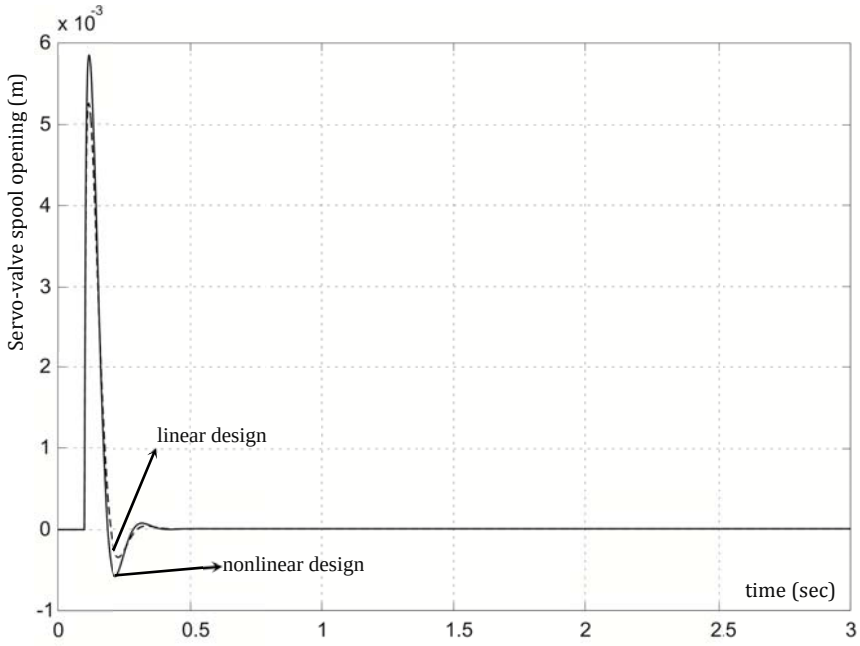


Figure 5-6 Servo-valve opening variation without servo-valve saturation

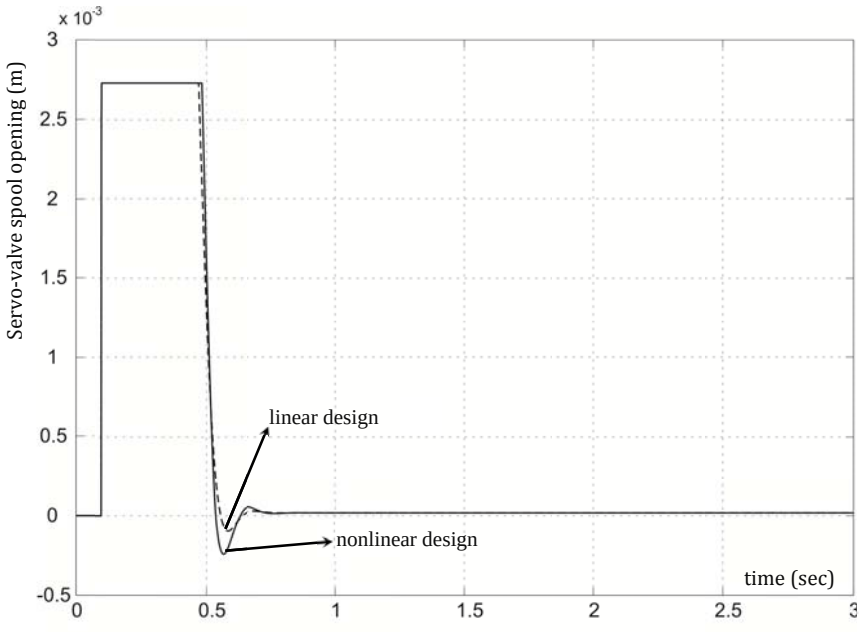


Figure 5-7 Servo-valve opening variation with servo-valve saturation

5.3 Implementing the Nonlinear State-Space controller in a PSD Test Simulation

The implementation of hybrid PSD test method in a flowchart is shown in Figure 5-8.

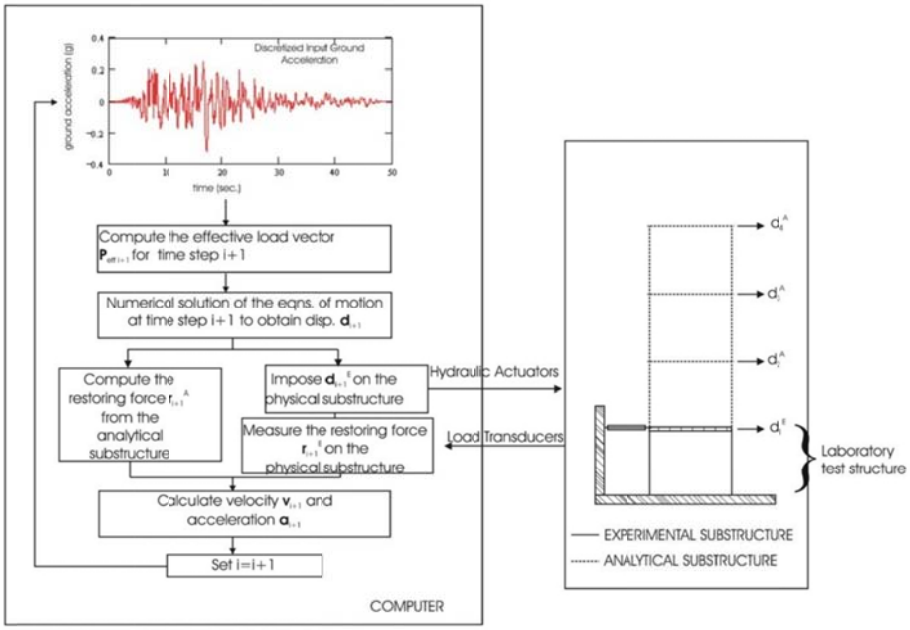


Figure 5-8 Hybrid PSD test method (Mercan 2007)

To investigate the behaviour of an elastomeric damper a hybrid PSD test was conducted at Lehigh University (Mercan 2007). Figure 5-9 gives the Simulink model of the test simulation that was used in the numerical simulations during that study, where a PID controller was used along with a velocity feed forward. In this model the same Simulink model is used to investigate the behaviour of the system when a NL state-space controller is used.

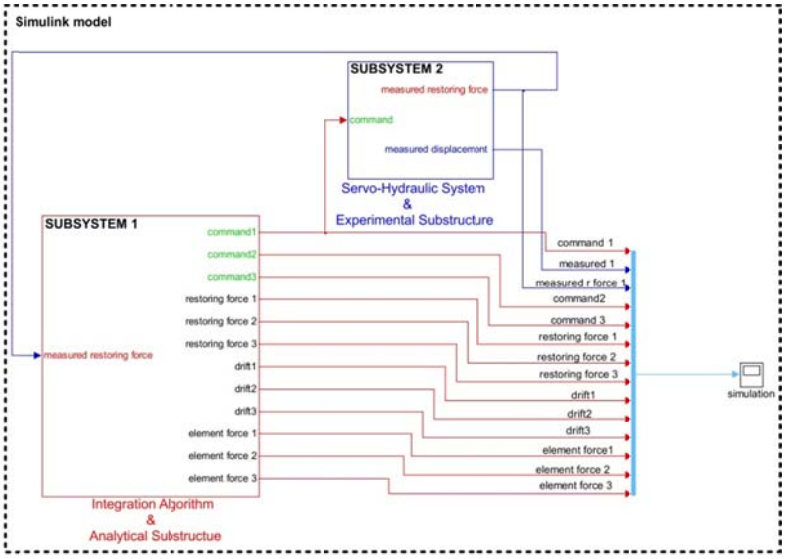


Figure 5-9 Simulink model for real-time hybrid PSD testing with MDOF analytical substructure and an elastomeric damper as the experimental substructure

As shown in Figure 5-9 subsystem 2 consists of the servo-hydraulic system and the experimental substructure. Figure 5-10 shows details for subsystem 2 when a PID controller is used and Figure 5-11 shows details for subsystem 2 when a NL state-space controller is used. As it can be seen in these figures the servo-hydraulic system has two similar servo-valves.

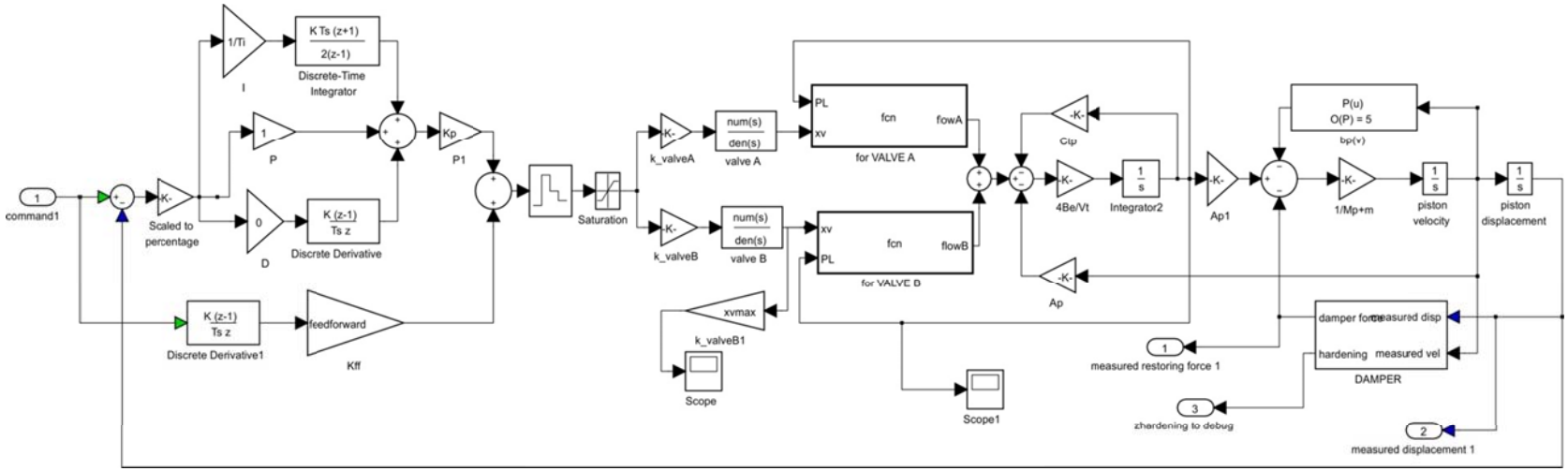


Figure 5-10 Simulink model for a servo-hydraulic subsystem with a PID controller

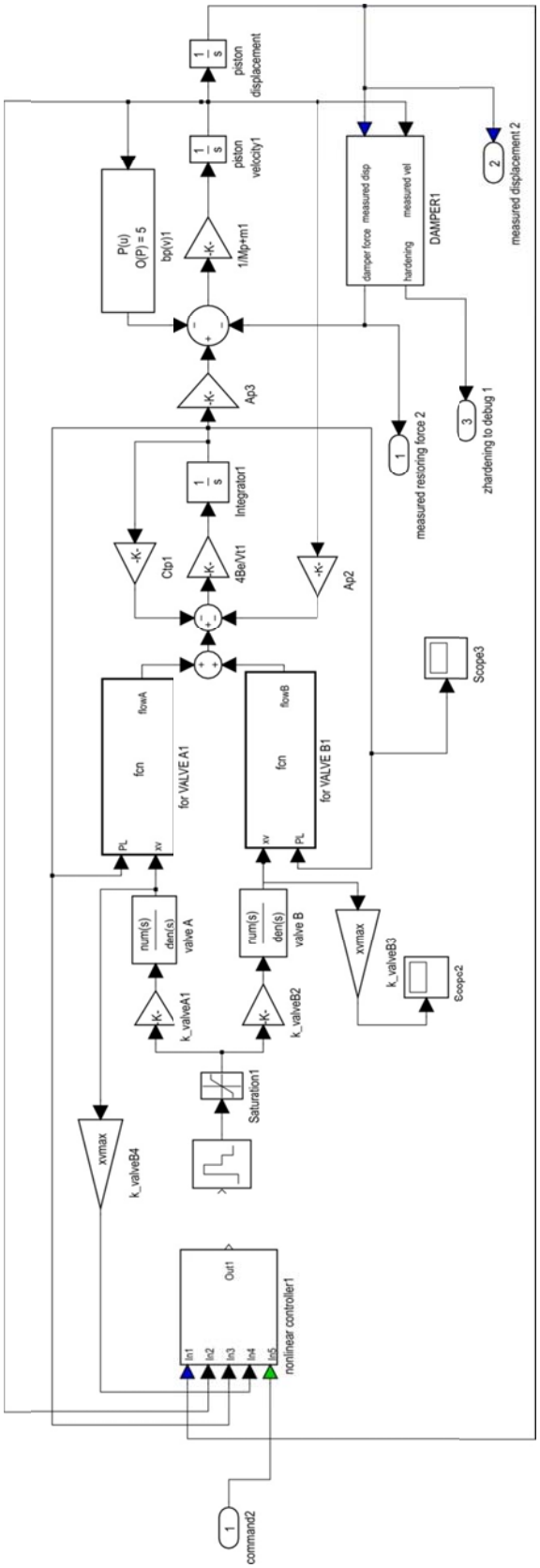


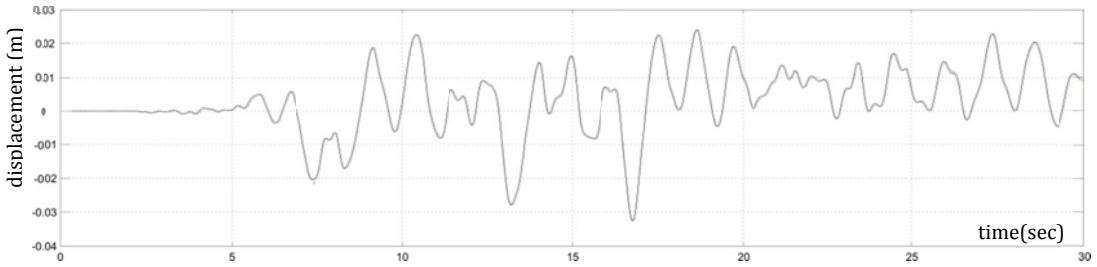
Figure 5-11 Simulink model for a servo-hydraulic subsystem with a NL state-space controller

Since the behaviour of the elastomeric damper is nonlinear, approximate values of stiffness and damping was used to design the state-space controller. Again, pole placement was done such that both systems have identical pole locations. Canoga Park earthquake ground motion was used in the following simulations. All the parameters used in design were the same as in Table 5-1 except for the test structure parameters that are as below.

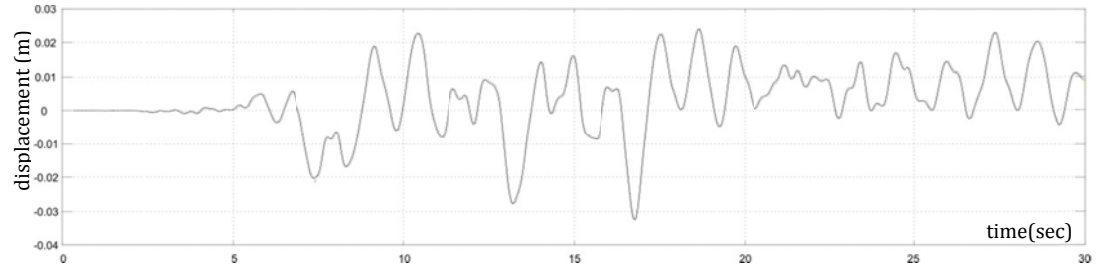
Table 5-2 Approximate values for the elastomeric damper parameters

parameter		value
m	mass of test structure	1335 kg
c	viscous damping coefficient of test structure	$0.5618E^6$N $-$ sec/m
k	stiffness test structure	$3.92E^6$ N/m

Figure 5-12 is the result simulations for PID and NL sate-space controller. In each of the figures the measured displacement is given against the command response from the controller.



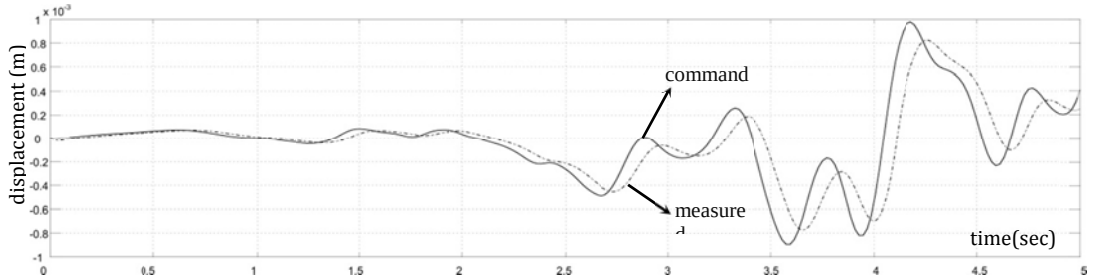
(a)



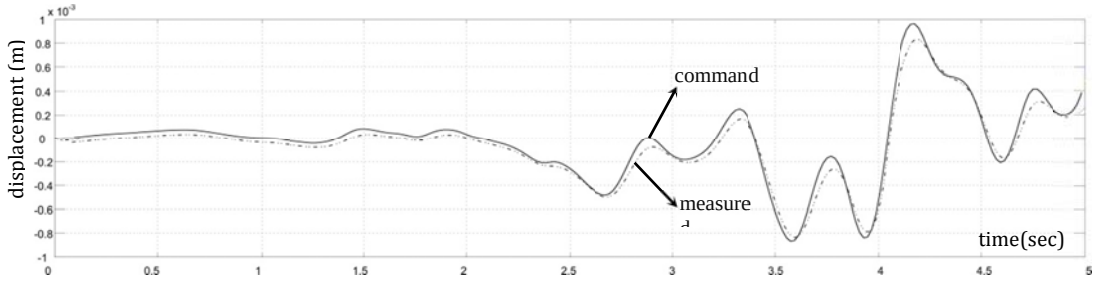
(b)

Figure 5-12 comparison between command and measured displacement (a)PID controller (b) NL controller

For both controllers, command and measured displacements are almost overlapping in this time scale. Hence a narrower view of the responses is given in Figure 5-13.



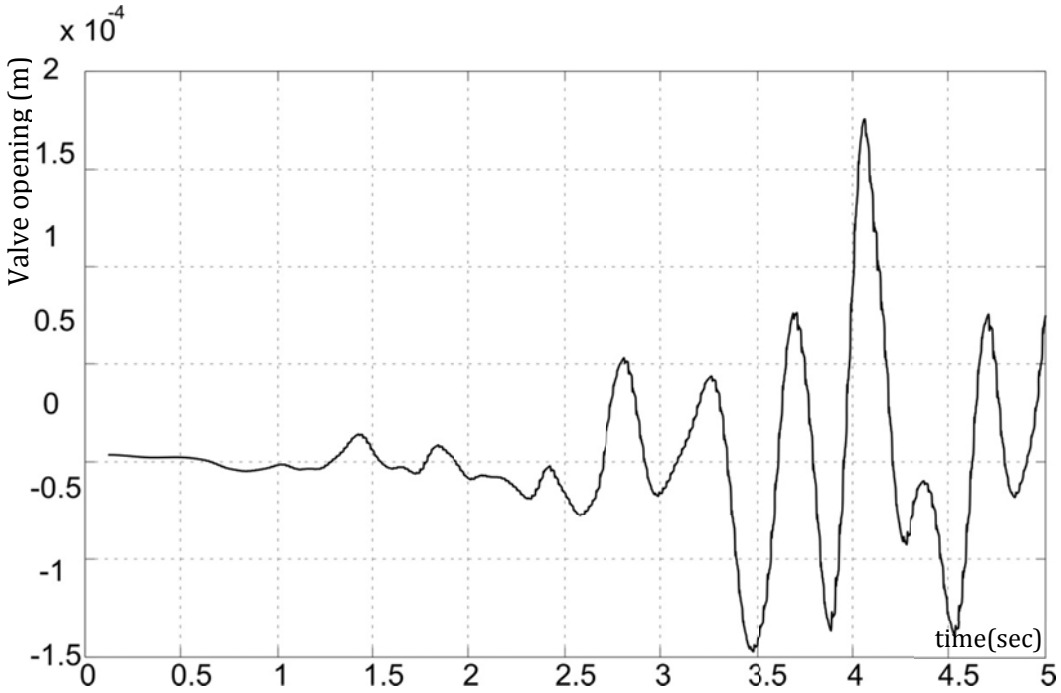
(a)



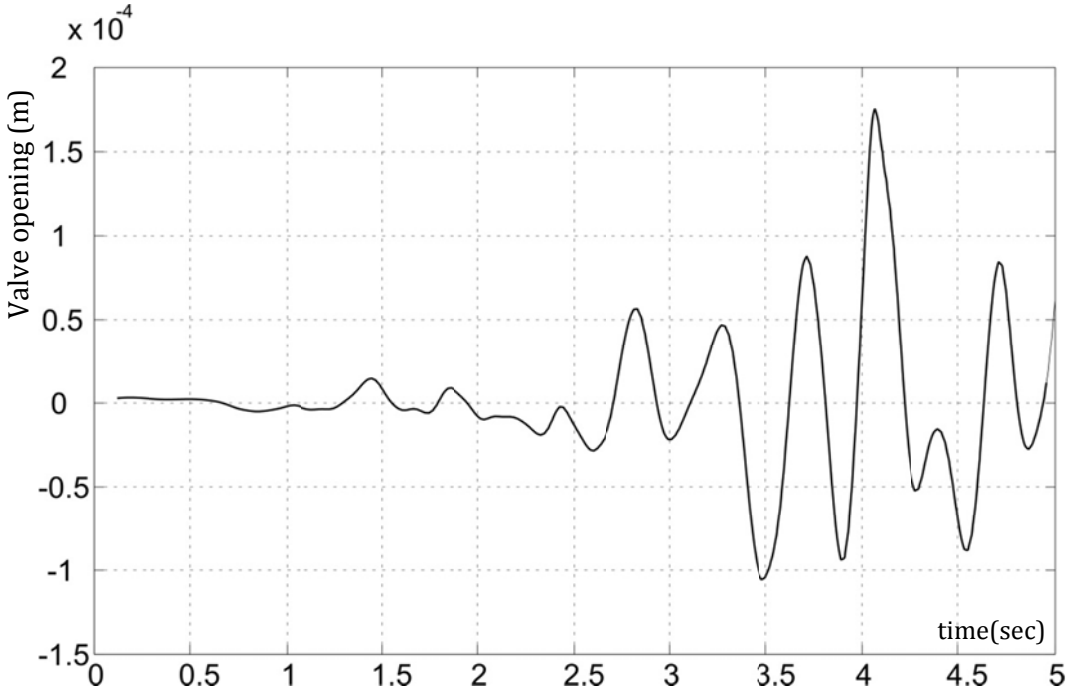
(b)

Figure 5-13 comparison between command and measured displacement (a)PID controller (b) NL controller

It can be seen that the NL controller is improving the tracking to some extent. This much of improvement could also be achieved by adding a velocity feed forward to PID controller but as shown in 5-14 (a) adding a velocity feed forward may distort the spool opening response. 5-14 (b) shows that spool opening has a much smoother behaviour when NL state-space controller is used.



(a)



(b)

Figure 5-14 comparison between servo-valve spool opening (a)PID controller with a velocity feed forward (b) NL controller

5.4 Robustness of the Control Design

There is always the possibility that the identified values of physical parameters used in the controller block are inaccurate and different from what is really in the system. The robustness of the control design needs to be examined with respect to potential errors in the identified physical parameters. Mintsa et al. (2009) addressed the robustness regarding the variations in supply pressure by designing a Lyapunov approach to derive an enhanced feedback-linearization-based control law. Herein to investigate the robustness of the proposed control design, the values of some identified physical parameters in the controller block were contaminated by some percent of over/underestimation error while their values in the system model remained intact. b_p, C_{tp}, $\frac{V_t}{4\beta_e}$ and η are the parameters considered. Supply pressure is considered almost constant due to the presence of accumulators.

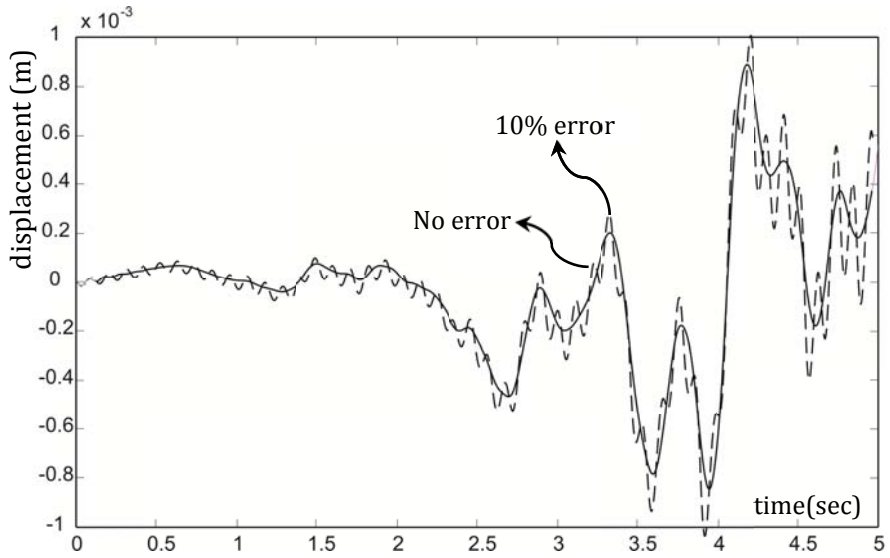


Figure 5-15 effect of 10% error in b_p on the response

Figure 5-15 compares the case when the value for b_p in the controller is overestimated by 10% with the case when there is no error. It is readily seen that 10% of error has made the system unstable. A similar graph was conducted for 5% error (not shown in figure for sake of clarity) and the control design was able to handle it without stability problems. Similar comparisons were done for negative percent errors (underestimation) wherein the accuracy did not change much and there were no instability problems.

Conducting the same procedure for C_{tp}, $\frac{V_t}{4\beta_e}$ and η showed that the proposed control design can easily handle errors within 20% of the real value.

5.5 Conclusion

The controller's ability to track the command displacements accurately is crucial in the overall stability and accuracy of the real-time PSD testing. For the displacement control loop, it was shown that a controller with improved performance can be designed using nonlinear state space control design techniques, provided that a representative model of the system is available.

Simulations were done for a servo-hydraulic system attached to a linear structure for cases with and without saturation in servo-valve and it was concluded that a NL controller can improve the tracking capabilities.

Moreover, the proposed NL controller was compared to PID controller by implementing it in a real-time PSD test simulation to show that it can decrease the time lag of the measured displacement.

Robustness of the nonlinear control design with respect to some identified physical parameters was investigated and it was observed that errors within reasonable range can be easily handled in terms of instability without loss of accuracy.

Future research is needed on application of the nonlinear controller to multi-input, multi-output system control together with a comprehensive numerical study and experimental validation.

6 Test setup design

The design of the test specimen considers the displacement and force limitations of the existing actuators. These actuators have a stroke range of ±5" (12.7 cm) and have a force limit of 5.5 Kip (24.5 kN).

The aim is to design a test setup that exhibits nonlinear behaviour within above actuators limitations. Figure 3-1 shows a sketch of the test setup. This single-degree-of-freedom system consists of a short column with wide flange section that is hinged to a support by means of a plate and a clevis. As it is seen in the figure, replaceable steel coupons are used to provide moment resistance for the support. Since the moment resistance capacity of the coupons is smaller than the columns, the nonlinear behaviour (yielding) initiates in the coupons and the column is expected to remain elastic. Therefore, after each nonlinear PSD test, the sacrificial coupons can be replaced easily at a low cost.

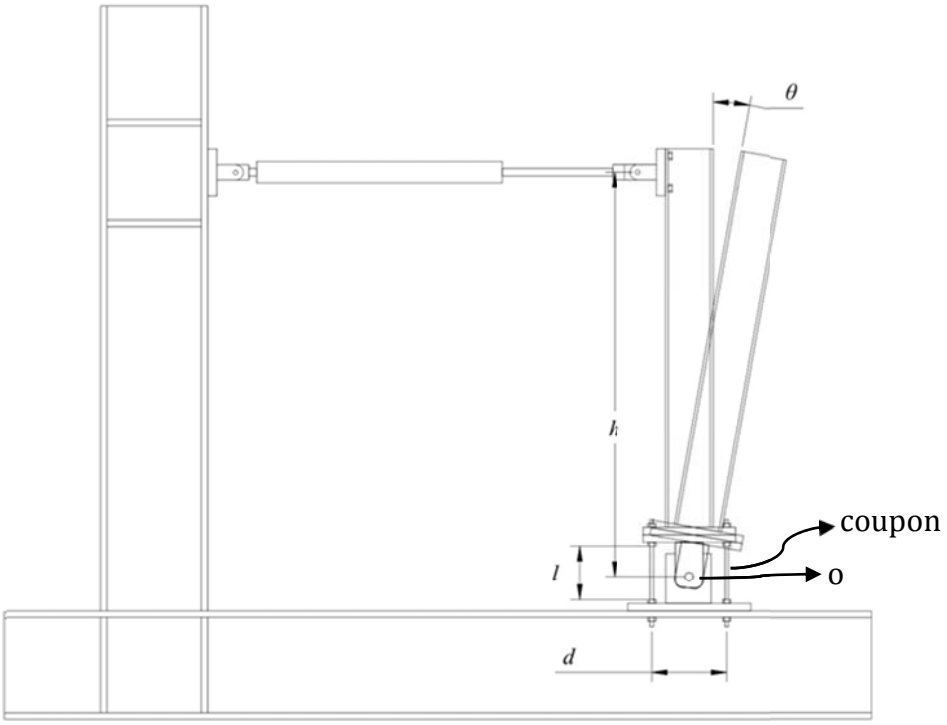


Figure 6-1 Experimental setup for PSD test

Looking at the deflected shape and assuming that two pairs of coupons have been used, the internal force in the coupons (T) at either side of the hinge can be worked out as the product of their deflection and axial stiffness.

$$T = \frac{d}{2}\theta \ \frac{2\,E\,A}{l} \qquad \text{Eq. 6-1}$$

Two of the coupons will experience tension and the other two will be in compression. In the above equation $\frac{d}{2}$ is the horizontal distance between the hinge and the coupons in the direction of applied force. l is the length of the coupons. E is the modulus of elasticity of the coupon material and A is the cross sectional area of one coupon.

Also it is obvious that the force in the coupons cannot exceed their yielding force, so

$$T_{max} = 2\, f_y\, A \qquad \text{Eq. 6-2}$$

T_{max} is the maximum force that two coupons can provide at one side of the beam and f_y is yielding stress of the coupon steel.

As mentioned before the coupons are intended to experience some nonlinearity. The rotation of the base at which the coupons yield (θ_y) can be calculated by equating T in Eq. 6-1 to T_{max} in Eq. 6-2.

$$\frac{d}{2}\theta_y \frac{2\, E\, A}{l} = f_y A \Rightarrow \theta_y = \frac{f_y\, l}{d\, E} \qquad \text{Eq. 6-3}$$

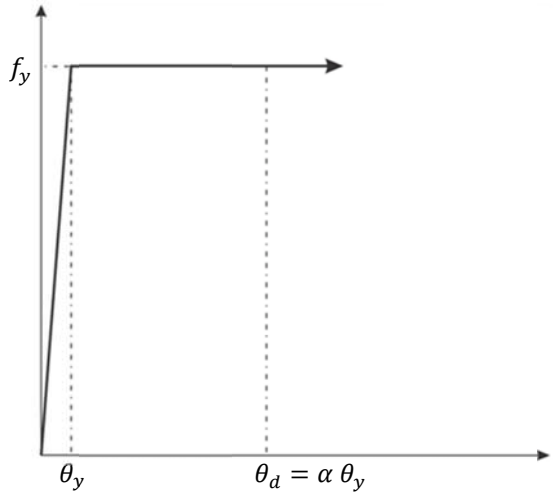


Figure 6-2 Expected base rotation at base used in design

In order to be able to observe nonlinear response in the experimental test set-up, the design base rotation θ_d will be set higher than the yield rotation θ_y.

$$\theta_d = \alpha\,\theta_y = \alpha\frac{f_y\,l}{d\,E} \quad where\ \alpha > 1 \qquad \text{Eq. 6-4}$$

On the other hand, because of the actuator stroke limit, test setup cannot move more than 5" (12.7 cm) to either side. Hence according to the Figure 3-1

$$h\sin(\theta_d) \leq 5" \qquad \text{Eq. 6-5}$$

or approximately

$$h\,\theta_d \leq 5" \Rightarrow h\,\alpha\frac{f_y\,l}{d\,E} \leq 5" \Rightarrow h \leq 5"\frac{d\,E}{\alpha\,l\,f_y} \qquad \text{Eq. 6-6}$$

Also for a single-degree of freedom system such as the one in Figure 3-1, ignoring the self-weight of the system, static equilibrium of moments about point *o* gives

$$p\,h = T\,d \Rightarrow p = \frac{T\,d}{h} \qquad \text{Eq. 6-7}$$

And the force applied by the actuator cannot exceed 5.5 Kip (24.5 kN). So,

$$p_{max} = \frac{T_{max}\, d}{h} \leq 5.5 \text{ Kip} \qquad \text{Eq. 6-8}$$

which according to Eq. 6-2 gives

$$\frac{2\, f_y\, A\, d}{h} \leq 5.5 \text{ Kip} \Rightarrow \frac{1}{h} \leq \frac{5.5 \text{ Kip}}{2\, f_y\, A\, d} \qquad \text{Eq. 6-9}$$

Combining Eq. 6-6 with Eq. 6-9 shows that the column needs to be in the following range.

$$\frac{2\, f_y\, A\, d}{5.5 \text{ Kip}} \leq h \leq 5" \frac{d\, E}{\alpha\, l\, f_y} \qquad \text{Eq. 6-10}$$

It was decided to use a wide-flange section of $W6 \times 20$ for the column and for the coupons a diameter of 0.25" (0.635 cm) was selected that gives an area of $A = 98.2 \times 10^{-3}$ in^2 (0.634 cm). The clevises available in the structural laboratory at the University of Alberta dictates a coupon length of $l = 8.13"$(20.65cm). The distance between the coupons, as indicated in the Figure 3-1, .was chosen to be $d = 6"$(15.24cm).

Assuming $f_y = 34$ ksi (234 $\frac{\text{N}}{\text{mm}^2}$), $E = 29000$ ksi ($2 \times 10^5 \frac{\text{N}}{\text{mm}^2}$) and $\alpha = 3$, Eq 3-10 becomes

$$2\frac{34\text{ ksi }(98.2)10^{-3}\text{ in}^2\ 6"}{5.6\text{ kips}} \le h \le 5"\frac{6"\ 29000\text{ ksi}}{3(8.13")\ 34\text{ ksi}} \qquad \text{Eq. 6-11}$$

$$\Rightarrow 7.16" \le h \le 1000"$$

As can be observed for height of the column there is a wide range to select from. However one should be aware that the upper limit is corresponding to $\alpha = 3$. That implies that any height less than $1000''$ corresponds to bigger values of α. According to the space available in the structural lab a height of $50''$(127cm) was chosen to be used.

Buckling of the coupons also needs to be considered. Knowing that the ends of the coupons are fixed, buckling load of a coupon is

$$\frac{\pi^2\ E\ I}{(k\ l)^2} = \frac{\pi^2\ \ 29000\text{ ksi }\ \frac{\pi}{4}\left(\frac{0.25''}{2}\right)^4}{(0.5\ \ 8.13'')^2} = 3.32\text{ kips (14.8 kN)}$$

while the yielding force is

$$f_y\ A = 34\ \ \frac{\pi\ 0.25^2}{4} = 1.67\text{ kips(7.43 kN)}$$

This means that the coupons will yield long before buckling.

Appendix A

Table of Laplace Transforms

Number	$F(s)$	$f(t), t \geq 0$
1	1	$\delta(t)$
2	$\frac{1}{s}$	$1(t)$
3	$\frac{1}{s^2}$	t
4	$\frac{2!}{s^3}$	t^2
5	$\frac{m!}{s^{m+1}}$	t^m
6	$\frac{1}{(s+a)}$	$e^{-a\,t}$
7	$\frac{1}{(s+a)^2}$	$t\,e^{-a\,t}$
8	$\frac{1}{(s+a)^m}$	$\frac{1}{m-1}t^{m-1}\,e^{-a\,t}$
9	$\frac{a}{s(s+a)}$	$1-e^{-a\,t}$
10	$\frac{a}{s^2(s+a)}$	$\frac{1}{a}(a\,t-1+e^{-a\,t})$
11	$\frac{b-a}{(s+a)(s+b)}$	$e^{-a\,t}-e^{-b\,t}$

12	$\frac{s}{(s+a)^2}$	$(1-a\,t)e^{-a\,t}$
13	$\frac{a^2}{s(s+a)^2}$	$1-e^{-a\,t}(1+a\,t)$
14	$\frac{(b-a)s}{(s+a)(s+b)}$	$b\,e^{-b\,t}-a\,e^{-a\,t}$
15	$\frac{a}{(s^2+a^2)}$	$\sin(a\,t)$
16	$\frac{s}{(s^2+a^2)}$	$\cos(a\,t)$
17	$\frac{s+a}{(s+a)^2+b^2}$	$e^{-at}\cos(b\,t)$
18	$\frac{b}{(s+a)^2+b^2}$	$e^{-at}\sin(b\,t)$
19	$\frac{a^2+b^2}{s[(s+a)^2+b^2]}$	$1-e^{-at}\left(\cos(b\,t)+\frac{a}{b}\sin(b\,t)\right)$

Appendix B Some on Differential Geometry (Lynch 2009)

1 Changes of Coordinates or Diffeomorphisms

A nonlinear change of coordinates $\boldsymbol{z} = \boldsymbol{z}(\boldsymbol{x})$ is a function defined on $U \subseteq \mathbb{R}^n$ and mapping to $\mathbb{R}^n$. It must have the properties

1. $\boldsymbol{z}(\boldsymbol{x})$ is an invertible function, i.e., there must exist an inverse function $\boldsymbol{z^{-1}}$ such that

$$\boldsymbol{z^{-1}}(\boldsymbol{z}(\boldsymbol{x})) = \boldsymbol{x}, \qquad \forall \boldsymbol{x} \in U$$

2. Both $\boldsymbol{z}$ and $\boldsymbol{z^{-1}}$ are C^∞ mappings.

When $U \subseteq \mathbb{R}^n$ the change of coordinates is called *global*. When this is not the case, the change of coordinates is said *local*. A sufficient condition for a mapping to be a *local* change of coordinates is given by the Inverse Function Theorem (Marino and Tomei 1995). Note that Condition 2 is required, i.e., that the change of coordinate is C^∞, since these coordinates changes will be applied to the state of a system, and the systems are required to be expressed in the new coordinates also to be C^∞ or smooth. An infinite degree of smoothness is not usually required, but it is assumed to avoid keeping track of the degree of

smoothness. The invertibility Condition 1 allows to uniquely recover the original state coordinate from the new one. In addition to applying a change of coordinates to a system's state, transformation of a system's output variable or time is also possible.

2 Vector Fields

A *vector field* on $U \subseteq \mathbb{R}^n$ is a C^∞ mapping from U to $\mathbb{R}^n$. It is customary to write vector fields using one of two notations. The first notation represents vector fields as column vectors. The basis used to represent the vector field is implied and this might lead to confusion.

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{pmatrix} f_1(\boldsymbol{x}) \\ f_2(\boldsymbol{x}) \\ \vdots \\ f_n(\boldsymbol{x}) \end{pmatrix} = \left(f_1(\boldsymbol{x}),\ f_2(\boldsymbol{x}), \dots, f_n(\boldsymbol{x})\right)^T$$

Alternately

$$\boldsymbol{f}(\boldsymbol{x}) = \sum_{i=1}^{n} f_i\,(\boldsymbol{x})\ \frac{\partial}{\partial x_i}$$

where $\frac{\partial}{\partial x_i}$ is the ith unit tangent vector in the x-coordinates. The latter notation provides more information in that the basis used to express the vector field is stated explicitly.

3 Differential Geometry Functions Used in the Text

Lie brackets

Lie brackets between two vector fields $\boldsymbol{f}$ and $\boldsymbol{g}$ is another vector field which is defined in local coordinates as

$$ad_{\boldsymbol{f}}^{1}\boldsymbol{g} = [\boldsymbol{f}, \boldsymbol{g}] = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\boldsymbol{f} - \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\boldsymbol{g}$$

Iterated Lie brackets are defined as

$$ad_{\boldsymbol{f}}^{k}\boldsymbol{g} = \left[\boldsymbol{f}, ad_{\boldsymbol{f}}^{k-1}\boldsymbol{g}\right], \qquad k \geq 1$$

$$ad_{\boldsymbol{f}}^{0}\boldsymbol{g} = \boldsymbol{g}$$

Lie Derivative

Lie derivative of function $h(\boldsymbol{x})$ along vector field $\boldsymbol{f}$ is the inner product of the gradient of $h(\boldsymbol{x})$ defined as

$$dh = \frac{\partial h}{\partial \boldsymbol{x}} = \left(\frac{\partial h}{\partial x_1}, \frac{\partial h}{\partial x_2}, \dots, \frac{\partial h}{\partial x_n}\right)$$

and vector field $\boldsymbol{f}$, therefore

$$L_{\boldsymbol{f}}h = \langle dh\,, \boldsymbol{f}\rangle = \sum_{i=1}^{n} f_i\,\frac{\partial h}{\partial x_i}$$

and the iterative form of it is explained as

$$L_{\boldsymbol{f}}^{k}h = L_{\boldsymbol{f}}L_{\boldsymbol{f}}^{k-1}h = \langle d\left(L_{\boldsymbol{f}}^{k-1}h\right), \boldsymbol{f}\rangle, \qquad k \geq 1$$

$$L_{\boldsymbol{f}}^{0}h = h$$

Sequential form of Lie Derivative is defined as

$$L_{\boldsymbol{g}}L_{\boldsymbol{f}}h = \langle d\left(L_{\boldsymbol{f}}h\right), \boldsymbol{g}\rangle$$

4 Distributions

A k-dimensional distribution Δ defined on $U \subseteq \mathbb{R}^n$ is a map which assigns to each $x \in U$, a k-dimensional subspace of $\mathbb{R}^n$ (or more precisely the tangent space $T_x\mathbb{R}^n$) such that for all $x_0 \in U$ there exists a neighbourhood $U_0 \subseteq U$ containing $x_0$and k smooth vector fields such that

1. $\{f_1(x), \dots, f_k(x)\}$ are linearly independent $\forall x \in U_0$
2. $\Delta(x) = span\{f_1(x), \dots, f_k(x)\}$ for all $x \in U_0$.

Note that in Condition 1, "linear independence" is the usual definition on $\mathbb{R}^n$ from linear algebra. In Condition 2 the span on the RHS involves real linear combinations of the constant vectors $f_1(x), \dots, f_k(x)$ in $\mathbb{R}^n$, that is, it is a subspace of $\mathbb{R}^n$. In many cases we use so-called generating vector fields $f_i(x), 1 \le i \le k$ to define a distribution, however, the $f_i(x)$ are only one choice of basis for the subspace $\Delta(x)$ and others can be chosen. For example if $\Delta(x) = span\{f_1(x), f_2(x)\}$ then we also have $\Delta(x) = span\{f_1(x) + f_2(x), f_1(x) - f_2(x)\}$.

The dimension of a distribution Δ, denoted as $\dim\big(\Delta(x)\big)$ is a function of x and is equal to the dimension of the subspace $\Delta(x)$. A distribution is said *nonsingular* on U if it has constant dimension on U. In nonlinear control, we often assume the relevant distributions are non-singular.

Given a k -dimensional distribution Δ defined on U and a vector field f on $U \subseteq \mathbb{R}^n$, we say f belongs to Δ if

$$f(x) \in \Delta(x), \forall x \in U$$

For more information please refer to the reference.

Appendix C Matlab Codes and Simulink Models

Typical Matlab Code to define the parameters and design the controller (pole placement) using the "place" function of Matlab.

```
clear all;
clc;
% Structure
    m=1000;
    c=500;
    k=5000000;

% Both systems
    mp=1025;
    bp=356.18e3;
    %ap=0.0808;
    ap=0.1095;
    %v4b=5.65e-10;
    v4b=4e-11;
    cp=3e-11;
    tm=0.004;
    xvmax=2.73e-3;
    %xvmax=2.73e+3;

% Linear
    kc=1.5e-11;
    kq=0.035;

% Nonlinear
    ps=207e5;
    n=2.617e-3;
    %xvmax=2.73e-3;
    kv=1;
    wv=1/0.004;

% PID
    kp=80;
    kd=0;
```

```
    ki=0.5;

 %State-space design for linear model with 3 states
    a1=[0 1 0; -k/(m+mp) -(bp+c)/(m+mp) ap/(m+mp); 0 -ap/v4b -
(kc+cp)/v4b];
    b1=[0; 0; kq/v4b];
    c1=[1 0 0];
    p=[-123,-26.8+29.9*1i,-26.8-29.9*1i];
    kcl1=place(a1,b1,p);
    nbar1=-inv(c1*((a1-b1*kcl1)\b1));

 %State-space design for linear model with 4
    a2=[0 1 0 0; -k/(m+mp) -(bp+c)/(m+mp) ap/(m+mp) 0; 0 -ap/v4b
-(kc+cp)/v4b kq/v4b; 0 0 0 -wv];
    b2=[0; 0; 0; kv*wv];
    c2=[1 0 0 0];
    p=[-242,-21.9+31.2*1i,-21.9-31.2*1i,-141];
    kcl2=place(a2,b2,p);
    nbar2=-inv(c2*((a2-b2*kcl2)\b2));

% Nonlinear state-space design for non-linear model with 4 states
    a3=[0 1 0 0; 0 0 1 0; 0 0 0 1; 0 0 0 0];
    b3=[0; 0; 0; 1];
    c3=[1 0 0 0];
    p=[-242,-21.9+31.2*1i,-21.9-31.2*1i,-141];
    kcl3=place(a3,b3,p);
    nbar3=-inv(c3*((a3-b3*kcl3)\b3));
```

Function definition for nonlinear behaviour of the servo-valve.

```
function flowB = fcn(xv,PL)
xv;PL;
%%% this fcn calculates the flow  considering the leakage
Cd=0.58;
rho=700; %kg/m^3, mass density
PS=207e5; %Pa , supply pressure
d=0.038; %m, spool diameter
% xvmax=2.73e-3; %m max spool opening
xvmax=2.73e-3;
%%%% leakage properties of the individual valve
%xvlap=6/100;
xvlap=0;
%Aeffnull=2.056e-6; %m^2
Aeffnull=0;
xvlapm=-1*xvlap;
%% default A1 and A4
A1=0;
A4=0;

%%% define the orifice areas
if xv <= xvlapm
    A1=0;
    A4=((pi*d*xvmax-2*Aeffnull)/(1-xvlap))*abs(xv)-(((pi*d*xvmax-
2*Aeffnull)/(1-xvlap))*xvlap-2*Aeffnull);
end
if xv >= xvlap
    A4=0;
    A1=((pi*d*xvmax-2*Aeffnull)/(1-xvlap))*abs(xv)-(((pi*d*xvmax-
2*Aeffnull)/(1-xvlap))*xvlap-2*Aeffnull);
end
if xv > xvlapm && xv <=0
    A1= -(Aeffnull/xvlap)*abs(xv)+Aeffnull;
    A4= (Aeffnull/xvlap)*abs(xv)+Aeffnull;
end
if xv > 0 && xv < xvlap
    A1= (Aeffnull/xvlap)*abs(xv)+Aeffnull;
    A4= -(Aeffnull/xvlap)*abs(xv)+Aeffnull;
end

%%% calculate the flow
flowB=(Cd*A1*((PS-PL)/rho)^0.5)-(Cd*A4*((PS+PL)/rho)^0.5);
```

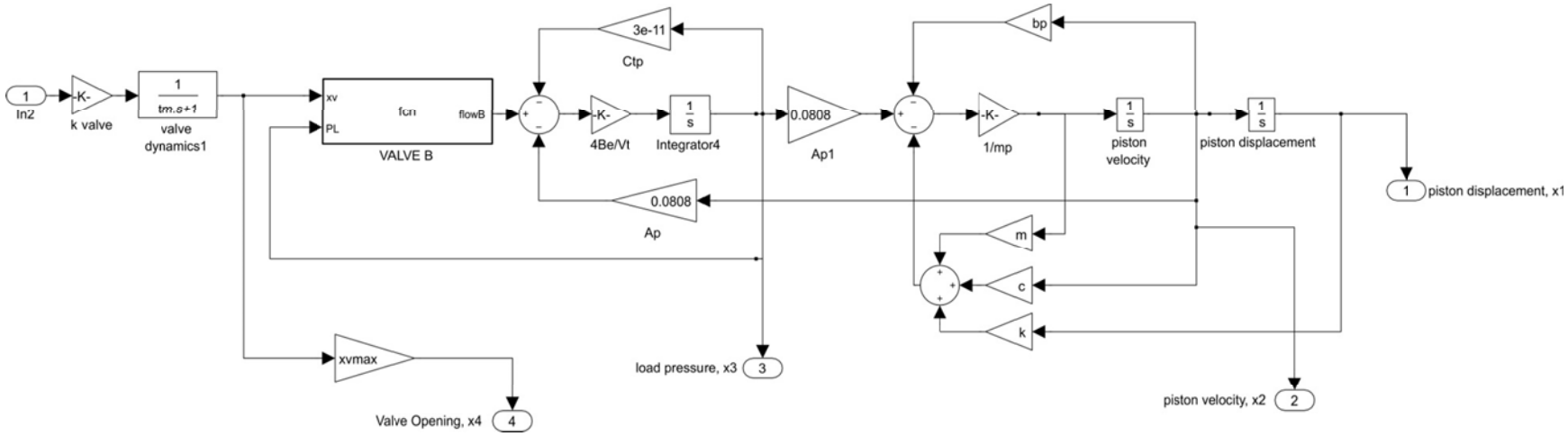


Figure C-1 Simulink model for nonlinear dynamics of a Servo-hydraulic

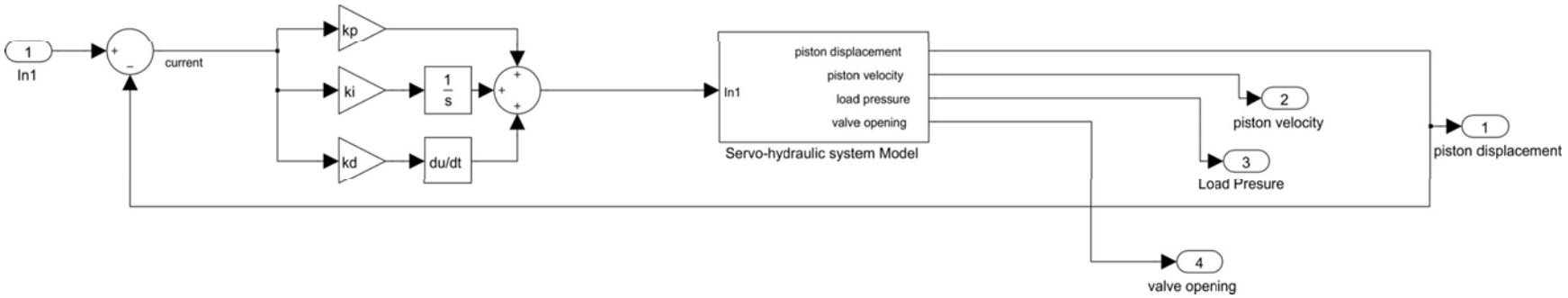


Figure C-2 Servo-hydraulic system with a PID controller

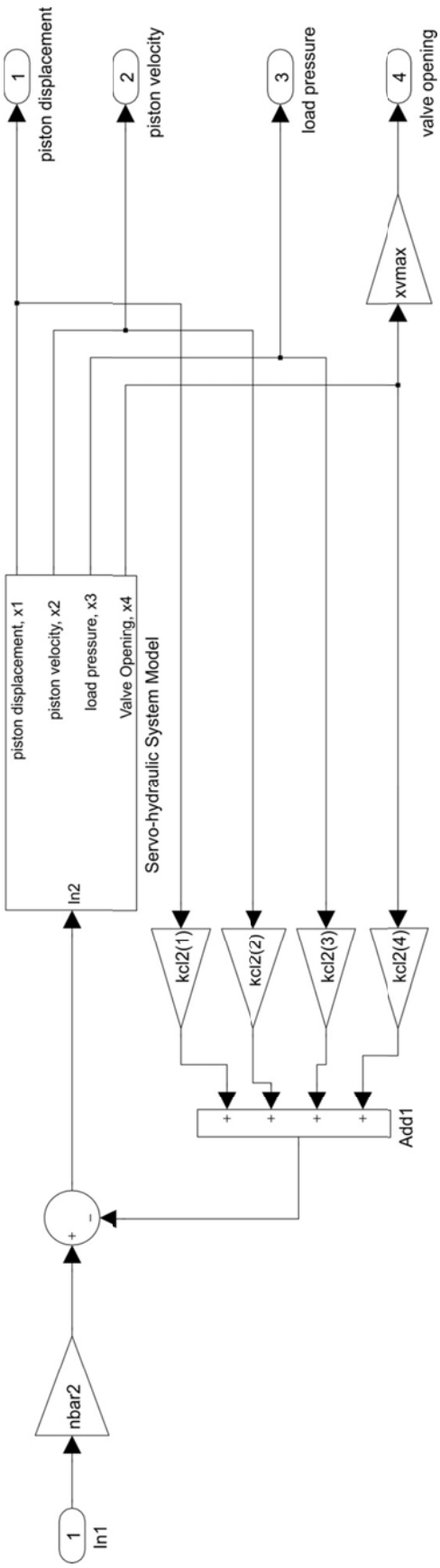


Figure C-3 Servo-hydraulic system with a linear state-space controller

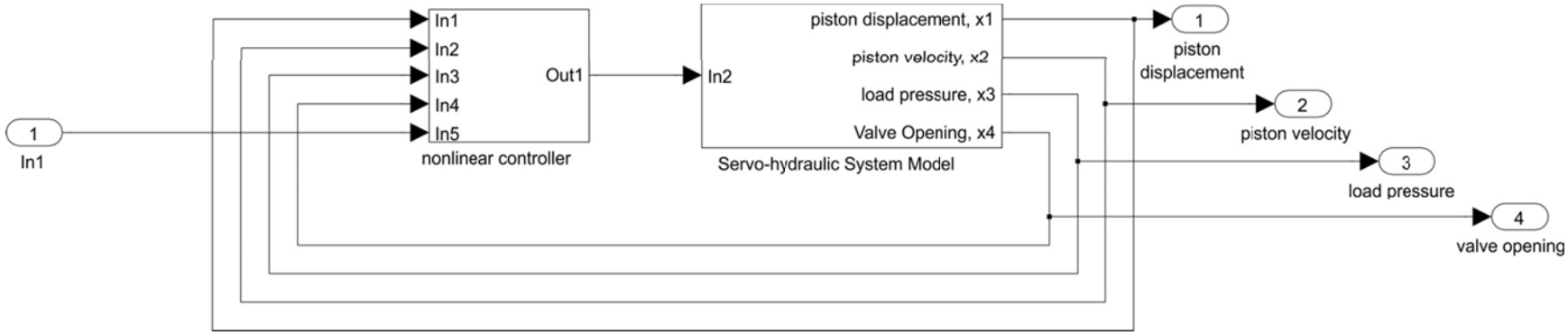


Figure C-4 Servo-hydraulic system with a nonlinear state-space controller

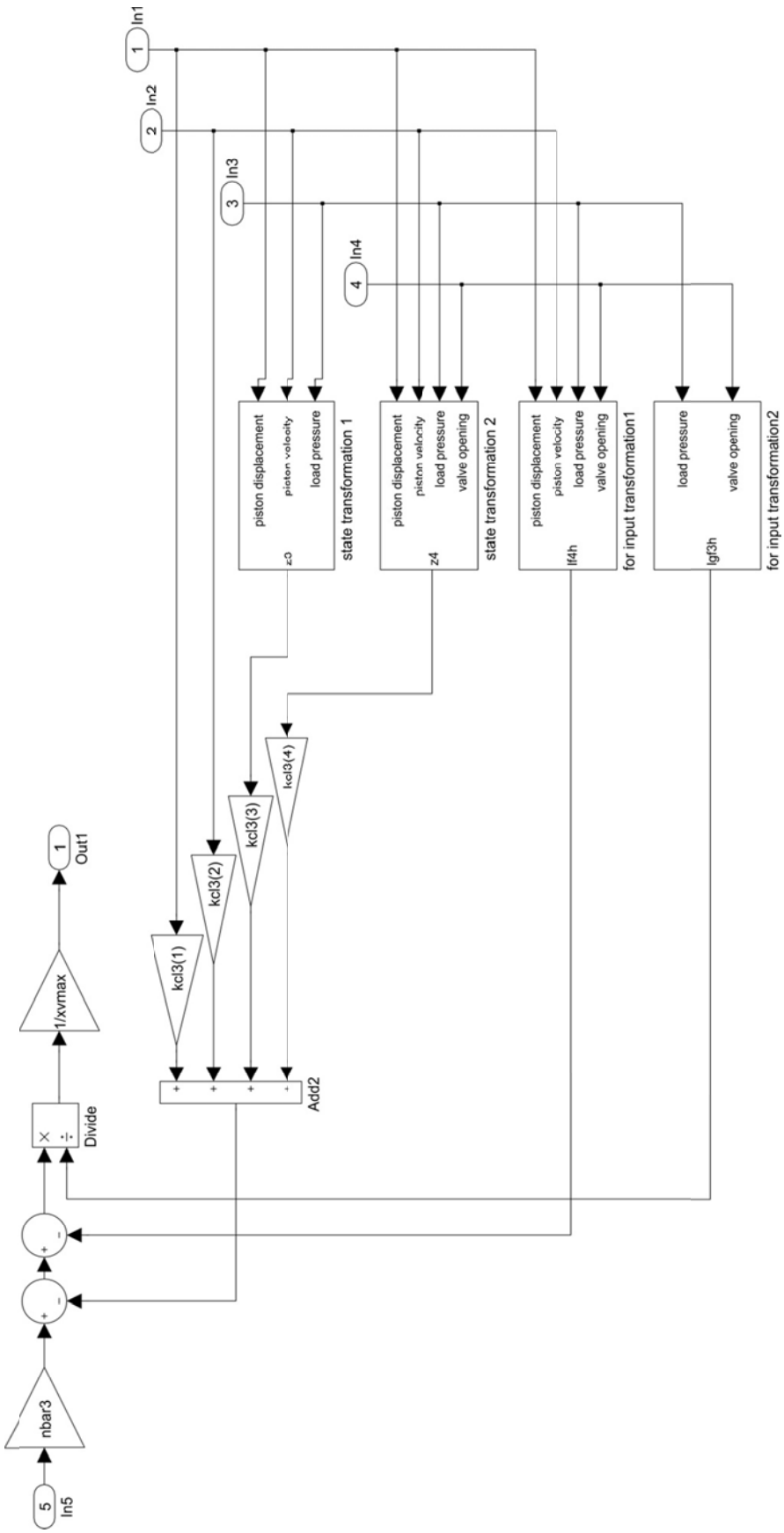


Figure C-5 nonlinear state-space controller

Appendix D Conference Paper

MODIFICATIONS OF INTEGRATION ALGORITHMS TO ACCOUNT FOR LOAD DISCONTINUITY IN PSEUDODYNAMIC TESTING

S. Hadi Moosavi[1], Oya Mercan[2]

ABSTRACT

When there is a sudden change in the loading (e.g., rectangular pulse), the discretized version of the load history will involve an artificial impulse, which manifests itself as an amplitude distortion in the structural response obtained by the numerical solution of the equation of motion. An approach to account for load discontinuity by modifying existing integration algorithms used in the solution of force equation of motion is introduced in this paper. Modified versions of four different algorithms, namely Central Difference, Newmark Explicit, α-method with a fixed number of iterations, and Rosenbrock-W integration algorithms are presented. The general approach in modifying an integration algorithm to account for load discontinuity is discussed and the improved accuracy of these modified algorithms is presented through numerical simulations.

Introduction

Pseudodynamic (PSD) test method is a displacement based experimental technique that can be used to determine the behavior of structural systems subjected to dynamic loading. In a PSD test, a direct step by step integration algorithm generates the command displacements by solving the force equation of motion. These displacements are imposed on the test structure by a servo-hydraulic system, and using the measured restoring force feedback from the deformed test structure, the integration algorithm computes the subsequent command displacements. For load-rate insensitive structures, PSD testing method can be applied in slow time (using an expanded time axis), or for structures that exhibit load-rate dependent vibration characteristics, it can be applied at fast rates (ideally in real-time). Both the slow-time and real-time PSD testing have been successfully applied for seismic loading (Mahin 1985, Nakashima 1999), but if the loading history has a sharp discontinuity as in the case of pulse loading (see Fig. 1-a), the numerical solution of the force equation of motion will have an amplitude distortion which may render the PSD test results inaccurate. This is due to the extra impulse (shaded area in Fig. 1-b) introduced during the discretization of the loading history.

To circumvent this problem, the use of step by step solution of the momentum equation of motion was suggested and the resulting improved accuracy was verified through numerical simulations (Chang, 2001, 2002, 2007a) and experiments (Chang, 1998). In this approach, the force equation of motion is replaced by its integral form, which is the momentum equation of motion. As a result of the time integration of the load that appears on the right hand side of the momentum equation, provided that the area under the load history is computed correctly, the discontinuity in the load history is eliminated. Although the success of the momentum approach

has been presented for Newmark explicit integration algorithm, replacing the force equilibrium equation with the momentum equation may not be a trivial task if one wishes to use an integration algorithm customized (e.g., unconditionally stable, implicit, real-time compatible) for particular testing needs. Other than the momentum approach, in an attempt to obtain an accurate solution in the presence of load discontinuity, Chang, (2007b) also proposed the use of a single time step immediately after the discontinuity that is much smaller than the discretization step size. This small step was recommended to be one-hundredth of the discretization step size or smaller. Especially within the context of real-time PSD testing, a variable time step is detrimental for it would result in inaccurate velocities as the displacement commands are typically imposed using a digital controller with a constant clock speed.

The study presented here introduces an approach, which is referred to as limit approach, to account for the load discontinuity by modifying a given integration algorithm that solves the force equation of motion in its final form. Depending on the way a particular integration algorithm is formulated, these modifications generally involve updated force and/or acceleration values at the time of discontinuity. In the paper, the general approach (i.e., the limit approach) that introduces modifications to a given integration algorithm is discussed and then implemented to derive modified versions of the Central difference, Newmark explicit, α-method with a fixed number of iterations, and Rosenbrock-W integration algorithms. Both α-method and Rosenbrock-W algorithms are suitable for real-time testing, where the former is an implicit scheme. For each of the four integration algorithms considered in this paper, a summary of the original formulation is provided together with its modified version. The improved accuracy of the modified algorithms is presented through numerical simulations.

The Limit Approach

The limit approach starts by defining an intermediate step j just after the load discontinuity between times i (where the discontinuity takes place) and $i+1$ (see Fig. 1-c). In order to be able to set it apart from the discretization time step size of Δt and thereby make the implementation of the limit approach for the modification of an algorithm easier to follow, the time step size associated with step j is identified as $\Delta t'$. It should be noted the load value p_j is the value of the load at the lower end of the discontinuity at step i. From the original formulation of a given integration algorithm, the information for step j (which may include the displacement (u), velocity ($\dot{u}$) and/ or acceleration ($\ddot{u}$), or an intermediate quantity defined by the particular integration algorithm to march forward) can be obtained using the information from step i (see Fig. 1-c) and considering p_j and $\Delta t'$. In order to obtain the information at the lower end of the discontinuity (i^*) and thereby account for discontinuity effects properly, next step involves taking the limit of the expressions that define step j information where $\Delta t'$goes to zero. On Fig. 1-d the information (i.e., the expressions for the load, displacement etc.) associated with $i^*$are the results of this limit process. In programming the resulting modified algorithm, a flag needs to be set in order to identify the time step when discontinuity (i.e., step i) takes place. When that happens, the numerical values of for p_{i^*}, u_{i^*} etc. need to be evaluated from the expressions obtained by the limit approach, and using these, the integration algorithm in its original form can march forward to compute information for step $i+1$.

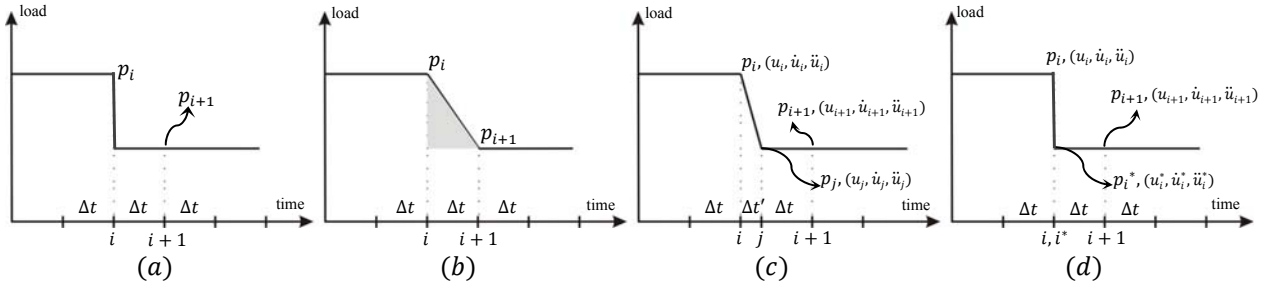


Fig. 1. (a) Load history with discontinuity, (b) Discretized load history, (c) Load history with step j, (d) Load history after performing the limit

Implementation of the Limit Approach

Central Difference Method

The discretized form of the equation of motion for a single-degree-of-freedom (SDOF) system at time step i is:

$$m\,\ddot{u}_i + c\,\dot{u}_i + k\,u_i = p_i \tag{1}$$

Central difference method uses a finite difference approximation for velocity and acceleration (Chopra, 2007). With a constant time step size of Δt, the velocity and acceleration at step i are expressed as:

$$\dot{u}_i = \frac{u_{i+1} - u_{i-1}}{2\Delta t} \tag{2}$$

$$\ddot{u}_i = \frac{1}{\Delta t}\left(\frac{u_{i+1} - u_i}{\Delta t} - \frac{u_i - u_{i-1}}{\Delta t}\right) = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta t^2} \tag{3}$$

Substituting the velocity and acceleration from Eqs. (2) and (3) in Eq. (1), and solving the expression for u_{i+1}:

$$u_{i+1} = \underbrace{\left[p_i - \left(\frac{m}{\Delta t^2} - \frac{c}{2\,\Delta t}\right)u_{i-1} - \left(k - \frac{2\,m}{\Delta t^2}\right)u_i\right]}_{\hat{p}_i} \Big/ \underbrace{\left[\frac{m}{\Delta t^2} + \frac{c}{2\,\Delta t}\right]}_{\hat{k}} \tag{4}$$

Implementation of Limit Approach to Obtain (i^*) Information

Considering Fig. 1-c and rewriting Eqs. (2) and (3) for step i, velocity and acceleration at that step are:

$$\dot{u}_i = \frac{u_j - u_{i-1}}{\Delta t + \Delta t'}\,, \qquad \ddot{u}_i = \frac{1}{\frac{1}{2}(\Delta t + \Delta t')}\left(\frac{u_j - u_i}{\Delta t'} - \frac{u_i - u_{i-1}}{\Delta t}\right) \tag{5), (6}$$

Also, using the same equations, velocity and acceleration at step j can be expressed as:

$$\dot{u}_j = \frac{u_{i+1} - u_i}{\Delta t + \Delta t'}\,, \qquad \ddot{u}_j = \frac{1}{\frac{1}{2}(\Delta t + \Delta t')}\left(\frac{u_{i+1} - u_j}{\Delta t} - \frac{u_j - u_i}{\Delta t'}\right) \tag{7), (8}$$

Eqs. (9) and (10) express the equation of motion at time steps i and j, respectively:

$$m\,\ddot{u}_i + c\,\dot{u}_i + k\,u_i = p_i \tag{9}$$

$$m\,\ddot{u}_j + c\,\dot{u}_j + k\,u_j = p_j \tag{10}$$

Once Eq. (9) is solved for u_j (after substituting for $\dot{u}_i$ and $\ddot{u}_i$ from Eqs. (5) and (6)), the result can be used to replace u_j in Eq.(10) which then can be solved for u_{i+1} (after substituting for $\dot{u}_j$ and $\ddot{u}_j$ from Eqs. (7) and (8)). Performing a limit where $\Delta t' \to 0$ gives an expression for u_{i+1} in terms of u_{i-1} and u_i:

$$u_{i+1} = \underbrace{\left[\underbrace{\frac{p_i + p_j}{2}}_{p_i^*} - \left(\frac{m}{\Delta t^2} - \frac{c}{2\,\Delta t}\right)u_{i-1} - \left(k - \frac{2\,m}{\Delta t^2}\right)u_i\right]}_{\hat{p}_i^*} \Big/ \underbrace{\left[\frac{m}{\Delta t^2} + \frac{c}{2\,\Delta t}\right]}_{\hat{k}} \tag{11}$$

Comparing Eq. (4) with Eq.(11), application of limit approach to Central difference method reveals that to account for the load discontinuity, only the load value for (i^*) needs to be redefined as the average value of the loads at each end of the discontinuity.

Explicit Newmark method

Newmark family integration methods (Newmark, 1959) can be customized by selecting two parameters (γ and β) which specify the variation of acceleration over a time step. These parameters also determine the accuracy and stability characteristics of the method. For $\gamma = 0.5$ $\beta = 0$ Newmark's method becomes explicit and conditionally stable:

$$u_{i+1} = u_i + \Delta t\,\dot{u}_i + 0.5\,\Delta t^2 \ddot{u}_i \tag{12}$$

$$\ddot{u}_{i+1} = \frac{p_{i+1} - k\,u_{i+1} - c[\dot{u}_i + 0.5\,\Delta t\,\ddot{u}_i]}{m + 0.5\,c\,\Delta t} \tag{13}$$

$$\dot{u}_{i+1} = \dot{u}_i + \Delta t[0.5\,\ddot{u}_i + 0.5\,\ddot{u}_{i+1}] \tag{14}$$

Implementation of Limit Approach to Obtain (i^*) Information

Defining the information for step j using Eqs. (12)-(14) as required by the limit approach:

$$u_j = u_i + \Delta t'\,\dot{u}_i + 0.5\,\Delta t'^2 \ddot{u}_i \tag{15}$$

$$\ddot{u}_j = \frac{p_j - k\,u_j - c[\dot{u}_i + 0.5\,\Delta t'\,\ddot{u}_i]}{m + 0.5\,c\,\Delta t'} \tag{16}$$

$$\dot{u}_j = \dot{u}_i + 0.5\,\Delta t'(\ddot{u}_i + \ddot{u}_j) \tag{17}$$

Performing a limit on the above expressions where $\Delta t' \to 0$, will yield

$$u_i{}^* = u_i,\ \dot{u}_i{}^* = \dot{u}_i \tag{18), (19}$$

$$\ddot{u}_i{}^* = \frac{p_i{}^* - k\,u_i - c\,\dot{u}_i}{m} \tag{20}$$

where $p_i{}^* = p_j$

From above equations, it turns out that, while the values for displacement and velocity remain the same, the acceleration value at the lower end of the discontinuity (i^*) needs to be updated. By stepping forward to step $i+1$ using the updated information from Eqs. (18)- (20) derived consistently with Newmark explicit method formulation, the load discontinuity will be taken care of properly.

Rosenbrock

Considering an SDOF system the general implementation of Rosenbrock integration method proposed by Lamarche (2009) for real-time pseudo-dynamic testing is described in Fig. 2.

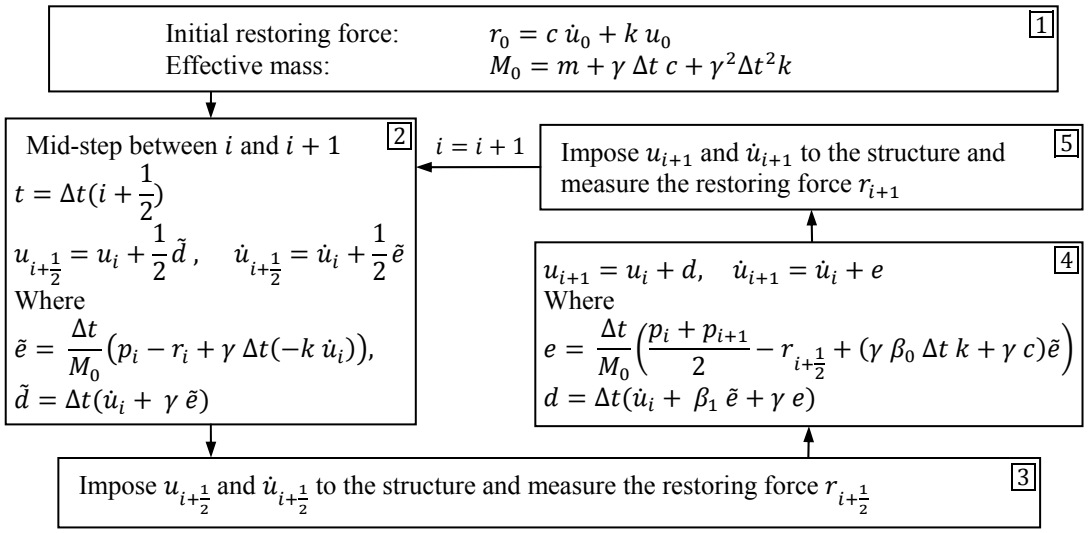


Fig. 2. Rosenbrock integration algorithm

Implementation of Limit Approach to Obtain (i^*) Information

Introducing step j and taking the limit for the Rosenbrock algorithm, it can be easily shown that u_j and $\dot{u}_j$ approach to u_i and $\dot{u}_i$, respectively (or equivalently $u_i{}^*$ and $\dot{u}_i{}^*$ are the same as u_i and $\dot{u}_i$, respectively). As a result, for Rosenbrock algorithm to handle the load

discontinuity properly, the information at step $i+1$ right after the discontinuity needs to be calculated using ${u_i}^*$, ${\dot{u}_i}^*$ and ${p_i}^*$ which are the same as u_i, $\dot{u}_i$, and p_j, respectively.

Alpha method

Alpha method (Shing *et al* 2002) is an implicit method used in real-time pseudo-dynamic testing which is based on Hilber α-method (Hilber *et al* 1977). As can be seen from Fig. 3, Alpha method starts by computing a predictor displacement in stage 1, which is followed by a fixed number of iteration substeps in stages 2 and 3; during the last iteration substep an equilibrium correction is performed. Through the equilibrium error correction (stages 4 and 5) the displacement and restoring force values are made available for the computation of the next step predictor displacement and, as a result, the actuator moves without interruption.

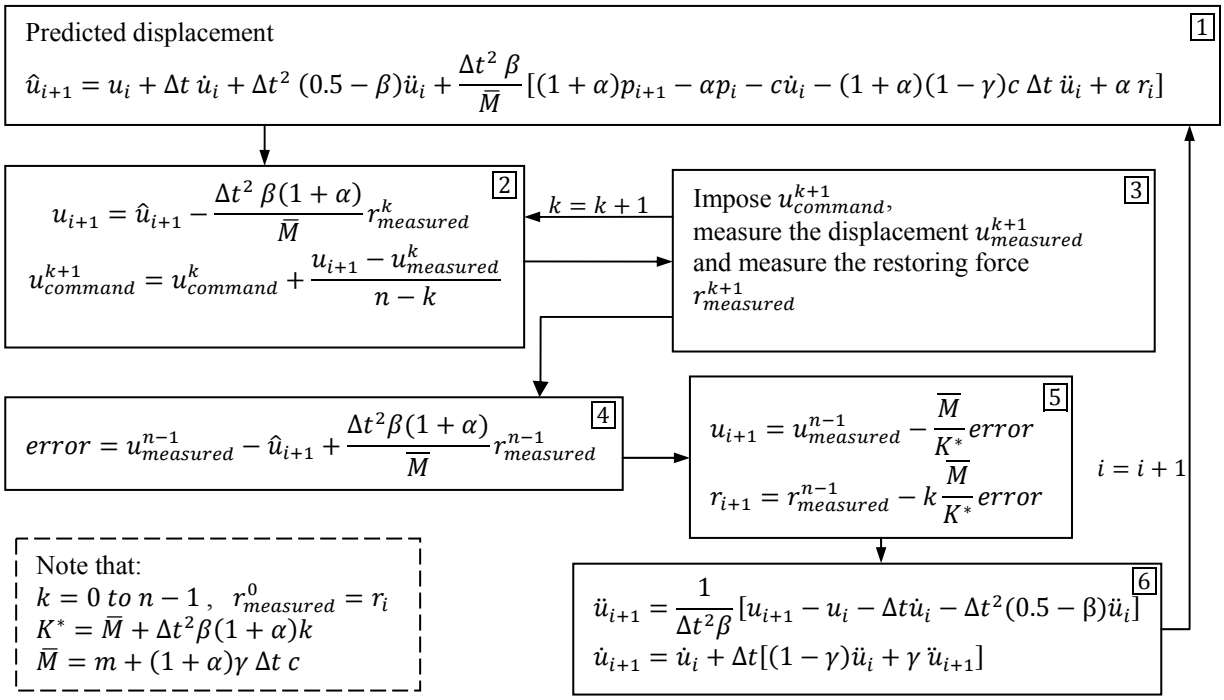


Fig. 3. Alpha method algorithm

Implementation of Limit Approach to Obtain (i^*) Information

Upon the application of the limit approach starting from stage 1 in Fig. 3, the predicted displacement at step j approaches to u_i after setting $\Delta t' \to 0$ (or equivalently ${\hat{u}_i}^* = u_i$) and the displacement term in stage 2 becomes constant (i.e., ${u_i}^* = u_i$). Considering that both parameters K^* and $\bar{M}$ approach to m application of the limit to stages 4 and 5 gives:

$$error = u_{measured}^{n-1} - u_i \tag{21}$$

$$\lim_{\Delta t' \to 0} u_j = u_i{}^* = u_{measured}^{n-1} - (u_{measured}^{n-1} - u_i) = u_i \tag{22}$$

$$\lim_{\Delta t' \to 0} r_j = r_i{}^* = r_{measured}^{n-1} - k(u_{measured}^{n-1} - u_i) = r_i \tag{23}$$

Eq. (23) is true for linear elastic systems. In stage 6 the expression for acceleration can be revised for step j as

$$\ddot{u}_j = \frac{1}{\Delta t'^2 \beta}\left[u_j - u_i - \Delta t' \dot{u}_i - \Delta t'^2 (0.5 - \beta)\ddot{u}_i\right] \tag{24}$$

Upon substituting the expression of u_j from stage 2 which has the expression for $\hat{u}_j$ embedded from stage 1, and as a result of the cancellations that take place, the zero over zero indeterminacy as $\Delta t'$ approaches to zero is eliminated; and the acceleration for i^* is obtained:

$$\ddot{u}_i{}^* = \frac{-p_i \alpha + (1+\alpha) p_i{}^* - c\, \dot{u}_i - r_i}{m} \tag{25}$$

where $p_i{}^* = p_j$

And the expression for velocity in stage 6 yields

$$\dot{u}_i{}^* = \dot{u}_i \tag{26}$$

The above application of the limit approach in modifying Alpha method reveals that, to compute the information at step $i+1$ right after the discontinuity, information from i^* needs to be used where the updated acceleration from Eq.(26) is used together with the value of the load at the lower end of the discontinuity.

Numerical Simulations

In order to verify the success of the proposed modifications in handling the load discontinuity, numerical simulation results for each integration algorithm are presented here. An undamped linear SDOF system with $m = 0.2533\ kip.sec^2/in$, and $k = 10 kips/in$ (i.e.,undamped natural period $T_n = 1\ sec.$) subjected to the two loading cases shown in Fig.4 is considered. Fig. 4 (a) is a step pulse with an amplitude of 10 *kips* and duration of 0.1 *sec*.; whereas in Fig. 4 (b) the load value changes from +10 to -10 *kips* at the discontinuity and then increases to zero linearly over a duration of 0.1 *sec*.

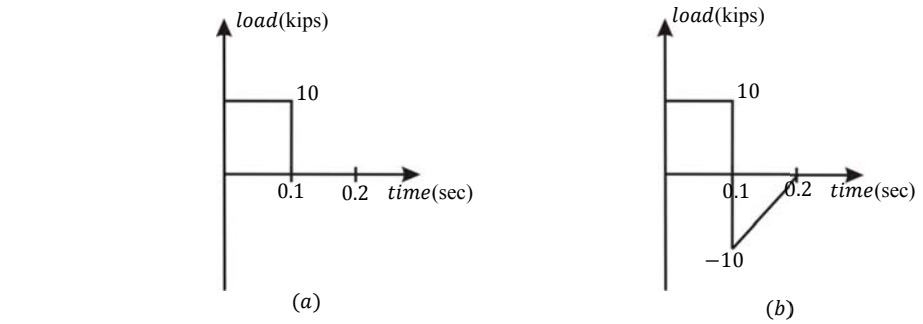


Fig. 4. Load cases with discontinuity

When discretizing the loading history for subsequent use in the step-by-step solution of the equation of motion, selection of a very small time step will minimize the extra impulse and in turn the amplitude distortion introduced in the response. To be able to check the ability of the modified algorithms in handling large extra impulses, the time step size for the numerical simulations was selected as 0.1 *sec.*; which also made the results of this study comparable with that of Chang (2001), where it was shown that time step can be selected as large as the impulse duration when the momentum equation is used. It needs to be pointed out that for Central difference and Newmark explicit algorithms $\frac{\Delta t}{T_n}$=0.1 sets the limit for accuracy, beyond which the amplitude decay and period elongation introduced by the algorithm may be significant (Chopra, 2007).

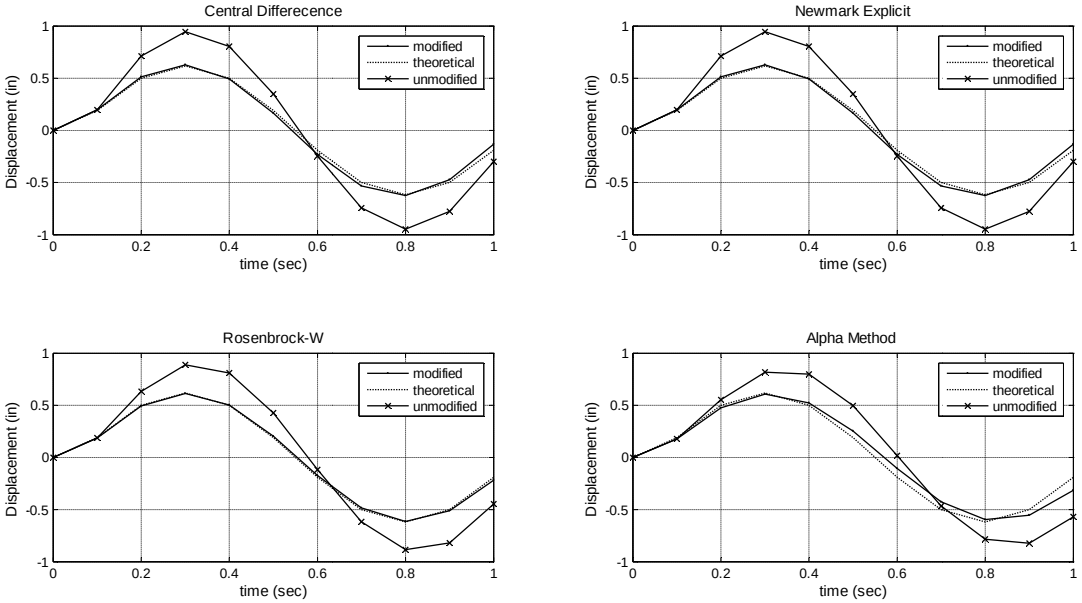


Fig. 5. Simulation results for the loading case in Fig. 4(a)

Fig. 5 compares the theoretical response with the results obtained from the original (unmodified) and modified versions of the integration algorithms considered in this study. Both versions of each algorithm were programmed in MATLAB. In the modified version, a flag was set to identify the discontinuity, and the modified expressions for load and/or acceleration were introduced as derived here. As can be seen from Fig. 5, there is a significant difference between the unmodified and theoretical responses; whereas the modified results are in good agreement with the theoretical response. Similarly, for Fig. 6 which considers nonzero load values after the

discontinuity, the proposed modifications provide considerable improvements in the accuracy of the numerical results. The unmodified responses from each algorithm between Fig. 5 and 6 are the same as expected, since with a time step size of 0.1 *sec.*, the discretized versions of the load cases in Fig. 4 (a) and (b) are the same (see Fig. 1 (b)). Depending on the accuracy characteristics of the particular integration algorithm, the agreement between the theoretical response and the modified numerical solution can be improved by using a smaller time step.

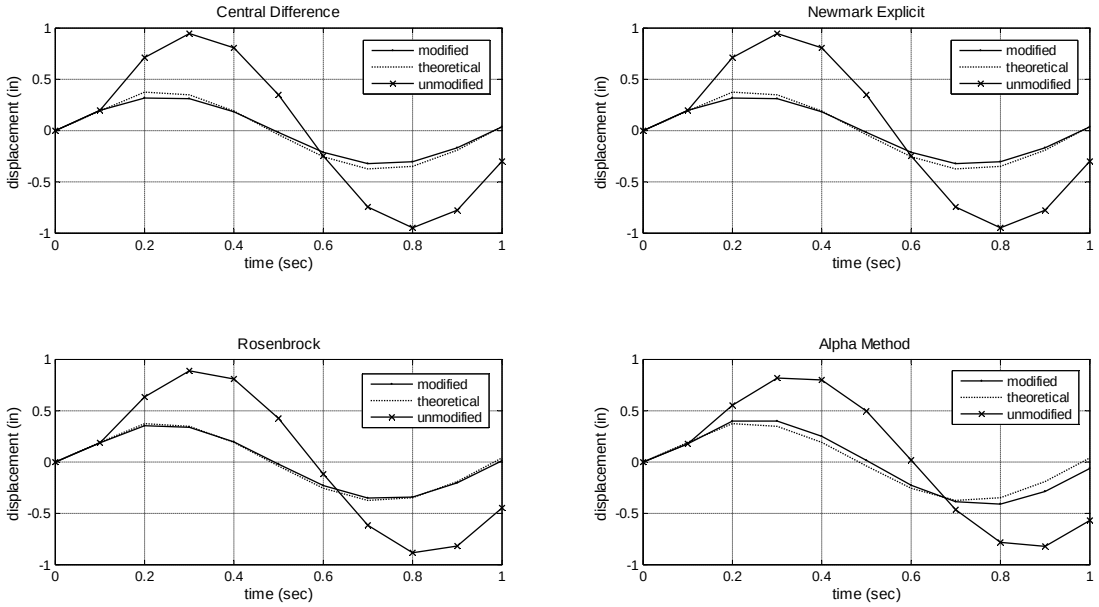


Fig. 6. Simulation results for the loading case in Fig. 4(b).

Conclusion

Pseudodynamic testing method has been implemented successfully both in slow and real-time for seismic loading of structures. However, when a sharp discontinuity exists in the loading history as in the case of pulse loading, the discretized version of the load will have an extra distortion which manifests itself as an amplitude distortion in the numerical response and may render the pseudodynamic test results inaccurate. Other than using very small time steps in discretizing the load, previous studies proposed the use of numerical solution of the momentum equation of motion that replaces the force equation of motion. Although the success of the momentum approach has been presented using Newmark explicit integration algorithm, replacing the force equilibrium with the momentum equation may not be a simple task if one wishes to use an integration algorithm customized for particular testing needs. The study presented here introduces a limit approach that considers the force equation of motion and modifies the integration algorithms in their final form to account for the load discontinuity. The general modification approach and its implementation to four integration algorithms are provided together with numerical simulation results that show the improved accuracy of the modified algorithms.

References

Chang, S. Y., Tsai, K. C., & Chen, K. C. (1998). Improved time integration for pseudodynamic tests. *Earthquake Engineering & Structural Dynamics*, 27 (7), 711-730.

Chang, S. Y. (2001). Analytical study of the superiority of the momentum equations of motion for impulsive loads. *Computers & Structures*, 79 (15), 1377-1394.

Chang, S. Y. (2002). An improved on-line dynamic testing method. *Engineering Structures* , 24 (5), 587-596.

Chang, S. Y. (2007 a). A technique for overcoming load discontinuity in using Newmark method. *Journal of Sound and Vibration* , 304, 556-569.

Chang, S. Y. (2007 b). Approach for overcoming numerical inaccuracy caused by load discontinuity. *Journal of Engineering Mechanics*, 133 (5), 555-565.

Chopra, A. K. (2007). *Dynamics of Structures*. New Delhi: Prentice-hall Of India Pvt Ltd.

Hilber, H. M., Hughes, T. J., & Taylor, R. L. (1977). Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 5, 283-292.

Lamarche, C. P., Bonelli, A., Bursi, O. S., & Tremblay, R. (2009). A Rosenbrock-W method for real-time dynamic substructuring and pseudo-dynamic testing. *Earthquake Engineering & Structural Dynamics*, 38, 1071–1092.

Mahin, S. A., & Shing, P.-s. B. (1985). Pseudodynamic method for seismic testing. *Journal of Structural Engineering*, 111 (7), 1482-1503.

MATLAB is a registered trademark of The MathWorks, Inc. http://www.mathworks.com/

Nakashima, M., & Masaoka, N. (1999). Real-time on-line test for MDOF systems. *Earthquake Engineering and Structural Dynamics*, 28 (4), 393-420.

Newmark, N. M. (1959). A method of compuation for structural dynamics. *Journal of Engineering Mechanics Division*, 67-94.

Shing, P. B., Spacone, E., & Stauffer, E. (2002). Conceptual design of fast hybrid test system at the University of Colorado. 7th U.S. National Conference on Earthquake Engineering, (2389-2398). Boston, MA (USA).

References

Ahmadizadeh, M. 2007. Real-time seismic hybrid simulation procedures for reliable structural performance testing. Doctor of philosophy, State University of New York at Buffalo, United States, New York.

Ayalew, B. and Jablokow, K. 2007. Partial feedback linearising force-tracking control: Implementation and testing in electrohydraulic actuation, Control Theory & Applications, IET, **1**: 689-698.

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**: 1028-1037.

Franklin, G.F., Powell, J.D., and Emami-Naeini, A. 2010. Feedback control of dynamic systems. Pearson, Upper Saddle River N.J.

How, J. 2007. Course materials for 16.31 feedback control systems. MIT OpenCourseWare, Massachusetts Institute of Technology.

Kwon, J., Kim, T., Jang, J.S., and Lee, I. 2007. Feedback linearization control of a hydraulic servo system. *In* SICE-ICASE, 2006. International Joint Conference, pp. 455-460.

Lim, T. 2002. Pole placement control of an electrohydraulic servo motor. *In* Power Electronics and Drive Systems, 1997. Proceedings., 1997 International Conference on, Vol. 1, pp. 350-356.

Lynch, A. 2009. Differential geometry - basic, notes for nonlinear control design. University of Alberta.

Mahin, S.A. and Williams, M.E. 1981. Computer controlled seismic performance testing. *In* Second ASCE-EMD Specialty Conference on Dynamic Response of Structures, Atlanta, GA.

Mahin, S., A. and Shing, P., B. 1985. Pseudodynamic method for seismic testing, Journal of Structural Engineering, **111**: 1482-1503.

Marino, R. and Tomei, P. 1995. Nonlinear control design: Geometric, adaptive, and robust. Prentice Hall, .

Mercan, O. and Ricles, J.M. 2008. Stability analysis for real-time pseudodynamic and hybrid pseudodynamic testing with multiple sources of delay, Earthquake Engineering & Structural Dynamics, **37**: 1269-1293.

Mercan, O., Zhang, X.P., and Ricles, J.M. 2006. State-space control design for real-time pseudodynamic testing applications. *In* Proceedings of the 8th U.S. National Conference on Earthquake Engineering, San Francisco, California, USA.

Mercan, O. 2007. Analytical and experimental studies on large scale, real-time pseudodynamic testing. Ph.D., Lehigh University, United States -- Pennsylvania.

Merritt, H.E. 1967. Hydraulic control systems. Wiley, New York.

Mintsa, H.A., Kenne, J.P., and Venugopal, R. 2009. Adaptive control of an electrohydraulic position servo system. *In* AFRICON, 2009. AFRICON'09., pp. 1-6.

Moog, I.C.D. 2010. G761 series servo-valves, product catalog. East Aurora, NY.

Okada, T., Seki, M., and Park, Y.J. 1980. A simulation of earthquake response of reinforced concrete building frames to bi-directional ground motion by IIS computer-actuator on-line system. *In* Seventh World Conference on Earthquake Engineering, Istanbul, Turkey.

Seo, J., Venugopal, R., and Kenné, J.P. 2007. Feedback linearization based control of a rotational hydraulic drive, Control Engineering Practice, **15**: 1495-1507.

Shing, P.B. and and Mahin, S.A. 1983. Experimental error propagation in pseudo-dynamic testing. UCB/EERC-83/12, Earthquake Engineering Research Center, University of California, Berkeley, California.

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. 8, Earthquake Resistant Structure Research Center, Institute of Industrial Science, University of Tokyo, Tokyo, Japan.

Thayer, W.J. 1958 (revised in 1965). Transfer functions for MOOG servo-valves, technical bulletin 103. MOOG Inc., Control Division, East Aurora, NY.

Yanada, H. and Furuta, K. 2007. Adaptive control of an electrohydraulic servo system utilizing online estimate of its natural frequency, Mechatronics, **17**: 337-343.

Zhang, X., Ricles, J.M., Mercan, O., and Chen, C. 2005. Servo-hydraulic system identification for the NEES real-time multi-directional earthquake simulation facilty. 05-14, ATLSS, Lehigh University.

Zhao, J. 2003. Development of EFT for nonlinear SDOF systems. Ph.D., University of Minnesota, United States -- Minnesota.