PES UNIVERSITY

(Established under Karanataka Act No. 16 of 2013)
100 Feet Ring Road, Bengaluru, Karnataka, India. Pin – 560 085

Report On

Mutual Adaptation

by

Vittal Srinivasan (PES1201700310)

Gayatri Sreenivasan (PES1201700434)

S R Swethaashri (PES1201700781)

August 2020 – May 2021

Under the guidance of

Prof. Rajini M

Assistant Professor
Department of Electronics and Communication Engineering
PES University
Bengaluru – 560 085

FACULTY OF ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

PROGRAM: BACHELOR OF TECHNOLOGY

CERTIFICATE

This is to certify that the Report entitled

“Mutual Adaptation”

is a bonafide work carried out by

Vittal Srinivasan (PES1201700310)

Gayatri Sreenivasan (PES1201700434)

S R Swethaashri (PES1201700781)

in partial fulfilment for the completion of 8^{th} Semester course work in the Program of Study Bachelor of Technology in Electronics and Communication Engineering, under the rules and regulations of PES University, Bengaluru during the period January – May 2021. It is certified that all corrections and suggestions indicated for internal assessment have been incorporated in the report. The report has been approved as it satisfies the 8^{th} Semester academic requirements with respect to Capstone Project Work.

Signature with Date and Seal
Prof. Rajini M
Internal Guide

Signature with Date and Seal
Dr. Anuradha M
Chairperson

Signature with Date and Seal
Dr. B. K. Keshavan
Dean – Faculty of Engineering and Technology

Name and Signature of Examiners:

1.

2.

DECLARATION

We, **Vittal Srinivasan**, **Gayatri Sreenivasan** and **S R Swethaashri** hereby declare that the report entitled **"Mutual Adaptation"** is an original piece of work carried out by us under the guidance of **Prof. Rajini M**, Assistant Professor, Department of Electronics and Communication Engineering, and is being submitted in partial fulfilment of the requirements for completion of 8^{th} Semester course work in the Program of Study Bachelor of Technology in Electronics and Communication Engineering.

PLACE: BENGALURU

DATE: 25^{th} June 2021

NAME AND SIGNATURE OF THE CANDIDATES:

1. Vittal Srinivasan

Vittal Srinivasan

2. Gayatri Sreenivasan

Gayatri Sreenivasan

3. S R Swethaashri

S. R. Swethaashri

ACKNOWLEDGEMENT

Firstly, we would like to thank our mentor, Professor Rajini M, for her constant support and guidance towards the completion of this project. With her discussions and invaluable help, we have been able to successfully undertake this project. We would also like to thank Dr Koshy George for his help on the subject matter. His course on Adaptive Systems inspired us to take on this project. We would also like to express our gratitude to Dr Anuradha M (Chairperson, Department of Electronics and Communication Engineering, PES University), Dr M. R. Doreswamy (Chancellor, PES University), Prof. Jawahar Doreswamy (Pro-Chancellor, PES University) and Dr J. Suryaprasad (Vice-Chancellor, PES University) for giving us this opportunity to widen the horizons of our knowledge.

ABSTRACT

The problem of mutual adaptation has far reaching consequences in various domains including Psychology, Biology and Robotics. Therefore, it is a problem of great interest and a field with great scope. Mutual adaptation tackles the question of whether two or more systems can adaptively stabilize each other. While this question may seem innocuous, several intricacies and assumptions have to be considered while ensuring the stability of the overall system. To appreciate the complexity of this multi-faceted problem, the fundamentals of simple adaptive systems must be studied. To facilitate this understanding, the note begins with the fundamentals of adaptive control and stability theory. Expanding on this, the note then delves into the key concepts of adaptive control - the identification and control problem. Various realisations of adaptive controllers and observers are investigated next. This is followed by a deep investigation into mutual adaptation and it's intricacies.

Initially the concept of mutual adaptation for two first order systems is explored. This is done through two approaches - the mathematical approach is done via stability analysis and through extensive simulation studies. The next section extends these approaches for subsystems of higher order. The theoretical observations are then applied to the problem of behavioural synchrony of simple physical systems of various types. This note culminates in an investigation of scenarios where mutual adaptation fails and methods to solve these redundancies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The concept of adaptation was introduced in 1957 into control systems by Drenick and Shabender to represent systems that monitor performance and adjust parameters in order to improve performance. This led to an explosion of ideas. However, it is not until the early 1980s that the stability of such adaptive systems was guaranteed [1]. An important ingredient in the stability analysis is the presence of stable reference models. In contrast, mutual adaptation deals with adaptive control of unknown systems in the absence of such reference models. Not surprisingly, there is very little work on mutual adaptation due to the lack of tractability and the high dimensionality of resulting nonlinear systems. More recently, the question of whether two or more plants can adaptively stabilise each other has been posed in [2]. In contrast to the 'learning with a teacher' paradigm from learning theory, the mutual adaptation problem follows the peer-to-peer learning paradigm in which two agents interact with each other and share information until they converge to appropriate behaviour.

Adaptive control is defined as the control of partially known systems. This approach of control is more accurate for real world processes as, in such scenarios, the designers seldom know all the parameters accurately. At the core of adaptive systems is the dual problem of identification and control. First, the controller should find the states of the system based on the information collected which is known as identification. Second, based on the knowledge acquired, the necessary actions for successful control must be determined. This is the control problem. Observers can also use the principles of adaptive systems. An observer which simultaneously adapts the parameters and state variables of a dynamical system is called an adaptive observe. The design of an adaptive observer includes the choice of a suitable parameterization of the plant as well as stable adaptive laws for the adjustment of the parameters.

Mutual Adaptation is a niche subsection of the domain of adaptive systems. In order to get a holistic view of the mutual adaptation problem, one must be well versed with the concepts of simple adaptive systems [1]. In mutual adaptation, two or more subsystems interact with one another and update the parameters of the respective subsystems accordingly. The important questions that remain is 'Can two subsystems reach to common consensus with adaptation?' [2]. This problem involves several complexities as neither of the subsystem have a stable reference model to emulate. The stability of the overall system essentially depends on the assumptions made regarding the individual subsystems and how they interact. The analysis becomes complex in the case of systems with higher order. The problem of mutual adaptation can lead to the overall system's instability even with the application of standard adaptive laws for higher order systems [3].

1.1 Formulation of the Mutual Adaptation Problem

Consider a plant with unknown parameters. The parameters of the controller are adapted such that the behaviour of the controlled plant emulates the behaviour of the reference model. In the standard adaptive systems, we have dealt with cases of adaptation where there was a reference model, using which the plant adapts its parameters. But what if we need to deal with a case where there is no reference model?

In contrast to 'learning with a teacher' paradigm from learning theory, the mutual adaptation problem follows the 'peer to peer learning' paradigm in which two students interact with each other and share information until they converge to appropriate answers.

In the mutual adaptation problem, the subsystems interact with each other such that the parameters of the respective subsystems update accordingly in the absence of reference model. The error between the outputs of the subsystems converge to zero as the plants reach a consensus on their behaviour to achieve the required output.

Mutual adaptation deals with the problem of two or more plants trying to adaptively stabilise each other.
If both the plants are unstable:

1. Neither system has a stable reference model to emulate.
2. Each depends upon the other to stabilize itself.

Therefore the answer is not simple as the problem depends on:

1. The assumptions made regarding adaptive subsystems.
2. The manner in which the subsystems interact with each other.

The analysis becomes complex in the case of systems with higher order. The problem of mutual adaptation can lead to the overall system's instability even with the application of standard adaptive laws for higher order systems [2, 3].

1.2 Related Work

The problem of mutual adaptation has far reaching implications in diverse fields like Psychology, Biology, Economics and Robotics, owing to the lack of available information it takes into consideration.

1.2.1 Mutual Adaptation in Science and Business

Mutual adaptation is seen in nature in:

1. Pheromone mediated navigation in ant colonies
2. School of fish synchronising their swimming
3. Human to human interaction

Mutual adaptation occurs in abundance in nature- within biochemical processes [4], physical phenomena and biological organisms [5]. It has also been used to analyse business relationships between various entities of interest [6, 7]. In psychology, interpersonal relationships have been studied between mutually adapting subjects in diverse environments [8–10].

Figure 1.1: Mutual Adaptation in Nature

1.2.2 Mutual Adaptation in Engineering

In engineering, there are several problems where there is no reference model for the systems to adapt. The systems update from other sub systems in the environment to reach desired outputs. Consider the self-adaptation of multi-agent systems in some dynamic environment based on exchanges in experiences [11]. An action matrix is formed based on immediate and retrospective experiences of the agents. This action matrix is used so that the agents can perform decision-making and adaptation. Another example dealing with the absence of a reference model is in the mutual adaptation of Neuro-Fuzzy Systems for applications like human posture recognition [12]. The posture and gesture recognition can be done using kinect sensor. Then the features extracted are used in the classification process. The Neuro Fuzzy System (NFS) is used in the classification of pose. Here, mutual adaptation is used between the information processing components of the NFS. In Physical Human-Robot Interaction (PHRI) [13], mutual learning and mutual adaptation are used to perform a couple of learning scenarios involving humans in the loop. These scenarios have been inspired by the behaviour of a human parent while interacting with their child. In most machine learning cases to date, they do not tackle the problem of how learning can be achieved for tightly-coupled physical interactions between a robotic learning component and a human counterpart. Here, the human and the robot learn to interact smoothly via mutual learning and adaptation. In all the above problems, the systems update from other sub systems in the environment to reach desired outputs.

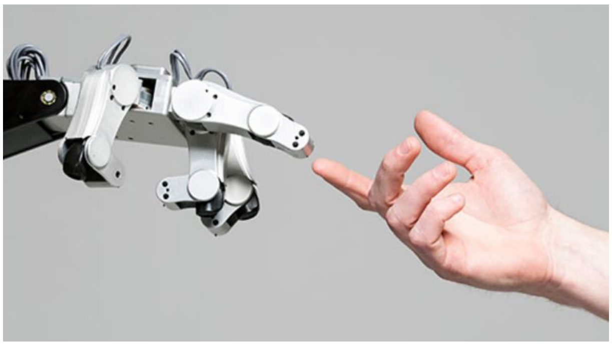


Figure 1.2: Mutual Adaptation in Engineering

In robotics, mutual adaptation and learning are extensively used to emulate the advanced behaviour of humans [13–16]. Human-robot interaction has been an area of considerable interest over the past decade. The human subject interacts with the robotic component with increasing ease and accuracy to carry out the desired task [17, 18]. In swarm robotics, the interactions naturally existing in insect colonies are mirrored in the robotic subsystems by exploiting the idea of mutual adaptation [19].

In the domain of adaptive control, a comprehensive study of mutual adaptation between two first order-systems and second-order systems are presented in [2]. Consequently this study is extended to higher order systems by Narendra *et al.* [3].

Figure 1.3: Mutual Adaptation in Swarm Robotics

1.3 Novelties Proposed

In chapter 7, behavioural synchronisation is dealt with from the viewpoint of mutual adaptation. The parameters of two systems are adapted constantly so that a consensus in the behaviour of two systems eventually emerges. An important difference between mutual adaptation and stable adaptive systems is the absence of a reference model. Stable adaptive laws ensure that the states of two systems converge to the same value asymptotically. Albeit the convergence of the parameters to values ensuring asymptotic stability is not guaranteed by the theory, in practice it is observed that they do so. It may be argued that this is due to the fact that a common consensus has been reached. We illustrate in the achievement of behavioural synchrony with four dynamical systems. We consider first-order, second-order and third-order systems. These models are inspired by three physical systems, namely an RC filter, a simple pendulum, and a DC motor.

1.4 Organization of the Report

This report is broadly classified into two sub-sections - the fundamentals of adaptive systems and the mutual adaptation problem. In order to appreciate the concept of mutual adaption, the key concepts of adaptive control must be understood in-depth. Keeping this in mind, the report is structured in the following manner: Chapters 1-4 presents a detailed explanation of the domain of adaptive systems by beginning with a deep delve into the fundamentals of adaptive control. The identification and control problem for simple adaptive systems is examined in chapter 2 and simulations for the same are presented. The successive chapter discusses the necessity for adaptive observers and controllers. Our exploration of mutual adaptation begins in the fifth chapter. In this chapter, the mutual adaptation problem is presented and the mathematical basis for mutual adaptation is then derived for first as well as higher order cases. Apart from this, in-depth simulation studies have also been performed in order to investigate the trends that are present in the first order case.In chapter 6, The principles of mutual adaptation is then applied to physical systems of various orders . In these applications, mutual adaptation is used to perform behavioural synchrony of the two physical systems with different initial conditions. Here, the two systems are assumed to have parameters that are tunable. This report concludes with an inspection into the limitations of mutual adaptation and the proposal of novel ideas to solve these deficiencies.

1.5 Summary

In this chapter, we discuss the history of adaptive systems and introduce the concept of mutual adaptation. We formulate the mutual adaptation problem and dived into the related works of mutual adaptation in the fields of psychology, biology, economics and engineering to get an idea of how to implement mutual adaptation. The following section introduces the issue of behavior synchrony as a mutual adaptation problem and end the chapter with workflow of the report.

Chapter 2

Fundamentals of Adaptive Systems

In this chapter, we explore the fundamentals of adaptive systems, starting with the terminology for adaptive systems in subsection 2.1. In section 2.2 we delve into the stability analysis. We define the different kinds of stability and further delve into their types in further subsections.

2.1 Terminology

A system is a collection of entities united by either reciprocity or interrelationship. While certain aspects of the system change with time, the system has a dynamic nature. Inputs are external entities impact the nature of the systems . Inputs arise from outside the system and influence it. These influences arising from outside the system are not directly impacted by what occurs in the system. The values that are affected by the inputs are called the outputs of the system. Mathematically, a dynamic system can is a system that receives the information $u(t)$ at each interval of time t. Here t is associated with a time set T and produces the output $y(t)$. The input elements are assumed to be a part of a set U. Similarly, the features of the output are assumed to belong to the set Y. For most cases, the output $y(t)$ is dependent on the input $u(t)$ as well as history of the inputs and the entities under scrutiny.

The notion of the state was instigated to forecast the future actions of the system based on the entities entered into the network starting at an initial time t_0. The vector u is the input to the system which is dynamic. It possesses components that are under the command of the designer as well as constituents that are not. The entities that are under the influence of the designer are called the control inputs. The vector $x(t)$ is known as the state of the system at a particular time t. The elements of this vector $x_i(t)(i = 1, 2, ..., n)$ are known as the state-variables. The state x(t) at time t is dependent on the state $x(t_0)$ at any time $t_0 < t$ and the input established over the period $[t_0, t)$. The output $y(t)$ is controlled by the time t as well as the state of the sub-system $x(t)$ at the particular time t.

The vector θ is defined as the parameter vector. It comprises of parameters that are linked with the process to be regulated (system parameters). It also involves those entities that can be chosen based on the designer's judgment (control parameters). Both the aforementioned parameters can be either fixed with respect to time or time-varying in nature. To regulate a system's behavior, we need to retain the pertinent outputs y_i within predefined limits. Control is generally impacted by either modifying the control inputs for a specified structure given by f and the values of the parameter θ or by a regulated the control parameters for specified inputs u.

As the mathematical model of a prearranged part of the system, known as the plant, is provided, the engineer can construct direct feed forward and direct feedback controllers so as to get the outputs of the plant to behave in the desired fashion. While similar performing controllers can be constructed for nonlinear systems, no such notions of control design exists currently for the class of nonlinear systems.

In biology, adaptation can be interpreted as "a favorable conformation of an organism to some changes in its environment." The phrase adaptive system was established in control theory to constitute control systems that can evaluate their own behavior and regulate parameters as per the desired behavior. Adaptive control processes are the last set in the development of control procedures. When the control process is completely defined, and the controller has access to entire data concerning the efforts of the inputs, the process is known as a deterministic control process. This process is associated with the scenario where the input is a control vector, and the engineer has entire knowledge of the functions $f_i(.)$ and the parameter vector θ. Mathematically, the unspecified elements existing in the process materialize as random variables with distribution functions which are known to the designer. Such processes are known as stochastic control processes. This process occurs when some of the input variables are random entities or when some parameters are unknown but have familiar distributions. The subsequent stage is when there is reduced information about the procedure accessible. The complete set of permitted decisions, the effect of the selected choices on the procedure, or the period of the process itself maybe unknown. In this scenario, the controller has to modify itself to improve its output by observing the states of the system. As the process gets implemented, supplementary data becomes obtainable and upgraded decisions become obtainable. Such a procedure is known as an adaptive control process [1].

2.2 Stability Theory

The fundamentals of stability are derived based on the exploration into unknown dynamical forces that disturb the system. In some cases this disturbance can be insignificant while in others it may lead to significant changes in the system's behaviour deviating from that when no disturbance is present. The Lyapunov stability theory and theory of input and output stability are based on techniques surrounding the concept of functional analysis. The problem of stability, that arises in control systems, is tackled with these two widely used approaches. In lyapunov stability, we consider the stability of the system as an internal property and we deal with the consequences of momentary perturbations that occur due to differences in initial conditions of the systems. Input output stability considers the effects of input on the network.

In this unit, we discuss some terminology related to stability and discuss the types of stability used in adaptive systems.

2.2.1 Definitions

Let the plant be defined as a nonlinear differential equation

$$x = f(x,t) \quad f(0,t) = 0 \tag{2.1}$$

where $x(t_0)$ and $f : R^+ \to R^n$ and the result $x(t; x_0, t_0)$ exists for all $t \geq t_0$. Since $f(0,t) = 0$, the origin is thus implied to be an equilibrium state.

1. The state $x = 0$ in Eq. (2.1) is considered stable if for every $\epsilon > 0$ and $t_0 > 0$, there exists a $\delta(\epsilon, t_0) > 0$ where $||x0|| < \delta$ implies $||x(t; x_0, t_0)|| < \epsilon$ for all $t \geq t_0$. ie, small disturbances result in small deviations from the equilibrium state or we should start sufficiently close to the origin to keep the trajectory close to the origin.

2. The state $x = 0$ in Eq. (2.1) is considered attractive if for $\rho > 0$ and every $\eta > 0$ and $t_0 > 0$, there exists a number $T(\eta, x_0, t_0)$ where $||x0|| < \eta$ implying $||x(t; x_0, t_0)|| < \eta$ for all $t \geq t_0 + T$, i.e., attractivity implies that all trajectories that begin along the close vicinity around the origin eventually approach the origin.

3. The state $x = 0$ in Eq. (2.1) is considered asymptotically stable when it satisfies the conditions for both definitions 1 and 2. In this case the trajectory approaches the origin as $t \to \infty$ and at the same time remains near the origin if it starts sufficiently close to origin.

4. The state $x = 0$ in Eq. (2.1) is considered uniformly stable if in definition 1 δ does not depend on initial time t_0.

All the definitions taken from [1] imply certain properties of the solution of the differential equations in the neighbourhood of the equilibrium state. While stability implies that the solution lies near the equilibrium state, asymptotic stability implies that solution tends to equilibrium as $t \to \infty$ and uniform asymptotic stability implies that the convergence of solution is independent of initial time.

2.2.2 Types of Stability

Here, we discuss the various kinds of stability which are used in adaptive systems.

1. Bounded-Input Bounded-Output Stability (BIBO)
   The plant is said to be BIBO stable, if and only if for any bounded input, the system has bounded output.A minimum requirement for BIBO stability is that all initial conditions of state variables and inputs have to be bounded [20].

2. Total Stability
   The plant is considered totally stable if it is BIBO stable for all inputs and the condition of boundedness is valid not only for zero initial conditions but for all initial conditions of the system [20].

3. Lyapunov Stability
   The lyapunov stability theorem allows one to conclusively state that a system is stable without explicitly determining solutions for the system [1]. The equilibrium state in Eq. (2.1) is uniformly asymptotically stable in the large if there exists a scalar function $V(x,t)$ with partial derivatives with respect to x and t such that $V(0,t) = 0$ and satisfies the following conditions:

   (a) $V(x,t)$ is positive definite.

   (b) $V(x,t)$ is decresent in nature.

   (c) First derivative of $V(x,t)$ is negative definite.

   (d) $V(x,t)$ is radially unbounded.

   On finding a scalar candidate function $V(x,t)$ for system defined in Eq. (2.1) such that it satisfies all of the above mentioned conditions, then using Lyapunov stability one can conclusively state that the system is uniformly asymptotically stable.

2.3 Summary

In this chapter, important definitions are introduced which are fundamental to understanding adaptive control. Stability definitions and types are discussed as well to help readers understand the concepts presented in future chapters with ease.

Chapter 3

The Identification and Control Problems

In adaptive systems, identifying the characteristics of the process is not enough to result in adequate control. Controlling the unknown plant without good identification could result in an the response of the system being poor. Therefore, the two problems are closely related and are known as the dual control problem. The controller determines both the characteristics and state of the plant depending on the information that is collected. Then the controller needs to decide the actions that are required for the successful control of the plant. The former problem presented is posed as one of identification, while the latter is posed as one of control.

Different approaches exist in order to obtain the solution of the adaptive control problem. The two approaches are indirect and direct control. In the first approach, the plant parameters are obtained by online estimation. Using these estimates, the control parameters can be adjusted. In direct control however, the control parameters are directly adjusted without estimating the plant parameters [1].

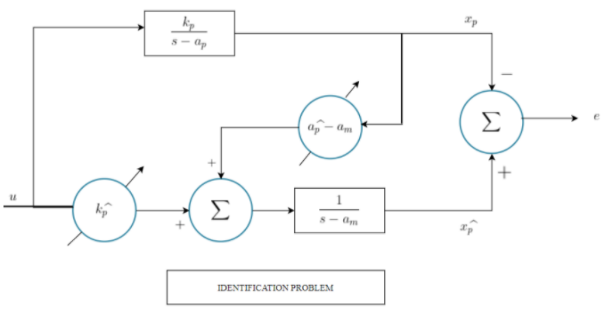


Figure 3.1: Block Diagram for Identification Problem

3.1 The Identification Problem

Given the problem of identification of an unknown plant which has bounded input u and bounded output x_p , where a_p and k_p are constant but unknown.

$$\dot{x}_p = a_p(t)x_p(t) + k_p(t)u(t) \tag{3.1}$$

The reference model is described as follows:

$$\dot{\hat{x}}_p = a_m(t)\widehat{x_p(t)} + (\widehat{a_p(t)} - a_m)x_p(t) + \widehat{k_p(t)} \tag{3.2}$$

The problem then reduces to the identification of a_p and k_p from the observed input - output pair. We use the estimator models to identify a_p and k_p.

3.1.1 Identification of a Linear Plant

The problem of identifying a dynamical plant with input $u(t)$ and an output x_p is described by the differential equation given in (3.1).

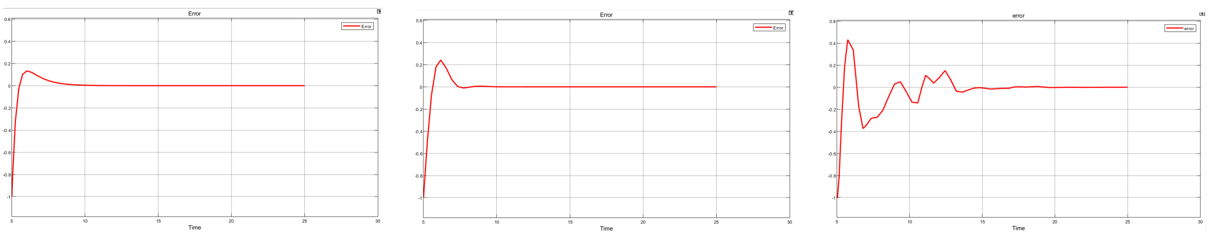
The error equation is given by:

$$e(t) = \widehat{x_p(t)} - x_p(t) \tag{3.3}$$

$$\dot{e}(t) = a_p e(t) + \varphi(t)\widehat{x_p(t)} + \psi(t)u(t). \tag{3.4}$$

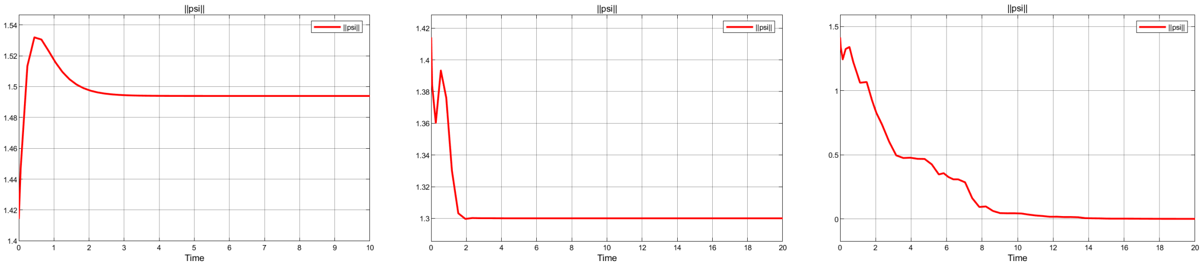
Simulation 3.1.1: Linear plant identification problem

Taking a plant described by $\dot{x}_p = -x_p + u$ with $a_p = -1$, $x_p = -1$ and $a_m = -3$ in (3.2). The estimator model is described as $\dot{x}_p = -x_m + r$. We take cases $u(t) = 0$,$u(t) = 2$ and $u(t) = 2cost + 3cos2t$.



(a) u(t) = 0 (b) u(t)= 2 (c) u(t) =2cos t + 3 cos2t

Figure 3.2: Error e(t)



(a) u(t)=0 (b) u(t) = 2 (c) u(t) =2cos t + 3 cos2t

Figure 3.3: Parametric error $\phi(t)$

Parameters: $e = \widehat{x}_p - x_p$, $\varphi = \widehat{a} + 1$, $\psi = \widehat{k} - 0.5$, $||\phi|| = \sqrt{\varphi^2 + \psi^2}$, $\dot{\widehat{x}}_p = -3\widehat{x}_p + (\widehat{a(t)} + 3)x_p + \widehat{k(t)}u$, $\dot{\widehat{a}} = -ex_p$,$\dot{\widehat{k}} = -eu$, $\dot{x}_p = -x_p + u$, $x_p(0) = 1$, $\widehat{x}_p(0) = 0$, $\widehat{a(0)} = 0$

Comments:
1) When $u(t) = 0$ it is seen that the output error converges to zero asymptotically while the parameter errors converge to non-zero constant values.
2) When $u(t) = 2$, similar behaviour of the errors is observed. The asymptomatic value of $||\phi(t)||$ is smaller in this case.
3) When $u(t) = 2cost + 3cos2t$,$e(t)$,$\phi(t)$ and $\psi(t)$ tend to zero asymptotically.

3.1.2 Identification of a Non-Linear Plant

The plant model is given as follows:

$$\dot{x}_p(t) = a_p(t)x_p(t) + \alpha f(x_p) + k_p(t)g(u) \tag{3.5}$$

a_p,α and k_p are unknown constant scalar parameters. f is a nonlinear function of x_p. The reference model is described as:

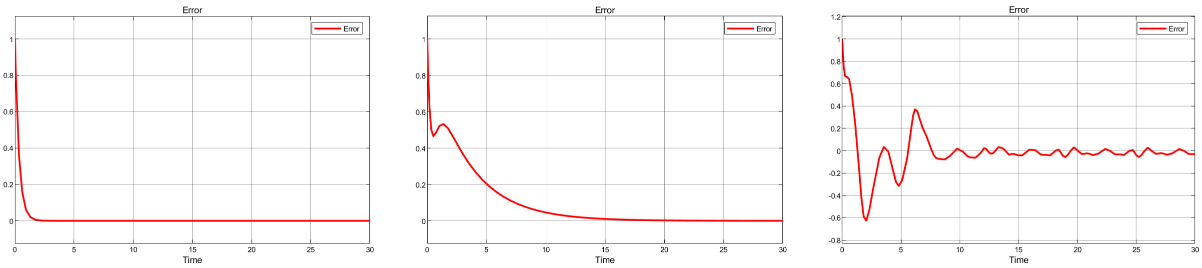
$$\dot{\widehat{x}}_p(t) = a_m(t)x_p(t) + (\widehat{a}_p(t) - a_m)x_p + \widehat{\alpha}(t)f(x_p) + \widehat{k}_p(t)g(u) \tag{3.6}$$

The error equation is as follows:

$$\dot{e} = a_m e + (\widehat{a}_p(t) - a_p)x_p + (\widehat{\alpha}(t) - a_m)x_p + (\widehat{k}_p(t) - k_p)g(u) \tag{3.7}$$

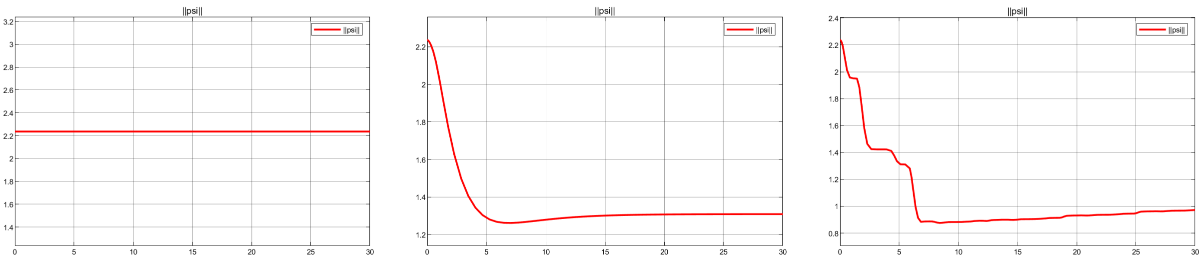
<u>Simulation 3.1.2:</u> Nonlinear plant identification problem
Taking a plant described by $\dot{x}_p = -x_p - 2x_p^3 + u$ with $a_m = -3$ in (3.2). We take cases

$u(t) = 0$,$u(t) = 2$ and $u(t) = 2cost + 3cos2t$.



(a) u(t) = 0 (b) u(t)=2 (c) u(t) =2cos t + 3 cos2t

Figure 3.4: Error e(t)



(a) u(t)=0 (b) u(t) = 2 (c) u(t) =2cos t + 3 cos2t

Figure 3.5: Parametric error $\phi(t)$

Parameters: $e = \widehat{x_p} - x_p$, $\varphi = \widehat{a} + 1$, $\psi = \widehat{k} - 1$, $||\phi|| = \sqrt{\varphi^2 + \psi^2}$, $\dot{\widehat{x_p}} = -3\widehat{x_p} + (\widehat{a(t)} + 3)x_p + \widehat{k(t)}u$, $\dot{\widehat{a}} = -ex_p$,$\dot{\widehat{k}} = -eu$, $\dot{x}_p = -x_p + u$, $x_p(0) = 1$,$\widehat{x_p}(0) = 0$, $\widehat{a(0)} = 0$
Comments:
1) When $u(t) = 0$ it is seen that the output error converges to zero while the parameter errors do not.
2) When $u(t) = 2cost + 3cos2t$, the output as well as the parameter errors tend to zero.
3) The speed of convergence of the parameter error in the nonlinear plant case is much slower than that in the case of a linear plant.

3.2 The Control Problem

3.2.1 Control of a Linear Plant

A plant with an input-output pair is described by the differential equation

$$x_p(t) = a_p(t)x_p(t) + k_p(t)u(t) \tag{3.8}$$

where the plant parameters are defined as $a_p(t)$ and $k_p(t)$. The reference model is described by the first order differential equation

$$x_m(t) = a_m(t)x_m(t) + k_m(t)r(t) \tag{3.9}$$

where $a_m(t) < 0$, $a_m(t)$ and $k_m(t)$ are constant values, and $r(t)$ is a piece wise continuous bounded function of time. The parameters are chosen to represent the required plant output at time t.

The aim is thus to determine a bounded input $u(t)$ such that all signal values remain bounded and $\lim_{t\to\infty} |x_p(t) - x_m(t)| = 0$

Simulation 3.2.1: Control Problem for a Linear Time Invariant Plant

Taking plant described by $x_p = x_p + 2u$ with $a_p = 1$ and $k_p = 2$. The reference model is described as $x_m = -x_m + r$.

We take cases $r(t) = 0$, $r(t) = 5$ and $r(t) = 2cost + 3cos2t$.

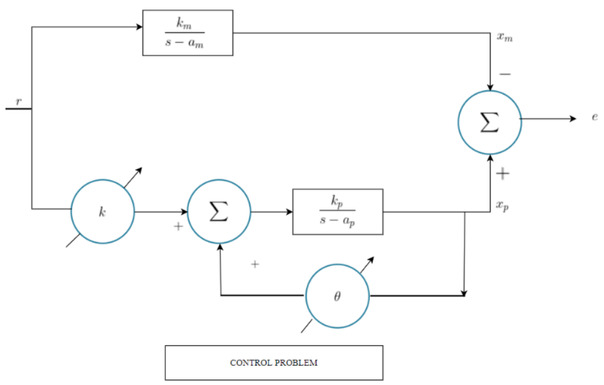
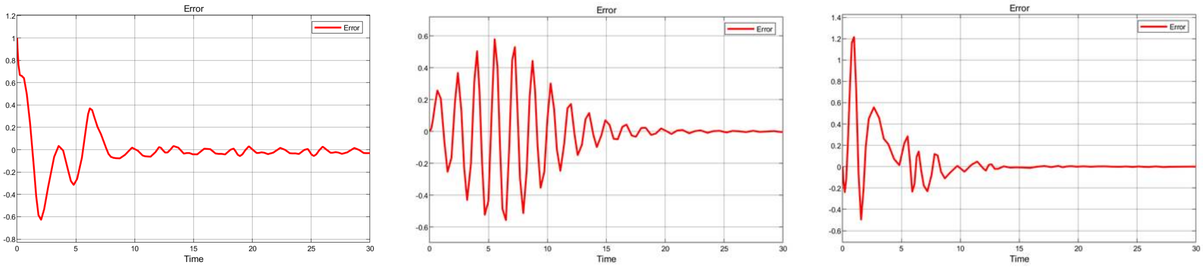
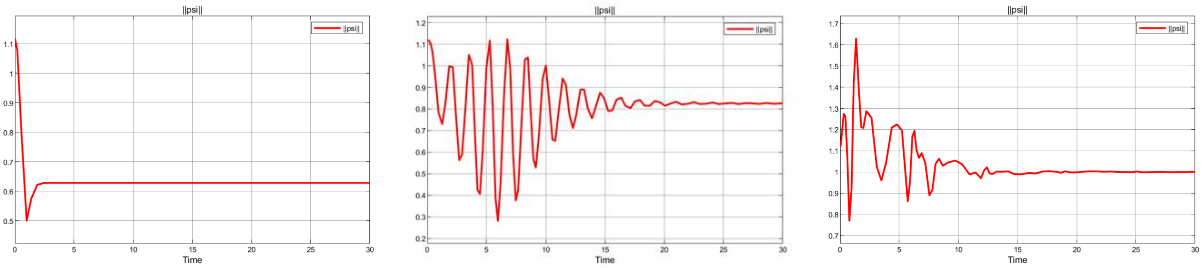


Figure 3.6: Block Diagram for Control Problem



(a) u(t) = 0 (b) u(t)=5 (c) u(t) =2cos t + 3 cos2t

Figure 3.7: Error e(t)



(a) u(t)=0 (b) u(t) = 5 (c) u(t) =2cos t + 3 cos2t

Figure 3.8: Parametric error $\phi(t)$

Parameters: $e = x_p - x_m$, $\phi = \theta + 1, \psi = k - 0.5$, $||\phi|| = \sqrt{\phi^2 + \psi^2}$, $\dot{x_m} = -x_m + r$, $\dot{x_p} = x_p + 2(\theta(t)x_p + k(t)r)$, $\theta = -e * x_p$, $k = -e * r$, $x_p(0) = 1$, $x_m(0) = 0$, $k(0) = 0$, $\theta(0) = 0$

Comments:
1) When $r(t) = 0$ and $r(t) = 5$, the plant stabilises but the parametric errors don't approach zero as $t \to \infty$.
2) When $r(t) = 2cost + 3cos2t$ both the system and parametric errors approach zero.

3.2.2 Control of a Non-Linear Plant

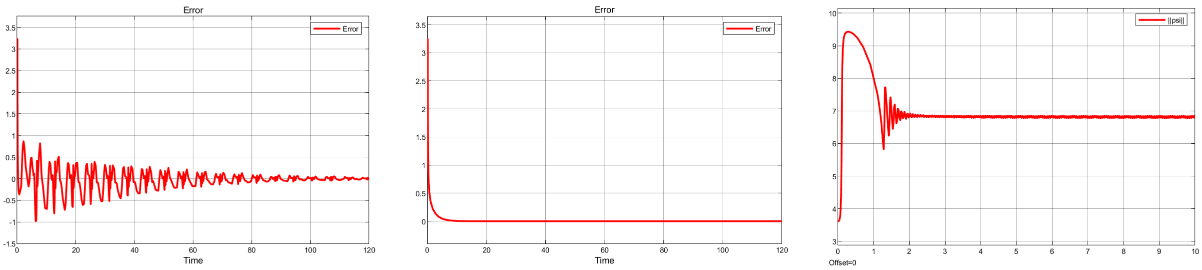
Plant is nonlinear and described as:

$$x_p = a_p x_p + \alpha f(x_p) + k_p u \tag{3.10}$$

where f is a nonlinear function of x_p and a_p, α and k_p are unknown.

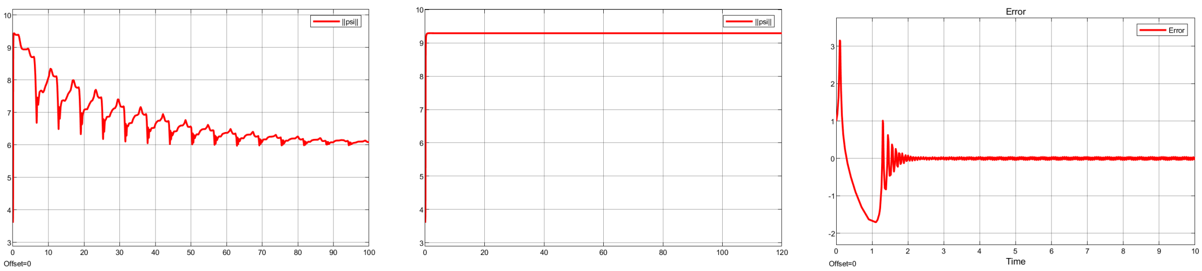
<u>*Simulation 3.2.2:*</u> *Control Problem for a Nonlinear Time Invariant Plant*

Taking plant described by $x_p = x_p + 3x_p^3 + u$. The reference model is described as $x_m = -x_m + r$. We take cases $r(t) = 0$, $r(t) = 5$ and $r(t) = 2cost + 3cos2t$.



(a) u(t) = 0 (b) u(t)=5 (c) u(t) =2cos t + 3 cos2t

Figure 3.9: Error e(t)



(a) u(t)=0 (b) u(t) = 5 (c) u(t) =2cos t + 3 cos2t

Figure 3.10: Parametric error $\phi(t)$

Parameters: $e = x_p - x_m$, $\phi = \theta + 2$,$\bar{\alpha} = \hat{\alpha} - 3$, $||\phi|| = \sqrt{\phi^2 + \psi^2}$,$\dot{x_m} = -x_m + r$,$\dot{x_p} = x_p + 3x_p^3 + u$, $u = \theta(t)x_p + \hat{\alpha}(t)x_p^3 + r$,$\dot{\theta} = -e * x_p$,$\dot{k} = -e * x_p^3$, $x_p(0) = 1$,$x_m(0) = 0$,$\hat{\alpha}(0) = 0$,$\theta(0) = 0$

Comments:

1)The results seen are similar to simulation 3.3.1

2)We see that the plant in control problem can be unstable. Unlike the identification problem the main objective in the control problem is for plant output to follow the model asymptotically.

3)In the control problem, parameters are adjusted in the plant feedback loop. Hence the control process may become unstable.

3.3 Summary

The identification and control problem problem has been discussed in this chapter. The identification problem for both the linear as well as the nonlinear case has been examined. The control problem is then investigated for the linear and the nonlinear case. The nuances of the identification and control problem can be appreciated through this chapter.

Chapter 4

Adaptive Observers and Controllers

In this chapter, we examine the necessity of adaptive observers and controllers. In Subsection 4.1, we investigate the two representations of adaptive observers and look at the differences between the two via the simulations. The chapter then delves into the concept of adaptive controllers in subsection 4.3. The objective is to ensure a plant with unknown parameters P follows a desired trajectory asymptotically.

4.1 Adaptive Observers

In the previous chapter, we assumed the entire state vectors were accessible for measurement and could be used in the identification and control problem. In practical situations however, only the output of the plant is available to be measured. In such cases we cannot directly apply methods of identification and control. Hence, we estimate the values of the state variables using adaptive observers. The observer equations are as follows:

$$\dot{\hat{x}}p = a_m(t)\widehat{x_p(t)} + (\widehat{a_p(t)} - a_m)x_p(t) + \widehat{k_p(t)} \tag{4.1}$$

$$\dot{\omega_1} = \Lambda\omega_1 + lu$$

$$\dot{\omega_2} = \Lambda\omega_2 + ly_p$$

$$y_p = x_1$$

$$\theta = [c_0, \bar{c}, d_0, \bar{d}]^T$$

$$\omega = [u, \omega_1, y_p, \omega_2]^T$$

Simulation 4.1.1: Adaptive Observers with Representation I

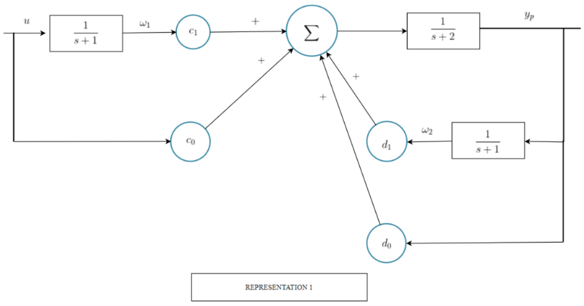
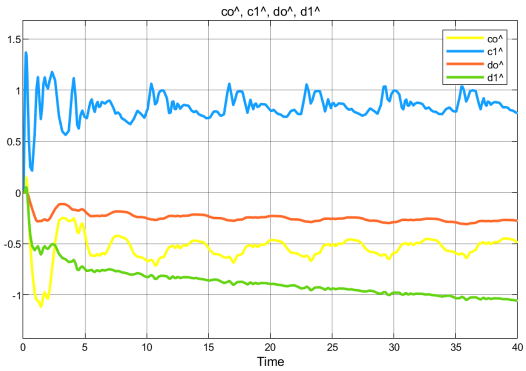
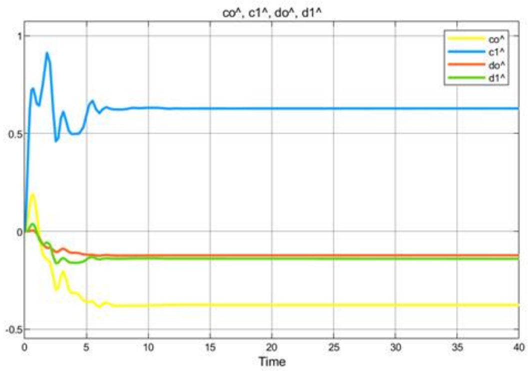


Figure 4.1: Block Diagram for Adaptive Observers Representation 1

The transfer function of a plant to be identified was chosen as $Wp(s) = \frac{s+1}{(s+2)(s+3)}$ The parameters c_0,c_1 ,d_0 and d_1 can be calculated as $c_0 = 1, c_1 = 0, d_0 = -2, d_1 = -2$. The observer has 4 adjustable parameters $\hat{c_0}, \hat{c_1}, \hat{d_0}, \hat{d_1}$.

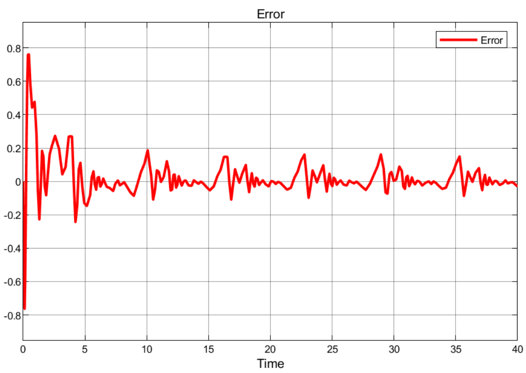


(a) u(t) = 5cost

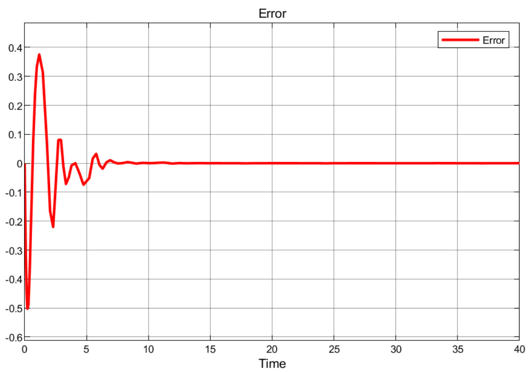


(b) u(t)=5 cos t+10 cos 2t

Figure 4.2: Parameters $\hat{c_0}, \hat{c_1}, \hat{d_0}, \hat{d_1}$



(a) u(t) = 5 cos t



(b) u(t)=5 cos t+10 cos 2t,

Figure 4.3: Error $e(t)$

Parameters: $\dot{\hat{c}}_0 = -\gamma_1 e_1 u$, $\dot{\hat{c}}_1 = -\gamma_2 e_1 \omega_1$, $\dot{\hat{d}}_0 = -\gamma_3 e_1 y_p$, $\dot{\hat{d}}_1 = -\gamma_4 e_1 \omega_2$

Comments:

1) For case u(t)=5cost the output error is seen to tend to zero. The parameter values tend to constant values that are not the same as those of the plant.

2) For u(t) = 5 cos t+10 cos 2t, all the parameter estimates approach their true values.

Simulation 4.1.2: Adaptive Observers with Representation II

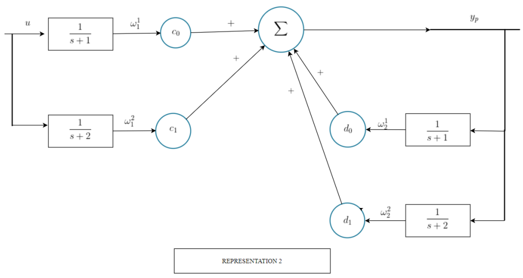
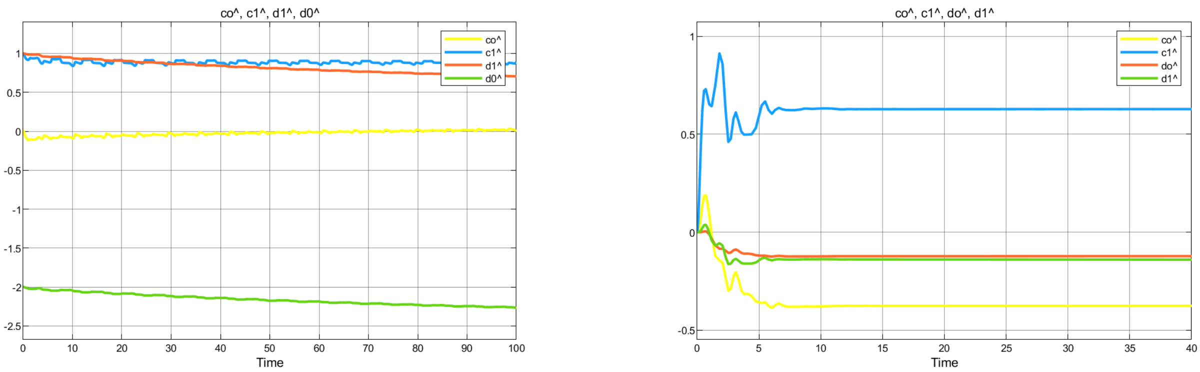


Figure 4.4: Block Diagram for Adaptive Observers Representation 2

The identification of plant is the same as simulation 4.1.1. The parameters c_0,c_1 ,d_0 and d_1 can be calculated as $c_0 = 0, c_1 = 1, d_0 = -2, d_1 = 0$.

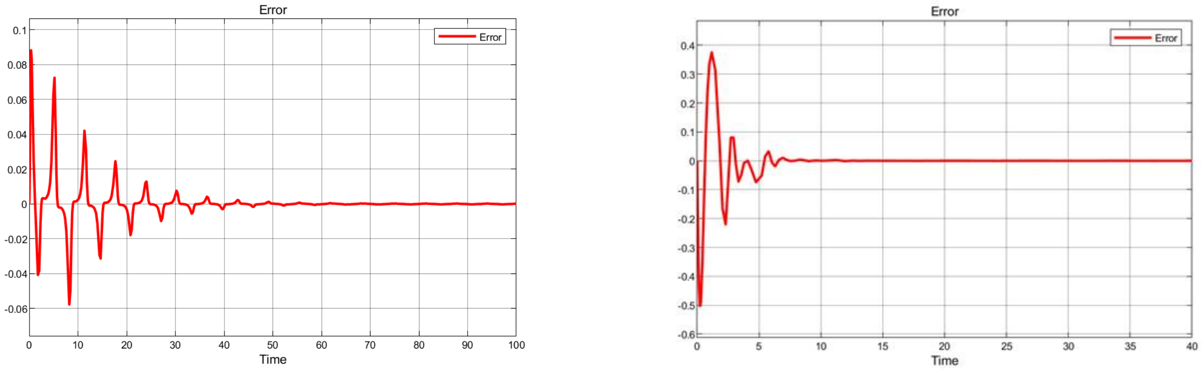
The observer has 4 adjustable parameters $\hat{c}_0, \hat{c}_1, \hat{d}_0, \hat{d}_1$.



(a) u(t) = 5cost

(b) u(t)=5 cos t+10 cos 2t

Figure 4.5: Parameters $\hat{c}_0, \hat{c}_1, \hat{d}_0, \hat{d}_1$



(a) u(t) = 5 cos t (b) u(t)=5 cos t+10 cos 2t,

Figure 4.6: Error $e(t)$

Parameters: $\dot{\hat{c}}_0 = -\gamma_1 e_1 \omega_1^1$, $\dot{\hat{c}}_1 = -\gamma_2 e_1 \omega_1^2$, $\dot{\hat{d}}_0 = -\gamma_3 e_1 \omega_2^1$, $\dot{\hat{d}}_1 = -\gamma_4 e_1 \omega_2^2$

Comments:

1) The output error tends to zero for u(t)=5cost, even while the parameter values tend to constant values that aren't the same as those of the plant.

2) The output error approaches zero and all parameter estimates approach their true values in the case of u(t)=5 cost + 10 cos2t.

4.2 Adaptive Controllers

The problem of tracking of control of the plant P with unknown parameters can be divided into two parts:

1) Regulation: The objective is to stabilise the plant P, using input output data.

2) Tracking problem: The objective is to make the model output follow a desired signal asymptotically.

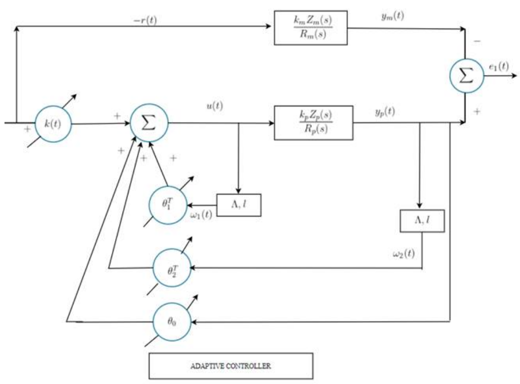


Figure 4.7: Block Diagram for Adaptive Controller

The adaptive controller is described by:

$$\dot{x}_p = A_p x_p(t) + b(\theta^T(t)\omega(t)) \tag{4.2}$$

$$\dot{\omega}_1(t) = \Lambda\omega_1(t) + l(\theta^T(t)\omega(t))$$

$$\dot{\omega}_2(t) = \Lambda\omega_2(t) + l(h_p^T(t)x_p(t))$$

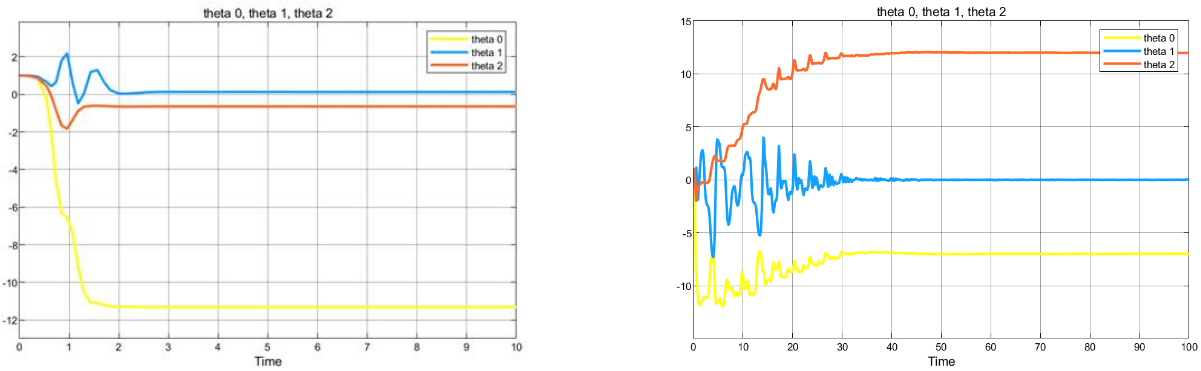
The adaptive laws for the controller are defined as follows:

$$\dot{k}(t) = -sgn(k_p)e_1(t)r(t) \tag{4.3}$$

$$\dot{\theta}_0(t) = -sgn(k_p)e_1(t)y_p(t)$$

$$\dot{\theta}_1(t) = -sgn(k_p)e_1(t)\omega_1(t)$$

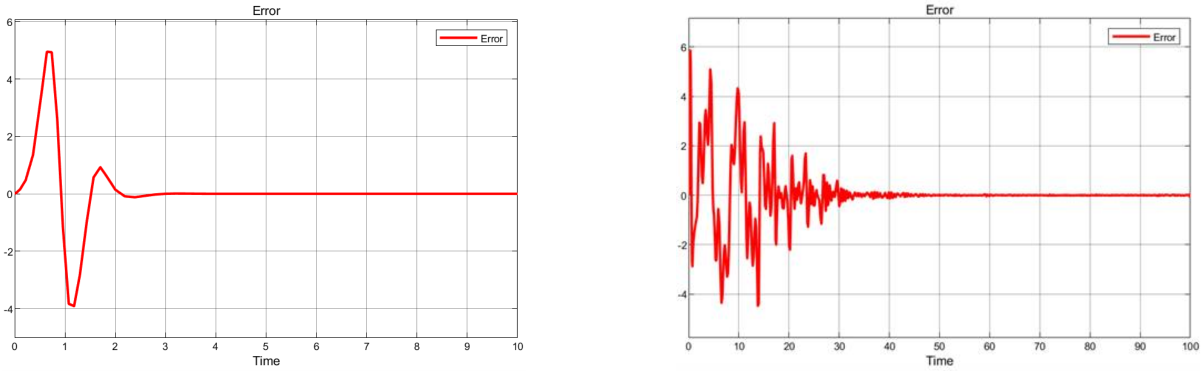
Simulation 4.2.1: Adaptive Controller



(a) r(t)=10

(b) r(t)=5 cos t+10 cos 2t

Figure 4.8: Parameters $\theta_0, \theta_1, \theta_2$



(a) r(t)=10 (b) r(t)=5 cos t+10 cos 2t,

Figure 4.9: Error $e(t)$

Parameters: $y_p = W_p(s)u$, $y_m = W_m(s)r$, $W_p(s) = \frac{s+1}{(s-2)(s-1)}$, $W_m(s) = \frac{1}{s+1}$,$\Lambda = -1$, $l = 1$, $\omega_1 = \frac{1}{s+1}u$, $\omega_2 = \frac{1}{s+1}y_p$, $e_1 = y_p - y_m$,$\theta_1^* = 0$, $\theta_0^* = -5$, $\theta_2^* = 6$, $\dot{\theta}_1. = -e_1\omega_1$, $\dot{\theta}_0 = -e_1 y_p$, $\dot{\theta}_2 = -e_1\omega_2$

Comments:
1) For the case $r(t) = 0$ error approaches 0 asymptotically while parameter errors don't converge to zero. Parameters stabilise to some values.
2) The error approaches 0 asymptotically while parameter errors do not converge to zero for $r(t) = 10$.
3) The error tends to 0 asymptotically and parameter errors converge to zero. Parameters stabilise to desired values for $r(t) = 5cost + 10cos2t$.

4.3 Summary

The necessity of having adaptive observers and controllers is scrutinized in the chapter. Two types of representations of adaptive observers are briefly introduced and simulations are given to observe their operation. The chapter ends with the presentation of the control of a plant P using adaptive controllers and a simulated example of the same.

Chapter 5

Mutual Adaptation: The First Order Case

In the previous chapters we have dealt with cases of adaptation where there was a reference model, using which the plant adapts its parameters. But what if we need to deal with a case where there is no reference model? Consider two first order systems with the following differential equations respectively and connected as shown in (Fig. 5.1),

$$\Sigma_1 : \dot{x_1}(t) = a_1(t)x_1(t) + bu(t), \quad x_1(t_0) = x_{10} \tag{5.1}$$

$$\Sigma_2 : \dot{x_2}(t) = a_2(t)x_2(t) + bu(t), \quad x_2(t_0) = x_{20} \tag{5.2}$$

where, $a_1(t)$ and $a_2(t) \in \mathbb{R}^{1\times 1}$ are the unknown parameters of the system, b is a known positive constant and the input $u(.)$ is a piecewise continuous and bounded signal. We assume that $a_1(t_0) \neq a_2(t_0)$ and $x_{10} \neq x_{20}$. The two subsystems Σ_1 and Σ_2 mutually interact with each other in order to adapt their parameters such that the overall system has a stable desired output. This occurs when the error between the two subsystems tends to zero.

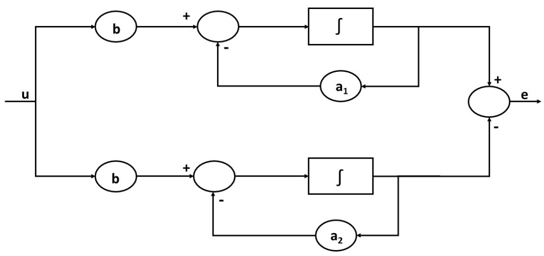


Figure 5.1: Block Diagram for First Order Mutual Adaptation

5.1 Approach

1) Derive adaptive laws using Lyapunov Stability.
2) Construct the model.
3) Vary the input to check the effect on parameter adaptation (effect of persistent and non persistent input).
4) Observe the effect on the rate of adaptation due to the changes in initial conditions of the state variables and parameters of the plant.

5.2 Stability Analysis

We consider the two sub systems mentioned below:

$$\sum_1 : \dot{x_1} = a_1(t)x_1 + bu \tag{5.3}$$

$$\sum_2 : \dot{x_2} = a_1(t)x_2 + bu$$

The error for the overall system is defined as

$$e(t) = x_1 - x_2 \tag{5.4}$$

Taking the derivative of error,

$$\dot{e}(t) = \dot{x}_1 - \dot{x}_2$$

Substituting, we get

$$\dot{e}(t) = a_1x_1 - a_2x_2$$

Let a^* be a unknown negative constant. a_1 and a_2
Thus we can define the parameter errors as

$$\phi_1 = a_1 - a^* \tag{5.5}$$

$$\phi_2 = a_2 - a^*$$

We can express $\dot{e}(t)$ in terms of e,ϕ_1,ϕ_2:

$$\dot{e} = a_1x_1 - a_2x_2 + (a^*x_1 - a^*x_1) + (a^*x_2 - a^*x_2)$$

Rearranging the equation,

$$\dot{e} = a^*(x_1 - x_2) + x_1(a_1 - a^*) - x_2(a_2 - a^*)$$

Substituting

$$\dot{e} = a^*e + \phi_1 x_1 - \phi_2 x_2$$

We define a candidate Lyapunov function

$$V = \frac{1}{2}(e^2 + \phi_1^2 + \phi_2^2) \tag{5.6}$$

$V(0,0,0) = 0$ and $V(\infty)$ is radially unbounded.
Thus we find the first derivative of $V(e, \phi_1, \phi_2)$

$$\begin{aligned}
\dot{V} &= \dot{e}e + \dot{\phi}_1\phi_1 + \dot{\phi}_2\phi_2 \\
&= e(a^*e + \phi_1 x_1 - \phi_2 x_2) + \dot{\phi}_1\phi_1 + \dot{\phi}_2\phi_2 \\
&= a^*e^2 + \phi_1(ex_1 + \dot{\phi}_1) + \phi_2(\dot{\phi}_2 - ex_2)
\end{aligned}$$

for $\dot{V} \leq 0$ we substitute,

$$\dot{\phi}_1 = \dot{a}_1 = -ex_1(t) \tag{5.7}$$

$$\dot{\phi}_2 = \dot{a}_2 = ex_2(t)$$

Therefore,

$$\dot{V} = a^*e^2 < 0 \, as \, a^* < 0$$

and

$$\dot{a}_1 + \dot{a}_2 = -e(x_1 - x_2) = -e^2 < 0$$

As $\dot{e}$ is bounded, from barbalat's lemma

$$\lim_{t->\infty} e(t) = 0$$

5.3 Simulink Model

The simulink model constructed to carry out mutual adaptation between two first order systems is as follows:

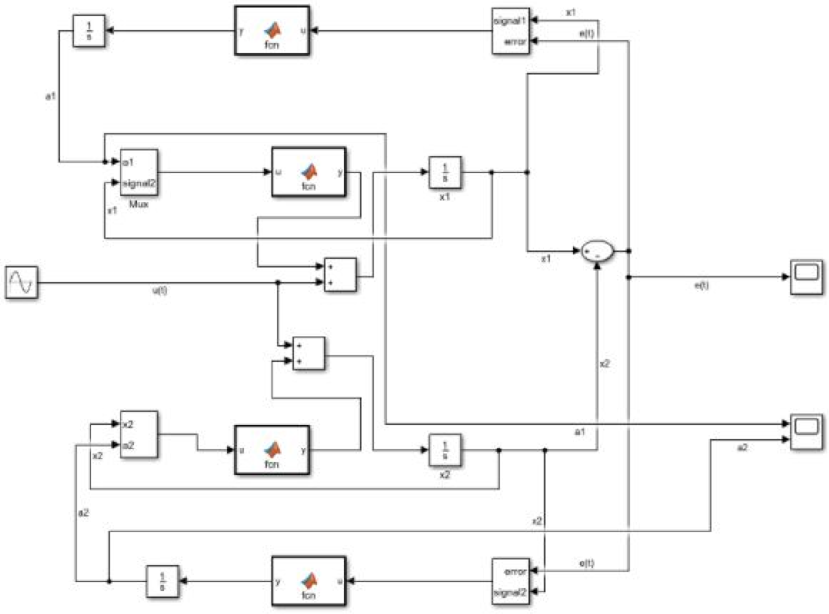


Figure 5.2: Simulink Model for First Order Mutual Adaptation

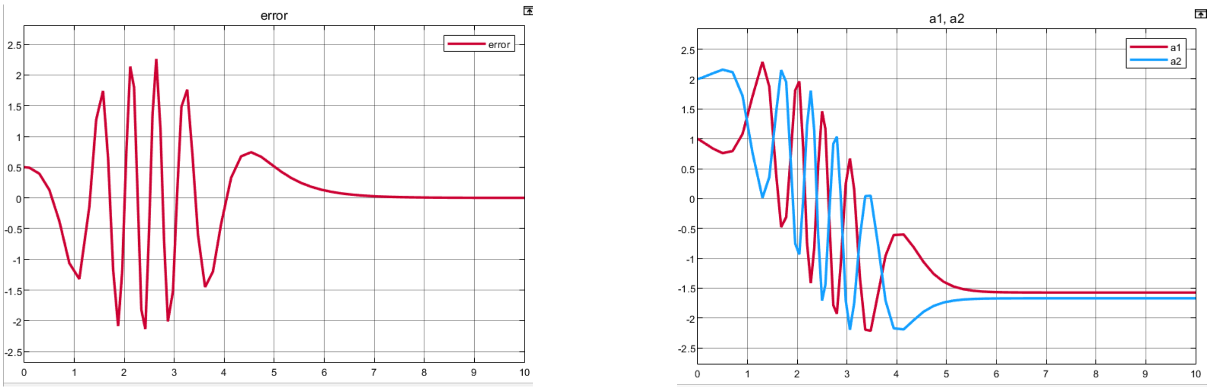
5.4 Simulation Studies

In this section, we perform the mutual adaptation problem on first order systems for cases in varying inputs, initial conditions of parameter values and states variables. We note some unique trends in the outputs of the systems which can be generalised for the first order case.

5.4.1 Persistency of Input

Simulation 5.4.1: Case of Non- Persistent Input

We take the input $u(t) = 0$ and initial conditions $x_t(0) = 3$, $x_2(0) = 5$, $a_1(0) = 0.5$ and $a_2(0) = 0.2$.



(a) error $e(t)$ (b) Parameters $a_1(t)$ and $a_2(t)$

Figure 5.3: $u(t) = 0$

Insights:

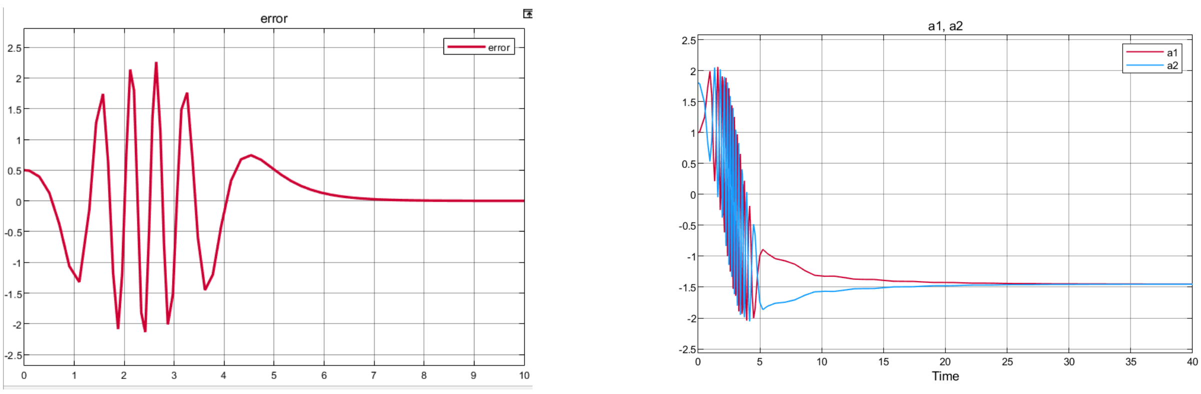
1) Final values of $a_1(t) = -1.55$ and $a_2(t) = -1.63$.
2) Error of system dies down to zero.
3) The parameters oscillate indicating adaptation.

Observation:

The parameter values a1 and a2 stabilize to two distinct negative values in the case of a non persistent input u(t).

Simulation 5.4.2: Case of Persistent Input

We take the input $u(t) = sin(t)$ and initial conditions $x_t(0) = 1$, $x_2(0) = 0.5$, $a_1(0) = 1$ and $a_2(0) = 2$.



(a) error $e(t)$ (b) Parameters $a_1(t)$ and $a_2(t)$

Figure 5.4: $u(t) = sin(t)$

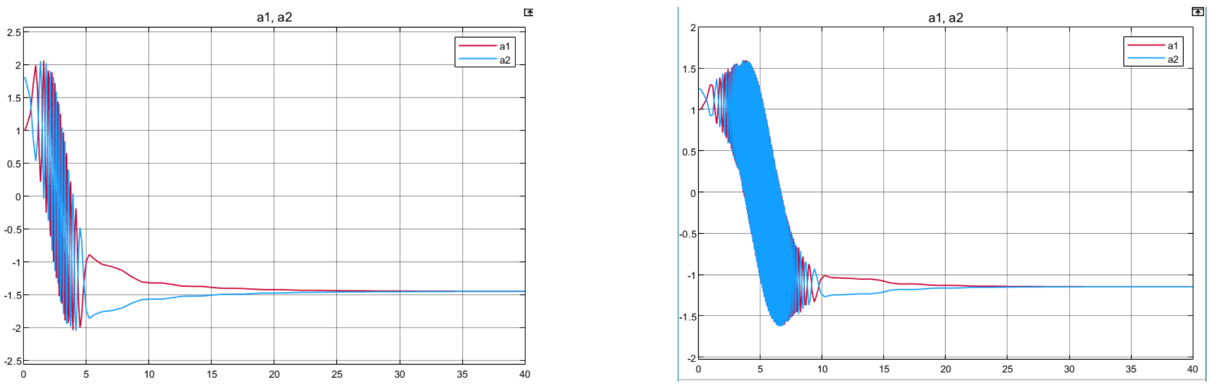
Insights:

1) Final values of $a_1(t) = a_2(t) = -1.49$.
2) Error of system dies down to zero.
3) The parameters oscillate indicating adaptation.
4) The rate of adaptation more in the persistent case.

Observation:
The parameter values a1 and a2 stabilize to a single negative value in the case of a persistent input u(t).

5.4.2 Initial Parameter Values

Simulation 5.4.3: Case of Decreasing Difference of Initial Parameter Values
We take the input $u(t) = sin(t)$ and initial conditions $x_1(0) = 1$, $x_2(0) = 1$ for this simulation. Here, $a_1(0) = 1$ and $a_2(0) = 1.8$ for (Fig 5.5a) and $a_1(0) = 1$ and $a_2(0) = 1.25$ for (Fig 5.5b)



(a) $a_1(0) = 1$ and $a_2(0) = 1.8$ (b) $a_1(0) = 1$ and $a_2(0) = 1.25$

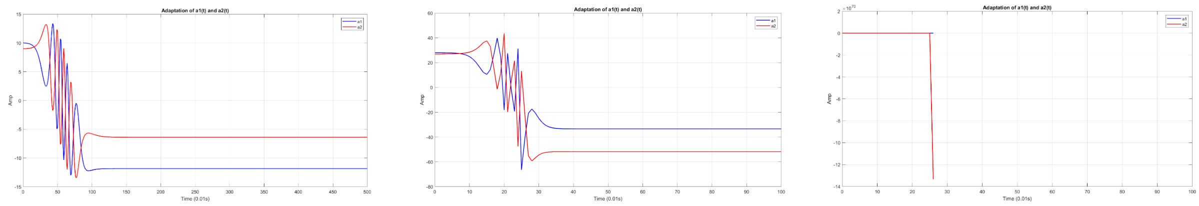
Figure 5.5: Decreasing Difference of Initial Parameter Values

Insights:
1) As the difference between initial values of parameters $a_1(t)$ and $a_2(t)$ increases, the number of oscillations decrease.
2) The initial conditions of $x_1(t)$ and $x_2(t)$ are the same for both cases.
3) The adaptation is more in (Fig: 5.5a) than (Fig: 5.5b).

Observation:
When the difference between initial values of a_1 and a_2 is lesser, more adaptation of the parameters occurs.

<u>*Simulation 5.4.4:*</u> *Case of Changing Difference of Initial Parameter Values*

We take the input $u(t) = 0$ and initial conditions $x_1(0) = 1$, $x_2(0) = 0.5$ for this simulation. We test different cases of $a_1(t)$ and $a_2(t)$ for difference of 1, 10 and 20 between the initial parameter values. We also test cases for when $a_1(t) > a_2(t)$ and vice versa.



(a) $a_1(0) = 10$ and $a_2(0) = 9$ (b) $a_1(0) = 28$ and $a_2(0) = 27$ (c) $a_1(0) = 29$ and $a_2(0) = 28$

Figure 5.6: Difference of 1 in Initial Parameter Values

Table 5.1: Border Conditions for Stability when $x_1(0) = 1$, $x_2(0) = 0.5$ and $u(t) = 0$

Difference b/w $a_1(t)$ and $a_2(t)$	$a_1(t)$	$a_1(t)$
1	28	27
1	12	13
10	64	54
10	24	34
20	96	76
20	40	60

Insights:

1) There is an upper limit of values seen while choosing initial values for a1 and a2. Above this value the system becomes unstable.

2) Table 7.4 shows the boundary conditions below which the overall system gives a stable output for $x_1(0) = 1$, $x_2(0) = 0.5$ and $u(t) = 0$. These boundary conditions change when any one or more of the initial conditions of the system change.

Observation:

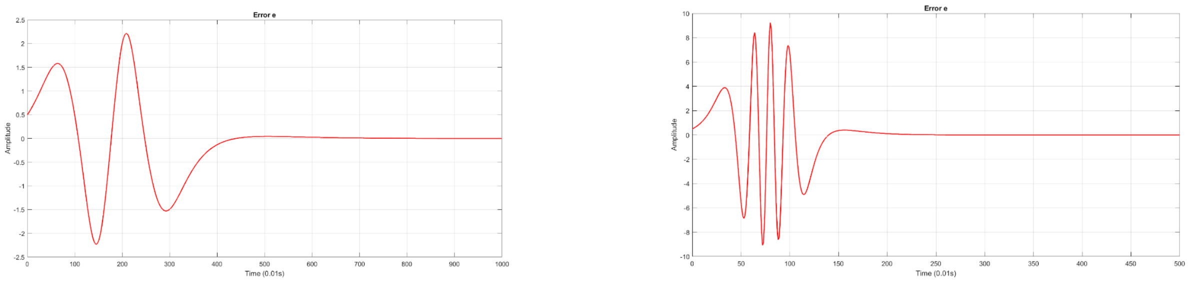
1) As the difference between initial values of a_1 and a_2 increase, the upper limit values of a_1 and a_2 for which the system is stable also increases.

2) The upper limit values of a_1 and a_2 for the system to be stable obtained for case of $a_1 > a_2$ is higher than case $a_2 > a_1$.

5.4.3 Convergence of Error

Simulation 5.4.5: Case of Rate of Convergence of Error

We take the input $u(t) = 0$ and initial conditions $x_1(0) = 1$, $x_2(0) = 0.5$ for this simulation. We compare the rate of convergence of error for cases when $a_1(0) = 2$, $a_2(0) = 1$ (Fig: 5.7a) and $a_1(0) = 7$, $a_2(0) = 6$ (Fig: 5.7b).



(a) $a_1(0) = 2$, $a_2(0) = 1$

(b) $a_1(0) = 7$, $a_2(0) = 6$

Figure 5.7: Rate of Convergence of Errors

Insights:

1) The error when $a_1 = 2$, $a_2 = 1$ convergences around 5s while error when $a_1 = 7$, $a_2 = 6$ converges around 2s.

Observation:

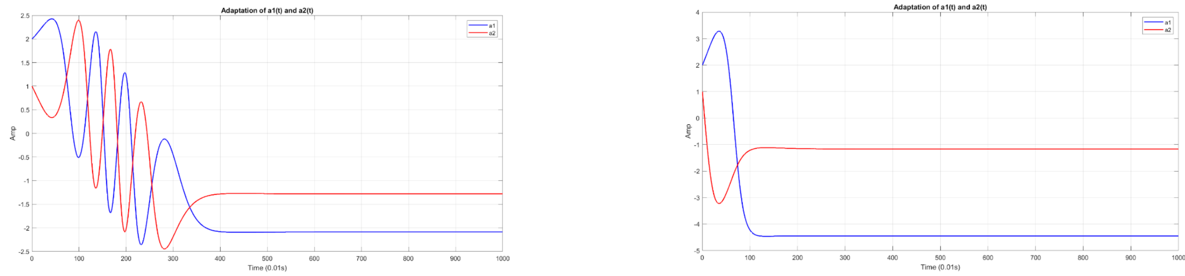
1) As the difference between initial values of a_1 and a_2 increase, the upper limit values of a_1 and a_2 for which the system is stable also increases.

2) The upper limit values of a_1 and a_2 for the system to be stable obtained for case of $a_1 > a_2$ is higher than case $a_2 > a_1$.

5.4.4 Initial State Variables

Simulation 5.4.6: Case of Difference in Initial Conditions

We take the input $u(t) = 0$ and initial parameter values $a_1(0) = 2$, $a_2(0) = 1$ for this simulation. We compare the cases when $x_1(0) = 2$, $x_2(0) = 1$ (Fig: 5.8a) and $x_1(0) = 1$, $x_2(0) = 5$ (Fig: 5.8b).



(a) $x_1(0) = 2,\ x_2(0) = 1$ (b) $x_1(0) = 1,\ x_2(0) = 5$

Figure 5.8: Difference in Initial State Variables

Insights:

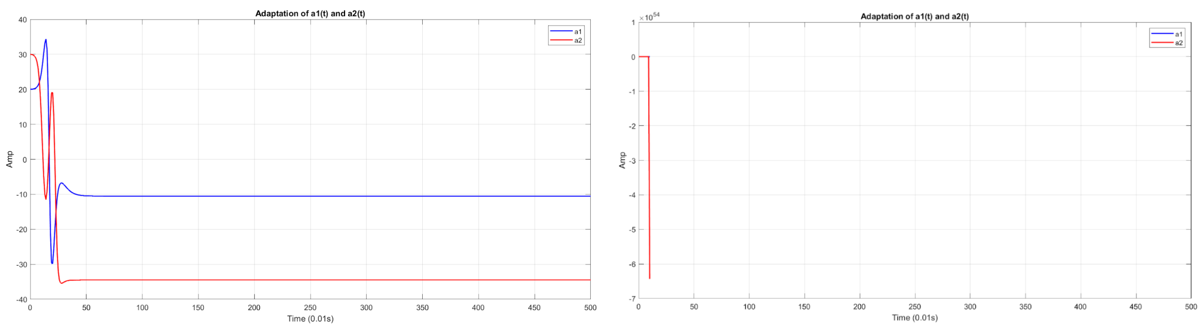
1) The final values of a_1 and a_2 are closer together in the case of $x_1 = 1$ and $x_2 = 2$ as compared to $x_1 = 1$ and $x_2 = 5$.

Observation:

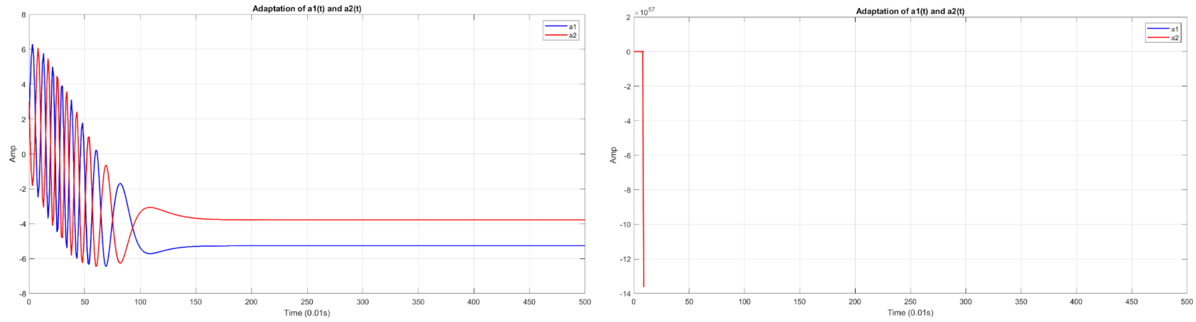
1) Given a non persistent input, for a fixed value of a_1 and a_2, as the difference between x_1 and x_2 increases, the difference between the final values of the unknown parameters a_1 and a_2 increases.

Simulation 5.4.7: Case of Difference in Initial Conditions and Parameter Values

In this simulation we take $u(t) = 0$ and compare four sets of initial conditions $x_1(0), x_2(0)$ and parameter values $a_1(0), a_2(0)$.



(a) $x_1(0) = 3,\ x_2(0) = 2,\ a_1(0) = 20,\ a_2(0) = 30.$ (b) $x_1(0) = 35,\ x_2(0) = 56,\ a_1(0) = 20,\ a_2(0) = 30.$



(c) $x_1(0) = 35$, $x_2(0) = 40$, $a_1(0) = 2$, $a_2(0) = 3$. (d) $x_1(0) = 35$, $x_2(0) = 40$, $a_1(0) = 27$, $a_2(0) = 31$.

Figure 5.9: Difference in Initial State Variables and Parameter Values

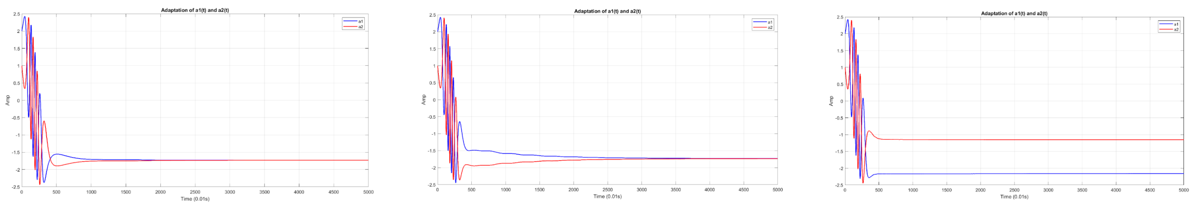
Insights:
1) We see that adaptation fails to occur when initial conditions of both a_1 and a_2 as well as x_1 and x_2 are large values.

Observation:
1) As initial values of unknown parameters a_1 and a_2 increase, the initial conditions of x_1 and x_2 must decrease so as to keep the system stable and vice versa.

5.4.5 Frequency of Persistent of Input

<u>*Simulation 5.4.8: Case of Changing Frequency of Input*</u>

In this simulation we take initial conditions $x_t(0) = 1$, $x_2(0) = 2$, $a_1(0) = 2$ and $a_2(0) = 1$. We provide a sinusoidal input changing the frequency in cases where $u(t) = sin(0.25t)$, $u(t) = sin(t)$ and $u(t) = sin(10t)$.



(a) $u(t) = sin(0.25t)$ (b) $u(t) = sin(t)$ (c) $u(t) = sin(10t)$

Figure 5.10: Frequency of Persistent of Input

Insights:
1) We see that the parameters converge at 20s when input is sin(0.25t) and parameters do not converge in the time interval when input is sin(10t).

Observation:
1) As the frequency of the sine input reduces, the parameter convergence occurs faster.

5.5 Summary

In this chapter the motivation for the mutual adaptation problem is provided . The mathematical analysis for the first order mutual adaptation problem is examined and the various characteristics of first order systems undergoing mutual adaptation are investigated via numerous simulations.

Chapter 6

Mutual Adaptation: The nth Order Case

In the previous chapter, we presented the concept of mutual adaptation using the example of two first-order systems which follows the results in [2,3]. In this chapter, we generalize this to nth order systems.
Let the two n^{th} order systems be represented as:

$$\Sigma_1 : \dot{x}_1(t) = A_1(t)x_1(t) + bu(t), \quad x_1(t_0) = x_{10} \tag{6.1}$$

$$\Sigma_2 : \dot{x}_2(t) = A_2(t)x_2(t) + bu(t), \quad x_2(t_0) = x_{20} \tag{6.2}$$

where $A_1(t)$ and $A_2(t) \in \mathbb{R}^{n\times n}$ are the unknown parameters of the system, $b \in \mathbb{R}^{n\times n}$ is a known positive constant, and the input $u(.)$ is piecewise continuous and bounded. We assume $A_1(t_0)$ and $A_2(t_0)$ are unstable system matrices. The systems are connected as shown in (Fig. 6.1). The objective here is to make the error between state vectors converge to zero asymptotically as the respective elements in parameter matrices $A_1(t)$ and $A_2(t)$ reach a consensus with adaptation.

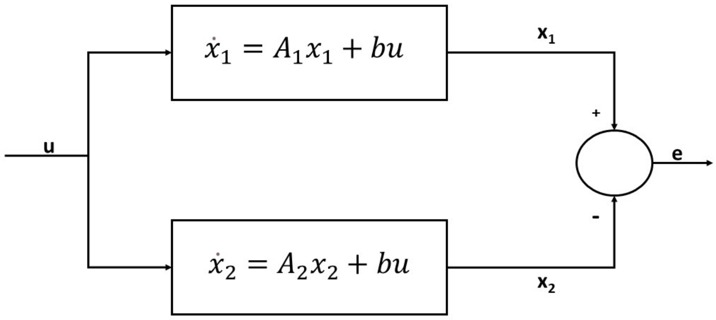


Figure 6.1: Block Diagram for nth Order Mutual Adaptation

6.1 Approach

1) Derive adaptive laws using Lyapunov Stability.
2) Construct the model.
3) Vary the input to check the effect on parameter adaptation (effect of persistent and non persistent input).

6.2 Stability Analysis

Consider the two sub systems mentioned below:

$$\sum_1 : \dot{x_1} = A_1(t)x_1(t) + bu(t)$$

$$\sum_2 : \dot{x_2} = A_2(t)x_2(t) + bu(t) \tag{6.3}$$

The error for the overall system is defined as

$$e(t) = x_1(t) - x_2(t) \tag{6.4}$$

Taking the derivative of error,

$$\dot{e}(t) = \dot{x}_1(t) - \dot{x}_2(t) \tag{6.5}$$

Substituting Eq (1) in (3), we get

$$\dot{e}(t) = A_1(t)x_1(t) - A_2(t)x_2(t) \tag{6.6}$$

Let A^* be a known negative constant. The parametric errors are defined as:

$$\phi_1(t) = A_1(t) - A^*$$

and

$$\phi_2(t) = A_2(t) - A^* \tag{6.7}$$

$\dot{e}(t)$ is expressed in terms of e,ϕ_1,ϕ_2 as:

$$\begin{aligned}\dot{e} &= A_1(t)x_1(t) - A_2(t)x_2(t) + (A^*x_1(t) - A^*x_1(t)) + (A^*x_2(t) - A^*x_2(t)) \\ &= A^*(x_1(t) - x_2(t)) + x_1(t)(A_1(t) - A^*) - x_2(t)(A_2(t) - A^*) \\ &= A^*e(t) + \phi_1(t)x_1(t) - \phi_2(t)x_2(t)\end{aligned} \tag{6.8}$$

Let a candidate Lyapunov function be defined as:

$$V(e, \phi_1, \phi_2) = \frac{1}{2}(e^T Pe + tr(\phi_1^T P\phi_1 + \phi_2^T P\phi_2))$$

It is observed that $V(0, 0, 0) = 0$ and $V(\infty)$ is radially unbounded.

$$\begin{aligned}\dot{V} &= \frac{1}{2}[\dot{e}^T(t)Pe(t) + e^T(t)P\dot{e}(t) + 2Tr(\dot{\phi_1}^T(t)P\phi_1(t) + \dot{\phi_2}^T(t)P\phi_2(t))] \\ &= \frac{1}{2}[(A^*e(t) + \phi_1(t)x_1(t) - \phi_2(t)x_2(t))^T Pe(t) + e^T(t)P(A^*e(t) + \phi_1(t)x_1(t) - \phi_2(t)x_2(t)) \\ &+ 2Tr(\dot{\phi_1}^T(t)P\phi_1(t) + \dot{\phi_2}^T(t)P\phi_2(t))] \\ &= \frac{1}{2}[e^T(t)(A^{*T}P + PA^*)e(t) + x_1^T(t)\phi_1^T(t)Pe(t) + e^T(t)Px_1(t)\phi_1(t) - \phi_2^T(t)x_2^T(t)Pe(t) \\ &- e^T(t)P\phi_2(t)x_2(t) + 2Tr(\dot{\phi_1}^T(t)P\phi_1(t) + \dot{\phi_2}^T(t)P\phi_2(t))] \\ &= \frac{1}{2}[-e^T(t)Pe(t) + 2Tr(e^T(t)Px_1(t)\phi_1(t)) - 2Tr(e^T(t)Px_2(t)\phi_2(t)) \\ &+ 2Tr(\dot{\phi_1}^T(t)P\phi_1(t) + \dot{\phi_2}^T(t)P\phi_2(t))] \\ &= \frac{1}{2}(-e^T(t)Pe(t)) + Tr(e^T(t)Px_1(t)\phi_1(t) + \dot{\phi_1}^T(t)P\phi_1(t)) \\ &+ Tr(-e^T(t)Px_2(t)\phi_2(t) + \dot{\phi_2}^T(t)P\phi_2(t))\end{aligned} \tag{6.9}$$

The adaptive laws are derived as:

$$\dot{\phi_1}^T = -e^T x_1(t)$$

$$\dot{\phi}_2^{\,T} = e^T x_2(t) \tag{6.10}$$

Therefore,

$$\dot{V} = -\frac{1}{2}(e^T(t)Pe(t)) \leq 0$$

As $\dot{e}$ is bounded, from Barbalat's lemma

$$\lim_{t->\infty} e(t) = 0$$

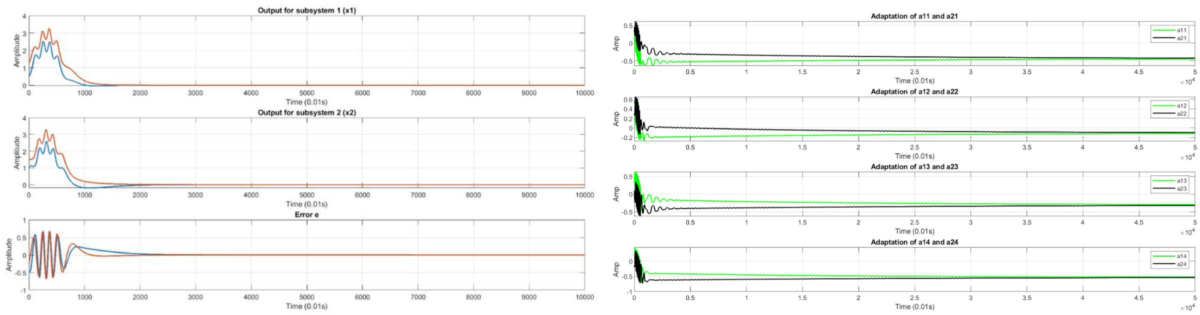
6.3 Simulation Studies

In this section, we perform the mutual adaptation problem for higher order systems. We see that as the system order increases, the likelihood of the system being unstable also increases. Thus we perform a few simulations on second order and third systems to get an understanding of mutual adaptation extended to higher order cases.

6.3.1 Second Order Systems

<u>*Simulation 6.3.1:*</u> *Case of Non-Persistent Input*

We take the input $u(t) = 0$ and initial conditions $x_{11}(0) = 0.5$, $x_{12}(0) = 1.25$,$x_{21}(0) = 1$,$x_{22}(0) = 1.5$ and initial parameter values $a_{11}(0) = 0.1$, $a_{12}(0) = 0.5$, $a_{13}(0) = 0.3$, $a_{14}(0) = 0.4$, $a_{21}(0) = 0.45$, $a_{22}(0) = 0.3$, $a_{23}(0) = 0.1$, $a_{24}(0) = 0.15$



(a) error $e(t)$ and States $x_1(t)$, $x_2(t)$ (b) Parameters $A_1(t)$ and $A_2(t)$

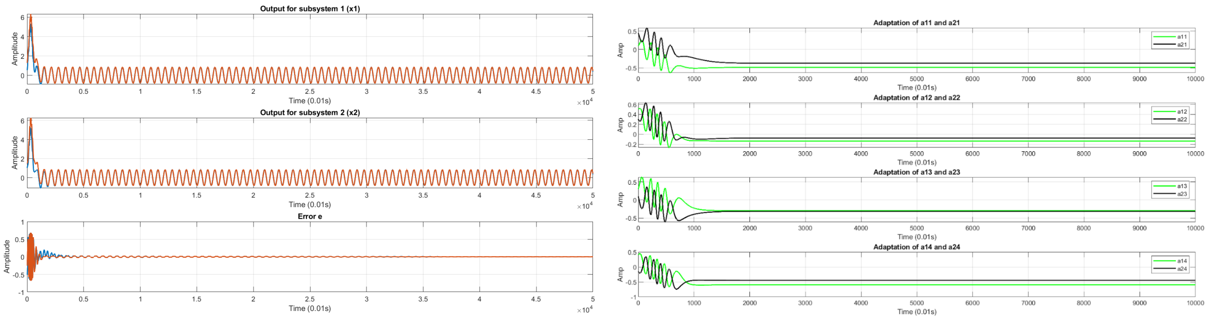
Figure 6.2: $u(t) = 0$

Insights:

1) Final values of matrices $A_1(t)$ and $A_2(t)$ stabilise to distinct values.
2) Error of system dies down to zero.
3) The parameters oscillate indicating adaptation.

Observation:
The parameter values in matrices A_1 and A_2 stabilize to distinct values in the case of a non persistent input u(t).

<u>*Simulation 6.3.2:*</u> *Case of Persistent Input*
We take the input $u(t) = sin(t)$ and initial conditions $x_{11}(0) = 0.5$, $x_{12}(0) = 1.25$, $x_{21}(0) = 1$, $x_{22}(0) = 1.5$ and initial parameter values $a_{11}(0) = 0.1$, $a_{12}(0) = 0.5$, $a_{13}(0) = 0.3$, $a_{14}(0) = 0.4$, $a_{21}(0) = 0.45$, $a_{22}(0) = 0.3$, $a_{23}(0) = 0.1$, $a_{24}(0) = 0.15$



(a) error $e(t)$ and States $x_1(t)$, $x_2(t)$ (b) Parameters $A_1(t)$ and $A_2(t)$

Figure 6.3: $u(t) = sin(t)$

Insights:
1)Final values $a_{1i}(t)$ and $a_{2j}(t)$ converge to the same value when $i = j$
2) Error of system dies down to zero.
3) The parameters oscillate indicating adaptation.
4) The parameters take longer duration to converge to their respective values.

Observation:
The corresponding parameter values in matrices A_1 and A_2 converge to a single values in the case of a persistent input u(t).

6.3.2 Third Order Systems

<u>*Simulation 6.3.3:*</u> *Case of Third Order System*
We take the input $u(t) = sin(t)$ and initial conditions $x_1(0) = [0.9; 1.5; 1.85]$ and $x_2(0) = [0.5; 1.25; 1.75]$ and initial parameter values $A_1(0) = [0.1; 0.2; 0.3; 0.4; 0.5; 0.6; 0.4; 0.7; 0.8]$ and $A_2(0) = [1.2; 1.4; 1.6; 1.2; 1.4; 1.6; 1.8; 1.2; 1.4]$

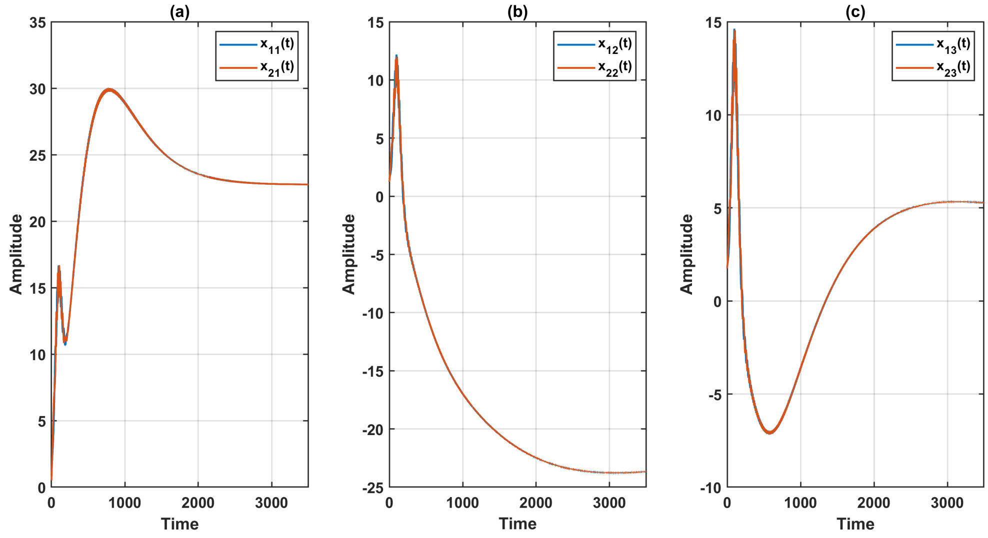


Figure 6.4: Output States of Third Order System

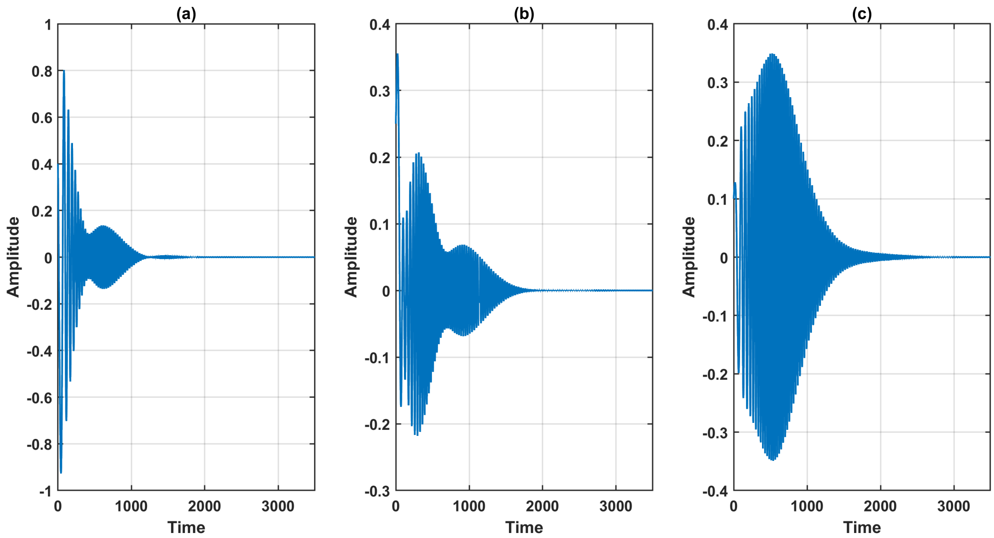


Figure 6.5: Error of Third Order System

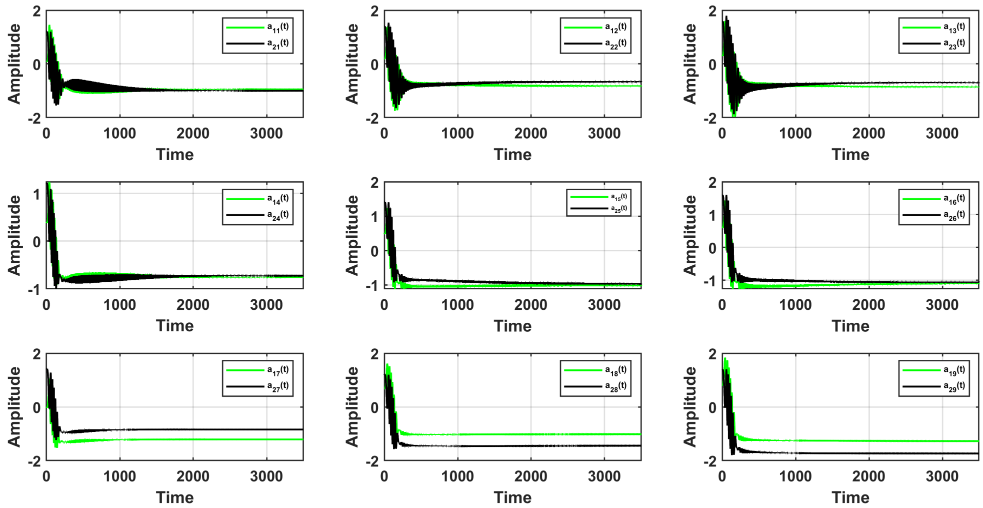


Figure 6.6: Parameter matrices $A_1(t)$ and $A_2(t)$

Insights:

1) We see the corresponding parameters of $A_1(t)$ and $A_2(t)$ of the systems adapt to stable values.
2) The outputs of the systems stabilize to distinct values.
3) The parameters oscillate indicating adaptation.
4) We see that the system is highly unstable in most cases.

Observation:

The mutual adaptation between two third order systems is highly unstable but works for certain initial conditions.

6.4 Summary

This chapter examines mutual adaptation for higher order systems. The mathematical analysis for the nth order mutual adaptation problem is analyzed and the various characteristics of higher order systems undergoing mutual adaptation are investigated via numerous simulations.

Chapter 7

Applications of Mutual Adaptation

Mutual adaptation has many applications across diverse fields. In this section we will explore its role in behavioural synchrony of physical systems.

7.1 Behavioural Synchrony

Behavioural synchrony is a feature of human cultural practices that results in physically keeping together in time with others. Examples of such synchrony include music, dancing and collective rituals which result in humans to operate in unison. Achieving synchronisation in engineered systems is critical in several applications that include synchronisation of clocks in transportation and communication. Here, we deal with achieving behavioural synchrony between dynamical systems through mutual adaptation. This intriguing problem appears to have extensive implications across various fields, owing to the lack of available information it takes into consideration. In this chapter is to exhibit synchrony between the states of two dynamical systems using the results of mutual adaptation. We assume that the parameters of the representations are unknown even though we have information regarding the structure of the systems. Previously, synchronisation of systems has been achieved via methods such as sample position data control [21], adaptation with a reference model [22] and nonlinear active control [23] among others. Here, the two systems are assumed to have parameters that are tunable. Adaptive laws that achieve synchronisation do not require a reference model. The manner in which the mutual adaptation problem is solved ensures that the error between the states of the two systems converge to zero thereby ensuring a consensus on their behaviour.

7.2 RC Circuit

The RC circuit, also known as the RC filter, is a circuit with a resistor and capacitor connected in series.

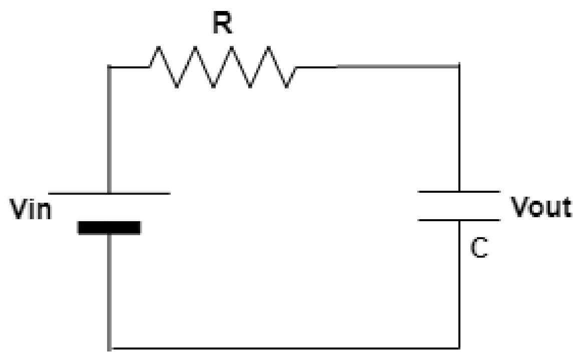


Figure 7.1: RC Circuit

The system under consideration is a first order series RC circuit with the transfer function:

$$\frac{V_{out}}{V_{in}} = \frac{\frac{1}{RC}}{s + \frac{1}{RC}} \tag{7.1}$$

The state space representation is as follows:

$$\dot{x}(t) = \left(\frac{-1}{RC}\right) x(t) + u(t) \tag{7.2}$$

The input to the system is the initial voltage V_{in} and the output of interest is voltage across the capacitor V_{out}. The parameter of concern is $-1/RC$, where $T = RC$ is the time constant of the circuit. Therefore, the time constants of the respective systems update to new values when the systems interact with each other.

Simulation 7.2.1: Mutual Adaptation between two RC Filters

The specifications of the two systems are tabulated in Table 7.1. The initial state conditions are $x_1(0) = 2$V and $x_2(0) = 3$V. The initial parameter values are $a_1(t_0) = a_1(0) = -0.1$ and $a_2(t_0) = a_2(0) = -0.133$. The subsystems are excited with an input $u(t) = 10$V. Adapting $a_1(t)$ and $a_2(t)$ according to Eq.(5.5), we see that they stabilise to negative values of approximately -0.7178 as shown in Fig. 7.2(b). Hence, both the subsystem are stable. We can observe oscillations between parameter values are more as they try to adapt and converge to the same value. The voltage across the two capacitors

Table 7.1: Specifications for Simulation 7.2.1

Parameters	System 1	System 2
Resistance	1 KΩ	7.5 KΩ
Capacitance	$10mF$	$1mF$
Time Constant RC	10	7.5

of the subsystems converge to 14V (Fig.7.2(a)). This is also evident from the error plot shown in (Fig. 7.3).

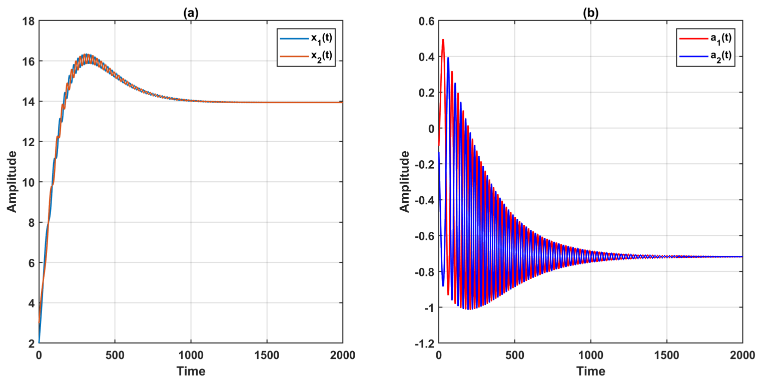


Figure 7.2: Mutual Adaptation of RC Circuits

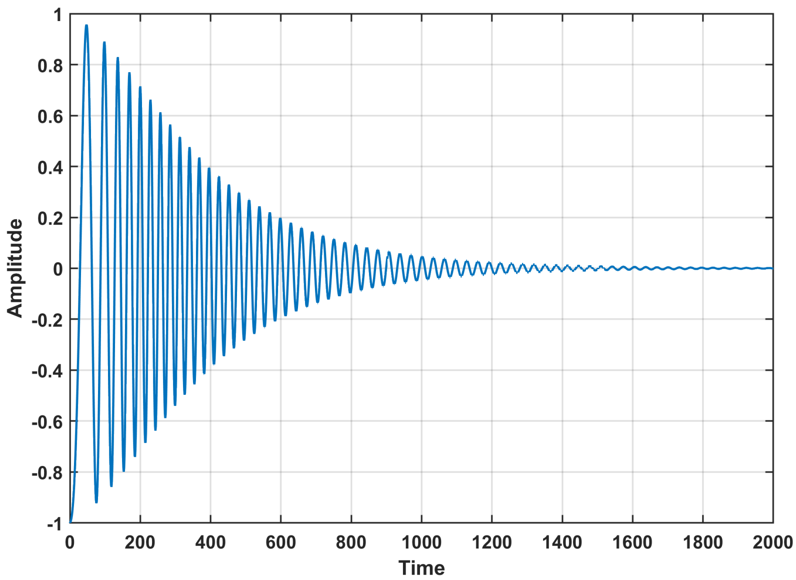


Figure 7.3: The error $e(t)$

7.3 Pendulum

A simple pendulum is a mechanical set up consisting of a bob of mass M suspended from a massless string or rod of length L.

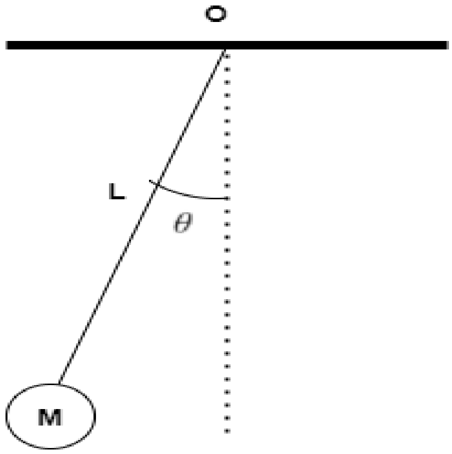


Figure 7.4: Simple Pendulum

The system under consideration is linearised around the origin $(0, 0)$. This results in a second order system with the following transfer function

$$\frac{\Theta(s)}{T(s)} = \frac{\frac{1}{ML^2}}{s^2 + \frac{B}{ML^2}s + \frac{g}{L}}$$

where B is viscous friction and assumed to be 1 and g is the acceleration due to gravity. The state space representation is as follows

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -g/L & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1/ML^2 \end{bmatrix} u(t)$$

where first state $x_1(t)$ is the angular position and second state $x_2(t)$ is the angular velocity for both the subsystems Σ_1 and Σ_2. The input to the system is torque $u(t) = 5 \sin t$.

Simulation 7.3.1: Mutual Adaptation between two Simple Pendulums

The specifications of the subsystems are tabulated in Table 7.2. From these specifications, we derive the initial parameter values of the respective subsystems as

$$A_1(t_0) = A_1(0) = \begin{bmatrix} 0 & 1 \\ -4.91 & 0 \end{bmatrix}$$

$$A_2(t_0) = A_2(0) = \begin{bmatrix} 0 & 1 \\ -1.23 & 0 \end{bmatrix}$$

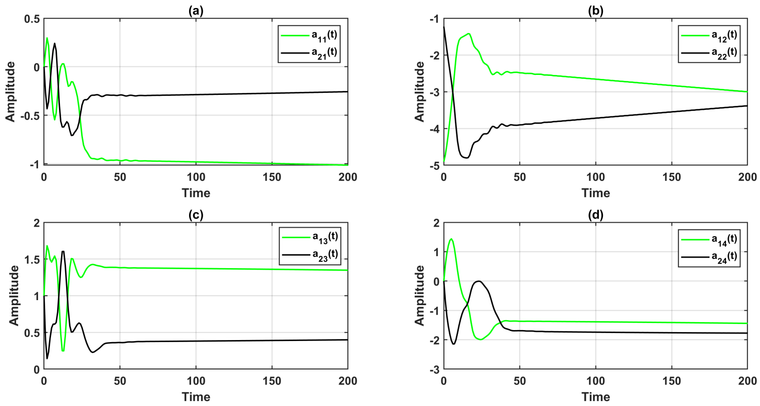


Figure 7.5: Adaptation of $A_1(t)$ and $A_2(t)$

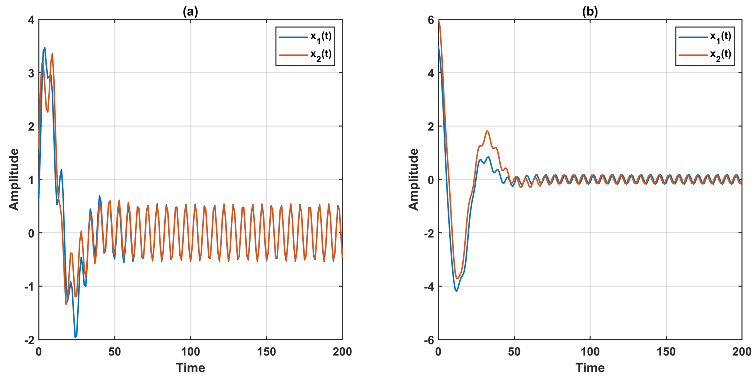


Figure 7.6: Evolutions of the states of Simple Pendulum

The initial state conditions are $x_1^T(0) = [0.2\pi\ 5]$ and $x_2^T(0) = [0.5\pi\ 6]$. We adapt $A_1(t)$ and $A_2(t)$ according to Eq.(6.7). On adaptation, we see the corresponding values in the matrices $A_1(t)$ and $A_2(t)$ adapt to stable values as depicted in Fig. 7.5(a) - (d). It can be observed that apart from $a_{11}(t)$ and $a_{21}(t)$, all the parameters of $A_1(t)$ and $A_2(t)$ do not converge to same value. The angular position and angular velocity of both subsystems oscillates around zero as shown in Fig. 7.6(a) and (b) respectively. That is also evident from error plot as in Fig. 7.7(a) and (b). Here, Fig. 7.7(a) and (b) respectively represents the error between the angular positions and angular velocity of the two subsystems. Moreover, we see the eigen values of $A_1(\infty)$ and $A_2(\infty)$ are $[-1.2288 + 1.9999i, -1.2288 - 1.9999i]$ and $[-1.0151 + 0.8803i, -1.0151 - 0.8803i]$ respectively. The eigen values are all negative, showing that the overall system is stable after adaptation.

Table 7.2: Specifications for Simulation 7.3.1

Parameters	System 1	System 2
Mass	1 Kg	2 Kg
Length	2 m	8 m
Gravity	9.8 m/s^2	9.8 m/s^2

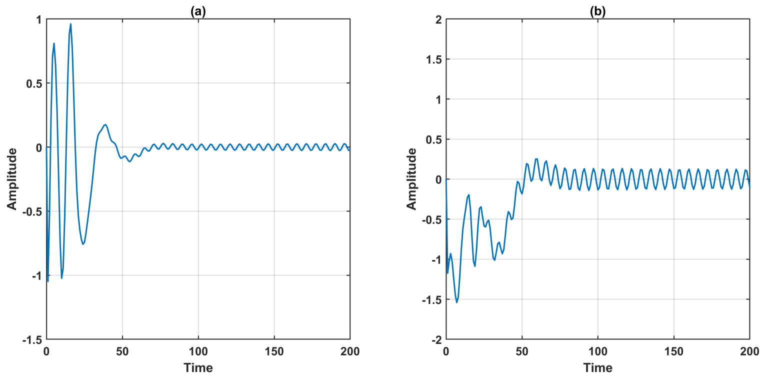


Figure 7.7: The error $e_1(t)$ and $e_2(t)$

7.4 DC Motor

7.4.1 The Second Order Case

A DC motor converts electrical current to mechanical energy by the process of electromagnetic induction. A motor comprises of the stator, which is a permanent magnet and a turning coil of wire called the armature. Consider a series DC motor system.

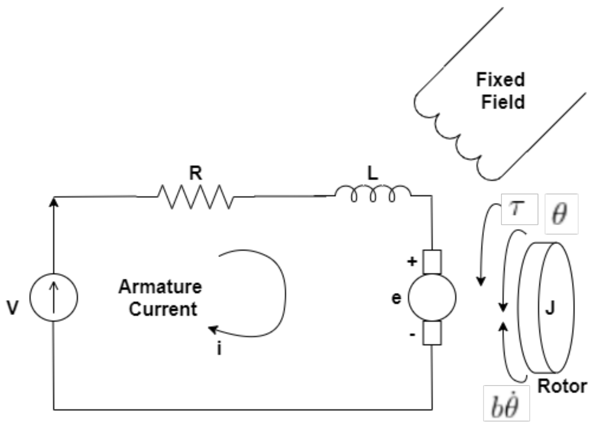


Figure 7.8: A Series DC Motor

It is a second order system with the transfer function:

$$\frac{\Phi(s)}{V(s)} = \frac{K}{(Js+b)(Ls+R)+K^2}$$

where the input to the system is the voltage of the motor and the output of the system is the rotational speed of the motor. The state space representation for the system is

$$\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \end{bmatrix} = \begin{bmatrix} -\frac{b}{J} & \frac{K}{J} \\ -\frac{K}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u(t)$$

where rotational speed and the armature current are the states $x_1(t)$ and $x_2(t)$ of the DC motor respectively. The parameters of the system are calculated based on the moment of inertia of the rotor (J), the motor viscous force (b), the motor torque constant (K), the electrical resistance (R) and the electrical inductance (L).

Simulation 7.4.1: Mutual Adaptation between two DC Motors

The specifications for the two subsystems are given in Table 7.3. An input voltage $u(t) = 70V$ is supplied to both the subsystems. The initial conditions of subsystem parameters are

$$A_1(0) = \begin{bmatrix} -10 & 4.8 \\ -0.63 & -6 \end{bmatrix} \text{and } A_2(0) = \begin{bmatrix} -8.57 & 2.67 \\ -1.43 & -5.33 \end{bmatrix}$$

The initial state conditions are $x_1^T(0) = [5\ 15]$ and $x_2^T(0) = [6\ 12]$. We see that the respective entries of the $A_1(t)$ and $A_2(t)$ adapt to stable values (refer Fig. 7.9(a) - (d)). It is seen that the states of the subsystems stabilise and the error between the states converge to zero as shown in Fig. 7.10 and Fig. 7.11 respectively. The speeds of the two DC motors are initially at $15rad/s$ and $16rad/s$. They eventually converge to $10.86rad/s$ as evident from Fig. 7.10(a). Similarly, the armature currents of the two DC motors are initially $5A$ and $6A$ respectively. They stabilise to $4.34A$ (refer (Fig. 7.10(b)). Finally, the eigen values of $A_1(\infty)$ and $A_2(\infty)$ are $[-9.6620, -6.8869]$ and $[-7.0854 + 1.8590i, -7.0854 - 1.8590i]$ respectively. As they are all negative, the overall system is stable after adaptation.

Table 7.3: Specifications for Simulation 7.4.1

Parameters	System 1	System 2
Resistance	3.0 Ω	4.0 Ω
Inductance	0.5 H	0.75 H
Torque Const.	0.3 Nm/A	0.5 Nm/A
Back EMF Const.	0.1 V/Rad/s	0.4 V/Rad/s
Friction Constant	0.22 Nms	0.6 Nms
Moment of Inertia	0.02 Kgm^2/s^2	0.01 Kgm^2/s^2

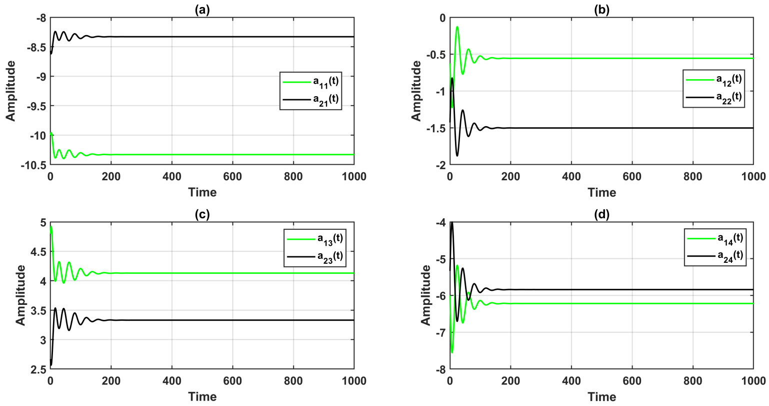


Figure 7.9: Adaptation of $A_1(t)$ and $A_2(t)$

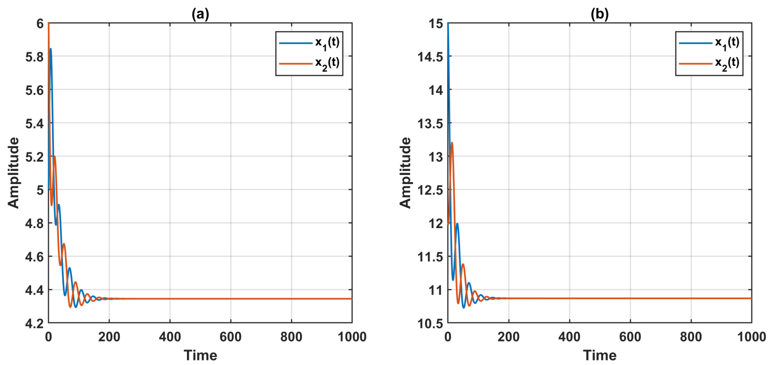


Figure 7.10: Adaptation of Outputs of DC Motor

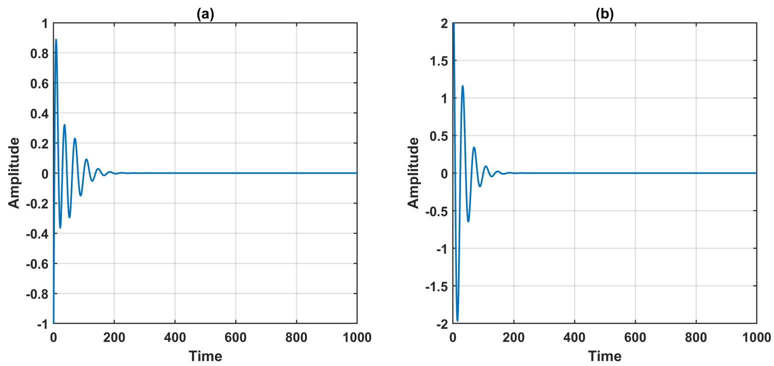


Figure 7.11: The error $e_1(t)$ and $e_2(t)$

7.4.2 The Third Order Case

In the previous section, we considered the simulation of two second order DC motors mutually adapting. The speed and armature current were the state variables of interest. If we consider the angle of rotation of the output shaft as well, the DC motor will be a third order system. The state space representation for the third order DC motor is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -R_a/L_a & -k_b/L_a & 0 \\ -k_b/J_m & -B_m/J_m & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1/L_a \\ 0 \\ 0 \end{bmatrix} u(t)$$

where R_a/L_a is the time constant of the DC motor, k_b is the back EMF constant, J_m is the rotor inertia and B_m is the frictional coefficient. Here, x_1 refers to the armature current, x_2 refers to angular speed and x_3 refers to the angle of rotation of the output shaft.

Simulation 7.4.2: Extension of Mutual Adaptation between two DC Motors
The specifications for the two systems are given in Table 7.4. The input $u(t) = 10V$ is applied to the subsystems. The initial parameter values are

$$A_1(0) = \begin{bmatrix} -1.2 & -1.6 & 0 \\ 1.1 & -0.6 & 0 \\ 0 & 1 & 0 \end{bmatrix} \text{ and } A_2(0) = \begin{bmatrix} -0.1 & -0.4 & 0 \\ 0.2 & -0.5 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

The initial state conditions are $x_1^T(0) = [0.5\ 1.25\ 1.75]$ and $x_2^T(0) = [0.9\ 1.5\ 1.85]$. It is evident form Fig. 7.13(a) that armature currents of two DC motors stabilise to $2.96A$. Similar trends can be seen for other two states of the subsystems. The speeds which were initially at $1.25rad/s$ and $1.5rad/s$ converge to $2.628rad/s$ (refer Fig. 7.13(b)).

Table 7.4: Specifications for Simulation 7.4.2

Parameters	System 1	System 2
Resistance R_a	1.2 Ω	0.2 Ω
Inductance L_a	1 H	1 H
Torque Const.	0.5 Nm/A	0.3 Nm/A
Back EMF k_b	0.4 V/Rad/s	0.1 V/Rad/s
Friction Constant B_m	0.6 Nms	0.2 Nms
Moment of Inertia J_m	1.4286 Kgm^2/s^2	2 Kgm^2/s^2

The angles of rotation adapt from the values $1.75rads$ and $1.85rads$ to same value of $13.64rads$ as shown in Fig. 7.13(c). Therefore, the errors between the respective states die down to zero as seen in Fig. 7.14. The eigen values of $A_1(\infty)$ and $A_2(\infty)$ are $[-0.89+1.23i\ -0.89-1.23i\ -0.35]$ and $[-0.87\ -0.29+0.39i\ -0.29-0.39i]$ respectively. Hence, overall system is stable after adapting. It can be observed that third order systems are highly unstable, owing to the large number of parameters that must interact in order for the subsystems to reach a common consensus.

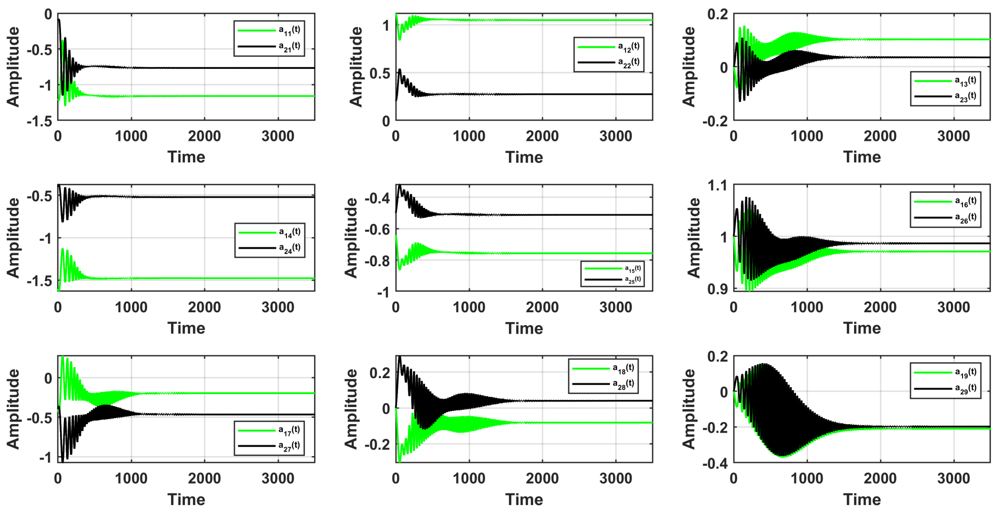


Figure 7.12: Adaptation of $A_1(t)$ and $A_2(t)$

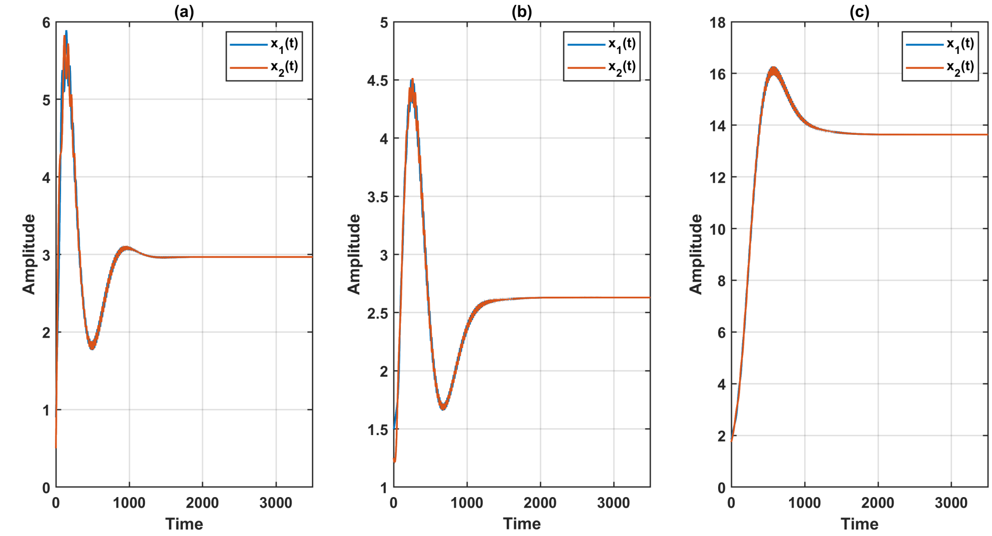


Figure 7.13: Evolution of states of DC Motor

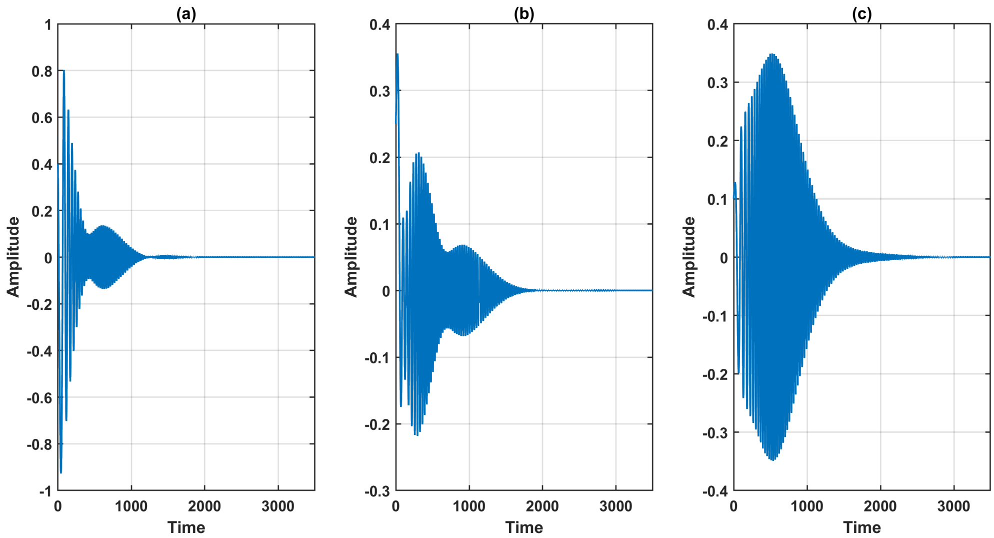


Figure 7.14: The error signals $e_1(t)$, $e_2(t)$ and $e_3(t)$

7.5 Summary

We illustrate in this chapter the achievement of behavioural synchrony with four dynamical systems, which is an important application of mutual adaptation. We consider firstorder, second-order and third-order systems. These models are inspired by three physical systems, namely an RC filter, a simple pendulum, and a DC motor.

Chapter 8

Limitations and Improvements on Mutual Adaptation

In this chapter, we discuss the case where mutual adaption seems to fail for higher order case while using the standard adaptive laws. These problems were noted studies of [2,3]. These interesting problems seem to give way for further research to be conducted in ways of modifying the adaptive laws for better performance of the systems. Two such approaches are briefly explained in subsection 8.2, which can be further investigated in the future.

8.1 When Adaptive Stabilisation Fails

This section discusses two scenarios where mutual adaptation between two nth order systems fails.

8.1.1 The Orthogonality Problem

Consider two systems $\sum_1$ and $\sum_2$ described by the following equations:

$$\sum_1 : \dot{x} = A_1 x$$

$$\sum_2 : \dot{z} = A_2 z \tag{8.1}$$

Where $x(0) = x_0$ and $z(0) = z_0$.

The error differential equation is given as:

$$\dot{e(t)} = A^* e(t) + b[\theta_1^T(t)x(t) - \theta_2^T(t)z(t)] \tag{8.2}$$

Where $A_1(t) = A + b\theta_1(t)^T$ and $A_2(t) = A + b\theta_2(t)^T$ for any constant matrix A^*.

When the two subsystems mutually adapt using the adaptive laws $\dot{\theta}_1 = -e^T Pbx(t)$ and $\dot{\theta}_2 = e^T Pbz(t)$, the vectors $e(t)$, θ_1 and θ_2 are bounded. Here, $e(t)$ dies down to zero. This implies that $\lim_{t\to\infty}[\theta_t^T(t)x(t) - \theta_2^T(t)z(t)] = 0$.

This however doesn't imply that $x(t)$ and $z(t)$ are signals that are bounded. Since $e(t)$ tends to zero, $[\theta_1^T(t) - \theta_2^T(t)]x(t) = [\theta_1^T(t) - \theta_2^T(t)]z(t) = 0$. As only the dot product of both terms becomes zero as the error dies down to zero, if the signal and the parametric difference term are *orthogonal* to each other, to the observer it will appear as though the subsystems tend to stability even if the values of the signals become unbounded.

8.1.2 The Instability Problem

Consider a second order system:

$$\sum_1 : \dot{x} = A_1 x1(t) \sum_2 : \dot{z} = A_2 x2(t) \tag{8.3}$$

Where $x_1(0) = x_{10}$ and $x_2(0) = x_{20}$. If $A_1(t)$ and $A_2(t)$ in 8.3 belong to $R^{2\times 2}$ and undergo adaptation mutually so that the error $e(t)$ between their outputs dies down to zero and the adaptive laws such that output states $x_1(t)$ and $x_2(t)$ grow in an unbounded fashion, $A_1(\infty) = \lim t \to \infty A_1(t)$ and $A_2(\infty) = \lim t \to \infty A_2(t)$ should only have one common eigenvalue which is unstable common between them such that the system is stable.

8.1.3 Adaptive Stabilization When One or Both Systems are Stable

In this section, we consider the following questions: Can two stable systems adapting mutually become unstable? What happens when one of the systems is stable and the other is unstable at time t = 0?
We consider the Candidate Lyapunov function as:

$$V(e, \theta_1, \theta_2) = \frac{1}{2}[e^T Pe + \theta_1^T \theta_2 + \theta_2^T \theta_1] \tag{8.4}$$

Here, $\dot{V} = -e^T Qe \leq 0$. Hence, the system will be stable locally as long as $e(0)$, θ_1, θ_2 are small. The proximity of $A_1(0)$ and $A_2(0)$ to A^* decides the overall system behavior.
Two stable systems with eigen values close to each other at $t(0)$ will have a stable result. However, large initial values of $e(0)$ can result in θ_1 and θ_2 having large values such

that $A_1(t)$ and $A_2(t)$ may have eigen values with positive real parts. By the Instability theorem, this may result in instability.

8.2 Improving Adaptive Stabilisation

The exploration of how higher-order cases of mutual adaptation can improve the stabilisation results of the overall system via modifications of the standard adaptive laws is detailed in this section.

8.2.1 Complementarity

The orthogonality problem occurs when the parametric difference vector $\phi_1(t) - \phi_2(t)$ tends to a value orthogonal to the input vectors $x(t)$ or $z(t)$. This scenario can be avoided by utilising complementary adaptation. Two systems are said to be complementary with respect to one another if they are constrained in a situation such that the conditions for the orthogonality problem cannot be satisfied.

In complementary adaptation, $\sum_1$ and $\sum_2$ use complementary signals and parameters to adapt with respect to each other. In such a scenario, it is assumed that the fixed parameters of the subsystems together form a stable system. Complementary adaptation can be performed by choosing appropriate adaptive parameters or initial conditions for each subsystem. It can also be executed by modifying the adaptive law that is used to facilitate mutual adaptation between the two subsystems.

8.2.2 Periodicity

In order to avoid the orthogonality condition, another possible approach is periodic adaptation. In this mode of adaptation, the two systems alternatively adapt their parameters periodically over a fixed time interval. This means that while ϕ_1 is adapted, ϕ_2 is fixed and vice versa. Thus, periodic adaptation ensures that instability due to the orthogonality problem will be avoided.
As the time period T was varied, it was observed that the subsystems were stable when T was present in the interval and if not, the subsystems were unstable. [2]

8.3 Summary

In this chapter, a few scenarios where mutual adaptation fails have been discussed namely the instability problem and orthogonality Problem. Cases of mutual adaptation where one or both systems are stable have also been stated. Novel approaches to counteract these limitations have also been proposed. These solutions provide alternate methods to perform mutual adaptation between two subsystems.

Chapter 9

Conclusion

This report addresses the complex adaptive problem of stabilization of subsystems in the absence of a stable reference model via mutual adaptation. The concept of mutual adaptation has a plethora of viable applications across various domains with far reaching consequences.This concept has been investigated in fields of economics, psychology, biology and robotics.

Chapters 1-4 extensively investigates the domain of adaptive systems by beginning with a deep delve into the fundamentals of adaptive control. The identification and control problem for simple adaptive systems is examined in chapter 2 and simulations for the same are presented. The successive chapter discusses the need for adaptive observers and controllers. In chapter 5, the mutual adaptation problem is presented and the mathematical basis for mutual adaptation is then derived for first as well as higher order cases. Apart from this, in-depth simulation studies have also been performed for a diverse set of trends in the first order case. The idea of mutual adaptation is then applied to physical systems of various orders in chapter 6. In these applications, mutual adaptation is used to perform behavioural synchrony of the two physical systems with different initial conditions. Here, the two systems are assumed to have parameters that are tunable. Adaptive laws that achieve synchronisation do not require a reference model. The manner in which the mutual adaptation problem is solved ensures that the error between the states of the two systems converge to zero thereby ensuring a consensus on their behaviour.

The report concludes with the limitations of mutual adaptation and suggestions of novel approaches to tackle these limitations in the future namely complementary and periodic adaptation.

Bibliography

[1] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems*. Englewood Cliffs: Prentice Hall, 1990.

[2] K. S. Narendra and P. Harshangi, "Unstable systems stabilizing each other through adaptation," in *2014 American Control Conference*, Portland, OR, USA, June 2014, pp. 7–12.

[3] ——, "Unstable systems stabilizing each other through adaptation - part ii," in *2015 American Control Conference*, Chicago, IL, USA, July 2015, pp. 856–861.

[4] Y. Sonntag, M. Musgaard, C. Olesen, B. Schiott, J. V. Moller, P. Nissen, and L. Thogersen, "Mutual adaptation of a membrane protein and its lipid bilayer during conformational changes," vol. 2, May 2011.

[5] K. Honka and B. Sures, "Mutual adaptations between hosts and parasites determine stress levels in eels," April 2021.

[6] A. Mukherji and J. D. Francis, "Mutual adaptation in buyer - supplier relationships," February 2008.

[7] D. de Lange, D. Armanios, J. Delgado-Ceballos, and S. Sandhu, "From foe to friend: Complex mutual adaptation of multinational corporations and nongovernmental organizations," vol. 55, pp. 1197–1228, February 2016.

[8] M. A. Forrester, "Mutual adaptation in parent-child interaction: learning how to produce questions and answers," vol. 14, pp. 190–211, January 2013.

[9] N. Semmer and U. S. Schallberger, "Selection, socialisation, and mutual adaptation: Resolving discrepancies between people and work," pp. 263–288, 1996.

[10] R. Wilkinson, S. Lock, K. Bryan, and K. Sage, "Interaction-focused intervention for acquired language disorders: Facilitating mutual adaptation in couples where one partner has aphasia," vol. 13, no. 1, pp. 74–87, 2011.

[11] W. Jiao and Y. Sun, "Self-adaptation of multi-agent systems in dynamic environments based on experience exchanges," vol. 122, no. C, 2016. [Online]. Available: https://doi.org/10.1016/j.jss.2016.09.025

[12] T. Obo, C. K. Loo, and N. Kubota, "Mutual adaptation in neuro fuzzy system for human posture recognition," in *2015 10th Asian Control Conference (ASCC)*, 2015, pp. 1–6.

[13] S. Ikemoto, H. B. Amo, T. Minato, B. Jung, and H. Ishiguro, "Physical human-robot interaction: Mutual learning and adaptation," vol. 19, no. 4, pp. 24–35, Dec 2012.

[14] T. Obo, C. K. Loo, and N. Kubota, "Mutual adaptation in neuro fuzzy system for human posture recognition," in *Asian Control Conference*, 2015.

[15] K. S. Narendra and S. Mukhopadhya, "Mutual learning: Part i – learning automata," in *Asian Control Conference*, Philadelphia, PA, USA, July 2019, pp. 916–921.

[16] D. Komagome, M. Suzuki, T. Ono, and S. Yamada, "Robotmeme - a proposal of human-robot mimetic mutual adaptation-," in *16th IEEE International Symposium on Robot and Human Interactive Communication*, 2007.

[17] S. Nikolaidis, A. Kuznetsov, D. Hsu, and S. Srinivasa, "Formalizing human-robot mutual adaptation: A bounded memory model," in *11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2016.

[18] S. Nikolaidis, D. Hsu, and S. Srinivasa, "Human-robot mutual adaptation in collaborative tasks: Models and experiments," vol. 36, no. 5-7, pp. 618–634, 2017.

[19] F. Ducatelle, G. A. D. Caro, and L. M. Gambardella, "Cooperative stigmergic navigation in a heterogeneous robotic swarm," in *Lecture Notes in Computer Science*, vol. 6226, 2010, pp. 607–617.

[20] M. Gopal, *Modern Control System Theory.* USA: Halsted Press, 1984.

[21] Q. Song, F. Liu, G. Wen, J. Cao, and Y. Tang, "Synchronization of coupled harmonic oscillators via sampled position data control," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 7, pp. 1079–1088, 2016.

[22] D. Z. Zhao, C. W. Li, and J. Ren, "Speed synchronisation of multiple induction motors with adjacent cross-coupling control," vol. 4, no. 1, pp. 119–128, 2010.

[23] N. Griba, F. Hamidi, K. Menighed, B. Boussaid, and M. N. Abdelkrim, "Synchronization of chaotic systems: a survey study," in *International Conference on Signal, Control and Communication (SCC)*, 2019, pp. 262–267.

[24] K. S. Narendra and S. Mukhopadhyay, "Mutual learning: Part ii –reinforcement learning," in *2020 American Control Conference (ACC)*, 2020, pp. 1105–1110.

[25] S. Ezzina, M. Scotti, C. RouSme, S. Pla, H. Blain, and D. Deligniéres, "Interpersonal synchronization processes in discrete and continuous tasks," pp. 1–15, 2020.

[26] B. Anderson, "Exponential stability of linear equations arising in adaptive identification," *IEEE Transactions on Automatic Control*, vol. 22, no. 1, pp. 83–88, 1977.

[27] R. Bitmead, "Persistence of excitation conditions and the convergence of adaptive schemes," *IEEE Transactions on Information Theory*, vol. 30, no. 2, pp. 183–191, 1984.

[28] Y. Miyake and J. Tanaka, "Mutual adaptation in human-robot cooperative walk-mutual-entrainment-based internal control," in *Proceedings 6th IEEE International Workshop on Robot and Human Communication. RO-MAN'97 SENDAI*, 1997, pp. 124–129.

[29] Y. Miyake, T. Miyagawa, and Y. Tamura, "Internal observation and mutual adaptation in human-robot cooperation," in *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.98CH36218)*, vol. 4, 1998, pp. 3685–3690 vol.4.

[30] "Necessary and sufficient conditions for parameter convergence in adaptive control," *Automatica*, vol. 22, no. 6, pp. 629–639, 1986.

[31] R. Carroll and D. Lindorff, "An adaptive observer for single-input single-output linear systems," *IEEE Transactions on Automatic Control*, vol. 18, no. 5, pp. 428–435, 1973.

[32] K. Narendra and P. Kudva, "Identification and adaptation using lyaponov's direct method," in *1973 IEEE Conference on Decision and Control including the 12th Symposium on Adaptive Processes*, 1973, pp. 521–527.

[33] G. Goodwin and E. K. Teoh, "Persistency of excitation in the presence of possibly unbounded signals," *IEEE Transactions on Automatic Control*, vol. 30, no. 6, pp. 595–597, 1985.

[34] M. Benosman, "Model-based vs data-driven adaptive control: An overview," *International Journal of Adaptive Control and Signal Processing*, vol. 32, 03 2018.