Course Conclusion Paper - DRAFT

DYNAMICAL MODELLING AND CONTROL OF CHEMICAL REACTIVE SYSTEMS

Otacílio Bezerra Leite Neto
Federal University of Ceará
Department of Teleinformatics Engineering
minhotmog@gmail.com

Contents

1 Introduction **1**
- 1.1 Control Systems Engineering . . . 1
- 1.2 Chemical Reactive Systems . . . 2
- 1.3 Motivation . . . 3
- 1.4 Objectives . . . 4
- 1.5 Chapters Organization . . . 4

2 Dynamical System Analysis **5**
- 2.1 Model from First Principles . . . 5
- 2.2 Mathematical Models of Systems . . . 9
- 2.3 Response Analysis in Time Domain . . . 14
- 2.4 Similarity Transformations . . . 21
- 2.5 Stability, Controlability and Observability . . . 25
- 2.6 Response Analysis in the Frequency Domain . . . 28

3 Controller Synthesis **31**
- 3.1 State Feedback Controllers . . . 31
- 3.2 Regulation and Reference Tracking . . . 34
- 3.3 Deterministic State Observers . . . 38
- 3.4 Properties of State-Feedback Controllers . . . 42

4 Optimal Control and Estimation **45**
- 4.1 General Formulation . . . 45
- 4.2 Linear Quadratic Regulator (LQR) . . . 48
- 4.3 Optimal State Estimators . . . 52
- 4.4 Linear Quadratic Gaussian (LQG) . . . 57
- 4.5 Stability and Robustness Analysis . . . 58

5 Receding Horizon Optimal Control **63**
- 5.1 Optimization in Moving Horizons . . . 63
- 5.2 Linear Parameter-Varying Models . . . 64
- 5.3 Receding Horizon Control for Switched Systems . . . 66

6 Methodology **67**
- 6.1 Non-Isothermal Continuous Stirred Tank . . . 67

7 Results and Discussion **69**

8 Conclusion **71**

Bibliography **72**

Appendix A - Proof of Theorems **74**

Chapter 1

Introduction

This chapter presents the main problem in discussion and the basic concepts concerning its formulation and solutions, which are detailed further in the next chapters. This is a work on Control Theory and its application to Chemical Reactive Systems, therefore the discussion will follow the notation common to the literature of this field, and a "modern" approach to this theory is explored.

The sections are organized as follows: Section 1.1 provides general definitions for control systems engineering, Section 1.2 discuss chemical reactive systems and its importance in both industry and academia, Sections 1.3 and 1.4 describes the motivation and justification of this work, respectively, and Section 1.5 details the subsequent chapters in this document.

1.1 Control Systems Engineering

The discipline of Control Systems Engineering deals with the design of devices, named *controllers*, that are integrated to a physical system (a *dynamical system*, in most cases) in order to impose a desired behavior to this system. To achieve this goal, the discipline covers topics ranging from applied mathematics, such as dynamical systems theory and signal processing, to a more engineering discussion, regarding instrumentation and implementation of these controllers in a real-life plant or individual system.

A system, in a broad physical sense, is defined as a ensemble of interacting components that responds to external stimuli producing a determined dynamical response, and whose individual parts are not able to produce the same functionality by their own. Thus, the first essential element in Control Theory is a mathematical model of the system of interest. One such model is the *Input-Output Representation*, as shown in **fig!!!**, in which an input stimuli, a signal $u(t)$, acts on the system producing an output response, a signal $y(t)$, described by the following differential equation:

$$\alpha_n \frac{d^n y(t)}{dt^n} + \alpha_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \cdots + \alpha_1 \frac{dy(t)}{dt} + \alpha_0 y(t) = \beta_m \frac{d^m u(t)}{dt^m} + \beta_{m-1} \frac{d^{m-1} u(t)}{dt^{m-1}} + \cdots + \beta_1 \frac{du(t)}{dt} + \beta_0 u(t) \quad . \tag{1.1}$$

[fig]

In this representation, the input $u(t)$ is called the *manipulated variable*, since it represents a arbitrary stimuli that can be given directly by human action or a by an automatic controller, while the output $y(t)$ is called the *controlled variable*, since it can only be modified indirectly through $u(t)$. This also leads to a *cause-and-effect* interpretation of the system.

A model can provide a quantitative understanding of the system that is useful both to access some response specifications and to design controllers to modify them based on some requirements. In the case of the model in Equation (1.1), it is possible to calculate the response

$y(t)$, and its derivatives, resulting from any specific action $u(t)$. Besides, a model can be used to perform computer simulations, in order to visualize the dynamical behavior of the system without actually manipulating it, since real experiments could be expensive or even damage the system. Consider, for instance, a schematic and a simulation for a model representing a mass-spring-damper system, shown at **!!!!**.

[fig]

In this simulation, the *rise time*, *peak time*, *overshoot ratio* and *steady-state value* are examples of response specifications that can be defined to describe the system behavior to a external stimuli (in this case, a constant force of unit magnitude). These specified parameters are characteristic to responses of a class of systems known as *underdamped second-order systems*, that will be discussed further in the document.

A controller is used to calculate, for a time $t \in [t_0, t_N)$, the necessary input $u(t)$ to produce an output $y(t)$ as close as possible to a desired reference signal $r(t)$. There are two common configurations, shown in **Fig!!!**, of how to connect the controller to the system.

[fig]

The configuration in **Fig!!!.a**, known as *Open-Loop Controller*, calculates the action as a function $u(t) = \pi(r, t)$, given an initial condition $y(t_0) = y_0$. In this case, the controller does not observe the output $y(t)$, and relies on the model to guarantee that the system is driven to the reference. Of course, if there are any external disturbances acting on this configuration, or if the model is not reliable enough, it is not possible to guarantee that the requirements are met. Thus, these type of controllers are not suitable for critical applications, and its use is restricted to systems where deviance from the desired reference can be tolerated. [**exemplos de aplicações?**]

In contrast, the configuration in **Fig!!!.b**, known as *Closed-Loop Controller* or *Feedback Controller*, calculates the action as a function $u(t) = \pi(e, t)$, where $e(t) = r(t) - y(t)$ is the error between the reference and the actual response. Now, the controller will observe the system output, trough some sensor device, and compares it to the desired reference in order to calculate a *corrective action*. This feedback property can make the system reject disturbances while still driving it to the desired reference. Thus, the Feedback Controller became the most popular choice of controller configuration in industry for a wide range of applications, even for critical ones. [**exemplos de aplicações? + references**]

1.2 Chemical Reactive Systems

A chemical reaction, the transformation of a chemical substance into another, is a process central to chemistry and to nature itself. A reaction equation is a intuitive representation of such transformations. For instance, consider the following equation representing a *synthesis reaction*:

$$A + 2B \longrightarrow 3C + D. \tag{1.2}$$

In this equation, the compounds A and $2B$ forms the set of *reactants*, $\mathbb{R}$, while $3C$ and D forms the set of *products*, $\mathcal{P}$. The coefficients in such equations are the *stoichiometric numbers*, providing an information about proportionality between the quantity of each substances in the reaction.

Usually the products can be directly used as reactants in another reaction, in which case they can also be referred as a intermediate product (or byproduct), and the equations can be appended in a "series" representation. In this case, each k-th intermediate product forms a set $\mathcal{I}_k$. In addition to a chain of series reactions, there is also the possibility of different reactions to occur in parallel, in the same system. The combination of these sets of reactants, byproducts, products and reactions are often referred as a *chemical reaction network*, and the associated

equation can be represented in general form as:

$$\begin{cases} \mathbb{R}^{(1)} & \longrightarrow & \mathcal{I}_1^{(1)} & \longrightarrow & \cdots & \longrightarrow & \mathcal{I}_{M_1}^{(1)} & \longrightarrow & \mathcal{P}^{(1)} \\ \mathbb{R}^{(2)} & \longrightarrow & \mathcal{I}_1^{(2)} & \longrightarrow & \cdots & \longrightarrow & \mathcal{I}_{M_2}^{(2)} & \longrightarrow & \mathcal{P}^{(2)} \\ \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ \mathbb{R}^{(N)} & \longrightarrow & \mathcal{I}_1^{(N)} & \longrightarrow & \cdots & \longrightarrow & \mathcal{I}_{M_N}^{(N)} & \longrightarrow & \mathcal{P}^{(N)} \end{cases}. \tag{1.3}$$

Moreover, chemical reactions displays a dynamical behavior concerning the speed at which a reaction occurs. This rate of reaction, its *kinetics*, are dependent on the conditions in the environment, such as temperature and pressure, and on some properties of the reaction itself. In the case of a *isothermal process*, i.e., when the temperature in the environment remains constant, this rate can be calculated as a constant K, leading to a representation on the form:

$$\mathbb{R} \xrightarrow{K} \mathcal{P}. \tag{1.4}$$

When the temperature in the environment is not constant, the process is said to be *endothermic* or *exothermic* if, respectively, it consumes or produces energy. The kinetics of the reactions in such processes are usually functions of the temperature which, given an activation energy E, are assumed to follow the Arrhenius equation:

$$K(T) = K_0 e^{-E/T}. \tag{1.5}$$

In practice, these chemical reactions are produced by mixing the reactants in some environment with adequate conditions. In order to control the quantities of these substances, actual processes consists in a manipulation of the concentrations of reactants in some container, usually by providing a mass flow of these substances through some fluid. A major interest is to manipulate the reactants in some way to produce a desired concentration of one or more products in the chemical reaction network, allowing this problem to be addressed by a control engineering perspective. A *chemical reactor system*, depicted in **Fig!!!**, is a system where a controller can manipulate the concentration of some reactants to produce a desired concentration of some products.

[img]

When the process is not isothermal, the occurrence of a reaction contributes to the entropy of the environment, and consequently affects the kinetics of the subsequent ones. To compensate for this, practical applications also try to control the conditions in the environment using instruments external to the reactions themselves. Because of the use of the Arrhenius equation to model these reaction rates, this control is usually implemented through a cooling or heating system coupled to the original reactor system, resulting in the schematic on **Fig!!!**.

[img]

1.3 Motivation

The use of automatic controllers to impose a desired behaviour to physical systems is a practice ubiquitous in many engineering fields. In the last years, the price of digital computers have been dropping while their performance have been growing. Consequently, digital controllers have became the central key in developments in important and innovative fields such as aeronautics [**reference**] and autonomous driving [**reference**]. In parallel, this theory is also useful to understand and bring inspiration from nature itself since, for instance, the mechanisms for temperature regulation observed in vertebrate animals behave as a feedback controller [Heller et al., 1978].

Most recent developments in Control Theory focus on using Feedback Controllers to achieve *Robust and Optimal Control*. This theory accounts for the design of controllers that deals with

uncertainty, either from the model or from the observation of the system, and are able to achieve the control objectives in a *optimal manner*. Despite being a few decades old, these fields have gained a lot of interest in the last years thanks to recent results in *Machine Learning*, particularly in *Reinforcement Learning*, that are having success in using optimization techniques for artificial agents to control themselves in environments loaded with uncertainty [**reference**].

Furthermore, the specific application of controlling chemical reactor systems brings benefits from the fact that chemical reactions are present in most biological and industrial processes. In this sense, controllers can be used to guarantee safety constraints, maximize productivity and minimize the use of resources, in such way that is unfeasible without automatic and high performing machinery.

1.4 Objectives

This works aims to provide a self-contained discussion of modelling and control of chemical reactive systems in the perspective of modern control theory. Therefore, the results are focused on *state feedback controllers* modelled in continuous and discrete time, but analysis in the frequency domain is also considered in order to explain some concepts. Several properties of these models, both in the open-loop and closed-loop regime, are summarized in the document and the intention is to have a generalized framework to understand, evaluate and design those systems. Finally, the theory of more advanced methods such as optimal estimation and optimal control is also developed in the same sense.

1.5 Chapters Organization

The chapters of this document are mainly organized in two parts. The first part, comprised by the chapters 2, 3 and 4, builds the necessary theoretical background and provides the mathematical framework for the applications. The second part, comprised by the chapters 5 and 6, describes the experiments and results of applying these methods in real-world applications.

Individually, the chapters are organized as follows: chapter 2 introduces the dynamical models and its several properties with respect to the real system behavior, chapter 3 discusses classical methods in developing automatic controllers and state observers, chapter 4 presents more advanced methods in optimal estimation and optimal control, chapter 5 describes the practical experiments used to validate the previous discussions, chapter 6 summarizes and discusses the results of the experiments and evaluate the several controllers performances and, finally, chapter 7 provides the conclusion of the document and possible future works.

Chapter 2

Dynamical System Analysis

This chapter discusses the mathematical models for dynamical systems and their use in response analysis. The sections starts by introducing a procedure to build models from physical principles and presenting equivalent common representations. Next, the response of systems, in a time domain, are analyzed in the light of such models, relating the mathematical structure with the dynamical behavior. Finally, some important properties are defined and proved using these formulations and the system response in a frequency domain is also presented.

2.1 Model from First Principles

A dynamical system is a physical system whose states evolves with time. For this reason, one can represent a dynamical system using the *first principles* from physics itself, and formulate the evolution in time by calculating the rate of change of the states in respect to time. Thus, dynamical models can be equated using differential equations with time derivatives.

A straightforward procedure to model a system consists of identifying the variables of interest and relate them using conservation laws, such as conservation of mass, conservation of energy or conservation of momentum. The resulting differential equations are in the form:

$$\begin{pmatrix}\text{Rate of}\\ \text{Mass/Energy/Momentum}\\ \text{Accumulation}\end{pmatrix} = \begin{pmatrix}\text{Mass/Energy/Momentum}\\ \text{entering the System}\end{pmatrix} - \begin{pmatrix}\text{Mass/Energy/Momentum}\\ \text{leaving the System}\end{pmatrix}. \tag{2.1}$$

The choice of which conservation law to use depends on the system itself, since the variables of interest can provide dynamics to the system in many forms. Usually, conservation of mass is used to relate dynamics of concentrations and volumes, or other material variables, while conservation of momentum is often used to relate dynamics of motion. Since energy can be converted on form, the conservation laws of this quantity can be used to model several dynamics, such as the rate of change in heat, electrical charges or velocity of a system.

In the case of a chemical reactor system, the variables of interest are the concentrations of the chemical substances in the system. Hence, the rate of accumulation of a substance can be represented using the mass conservation law, or mass balance:

$$\begin{aligned}\begin{pmatrix}\text{Accumulation}\\ \text{of mass}\\ \text{in the system}\end{pmatrix} &= \begin{pmatrix}\text{Mass}\\ \text{entering}\\ \text{the System}\end{pmatrix} - \begin{pmatrix}\text{Mass}\\ \text{leaving}\\ \text{the System}\end{pmatrix}\\ &= \left[\begin{pmatrix}\text{Mass flow}\\ \text{entering}\\ \text{System}\end{pmatrix} + \begin{pmatrix}\text{Mass}\\ \text{produced}\\ \text{by reactions}\end{pmatrix}\right] - \left[\begin{pmatrix}\text{Mass flow}\\ \text{leaving}\\ \text{System}\end{pmatrix} + \begin{pmatrix}\text{Mass}\\ \text{consumed}\\ \text{by reactions}\end{pmatrix}\right].\end{aligned} \tag{2.2}$$

Theorem 2.1. *(Mass Balance of Reactors) Consider a closed isothermal reactor system comprised of a diluted solution of constant volume V, whose reactions are described by a chemical reaction network as in Definition 1.3. The change of concentration for any compound A in this reactor is described by the dynamical model:*

$$\frac{d(\rho_A)}{dt} = q(\rho_{in}^{(A)} - \rho_{out}^{(A)}) + \left(\sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} (\rho_X)^\alpha \right) - \left(\sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} (\rho_A)^\alpha \right), \tag{2.3}$$

where $\rho_{in}^{(A)}$ and $\rho_{out}^{(A)}$ are the densities of A in the flows entering and leaving the system, respectively, and where $\alpha X \to \beta A$ and $\alpha A \to \beta X$ represents the reactions in the network between A and any other compound X with density ρ_X, each occurring with kinetic rates K_{XA} and K_{AX}, respectively.

Proof. First of all, as denoted in (2.2), the mass flow of any substance A entering and leaving the system, M_{in} and M_{out}, respectively, given a fluid inflow F_{in} with density $\rho_{in}^{(A)}$ and a fluid outflow F_{out} with $\rho_{out}^{(A)}$, can be calculated as:

$$M_{in} = \rho_{in}^{(A)} F_{in} \qquad M_{out} = \rho_{out}^{(A)} F_{out}. \tag{2.4}$$

To calculate the mass contribution from the reactions, it is necessarily first to formulate the mass contribution for a single reaction. In this case, consider a reaction between two chemical compounds X and Y, with stoichiometric numbers α and β:

$$\alpha X \xrightarrow{K_{XY}} \beta Y. \tag{2.5}$$

Under the assumption that the reactant is in a dilute solution, the rate of this equation obeys the *law of mass action* [Horn and Jackson, 1972]. Given a constant kinetic rate K_{XY}, since the system is isothermal, and the volume of the solution as V, the mass of X consumed, $M_{\text{cons}}^{(X)}$, and the mass of Y produced, $M_{\text{prod}}^{(Y)}$, are given by the power-laws:

$$M_{\text{cons}}^{(X)} = \frac{V}{\beta} K_{XY} (\rho_X)^\alpha \qquad M_{\text{prod}}^{(Y)} = \frac{V}{\alpha} K_{XY} (\rho_X)^\alpha, \tag{2.6}$$

where ρ_X and ρ_Y are the respective densities of these compounds. Assuming that the network represents a set of reactions occurring within an chemical solution of volume V, the mass of a substance A that is consumed and produced by the reactions, named respectively M_{cons} and M_{prod}, are given by summing over the contribution of each reaction on the network where A is either a reactant or a product to any other compound X:

$$M_{\text{cons}} = V \sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} (\rho_A)^\alpha \qquad M_{\text{prod}} = V \sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} (\rho_X)^\alpha. \tag{2.7}$$

Finally, packing all together, the mass balance of any substance A in a isothermal chemical reactive system can be represented by the general dynamical model:

$$\begin{pmatrix} \text{Accumulation} \\ \text{of mass} \\ \text{in the system} \end{pmatrix} = \left[\begin{pmatrix} \text{Mass flow} \\ \text{entering} \\ \text{System} \end{pmatrix} + \begin{pmatrix} \text{Mass} \\ \text{produced} \\ \text{by reactions} \end{pmatrix} \right] - \left[\begin{pmatrix} \text{Mass flow} \\ \text{leaving} \\ \text{System} \end{pmatrix} + \begin{pmatrix} \text{Mass} \\ \text{consumed} \\ \text{by reactions} \end{pmatrix} \right]$$

$$\frac{d(\rho_A V)}{dt} = \left[\rho_{in}^{(A)} F_{in} + V \sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} (\rho_X)^\alpha \right] - \left[\rho_{out}^{(A)} F_{out} + V \sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} (\rho_A)^\alpha \right]. \tag{2.8}$$

Since the system is closed, i.e., there are no leaks or unknown sources of fluids, the assumptions of a constant volume implies that $F_{in} = F_{out} = F$. Normalizing each term by the volume and substituting a new variable $q = F/V$ results in:

$$\frac{d(\rho_A)}{dt} = q(\rho_{in}^{(A)} - \rho_{out}^{(A)}) + \left(\sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} (\rho_X)^\alpha \right) - \left(\sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} (\rho_A)^\alpha \right). \tag{2.9}$$

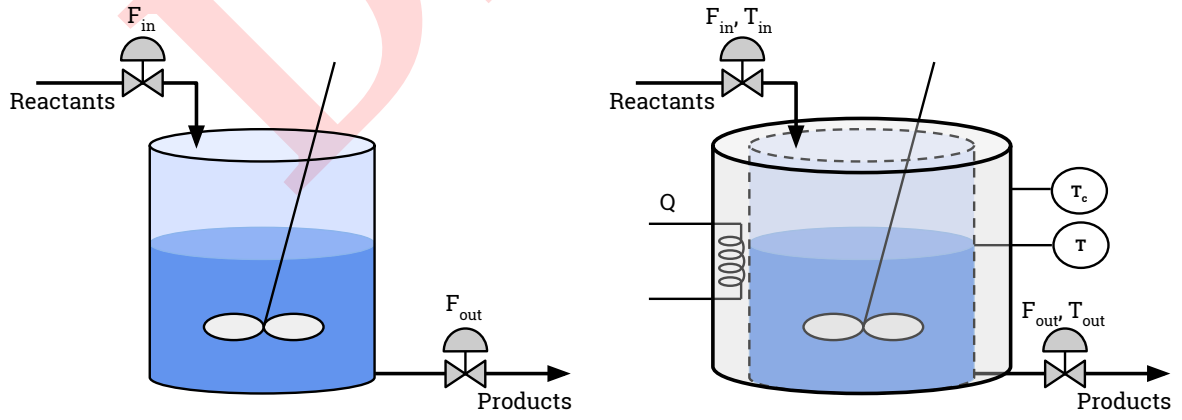
□

Notice some important restrictions to the use of the model just presented. First of all, to calculate the mass contribution of an individual reaction was necessary to use a model which assumes that the reactor system is actually comprised of a dilute solution in some closed container. In industry, this means that the reactor system is actually a tank containing the solution. The inflow and outflow of fluid can be represented by flows through pipes which can be manipulated by some valve or pump. An illustration of such physical system is exhibited at Fig. 2.1a.

Furthermore, this model accounts for a single substance A, but the system is actually a solution of several compounds, each one with a specific concentration. From the model presented, it is visible that it is necessary to compute each concentration ρ_X before actually computing the rate of change in ρ_A. However, from the same model, the computation of the rate of change of any ρ_X may depend on ρ_A itself. Therefore, the change of concentration inside the whole system is actually the result of a system of differential equations:

$$\begin{cases} \dfrac{d(\rho_{X_1})}{dt} &= f(\rho_{X_1}, \rho_{X_2}, ..., \rho_{X_n}, \rho_{in}^{(X_1)}, \rho_{out}^{(X_1)}, t) \\ \dfrac{d(\rho_{X_2})}{dt} &= f(\rho_{X_1}, \rho_{X_2}, ..., \rho_{X_n}, \rho_{in}^{(X_2)}, \rho_{out}^{(X_2)}, t) \\ \vdots & \\ \dfrac{d(\rho_{X_n})}{dt} &= f(\rho_{X_1}, \rho_{X_2}, ..., \rho_{X_n}, \rho_{in}^{(X_n)}, \rho_{out}^{(X_n)}, t) \end{cases}, \tag{2.10}$$

where $X_1, X_2, ..., X_n$ are the chemical compounds inside the reactor and $f(\cdot)$ is the dynamical model presented in Theorem 2.1.



(a) (b)

Figure 2.1: Schematic representations of industrial reactor tanks for (a) a simple isothermal process and (b) a non-isothermal process with heating/cooling system.

The model discussed so far is very simple and can describe several applications. To account for more complex systems, the same modeling procedure can be applied. For instance, it is

possible to extend the description to account for exothermic and endothermic processes, when the temperature inside the system has a dynamical evolution and the dynamics of the reactions starts to depend on it.

When discussing non-isothermal processes, it is also common to discuss heating or cooling systems that tries to impose certain operational conditions to the reactions, as illustrated in Fig. 2.1b. In exothermic processes, for instance, the heat accumulated in the system tends to grows as the reactions occurs, which can be very dangerous. One approach to regulate the temperature consists in involving the chemical solution, or the container containing it, with a material whose temperature can be manipulated, transferring or absorbing heat by conductance. The temperature of this material can be manipulated by, for instance, running a heated fluid or converting electrical energy to heat energy.

Theorem 2.2. *(Non-isothermal Reactor System) Consider a closed non-isothermal reactor system comprised of a diluted solution of constant volume V. Assume, also, that the system is involved by a cooling/heating system with capacity Q and that the reactions obeys the Arrhenius equation. The change of concentration for any compound A in this reactor is described by the dynamical model:*

$$\frac{d(\rho_A)}{dt} = q(\rho_{in}^{(A)} - \rho_{out}^{(A)}) + \left(\sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} e^{-\frac{E_{XA}}{T}} (\rho_X)^\alpha \right) - \left(\sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} e^{-\frac{E_{AX}}{T}} (\rho_A)^\alpha \right), \tag{2.11}$$

where E_{XA} and E_{AX} are the activation energy needed for each reaction, and the rest of the parameters are the same as defined in Theorem 2.1. Furthermore, the change of temperature inside the reactor system, T, and in the cooling/heating system, T_C, are described by the dynamical models:

$$\begin{cases} \dfrac{d(T)}{dt} = q(T_{in} - T_{out}) + \eta(T_C - T) + \delta \sum_{\alpha A \to \beta X} K_{AX} e^{-\frac{E_{AX}}{T}} (\rho_A)^\alpha \Delta H_{AX} \\ \dfrac{d(T_C)}{dt} = \gamma Q + \beta(T - T_C) \end{cases}, \tag{2.12}$$

where T_{in} and T_{out} are the temperatures of the fluid inflow and outflow, respectively, ΔH_{AX} is the energy change from each reaction $\alpha A \to \beta X$ and $\eta, \delta, \gamma, \beta \in \mathbb{R}$ are proportionality parameters specific to the system and environmental conditions.

A proof of this theorem can be found in Appendix A. The non-isothermal reactor system is a more general model that accounts for the fact that the temperature of the environment is usually not constant. From this assumption, the flow entering and leaving the system are also not assumed to have the same temperature that the fluid inside the reactor. In practical applications, the temperature of the fluid inflow can either be manipulated or measured, where the temperature of the fluid outflow is actually assumed to be equal to the temperature inside the reactor. In addition, the proportionality constants are not functions of any dynamical variable, so they can be calculated before the operation of the system by using the properties of the materials and containers.

In the case of this model, the rate of changes in the chemical concentrations depends on the temperature through the Arrhenius equation. However, the temperature of the reactor itself depends on those concentrations. So, as noted in Equation (2.10), the dynamical model of the entire reactor is a system of differential equations relating all those quantities.

2.2 Mathematical Models of Systems

The last section presented the foundation for modeling a dynamical system using first principles from physics. Although it was a well-defined formulation, the resulting models are not guaranteed to be practical in a mathematical sense. This is to due to the fact that the differential equations, as evidenced in Equation (2.11), are usually nonlinear functions of the variables of interest, and the analysis of such functions are quite more challenging. In the perspective of control theory, that are two main formats for the model of a system: the *Input-Output* (IO) and the *State-Space* (SS) representations.

Definition 2.1. (Input-Output Representation) An Input-Output (IO) representation of a dynamical system with $p \geq 1$ output variables, represented by $\boldsymbol{y} : \mathbb{R} \to \mathbb{R}^p$, and $r \geq 1$ input variables, represented by $\boldsymbol{u} : \mathbb{R} \to \mathbb{R}^r$, is the system of differential equations:

$$\begin{cases} h_1\left(y_1, \dot{y_1}, ..., y_1^{(n_1)}, u_1, \dot{u_1}, ..., u_1^{(m_{11})}, u_2, \dot{u_2}, ..., u_2^{(m_{12})}, ..., u_r, \dot{u_r}, ..., u_r^{(m_{1r})}, t\right) = 0 \\ h_2\left(y_2, \dot{y_2}, ..., y_2^{(n_2)}, u_1, \dot{u_1}, ..., u_1^{(m_{21})}, u_2, \dot{u_2}, ..., u_2^{(m_{22})}, ..., u_r, \dot{u_r}, ..., u_r^{(m_{2r})}, t\right) = 0 \\ \vdots \\ h_p\left(y_p, \dot{y_p}, ..., y_p^{(n_p)}, u_1, \dot{u_1}, ..., u_1^{(m_{p1})}, u_2, \dot{u_2}, ..., u_2^{(m_{p2})}, ..., u_r, \dot{u_r}, ..., u_r^{(m_{pr})}, t\right) = 0 \end{cases}, \quad (2.13)$$

where:

$$\dot{y}(t) = \frac{dy(t)}{dt}, \quad \ddot{y}(t) = \frac{d^2y(t)}{dt^2}, \quad \cdots, \quad y^{(n)}(t) = \frac{d^ny(t)}{dt^n}$$

and

$$\dot{u}(t) = \frac{du(t)}{dt}, \quad \ddot{u}(t) = \frac{d^2u(t)}{dt^2}, \quad \cdots, \quad u^{(n)}(t) = \frac{d^nu(t)}{dt^n}.$$

An input-output representation is a simple model that describes the entire system using only two types of variables, and their derivatives. Therefore, the dimension of these variables and the order of derivatives at each differential equation provides the information about the structure of the model. For instance, in the case where $p = r = 1$ the system can be classified as a *Single-Input Single-Output* (SISO) configuration, whereas it is classified as a *Multiple-Input Multiple-Output* (MIMO) configuration if $p, r > 1$.

This model presents a cause-and-effect interpretation of the system where the direct relationship between the input and output signal, and its derivatives, are equated as if the system was a processing unit. In practice, the input signals $\boldsymbol{u}(t)$ are the manipulated variables of the system, where the output signals $\boldsymbol{y}(t)$ are the observations of the controlled variables. This representation brings an easy visualization on how a desired system behavior can be achieved by applying a specific input signal, posing as a practical framework for designing controllers.

Definition 2.2. (State-Space Representation) A State-Space (SS) representation of a system with $n \geq 1$ states variables, represented by $\boldsymbol{x} : \mathbb{R} \to \mathbb{R}^n$, for $p \geq 1$ output variables, represented by $\boldsymbol{y} : \mathbb{R} \to \mathbb{R}^p$, and $r \geq 1$ input variables, represented by $\boldsymbol{u} : \mathbb{R} \to \mathbb{R}^r$, is given by the systems of

state and output equations:

State Equations:

$$\begin{cases} \dot{x}_1(t) = f_1(x_1, ..., x_n, u_1, ..., u_r, t) \\ \dot{x}_2(t) = f_2(x_1, ..., x_n, u_1, ..., u_r, t) \\ \vdots \\ \dot{x}_n(t) = f_n(x_1, ..., x_n, u_1, ..., u_r, t) \end{cases}$$

Output Equations:

$$\begin{cases} y_1(t) = g_1(x_1, ..., x_n, u_1, ..., u_r, t) \\ y_2(t) = g_2(x_1, ..., x_n, u_1, ..., u_r, t) \\ \vdots \\ y_p(t) = g_p(x_1, ..., x_n, u_1, ..., u_r, t) \end{cases}, \tag{2.14}$$

or, in the matrix form:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \\ \boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \end{cases}. \tag{2.15}$$

The State-Space representation is yet another formulation for a dynamical model, but centered in the concept of *state variables*. In a formal definition, the set of state variables is the smallest set of linearly independent variables that can unequivocally determine the value of all the states variables given an initial state $\boldsymbol{x}(t_0)$ and a forcing function $\boldsymbol{u}(t)$, for any time $t \geq t_0$. In a physical perspective, however, these variables accounts for quantities that can describe the dynamics of the system, such as position or velocities, or they are latent variables that somehow stores intrinsic information about the system behavior. In comparison to the Input-Output representation, this formulation poses a simpler mathematical model, since it is composed by a system of ordinary differential equations and a system of algebraic equations. However, this model presents a semantical improvement over the latter since the inclusion of the state variables expands the internal description of the system.

(a) (b)

Figure 2.2: Graphical interpretation of (a) Input-Output models and (b) State-Spaces models.

An graphical illustration of both representations is shown at Fig. 2.2. In the light of these formulations, a system can be classified in respect to the model mathematical structure. There are five main properties used for this classification: if the system is causal or non-causal, linear or nonlinear, dynamical or instantaneous, time-invariant or time-varying and with or without delay. The necessary and sufficient conditions for each one of these properties are summarized in Table 2.1.

This work focus on dynamical linear systems, since its models are the most well studied in the control theory community. In reality, a physical system is always causal, nonlinear and time-varying [Vidyasagar, 2002], but the models can be assumed differently with fairly accuracy. The benefit of linear systems is that it obeys the superposition principle, and a linear combination of the inputs directly causes the exact same linear combination of the individual outputs. Under the assumption of a linear system, a nice result is that the vectorial functions $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ of the State-Space representation in (2.23) reduces to simple matrix forms.

	Input-Output	State-Space
Causal	$m_{ij} \leq n_k$ $i \in [1, ..., p],\ j \in [1, ..., r]$	Always causal
Linear	$h_i(\cdot) = \sum_{j=0}^{n_i} y^{(j)} + ...$ $... + \sum_{k=1}^{r} \sum_{l=0}^{m_{ik}} u_k^{(l)}$ $i \in [1, 2, ..., p]$	$f_i = \boldsymbol{a}_i(t)\boldsymbol{x}(t) + \boldsymbol{b}_i(t)\boldsymbol{u}(t), i = 1, 2, ..., n$ $g_j = \boldsymbol{c}_j(t)\boldsymbol{x}(t) + \boldsymbol{d}_j(t)\boldsymbol{u}(t), j = 1, 2, ..., p$ $\boldsymbol{a}_i, \boldsymbol{c}_j \in \mathbb{R}^{1\times n}$ and $\boldsymbol{b}_i, \boldsymbol{d}_j \in \mathbb{R}^{1\times r}$
Dynamical	$n_i > 0$ or $m_{jk} > 0$ $i, j \in [1, ..., p],\ k \in [1, ..., r]$	$n > 0$
Time-Invariant	$h_i(y_i(t), ..., u_1(t), ..., u_r(t)) = 0$ $i \in [1, 2, ..., p]$	$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t))$ $\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t))$
Without-Delay	All the signals share the same arguments	All the signals share the same arguments

Table 2.1: Necessary and sufficient conditions for different classes of models.

Definition 2.3. (Linear State-Space Representation) A State-Space representation describing a linear system with state vector $\boldsymbol{x}(t) : \mathbb{R} \to \mathbb{R}^n$, output vector $\boldsymbol{y}(t) : \mathbb{R} \to \mathbb{R}^p$ and input vector $\boldsymbol{u}(t) : \mathbb{R} \to \mathbb{R}^r$ is given by the system of equations:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}(t)\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}(t)\boldsymbol{x}(t) + \boldsymbol{D}(t)\boldsymbol{u}(t) \end{cases}, \tag{2.16}$$

where $\boldsymbol{A}(t) : \mathbb{R} \to \mathbb{R}^{n\times n}$, $\boldsymbol{B}(t) : \mathbb{R} \to \mathbb{R}^{n\times r}$, $\boldsymbol{C}(t) : \mathbb{R} \to \mathbb{R}^{p\times n}$ and $\boldsymbol{D}(t) : \mathbb{R} \to \mathbb{R}^{p\times r}$. In the case of a time-invariant linear system, these matrices becomes constants.

This formulation has the advantages that the time response of the system can be easily calculated and that the analysis of the dynamics follows well-established results from linear algebra applied to the matrices $\boldsymbol{A}(t)$, $\boldsymbol{B}(t)$, $\boldsymbol{C}(t)$ or $\boldsymbol{D}(t)$, as well as for the vectors $\boldsymbol{x}(t)$ and $\boldsymbol{u}(t)$. Furthermore, the physical interpretation of the system through the state variables becomes straightforward in this model, even in the case of latent variables.

In addition to the State-Space representation, the linear assumption also benefits Input-Output representations. One major analytical tool that can be used in these cases is to transform this model to a frequency domain, using a linear transform operator, in order to simplify the solution for the differential equations. The most popular choice of transformation is the *Laplace transform*, $\mathcal{L}\{h(t)\}$, which converts functions in time to functions in complex frequencies. Using the properties of this operator, differential equations are converted to simple algebraic equations.

Theorem 2.3. *(Transfer Function) Given a linear model for a SISO system, with initial conditions* $\boldsymbol{y}(0^-) = \boldsymbol{u}(0^-) = \boldsymbol{0}$*, in the Input-Output formulation:*

$$\alpha_n \frac{d^n y(t)}{dt^n} + \cdots + \alpha_1 \frac{dy(t)}{dt} + \alpha_0 y(t) = \beta_m \frac{d^m u(t)}{dt^m} + \cdots + \beta_1 \frac{du(t)}{dt} + \beta_0 u(t). \tag{2.17}$$

Its transfer function, in the Laplace domain, is calculated as:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\beta_m s^m + \beta_{m-1} s^{m-1} + \cdots + \beta_1 s + \beta_0}{\alpha_n s^n + \alpha_{n-1} s^{n-1} + \cdots + \alpha_1 s + \alpha_0}. \tag{2.18}$$

An indirect result of this is that the SS representation can be converted to the IO representation using the Laplace transform operator, leading to a notion of equivalence between the two representations. Notice that the extension to the MIMO case is straightforward: just compute the transfer function between each pair of input and output, leading to the matrix $\boldsymbol{G} \in \mathbb{C}^{n \times m}$.

Theorem 2.4. *(Passage from SS to IO) Consider a linear and time-invariant system in State-Space form with initial states $\boldsymbol{x}(0^-) = \boldsymbol{0}$ and represented as:*

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}(t) \end{cases}. \tag{2.19}$$

The equivalent system in Input-Output representation is given by the transfer function:

$$\boldsymbol{G}(s) = \boldsymbol{Y}(s)\boldsymbol{U}^{-1}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}. \tag{2.20}$$

Proof. Applying the Laplace transfer in both sides of the equation:

$$\begin{aligned} \mathcal{L}\{\dot{\boldsymbol{x}}(t)\} &= \boldsymbol{A}\mathcal{L}\{\boldsymbol{x}(t)\} + \boldsymbol{B}\mathcal{L}\{u(t)\} \\ s\boldsymbol{X}(s) - \dot{\boldsymbol{x}}(0^-) &= \boldsymbol{A}\boldsymbol{X}(s) + \boldsymbol{B}\boldsymbol{U}(s) \\ (s\boldsymbol{I} - \boldsymbol{A})\boldsymbol{X}(s) &= \boldsymbol{B}\boldsymbol{U}(s) \\ \boldsymbol{X}(s) &= (s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}\boldsymbol{U}(s) \end{aligned}. \tag{2.21}$$

By applying the same procedure in the output equations, using the previous result, the Laplace transform of the output is:

$$\begin{aligned} \mathcal{L}\{\boldsymbol{y}(t)\} &= \boldsymbol{C}\mathcal{L}\{\boldsymbol{y}(t)\} + \boldsymbol{D}\mathcal{L}\{u(t)\} \\ \boldsymbol{Y}(s) &= \boldsymbol{C}\boldsymbol{X}(s) + \boldsymbol{D}\boldsymbol{U}(s) \\ &= \boldsymbol{C}\left((s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}\boldsymbol{U}(s)\right) + \boldsymbol{D}\boldsymbol{U}(s) \\ &= \left(\boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}\right)\boldsymbol{U}(s) \end{aligned}. \tag{2.22}$$

In conclusion, since by definition $\boldsymbol{Y}(s) = \boldsymbol{G}(s)\boldsymbol{U}(s)$, it is possible to obtain the transfer function matrix $\boldsymbol{G}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}$ as an equivalent representation of the system. □

Despite of the discussion about the benefits of linear models, it is necessary to account for the fact that physical systems will present, in most situations, nonlinear behavior. For this reason, some effort must be done to develop a linear model that can describe the nonlinear behavior with certain accuracy, even if over some small region of the space. With this motivation, a technique for *linearization* of a nonlinear model is detailed below.

Theorem 2.5. *(Linearization by Taylor Expansion) Consider a nonlinear time-invariant system:*

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \end{cases}. \tag{2.23}$$

Given steady-state operating points $\boldsymbol{x}_o$, $\boldsymbol{y}_o$ and $\boldsymbol{u}_o$, the dynamics of the system in the neighborhood of these points can be represented by the linear model:

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\Delta\boldsymbol{x}(t) + \boldsymbol{B}\Delta\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\Delta\boldsymbol{x}(t) + \boldsymbol{D}\Delta\boldsymbol{u}(t) \end{cases}, \tag{2.24}$$

where

$$\boldsymbol{A} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o}, \quad \boldsymbol{B} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o}, \quad \boldsymbol{C} = \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o}, \quad \boldsymbol{D} = \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \tag{2.25}$$

and

$$\Delta\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_o; \qquad \Delta\boldsymbol{u}(t) = \boldsymbol{u}(t) - \boldsymbol{u}_o. \tag{2.26}$$

Proof. Consider a system represented by state equations $\boldsymbol{f}(\cdot)$ and output equations $\boldsymbol{g}(\cdot)$, with steady-state points $\boldsymbol{x}_o$, $\boldsymbol{y}_o$ and $\boldsymbol{u}_o$. Now, consider a very small perturbation $\Delta u(t)$ in the input signal around these operation points. This perturbation will result in small changes in the state and output variables:

$$\boldsymbol{x}(t) = \boldsymbol{x}_o + \Delta\boldsymbol{x}(t); \qquad \boldsymbol{u}(t) = \boldsymbol{u}_o + \Delta\boldsymbol{u}(t); \qquad \boldsymbol{y}(t) = \boldsymbol{y}_o + \Delta\boldsymbol{y}(t). \tag{2.27}$$

This results in the following configuration on the State-Space:

$$\begin{cases} \dfrac{d(\boldsymbol{x}_o + \Delta\boldsymbol{x}(t))}{dt} & = \boldsymbol{f}(\boldsymbol{x}_o + \Delta\boldsymbol{x}(t), \boldsymbol{u}_o + \Delta\boldsymbol{u}(t)) \\ \boldsymbol{y}_o + \Delta\boldsymbol{y}(t) & = \boldsymbol{g}(\boldsymbol{x}_o + \Delta\boldsymbol{x}(t), \boldsymbol{u}_o + \Delta\boldsymbol{u}(t)) \end{cases}, \tag{2.28}$$

where $d(\boldsymbol{x}_o + \Delta\boldsymbol{x}(t))/dt = d(\Delta\boldsymbol{x}(t))/dt$, since $\boldsymbol{x}_o$ is constant. The perturbed variables are very close to the operation points, hence the functions $\boldsymbol{f}(\cdot)$ and $g(\cdot)$ can be approximated by a Taylor series expansion, yielding:

$$\begin{cases} \dfrac{d(\Delta\boldsymbol{x}(t))}{dt} & = \boldsymbol{f}(\boldsymbol{x}_o, \boldsymbol{u}_o) + \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) + \text{high order terms} \\ \boldsymbol{y}_o + \Delta\boldsymbol{y}(t) & = \boldsymbol{g}(\boldsymbol{x}_o, \boldsymbol{u}_o) + \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) + \text{high order terms} \end{cases}. \tag{2.29}$$

Since the steady-state condition implies zero variation, it is possible to assume $\boldsymbol{f}(\boldsymbol{x}_o, \boldsymbol{u}_o) = 0$ and $g(\boldsymbol{x}_o, \boldsymbol{u}_o) = 0$, since they are ordinary differential equations. Truncating in the first order terms and substituting $\boldsymbol{y}(t) = \boldsymbol{y}_o + \Delta\boldsymbol{y}(t)$ results in:

$$\begin{cases} \dfrac{d(\Delta\boldsymbol{x}(t))}{dt} = \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) \end{cases}. \tag{2.30}$$

Finally, since all the Jacobians involved are actually matrices of appropriate dimensions, the final linear approximation of the system is the SS model given by:

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\Delta\boldsymbol{x}(t) + \boldsymbol{B}\Delta\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\Delta\boldsymbol{x}(t) + \boldsymbol{D}\Delta\boldsymbol{u}(t) \end{cases}. \tag{2.31}$$

□

Note that this method can be used to convert non-linear models obtained from the first-principle procedure to a desirable State-Space form, since those models are usually in the non-linear State-Space representation. Consider for instance, a reactor system describing an isothermal process that follows the Van de Vusse reaction scheme [Van de Vusse, 1964]:

$$\begin{array}{cc} \textbf{Reaction Scheme:} & \textbf{Flow parameters:} \\ A \overset{K_{AB}}{\rightarrow} B \overset{K_{BC}}{\rightarrow} C & \rho_{in}^{(B)}(t) = \rho_{in}^{(C)}(t) = \rho_{in}^{(D)}(t) = 0 \ . \\ 2A \overset{K_{AD}}{\rightarrow} D & \rho_{out}^{(k)}(t) = \rho_k(t), \ k \in \{A, B, C, D\} \end{array} \tag{2.32}$$

Consider $\boldsymbol{x} = [\rho_A, \rho_B, \rho_C, \rho_D]^T$ and $u = q$, and consider also that the states are perfectly observed, so that $\boldsymbol{y} = \boldsymbol{x}$. Using Theorem 2.1, the state-equation for the nonlinear State-Space representation of this reactor is:

$$\begin{cases} \dot{x_1}(t) = u(t)(\rho_{in}^{(A)} - x_1(t)) - (K_{AB}x_1(t) + K_{AD}(x_1(t))^2) \\ \dot{x_2}(t) = -u(t)x_2(t) + K_{AB}x_1(t) - K_{BC}x_2(t) \\ \dot{x_3}(t) = -u(t)x_3(t) + K_{BC}x_2(t) \\ \dot{x_4}(t) = -u(t)x_4(t) + \dfrac{1}{2}K_{AD}(x_1(t))^2 \end{cases} . \tag{2.33}$$

To select steady-state points $\boldsymbol{x}_o$ and $\boldsymbol{u}_o$, it is necessary to find values for $\boldsymbol{x}(t)$ and $u(t)$ that satisfies the system $\dot{\boldsymbol{x}}(t) = \boldsymbol{0}$. This can be done by analytical formulas, in some cases, by numerical methods or by simply simulating the nonlinear system and measuring these quantities when the states displays zero variation. Finally, the linear matrices $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$ are obtained by deriving each one of these equations in respect to each state and input, resulting:

$$\begin{cases} \boldsymbol{A} = \begin{bmatrix} -u_o - K_{AB} - 2K_{AC}x_{1o} & 0 & 0 & 0 \\ K_{AB} & -u_o - K_{BC} & 0 & 0 \\ 0 & K_{BC} & -u_o & 0 \\ K_{AD}x_{1o} & 0 & 0 & -u_o \end{bmatrix} & \boldsymbol{B} = \begin{bmatrix} \rho_{in}^{(A)} - x_{1o} \\ -x_{2o} \\ -x_{3o} \\ -x_{4o} \end{bmatrix} \\ \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \boldsymbol{D} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{cases} . \tag{2.34}$$

Notice that the response of this model represents the deviations around the given operation points. If the errors in approximating the non-linearities of the actual system are somehow tolerable, this model can still be used as a nominal model to represent all the operations, given that the response is vertically "correted" as $\boldsymbol{x}(t) = \Delta\boldsymbol{x}(t) + \boldsymbol{x}_o$ (and for initial states $\Delta\boldsymbol{x}_0 = \boldsymbol{x}(t_0) - \boldsymbol{x}_o$ and input signal $\Delta\boldsymbol{u}(t) = \boldsymbol{u}(t) - \boldsymbol{u}_o$). The simulation of both the nonlinear and the linear systems are displayed in Fig. 2.3, with the dashed line the linear response of a system given $\boldsymbol{x}_o = [6.19, 1.09, 0.60, 1.05]^T$ and $u_0 = 3.03$.

2.3 Response Analysis in Time Domain

Once that a model is well-established, it is possible both to simulate the system and to analyze its response, both for a natural or forced regime. This section, then, focus on developing a quantitative understanding of a system behavior through a linear model. The results are focused on continuous-time response of linear and time-invariant (LTI) systems in the form:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases} . \tag{2.35}$$

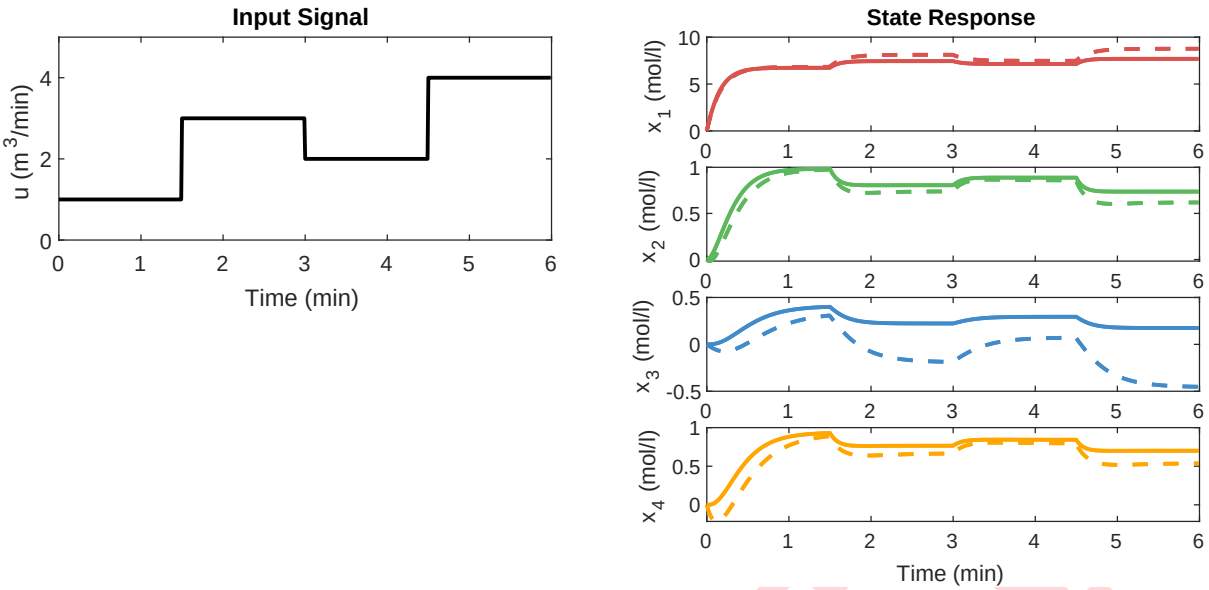


Figure 2.3: Comparison between the nonlinear and linear response of the reactor system.

First of all, it is necessary to access some properties of the matrix $\boldsymbol{A}$, which describes the natural evolution of the states.

Definition 2.4. (State-Transition Matrix) Consider a system in State-Space representation with matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$. Its *State-Transition Matrix*, $e^{\boldsymbol{A}t} \in \mathbb{R}^{n\times n}$, is the converging series:

$$e^{\boldsymbol{A}t} = \boldsymbol{I} + \boldsymbol{A}\boldsymbol{t} + \frac{\boldsymbol{A}^2t^2}{2!} + \frac{\boldsymbol{A}^3t^3}{3!} + \cdots = \sum_{k=0}^{\infty} \frac{A^kt^k}{k!}. \tag{2.36}$$

This matrix is central to the computation of the system time response, as it will be shown shortly. Since the matrix A is a square matrix, this operation leads to some useful properties.

Theorem 2.6. *Consider a matrix exponential as in Definition 2.4, for a matrix* $\boldsymbol{A} \in \mathbb{R}^{n\times n}$. *Then, the following properties holds:*

$$1)\ \frac{d(e^{\boldsymbol{A}t})}{dt} = \boldsymbol{A}e^{\boldsymbol{A}t} \qquad 2)\ e^{\boldsymbol{A}t}e^{\boldsymbol{A}\tau} = e^{\boldsymbol{A}(t+\tau)} \qquad 3)\ e^{-\boldsymbol{A}t}e^{\boldsymbol{A}t} = e^{\boldsymbol{A}t}e^{-\boldsymbol{A}t} = \boldsymbol{I}. \tag{2.37}$$

The proof is direct from Definition 2.4, and a detailed discussion of these properties is omitted. Given these results, the calculation of the time response of a system in SS representation becomes straightforward.

Theorem 2.7. *(Lagrange Formula) Consider a LTI system in State-Space representation. Its response for any time* $t \geq t_0$, *initial state* $\boldsymbol{x}(t_0)$ *and input signal* $\boldsymbol{u}(t)$ *is given by the solutions of*

the state and output equations:

$$\begin{cases} \boldsymbol{x}(t) = e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t) + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \\ \boldsymbol{y}(t) = \boldsymbol{C}e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t) + \boldsymbol{C}\int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau + \boldsymbol{D}\boldsymbol{u}(t) \end{cases} . \tag{2.38}$$

Proof. First of all, consider a system in State-Space representation with state equation as defined in Equation (2.23). Multiplying both sides by $e^{-\boldsymbol{A}t}$:

$$\begin{aligned} e^{-\boldsymbol{A}t}\dot{\boldsymbol{x}}(t) &= e^{-\boldsymbol{A}t}(\boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)) \\ e^{-\boldsymbol{A}t}\dot{\boldsymbol{x}}(t) - \boldsymbol{A}e^{-\boldsymbol{A}t}\boldsymbol{x}(t) &= e^{-\boldsymbol{A}t}\boldsymbol{B}\boldsymbol{u}(t) \end{aligned} . \tag{2.39}$$

Using the first assertive in Theorem 2.6, it is easy to see that $d[e^{-\boldsymbol{A}t}x(t)]/dt = e^{-\boldsymbol{A}t}\dot{\boldsymbol{x}}(t) - \boldsymbol{A}e^{-\boldsymbol{A}t}\boldsymbol{x}(t)$. Substituting this result in (2.39) and integrating both sides from t_0 to t:

$$\begin{aligned} \frac{d(e^{-\boldsymbol{A}t}\boldsymbol{x}(t))}{dt} &= e^{-\boldsymbol{A}t}\boldsymbol{B}\boldsymbol{u}(t) \\ e^{-\boldsymbol{A}t}\boldsymbol{x}(t)\Big|_{t_0}^{t} &= \int_{t_0}^{t} e^{-\boldsymbol{A}\tau}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau. \\ e^{-\boldsymbol{A}t}\boldsymbol{x}(t) - e^{-\boldsymbol{A}t_0}\boldsymbol{x}(t_0) &= \int_{t_0}^{t} e^{-\boldsymbol{A}\tau}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \end{aligned} \tag{2.40}$$

Multiplying both sides by $e^{\boldsymbol{A}t}$ and using the second and third assertive from Theorem 2.6, the state response can be calculated as:

$$\begin{aligned} e^{\boldsymbol{A}t}\left(e^{-\boldsymbol{A}t}\boldsymbol{x}(t) - e^{-\boldsymbol{A}t_0}\boldsymbol{x}(t_0)\right) &= e^{\boldsymbol{A}t}\int_{t_0}^{t} e^{-\boldsymbol{A}\tau}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \\ \boldsymbol{I}\boldsymbol{x}(t) - e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t_0) &= \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \\ \boldsymbol{x}(t) &= e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \end{aligned} . \tag{2.41}$$

Finally, substituting (2.41) into the output equation leads to:

$$\begin{aligned} \boldsymbol{y}(t) &= \boldsymbol{C}\left(e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau\right) + \boldsymbol{D}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t) + \boldsymbol{C}\int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau + \boldsymbol{D}\boldsymbol{u}(t) \end{aligned} . \tag{2.42}$$

□

When discussing the response of a system, the focus is actually directed to the state equation describing its dynamics, since the output equation only represents an observation of the system through the states. In this sense, the Lagrange formula exposes the nice characteristic of linear systems that the total response is a composition of two separated actions:

$$\boldsymbol{x}(t) = \boldsymbol{x}_{\mathrm{n}}(t) + \boldsymbol{x}_{\mathrm{f}}(t), \tag{2.43}$$

where the natural response, $\boldsymbol{x}_{\mathrm{n}}(t)$, corresponds to the state-transition matrix multiplication term and the forced response, $\boldsymbol{x}_{\mathrm{f}}(t)$, corresponds to the integral term. This concept is visualized in Fig. 2.4 for the total response of the first two states of the model derived in Equation (2.34),

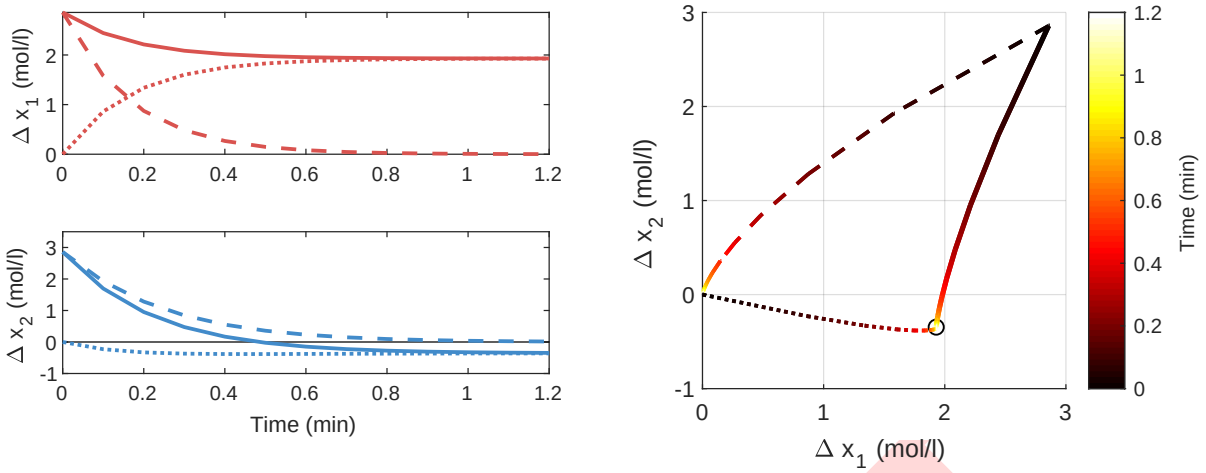


Figure 2.4: Simulation of the total response decomposition for the two first states of the model from (2.34) shown separately (left) and in a single visualization (right) in which the steady-state point is indicated by a circle.

given the operation points $\boldsymbol{x}_o = [6.19, 1.09]$ and $u_o = 3.03$, excited with a step input $u(t) = 3$, $t \in [0, 1.2]$. In this plot, the solid line represents the total response, whereas the dashed and dotted lines represents the natural and forced response, respectively. It is easy to verify that the decomposition of the total response is equal to the sum of those independent components.

The only problem remaining to fully characterize the dynamical response of a system is to compute the state-transition matrix, which can be done in several ways [Moler and Loan, 2003]. A specific method, known as the *Sylvester expansion*, is an analytical solution that brings an interesting understanding of the system behavior through the state-state matrix A.

Theorem 2.8. *(Sylvester expansion) Consider a matrix exponential function $\boldsymbol{f}(\boldsymbol{A}) = e^{\boldsymbol{A}t}$ for any square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, whose distinct eigenvalues are $\boldsymbol{\lambda} \in \mathbb{R}^m$, $m \leq n$, with associated multiplicity vector $\boldsymbol{\nu} \in \mathbb{R}^m$ such that $\sum_j \nu_j = n$. The result of this function can be expanded as:*

$$f(\boldsymbol{A}) = e^{\boldsymbol{A}t} = \beta_0(t)\boldsymbol{I} + \beta_1(t)\boldsymbol{A} + \beta_2(t)\boldsymbol{A}^2 + ... + \beta_{n-1}(t)\boldsymbol{A}^{n-1} = \sum_{i=0}^{n-1} \beta_i(t)\boldsymbol{A}^i, \tag{2.44}$$

where $\beta_i(t) : \mathbb{R} \to \mathbb{R}$, $i \in [1, 2, ..., n-1]$, are scalar functions of time that solves the linear system:

$$\boldsymbol{V}\boldsymbol{\beta} = \boldsymbol{\eta} \tag{2.45}$$

for the parameter matrix $\boldsymbol{\beta} = [\beta_0(t), \beta_1(t), ..., \beta_n(t)]^T$, and the vector of modes $\boldsymbol{\eta} = [\eta_1, \eta_2, ..., \eta_m]^T$ and the confluent Vandermonde matrix $\boldsymbol{V} = [V_1, V_2, ..., V_m]^T$ given as

$$\eta_i = \begin{bmatrix} e^{\lambda_i t} & te^{\lambda_i t} & t^2 e^{\lambda_i t} & \cdots & t^{\nu_i - 1} e^{\lambda_i t} \end{bmatrix}^T$$

$$V_j = \begin{bmatrix} 1 & \lambda_j & \lambda_j^2 & \cdots & \lambda_j^{(\nu_j-1)} & \cdots & \lambda_j^{n-1} \\ 0 & 1 & 2\lambda_j & \cdots & (\nu_j - 1)\lambda_j^{(\nu_j-1)} & \cdots & (n-1)\lambda_j^{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & (\nu_j - 1)! & \cdots & \dfrac{(n-1)!}{(n-\nu_j)!}\lambda_j^{n-\nu_j} \end{bmatrix}$$

The proof of this expansion is somewhat extensive, but a detailed discussion can be found in [Chen, 1998]. Basically, the expansion in Equation (2.44) is a direct application of the Cayley-Hamilton theorem [Atiyah, 2018] and the linear system in Equation (2.45) is a result of the Sylvester's matrix theorem [Horn and Johnson, 2012].

From the expansion presented in Theorem 2.8, it is possible to understand the relationship between the state-transition matrix $e^{\boldsymbol{A}t}$ and each eigenvalue λ of the matrix $\boldsymbol{A}$, also known as the *poles* of the system. First of all, notice that the formulation of the linear system that defines the parameters $\beta_0(t), \beta_1(t), ..., \beta_{n-1}(t)$ implies that each one of these functions are linear combinations of the exponentials $e^{\lambda_i t}$ for each eigenvalue $\lambda_i,\ i = 1, 2, ...m$. These exponentials are known as the *modes* of the matrix $\boldsymbol{A}$. Since the Sylvester expansion is linear in those parameters, it is possible to conclude that the state-transition matrix, and consequently the response of a system, is a linear combination of the modes.

Consider, for the sake of illustration, the same reactor model from Equation (2.34), but considering only the first two states (since they are independent of the others). Let $\boldsymbol{x}_o = [6.19, 1.09]^T$ and $u_0 = 3.03$, just as before. The resulting matrix $\boldsymbol{A}$ and state-transition matrix $e^{\boldsymbol{A}t}$ are shown below, where is easy to see that $\boldsymbol{\lambda} = [-5.93, -4.70]$, since $\boldsymbol{A}$ is diagonal:

$$\boldsymbol{A} = \begin{bmatrix} -5.93 & 0 \\ 0.83 & -4.70 \end{bmatrix} \qquad e^{\boldsymbol{A}t} = \begin{bmatrix} e^{-5.93t} & 0 \\ 0.68e^{-4.7t} - 0.68e^{-5.93t} & e^{-4.70t} \end{bmatrix}. \tag{2.46}$$

A simulation in time, shown in Fig 2.5, shows the evolution of each element of $e^{\boldsymbol{A}t}$ for a given time interval. Considering only the natural response $\boldsymbol{x}_n(t)$, i.e., setting $\boldsymbol{u}(t) = \boldsymbol{0},\ t \in [t_0, t]$, it is evident that the actual response of each state in the system is a row-wise weighted sum of these elements, where the weights are given by the initial state $\boldsymbol{x}(t_0)$. Therefore, the (i, j) element of this matrix describes how the j-th state affects the response of the i-th state.

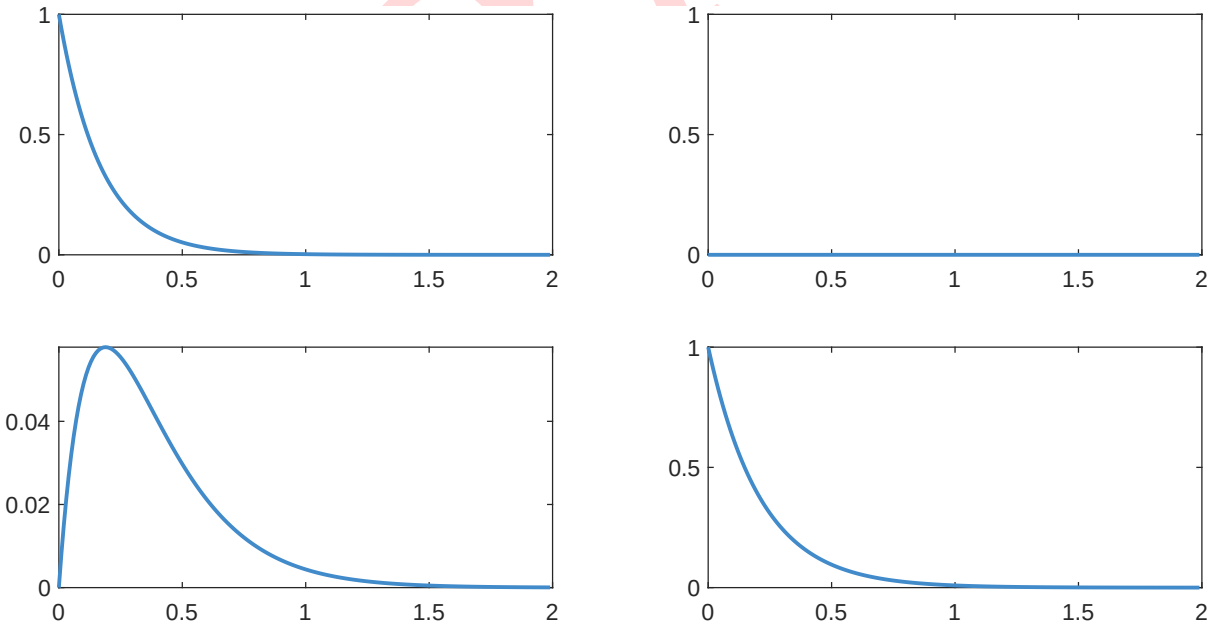


Figure 2.5: Simulation of the state-transition matrix $e^{\boldsymbol{A}t}$ in Equation (2.46) for $t \in [0, 2]$

Now, attention must be drawn to the case where the eigenvalues are not real, but complex and conjugate. Despite that the Sylvester expansion is still defined as in Theorem 2.8, this case introduces a slightly different interpretation of the modes contributions to the natural response.

Theorem 2.9. *Consider the same expansion defined in Theorem 2.8. Consider, now, that the matrix $\boldsymbol{A}$ has two distinct eigenvalues $\lambda_c, \lambda_c' \in \mathbb{C}$ in the form $\lambda_c, \lambda_c' = \alpha \pm j\omega$. In this case, the linear system solved by the parameters $\beta_0(t), \beta_1(t), ..., \beta_{n-1}(t)$ will have two equations:*

$$\begin{cases} \beta_0 + \alpha\beta_1 + \alpha^2\beta_2 + ... + \alpha^{n-1}\beta_{n-1} = e^{\alpha t}cos(\omega t) \\ 0 + \omega\beta_1 + \omega^2\beta_2 + ... + \omega^{n-1}\beta_{n-1} = e^{\alpha t}sin(\omega t) \end{cases}. \tag{2.47}$$

Proof. Consider the matrix $\boldsymbol{A}$ with eigenvalues as specified and the Sylvester expansion as presented. In this case, there will be two equations in the system:

$$\begin{cases} \beta_0 + \lambda_c\beta_1 + \lambda_c^2\beta_2 + ... + \lambda_c^{n-1}\beta_{n-1} = e^{\lambda_c t} \\ \beta_0 + \lambda_c'\beta_1 + (\lambda_c')^2\beta_2 + ... + (\lambda_c')^{n-1}\beta_{n-1} = e^{\lambda_c' t} \end{cases}. \tag{2.48}$$

Since the eigenvalues are complex and conjugate, it has that $\lambda_c + \lambda_c' = 2\mathbb{R}e[\lambda_c] = 2\alpha$ and $\lambda_c - \lambda_c' = 2j\mathcal{I}m[\lambda_c] = 2j\omega$. Moreover, the Euler identity $e^{\alpha \pm j\omega} = e^{\alpha t}(cos(\omega) \pm jsin(\omega))$ shows that $e^{\lambda} + e^{\lambda'} = 2e^{\alpha t}cos(\omega t)$ and $e^{\lambda} - e^{\lambda'} = 2e^{\alpha t}sin(\omega t)$. In this case, summing the two rows and subtracting the first row by the second one results in:

$$\begin{cases} \beta_0 + 2\alpha\beta_1 + 2\alpha^2\beta_2 + ... + 2\alpha^{n-1}\beta_{n-1} = 2e^{\alpha t}cos(\omega t) \\ 0 + 2j\omega\beta_1 + 2j\omega^2\beta_2 + ... + 2j\omega^{n-1}\beta_{n-1} = 2je^{\alpha t}sin(\omega t) \end{cases}. \tag{2.49}$$

Finally, dividing the first row by 2 and the second row by $2j$ results in (2.47). □

From the same reasons stated before, this result implies that the actual response of the system will have sinusoidal components that produces oscillations in the response. The modes associated with complex and conjugate eigenvalues are classified as *pseudo-periodic*, since they are composed by an exponential growth (or decay) enveloping a sinusoidal function. Consider, again for the sake of illustration, the following toy example:

$$\boldsymbol{A} = \begin{bmatrix} -0.1 & 0.5 \\ -0.5 & -0.1 \end{bmatrix} \qquad e^{\boldsymbol{A}t} = \begin{bmatrix} e^{-0.1t}cos(0.5t) & e^{-0.1t}sin(0.5t) \\ -e^{-0.1t}sin(0.5t) & e^{-0.1t}cos(0.5t) \end{bmatrix}. \tag{2.50}$$

The elements of the resulting state-transition matrix are simulated for a specific time-interval and shown in Fig. 2.6. It is possible to see, in this case, the pseudo-periodic behavior of the complex conjugate modes, where the dashed lines represents the exponential envelope.

The previous discussion on the modes of matrix $\boldsymbol{A}$ introduced the importance of the eigenvalues of this matrix in analyzing the system response. The analysis, however, was focused on the natural response $\boldsymbol{x}_n(t)$, whereas the forced response $\boldsymbol{x}_f(t)$ was deliberately omitted. Now, the analysis focus the opposite case. Consider, for instance, the response of the first state for the two systems described by the matrices $(\boldsymbol{A}^{(1)}, b^{(1)})$ and $(\boldsymbol{A}^{(2)}, b^{(2)})$ given below, which are more complete descriptions of the systems in (2.46) and (2.50). Considering initial states $\boldsymbol{x}^{(1)}(0) = \boldsymbol{x}^{(2)}(0) = \boldsymbol{0}$ and the input signals $\boldsymbol{u}^{(1)}(t) = \boldsymbol{u}^{(2)}(t) = \boldsymbol{1}$, for time $t \in [0, 60)$, the evolution of these states are shown in Fig. 2.7.

$$\boldsymbol{A}^{(1)} = \begin{bmatrix} -5.93 & 0 \\ 0.83 & -4.70 \end{bmatrix} \quad b^{(1)} = \begin{bmatrix} 3.81 \\ -1.09 \end{bmatrix} \qquad \boldsymbol{A}^{(2)} = \begin{bmatrix} -0.1 & 0.5 \\ -0.5 & -0.1 \end{bmatrix} \quad b^{(2)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \tag{2.51}$$

Since $\boldsymbol{A}^{(1)}$ is diagonal and $\boldsymbol{A}^{(2)}$ has a single pair of complex conjugate eigenvalues, it is possible to consider that those are respectively the characteristic responses of aperiodic and pseudo-periodic modes to an unitary step input. Since any signal can be decomposed by a sequence of step signals, the forced response to an unitary step is the default evolution used in literature to characterize the behavior of modes (and ultimately, of systems) in respect to some specifications of the transient response defined below.

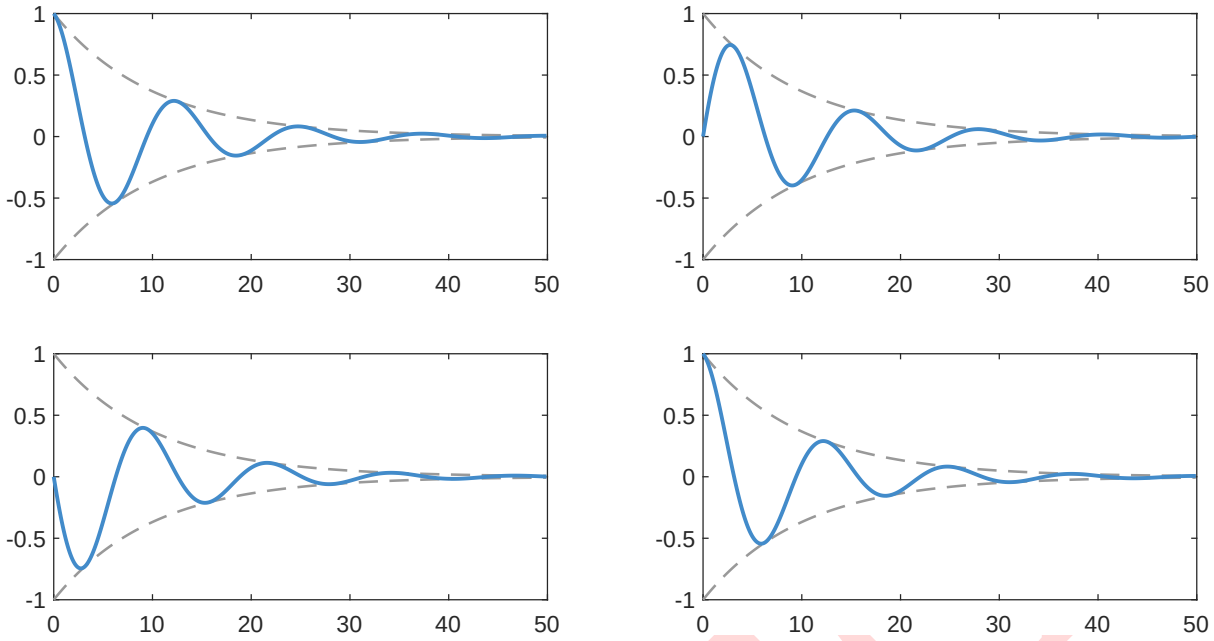


Figure 2.6: Simulation of the state-transition matrix $e^{\boldsymbol{A}t}$ in Equation (2.50) for $t \in [0, 50]$

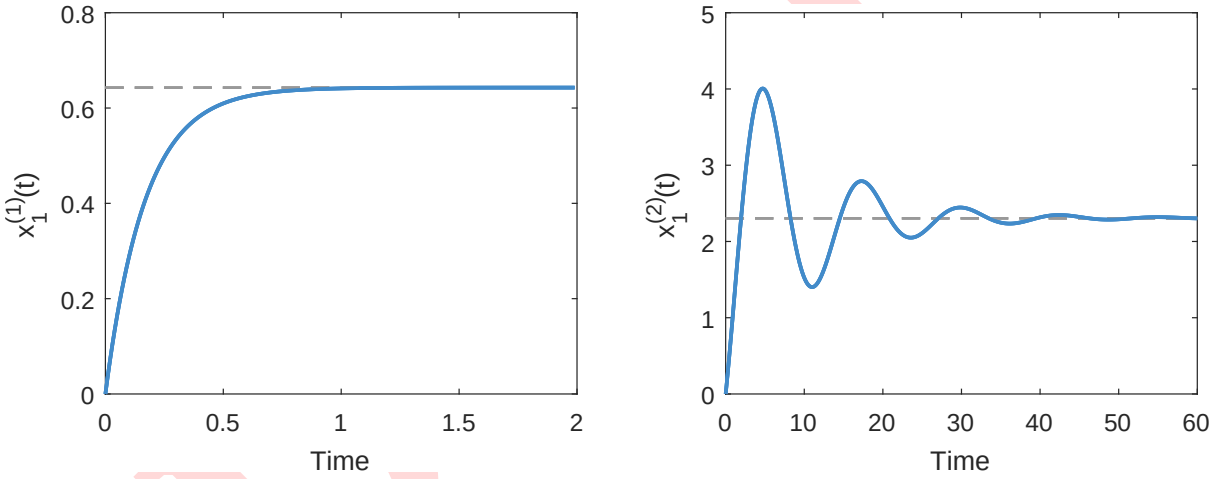


Figure 2.7: One-state forced response of the two systems from (2.51).

Definition 2.5. (Unit-Step Response Specifications) Given a mode $e^{\lambda t}$ associated with the eigenvalue $\lambda \in \mathbb{R}$, its contribution to the response has a time constant (τ) defined as

$$\tau = -\frac{1}{\lambda}. \tag{2.52}$$

Furthermore, given pseudo-periodic modes $e^{\lambda t}$ and $e^{\lambda' t}$ of the matrix $\boldsymbol{A}$ associated with the eigenvalues $\lambda, \lambda' = \alpha \pm j\omega$, their contribution to the response has a time constant (τ), a natural frequency (ω_n) and a damping coefficient (ζ) defined as:

$$\tau = -\frac{1}{\alpha} \qquad \omega_n = \sqrt{\alpha^2 + \omega^2} \qquad \zeta = -\frac{\alpha}{\omega_n}. \tag{2.53}$$

The specifications just defined provides a quantitative way to describe the response of a system in terms of time and frequencies. The time constant, for instance, is a quantity that represents the time needed for the mode to lost 63% of its initial value, since $e^{\lambda\tau} = e^{-1} = 0.37$. A greater value of a time constant indicates that the system is able to "discharge" energy faster. The damping coefficient, in turn, provides an information about the intensity of the peak in the pseudo-periodic responses, which is known as *overshoot* (or *undershoot* in the case that of a negative peak) and the natural frequency represents the oscillation of the response before reaching steady-state. From the perspective of control theory, these are some of the specifications used to define desirable transient responses to a controlled system.

Notice that the response specifications are always functions of the real and imaginary parts of the discussed eigenvalues. This brings the possibility of a visualization in the complex plane to interpret how each eigenvalue contributes to the total response. A straightforward notion is that the closer an eigenvalue is to the imaginary axis, the faster is its contribution. Similarly, the furthest an eigenvalue is to the real axis, the more oscillatory is its contribution. Finally, a vector from the origin of the plane to a complex eigenvalue has a norm equal to the natural frequency (ω_n) and the cosine of the angle formed with the imaginary axis is equal to the damping factor (ζ). A simulation of the contributions from different eigenvalues are shown in Fig. 2.8.

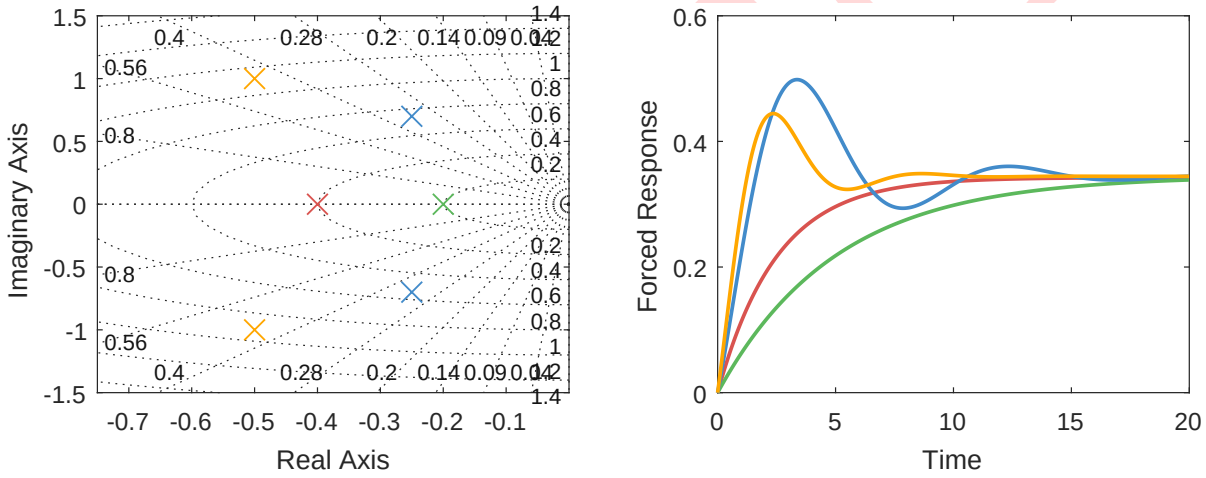


Figure 2.8: Eigenvalues of a system and the forced response associated with their modes.

2.4 Similarity Transformations

A State-Space representation can be interpreted through a system of coordinates. A state, in this context, represents a vector as visualized through this reference. Under the assumption of a linear time-invariant system, there is an intuition that is possible to change the representation of the states by changing this system of coordinates through some linear transformation, obtaining a different model to the same system. This is the motivation for the discussion in this section.

First of all, consider a brief reflection about the geometrical interpretation of a State-Space model for a physical system. Let a state-vector in an arbitrary time t be $\boldsymbol{x}(t) = [1, 3]^T$, defined in the $\mathbb{R}^2$ space, as shown in the left side of Fig. 2.9 in a Cartesian coordinate system. It is possible to associate to this vector an orthonormal basis $\boldsymbol{I} \in \mathbb{R}^2$, the 2-dimensional identity matrix, such that the vector $\boldsymbol{x}(t)$ observed through this basis standard basis is equal to itself. This concept provides the possibility to associate any other arbitrary basis to represent a state-vector and visualize the states through this perspective. When the basis is not orthogonal, a change in the state-vector with a direction parallel to a component of the basis, i.e., a change in only one element of this vector, produces a change in other directions if observed through the original orthonormal basis. This is an interesting result, since it shows the interactions between two state

variables through a basis, making it possible to observe mutual changes in those variables from a single direction. The right side of Fig. 2.9 illustrates the vector $\boldsymbol{x}(t)$ as referenced through the basis $\boldsymbol{Q} = [\boldsymbol{q}_1, \boldsymbol{q}_2]^T$, for $\boldsymbol{q}_1 = [3, 1]^T$ and $\boldsymbol{q}_1 = [2, 2]^T$, given as:

$$\boldsymbol{x}(t) = \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 3 & 2 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \boldsymbol{G}\boldsymbol{z}(t). \tag{2.54}$$

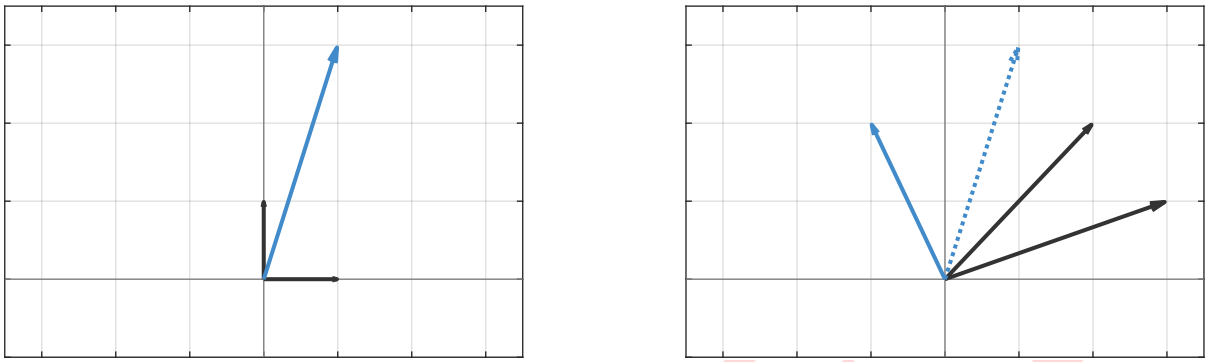


Figure 2.9: Visualization of a basis transformation applied to a vector.

In the State-Space formulation, the matrix $\boldsymbol{A}$ represents a linear function that maps state-vectors from $\mathbb{R}^n$ to itself. When applying a new basis to represent the state-vectors, it is intuitive that the mapping performed by this function also changes so that it still represents the same linear combination of the states.

Theorem 2.10. *(Similarity Transformation) Consider a system in SS representation described by the matrices* $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$ *and a nonsingular transformation matrix* $\boldsymbol{P} \in \mathbb{R}^{n \times n}$*. An equivalent representation for the transformation* $\boldsymbol{z}(t) = \boldsymbol{P}\boldsymbol{x}(t)$ *is:*

$$\begin{cases} \dot{\boldsymbol{z}}(t) = \tilde{\boldsymbol{A}}\boldsymbol{z}(t) + \tilde{\boldsymbol{B}}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \tilde{\boldsymbol{C}}\boldsymbol{z}(t) + \tilde{\boldsymbol{D}}\boldsymbol{u}(t) \end{cases}, \tag{2.55}$$

where:

$$\tilde{\boldsymbol{A}} = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1} \qquad \tilde{\boldsymbol{B}} = \boldsymbol{P}\boldsymbol{B} \qquad \tilde{\boldsymbol{C}} = \boldsymbol{C}\boldsymbol{P}^{-1} \qquad \tilde{\boldsymbol{D}} = \boldsymbol{D}. \tag{2.56}$$

Proof. Consider a SS representation and any nonsingular matrix $\boldsymbol{P} \in \mathbb{R}^{n \times n}$. Making $\boldsymbol{z}(t) = \boldsymbol{P}\boldsymbol{x}(t)$ leads to $\boldsymbol{x}(t) = \boldsymbol{P}^{-1}\boldsymbol{z}(t)$. Substituting this in the state equation results in:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{P}^{-1}\dot{\boldsymbol{z}}(t) &= \boldsymbol{A}\boldsymbol{P}^{-1}\boldsymbol{z}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{I}\dot{\boldsymbol{z}}(t) &= \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1}\boldsymbol{z}(t) + \boldsymbol{P}\boldsymbol{B}\boldsymbol{u}(t) \\ \dot{\boldsymbol{z}}(t) &= \tilde{\boldsymbol{A}}\boldsymbol{z}(t) + \tilde{\boldsymbol{B}}\boldsymbol{u}(t) \end{aligned}. \tag{2.57}$$

Thus, substituting $\boldsymbol{x}(t) = \boldsymbol{P}^{-1}\boldsymbol{z}(t)$ in the output equation results in:

$$\begin{aligned} \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{P}^{-1}\boldsymbol{y}(t) + \boldsymbol{D}\boldsymbol{u}(t). \\ \boldsymbol{y}(t) &= \tilde{\boldsymbol{C}}\boldsymbol{z}(t) + \tilde{\boldsymbol{D}}\boldsymbol{u}(t) \end{aligned} \tag{2.58}$$

□

By this theorem is it clear that, after transforming the state-vector, the entire dynamical model $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$ changes. This would imply that the analysis of the original system, for its properties and time response, is not valid for the similar transformed system. However, as discussed before, these transformations accounts for the same system when observed through a different reference basis. Therefore, it is expected that the model presents the same analysis results, as demonstrated below.

Theorem 2.11. *Consider a system in State-Space form with matrix $\boldsymbol{A}$. Consider also a similarity transformation $\boldsymbol{z}(t) = \boldsymbol{P}\boldsymbol{x}(t)$ that results in a similar matrix $\tilde{\boldsymbol{A}} = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1}$. In this case, $\boldsymbol{A}$ and $\tilde{\boldsymbol{A}}$ have the same set of eigenvalues.*

Proof. The eigendecomposition problem is defined as $\boldsymbol{A}\boldsymbol{v} = \boldsymbol{\lambda}\boldsymbol{v}$. From the similarity transformation, $\tilde{\boldsymbol{A}} = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1}$ leads to $\boldsymbol{A} = \boldsymbol{P}^{-1}\tilde{\boldsymbol{A}}\boldsymbol{P}$. Substituting this in the eigendecomposition:

$$\begin{aligned} \boldsymbol{P}^{-1}\tilde{\boldsymbol{A}}\boldsymbol{P}\boldsymbol{v} &= \boldsymbol{\lambda}\boldsymbol{v} \\ \tilde{\boldsymbol{A}}\boldsymbol{P}\boldsymbol{v} &= \boldsymbol{\lambda}\boldsymbol{P}\boldsymbol{v} \end{aligned}. \tag{2.59}$$

Considering the transformed eigenvector $\tilde{\boldsymbol{v}} = \boldsymbol{P}\boldsymbol{v}$, it is clear that $\tilde{\boldsymbol{A}}\tilde{\boldsymbol{v}} = \boldsymbol{\lambda}\tilde{\boldsymbol{v}}$, implying that the matrices $\boldsymbol{A}$ and $\tilde{\boldsymbol{A}}$ shares the same set of eigenvalues $\boldsymbol{\lambda}$. □

It is clear from past results that the fact that both the matrices shares the same set of eigenvalues directly implies that they provide the same dynamical responses and, as will be shown later, the same general properties. Therefore, the similarity transformation consists in a method to produce new State-Spaces representations that emphasizes some geometrical perspective of the model, hopefully in some perspective that helps analyze a specific property, without actually changing the relationship of the original model with the physical system.

The use of similarity transformations can also benefits the computation of functions of the matrices of the State-Space representation, given that they impose a desirable structure to this matrix. With this motivation, a popular transformation that provides a new representation with computational advantages is the Similarity transformation.

Theorem 2.12. *(Diagonalization) Consider a n-dimensional system in State-Space form represented by the matrices $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$, such that the matrix $\boldsymbol{A}$ has n distinct real eigenvalues, i.e., $\boldsymbol{\lambda} \in \mathbb{R}^n$. Performing the transformation $\boldsymbol{z}(t) = \boldsymbol{V}\boldsymbol{x}(t)$, where $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_n]$ is the modal matrix composed by the eigenvectors $\boldsymbol{v}_i \in \mathbb{R}^n$, $i \in [1, ..., n]$, of matrix $\boldsymbol{A}$, the resulting transformed matrix $\boldsymbol{\Lambda} = \boldsymbol{V}\boldsymbol{A}\boldsymbol{V}^{-1}$ is diagonal.*

Proof. Since the eigenvalues are real and distinct, the eigenvectors must be linearly independent, proving that the inverse $\boldsymbol{V}^{-1}$ always exist and that it is a feasible transformation matrix. Using

the identity for the eigendecomposition of matrix $\boldsymbol{A}$:

$$\begin{aligned}
\boldsymbol{\lambda v} &= \boldsymbol{Av} \\
\begin{bmatrix}\lambda_1\boldsymbol{v}_1 & \lambda_2\boldsymbol{v}_2 & \cdots & \lambda_n\boldsymbol{v}_n\end{bmatrix} &= \begin{bmatrix}\boldsymbol{A}\boldsymbol{v}_1 & \boldsymbol{A}\boldsymbol{v}_2 & \cdots & \boldsymbol{A}\boldsymbol{v}_n\end{bmatrix} \\
\begin{bmatrix}\boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n\end{bmatrix}\begin{bmatrix}\lambda_1 & 0 & \cdots & 0\\ 0 & \lambda_2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \lambda_n\end{bmatrix} &= \boldsymbol{A}\begin{bmatrix}\boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n\end{bmatrix} \\
\begin{bmatrix}\lambda_1 & 0 & \cdots & 0\\ 0 & \lambda_2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \lambda_n\end{bmatrix} &= \begin{bmatrix}\boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n\end{bmatrix}^{-1}\boldsymbol{A}\begin{bmatrix}\boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n\end{bmatrix} \\
\boldsymbol{\Lambda} &= \boldsymbol{V}^{-1}\boldsymbol{AV}
\end{aligned} \quad . \tag{2.60}$$

Concluding that $\boldsymbol{\Lambda}$ is a diagonal matrix which elements are the eigenvalues of matrix $\boldsymbol{A}$. □

This result can be easily extended to the case where the eigenvalues are conjugate complex pairs, but each pair is distinct. A model with a diagonal matrix $\boldsymbol{A}$ has the nice property that the evolution of the states are decoupled, in the sense that each state evolution is a linear function of itself. A geometrical interpretation of this transformation is that the eigenvectors of this matrix produces a basis that encode information about the interaction between those states, while the interaction in the original formulation was the linear combination of the modes.

In this diagonalization procedure, the resulting elements of the matrix $\boldsymbol{\Lambda}$ are the very own eigenvalues of the original matrix $\boldsymbol{A}$, which allows for a direct interpretation of the system response by just visualizing this matrix. Furthermore, it is easy to verify that the state-transition matrix for the transformed matrix, $e^{\boldsymbol{\Lambda}t}$, is also a diagonal matrix whose elements are the modes of the system, and can be easily computed:

$$\begin{aligned}
e^{\boldsymbol{\Lambda}t} = \sum_{k=0}^{\infty}\frac{A^k t^k}{k!} = \sum_{k=0}^{\infty}\frac{t^k}{k!}\begin{bmatrix}\lambda_1^k & 0 & \cdots & 0\\ 0 & \lambda_2^k & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \lambda_n^k\end{bmatrix} &= \begin{bmatrix}\frac{t^k\lambda_1^k}{k!} & 0 & \cdots & 0\\ 0 & \frac{t^k\lambda_2^k}{k!} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{t^k\lambda_n^k}{k!}\end{bmatrix} \\
&= \begin{bmatrix}e^{\lambda_1 t} & 0 & \cdots & 0\\ 0 & e^{\lambda_2 t} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & e^{\lambda_n t}\end{bmatrix}
\end{aligned} . \tag{2.61}$$

In the case that the eigenvalues are not all distinct, it may be not possible to design a modal matrix in the same way, since the eigenvectors could not be all linearly independent, and the matrix would not form a basis. In those cases, however, it is still possible to design a generalized modal matrix that transform the original matrix $\boldsymbol{A}$ to a quasi-diagonal matrix $\boldsymbol{J}$, where there will be decoupled block of states. This similarity transformation is known as the Jordan form [Strang, 2016], and it generalizes the notion of diagonalization for any arbitrary matrix.

2.5 Stability, Controlability and Observability

When moving from a discussion of the models in the perspective of dynamical system analysis to a perspective of control theory, it is necessary to define and analyze some important properties of a system. These properties are characteristic to the system, analyzed through a model, but they account directly to questions relating the control objectives and the instrumentation, as it will be shown later.

The first property to be discussed consists in the stability of a system. An instable system, as the name suggests, is a system whose response does not converge to a specific value and rather oscillates or grows unbounded. In physical scenarios, unstable systems are problematic, since their response to external stimuli can result in dangerous situations to itself and, maybe, to the environment around it. Because of this, determining the stability of a system is a crucial procedure into analyzing a system that will be controlled. Under the several quantitative methods to determine if a system is indeed stable, given a mathematical model, a popular and practical one is the Bounded-Input Bounded-Output (BIBO) stability criteria.

Definition 2.6. (BIBO Stability) A dynamical system is defined as BIBO stable if every bounded input stimuli $|\boldsymbol{u}(t)| \leq \epsilon < \infty$ produces in it a bounded output response $|\boldsymbol{y}(t)| \leq \delta < \infty$.

The main result behind this criteria is that the natural response of a system should vanish as time evolves, i.e., $\boldsymbol{x}(t) = \boldsymbol{0}$ as $t \to \infty$. This result is very intuitive from Theorem 2.7, since the vanishing of the natural response implies that $e^{\boldsymbol{A}t} = \boldsymbol{0}$ as $t \to \infty$ and the forced response is expected to be bounded, if $\boldsymbol{u}(t)$ is bounded. Since the state-transition matrix is a linear combination of the modes, and the modes are exponential functions of the eigenvalues of the matrix $\boldsymbol{A}$, it is possible to determine a condition for stability in the light of these quantities.

Theorem 2.13. *(BIBO Stability in SS) A system in State-Space form, represented by a matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ with eigenvalues $\boldsymbol{\lambda} \in \mathbb{R}^n$, is BIBO stable if and only if $\mathbb{R}e[\lambda_i] < 0, \ \forall i \in [1, 2, ..., n]$.*

A detailed proof of this theorem can be found in Appendix A. First of all, note that this criteria depends only on the eigenvalues of matrix $\boldsymbol{A}$, so the stability property of a system is invariant to any similarity transformation, since $\boldsymbol{A}$ and any transformed matrix $\tilde{\boldsymbol{A}} = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1}$ shares the same eigenvalues. From the previous results it is also known that the real part of the eigenvalues, independent of their multiplicity or domain, appears as the arguments of the exponential functions that are the system modes. Therefore, an eigenvalue with a negative real part will produce a mode that is a exponential decay, as this theorem indicates. By the same argument, if the matrix $\boldsymbol{A}$ has at least one eigenvalue $\lambda_j = 0$ such that $\mathbb{R}e[\lambda_i] \leq 0, \ \forall i \in [1, 2, ..., n]$, then the mode associated with this eigenvalue is a constant and the natural response becomes bounded. This configuration is known as a marginally stable condition, in the BIBO perspective. The time response of a 2-nd order system is shown Fig. 2.10 for three different poles configurations, given an unitary step.

Later chapters will discuss the possibility of stabilize an unstable system through a controller. There are, however, some restrictions to the possibility of controlling or not the states of a system, which includes the necessity of discussing *controllability*. The controllability of a system states whether it is possible to calculate an input signal that drive the system to any arbitrary state or not, given some time restriction. This property accounts exclusively for this possibility, in the sense that it does not account for the operational feasibility of actually applying this input signal into a physical system, since it may need more energy than an actuator can produce.

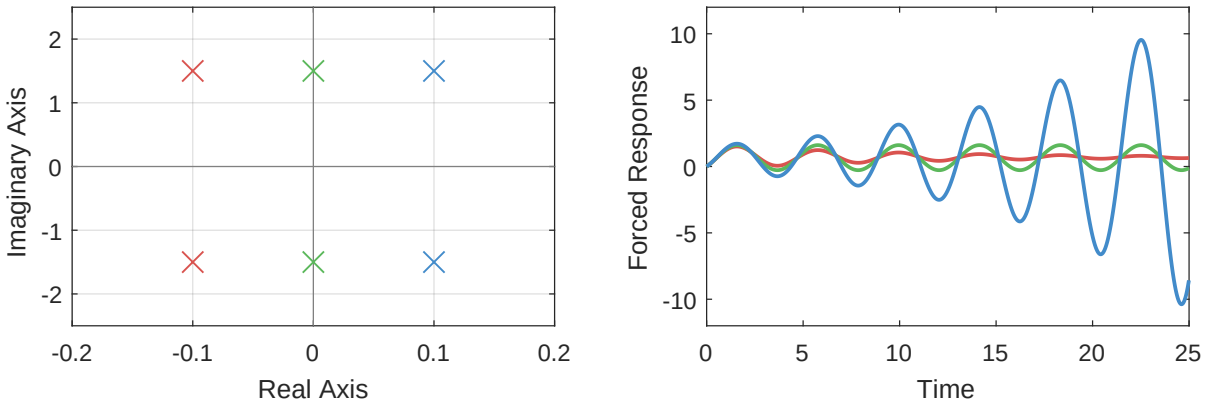


Figure 2.10: Stability of forced responses given the positions of the system poles.

Definition 2.7. (Controllability) A system in State-Space form with matrices $(\boldsymbol{A}, \boldsymbol{B})$ is said to be controllable if, for any initial state $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ and terminal state $\boldsymbol{x}(T) = \boldsymbol{x}_T$, $T < \infty$, there exists an input signal $\boldsymbol{u}(t)$, $t \in [t_0, T]$, that can transfer $\boldsymbol{x}(t_0)$ to $\boldsymbol{x}(T)$. Otherwise, the system is said to be uncontrollable.

There are several methods to analyze the controllability of a system given a mathematical model and the definition above. A popular criteria introduces the concept of a controllability matrix and has a nice geometrical interpretation.

Theorem 2.14. *(Controllability in SS) Consider a system in linear State-Space form with matrices* $(\boldsymbol{A}, \boldsymbol{B})$ *and the controllability matrix* $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{n \times nr}$ *defined as:*

$$\boldsymbol{\mathcal{C}} = \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \boldsymbol{A}^2\boldsymbol{B} & \cdots & \boldsymbol{A}^{n-1}\boldsymbol{B} \end{bmatrix}. \tag{2.62}$$

The system is controllable if and only if $\boldsymbol{\mathcal{C}}$ *has full row rank.*

An intuition behind this theorem is that the full row rank condition implies that $\mathcal{C}$ has n linearly independent columns, therefore these columns can be used as a basis for what is known as the *controllable subspace*. To better understand that, consider the following forced solution $\boldsymbol{x}_f(t)$ given by the Lagrange formula:

$$\boldsymbol{x}_f(t) = \int_0^t e^{\boldsymbol{A}(t-\tau)} \boldsymbol{B}\boldsymbol{u}(\tau) d\tau. \tag{2.63}$$

From the Cayley-Hamilton theorem, shown in the Sylvester expansion at Theorem 2.8, it is possible to represent $e^{\boldsymbol{A}(t-\tau)}$ as a linear combination of scalars $\beta_i(t-\tau)$ and powers of the matrix $\boldsymbol{A}^i$, $i \in [0, 1, ..., n-1]$. Thus, using this theorem and substituting $\tau_2 = t - \tau$ for an easier manipulation, the forced response can be represented as:

$$\begin{aligned} \boldsymbol{x}_f(t) &= \int_0^t \left(\sum_{i=0}^{n-1} \beta_i(\tau_2) \boldsymbol{A}^i \right) \boldsymbol{B}\boldsymbol{u}(t-\tau_2) d\tau_2 = \sum_{i=0}^{n-1} \left(\boldsymbol{A}^i \boldsymbol{B} \right) \int_0^t \beta_i(\tau_2) \boldsymbol{u}(t-\tau_2) \\ &= \sum_{i=0}^{n-1} \left(\boldsymbol{A}^i \boldsymbol{B} \right) \tilde{\beta}_i(\boldsymbol{u}, t) \end{aligned}, \tag{2.64}$$

where $\boldsymbol{A}^i\boldsymbol{B}$ are the columns of the matrix $\boldsymbol{\mathcal{C}}$ and $\tilde{\beta}_i(\boldsymbol{u}, t)$ is a function that depends only on the input signal $\boldsymbol{u}(t)$ and time t. This result implies that the unforced response $\boldsymbol{x}_f(t)$ is a linear combination given by the columns of $\boldsymbol{\mathcal{C}}$. If $\boldsymbol{\mathcal{C}}$ has n linearly independent columns, than this linear combination spans the entire n-dimensional space, i.e., the entire state space, and thus any desirable state vector $\boldsymbol{x}^*$ can be reached. If the column rank of $\boldsymbol{\mathcal{C}}$ is less than n, then only a subspace of smaller dimension can be reached through the forced response.

While the discussion on controllability concerns the possibility of driving a system to a desirable state through an actuator signal, there is also the necessity to discuss the possibility of determining the internal state of a system given the output signal $\boldsymbol{y}(t)$. This property, known as *observability*, comes from the fact that any output of a system, related to the states through the matrix $\boldsymbol{C}$, may be a combination of states, and that some states may not even be present in the output signal. In conclusion, it is necessary to know if it is possible to reconstruct $\boldsymbol{x}(t)$ directly through $\boldsymbol{y}(t)$.

Definition 2.8. Observability A system in State-Space form with matrices $(\boldsymbol{A}, \boldsymbol{C})$ is said to be observable if, given an input signal $\boldsymbol{u}(t)$ and output signal $\boldsymbol{y}(t)$, over the interval $t \in [t_0, T]$, it is possible to uniquely determine the value of the initial state $\boldsymbol{x}(t_0)$. Otherwise, the system is said to be unobservable.

Similarly to the controllability property, there are several ways to analyze the observability of a system, given a mathematical model. A popular criteria introduces the concept of a observability matrix.

Theorem 2.15. *(Observability in SS) Consider a system in linear State-Space form with matrices* $(\boldsymbol{A}, \boldsymbol{C})$ *and the observability matrix* $\boldsymbol{\mathcal{O}} \in \mathbb{R}^{nq \times n}$ *defined as:*

$$\boldsymbol{\mathcal{O}} = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \boldsymbol{CA}^2 \\ \vdots \\ \boldsymbol{CA}^{n-1} \end{bmatrix}. \tag{2.65}$$

The system is observable if and only if $\boldsymbol{\mathcal{O}}$ *has full column rank.*

The interpretation of this theorem follows the same intuition of before: if the matrix $\boldsymbol{\mathcal{O}}$ has full column rank, then it can be used as a basis to span a subspace with the same dimension as the State-Space. In fact, the proof of both theorems follows the same procedures and there is a direct relationship between controllability and observability, known as the Theorem of Duality.

The concepts of controllability and observability just presented, together with the conditions to fulfill these properties, were first introduced by [Kalman, 1960]. The geometrical interpretations of both theorems may suggest a practical solution for the cases where a system is uncontrollable or unobservable. In the first case, the solution would be to add specific actuators to the system, in the condition that they are linearly independent between themselves and the ones actually in operation. The same procedure can be done to solve the unobservable problem, but adding more sensors instead. Those procedures would change the matrices $\boldsymbol{B}$ and $\boldsymbol{C}$, without changing the system dynamics, and could ensure the necessary conditions in matrices $\boldsymbol{\mathcal{C}}$ and $\boldsymbol{\mathcal{O}}$. Of course, the implementation of such instrumentation may not be practical, due to technical or economical

constraints. Of course, if including these additional devices is not feasible, one can still control and observe a given subset of the state variables.

2.6 Response Analysis in the Frequency Domain

Although the response of dynamical systems are naturally perceived in time, there are advantages of analyzing the models in a frequency domain perspective. This analysis differs from a simple time domain analysis from the fact that, in a steady-state regime, the response of a linear system for a sinusoidal input is itself sinusoidal, with the same frequency but different amplitude and phase, as illustrated at Fig. 2.11.

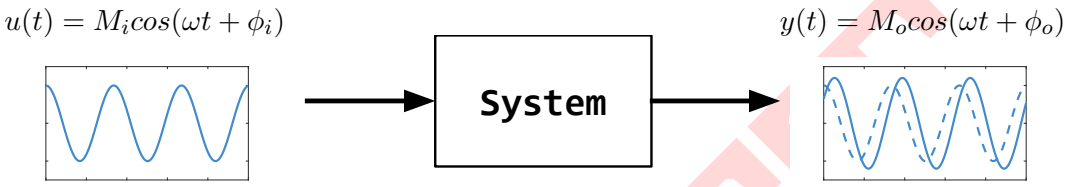


Figure 2.11: Illustration of the steady-state response of LTI systems to sinusoidal inputs.

A common representation is the phasor representation, where $M\angle\phi = Mcos(\omega t + \phi)$. In the context of SISO Input-Output models, the response of the system can be summarized by a transfer function which is also a phasor:

$$g(t) = \frac{y(t)}{u(t)} = \frac{M_o\angle\phi_o}{M_i\angle\phi_i} = M_g\angle\phi_g, \tag{2.66}$$

where $M_g = M_o/M_i$ and $\phi_g = \phi_o - \phi_i$. Notice that this formulation makes the time-dependance implicit in the system response, since it is periodic in this case. For this reason, the system response can be visualized as a function of frequency rather than a function of time, and the properties of the system can be accessed in this way. The two most popular techniques for frequency response analysis are Bode plots [Bode, 1945] and Nyquist diagrams [Nyquist, 1932]. The first is a direct plot of $M_g(\omega)$ and $\phi_g(\omega)$ for several values of ω, making $M_g(\omega) = 20log\,|G(j\omega)|$ and $\phi_g(\omega) = \angle G(j\omega)$, where $G(j\omega)$ is a transfer function of a system evaluated for an input signal with exclusively oscillatory components. The Nyquist diagram, in the other hand, is a direct phasor visualization obtained by applying the Argument Principle to a contour containing the entire right-hand side of the complex plane. Both visualizations are depicted in Fig. 2.12, for a Transfer function obtained by applying the transformation of Theorem 2.4 into the system from (2.46):

$$\boldsymbol{G}(s) = \begin{bmatrix} \dfrac{3.81}{s+5.92} & \dfrac{-1.09s-3.33}{s^2+10.62s+27.84} \end{bmatrix}. \tag{2.67}$$

The immediate advantage of these visualizations is that is not necessary to compute an entire simulation of this system, until it reaches steady-state, to be able to perform analysis, which is really critic for high-dimensional systems with slow time-constants. In addition to that, these visualizations (specially the Bode plot) can be easily sketched by hand with fairly accuracy, allowing for some understanding of the system without need for solving differential equations or the inverse Laplace transforms. For these reasons, frequency responses methods for analyzing systems were ubiquitous for many years in industry applications, and some properties assessments, such as closed-loop stability, are still better understood under this formulation, as it will be shown in later chapters.

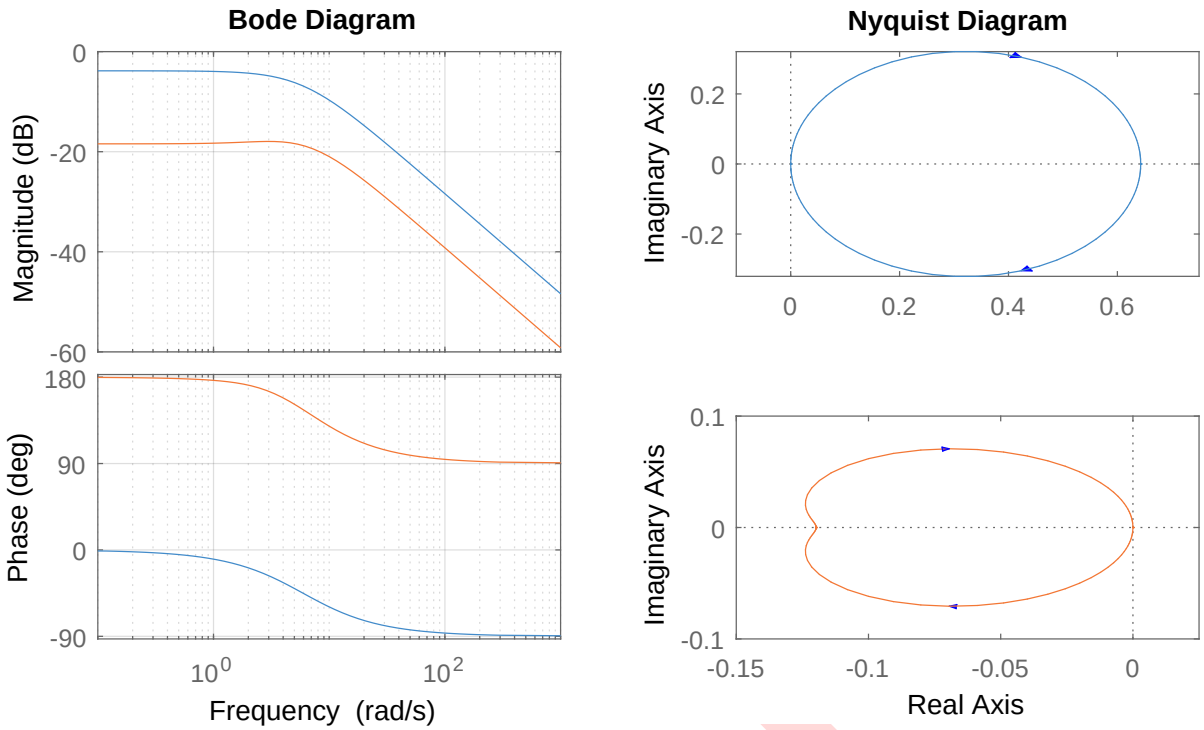


Figure 2.12: Visualization of the two-states system in (2.67) in Bode plots (left) and Nyquist diagrams (right). The blue and orange lines represents the states x_1 and x_2, respectively.

Chapter 3

Controller Synthesis

This chapter discusses the general results and properties for the design of dynamical controllers, focusing on feedback architectures. The devices are motivated and formulated using the State-Space model for dynamical systems, so the feedback is performed on the state response rather than the system output. For this reason, the results in this section focus on these equations, where the output equations is made implicit.

3.1 State Feedback Controllers

In modern control theory, advances in computer performance and in the methods themselves have made the design of controllers using State-Space models feasible for real-world applications. This is usually desirable since, as shown in the latter chapter, these kind of models provides a practical solution to understand dynamical system response and properties, so it is natural to want a controller design technique that accounts for that representation. The most basic, yet most popular, feedback controller used in those settings is the *Full-State Feedback Controller*, defined below.

Definition 3.1. Full-State Feedback Given a linear system in State-Space representation, the input signal $\boldsymbol{u}(t)$ is calculated through feedback of the states as application of the linear control law:

$$\boldsymbol{u(t)} = \pi(\boldsymbol{r}, \boldsymbol{x}, t) = \boldsymbol{r}(t) - \boldsymbol{K}\boldsymbol{x}(t), \tag{3.1}$$

where $\boldsymbol{r} : \mathbb{R} \to \mathbb{R}^n$ is a state reference signal that the system must follows and $\boldsymbol{K} \in \mathbb{R}^{r \times n}$ is the *feedback gain matrix*.

The control law $\pi(\cdot)$ is linear and time-invariant, which makes the analysis of the closed-loop system similar to the one used in open-loop configurations. A schematic of the closed-loop system is shown at Fig. 3.1. Of course, this is a choice of control law, and feedback controllers can also be defined using nonlinear or time-dependent functions. Notice that this new definition for the calculation of $\boldsymbol{u}(t)$ allows for the following closed-loop representation of the system:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\left(\boldsymbol{r}(t) - \boldsymbol{K}\boldsymbol{x}(t)\right) \\ \dot{\boldsymbol{x}}(t) &= \left(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}\right)\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{r}(t) \end{aligned}. \tag{3.2}$$

From this is clear that the inclusion of a feedback controller in the loop is equivalent to transform an open-loop system into a new system $(\boldsymbol{A}_{cl}, \boldsymbol{B})$, with $\boldsymbol{A}_{cl} = \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$, whose manipulated variables is a reference signal $\boldsymbol{r}(t)$ but the controlled variables are still the same. Since $\boldsymbol{K}$ is an arbitrary matrix, it is possible to change the behavior of the closed-loop system,

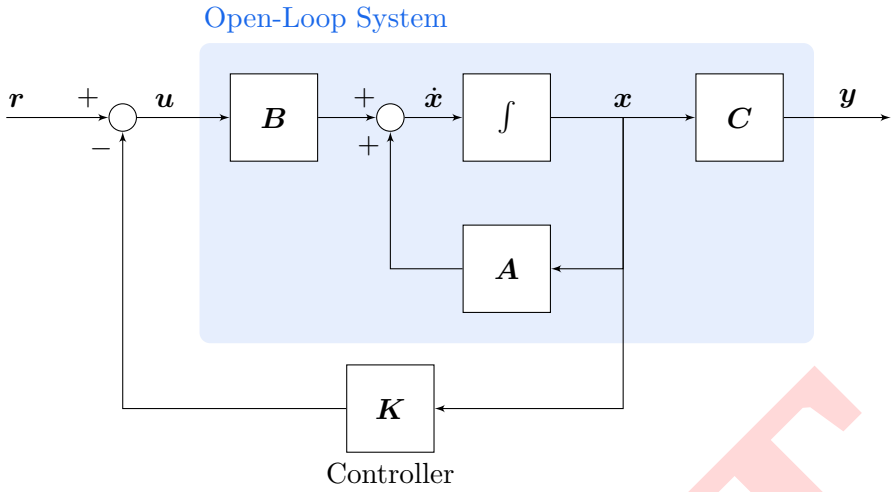


Figure 3.1: Block diagram of a state-feedback closed-loop system.

and consequently the controlled variables response, by designing this matrix. To understand better the capabilities of the state feedback, consider the following theorems.

Theorem 3.1. *(Controller Canonical Form) If a SISO system in State-Space representation is controllable, then by applying the transformation* $\boldsymbol{z}(t) = \boldsymbol{P}\boldsymbol{x}(t)$*, for a matrix* $\boldsymbol{P}$ *calculated as the inverse of:*

$$\boldsymbol{P}^{-1} = \boldsymbol{\mathcal{C}} \begin{bmatrix} 1 & \alpha_1 & \cdots & \alpha_{n-1} \\ 0 & 1 & \cdots & \alpha_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}, \tag{3.3}$$

where $[\alpha_1, \alpha_2, ..., \alpha_{n-1}]$ *are the* $n-1$ *first coefficients of the characteristic polynomial* $\Delta(s) = det(s\boldsymbol{I} - \boldsymbol{A})$*, the resulting representation is in the controller canonical form given as:*

$$\begin{cases} \dot{\boldsymbol{z}}(t) = \begin{bmatrix} -\alpha_1 & -\alpha_2 & \cdots & -\alpha_{n-1} & -\alpha_n \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \boldsymbol{z}(t) + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_n \end{bmatrix} \boldsymbol{z}(t) \end{cases}. \tag{3.4}$$

Details for the proof of this theorem can be found in [**reference**]. An equivalent result can be proved for MIMO systems [**reference**], but the result is very verbose and does not highlight the main results for the state-feedback. Using the canonical form just presented, it is possible to discuss a strong result for the state feedback controller.

Theorem 3.2. *(Pole-Placement Method) If a system in State-Space representation is controllable, then by state feedback using a gain matrix* $\boldsymbol{K} \in \mathbb{R}^{r \times n}$ *the eigenvalues of* $\boldsymbol{A}_{cl} = \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$*, the poles of the closed-loop system, can arbitrarily be assigned anywhere in the complex plane, given that complex conjugate eigenvalues are assigned in pairs.*

Proof. Consider that the system is controllable. In this case, it can be converted to the controller canonical form of Theorem 3.1. Substituting $\boldsymbol{z}(t) = \boldsymbol{P}\boldsymbol{x}(t)$ results in the following control law for the state feedback:

$$u(t) = r(t) - \boldsymbol{K}\left(\boldsymbol{P}^{-1}\boldsymbol{z}(t)\right) = r(t) - \tilde{\boldsymbol{K}}\boldsymbol{z}(t). \tag{3.5}$$

Applying the state feedback, the transformed closed-loop is given by:

$$\tilde{\boldsymbol{A}}_{cl} = \boldsymbol{P}(\boldsymbol{A} - B\boldsymbol{K})\boldsymbol{P}^{-1} = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1} - \boldsymbol{P}B\boldsymbol{K}\boldsymbol{P}^{-1} = \tilde{\boldsymbol{A}} - \tilde{B}\tilde{\boldsymbol{K}}. \tag{3.6}$$

From Theorem 2.11 it is known that $\boldsymbol{A}_{cl}$ and $\tilde{\boldsymbol{A}}_{cl}$ shares the same set of eigenvalues, and, therefore, the same characteristic equations. Consider this characteristic equation in the form:

$$\Delta(s) = det(s\boldsymbol{I} - \boldsymbol{A}) = s^n + \alpha_1 s^{n-1} + \alpha_2 s^{n-2} + \cdots + \alpha_{n-1}s + \alpha_n. \tag{3.7}$$

Given a desired a set of coefficients $[\tilde{\alpha}_1, \tilde{\alpha}_2, ..., \tilde{\alpha}_n]$ of a characteristic polynomial whose roots are the desired closed-loop eigenvalues, define the transformed feedback gain matrix as:

$$\tilde{\boldsymbol{K}} = \begin{bmatrix} \tilde{\alpha}_1 - \alpha_1 & \tilde{\alpha}_2 - \alpha_2 & \cdots & \tilde{\alpha}_n - \alpha_n \end{bmatrix}. \tag{3.8}$$

Plugging this in (3.6), it is easy to see that the resulting representation is:

$$\begin{cases} \dot{\boldsymbol{z}}(t) = \begin{bmatrix} -\tilde{\alpha}_1 & -\tilde{\alpha}_2 & \cdots & -\tilde{\alpha}_{n-1} & -\tilde{\alpha}_n \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \boldsymbol{z}(t) + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t) \\ y(t) = \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_n \end{bmatrix} \boldsymbol{z}(t) \end{cases}, \tag{3.9}$$

whose characteristic polynomial is now described by the designed coefficients to yield the desirable eigenvalues. Since $\tilde{\boldsymbol{A}}_{cl}$ and $\boldsymbol{A}_{cl}$ shares the same set of eigenvalues, it is concluded that it is possible to assign the system poles directly through matrix $\tilde{\boldsymbol{K}}$. □

Notice, from that procedure, that an "original" feedback gain matrix can be obtained as $\boldsymbol{K} = \tilde{\boldsymbol{K}}\boldsymbol{P}$ and still yield the same eigenvalues assignment directly in $\boldsymbol{A}_{cl}$ (since $\boldsymbol{P}$ is just a linear transformation). This theorem has the direct result that, under full-state feedback, the transient response of a linear system can be completely determined by including a controller, which is described by this matrix $\boldsymbol{K}$. This result is still preserved for MIMO systems, although the design of $\boldsymbol{K}$ is not so straightforward since it is not unique for a desired set of eigenvalues in this case [**reference**].

Using the parameters from Definition 2.5, the positions of the closed-loop system poles can be evaluated given desirable operations, and the matrix $\boldsymbol{K}$ can be hand-designed to meet these requirements. This method is known as the *Pole-Placement method* for control synthesis. The algorithm below summarizes a simple procedure of designing an appropriate feedback gain matrix given desirable pole positions.

Albeit being a simples formula, this method can be used in several applications to yield controllers capable of matching performance requisites. Of course, the designer must take into account that, besides the eigenvalue assignment allows for the whole complex plane, a careless choice of eigenvalues could result in unpractical controllers, with very aggressive or oscillatory input signals. For this reason, it is necessary some knowledge of the instruments limits before designing the matrix $\boldsymbol{K}$.

Algorithm 1: Pole-Placement Method for SISO Systems

Input : state-space model $(\boldsymbol{A}, B)$ and a set of n desired eigenvalues $\boldsymbol{\lambda}^*$.
Output : feedback gain matrix $\boldsymbol{K}$.

1 Calculate $[\alpha_1, \alpha_2, ..., \alpha_n]$ as the coefficients of the polynomial $\Delta(s) = det(s\boldsymbol{I} - \boldsymbol{A})$;
2 Let $\boldsymbol{P}^{-1} = \begin{bmatrix} B & \boldsymbol{A}B & \boldsymbol{A}^2B & \cdots & \boldsymbol{A}^{n-1}B \end{bmatrix} \begin{bmatrix} 1 & \alpha_1 & \cdots & \alpha_{n-1} \\ 0 & 1 & \cdots & \alpha_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$;
3 Let $\boldsymbol{P} = \left(\boldsymbol{P}^{-1}\right)^{-1}$;
4 Calculate $[\tilde{\alpha}_1, \tilde{\alpha}_2, ..., \tilde{\alpha}_n]$ as the coefficients of the polynomial $\Delta_{cl}(s) = \prod_{i=1}^{n}(s - \lambda_i^*)$;
5 Let $\tilde{\boldsymbol{K}} = \begin{bmatrix} \alpha_1 - \tilde{\alpha}_1 & \alpha_2 - \tilde{\alpha}_2 & \cdots & \alpha_n - \tilde{\alpha}_n \end{bmatrix}$;
6 Return $\boldsymbol{K} = \tilde{\boldsymbol{K}}\boldsymbol{P}$

3.2 Regulation and Reference Tracking

When discussing controller synthesis it is also necessary to account for which objective this device is expected to fulfill. In this case, there are two main classifications of controllers based on the operation that they impose to the system: regulators and tracking (or servo) controllers. In the case of state-feedback, these two classes of controllers differs only by what type of reference signal $\boldsymbol{r}(t)$, for an operation in a time interval $t \in [t_0, t_f]$, the system is expected to follow. The following statements gives a formal definition of a controller for regulation:

Definition 3.2. (Regulator) If a state-feedback controller has to make a system follows the reference $\boldsymbol{r}(t) = \boldsymbol{0}$, as $t \to \infty$, it is said to be a *regulator*. In this case, the closed-loop state equation and equivalent solutions reduces to the following:

$$\begin{array}{ll} \textbf{State Equation:} & \textbf{Lagrange solution:} \\ \dot{\boldsymbol{x}}(t) = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\,\boldsymbol{x}(t) & \boldsymbol{x}(t) = e^{(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K})t}\boldsymbol{x}(t_0) \end{array}. \tag{3.10}$$

Of course, if the feedback gain matrix impose that all poles of the system are in the left-half plane, the closed-loop is stable and the natural response will eventually converge to zero (Theorem 2.13). Therefore, all stable feedback controllers are able to impose regulation to a system, and the characteristics of the transient response can be fully determined by the matrix $\boldsymbol{K}$. These type of controllers are used to make systems goes from nonzero initial states to the zero-state $\boldsymbol{x}(t) = \boldsymbol{0}$ and stays there, meaning that the the controller can also be used to reject disturbances. Now, one may wonder if this operation is too restrictive in the sense that the zero-state is not the desirable state in many control objectives, as is the case of reactor systems: a zero-state means that no chemical substances are being produced. However, it is common to have linear systems that are linearized versions of nonlinear models, using the approximation from Theorem 2.5, meaning that the regulator actually is to impose $\Delta\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_o = \boldsymbol{0}$, i.e., drives the system to the steady-state point $\boldsymbol{x}_o$ and reject disturbances. For the sake of illustration, considers the first system of (2.51), repeated here:

$$\boldsymbol{A} = \begin{bmatrix} -5.93 & 0 \\ 0.83 & -4.70 \end{bmatrix} \quad b = \begin{bmatrix} 3.81 \\ -1.09 \end{bmatrix}. \tag{3.11}$$

Since this system was obtained by the linearized model in (2.34) for $\boldsymbol{x}_o = [6.19, 1.09]^T$ and $u_o = 3.03$, the regulator will impose the zero-state $\Delta\boldsymbol{x} = \boldsymbol{0}$ which actually means imposing

$\boldsymbol{x}(t) = [6.19, 1.09]^T$ as $t \to \infty$. A simulation is shown in Fig. 3.2 for three different regulators, considering initial state $\delta\boldsymbol{x}(t) = \mathbf{0} - \boldsymbol{x}_o$ and vertically correcting the response and input signal by $\boldsymbol{x}_o$ and $\boldsymbol{u}_o$, respectively.

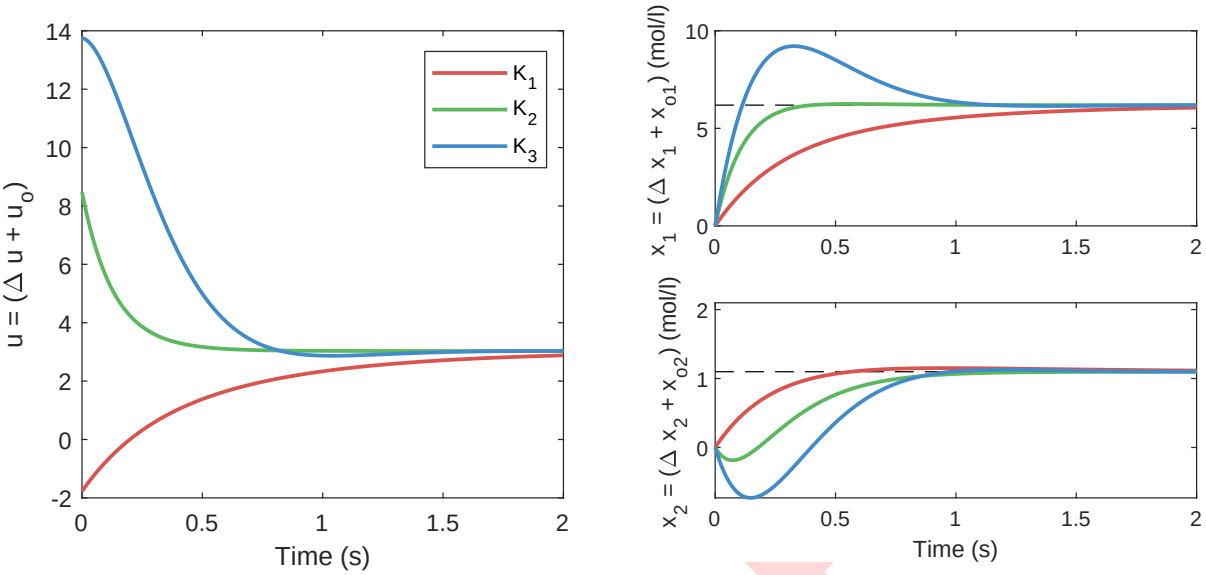


Figure 3.2: Simulation of three closed-loop systems showing the input signal (left) and state responses (right) for gains $K_1 = [-0.99, 1.22]$, $K_2 = [0.78, 0.55]$ and $K_3 = [0.81, 5.20]$. The reference signal is indicated by the black dashed line.

In contrast, the control objective could be to follow a non-constant signal $\boldsymbol{r}(t)$, or to follow a constant signal different from the zero-state, giving rise to the tracking or servomechanism controllers. A more complete discussion is needed in such cases, since there is a possibility that state-feedback is not capable of actually performing this tracking. To understand this question, consider, for simplicity, that the system must follow a constant reference $\boldsymbol{r}(t) = a$. Consider, now, an Input-Output conversion of a closed-loop SISO State-Space model, which from (3.9) directly results in the transfer function:

$$G(s) = \frac{Y(s)}{R(s)} = \frac{\beta_1 s^{n-1} + \beta_2 s^{n-2} + \cdots + \beta_{n-1} s + \beta_n}{s^n + \tilde{\alpha}_1 s^{n-1} + \tilde{\alpha}_2 s^{n-2} + \cdots + \tilde{\alpha}_{n-1} s + \tilde{\alpha}_n}. \tag{3.12}$$

From that formulation it is clear that the response $Y(s) = G(s)R(s)$ will yield a perfect tracking if $G(s) = 1$. Moreover, if the system has to track asymptotically track this reference, this operation can be evaluated as time $t \to \infty$ or, equivalently, as the frequency $s \to 0$. Plugging this limit in the transfer function implies that a perfect tracking is always possible if $G(0) = \beta_n/\tilde{\alpha}_n = 1$, which is not guaranteed a priori. A possible solution is to transform the reference with as $\tilde{r}(t) = Fr(t)$, so that $Y(s) = G(s)\tilde{R}(s) = G(s)FR(s)$, resulting that:

$$G(0)F = 1 \Rightarrow F = \frac{\tilde{\alpha}_n}{\beta_n}, \tag{3.13}$$

which allows for perfect asymptotically tracking in all cases but when $\beta_n = 0$. This same reasoning can easily be extended to MIMO systems (the gain F turns into a matrix). In the case of non-constant references, the same intuition could still be used, but the analysis and design of F becomes more complex [**reference**]. This, however, allows for the definition of tracking controllers.

Definition 3.3. (Tracking Controllers) If a state-feedback controller has to make a system track any step reference $\boldsymbol{r}(t) \neq \boldsymbol{0}$, as $t \to \infty$, it is said to be a *tracking controller*. In this case, one has to apply the *feedforward gain* F to correct the reference as $\tilde{\boldsymbol{r}}(t) = \boldsymbol{F}\boldsymbol{r}(t)$, resulting in the following closed-loop state equation and equivalent solution:

$$\begin{array}{ll} \textbf{State Equation:} & \textbf{Lagrange solution:} \\ \dot{\boldsymbol{x}}(t) = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\,\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{F}\boldsymbol{r}(t) & \boldsymbol{x}(t) = e^{(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K})t}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K})(t-\tau)}\boldsymbol{B}\boldsymbol{F}\boldsymbol{r}(\tau)d\tau \end{array}. \tag{3.14}$$

Despite being a feasible solution, there are still problems with this definition of tracking controllers. For instance, if the system is subject to a *constant additive disturbance*, which as not anticipated in the model, the resulting operation will not yield a perfect tracking.

The problem of the previous formulation for a tracking controller is that it is not robust to actions that happens outside the model. A direct cause of this is the fact that the feedforward gain $\boldsymbol{F}$ does not benefits from the real-time corrective action of the state-feedback, but rather is calculated *off-line* using the model properties. Therefore, a way to ensure a more robust operation could be to insert real-time information about the tracking error directly to the feedback corrective action. With this motivation, a new formulation of the tracking controller is given below.

Definition 3.4. (Robust Tracking Controllers) Given a State-space system and augmented state $\boldsymbol{x}_a(t)$ defined as:

$$\boldsymbol{x}_a(t) = \int_0^t \boldsymbol{r}(\tau) - \boldsymbol{C}\boldsymbol{x}(\tau)d\tau \Longrightarrow \dot{\boldsymbol{x}}_a(t) = \boldsymbol{r}(t) - \boldsymbol{C}\boldsymbol{x}(t). \tag{3.15}$$

A robust tracking (or servo) controller, defined by the gain $\tilde{\boldsymbol{K}} = \begin{bmatrix}\boldsymbol{K} & \boldsymbol{K}_a\end{bmatrix}$, is the one which operates on the following augmented version of the original system:

$$\begin{cases} \begin{bmatrix}\dot{\boldsymbol{x}}(t) \\ \dot{\boldsymbol{x}}_a(t)\end{bmatrix} &= \begin{bmatrix}\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} & -\boldsymbol{B}\boldsymbol{K}_a \\ -\boldsymbol{C} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}(t) \\ \boldsymbol{x}_a(t)\end{bmatrix} + \begin{bmatrix}\boldsymbol{0} \\ \boldsymbol{I}\end{bmatrix}\boldsymbol{r}(t) \\ \boldsymbol{y}(t) &= \begin{bmatrix}\boldsymbol{C} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}(t) \\ \boldsymbol{x}_a(t)\end{bmatrix} \end{cases} \tag{3.16}$$

or, equivalently:

$$\begin{cases} \tilde{\boldsymbol{x}}(t) = \tilde{\boldsymbol{A}}\tilde{\boldsymbol{x}}(t) + \tilde{\boldsymbol{B}}\tilde{\boldsymbol{r}}(t) \\ \boldsymbol{y}(t) = \tilde{\boldsymbol{C}}\tilde{\boldsymbol{x}}(t) \end{cases}. \tag{3.17}$$

Since the augmented state $\boldsymbol{x}_a(t)$ represents an integral of the tracking error until a time t, this formulation is usually characterized as imposing "integral action" to the controller. The schematic at Fig. 3.3 illustrates how an integrator can be included to the block diagram of the control loop. Therefore, it must be discussed if the new gain $\tilde{\boldsymbol{K}}$ still preserves the eigenvalue assignment property of regular state-feedback gains, so that this imposed regulator would work.

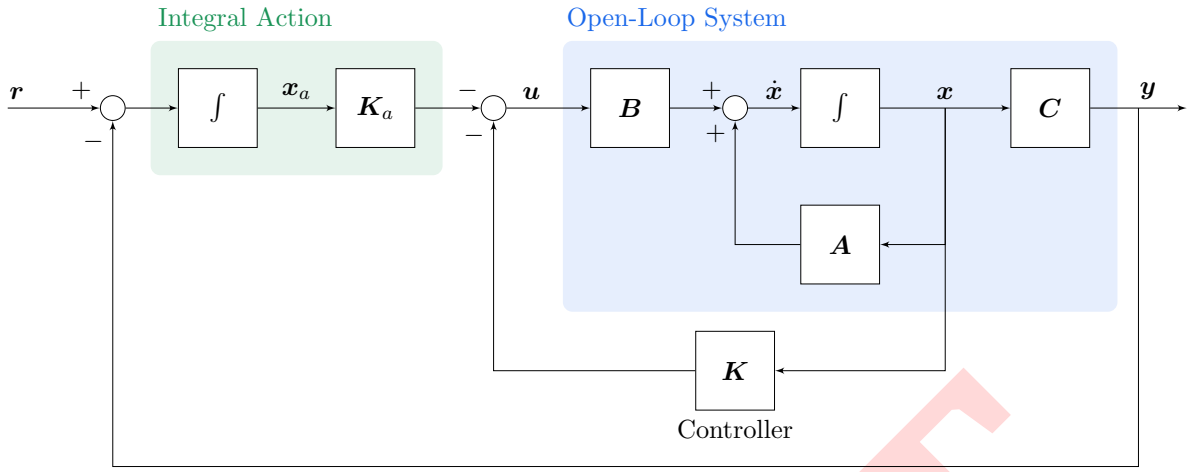


Figure 3.3: Block diagram of a state-feedback closed-loop system with integral action.

Theorem 3.3. *If the SISO system described by $(\boldsymbol{A}, B)$ is controllable and its transfer function $G(s)$ has no zero at $s = 0$, then the eigenvalues of the augmented matrix $\tilde{\boldsymbol{A}}$ can be assigned arbitrary by the feedback gain $\tilde{\boldsymbol{K}}$.*

Proof. Consider a SISO controllable system. After the augmentation, the controllability matrix $\tilde{\boldsymbol{\mathcal{C}}}$ is calculated as:

$$\begin{aligned}\tilde{\boldsymbol{\mathcal{C}}} &= \begin{bmatrix} B & \boldsymbol{A}B & \boldsymbol{A}^2B & \boldsymbol{A}^3B & \cdots & \boldsymbol{A}^{n-1}B \\ 0 & -CB & -C\boldsymbol{A}B & -C\boldsymbol{A}^2B & \cdots & -C\boldsymbol{A}^{n-2}B \end{bmatrix} \\ &= \begin{bmatrix} 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & -\alpha_1(\alpha_1^2-\alpha_2)+\alpha_2\alpha_1-\alpha_3 & \cdots & \Delta_2(\alpha_1,...,\alpha_n) \\ 0 & 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & \cdots & \Delta_3(\alpha_1,...,\alpha_n) \\ 0 & 0 & 1 & -\alpha_1 & \cdots & \Delta_1(\alpha_1,...,\alpha_n) \\ 0 & 0 & 0 & 1 & \cdots & \Delta_4(\alpha_1,...,\alpha_n) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & -\beta_1 & -\beta\alpha_1-\alpha_2 & \cdots & \Delta_n(\alpha_1,...,\alpha_n) \end{bmatrix},\end{aligned} \tag{3.18}$$

where $\Delta_i(\alpha_1, ..., \alpha_n)$ is a polynomial created to save space in this representation. By inspection of this matrix, it is possible to discover a pattern between the rows. Since elementary operations between the rows $r_1, r_2, ..., r_n$ doesn't change the matrix row rank, the last row of the matrix can be transformed as $r_n = r_n + r_{n-1}\beta_{n-2} + r_{n-2}\beta_{n-3} + \cdots + r_2\beta_1$. The result is the triangular matrix in the form:

$$\tilde{\boldsymbol{\mathcal{C}}} = \begin{bmatrix} 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & -\alpha_1(\alpha_1^2-\alpha_2)+\alpha_2\alpha_1-\alpha_3 & \cdots & \Delta_2(\alpha_1,...,\alpha_n) \\ 0 & 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & \cdots & \Delta_3(\alpha_1,...,\alpha_n) \\ 0 & 0 & 1 & -\alpha_1 & \cdots & \Delta_1(\alpha_1,...,\alpha_n) \\ 0 & 0 & 0 & 1 & \cdots & \Delta_4(\alpha_1,...,\alpha_n) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \beta_n \end{bmatrix}. \tag{3.19}$$

Since $\boldsymbol{G}(s)$ has no zeros at $s = 0$, then $\beta_n \neq 0$, meaning that $\tilde{\boldsymbol{\mathcal{C}}}$ is nonsingular and, therefore has full row rank. This concludes that the augmented system $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$ is controllable and, from Theorem 3.2, the eigenvalues of $\tilde{\boldsymbol{A}}$ can be assigned anywhere in the complex plane. □

This result does not scale naturally to MIMO systems, but the main intuition is the same. Basically, if the augmented matrices are controllable, the robust tracking can be achieved by the same formulation. A possible intuition on how this system performs the robust tracking and disturbance rejection can be taken from the fact that the first row of (3.16) is basically a regulator, albeit from the term $\boldsymbol{BK}_a\boldsymbol{x}_a(t)$. Because of this, a control action will always be applied whenever $\boldsymbol{x}_a(t) \neq \mathbf{0}$, i.e., when there is an error between the reference and the output signal, following the direction that minimizes this difference. When $\boldsymbol{x}_a(t) = \mathbf{0}$, this equation reduces to a simple regulator, and the disturbances are expected to be rejected. A more quantitative analysis on why this controller yields both robust tracking and disturbance rejection can be found in [**reference**].

For the sake of illustration, consider the same system used in (3.11). Unfortunately, the augmented version of this *single-input multiple-output* (SIMO) formulation is not controllable. However, the SISO version obtained by letting $C = \begin{bmatrix}0, 1\end{bmatrix}$ obeys Theorem 3.3, so a robust tracking controller can be designed by state-feedback. Some simulations of closed-loop systems for this system to track are shown in Fig. 3.4 for a non-constant reference consisting of a sequence of step signals.

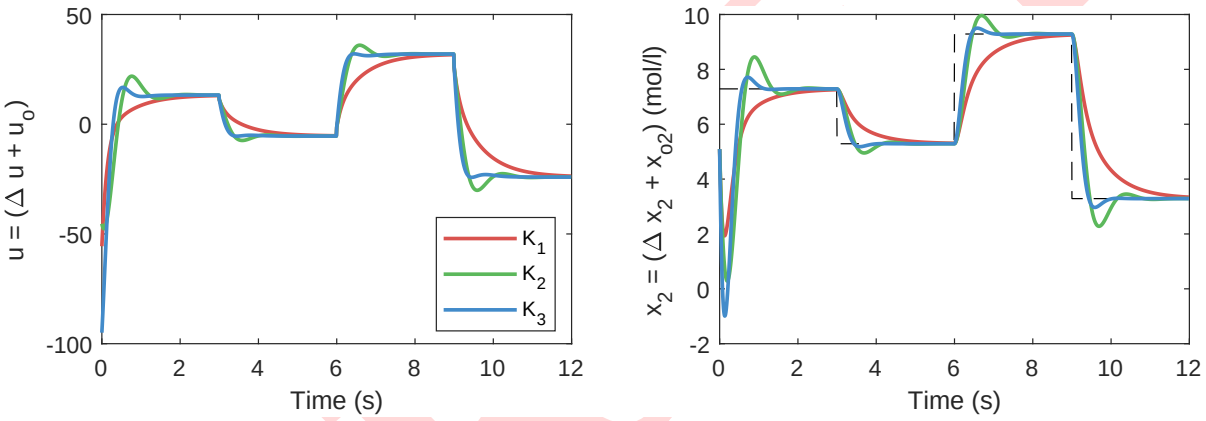


Figure 3.4: Simulation of three closed-loop systems showing the input signal (left) and state response (right) for augmented gains $\tilde{K}_1 = [-0.99, 1.22, -25]$, $\tilde{K}_2 = [0.78, 0.55, -25]$ and $\tilde{K}_3 = [0.81, 5.20, -50]$. The reference signal is indicated by the black dashed line.

3.3 Deterministic State Observers

Until now, the state feedback was discussed in the perspective that the device has direct to the real value of all states of the system. This is assumption is actually unrealistic, since the states are only observed through the output signal $\boldsymbol{y}(t)$, that maps the states through the matrix $\boldsymbol{C}$, which itself is not assumed to be always equal to the identity matrix. In practice, this means that some states could not be measured, due to technical difficulties or economic reasons, or that the instrumentation available is not perfect, and the observations are prone to deviate from the real value. In reactor systems, for instance, is hard to measure the actual value of chemical compounds concentrations in small scales, and the measuring process itself could be very slow or simply unpractical in operational conditions [**reference**]. Since the state-vector is necessary for the state-feedback to compute the input to the system, this section discusses how to develop devices that can reconstruct information about the states from the observations of the available sensor.

A device that generates a state-vector $\boldsymbol{x}(t)$ from the output signal $\boldsymbol{y}(t)$ is known as *state observer*, or *state estimator* in some cases. Amongst the several possible configurations, a very practical and popular one is the *Luenberger observer*, which is defined below.

Definition 3.5. (Luenberger Observer) Given a system in State-Space with output signal $\boldsymbol{y}(t) : \mathbb{R} \to \mathbb{R}^p$ and a observer gain $\boldsymbol{L} \in \mathbb{R}^{n \times p}$, the estimated state-vector $\hat{\boldsymbol{x}}(t)$ is represented by the observer system:

$$\dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{A}\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{L}\left(\boldsymbol{y}(t) - \boldsymbol{C}\hat{\boldsymbol{x}}(t)\right), \tag{3.20}$$

or, equivalently:

$$\dot{\hat{\boldsymbol{x}}}(t) = \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{L}\boldsymbol{y}(t). \tag{3.21}$$

The observer system works as a parallel system that is simulated alongside with the actual system, as illustrate in Fig. 3.5. The expected result is that the observer yields $\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t)$, as time $t \to \infty$. Alternatively, it is possible to create a variable $\boldsymbol{e}(t) = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)$ such that, using (3.21):

$$\begin{aligned}
\dot{\boldsymbol{e}} &= \boldsymbol{x} - \hat{\boldsymbol{x}} \\
&= \left(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}\right) - \left(\left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\hat{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{L}\boldsymbol{C}\boldsymbol{x}\right) \\
&= \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\boldsymbol{x} - \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\hat{\boldsymbol{x}} \\
&= \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\left(\boldsymbol{x} - \hat{\boldsymbol{x}}\right) \\
&= \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\boldsymbol{e}
\end{aligned}, \tag{3.22}$$

which implies that the observer asymptotically tracks the actual state-vector if $\boldsymbol{e}(t) = \boldsymbol{0}$ as $t \to \infty$. Analyzing the equation above, it is intuitive to notice that this result can be guaranteed if all the eigenvalues of matrix $\boldsymbol{A}_{obs} = \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ have negative real parts. The following theorem relates this statement with the choice of a gain $\boldsymbol{L}$.

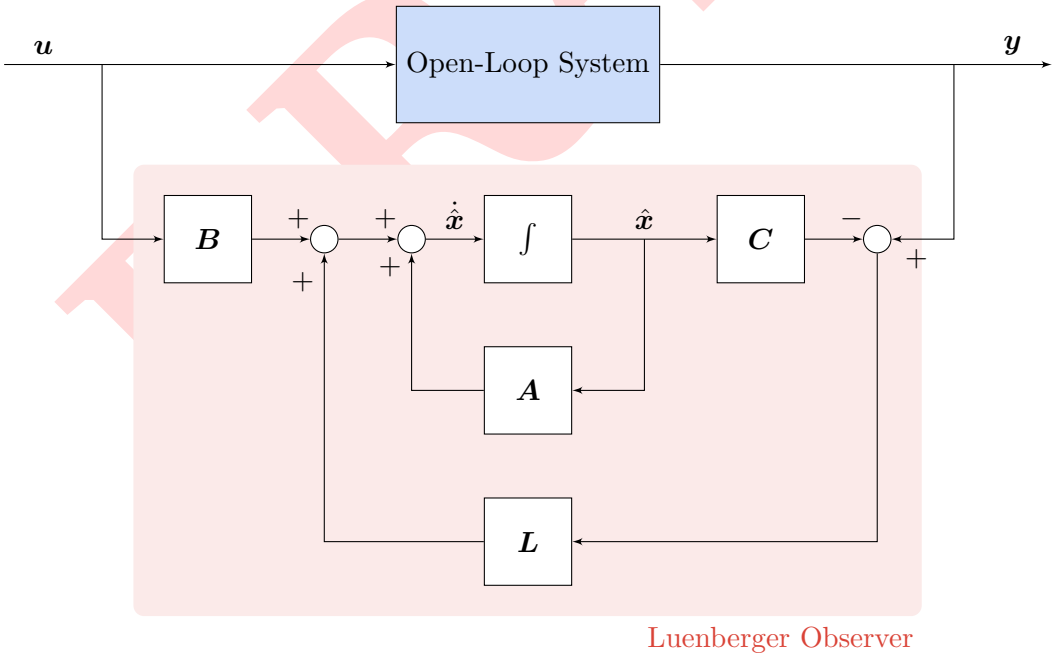


Figure 3.5: Block diagram of a State-Space system with Luenberger observer.

Theorem 3.4. *If a system in State-Space representation is observable, then by a Luenberger observer with gain matrix $\boldsymbol{L} \in \mathbb{R}^{n \times p}$ the eigenvalues of $\boldsymbol{A}_{obs} = \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ can arbitrarily be assigned anywhere in the complex plane, given that complex conjugate eigenvalues are assigned in pairs.*

Proof. Consider that a State-Space with matrices $(\boldsymbol{A}, \boldsymbol{C})$ is observable. From the Duality Theorem [**reference**] if the pair $(\boldsymbol{A}, \boldsymbol{C})$ is observable then the pair $(\boldsymbol{A}^T, \boldsymbol{C}^T)$ is controllable. In this case, it is possible to design a gain matrix $\boldsymbol{K}$ to assign the eigenvalues of $\tilde{\boldsymbol{A}}_{obs} = \boldsymbol{A}^T - \boldsymbol{C}^T\boldsymbol{K}$ in any desirable points in the complex space. Since the eigenvalues of a matrix are invariant to the transpose operation, the design of K can also place the eigenvalues of the matrix $(\tilde{\boldsymbol{A}}_{obs})^T = \boldsymbol{A} - \boldsymbol{K}^T\boldsymbol{C}$. Therefore, making $\boldsymbol{L} = \boldsymbol{K}^T$ establishes the theorem. □

The procedure stated in this proof presents the similarities between closed-loop observers, such as the Luenberger observer, and closed-loop controllers as the state-feedback. Basically, the same design considerations that concerns state-feedback are important in the design of the observer gain $\boldsymbol{L}$. For instance, the eigenvalues of $\tilde{\boldsymbol{A}}_{obs} = \boldsymbol{A}^T - \boldsymbol{C}^T\boldsymbol{K}$ can be assigned such that the time evolution of the error $\boldsymbol{e}(t)$ has a desirable time constant, damping coefficient or natural frequency. In conclusion, the state-vector $\boldsymbol{x}(t)$ can be reconstructed by using a observer gain such that each eigenvalue of $\boldsymbol{A}_{obs}$ is on the left-half side of the complex plane.

One of the main reasons to develop an observer is to allows for state-feedback controllers to access the values of the state-vector, thus being able to calculate an appropriate action to follow the reference signal. If a controller can only access the estimated state-vector $\tilde{\boldsymbol{x}}(t)$, it is possible to define a State-Space formulation for a closed-loop based on feedback from estimated states given data from a, possibly non-linear and time-varying, disturbed system.

Definition 3.6. (Feedback from Estimated States) Given a system in State-Space representation whose state-vector is reconstructed from a Luenberger observer of gain $\boldsymbol{L} \in \mathbb{R}^{n\times p}$ and input signal is calculated through state-feedback with gain $\boldsymbol{K} \in \mathbb{R}^{n\times r}$, its time evolution can be represented through the model:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) - \boldsymbol{B}\boldsymbol{K}\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{r}(t) \\ \dot{\hat{\boldsymbol{x}}}(t) = (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C} - \boldsymbol{B}\boldsymbol{K})\,\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{r}(t) + \boldsymbol{L}\boldsymbol{C}\boldsymbol{x}(t) \end{cases} . \tag{3.23}$$

A schematic for this formulation of feedback from estimated states is shown Fig. 3.6, including the possibility for integral action. In fact, this schematic summarizes several different control architectures that includes both feedback action and state estimation, and it will be a reference whenever this work mentions physical control loops and instrumentation.

Notice, now, that the formulation just defined imposes that the dynamics of the estimated state $\hat{\boldsymbol{x}}(t)$ is dependent both in $\boldsymbol{K}$ and $\boldsymbol{L}$, which are arbitrary matrices chosen by the control designer. This leads to the possible conclusion that the choice of $\boldsymbol{K}$ is now restricted by the effect that it will produce in the choice of $\boldsymbol{L}$, which is not true. The following theorem, known as the Separation Principle [**reference**], states that design of these two gains are independent.

Theorem 3.5. *(Separation Principle) Given a system in State-Space with a Luenberger observer of gain $\boldsymbol{L}$ and state-feedback controller of gain $\boldsymbol{K}$, the closed-loop eigenvalues contributions of $(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})$ are independent from those of $(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})$.*

Proof. Consider a feedback from estimated states as defined in (3.23). That controller-estimator system can be rewritten as a single state equation:

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\hat{\boldsymbol{x}}} \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{A} & -\boldsymbol{B}\boldsymbol{K} \\ \boldsymbol{L}\boldsymbol{C} & \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C} - \boldsymbol{B}\boldsymbol{K} \end{bmatrix}}_{\tilde{\boldsymbol{A}}} \begin{bmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{bmatrix} + \underbrace{\begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{B} \end{bmatrix}}_{\tilde{\boldsymbol{B}}} \boldsymbol{r}. \tag{3.24}$$

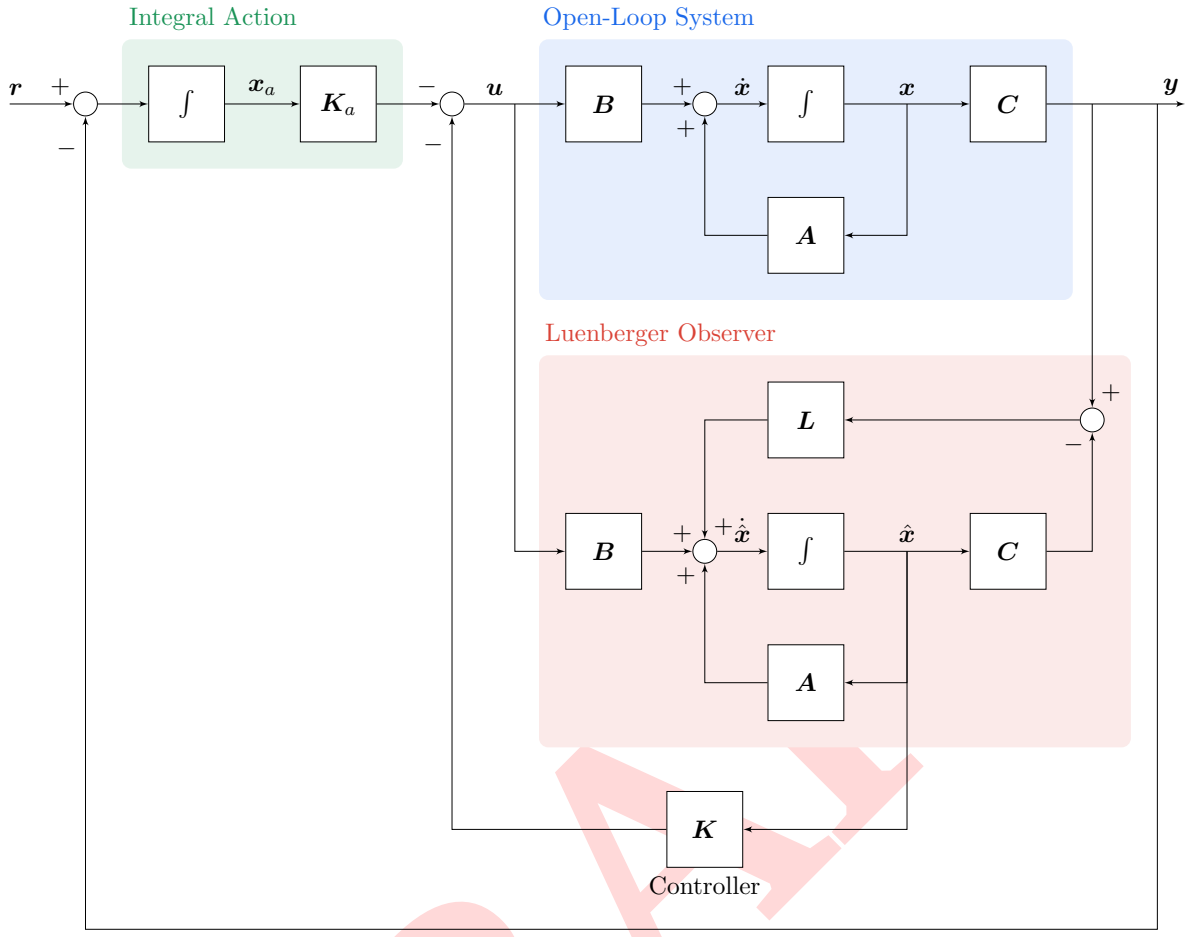


Figure 3.6: Block diagram of a state-feedback closed-loop system with integral action and Luenberger observer.

Consider, now, the following similarity transformation $\boldsymbol{z}(t) = \boldsymbol{P}\boldsymbol{x}(t)$:

$$\underbrace{\begin{bmatrix} \boldsymbol{I} & 0 \\ \boldsymbol{I} & -\boldsymbol{I} \end{bmatrix}}_{\boldsymbol{P}} \begin{bmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{x} - \hat{\boldsymbol{x}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{e} \end{bmatrix}. \tag{3.25}$$

Since $\boldsymbol{P} = \boldsymbol{P}^{-1}$, and this is a valid similarity transformation that does not alters the system eigenvalues, the equivalent system for state $\boldsymbol{z}(t)$ is obtained as:

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{e}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K} & -\boldsymbol{B}\boldsymbol{K} \\ \boldsymbol{0} & \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{e} \end{bmatrix} + \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{r}. \tag{3.26}$$

Since the matrix obtained is block triangular, it is possible to conclude that the system in such configuration has eigenvalues that are contributions from the eigenvalues of $(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})$ and $(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})$ separately. □

The Separation Principle is a nice results that further motivates the topology of Fig. 3.6, since it explicitly states that the design of the controller and the state observer can be done separately. Thus, any structure that obeys the state feedback formulation can be used as a controller and the same is valid for the observer device. In the next chapter, this result will be explored to motivate the use of more advanced control and state estimation techniques without having to redefine the analytical tools and intuitions built for traditional state-feedback controllers from pole-placement methods.

3.4 Properties of State-Feedback Controllers

The last section introduces the first considerations into applying state-feedback in real-world systems, given limitations on the instruments and uncertainty on the environment. Basically, a mathematical analysis of such closed-loop controllers allows for a full characterization of the system behavior, but the real system will exhibit a different response due to these limitations and, essentially, due to possible external disturbances. Therefore, it is desirable to anticipate these variations and discuss the properties of the closed-loop system in the sense of robustness and stability margins. Since this goal requires that the system response is evaluated in as general as possible context, the discussion in this section relies on frequency response analysis, which consider a broader class of input signals: sinusoids of any frequency. Consider the representation of closed-loop systems shown in Fig. 3.7, which consists of a simplified version of Fig. 3.1 in the Laplace frequency domain considering only the state response.

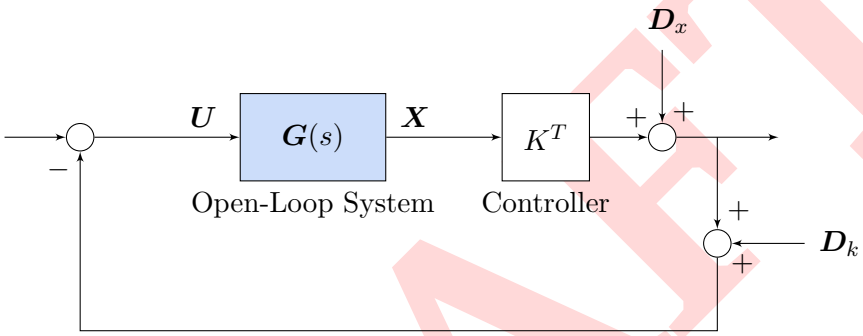


Figure 3.7: Simplified block diagram of a perturbed state-feedback closed-loop.

Using block diagram algebra and Theorem 2.4, it is possible to associate a forward-path transfer function $\boldsymbol{G}_f(s)$ given as [**reference**]:

$$\boldsymbol{G}_f(s) = -\boldsymbol{K}^T G(s) = -\boldsymbol{K}^T \left(s\boldsymbol{I} - \boldsymbol{A}\right)^{-1} B. \tag{3.27}$$

Now, it is possible to characterize the properties of this quantity which relates the state-feedback gain $\boldsymbol{K}$ with the system disturbance $\boldsymbol{D}_x(s)$ and gain disturbance $\boldsymbol{D}_k(s)$. To facilitate the definitions and discussion, the results are shown for single-state single-input systems, so $\boldsymbol{G}_f(s)$ is a scalar function, where the extension for $n > 1$ states is intuitive in most cases. The first property to be discussed, then, considers the stability of closed-loop feedback controllers. Consider the following closed-loop stability criterion from a Bode plot visualization.

Theorem 3.6. *(Bode stability criterion) Consider a feedback system whose closed-loop transfer function is defined, assuming perfect measuring sensors, as:*

$$T(s) = \frac{KG(s)}{1 + KG(s)}. \tag{3.28}$$

The closed-loop system is said to be stable if $|G(j\omega_{pc})| \leq 0$, where ω_{pc} is the phase crossover frequency obtained such that $\angle G(j\omega_{pc}) = -180^o$.

Additionally, it is possible to define a stability criterion through a Nyquist diagram of the closed-loop system.

Theorem 3.7. *(Nyquist criterion) Consider a system with feedforward transfer function as defined in (3.27). Now, let P and Z be respectively the number of poles of $G_f(s)$ and zeros of $1 + G_f(s)$ that are in the right-half plane. In this case, the Nyquist contour shall clockwise encircle the point $s = -1$ a number of times N such that $N = Z - P$.*

A detailed proof of both criterion can be found in [**reference**]. The introduction of these stability evaluation techniques may seem redundant, given that the closed-loop BIBO stability can still be characterized from Theorem 2.13. However, their graphical nature allows for an easy understanding of how disturbances can affect the stability of state-feedback systems. For instance, a system subject to a sinusoidal disturbance of constant magnitude, but with the same frequency as the natural frequency of the system, will show a response with higher magnitude for the phase crossover frequency than the one visualized in the Bode plot. This phenomenon is widely known in Physics as "resonance". Therefore, the influence of disturbances can inflict instability to a stable system.

Of course, not all disturbances observed in real operations are strong enough to bring any reasonable stable controller to an unstable condition. However, depending on the choice of the gain $\boldsymbol{K}$, some closed-loop systems can be more prone to these undesired problems than others. This motivates the discussion on stability margins.

Definition 3.7. (Stability Margins) Given a closed-loop system with transfer function $T(s)$, the *Gain Margin* (GM) is defined as a factor of how much a gain can be increased before the system becomes unstable, and is equated as:

$$GM = \frac{1}{|T(j\omega_{pc})|}, \tag{3.29}$$

where ω_{pc} is the phase crossover frequency such that $\angle T(j\omega_{pc}) = -180^{\text{o}}$ (or the point where a Nyquist diagram crosses the real axis for $-1 < s < 0$). In addition, the *Phase Margin* (PM) is defined as how much phase lag can be added to $T(s)$ the system becomes unstable, and is equated as:

$$PM = \angle T(j\omega_{gc}), \tag{3.30}$$

where ω_{gc} is the *gain crossover frequency* such that $|T(j\omega_{gc})| = 0dB$ (or the angle when a Nyquist diagram crosses the unit circle centered at $s = 0$).

A graphical representation of these margins, for both Bode plots and Nyquist diagrams is shown in Fig. 3.8, for the same system as the one from last figure. If a closed-loop system has small Gain Margin, it is clear that its stability is not robust to gain uncertainties, while a small Phase Margins implies that its stability is not robust to time delay uncertainties in the control actions. Despite the fact that high gain controllers are usually beneficial for performance requirements, it is clear that they also can lead to disastrous operations in uncertain environments [**reference**]. Therefore, the design of state-feedback gains must account for these quantities, and a trade-off between performance and robustness is always necessary for this formulation.

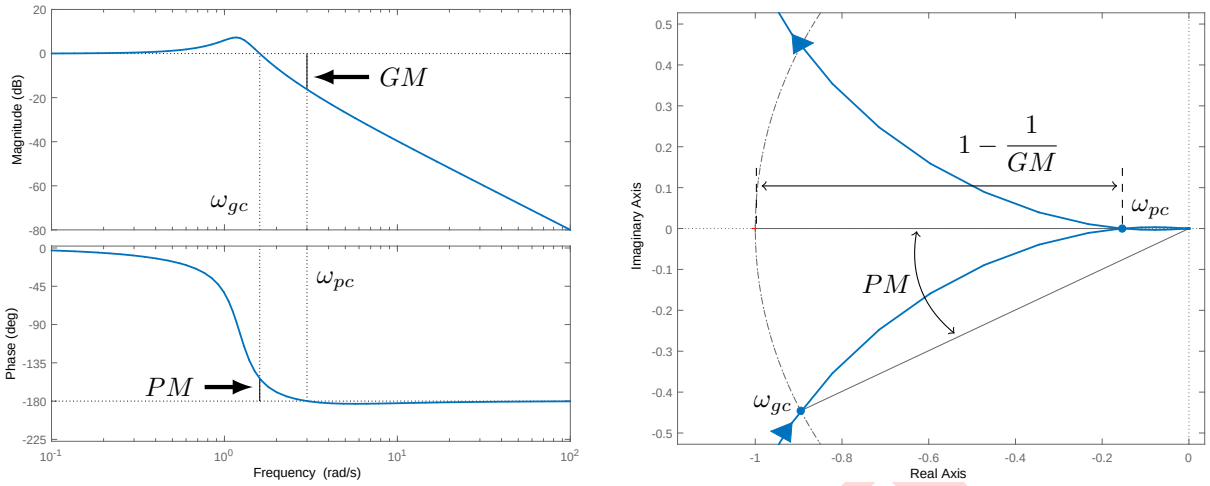


Figure 3.8: Stability margins visualizations given Bode plots (left) and Nyquist diagrams (right) of closed-loop dynamical systems.

Chapter 4

Optimal Control and Estimation

This chapter introduces the vast field of optimal control and optimal estimation of dynamical systems. The developments are focused in optimize a cost function that produces an optimal state trajectory, and the control actions that causes it, given some conditions. The chapter starts by discussing a general formulation of the optimal control problem and then specializing this formulation to a case of a linear system with quadratic cost functions. After that, the dual optimal state estimation problem is discussed and a result that merges the optimal control with the optimal estimator is presented. Finally, the main stability and robustness properties of the controllers derived in this chapter are accessed.

4.1 General Formulation

The last chapter introduced the notion of controller synthesis as an engineering procedure to be done "by hand" from a designer with some knowledge about the system and the control environment. This approach, despite being very popular and practical, introduces some design issues that make it difficult to generalize a controller to different systems within a same class or, more clearly, to MIMO configurations. Thus, a more general and automatic procedure to control synthesis is desirable.

With this motivation, the concept of *Optimal Control* [**references**] was introduced as an alternative strategy for controlling dynamical systems that determines the necessary action by optimizing a cost function (or maximizing a reward function). Usually, these formulations are data-driven methods that autonomously produces optimal input signals given a desired objective and restrictions, and can be easily generalized to different systems.

Definition 4.1. (Optimal Control) Given a system in State-Space formulation, with state-vector signal $\boldsymbol{x} : \mathbb{R} \to \mathbb{R}^n$, and a reference signal $\boldsymbol{r} : \mathbb{R} \to \mathbb{R}^n$, the input signal $u(t) \in \mathbb{R}^r$, for any time t, is calculated by an optimal control through an optimal control law $\pi^* : \mathbb{R}^{n \times n \times 1} \to \mathbb{R}^r$ as:

$$\boldsymbol{u}(t) = \pi^*(\boldsymbol{x}, \boldsymbol{r}, t) = \min_{\boldsymbol{u}} J(\boldsymbol{x}, \boldsymbol{r}, t), \tag{4.1}$$

where $J : \mathbb{R}^{n \times n \times 1} \to \mathbb{R}$ is known as a *cost function* of the states and reference signals.

First of all, note that this optimization can be converted to maximizing a function $\boldsymbol{V}$ by making $\boldsymbol{V} = -\boldsymbol{J}$, so this document will only refers to optimization as minimizing some cost function. Basically, the problem of finding an optimal control law, or optimal control policy, $pi^*(\cdot)$ cares both for the choice of the cost function J and for what optimization technique will be used to determine the value of $u(t)$ that achieve this minimum. This problem differs from

standard optimization problems in data science because not only the data is usually obtained online from the system, but it is dependent through time and constrained by the dynamics of the model and optimal policy action. Thus, the solutions for this optimization are usually obtained through Calculus of Variations [**reference**] or, as is the approach used in this work, through Dynamic Programming [**reference**].

To facilitate the discussion and analysis of optimal controllers, consider a subclass of these controllers (that is still very general) defined below by a specific choice of functional for the cost function.

Definition 4.2. (Finite-Horizon Optimal Regulators) Consider a controller setup as in Definition 4.1. A *Finite-Horizon Optimal Regulators* is defined as any controller whose optimal policy over a time interval $t \in [t_0, T]$ minimizes the cost functional:

$$J = \int_{t_0}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}, T), \tag{4.2}$$

where $l(\cdot) : \mathbb{R}^{n \times r \times 1} \to \mathbb{R}$ and $l_f(\cdot) : \mathbb{R}^n \to \mathbb{R}$ are, respectively, the trajectory and terminal *loss functions*. In the case that $t_0 = 0$, T is also known as the *control horizon*.

This formulation presents a notion of, basically, optimize for an trajectory $P^* = [\boldsymbol{x}(t_0), \cdots, \boldsymbol{x}(T)]$ which is optimal given the loss functions, and then find the control input that can achieve this trajectory. As will be shown later, this is a very feasible strategy for controllable linear systems. In a optimization notation, this problem can also be presented as the following *program*:

$$\begin{array}{rl} \textbf{minimize} & \int_{t_0}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}, T) \\ \textbf{s.t.} & \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \\ & \boldsymbol{u}(t) = \pi(\boldsymbol{x}(t_0), \cdots, \boldsymbol{x}(t)) \end{array} . \tag{4.3}$$

Now, the discussion turns to how to solve this general problem. In this work, the solution for the optimal control will follow the same dynamic programming formulation as in [**reference**]. The first necessary effort, then, is to define an important partial differential equation known as the Hamilton-Jacobi equation.

Theorem 4.1. *(Hamilton-Jacobi equation) Consider a finite-horizon cost function in the form of Definition 4.2, for a system described by the state-equation $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)$. Consider also that the loss $l(\cdot)$ and state function $\boldsymbol{f}$ are smooth on their parameters. Then, minimizing any functional in the form of J is equivalent to minimize the solution of the Hamilton-Jacobi equation, which is given by the partial differential equation:*

$$\frac{\partial V^*}{\partial t} = -\min_{u(t)} \left[l(\boldsymbol{x}, \boldsymbol{u}, t) + \left[\frac{\partial V^*}{\partial x} \right]^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \right] \tag{4.4}$$

and the boundary condition:

$$V^*(\boldsymbol{x}, T) = l_f(\boldsymbol{x}(T)). \tag{4.5}$$

Proof. First of all, consider the restatement of the cost functional as the function $V(\cdot)$:

$$V(\boldsymbol{x}, \boldsymbol{u}, t_0) = \int_{t_0}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}(T)). \tag{4.6}$$

Minimizing this cost functional over control inputs $\boldsymbol{u}(t_0), \cdots, \boldsymbol{u}(T)$ consists in evaluating the optimal cost:

$$V^*(\boldsymbol{x}, t_0) = \min_{\boldsymbol{u}(t_0),\cdots,\boldsymbol{u}(T)} \left[\int_{t_0}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}(T)) \right]. \tag{4.7}$$

Now, consider any $t \in [t_0, T]$ and $t_r \in [t, T]$. Since the original control action $\boldsymbol{u}(t), \cdots, \boldsymbol{u}(T)$ can be obtained through the concatenation of $\boldsymbol{u}(t), \cdots, \boldsymbol{u}(t_r)$ and $\boldsymbol{u}(t_r), \cdots, \boldsymbol{u}(T)$, the optimal cost in this interval can be represented in the recursive form:

$$\begin{aligned} V^*(\boldsymbol{x}, t) &= \min_{\boldsymbol{u}(t),\cdots,\boldsymbol{u}(T)} \left[\int_{t}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}(T)) \right] \\ &= \min_{\boldsymbol{u}(t),\cdots,\boldsymbol{u}(t_r)} \left\{ \min_{\boldsymbol{u}(t_r),\cdots,\boldsymbol{u}(T)} \left[\int_{t}^{t_r} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + \int_{t_r}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}(T)) \right] \right\} \\ &= \min_{\boldsymbol{u}(t),\cdots,\boldsymbol{u}(t_r)} \left\{ \int_{t}^{t_r} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + \min_{\boldsymbol{u}(t_r),\cdots,\boldsymbol{u}(T)} \left[\int_{t_r}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}(T)) \right] \right\} \\ &= \min_{\boldsymbol{u}(t),\cdots,\boldsymbol{u}(t_r)} \left\{ \int_{t}^{t_r} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + V^*(\boldsymbol{x}, t_r) \right\} \end{aligned}. \tag{4.8}$$

Without loss of generalization, let $t_r = t + \delta t$, where δt is a small number. Since $l(\cdot)$ is a smooth function, the right-hand side of the recursive form above can be expanded by a Taylor series expansion:

$$\begin{aligned} V^*(\boldsymbol{x}, t) = \min_{\boldsymbol{u}(t),\cdots,\boldsymbol{u}(t+\delta t)} \Bigg\{ & \delta t l(\boldsymbol{x}, \boldsymbol{u}, t + \delta t) \\ & + V^*(\boldsymbol{x}, t) + \left[\frac{\partial V^*(\boldsymbol{x}, t)}{\partial \boldsymbol{x}} \right]^T \frac{d\boldsymbol{x}(t)}{dt} \delta t + \frac{\partial V^*(\boldsymbol{x}, t)}{\partial t} \delta t + O(\delta t)^2 \Bigg\} \end{aligned}, \tag{4.9}$$

where $O(\delta t)^2$ denotes the high order terms. Since the terms $V^*(\boldsymbol{x}, t)$ and $(\partial V^*/\partial t)\delta t$ does not depend on $\boldsymbol{u}(t)$, they can be taken out of the minimization. Rearranging the terms and substituting $d\boldsymbol{x}/dt = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)$ results in:

$$\frac{\partial V^*}{\partial t}(\boldsymbol{x}, t) = - \min_{\boldsymbol{u}(t),\cdots,\boldsymbol{u}(t+\delta t)} \left\{ l(\boldsymbol{x}, \boldsymbol{u}, t + \delta t) + \left[\frac{\partial V^*(\boldsymbol{x}, t)}{\partial \boldsymbol{x}} \right]^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) + O(\delta t)^2 \right\}. \tag{4.10}$$

Finally, making $\delta t \to 0$ results in:

$$\frac{\partial V^*}{\partial t} = - \min_{\boldsymbol{u}(t)} \left\{ l(\boldsymbol{x}, \boldsymbol{u}, t) + \left[\frac{\partial V^*}{\partial \boldsymbol{x}} \right]^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \right\}. \tag{4.11}$$

To establish the theorem it remains to derive the boundary condition. This result, however, is direct from the form of the cost function, since $V^*(\boldsymbol{x}, T) = l_f(\boldsymbol{x}(T))$ can not be changed through any more control action inside the time horizon. □

The Hamilton-Jacobi equation implies that finite-horizon optimal controllers can be optimized in a recursive manner, starting from the boundary condition in $t = T$ to the beginning of the horizon at $t = t_0$. This is a result from the fact that an optimal action $u(t)$ depends on the loss function $l(\cdot)$ at time t and on the optimal cost $V^*(\cdot)$ for a time immediately after that (for instance, $t+\delta t$), but the last action $u(T)$ depends only on $l_f(\cdot)$ since it no longer affects the system inside the control horizon. The recursive property, directly evidenced in (4.8), is a statement of the Bellman's Principle of Optimality [**reference**] and, for this reason, the Hamilton-Jacobi equation in the context of optimal control theory is known as the Hamilton-Jacobi-Bellman equation. A illustration of this principle is shown in Fig. 4.1

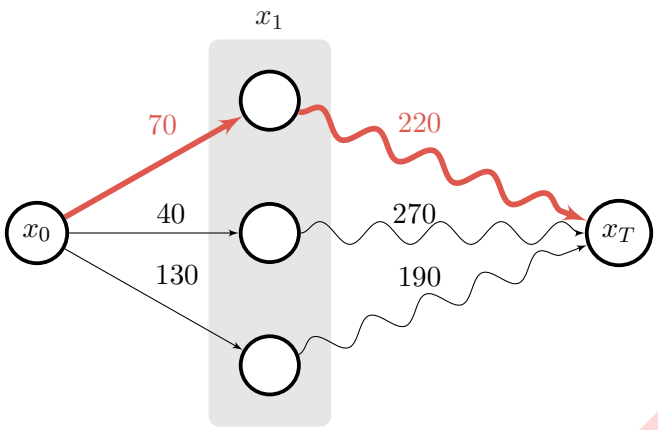


Figure 4.1: Illustration of the Bellman's Principle of Optimality. Each column represents a discrete time instance and each node represents a possible discrete state. The straight lines represents state transitions given an action with associated costs, whereas the curves represents the trajectory from that state to the terminal state with associated optimal cost. The optimal trajectory between the initial and terminal state is shown in red.

4.2 Linear Quadratic Regulator (LQR)

The last section introduced a general condition for solving a finite-horizon optimal control problem. Developing an analytical solution of that condition for any arbitrary loss function $l(\cdot)$ and state-equation $\boldsymbol{f}(\cdot)$ is usually intractable. However, there is a choice of loss function that, under a linear system, allows for a nice closed-form solution to the Hamilton-Jacobi-Bellman equation. This defines the popular class of optimal controllers known as the Linear Quadratic Regulators.

Definition 4.3. (Linear Quadratic Regulator) Given a linear State-Space system in the form:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases} . \tag{4.12}$$

A *Linear Quadratic Regulator* (LQR) for this system is an optimal controller defined by the quadratic cost function:

$$J(\boldsymbol{x}, \boldsymbol{u}, t_0) = \int_{t_0}^{T} \left(\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} + \boldsymbol{u}^T \boldsymbol{R} \boldsymbol{u} \right) dt + \boldsymbol{x}^T(T) \boldsymbol{Q}_f \boldsymbol{x}(T), \tag{4.13}$$

where is assumed that $\boldsymbol{Q}, \boldsymbol{Q}_f \succ 0$ and $\boldsymbol{R} \succ 0$ are matrices penalizing, respectively, the state-vector magnitude and the control effort.

The LQR optimal controller was first introduced in [**reference**] and has been a foundation of optimal control theory ever since. In a engineering point of view, this choice of cost function has the advantage that it breaks the controller design to simply choose matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ as a trade-off between performance and actuator restrictions. In a mathematical point of view, the cost function has the advantage of being a quadratic function of the state and input signals, which is nice since quadratic optimization problems are widely studied in the literature [**reference**]. Now, it is necessary to develop a solution for the Hamilton-Jacobi equation in the light of this formulation. First of all, consider the following theorem.

Theorem 4.2. *Consider a continuous cost function $V : \mathbb{R}^{n\times r\times 1} \to \mathbb{R}$ given as the LQR cost function defined in Definition 4.3 for a linear system. Then the optimal cost $V^*(\boldsymbol{x}, t)$ has the quadratic form:*

$$V^*(\boldsymbol{x}, t) = \boldsymbol{x}^T(t)\boldsymbol{P}(t)\boldsymbol{x}(t) \tag{4.14}$$

for any (possibly symmetric) matrix $\boldsymbol{P}(t)$ of appropriate dimensions. More precisely, the optimal cost $V^*(\boldsymbol{x}, t)$ satisfies the necessary and sufficient conditions for a quadratic function given as, for any $\lambda \in \mathbb{R}$:

$$V^*(\lambda\boldsymbol{x}, t) = \lambda^2 V^*(\boldsymbol{x}, t) \tag{4.15}$$

$$V^*(\boldsymbol{x}_1, t) + V^*(\boldsymbol{x}_2, t) = \frac{1}{2}\left(V^*(\boldsymbol{x}_1 + \boldsymbol{x}_2, t) + V^*(\boldsymbol{x}_1 - \boldsymbol{x}_2, t)\right). \tag{4.16}$$

Proof. Since $V^*(\boldsymbol{x}, t)$ is the minimum value of $V(\boldsymbol{x}, \boldsymbol{u}^*, t)$ given an optimal input $\boldsymbol{u}^*(t)$ for $t \in [t, T]$, then any deviance $\lambda \in \mathbb{R}$ in this parameter will produce a greater value of the cost. Thus, a direct result from this and the fact that $V(\boldsymbol{x}, \boldsymbol{u}, t)$ is a quadratic function of $\boldsymbol{x}(t)$ and $\boldsymbol{u}(t)$ is:

$$V^*(\boldsymbol{x}, t) \leq V(\lambda\boldsymbol{x}, \lambda\boldsymbol{u}^*, t) = \lambda^2 V^*(\boldsymbol{x}, t) \leq \lambda^2 V(\boldsymbol{x}, \lambda^{-1}\boldsymbol{u}^*, t) = V^*(\boldsymbol{x}, t), \tag{4.17}$$

or, simply:

$$V^*(\boldsymbol{x}, t) \leq \lambda^2 V^*(\boldsymbol{x}, t) \leq V^*(\boldsymbol{x}, t), \tag{4.18}$$

which implies $V^*(\boldsymbol{x}, t) = \lambda^2 V^*(\boldsymbol{x}, t)$ and establishes (4.15). Similarly:

$$\begin{aligned} V^*(\boldsymbol{x}_1, t) + V^*(\boldsymbol{x}_2, t) &= \frac{1}{4}\left(V^*(2\boldsymbol{x}_1, t) + V^*(2\boldsymbol{x}_2, t)\right) \\ &\leq \frac{1}{4}\left(V(2\boldsymbol{x}_1, \boldsymbol{u}^*_{x_1+x_2} + \boldsymbol{u}^*_{x_1-x_2}, t) + V(2\boldsymbol{x}_2, \boldsymbol{u}^*_{x_1+x_2} - \boldsymbol{u}^*_{x_1-x_2}, t)\right) \\ &= \frac{1}{2}\left(V(\boldsymbol{x}_1 + \boldsymbol{x}_2, \boldsymbol{u}^*_{x_1+x_2}, t) + V(\boldsymbol{x}_1 - \boldsymbol{x}_2, \boldsymbol{u}^*_{x_1-x_2}, t)\right) \\ &= \frac{1}{2}\left(V^*(\boldsymbol{x}_1 + \boldsymbol{x}_2, t) + V^*(\boldsymbol{x}_1 - \boldsymbol{x}_2, t)\right) \\ &\leq V^*(\boldsymbol{x}_1, t) + V^*(\boldsymbol{x}_2, t) \end{aligned}, \tag{4.19}$$

which implies $V^*(\boldsymbol{x}_1, t) + V^*(\boldsymbol{x}_2, t) = (1/2)\left(V^*(\boldsymbol{x}_1 + \boldsymbol{x}_2, t) + V^*(\boldsymbol{x}_1 - \boldsymbol{x}_2, t)\right)$ and establishes (4.16). Therefore, the optimal cost function has a quadratic form $V^*(\boldsymbol{x}, t) = \boldsymbol{x}^T(t)\boldsymbol{P}(t)\boldsymbol{x}(t)$. □

Notice that only the fact that the LQR cost function is quadratic directly implies that the optimal cost is also quadratic. Therefore, it is possible to define the optimal action $\boldsymbol{u}^*(t)$ that produced this optimal cost by evaluating a relationship between it and the matrix $\boldsymbol{P}$. Since the quadratic cost just evaluated is defined for a finite-horizon optimal controller, it has to obey the condition imposed by the Hamilton-Jacobi equation, and the optimal controller can be solved as shown in the following theorem.

Theorem 4.3. *(LQR Control Action) Given a Linear Quadratic Regulator as defined in Definition 4.3, the optimal action produced by this optimal controller at any time $t \in [t_0, T]$ is given by:*

$$\boldsymbol{u}^*(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t)\boldsymbol{x}(t), \tag{4.20}$$

where $\boldsymbol{P}(t)$ is the solution of the matrix Riccati differential equation:

$$-\dot{\boldsymbol{P}}(t) = \boldsymbol{A}^T\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{A} - \boldsymbol{P}(t)\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t) + \boldsymbol{Q}, \tag{4.21}$$

with terminal condition $\boldsymbol{P}(T) = \boldsymbol{Q}_f$.

Proof. Consider the Hamilton-Jacobi equation from (4.4) restated below for a quadratic loss function $l(\boldsymbol{x}, \boldsymbol{u}, t$ and a linear system with state equation $\boldsymbol{f}(t)$ as given by Definition 4.3:

$$\begin{aligned} \frac{\partial(\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x})}{\partial t} &= -\min_{\boldsymbol{u}(t)} \left\{ \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u} + \left[\frac{\partial(\boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x})}{\partial \boldsymbol{x}}\right]^T (\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}) \right\}, \\ \boldsymbol{x}^T\dot{\boldsymbol{P}}\boldsymbol{x} &= -\min_{\boldsymbol{u}(t)} \left\{ \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u} + (2\boldsymbol{x}^T\boldsymbol{P})(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}) \right\} \end{aligned} \tag{4.22}$$

where, without loss of generalization, $\boldsymbol{P}(t)$ was assumed to be symmetric. Because of the quadratic nature, it is possible to calculate the minimum of the right-hand side of the equation by taking the derivative with respect to the control action and evaluate it for zero:

$$\begin{aligned} 0 &= \frac{\partial}{\partial \boldsymbol{u}(t)} \left(\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u} + (2\boldsymbol{x}^T\boldsymbol{P})(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}) \right) \\ 0 &= 2\boldsymbol{u}^T(t)\boldsymbol{R} + (2\boldsymbol{x}^T(t)\boldsymbol{P}(t))\boldsymbol{B} \\ \boldsymbol{u}(t) &= -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t)\boldsymbol{x}(t) \end{aligned} . \tag{4.23}$$

To solve for $\boldsymbol{P}$ first note that the following identity can be found by completing the squares:

$$\begin{aligned} \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u} + (2\boldsymbol{x}^T\boldsymbol{P})(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}) =& (u + \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}x)^T\boldsymbol{R}(u + \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}x) \\ &+ \boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q})\boldsymbol{x} \end{aligned} . \tag{4.24}$$

Thus, plugging this identity and substituting the optimal control action results in the matrix Riccati equation:

$$\begin{aligned} \boldsymbol{x}^T\dot{\boldsymbol{P}}\boldsymbol{x} &= \boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q})\boldsymbol{x} \\ \dot{\boldsymbol{P}} &= \boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} \end{aligned} . \tag{4.25}$$

Finally, the terminal condition is direct from the fact that $V^*(\boldsymbol{x}, T) = \boldsymbol{x}^T(T)\boldsymbol{Q}_f\boldsymbol{x}(T)$. □

A closed-form solution for the LQR problem makes this controller a very appealing solution in several applications, since not only it is an optimal controller but the controller action can be calculated in real-time. Several others optimal controllers may pose performance improvements, but usually depends on iterative optimization procedures, that can be hard to compute in high-dimensional systems (or even in critical low-dimensional systems). Furthermore, the time complexity of the LQR control action computation is governed by the solution of the Riccati differential equation which can be done very efficiently [**reference**].

Now, consider the optimal control action $\boldsymbol{u}^*(t)$. Applying this action to a linear system in State-Space representation results in the following state-equation:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\left(-\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t)\boldsymbol{x}(t)\right) \\ &= \left(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t)\right)\boldsymbol{x}(t) \\ &= \left(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}(t)\right)\boldsymbol{x}(t) \end{aligned} . \tag{4.26}$$

This formulation indicates that the Linear Quadratic Regulator solution follows the exact same form of a standard state-feedback regulator, represented by the linear but time-variant state-feedback gain $\boldsymbol{K}(t) = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t)$. Therefore, the past results of state-feedback regulators are directly applied to the operation of this class of optimal controllers. Notice, also, that both the computation of $\boldsymbol{K}(t)$ and $\boldsymbol{P}(t)$ does not explicitly depends on $\boldsymbol{x}(t)$, meaning that, for a specific control horizon, they can be calculated off-line and then provided to the controller actuator for the on-line operation. For this reason, despite being a closed-loop controller with corrective action, the LQR can be considered a open-loop optimizer, since the optimization solution itself does not depend on the state signal.

Now, as discussed in the previous chapter, the class of regulation controllers are broad but still restricted to a zero-state reference signal. To expand the range of application for the optimal controller just derived, it is possible to introduce integral action to the feedback, just as in the previous case.

Definition 4.4. (Linear Quadratic Servo) Given a linear State-Space system represented by matrices $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$, augmented with state $\dot{\boldsymbol{x}}_a(t) = \boldsymbol{r}(t) - \boldsymbol{C}\boldsymbol{x}(t)$:

$$\begin{cases} \begin{bmatrix} \dot{\boldsymbol{x}}(t) \\ \dot{\boldsymbol{x}}_a(t) \end{bmatrix} &= \underbrace{\begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ -\boldsymbol{C} & \boldsymbol{0} \end{bmatrix}}_{\tilde{\boldsymbol{A}}} \underbrace{\begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{x}_a(t) \end{bmatrix}}_{\tilde{\boldsymbol{x}}(t)} + \underbrace{\begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix}}_{\tilde{\boldsymbol{B}}} \boldsymbol{u}(t) + \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{I} \end{bmatrix} \boldsymbol{r}(t) \\ \boldsymbol{y}(t) &= \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{x}_a(t) \end{bmatrix} \end{cases}. \tag{4.27}$$

A *Linear Quadratic Servo* (LQ-Servo) for this system is an optimal controller defined by the quadratic cost function:

$$J(\boldsymbol{x}, \boldsymbol{u}, t_0) = \int_{t_0}^{T} \left(\tilde{\boldsymbol{x}}^T \tilde{\boldsymbol{Q}} \tilde{\boldsymbol{x}} + \boldsymbol{u}^T \boldsymbol{R} \boldsymbol{u} \right) dt + \tilde{\boldsymbol{x}} \tilde{\boldsymbol{Q}}_f \tilde{\boldsymbol{x}}(T), \tag{4.28}$$

where is assumed that $\tilde{\boldsymbol{Q}}, \tilde{\boldsymbol{Q}}_f \succ 0$ and $\boldsymbol{R} \succ 0$ are matrices penalizing, respectively, the state-vector magnitude and the control effort.

As before, the integral action basically turns the tracking problem into a regulation problem. In this case, the optimal controller will try to optimize a zero-state for $\boldsymbol{x}_a(t)$, producing the reference tracking. The new p diagonal entries of the augmented matrix $\tilde{\boldsymbol{Q}}$ can be interpreted as weights penalizing the state deviance from the reference signal. The solution for this controller is the same as the one derived in Theorem 4.3, but making $\boldsymbol{A} = \tilde{\boldsymbol{A}}$ and $\boldsymbol{B} = \tilde{\boldsymbol{B}}$, and the resulting control action can be equated as:

$$\boldsymbol{u}(t) = \tilde{\boldsymbol{K}}(t)\tilde{\boldsymbol{x}}(t) = \begin{bmatrix} \boldsymbol{K}(t) & \boldsymbol{K}_a(t) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{x}_a(t) \end{bmatrix}. \tag{4.29}$$

To finish the section, it is also worth mentioning the fact that, for the LQR cost function, it is possible to determine an optimal controller for an infinite-horizon operation, that is, when $T \to \infty$. The fact that the control horizon is now infinite results in a stationary state-feedback gain $\boldsymbol{K}$, as stated below.

Theorem 4.4. *(Infinite-Horizon LQR) Consider a Linear Quadratic Regulator as defined in Definition 4.3, but with $T = \infty$. The optimal control action produced by this optimal controller at any time $t \in [t_0, \infty]$ is given by:*

$$\boldsymbol{u}^*(t) = -\boldsymbol{K}\boldsymbol{x}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}\boldsymbol{x}(t), \tag{4.30}$$

where $\boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}$ and $\boldsymbol{P}(t)$ is the solution of the matrix algebraic Riccati equation:

$$0 = \boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q}. \tag{4.31}$$

A detailed proof of this result can be found in [**reference**]. The possibility for an infinite-horizon LQR is desirable in the sense that it doesn't restrict the operation to a fixed time interval, however, the resulting optimal controller suffers from a worst performance when compared to finite-time horizon controllers.

4.3 Optimal State Estimators

As already discussed, the state-vector $\boldsymbol{x}(t)$ for a time t is a necessary information to calculate $\boldsymbol{u}(t)$ in state-feedback controllers, but is also inaccessible in practice. In the last chapter, a deterministic observer, namely the Luenberger observer, was derived to deal with this problem by the design of a observer gain $\boldsymbol{L}$. Despite being possible to utilize the Luenberger observer together with optimal controllers, this method suffers from the same design issues that the Pole-Placement method does, since they are dual problems. Therefore, this section develops an optimal state estimation approach which, from the estimation nature of the problem, relies on a statistical interpretation of the dynamical system and its observations.

First of all, consider a system perturbed by disturbances in the state-response and in the measurements, respectively $\boldsymbol{w} : \boldsymbol{R} \to \boldsymbol{R}^n$ and $\boldsymbol{v} : \boldsymbol{R} \to \boldsymbol{R}^p$. This configuration is illustrated at Fig. 4.2. If these disturbances are assumed to be white Gaussian noises, i.e., $\boldsymbol{w}_k \sim \mathcal{N}(0, \boldsymbol{Q}_{kf})$ and $\boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R}_{kf})$ where $\boldsymbol{Q}_{kf}$ and $\boldsymbol{R}_{kf}$ are covariances matrices of appropriate sizes, then it is possible to motivate a stochastic formulation of the linear State-Space model.

Definition 4.5. (Stochastic Discrete State-Space) Given an additive process noise $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{kf})$ with covariance $\boldsymbol{Q}_{kf} \in \mathbb{R}^{n\times n}$ and an additive measurement noise $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_{kf})$ with covariance $\boldsymbol{R}_{kf} \in \mathbb{R}^{p\times p}$, the stochastic version of a discrete-time State-Space model is given by the equations:

$$\begin{cases} \boldsymbol{x}_{k+1} &= \boldsymbol{A}_d\boldsymbol{x}_k + \boldsymbol{B}_d\boldsymbol{u}_k + \boldsymbol{w}_k \\ \boldsymbol{y}_k &= \boldsymbol{C}_d\boldsymbol{x}_k + \boldsymbol{v}_k \end{cases}. \tag{4.32}$$

With this formulation, the noises can actually represent both the effect of external disturbances and the uncertainty about the model accuracy in respect to the physical system. This is important since the dynamical models, even those developed by the first-principle methodology, are not guaranteed to be the absolute true representation of a dynamical system or process. The motivation for a discussion over a discrete State-Space model is to facilitate the interpretation of the statistical properties over finite data sample collections. Since it is always possible to convert a continuous-time State-Space into a discrete-time State-Space [**reference**], this discussion

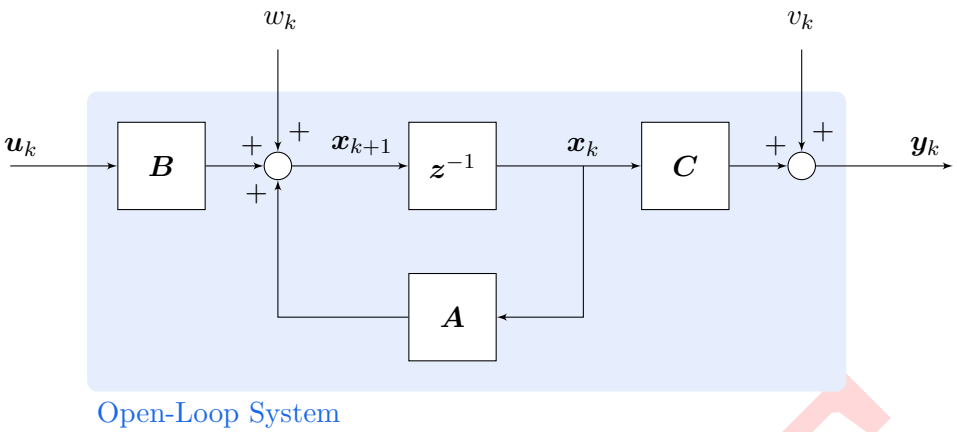


Figure 4.2: Block diagram of a stochastic discrete State-Space system.

can be easily extended to more general cases, and results for continuous-time systems will be addressed.

A direct result from Definition 4.5 is that, since $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are random variables, $\boldsymbol{x}_k$ and $\boldsymbol{y}_k$ are also random variables. More precisely, since the noises are Gaussian, both variables $\boldsymbol{x}_k \sim \mathcal{N}(\mu_{x_k}, \Sigma_{x_k})$ and $\boldsymbol{y}_k \sim \mathcal{N}(\mu_{y_k}, \Sigma_{y_k})$ are Gaussian, since this distribution is closed to linear operations [**reference**]. For this reason, a state and output trajectory over a discrete time interval is a collection of random variables following the Gaussian distribution, i.e., a stochastic process known as a Gaussian Process (GP) [**reference**].

Definition 4.6. (Gaussian Process) Given a Gaussian process defined as a time-series collection of random variables between a discrete time interval:

$$\boldsymbol{X} = \{\boldsymbol{x}_k \sim \mathcal{N}(\mu_{\boldsymbol{x}_k}, \Sigma_{\boldsymbol{x}_k}); k \in [0, K]\}\,. \tag{4.33}$$

If this data is obtained from a causal system, the joint probability of the random variables is modeled as:

$$p(\boldsymbol{X}) = p(\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_K) = \prod_{k=0}^{K} p(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, \cdots, \boldsymbol{x}_1, \boldsymbol{x}_0). \tag{4.34}$$

Using this definition, the dynamic response bmx_{k+1} of a system for a time instant $k+1$, given its trajectory until the actual state bmx_k, is given by the conditional probability $p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k, \cdots, \boldsymbol{x}_1, \boldsymbol{x}_0)$. One can assume, however, that an actual state stores enough information from the past states to unequivocally determine the state response, so that the conditional probability can be restated as:

$$p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k, \cdots, \boldsymbol{x}_1, \boldsymbol{x}_0) = p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k). \tag{4.35}$$

This assumption is known as the *Markov property*, and a process that exhibits this property is known as a Markov process [**reference**]. With this formulation, is always possible to marginalize the probability distribution of a single random variable $\boldsymbol{x}_{k+1}$ given a probability distribution over the initial state $\boldsymbol{x}_0$, which are related through the discrete Chapman-Kolmogorov equation:

$$p(\boldsymbol{x}_{k+1}) = p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) \cdots p(\boldsymbol{x}_1|\boldsymbol{x}_0)p(\boldsymbol{x}_0) = p(\boldsymbol{x}_0) \prod_{i=0}^{k} p(\boldsymbol{x}_{i+1}|\boldsymbol{x}_i). \tag{4.36}$$

Combining this assumption with Definition 4.6, in order to describe a Gaussian-Markov Process, and using the model from Definition 4.5 it is possible to completely define a stochastic dynamical system in respect to the random variables probabilities.

Definition 4.7. (Hidden Markov Model) Given a stochastic State-Space model as in Definition 4.5, and assuming the Markov property, a *Hidden Markov Model* for this dynamical system at a time instant k is defined through the two distributions:

$$\begin{cases} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k-1}) & \text{(Transition Distribution)} \\ p(\boldsymbol{y}_k|\boldsymbol{x}_k) & \text{(Emission Distribution)} \end{cases}. \tag{4.37}$$

Furthermore, since the noises $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are white Gaussian noises, these distributions are modeled as the Normal conditional distributions:

$$\begin{cases} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) &= \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}, \boldsymbol{Q}_{kf}) \\ p(\boldsymbol{y}_k|\boldsymbol{x}_k) &= \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{C}\boldsymbol{x}_k, \boldsymbol{R}_{kf}) \end{cases}. \tag{4.38}$$

A Hidden-Markov Model is a statistical interpretation of dynamical system that is well-known in the literature [**reference?**]. An illustration of the dependence between these variables as a graphical model is depicted in Fig. 4.3. Notice that, in this formulation, the state-vector $\boldsymbol{x}_k$ at each time instance is a latent variable whose actual value is not known, but can be observed through the emission variable $\boldsymbol{y}_k$. This raises the problem of state estimation as a common inference problem in engineering applications known as the *filtering*, which consists on infer the actual value of $\boldsymbol{x}_k$ given the entire history of observations $\boldsymbol{y}_0, \cdots, \boldsymbol{y}_k$ and control actions $\boldsymbol{u}_0, \cdots, \boldsymbol{u}_k$. Thus, the actual state can be obtained through the distribution:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k, \boldsymbol{y}_{k-1}, \cdots, \boldsymbol{y}_0, \boldsymbol{u}_{k-1}, \boldsymbol{u}_{k-2}, \cdots, \boldsymbol{u}_0), \tag{4.39}$$

from where a representative value, such as the expected value $\mathbb{E}\left[p(\boldsymbol{x}_k|\boldsymbol{y}_k, \cdots, \boldsymbol{y}_0, \boldsymbol{u}_{k-1}, \cdots, \boldsymbol{u}_0)\right]$ can be selected as the value used by the controller to determine an action.

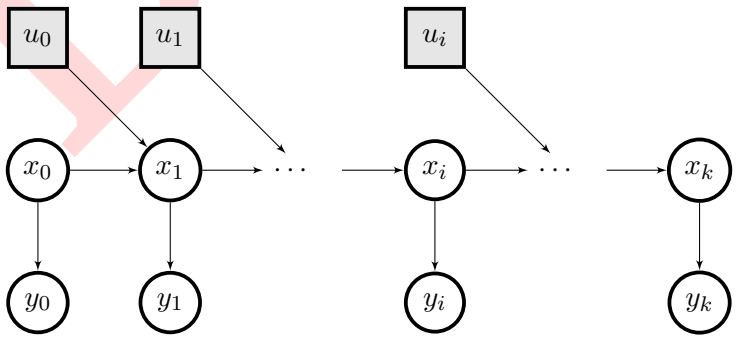


Figure 4.3: A Hidden Markov Model for a discrete-time system, where the shaded boxes indicates that the inputs are not random variables.

There are several approaches to solve the filtering and other related problems in statistical state estimation [**reference-Sarkka**], but the most popular method has been the Kalman filter [**reference**] which is discussed below for a discrete-time system.

Theorem 4.5. *(Kalman Filter) Given a Hidden Markov Model as in Definition 4.7 and initial state distribution $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{m}_0, \boldsymbol{P}_0)$, the filtering distribution can be solved in closed-form as:*

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k, \cdots, \boldsymbol{y}_0, \boldsymbol{u}_{k-1}, \cdots, \boldsymbol{u}_0) = \mathcal{N}(\hat{\boldsymbol{x}}_k, \boldsymbol{P}_k), \tag{4.40}$$

where the mean $\boldsymbol{m}_k$ and covariance $\boldsymbol{P}_k$ are recursively computed through the prediction and update steps below:

Predict Step:	***Update Step:***
$\hat{\bar{\boldsymbol{x}}}_k = \boldsymbol{A}\hat{\boldsymbol{x}}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}$	$\boldsymbol{K}_k = \bar{\boldsymbol{P}}_k\boldsymbol{C}^T(\boldsymbol{R}_{kf} + \boldsymbol{C}\bar{\boldsymbol{P}}_k\boldsymbol{C}^T)^{-1}$
$\bar{\boldsymbol{P}}_k = \boldsymbol{A}\boldsymbol{P}_{k-1}\boldsymbol{A}^T + \boldsymbol{Q}_{\boldsymbol{kf}}$	$\hat{\boldsymbol{x}}_k = \hat{\bar{\boldsymbol{x}}}_k + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k)$
	$\boldsymbol{P}_k = \bar{\boldsymbol{P}}_k - \boldsymbol{K}_k(\boldsymbol{R}_{kf} + \boldsymbol{C}\bar{\boldsymbol{P}}_k\boldsymbol{C}^T)\boldsymbol{K}_k^T$

(4.41)

Proof. Consider the transition and emission distributions given as:

$$\begin{cases} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) &= \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}, \boldsymbol{Q}_{kf}) \\ p(\boldsymbol{y}_k|\boldsymbol{x}_k) &= \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{C}\boldsymbol{x}_k, \boldsymbol{R}_{kf}) \end{cases}. \tag{4.42}$$

Consider, now, the joint distribution of $(\boldsymbol{x}_{k-1}, \boldsymbol{x}_k)$ conditioned on measurement history $(\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1})$. Since the joint and condition distributions between two Gaussians distributions are themselves Gaussian, this density can be expressed as:

$$\begin{aligned} p(\boldsymbol{x}_k, \boldsymbol{x}_{k-1}|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) &= p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) \\ &= \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}, \boldsymbol{Q}_{kf})\mathcal{N}(\boldsymbol{x}_{k-1}|\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{P}_{k-1}) \\ &= \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_{k-1} \\ \boldsymbol{x}_k\end{bmatrix}\middle|\middle|\begin{bmatrix}\hat{\boldsymbol{x}}_{k-1} \\ \boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}\end{bmatrix}, \begin{bmatrix}\boldsymbol{P}_{k-1} & \boldsymbol{P}_{k-1}\boldsymbol{A}^T \\ \boldsymbol{A}\boldsymbol{P}_{k-1} & \boldsymbol{A}\boldsymbol{P}_{k-1}\boldsymbol{A}^T + \boldsymbol{Q}_{kf}\end{bmatrix}\right) \end{aligned}. \tag{4.43}$$

Marginalizing the joint distribution, it is possible to obtain the distribution of $\boldsymbol{x}_k$ conditioned in the measurement history as:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) = \mathcal{N}(\underbrace{\boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}}_{\hat{\bar{\boldsymbol{x}}}_k}, \underbrace{\boldsymbol{A}\boldsymbol{P}_{k-1}\boldsymbol{A}^T + \boldsymbol{Q}_{kf}}_{\bar{\boldsymbol{P}}_k}), \tag{4.44}$$

which completes the *prediction step* of the Kalman filter. For the update step, consider the joint distribution between $(\boldsymbol{x}_k, \boldsymbol{y}_k)$, the actual state and actual measurement:

$$\begin{aligned} p(\boldsymbol{x}_k, \boldsymbol{y}_k|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) &= p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) \\ &= \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{C}\boldsymbol{x}_k, \boldsymbol{R}_{kf})\mathcal{N}(\boldsymbol{x}_k|\hat{\bar{\boldsymbol{x}}}_k, \bar{\boldsymbol{P}}_k) \\ &= \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_k \\ \boldsymbol{y}_k\end{bmatrix}\middle|\middle|\begin{bmatrix}\hat{\bar{\boldsymbol{x}}}_k \\ \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k\end{bmatrix}, \begin{bmatrix}\bar{\boldsymbol{P}}_k & \bar{\boldsymbol{P}}_k\boldsymbol{C}^T \\ \boldsymbol{C}\bar{\boldsymbol{P}}_k & \boldsymbol{C}\bar{\boldsymbol{P}}_k\boldsymbol{C}^T + \boldsymbol{R}_{kf}\end{bmatrix}\right) \end{aligned}. \tag{4.45}$$

Therefor, the filtering distribution can be obtained as:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k, \boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) = \mathcal{N}(\boldsymbol{x}_k|\hat{\boldsymbol{x}}_k, \boldsymbol{P}_k), \tag{4.46}$$

where $\hat{\boldsymbol{x}}_k$ and $\boldsymbol{P}_k$ are the computations of the update step, given as:

$$\begin{cases} \hat{\boldsymbol{x}}_k = \hat{\bar{\boldsymbol{x}}}_k + \bar{\boldsymbol{P}}_k\boldsymbol{C}^T(\boldsymbol{R}_{kf} + \boldsymbol{C}\bar{\boldsymbol{P}}_k\boldsymbol{C}^T)^{-1}(\boldsymbol{y}_k - \boldsymbol{C}\boldsymbol{x}_k) \\ \boldsymbol{P}_k = \bar{\boldsymbol{P}}_k - \bar{\boldsymbol{P}}_k\boldsymbol{C}^T(\boldsymbol{R}_{kf} + \boldsymbol{C}\bar{\boldsymbol{P}}_k\boldsymbol{C}^T)^{-1}\boldsymbol{C}\bar{\boldsymbol{P}}_k \end{cases}. \tag{4.47}$$

Finally, making $\boldsymbol{K}_k = \bar{\boldsymbol{P}}_k\boldsymbol{C}^T(\boldsymbol{R}_{kf} + \boldsymbol{C}\bar{\boldsymbol{P}}_k\boldsymbol{C}^T)^{-1}$ establishes the theorem. □

Notice that the Kalman filter solution implies that an "open-loop" estimation of the system, $\hat{\bar{\boldsymbol{x}}}_k$, can be corrected by the equation:

$$\hat{\boldsymbol{x}}_k = \hat{\bar{\boldsymbol{x}}}_k + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k), \tag{4.48}$$

which, by expanding $\hat{\bar{\boldsymbol{x}}}_k$, results in a similar state-equation expression for a system with a deterministic observer as defined earlier, but for a time-varying gain $\boldsymbol{K}_k$:

$$\begin{aligned} \hat{\boldsymbol{x}}_k &= \boldsymbol{A}\hat{\bar{\boldsymbol{x}}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k) \\ &= (\boldsymbol{A} - \boldsymbol{K}_k\boldsymbol{C})\,\hat{\bar{\boldsymbol{x}}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{K}_k\boldsymbol{y}_k \end{aligned}. \tag{4.49}$$

Notice that, despite the similarity with Luenberger observers, the Kalman gain has the advantage that the gain matrix $\boldsymbol{K}_k$ is not resulting from a designer procedure but rather from solving an inference problem whose only additional information needed are the covariances $\boldsymbol{Q}_{kf}$ and $\boldsymbol{R}_{kf}$, which can be assumed or estimated. In addition to that, it can be shown that the recursive steps that solves the filtering problem are also responsible for minimizing the expectation of the error between the measurements and actual state response, given as:

$$\boldsymbol{J} = \mathbb{E}\left[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^T\right]. \tag{4.50}$$

Since this method minimizes some cost function, it is indeed an optimal estimator. In fact, the Kalman filter can be formulated through an optimal control formulation, which motivates the development of a continuous-time version of this estimator, as shown below.

Theorem 4.6. *(Kalman-Bucy Filter) Consider a continuous-time State-Space linear system subject to additive process noise* $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{kf})$ *and measurement noise* $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_{kf})$*, where the covariances* $\boldsymbol{Q}_{kf} \in \mathbb{R}^{n\times n}$ *and* $\boldsymbol{R}_{kf} \in \mathbb{R}^{p\times p}$ *represents the* power spectral density *of the noises. In this case, for an estimated state* $\hat{\boldsymbol{x}}(t)$ *at time* t*, the error covariance:*

$$\boldsymbol{J}(\boldsymbol{x}, \hat{\boldsymbol{x}}, t) = \mathbb{E}\left\{[\boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)][\boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)]^T\right\} \tag{4.51}$$

is minimized by $\hat{\boldsymbol{x}}(t)$ *obtained through the system:*

$$\dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{A}\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{K}_e(t)\left(\boldsymbol{y}(t) - \boldsymbol{C}\hat{\boldsymbol{x}}(t)\right), \tag{4.52}$$

where $\boldsymbol{K}_e(t) = \boldsymbol{P}_e(t)\boldsymbol{C}\boldsymbol{R}^{-1}$*, being* $\boldsymbol{P}_e(t)$ *the solution of the Riccati differential matrix equation:*

$$\dot{\boldsymbol{P}}_e(t) = \boldsymbol{A}\boldsymbol{P}_e(t) + \boldsymbol{P}_e(t)\boldsymbol{A}^T - \boldsymbol{P}_e(t)\boldsymbol{C}^T\boldsymbol{R}_{kf}^{-1}\boldsymbol{C}\boldsymbol{P}_e(t) + \boldsymbol{Q}_{kf}, \tag{4.53}$$

with initial condition $\boldsymbol{P}_e(t_0) = \mathbb{E}\left\{[\boldsymbol{x}(t_0) - \bar{\boldsymbol{x}}(t_0)][\boldsymbol{x}(t_0) - \bar{\boldsymbol{x}}(t_0)]^T\right\}$ *for* $t_0 > -\infty$*.*

A detailed proof of this theorem can be found in [**reference**]. The formulation of an optimal state estimator further emphasizes the duality between controllers and observers. Notice that the gain $\boldsymbol{K}_e(t)$ of a Kalman-Bucy filter depends on a matrix $\boldsymbol{P}_e(t)$ that solves the exact same Riccati differential equation as the one in Theorem 4.3, but forward in time for matrices $\boldsymbol{A}^T$ and $\boldsymbol{C}^T$ instead of $\boldsymbol{A}$ and $\boldsymbol{B}$. Notice, however, that these filters are not limited to a terminal time T, since the boundary condition only requires information on the initial time $t_0 > -\infty$, which implies that they could also be used in forward infinite-horizon operations. Another direct result of the duality is that is possible to derive a estimator considering $t_0 \to -\infty$, which, as in Theorem 4.4, results in a time-invariant filter with gain K_e [**reference**].

4.4 Linear Quadratic Gaussian (LQG)

This chapter presented formulation for controllers and state estimators that are optimal, in the sense that they minimize some cost function based on information about the model and measurements. In the previous chapter, the Theorem 3.5 stated that a controller and a estimator can be designed separately, and a resulting control by feedback of estimated states is always feasible. In this section, a similar configuration is shown for optimal control and estimation operations, as depicted in Fig. 4.4. In this illustration, the open-loop system can represent either a mathematical model (possibly non-linear and time-varying) being simulated or a physical system whose output is only measured through some sensor device.

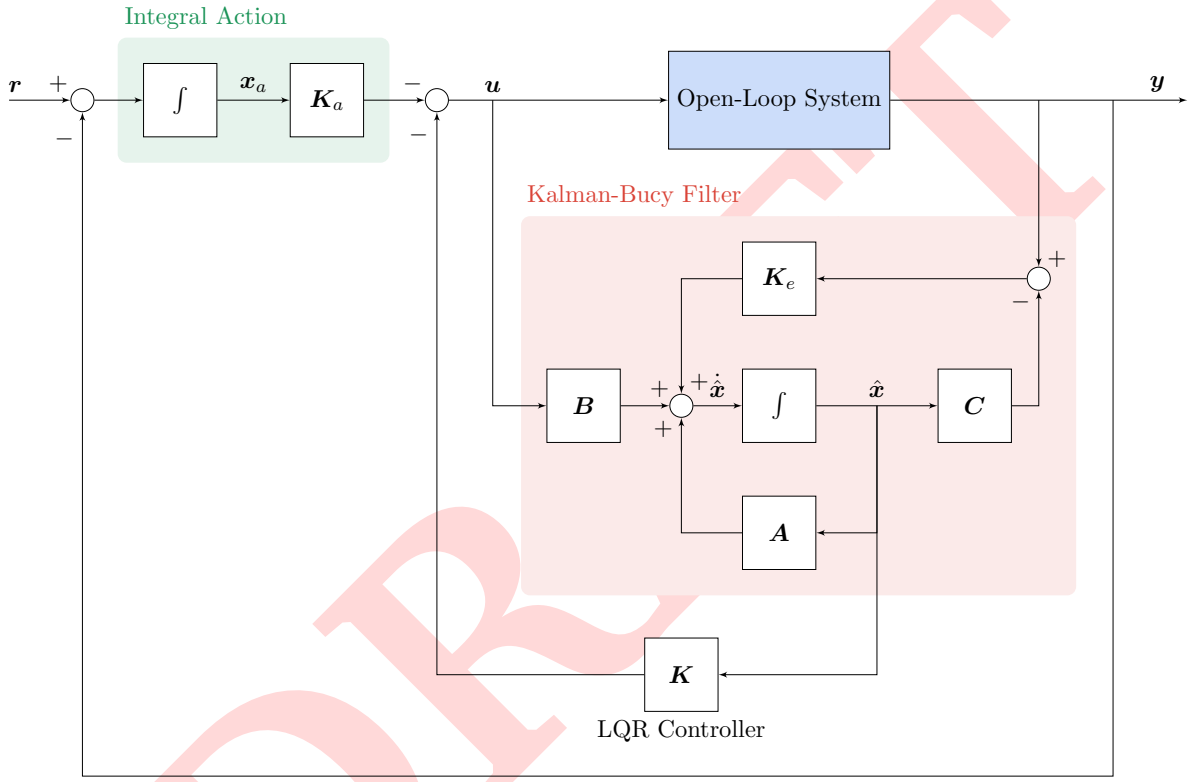


Figure 4.4: Block diagram of a Linear Quadratic Gaussian control configuration.

The control configuration that connects a optimal control action of a Linear Quadratic Regulator together with the estimation states from a Kalman filter (or Kalman-Bucy filter) is know as a Linear Quadratic Gaussian (LQG) controller [**reference**]. Since the controller is independent of the estimator, it is also possible to include integral action to the configuration, as shown in the block diagram, presenting this architecture as a general configuration that can deal with a broad range of applications. Because of the stochastic nature of the estimation and the optimal procedure of the control, this configuration is central to the field of Stochastic Optimal Control [**reference?**]. A formal definition is given below.

Definition 4.8. (Linear Quadratic Gaussian) Consider a stochastic system in State-Space representation:

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{w}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{v}(t) \end{cases}, \tag{4.54}$$

whose estimated state-vector $\hat{\boldsymbol{x}}(t)$ is reconstructed from a Kalman-Bucy filter and whose input signal $\boldsymbol{u}(t)$ is calculated through a finite-horizon LQR. The Linear Quadratic Gaussian (LQG) control for the horizon $t \in [t_0, T]$, with $-\infty < t_0 \leq T < \infty$, is defined as:

$$\dot{\hat{\boldsymbol{x}}}(t) = [\boldsymbol{A} - \boldsymbol{K}_e(t)\boldsymbol{C} - \boldsymbol{B}\boldsymbol{K}(t)]\,\hat{\boldsymbol{x}}(t) + \boldsymbol{K}_e(t)\boldsymbol{y}(t), \tag{4.55}$$

where $\boldsymbol{K}(t) = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t)$ and $\boldsymbol{K}_e(t) = \boldsymbol{P}_e(t)\boldsymbol{C}\boldsymbol{R}^{-1}$ are, respectively, the LQR and Kalman-Bucy gains for matrices $\boldsymbol{P}(t)$ and $\boldsymbol{P}_e(t)$ that solves the Riccati differential equations:

$$\begin{cases} -\dot{\boldsymbol{P}}(t) = \boldsymbol{A}^T\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{A} - \boldsymbol{P}(t)\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t) + \boldsymbol{Q} \\ \dot{\boldsymbol{P}}_e(t) = \boldsymbol{A}\boldsymbol{P}_e(t) + \boldsymbol{P}_e(t)\boldsymbol{A}^T - \boldsymbol{P}_e(t)\boldsymbol{C}^T\boldsymbol{R}_{kf}^{-1}\boldsymbol{C}\boldsymbol{P}_e(t) + \boldsymbol{Q}_{kf} \end{cases} \tag{4.56}$$

for boundary conditions $\boldsymbol{P}(T) = \boldsymbol{Q}_f$ and $\boldsymbol{P}_e(t_0) = \mathbb{E}\left\{[\boldsymbol{x}(t_0) - \bar{\boldsymbol{x}}(t_0)][\boldsymbol{x}(t_0) - \bar{\boldsymbol{x}}(t_0)]^T\right\}$.

Notice that another advantage of this controller-estimator configuration is that the solution for both Riccati differential equations does not depends on the states at any time $\boldsymbol{x}(t)$, with the exception of the boundary condition $\boldsymbol{P}_e(t_0)$ which can in any case be obtained before the system operation. Therefore, the values for matrices $\boldsymbol{P}(t)$ and $\boldsymbol{P}_e(t)$, and consequently $\boldsymbol{K}(t)$ and $\boldsymbol{K}_e(t)$, for $t \in [t_0, T]$, can be evaluated beforehand and then applied in the on-line operation.

The LQG controller can also be used for tracking non-constant references, given that the system can be augmented while maintaining controllability. The augmented system is used to calculate the controller gains $\boldsymbol{K}(t)$ using Definition 4.4, while the estimator gains $\boldsymbol{K}_e(t)$ are optimized through the original system. Given that it can optimally solve both the regulator and tracking problems, while still accounting for uncertainty in the system, the Linear Quadratic Gaussian poses as a high performing multi-purpose controller. The resulting augmented system can be represented by the closed-loop state equation:

$$\begin{cases} \begin{bmatrix} \dot{\hat{\boldsymbol{x}}}(t) \\ \dot{\hat{\boldsymbol{x}}}_a(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}(t) - \boldsymbol{K}_e(t)\boldsymbol{C} & -\boldsymbol{B}\boldsymbol{K}_a(t) \\ -\boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}(t) \\ \hat{\boldsymbol{x}}_a(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{K}_e(t) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{r}(t) \end{bmatrix} \\ \boldsymbol{y}(t) = \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{x}_a(t) \end{bmatrix} \end{cases}. \tag{4.57}$$

For the sake of illustration, consider the same system as controlled in (3.11). Consider, now, that it is perturbed by noises $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{kf})$ and $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_{kf})$, with covariances:

$$\boldsymbol{Q}_{kf} = \begin{bmatrix} 0.2741 & 0.0036 \\ 0.0036 & 0.0024 \end{bmatrix} \quad \boldsymbol{R}_{kf} = 0.233. \tag{4.58}$$

A pair of LQG controllers for tracking are obtained by augmenting the system and solving the controller Ricatti differential equation for weights in the form $\boldsymbol{Q} = diag(\alpha_1, \alpha_2, \alpha_3)$ and $\boldsymbol{R} = \beta$, for terminal weight $\boldsymbol{P}(T) = \boldsymbol{Q}$. The estimator is solved forward in time for initial condition $\boldsymbol{P}_e(t_0) = \boldsymbol{Q}_{kf}$. The resulting simulations are shown in Fig. 4.5, demonstrating how the controller can be more or less aggressive given the choice of weighting matrices.

4.5 Stability and Robustness Analysis

In the last chapter, the poles of the feedback controlled system where directly assigned from the pole-placement method. In this chapter, however, the controllers were results of optimization procedures, so it is necessary to discuss whether or not these controllers are stable and robust given the criteria already presented.

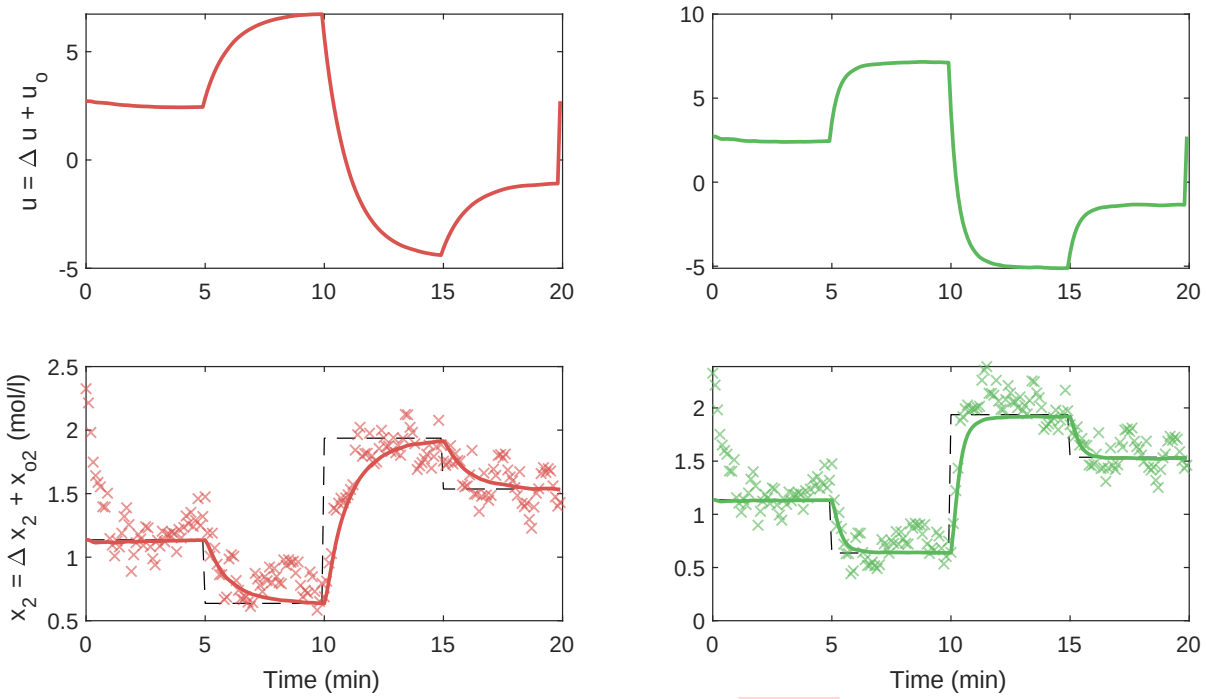


Figure 4.5: Simulation of the LQG controllers showing the input signal (up) and correspondent state response (down). The markers denotes the observations from the real system. The control on the left was solved for $\boldsymbol{Q}^{(1)} = \text{diag}(20, 20, 5000)$ and $\boldsymbol{R}^{(1)} = 75$, and the controller on the right was solved for $\boldsymbol{Q}^{(2)} = \text{diag}(1, 1, 10000)$ and $\boldsymbol{R}^{(2)} = 20$.

First of all, the stability of the closed-loop system via Linear Quadratic Regulator is discussed. One might conclude that, since this controller is obtained from a optimization process, the resulting controller can not be unstable. Consider, however, the popular counter-intuition: consider a scalar system $\dot{x}(t) = x(t) + u(t)$, with $R = 1$ and $Q = 0$, such that its associated LQR cost function is:

$$J(x, u, t_0) = \int_{t_0}^{\infty} u^2(t)dt. \tag{4.59}$$

In this case, the optimal control action is always $u(t) = 0$, for $t \in [t_0, \infty]$. However, this control action results in the system $\dot{x}(t) = x(t)$ which is clearly BIBO unstable. Therefore, there are conditions in whether the optimization process results in a stable or unstable closed-loop system, as shown by the following theorem.

Theorem 4.7. *(LQR Assymptoptic Stability) Consider the infinite-horizon Linear Quadratic Regulator as defined in Theorem 4.4. If the system is observable, then the closed-loop matrix* $\boldsymbol{A}_{cl} = (\boldsymbol{A} - \boldsymbol{BK}) = (\boldsymbol{A} - \boldsymbol{BR}^{-1}\boldsymbol{B}^T\boldsymbol{P})$ *is BIBO stable.*

Proof. Consider, without loss of generalization, that $\boldsymbol{Q} = \boldsymbol{C}^T\boldsymbol{C}$, which is still positive definite. In this case $\boldsymbol{C} = \boldsymbol{Q}^{1/2}$, which is nonsingular and the observability assumption holds. Furthermore, consider the LQR cost function given an optimal input trajectory $\boldsymbol{u}^* : \mathbb{R} \to \mathbb{R}^r$:

$$J(\boldsymbol{x}, \boldsymbol{u}^*, t_0) = \int_{t_0}^{\infty} \left(\boldsymbol{x}^T(\boldsymbol{C}^T\boldsymbol{C})\boldsymbol{x} + (\boldsymbol{u}^*)^T\boldsymbol{R}\boldsymbol{u}^*\right) dt = \int_{t_0}^{\infty} \left(\boldsymbol{y}^T\boldsymbol{y} + (\boldsymbol{u}^*)^T\boldsymbol{R}\boldsymbol{u}^*\right) dt. \tag{4.60}$$

The optimal output trajectory resulting from this control action is given by $\boldsymbol{y}^*(t) = \boldsymbol{C}\boldsymbol{x}^*(t)$, where $\boldsymbol{x}^*(t)$ is the correspondent optimal state trajectory. Since $\boldsymbol{u}^*(t)$ was obtained by solving

the algebraic Riccati equation on Theorem 4.4, a well-known result is that the optimal cost function is bounded:

$$\boldsymbol{V}^*(\boldsymbol{x}, t_0) \leq \boldsymbol{x}^T(t_0)\boldsymbol{P}\boldsymbol{x}(t_0), \tag{4.61}$$

and, therefore, (4.60) must also be bounded. This is only possible if $\boldsymbol{y}^*(t) \to 0$ and $\boldsymbol{u}^*(t) \to 0$ as $t \to \infty$, which, since the system is observable, directly implies $\boldsymbol{x}^*(t) \to 0$ as $t \to \infty$, concluding that the system is BIBO stable. □

Notice that, in contrast to the Pole-Placement method where the stability must be explicitly determined by the designer, the LQR controller can guarantee a stable closed-loop system directly by the observability property, even when the open-loop system is unstable. Since the BIBO stability (Definition 2.6) discusses the magnitude of a system response as $t \to \infty$, the results are derived for infinite-horizon LQR. Despite of this, it is also possible to interpret some notion of stable response for finite-horizon LQR, since the cost function is always bounded by the optimal cost as conditioned by the Hamilton-Jacobi equation.

Now, the discussion turns to the robustness properties of these optimal controllers. It can be shown that the LQR has a gain margin $GM = \infty$ and a phase margin $PM \geq 60°$ **[reference]**. This assumption can be verified from a visual inspection of the Nyquist plot, as shown in Fig. 4.6. The reason for these values comes from the fact that the Nyquist diagram of an optimal regulator never crosses the unit circle centered at $s = -1$ (implied by the Return Difference Equality). Therefore, the diagram never crosses the point $s = -1$ no matter how much the gain K is changed, which establishes the infinite gain margin. In addition, the angle that the closest permissible point of a unit circle centered in $s = 0$ makes with the point $s = -1$ is exactly 60°, which establishes the lower bound on the gain phase.

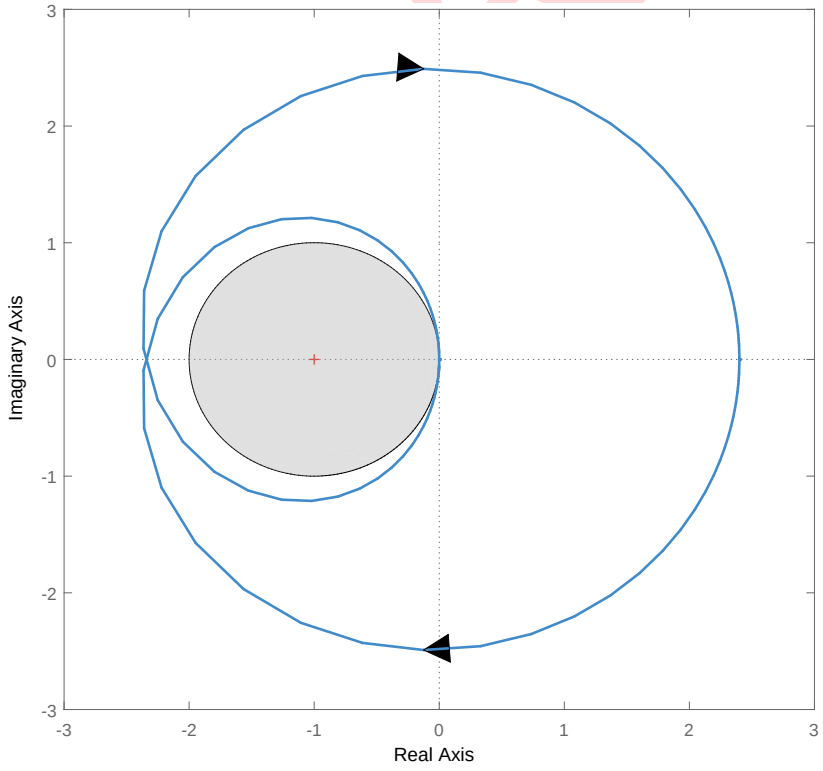


Figure 4.6: Nyquist diagram of a single state system with a open-loop unstable pole but which is closed-loop stable. The gain calculated by a LQR stabilizing controller never crosses the unit circle centered at $s = -1 + j0$.

Of course, there is no physical support of saying that a controller device exhibits a infinite gain margin, since there is always uncertainty about the model, the instrumentation and the

environment. However, the fact that the mathematical model displays infinite gain margin allows for the assumption that the physical controller will have an extremely large margin. In fact, the uncertainty just mentioned is the main motivation into using optimal state estimators to reconstruct the state-vector from data, and it was shown by [**reference**] that an optimal controller-estimator configuration, i.e., the Linear Quadratic Gaussian, has no guaranteed margins.

Chapter 5

Receding Horizon Optimal Control

The previous chapter introduced the concept of finite-horizon optimal control, presenting the Linear Quadratic Regulator as a solution to the condition imposed by the Hamilton-Jacobi equation. Despite being a popular solution with nice properties, there are some downsides of the LQR standard formulation. This chapter explores a optimal controller formulation based on a receding horizon control strategy. In the first section the motivation and definition of the moving horizon strategy are stated. The second sections explores the possibility of combining the receding horizon strategy with an iterative linearization procedure to yield an optimal operation in respect to the nonlinear system itself.

5.1 Optimization in Moving Horizons

A major disadvantage of the standard finite-horizon optimal controller is that the time in which the system will operate becomes restricted to the time horizon used for the optimization. This might becomes a problem in the case that the system is interrupted or subjected to strong disturbances, given that corrective action would require time and the desired state could not be achieved in the given horizon. Similarly, the restricted time horizon is also problematic in the case that the system should track a non-constant reference signal, since it also has be assumed of fixed length for the optimization to account for the entire signal.

A possible solution to this issue is to perform a Receding Horizon Control (RHC) strategy [**references**]. In the context of optimal control, a receding horizon controller is also known in literature as the *Model-Predictive Controller* [**references**], although this name is usually associated with constrained optimization procedures. The strategy is defined as follows:

Definition 5.1. (Receding-Horizon Control) Considers a linear system in State-Space form and a control horizon of size $T \in \mathbb{R}^+$. The *Receding-Horizon Control* strategy for an optimal controller of this system is represented as:

$$\dot{\boldsymbol{x}}(t) = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}(t))\,\boldsymbol{x}(t), \tag{5.1}$$

where $\boldsymbol{K}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t)$ represents the optimal gain from the LQR cost function (Definition 4.3) for an iterative time interval $[t, t+T]$, and $\boldsymbol{P}(t)$ which solves the the Riccati differential equation:

$$-\dot{\boldsymbol{P}}(t) = \boldsymbol{A}^T\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{A} - \boldsymbol{P}(t)\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t) + \boldsymbol{Q}, \tag{5.2}$$

with terminal condition $\boldsymbol{P}(t+T) = \boldsymbol{Q}_f$.

In the optimization community, this problem is also popular in the following notation, where the moving horizon becomes explicit:

$$\begin{aligned} \textbf{minimize} \quad & \int_{t}^{t+T} \left(\boldsymbol{x}^T \boldsymbol{Q} \boldsymbol{x} + \boldsymbol{u}^T \boldsymbol{R} \boldsymbol{u}\right) d\tau \\ \textbf{s.t.} \quad & \left.\begin{aligned} \dot{\boldsymbol{x}}(\tau) &= \boldsymbol{A}\boldsymbol{x}(\tau) + \boldsymbol{B}\boldsymbol{u}(\tau) \\ \boldsymbol{u}(\tau) &= \pi(\boldsymbol{x}(t), \cdots, \boldsymbol{x}(\tau)) \end{aligned}\right\} \forall \tau \in [t, t+T] \end{aligned} . \tag{5.3}$$

This definition basically states that the optimization to be performed is similar to the standard case, but, since the time horizon is now a moving window of size T, the gain matrices $\boldsymbol{K} : \mathbb{R} \to \mathbb{R}^{r \times n}$ changes together with time. This is a consequence for the fact the the Ricatti differential equation has to be solved backwards in time for each new horizon at each new time instance. Of course, this means that for each which time interval that is optimized, only the first action is actually applied to the system. The Fig. 5.1 illustrates this control strategy.

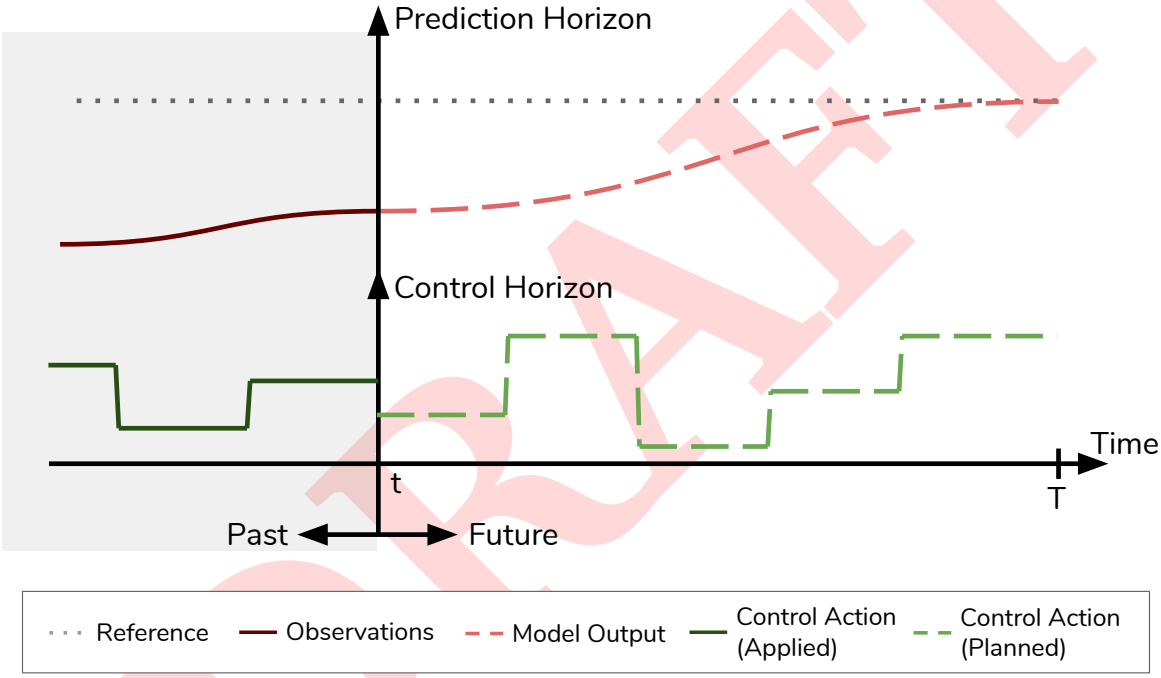


Figure 5.1: Illustration of the Receding Horizon Control strategy.

From the Separation Principle, it is always possible to increment this approach by introducing a optimal state estimator in a receding horizon fashion, as well. In this case, similarly to the controller, the optimal estimator gains are obtained for the entire horizon, but only the first gain is used to correct the state, before the horizon is updated.

[**properties?**]

5.2 Linear Parameter-Varying Models

The optimal controllers discussed so far suffers from a performance disadvantage when regarding the control of nonlinear systems. As shown in Theorem 2.5, a nonlinear system can be approximated by a linear model around a specific steady-state point. The problem, however, is that the optimization accounts for a nominal linear system that, in the case of a linearized model, will not represent the real dynamics of the nonlinear system as its states deviates from the steady-state value used. For this reason, the LQR solution will not provide an optimal solution in the light of the real system.

In this section, a solution to the nonlinear problematic is presented through an iterative linearization process, where the full dynamics are approximated by a series of linear models obtained for each time instance. This approach is referred in literature as Linear Switched Systems [**references**]. Before any further discussion, first consider the following definition.

Definition 5.2. (Linear Parameter-Varying System) A *Linear Parameter-Varying* (LPV) State-Space representation describing a system with state vector $\boldsymbol{x} : \mathbb{R} \to \mathbb{R}^n$, output vector $\boldsymbol{y} : \mathbb{R} \to \mathbb{R}^p$ and input vector $\boldsymbol{u} : \mathbb{R} \to \mathbb{R}^r$ is given by the system of equations:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}(\boldsymbol{\theta}(t))\boldsymbol{x}(t) + \boldsymbol{B}(\boldsymbol{\theta}(t))\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}(\boldsymbol{\theta}(t))\boldsymbol{x}(t) + \boldsymbol{D}(\boldsymbol{\theta}(t))\boldsymbol{u}(t) \end{cases}, \tag{5.4}$$

where $\boldsymbol{A}(\cdot) \in \mathbb{R}^{n\times n}$, $\boldsymbol{B}(\cdot) \in \mathbb{R}^{n\times r}$, $\boldsymbol{C}(\cdot) \in \mathbb{R}^{p\times n}$ and $\boldsymbol{D}(\cdot) \in \mathbb{R}^{p\times r}$ are appropriate functions of the parameter function $\boldsymbol{\theta} : \mathbb{R} \to \mathbb{R}^z$.

The LPV State-Space model is a linear realization of a system that can change with time through the result of a function $\boldsymbol{\theta}(\cdot)$. The first discussions on this representation for dynamical models were motivated from the Gain Scheduling control method [**references**], where it serves as an linear embedding of the nonlinear model. From Theorem 2.5 it is clear that a possible formulation for such a model is given by:

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t)) = \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) \end{cases}, \tag{5.5}$$

in which case it is possible to set $\boldsymbol{\theta}(t) = \left[\boldsymbol{x}(t-\delta t), \boldsymbol{u}(t-\delta t)\right]^T$ and denote the Jacobians as:

$$\begin{aligned} \boldsymbol{A}(\boldsymbol{x}(t-\delta t), \boldsymbol{u}(t-\delta t)) &= \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}(t-\delta t),\boldsymbol{u}(t-\delta t)}; \quad \boldsymbol{B}(\boldsymbol{x}(t-\delta t), \boldsymbol{u}(t-\delta t)) = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}(t-\delta t),\boldsymbol{u}(t-\delta t)}; \\ \boldsymbol{C}(\boldsymbol{x}(t-\delta t), \boldsymbol{u}(t-\delta t)) &= \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}(t-\delta t),\boldsymbol{u}(t-\delta t)}; \quad \boldsymbol{D}(\boldsymbol{x}(t-\delta t), \boldsymbol{u}(t-\delta t)) = \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}(t-\delta t),\boldsymbol{u}(t-\delta t)} \end{aligned} \tag{5.6}$$

The formulation in (5.6) might not be a perfect linearization of the system, since the values $\boldsymbol{x}(t)$ and $\boldsymbol{u}(t)$ are not constrained to yield $\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t))$, i.e., they may be non-stationary points of the system. In the case that $\delta t > 0$, these matrices functions represents a causal switching law for linear systems [**reference**], and the LPV system tends to a nonlinear system as $\delta t \to 0$. For some families of state and input trajectories it is possible to guarantee stability and other asymptotic system properties [**reference**], which are heavily based on Lyapunov's theory [**reference**].

Consider the linearization Jacobians stated in (2.34). In the case that only the two first

states are modelled, a LPV formulation for that system is given as:

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t) = \begin{bmatrix} -\theta^{(1)} - K_{AB} - 2K_{AC}\boldsymbol{\theta}_1^{(2)} & 0 \\ K_{AB} & -\theta^{(1)} - K_{BC} \end{bmatrix} \Delta\boldsymbol{x}(t) + \begin{bmatrix} \rho_{in}^{(A)} - \boldsymbol{\theta}_1^{(2)} \\ -\boldsymbol{\theta}_2^{(2)} \end{bmatrix} \Delta u(t) \\ \boldsymbol{y}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Delta\boldsymbol{x}(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Delta u(t) \end{cases}, \quad (5.7)$$

where $\theta^{(1)} = u(t - \delta t)$ and $\boldsymbol{\theta}^{(2)} = \boldsymbol{x}(t - \delta t)$. Notice that the matrix $\boldsymbol{A}(\cdot)$ in this case should always result in a stable system, since the variables in the diagonal are constrained to be positive, given their physical meaning. A simulation is shown in Fig. **??** for this system, given a sequence of different amplitude steps as inputs, to visualize how well this system response can approxima te the nonlinear response.

[fig]

5.3 Receding Horizon Control for Switched Systems

Combining the receding horizon strategy together with the linear switching approach, it is possible to develop a method to controlling nonlinear systems by optimizing over a series of linearized systems. The receding horizon, in this case, becomes a necessity since the optimization has to be repeated every time that the matrices $\boldsymbol{A}(\cdot)$ and $\boldsymbol{B}(\cdot)$ changes. A formal definition of such controller is given below.

Definition 5.3. (RHC for LPV Systems) Considers a linear parameter-varying system in State-Space form, a control horizon of size $T \in \mathbb{R}^+$ and a switching time distance δt. The receding horizon control strategy for an optimal controller of this switched system is represented as:

$$\dot{\boldsymbol{x}}(t) = \left(\boldsymbol{A}(\boldsymbol{\theta}(t)) - \boldsymbol{B}(\boldsymbol{\theta}(t))\boldsymbol{K}(t)\right)\boldsymbol{x}(t), \quad (5.8)$$

where $\boldsymbol{\theta}(t) = \left[\boldsymbol{x}(t - \delta t), \boldsymbol{u}(t - \delta t)\right]$ is the parameter function and $\boldsymbol{K}(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^T(\boldsymbol{\theta}(t))\boldsymbol{P}(t)$ represents the optimal gain for an iterative time interval $[t, t + T]$, given $\boldsymbol{P}(t)$ which solves the the Riccati differential equation:

$$-\dot{\boldsymbol{P}}(t) = \boldsymbol{A}^T(\boldsymbol{\theta}(t))\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{A}(\boldsymbol{\theta}(t)) - \boldsymbol{P}(t)\boldsymbol{B}(\boldsymbol{\theta}(t))\boldsymbol{R}^{-1}\boldsymbol{B}^T(\boldsymbol{\theta}(t))\boldsymbol{P}(t) + \boldsymbol{Q}, \quad (5.9)$$

with terminal condition $\boldsymbol{P}(t + T) = \boldsymbol{Q}_f$.

A simulation of controlling the system from (5.7) is shown at Fig. **??**, demonstrating the quality of this approach to directly control the nonlinear system. For this configuration, the transient characteristics of the response can then be tuned by the design of $\boldsymbol{Q}$, $\boldsymbol{R}$ and the time parameters $\boldsymbol{T}$ and δt.

[fig]

As stated earlier, it is also possible to apply this approach to stochastic systems, which are modelled as the nominal State-Space models perturbed by white Gaussian noise. The main difference is that the switching is done through the estimated states, making the approximation to the nonlinear response dependent on the estimator capabilities. A simulation of such configuration is shown in Fig. **??**. Notice how the action of noise made the closed-loop system exhibits some pseudo-periodic behavior, while still being able to follow the reference signal.

[fig]

Chapter 6

Methodology

This chapter details the methodology used in order to attest the discussion presented until now. The experiments were performed in a simulated environment, using the MATLAB software [**reference**] as the framework to compute such simulations. The results obtained are presented and discussed in the next chapter.

6.1 Non-Isothermal Continuous Stirred Tank

The chemical reactor used for the experiments was the non-isothermal Continuous Stirred Tank (CSTR) presented by [**reference-Engel**]. This system is a realization of a class of systems that characterize a wide range of industrial applications, hence being considered a classical benchmark for reactive systems. In particular, the system proposed for the experiment is proposed as a multiple-input multiple-output (MIMO) system which is highly nonlinear, with non-minimum phase behavior and unmeasurable states, thus makes it very challenging to control. This system is comprised by a tank containing a dilute solution of cyclopentadiene ($\mathbf{C_5H_6}$), which suffers reactions together with water molecules and together with the side-products of these reactor. The reaction scheme follows the same Van de Vusse scheme explored throughout this report:

$$\begin{cases} \overbrace{\mathbf{C_5H_6}}^{A} \xrightarrow{K_1(T)} \overbrace{\mathbf{C_5H_7OH}}^{B} \xrightarrow{K_2(T)} \overbrace{\mathbf{C_5H_8(OH)_2}}^{C} \\ \overbrace{2\mathbf{C_5H_6}}^{2A} \xrightarrow{K_3(T)} \overbrace{\mathbf{C_10H_12}}^{D} \end{cases}. \tag{6.1}$$

Since the reaction is non-isothermal, the kinetics rates $[K_1(T), K_2(T), K_3(T)]$ are function of the temperature, and they are assumed to follow the Arrhenius equation (1.5). In order to control the temperature, to yield a better control of the chemical reactions and guarantee some safety constraints to the system, a coolant system is coupled to the tank reactor to perform heat exchange by conduction. This scenario assumes that only the concentration of chemical $\mathbf{C_5H_7OH}$ and the temperature inside the tank can be measured. Additionally, the liquid inflow into the tank carries only the chemical $\mathbf{C_5H_6}$ with a concentration $\rho_{in}^{(A)}$ and temperature T_{in}. The schematic of this operation is shown in Fig. **??**.

[fig]

A nonlinear dynamical model, which was obtained from the first principles approach, is proposed to represent the evolution of each state:

Chapter 7

Results and Discussion

a

Chapter 8

Conclusion

a

Bibliography

[Atiyah, 2018] Atiyah, M. (2018). *Introduction to commutative algebra*. CRC Press.

[Bode, 1945] Bode, H. W. (1945). Network analysis and feedback amplifier design.

[Chen, 1998] Chen, C.-T. (1998). *Linear system theory and design*. Oxford University Press, Inc.

[Heller et al., 1978] Heller, H. C., Crawshaw, L. I., and Hammel, H. T. (1978). The thermostat of vertebrate animals. *Scientific American*, 239(2):102–115.

[Horn and Jackson, 1972] Horn, F. and Jackson, R. (1972). General mass action kinetics. *Archive for rational mechanics and analysis*, 47(2):81–116.

[Horn and Johnson, 2012] Horn, R. A. and Johnson, C. R. (2012). *Matrix analysis*. Cambridge university press.

[Kalman, 1960] Kalman, R. E. (1960). On the general theory of control systems. In *Proceedings First International Conference on Automatic Control, Moscow, USSR*.

[Moler and Loan, 2003] Moler, C. and Loan, C. V. (2003). Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Review*, 45(1):3–49.

[Nyquist, 1932] Nyquist, H. (1932). Regeneration theory. *Bell system technical journal*, 11(1):126–147.

[Strang, 2016] Strang, G. (2016). *Introduction to linear algebra*, volume 5. Wellesley-Cambridge Press.

[Van de Vusse, 1964] Van de Vusse, J. (1964). Plug-flow type reactor versus tank reactor. *Chemical Engineering Science*, 19(12):994–996.

[Vidyasagar, 2002] Vidyasagar, M. (2002). *Nonlinear systems analysis*, volume 42. Siam.

Appendix A - Proof of Theorems

Chapter 2

Proof. **(Theorem 2.2)** Considering that the reactions obeys the Arrhenius equation, the dynamical model for the non-isothermal reactor is a direct application of Theorem 2.1 when assuming that the kinetics are in the form:

$$K_{XY} = K_{XY}(T) = K_{XY} e^{-\frac{E_{XY}}{T}}, \tag{8.1}$$

with T being the temperature in Kelvins and E_{XY} being the activation energy needed for a reaction $X \to Y$ to occur.

In respect to the change in temperatures, the model is obtained by a analysis of the conservation of heat, an energy that obeys this principle:

$$\begin{pmatrix}\text{Accumulation}\\ \text{of heat}\end{pmatrix} = \begin{pmatrix}\text{Heat entering}\\ \text{the System}\end{pmatrix} - \begin{pmatrix}\text{Heat leaving}\\ \text{the System}\end{pmatrix}. \tag{8.2}$$

Consider the temperature inside the reactor. Since the system is closed, and involved by the heating/cooling system, this quantity is a result of a change in heat given by the flow of fluid entering and leaving the system, the direct transfer of heat by conduction from the contact with the cooling/heating system and the entropy contribution from the reactions. In this case, the conservation law becomes:

$$\begin{aligned}\begin{pmatrix}\text{Accumulation}\\ \text{of heat}\end{pmatrix} &= \begin{pmatrix}\text{Heat entering}\\ \text{the System}\end{pmatrix} - \begin{pmatrix}\text{Heat leaving}\\ \text{the System}\end{pmatrix}\\ &= \begin{pmatrix}\text{Heat transfer}\\ \text{from mass-flow}\end{pmatrix} + \begin{pmatrix}\text{Heat transfer}\\ \text{from regulator}\end{pmatrix} + \begin{pmatrix}\text{Entropy}\\ \text{contribution}\\ \text{from reactions}\end{pmatrix}.\end{aligned} \tag{8.3}$$

From Fourier's law **[seek reference]**, the transfer of heat by convection between the fluids, h_{conv} , and by the conduction between the systems, h_{cond}, obeys the proportionality:

$$h_{conv} \sim T_{in} - T_{out} \qquad h_{cond} \sim T_C - T. \tag{8.4}$$

The entropy contribution from a reaction $\alpha X \to \beta Y$, denoted as S_{XY}, is proportional to the concentration of ρ_X consumed multiplied by the energy that it liberates or absorbs:

$$S_{XY} \sim K_{XY} e^{-\frac{E_{XY}}{T}} (\rho_X)^\alpha \Delta H_{XY}. \tag{8.5}$$

All the proportionality can be turned into equalities by imposing real constant factors that are calculated without the dynamical variables. In the case of the heat transfer by convection, the change in temperature is directly given by the ratio of volume entering the system. Plugging all together, and summing the entropy contribution from each reaction, the accumulation of heat is modeled as:

$$\frac{d(T)}{dt} = q(T_{in} - T_{out}) + \eta(T_C - T) + \delta \sum_{\alpha A \to \beta X} K_{AX} e^{-\frac{E_{AX}}{T}} (\rho_A)^\alpha \Delta H_{AX}. \tag{8.6}$$

Consider the heating/cooling system involving the reactor system. From the choice of design of this apparatus, it is possible to directly manipulate the temperature of its material using a cooling/heating capacity Q up to a real factor of γ. Similar to the temperature inside the container, there is the conduction of heat between the healing/cooling system and the walls of the container for the reactor. Therefore, the model of heat accumulation for this quantity is given by:

$$\frac{d(T_C)}{dt} = \gamma Q + \beta(T - T_C). \tag{8.7}$$

□

Proof. **(Theorem 2.3)** Applying the Laplace transform in both sides of the equation, and using some properties of this operator, the result is:

$$\mathcal{L}\left\{\alpha_n \frac{d^n y(t)}{dt^n} + \cdots + \alpha_1 \frac{dy(t)}{dt} + \alpha_0 y(t)\right\} = \mathcal{L}\left\{\beta_m \frac{d^m u(t)}{dt^m} + \cdots + \beta_1 \frac{du(t)}{dt} + \beta_0 u(t)\right\}$$
$$\alpha_0 Y(s) + \sum_{i=1}^{n} \alpha_i \left(s^i Y(s) - \sum_{j=0}^{i-1} s^j \frac{d^j y(0^-)}{dt^j}\right) = \beta_0 U(s) + \sum_{k=1}^{m} \beta_i \left(s^i U(s) - \sum_{l=0}^{k-1} s^l \frac{d^l u(0^-)}{dt^l}\right). \tag{8.8}$$

By the assumption that all the initial conditions are zero, $y(0^-) = u(0^-) = 0$:

$$\alpha_0 Y(s) + \sum_{i=1}^{n} \alpha_i s^i Y(s) = \beta_0 U(s) + \sum_{k=1}^{m} \beta_i s^i U(s). \tag{8.9}$$

Factoring the left side by $\boldsymbol{Y}(s)$ and the right side by $\boldsymbol{U}(s)$, and doing some basic algebra, the transfer function $G(s)$, between these variables is:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\beta_m s^m + \beta_{m-1} s^{m-1} + \cdots + \beta_1 s + \beta_0}{\alpha_n s^n + \alpha_{n-1} s^{n-1} + \cdots + \alpha_1 s + \alpha_0}. \tag{8.10}$$

□