Universidade Federal do Ceará
Departamento de Engenharia de Teleinformática
Curso de Graduação em Engenharia de Computação

OPTIMAL CONTROL: AN APPLICATION TO A NON-ISOTHERMAL CONTINUOUS REACTOR

Otacílio Bezerra Leite Neto

Fortaleza – Ceará
2019

Universidade Federal do Ceará
Departamento de Engenharia de Teleinformática
Curso de Graduação em Engenharia de Computação

OPTIMAL CONTROL: AN APPLICATION TO A NON-ISOTHERMAL CONTINUOUS REACTOR

Autor
Otacílio Bezerra Leite Neto

Orientador
Prof. Dr. Francesco Corona

Co-orientador
Profa. Dra. Michela Mulas

Trabalho de Conclusão de Curso submetido à Coordenação do Programa de Graduação em Engenharia de Computação da Universidade Federal do Ceará como parte dos requisitos para a obtenção do grau de ***Bacharel em Engenharia de Computação****.*

Fortaleza – Ceará
2019

Contents

List of Figures

List of Tables

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 Reactors, therefore the discussion will follow the notation common to the literature of this field, and the modern approach to this theory is explored.

The sections are organized as follows: Section 1.1 provides a general view on control systems engineering, Section 1.2 discuss chemical reactor 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 an 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*, 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:

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

In this representation, the input $u(t)$ is called the *manipulated variable*, since it represents an 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 (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. 1.1.

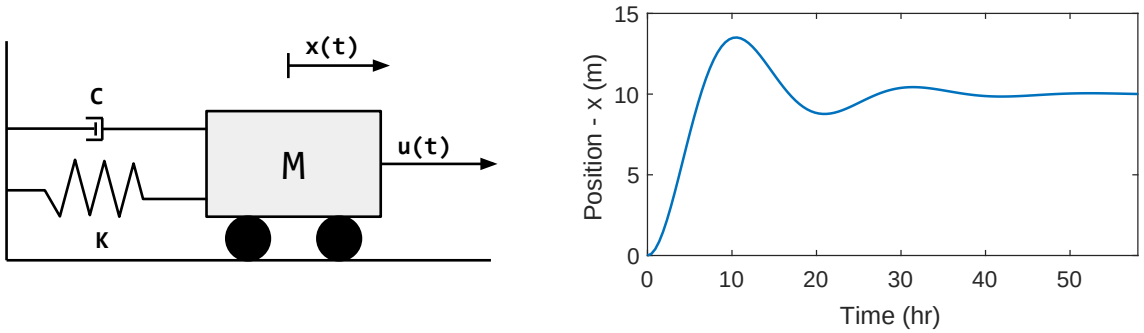


Figure 1.1: The schematic of a simple Mass-Spring-Damper system and its dynamical response to being forced by an unitary constant force.

In this simulation, quantities as the rise time, peak time, overshoot ratio and steady-state value of the visualized signal are examples of response specifications that can be defined to describe the system behavior to an 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. 1.2, of how to connect the controller to the system.

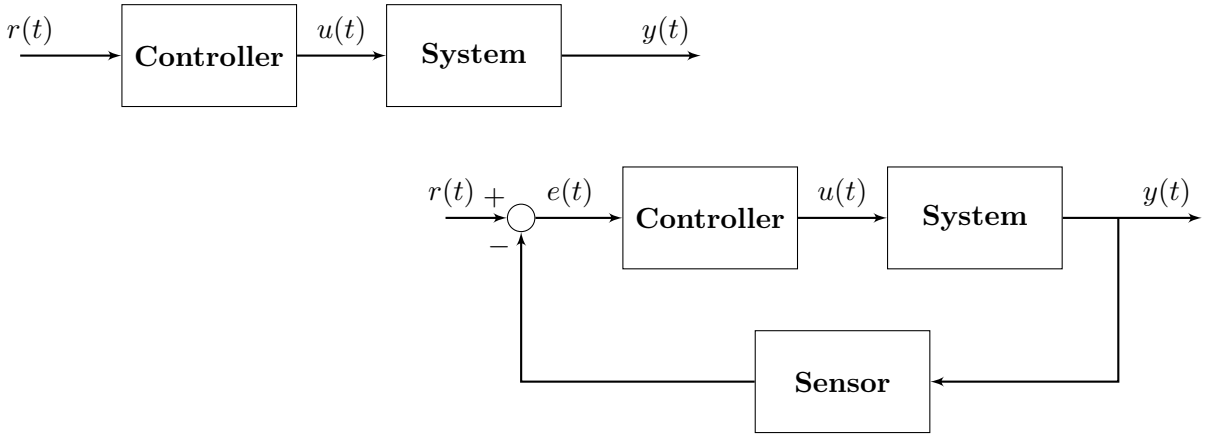


Figure 1.2: The two most popular configurations for connecting controllers to dynamical systems.

The configuration shown in the upper left of Fig. 1.2, known as the *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.

In contrast, the configuration in the lower right of Fig. 1.2, known as the *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. [Syrmos et al., 1997].

1.2 Chemical reactor 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 an 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, $\mathcal{R}$, 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 this reaction.

Usually the products can be directly used as reactants in another reaction, in which case they can also be referred as an 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:

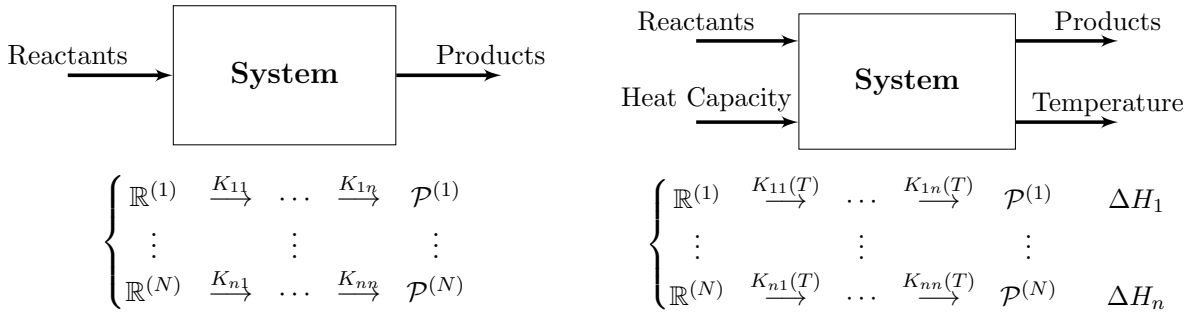
$$\mathcal{R} \xrightarrow{K} \mathcal{P}. \tag{1.4}$$

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 T which are assumed to follow the following form, known as the Arrhenius equation

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

where K_0 is the nominal kinetic rate, E is an activation energy associated with the reaction, R is the universal gas constant and T is the absolute temperature (in kelvins).

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



(a) Isothermal chemical reactor

(b) Non-isothermal chemical reactor

Figure 1.3: General graphical representations of common chemical reactor systems.

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. Fig. 1.3a, is a system where the concentration of some reactants is manipulated to produce a desired concentration of some products, given a chemical reaction network describing the reaction between these entities.

When the process is non-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. 1.3b. Moreover, note that the equation associated with non-isothermal reactions suffers a change of notation to include the value of entropy change, denoted ΔH, resulting in the following general equation known as the *reaction heat equation*:

$$\mathcal{R} \overset{K(T)}{\longrightarrow} \mathcal{P} \qquad \Delta H. \tag{1.6}$$

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 [Eren et al., 2017, Goerzen et al., 2009, Wang and Su, 2015], autonomous driving [González et al., 2016, Falcone et al., 2007, Cairano et al., 2014] and several industrial applications [Qin and Badgwell, 2003, Tang et al., 2016, Holenda et al., 2008, Lucia et al., 2013]. In parallel, this theory is also useful to understand and bring inspiration from nature itself since, for instance, feedback controller characteristics were observed in the mechanisms for temperature regulation observed in vertebrate animals [Heller et al., 1978] and in [Todorov and Jordan, 2002].

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* theory, particularly in *Reinforcement Learning*, which have been successful in applying optimization techniques for controlling artificial agents in environments loaded with uncertainty [Sutton and Barto, 2018, Mnih et al., 2015].

Furthermore, the specific application of controlling chemical reactor systems brings benefits from the fact that this class of dynamical system are present in many important biological and industrial processes, such as wastewater treatment [Mulas et al., 2015, Gupta et al., 2012]. In general, controllers can be used to guarantee safety constraints, maximize productivity and minimize the use of resources, in such a way that could not be achieved without automatic and high performing machinery. The inherent complexity and uncertainty existent in these systems also poses this control problem as a challenging research field still open to be explored.

1.4 Objectives

This works aims to provide a self-contained discussion of modeling and control of chemical rector systems in the perspective of modern control, with focus on optimal control theory. Therefore, the results are focused on *state-feedback controllers* modeled in continuous time. Several properties of the dynamical models of such systems, both in the open-loop and feedback regime, are summarized in the document and the intention is to have a generalized mathematical framework to understand, evaluate and design those control systems. Finally, a specific problem concerning the control of a non-isothermal continuous reactor is explored to demonstrate an application of the theory developed throughout the document.

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 modeling and control of the discussed systems. 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 concept of dynamical models and their several properties with respect to a real system behavior. Chapter 3 discusses general results of state-feedback controllers and state observers. Chapter 4 presents the theory of optimal estimation and optimal control, in a broad perspective. Chapter 5 describes practical experiments in respect to a real-world continuous reactor system. Chapter 6 summarizes and discusses the results of these experiments, emphasizing the theory from the previous sections. Finally, chapter 7 concludes the document and highlight the most important results.

Chapter 2

Dynamical System Analysis

This chapter discusses the mathematical models for dynamical systems. The sections starts by introducing a procedure to build models from physical principles and presenting equivalent common representations. Next, the time evolution of dynamical systems is analyzed in relation to the mathematical structure of such models. Finally, some important properties of the systems are defined and discussed. A discussion of the system evolution in the frequency domain is also presented.

2.1 First Principles Models

A dynamical system is a physical system whose state evolves with time. One can represent a dynamical system using the *first principles* from physics, and formulate the evolution in time by calculating the rate of change of its state in respect to time. Thus, dynamical models are represented 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 models are in the form:

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

The choice of which conservation law to use depends on the system itself. Usually, conservation of mass is used to relate dynamics of concentrations and volumes, or other material variables, while conservation of momentum is 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:

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

Example 2.1. (*Mass Balance of Reactors*) For the sake of illustration, consider the process of obtaining a first principles model for the concentration of a general substance in a reactor system, as follows. Consider a system such as the one depicted in 1.3a. First of all, one has to evaluate the mass flow through the system, as denoted in (2.2). The mass of any substance A entering and leaving the system, denoted respectively by M_{in} and M_{out}, are assumed to be provided by a flow of fluid carrying the substance. Given a fluid inflow F_{in}, with density $\rho_{in}^{(A)}$, and an outflow F_{out}, with density $\rho_{out}^{(A)}$, the mass flow is calculated by equations:

$$M_{in} = \rho_{in}^{(A)} F_{in} \quad ; \quad M_{out} = \rho_{out}^{(A)} F_{out}. \tag{2.3}$$

Now, one has to consider the quantities of mass that are consumed and produced by the reactions of the chemical reaction network associated to the system. In this case, consider the following reaction between two chemical compounds X and Y, with stoichiometric numbers α and β:

$$\alpha X \xrightarrow{K_{XY}} \beta Y. \tag{2.4}$$

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} 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:

$$M_{\text{cons}}^{(X)} = \frac{V}{\beta} K_{XY} (\rho_X)^{\alpha} \quad ; \quad M_{\text{prod}}^{(Y)} = \frac{V}{\alpha} K_{XY} (\rho_X)^{\alpha}, \tag{2.5}$$

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:

$$M_{\text{cons}} = V \sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} (\rho_A)^{\alpha} \quad ; \quad M_{\text{prod}} = V \sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} (\rho_X)^{\alpha}. \tag{2.6}$$

Finally, directly substituting these values in (2.2), the mass balance of any substance A in an isothermal chemical reactor system can be represented by the general dynamical model:

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

$$\frac{d(\rho_A V)}{dt} = \left[\rho_{in}^{(A)} F_{in} + V \sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} (\rho_X)^{\alpha} \right] - \left[\rho_{out}^{(A)} F_{out} + V \sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} (\rho_A)^{\alpha} \right] \tag{2.7}$$

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:

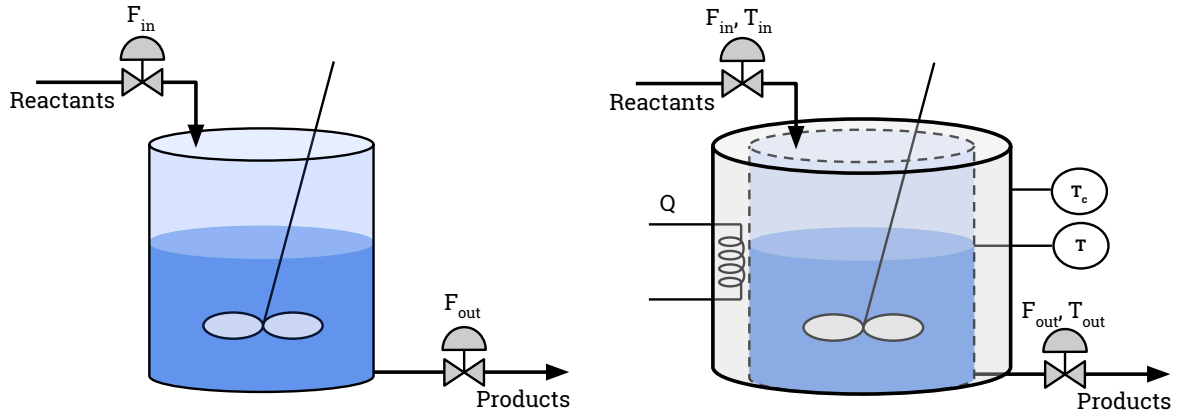
$$\frac{d(\rho_A)}{dt} = q(\rho_{in}^{(A)} - \rho_{out}^{(A)}) + \left(\sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} (\rho_X)^{\alpha} \right) - \left(\sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} (\rho_A)^{\alpha} \right). \tag{2.8}$$

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:

$$\begin{cases} \dfrac{d(\rho_{X_1})}{dt} &= f(\rho_{X_1}, \rho_{X_2}, ..., \rho_{X_n}, \rho_{in}^{(X_1)}, \rho_{out}^{(X_1)}, t) \\ \dfrac{d(\rho_{X_2})}{dt} &= f(\rho_{X_1}, \rho_{X_2}, ..., \rho_{X_n}, \rho_{in}^{(X_2)}, \rho_{out}^{(X_2)}, t) \\ &\vdots \\ \dfrac{d(\rho_{X_n})}{dt} &= f(\rho_{X_1}, \rho_{X_2}, ..., \rho_{X_n}, \rho_{in}^{(X_n)}, \rho_{out}^{(X_n)}, t) \end{cases}, \tag{2.9}$$

where $X_1, X_2, ..., X_n$ are the chemical compounds inside the reactor and $f(\cdot)$ is the dynamical model presented in Example 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 fairly simple. To account for more complex systems, the same modeling procedure can still 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.

Example 2.2. (Non-isothermal Reactor System) Consider the process of obtaining a first principle model for a non-isothermal reactor, such as the one depicted in 1.3b. In this case, the kinetic rates will follow the Arrhenius equation from (1.5). Following the same procedure from Example 2.1, a dynamical model for the concentrations of a substance A, inside this reactor, can be formulated as:

$$\frac{d(\rho_A)}{dt} = q(\rho_{in}^{(A)} - \rho_{out}^{(A)}) + \left(\sum_{\alpha X \to \beta A} \frac{1}{\beta} K_{XA} e^{-\frac{E_{XA}/R}{T}} (\rho_X)^\alpha \right) - \left(\sum_{\alpha A \to \beta X} \frac{1}{\beta} K_{AX} e^{-\frac{E_{AX}/R}{T}} (\rho_A)^\alpha \right), \tag{2.10}$$

where E_{XA} and E_{AX} are the activation energy needed for each reaction, and the rest of the parameters are still the same as defined in Example 2.1.

Furthermore, since the process is non-isothermal, the reactions will cause a change of temperature inside the reactor system. Consider the temperature inside the reactor, denoted as T. In this case, it is possible to relate its dynamics to the conservation law:

$$\begin{aligned} \begin{pmatrix} \text{Accumulation} \\ \text{of thermal energy} \\ \text{in the system} \end{pmatrix} &= \begin{pmatrix} \text{Thermal energy} \\ \text{entering} \\ \text{the system} \end{pmatrix} - \begin{pmatrix} \text{Thermal energy} \\ \text{leaving} \\ \text{the system} \end{pmatrix} \\ &= \begin{pmatrix} \text{Heat flow} \\ \text{entering} \\ \text{the system} \end{pmatrix} - \begin{pmatrix} \text{Heat flow} \\ \text{leaving} \\ \text{the system} \end{pmatrix} + \begin{pmatrix} \text{Entropy} \\ \text{contribution} \\ \text{from reactions} \end{pmatrix}. \end{aligned} \tag{2.11}$$

It is well known, from Fourier's law [Bergman et al., 2017], that the transfer of heat by conduction between the reactor and the coolant/heater, in their respective contact interfaces, is proportional to $(T_C - T)$, where T_C represents the temperature of the latter. Moreover, another identifiable source of heat transfer consists in the fluid flow entering and leaving the system. Similarly for the conduction, this heat contribution should also be proportional to $(T_{in} - T_{out})$, where T_{in} and T_{out} are respectively the temperature of the fluid entering and leaving the reactor.

The entropy contribution from a reaction $\alpha X \to \beta Y$, denoted as S_{XY}, is proportional to the concentration of ρ_X consumed, or produced, by the reactions multiplied by the energy that it liberates or absorbs, as stated by the Hess' Law [Atkins and de Paula, 2011]:

$$S_{XY} \sim K_{XY} e^{-\frac{E_{XY}}{T}} (\rho_X)^\alpha \Delta H_{XY}. \tag{2.12}$$

All the proportionality can be turned into equalities by imposing real constant factors that are calculated independent of the dynamical variables. In the case of the heat transfer from the fluid, this constant factor is the flow-rate q itself. Plugging all together, and summing the entropy contribution from each reaction, the accumulation of heat can be modeled:

$$\frac{d(T)}{dt} = q(T_{in} - T_{out}) + \eta(T_C - T) + \delta \sum_{\alpha A \to \beta X} K_{AX} e^{-\frac{E_{AX}}{T}} (\rho_A)^\alpha \Delta H_{AX}. \tag{2.13}$$

Consider now the heating/cooling system involving the reactor system. From the choice of design of this apparatus, it is possible to manipulate the temperature of its material using a heat 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 simply given by:

$$\frac{d(T_C)}{dt} = \gamma Q + \beta(T - T_C). \tag{2.14}$$

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 (2.9), the dynamical model of the entire reactor is a system of differential equations relating all those quantities.

2.2 Canonical Representations for Dynamical Models

The last section presented the foundation for modeling a dynamical system using first principles from physics. Although it is a well-defined formulation, the resulting models are not guaranteed to be practical in a mathematical sense. This motivates a discussion about possible canonical formats for such models, which would allow for systematic analysis approaches. 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.

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 $p, r > 1$. A formal definition is given below.

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:

$$\begin{cases} h_1\left(y_1, \dot{y_1}, ..., y_1^{(n_1)}, u_1, \dot{u}_1, ..., u_1^{(m_{11})}, u_2, \dot{u}_2, ..., u_2^{(m_{12})}, ..., u_r, \dot{u}_r, ..., u_r^{(m_{1r})}, t\right) = 0 \\ h_2\left(y_2, \dot{y_2}, ..., y_2^{(n_2)}, u_1, \dot{u}_1, ..., u_1^{(m_{21})}, u_2, \dot{u}_2, ..., u_2^{(m_{22})}, ..., u_r, \dot{u}_r, ..., u_r^{(m_{2r})}, t\right) = 0 \\ \vdots \\ h_p\left(y_p, \dot{y_p}, ..., y_p^{(n_p)}, u_1, \dot{u}_1, ..., u_1^{(m_{p1})}, u_2, \dot{u}_2, ..., u_2^{(m_{p2})}, ..., u_r, \dot{u}_r, ..., u_r^{(m_{pr})}, t\right) = 0 \end{cases}, \quad (2.15)$$

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

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

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 is 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.

The State-Space representation is another formulation for a dynamical model, 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 $\boldsymbol{x}(t_0)$, at time $t_0 \in \mathbb{R}$, and a forcing function $\boldsymbol{u}(t)$, for any time $t \geq t_0$. From a physical perspective, however, these variables may account for quantities that can describe the dynamics of the system, such as position or velocities. In comparison to the Input-Output representation, this formulation poses a simpler mathematical model, since it is composed by a system of first order 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 formal definition is given below.

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: **Output Equations:**

$$\begin{cases} \dot{x}_1(t) = f_1(x_1, ..., x_n, u_1, ..., u_r, t) \\ \dot{x}_2(t) = f_2(x_1, ..., x_n, u_1, ..., u_r, t) \\ \vdots \\ \dot{x}_n(t) = f_n(x_1, ..., x_n, u_1, ..., u_r, t) \end{cases} ; \begin{cases} y_1(t) = g_1(x_1, ..., x_n, u_1, ..., u_r, t) \\ y_2(t) = g_2(x_1, ..., x_n, u_1, ..., u_r, t) \\ \vdots \\ y_p(t) = g_p(x_1, ..., x_n, u_1, ..., u_r, t) \end{cases} , \tag{2.16}$$

or, in the matrix form:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \\ \boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \end{cases} . \tag{2.17}$$

(a) (b)

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

A graphical illustration of both representations is shown at Fig. 2.2. In the light of these formulations, a system can be classified with respect to the model mathematical structure. There are five main properties used for this classification: whether 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

	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 time arguments	All the signals share the same time arguments

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

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 (2.27) reduces to simple matrix forms.

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:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{B}(t)\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}(t)\boldsymbol{x}(t) + \boldsymbol{D}(t)\boldsymbol{u}(t) \end{cases}, \tag{2.18}$$

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 advantage 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.

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.

Definition 2.4. (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:

$$\alpha_n \frac{d^n y(t)}{dt^n} + \cdots + \alpha_1 \frac{dy(t)}{dt} + \alpha_0 y(t) = \beta_m \frac{d^m u(t)}{dt^m} + \cdots + \beta_1 \frac{du(t)}{dt} + \beta_0 u(t), \tag{2.19}$$

its correspondent transfer function, calculated in the Laplace domain, is defined as:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{\beta_m s^m + \beta_{m-1} s^{m-1} + \cdots + \beta_1 s + \beta_0}{\alpha_n s^n + \alpha_{n-1} s^{n-1} + \cdots + \alpha_1 s + \alpha_0}. \tag{2.20}$$

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 $\boldsymbol{G} \in \mathbb{C}^{p \times r}$, for a system with p outputs and r inputs. For the general linear case, this matrix has the form:

$$\boldsymbol{G}(s) = \begin{bmatrix} \frac{Y_1(s)}{U_1(s)} & \frac{Y_1(s)}{U_2(s)} & \cdots & \frac{Y_1(s)}{U_r(s)} \\ \frac{Y_2(s)}{U_1(s)} & \frac{Y_2(s)}{U_2(s)} & \cdots & \frac{Y_2(s)}{U_r(s)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{Y_p(s)}{U_1(s)} & \frac{Y_p(s)}{U_2(s)} & \cdots & \frac{Y_p(s)}{U_r(s)} \end{bmatrix} = \begin{bmatrix} \frac{\beta_{m_{11}}^{(11)} s^{m_{11}} + \cdots + \beta_0^{(11)}}{\alpha_{n_1}^{(1)} s^{n_1} + \cdots + \alpha_0^{(1)}} & \cdots & \frac{\beta_{m_1}^{(1r)} s^{m_{1r}} + \cdots + \beta_0^{(1r)}}{\alpha_{n_1}^{(1)} s^{n_1} + \cdots + \alpha_0^{(1)}} \\ \frac{\beta_{m_{21}}^{(21)} s^{m_{21}} + \cdots + \beta_0^{(21)}}{\alpha_{n_2}^{(2)} s^{n_2} + \cdots + \alpha_0^{(2)}} & \cdots & \frac{\beta_{m_2}^{(2r)} s^{m_{2r}} + \cdots + \beta_0^{(2r)}}{\alpha_{n_2}^{(2)} s^{n_2} + \cdots + \alpha_0^{(2)}} \\ \vdots & \ddots & \vdots \\ \frac{\beta_{m_{p1}}^{(p1)} s^{m_{p1}} + \cdots + \beta_0^{(p1)}}{\alpha_{n_p}^{(p)} s^{n_p} + \cdots + \alpha_0^{(p)}} & \cdots & \frac{\beta_{m_p}^{(pr)} s^{m_{pr}} + \cdots + \beta_0^{(pr)}}{\alpha_{n_p}^{(p)} s^{n_p} + \cdots + \alpha_0^{(p)}} \end{bmatrix}. \tag{2.21}$$

Theorem 2.1. *(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:*

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases}. \tag{2.22}$$

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

$$\boldsymbol{G}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}. \tag{2.23}$$

Proof. Applying the Laplace formation to both sides of the equation, using the property of derivative in time, results in:

$$\begin{aligned} \mathcal{L}\{\dot{\boldsymbol{x}}(t)\} &= \boldsymbol{A}\mathcal{L}\{\boldsymbol{x}(t)\} + \boldsymbol{B}\mathcal{L}\{\boldsymbol{u}(t)\} \\ s\boldsymbol{X}(s) - \dot{\boldsymbol{x}}(0^-) &= \boldsymbol{A}\boldsymbol{X}(s) + \boldsymbol{B}\boldsymbol{U}(s) \end{aligned}. \tag{2.24}$$

Substituting the derivative $\dot{\boldsymbol{x}}(0^-) = 0$ and doing some manipulations results in:

$$\begin{aligned} (s\boldsymbol{I} - \boldsymbol{A})\boldsymbol{X}(s) &= \boldsymbol{B}\boldsymbol{U}(s) \\ \boldsymbol{X}(s) &= (s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}\boldsymbol{U}(s) \end{aligned}. \tag{2.25}$$

Finally, applying the same procedure to the output equations, and using the previous result, the Laplace transform of the output is:

$$\begin{aligned} \mathcal{L}\{\boldsymbol{y}(t)\} &= \boldsymbol{C}\mathcal{L}\{\boldsymbol{y}(t)\} + \boldsymbol{D}\mathcal{L}\{u(t)\} \\ \boldsymbol{Y}(s) &= \boldsymbol{C}\boldsymbol{X}(s) + \boldsymbol{D}\boldsymbol{U}(s) \\ &= \boldsymbol{C}\left((s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}\boldsymbol{U}(s)\right) + \boldsymbol{D}\boldsymbol{U}(s) \\ &= \left(\boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}\right)\boldsymbol{U}(s) \end{aligned}. \tag{2.26}$$

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

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \end{cases} . \tag{2.27}$$

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:

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\Delta\boldsymbol{x}(t) + \boldsymbol{B}\Delta\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\Delta\boldsymbol{x}(t) + \boldsymbol{D}\Delta\boldsymbol{u}(t) \end{cases} , \tag{2.28}$$

where

$$\boldsymbol{A} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} ; \quad \boldsymbol{B} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} ; \quad \boldsymbol{C} = \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} ; \quad \boldsymbol{D} = \left.\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \tag{2.29}$$

and

$$\Delta\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_o; \qquad \Delta\boldsymbol{u}(t) = \boldsymbol{u}(t) - \boldsymbol{u}_o. \tag{2.30}$$

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 $\Delta\boldsymbol{u}(t)$ in the input signal around these operation points. This perturbation will result in small changes in the state and output variables:

$$\boldsymbol{x}(t) = \boldsymbol{x}_o + \Delta\boldsymbol{x}(t); \qquad \boldsymbol{u}(t) = \boldsymbol{u}_o + \Delta\boldsymbol{u}(t); \qquad \boldsymbol{y}(t) = \boldsymbol{y}_o + \Delta\boldsymbol{y}(t). \tag{2.31}$$

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

$$\begin{cases} \dfrac{d(\boldsymbol{x}_o + \Delta\boldsymbol{x}(t))}{dt} & = \boldsymbol{f}(\boldsymbol{x}_o + \Delta\boldsymbol{x}(t), \boldsymbol{u}_o + \Delta\boldsymbol{u}(t)) \\ \boldsymbol{y}_o + \Delta\boldsymbol{y}(t) & = \boldsymbol{g}(\boldsymbol{x}_o + \Delta\boldsymbol{x}(t), \boldsymbol{u}_o + \Delta\boldsymbol{u}(t)) \end{cases} , \tag{2.32}$$

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 steady-state points, hence the functions $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ can be approximated by a Taylor series expansion, yielding:

$$\begin{cases} \dfrac{d(\Delta\boldsymbol{x}(t))}{dt} & = \boldsymbol{f}(\boldsymbol{x}_o, \boldsymbol{u}_o) + \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) + \mathcal{O}(\Delta\boldsymbol{x}^2, \Delta\boldsymbol{u}^2) \\ \boldsymbol{y}_o + \Delta\boldsymbol{y}(t) & = \boldsymbol{g}(\boldsymbol{x}_o, \boldsymbol{u}_o) + \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) + \mathcal{O}(\Delta\boldsymbol{x}^2, \Delta\boldsymbol{u}^2) \end{cases} . \tag{2.33}$$

Since the steady-state condition implies zero variation, it is possible to assume $\boldsymbol{f}(\boldsymbol{x}_o, \boldsymbol{u}_o) = \boldsymbol{0}$ and $g(\boldsymbol{x}_o, \boldsymbol{u}_o) = \boldsymbol{0}$, 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:

$$\begin{cases} \dfrac{d(\Delta\boldsymbol{x}(t))}{dt} = \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) \\ \\ \boldsymbol{y}(t) = \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{x}(t) + \left.\dfrac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}_o,\boldsymbol{u}_o} \Delta\boldsymbol{u}(t) \end{cases}. \tag{2.34}$$

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:

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

□

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.

Example 2.3. (Van de Vusse Reactor) Consider, for instance, a reactor system describing an isothermal process that follows the Van de Vusse reaction scheme [Van de Vusse, 1964]:

$$\begin{cases} A \xrightarrow{K_{AB}} B \xrightarrow{K_{BC}} C \\ 2A \xrightarrow{K_{AD}} D \end{cases}. \tag{2.36}$$

Consider, also, that the concentration of substances A, B, C and D inside the fluid inflow and outflow are described by the following conditions:

$$\begin{cases} \rho_{in}^{(B)}(t) = \rho_{in}^{(C)}(t) = \rho_{in}^{(D)}(t) = 0 \\ \rho_{out}^{(k)}(t) = \rho_k(t), \ \forall k \in \{A, B, C, D\} \end{cases}. \tag{2.37}$$

Now, let $\boldsymbol{x} = [\rho_A, \rho_B, \rho_C, \rho_D]^T$ and $u = q$, and assume that the states are perfectly observed, so that $\boldsymbol{y} = \boldsymbol{x}$. Using Example 2.1, the nonlinear State-Space representation of this reactor is can be modeled by the state-equations:

$$\begin{cases} \dot{x_1}(t) = u(t)(\rho_{in}^{(A)} - x_1(t)) - (K_{AB}x_1(t) + K_{AD}(x_1(t))^2) \\ \dot{x_2}(t) = -u(t)x_2(t) + K_{AB}x_1(t) - K_{BC}x_2(t) \\ \dot{x_3}(t) = -u(t)x_3(t) + K_{BC}x_2(t) \\ \dot{x_4}(t) = -u(t)x_4(t) + \dfrac{1}{2}K_{AD}(x_1(t))^2 \end{cases}. \tag{2.38}$$

To select steady-state points $\boldsymbol{x}_o$ and 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 analytically, in some cases, or by using numerical methods. Finally, the linear State-Space matrices $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$ are obtained by taking the

derivative each one of these equations in respect to each state and input, resulting in:

$$\begin{cases} \boldsymbol{A} = \begin{bmatrix} -u_o - K_{AB} - 2K_{AC}x_{1o} & 0 & 0 & 0 \\ K_{AB} & -u_o - K_{BC} & 0 & 0 \\ 0 & K_{BC} & -u_o & 0 \\ K_{AD}x_{1o} & 0 & 0 & -u_o \end{bmatrix}; \quad \boldsymbol{B} = \begin{bmatrix} \rho_{in}^{(A)} - x_{1o} \\ -x_{2o} \\ -x_{3o} \\ -x_{4o} \end{bmatrix}; \\ \boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad \boldsymbol{D} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \end{cases}. \tag{2.39}$$

Notice that the response of a linearized model represents the deviations around the given steady-state points. If the errors in approximating the non-linearities of the actual system are somehow tolerable, however, this model can still be used as a nominal model to represent the system in any point in space. For the sake of illustration, the simulation of both the nonlinear and the linear systems are displayed in Fig. 2.3, for a realization of the system accounted in Example 2.3. In this realization, the kinetic rates were set as $K_{AB} = 5/6$, $K_{BC} = 5/3$ and $K_{AC} = 1/6$, and the inflow concentration was consider constant as $\rho_{in}^{(A)} = 10\ (mol/l)$. In this simulation, the dashed line represents the linear response of the linearized model given $\boldsymbol{x}_o = [6.19, 1.09, 0.60, 1.05]^T$ and $u_0 = 3.03$. Notice how the linearized system is capable of approximating the behavior of the non-linear system when closer to the steady-state point, but starts to display an error when the system leaves that state.

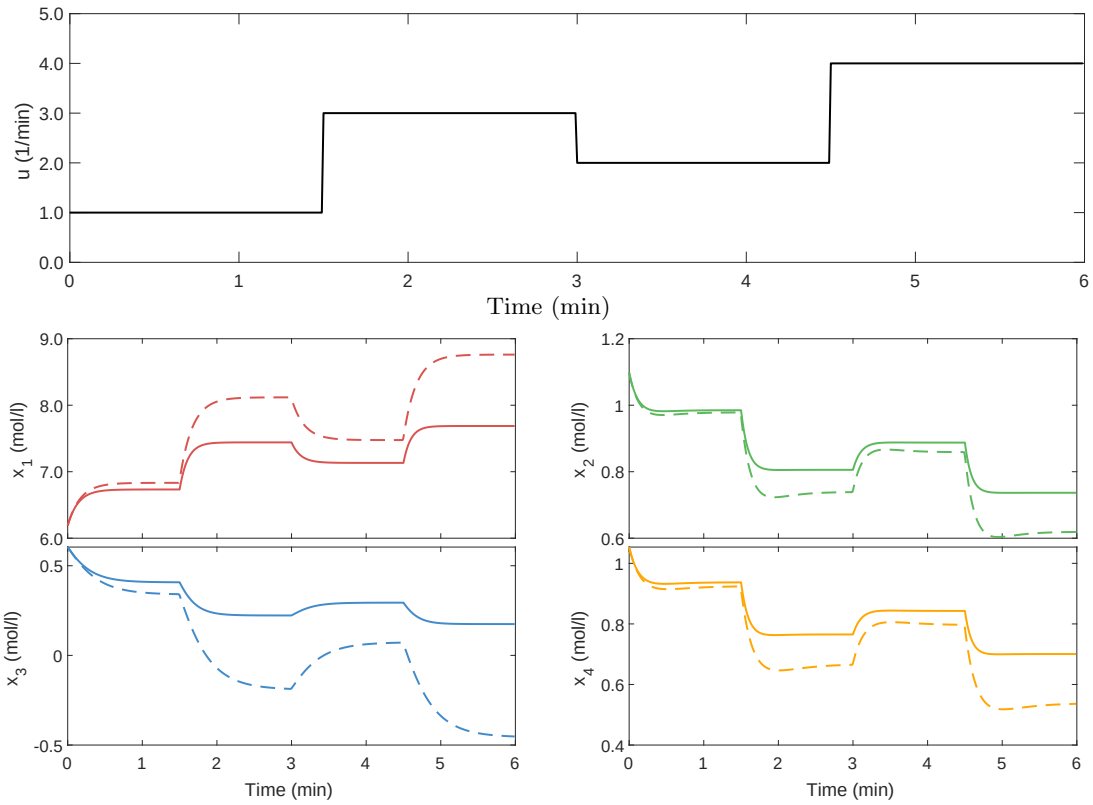


Figure 2.3: Comparison between the nonlinear and linear response of the reactor system.

2.3 Response Analysis in the Time Domain

Once that a model is well-established, it is possible both to simulate the system and to analyze its response, given an initial state and input signal. This section 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 systems in the form:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases}. \tag{2.40}$$

First of all, it is necessary to access some properties of the matrix $\boldsymbol{A}$, which describes the influence of the states in the evolution of the system. Consider, in this case, the following matrix defined by computing the exponential of the matrix $\boldsymbol{A}$.

Definition 2.5. (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:

$$e^{\boldsymbol{A}t} = \boldsymbol{I} + \boldsymbol{A}\boldsymbol{t} + \frac{\boldsymbol{A}^2t^2}{2!} + \frac{\boldsymbol{A}^3t^3}{3!} + \cdots = \sum_{k=0}^{\infty} \frac{A^kt^k}{k!}. \tag{2.41}$$

This matrix is central to the computation of the system time response, since it can directly relate the initial state of a system to any other state forward in time. Moreover, the exponential of a matrix is not a trivial operation. However, the fact that the State-Space matrix $\boldsymbol{A}$ is always a square matrix results in the following properties for the State-Transition matrix, stated below.

Theorem 2.3. *Consider a matrix exponential as in Definition 2.5, for a matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$. Then, the following properties holds:*

$$\text{I) } \frac{d(e^{\boldsymbol{A}t})}{dt} = \boldsymbol{A}e^{\boldsymbol{A}t}; \qquad \text{II) } e^{\boldsymbol{A}t}e^{\boldsymbol{A}\tau} = e^{\boldsymbol{A}(t+\tau)}; \qquad \text{III) } e^{-\boldsymbol{A}t}e^{\boldsymbol{A}t} = e^{\boldsymbol{A}t}e^{-\boldsymbol{A}t}; = \boldsymbol{I}. \tag{2.42}$$

The proof of these properties can be obtained directly from using the Definition 2.5 and doing some algebra. Given these results, the calculation of the time response of a system in SS representation becomes straightforward.

Theorem 2.4. *(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:*

$$\begin{cases} \boldsymbol{x}(t) = e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t) + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \\ \boldsymbol{y}(t) = \boldsymbol{C}e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t) + \boldsymbol{C}\int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau + \boldsymbol{D}\boldsymbol{u}(t) \end{cases}. \tag{2.43}$$

Proof. First of all, consider a system in State-Space representation with state equation as defined in (2.27). Multiplying both sides by $e^{-\boldsymbol{A}t}$:

$$\begin{aligned} e^{-\boldsymbol{A}t}\dot{\boldsymbol{x}}(t) &= e^{-\boldsymbol{A}t}(\boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)) \\ e^{-\boldsymbol{A}t}\dot{\boldsymbol{x}}(t) - \boldsymbol{A}e^{-\boldsymbol{A}t}\boldsymbol{x}(t) &= e^{-\boldsymbol{A}t}\boldsymbol{B}\boldsymbol{u}(t) \end{aligned}. \tag{2.44}$$

Using the multiplication rule and the first property from Theorem 2.3, it is easy to see that $d[e^{-\boldsymbol{A}t}\boldsymbol{x}(t)]/dt = e^{-\boldsymbol{A}t}\dot{\boldsymbol{x}}(t) - \boldsymbol{A}e^{-\boldsymbol{A}t}\boldsymbol{x}(t)$. Substituting this result in (2.44) and integrating both sides from t_0 to t:

$$\begin{aligned} \frac{d(e^{-\boldsymbol{A}t}\boldsymbol{x}(t))}{dt} &= e^{-\boldsymbol{A}t}\boldsymbol{B}\boldsymbol{u}(t) \\ e^{-\boldsymbol{A}t}\boldsymbol{x}(t)\Big|_{t_0}^{t} &= \int_{t_0}^{t} e^{-\boldsymbol{A}\tau}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau. \\ e^{-\boldsymbol{A}t}\boldsymbol{x}(t) - e^{-\boldsymbol{A}t_0}\boldsymbol{x}(t_0) &= \int_{t_0}^{t} e^{-\boldsymbol{A}\tau}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \end{aligned} \tag{2.45}$$

Multiplying both sides of (2.45) by $e^{\boldsymbol{A}t}$ and using the second and third property from Theorem 2.3, the state response can be calculated as:

$$\begin{aligned} e^{\boldsymbol{A}t}\left(e^{-\boldsymbol{A}t}\boldsymbol{x}(t) - e^{-\boldsymbol{A}t_0}\boldsymbol{x}(t_0)\right) &= e^{\boldsymbol{A}t}\int_{t_0}^{t} e^{-\boldsymbol{A}\tau}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \\ \boldsymbol{I}\boldsymbol{x}(t) - e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t_0) &= \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \ . \\ \boldsymbol{x}(t) &= e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau \end{aligned} \tag{2.46}$$

Finally, substituting (2.46) into the output equation leads to:

$$\begin{aligned} \boldsymbol{y}(t) &= \boldsymbol{C}\left(e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau\right) + \boldsymbol{D}\boldsymbol{u}(t) \\ &= \boldsymbol{C}e^{\boldsymbol{A}(t-t_0)}\boldsymbol{x}(t) + \boldsymbol{C}\int_{t_0}^{t} e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau + \boldsymbol{D}\boldsymbol{u}(t) \end{aligned} \ . \tag{2.47}$$

□

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

$$\boldsymbol{x}(t) = \boldsymbol{x}_{\mathrm{n}}(t) + \boldsymbol{x}_{\mathrm{f}}(t), \tag{2.48}$$

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 (2.39), 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.

In order to characterize the dynamical response of a system, there is still the need to compute the state-transition matrix. This 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 understanding of the system behavior through the eigenvalues of the state-state matrix $\boldsymbol{A}$.

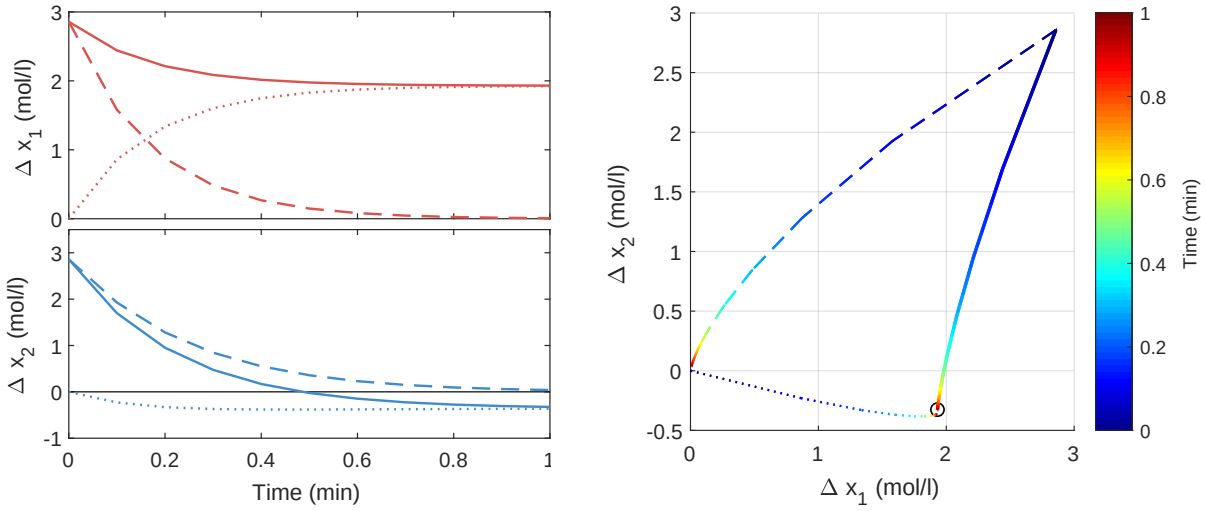


Figure 2.4: Simulation of the total response decomposition for the two first states of the model from (2.39). Each state response is shown separately (left) and together into a trajectory visualization (right), in which the steady-state point is indicated by a circle.

Theorem 2.5. *(Sylvester expansion) Consider a matrix exponential function $e^{\boldsymbol{A}t}$ 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 $\|\boldsymbol{\nu}\|_1 = n$. The result of this function can be expanded as*

$$e^{\boldsymbol{A}t} = \beta_0(t)\boldsymbol{I} + \beta_1(t)\boldsymbol{A} + \beta_2(t)\boldsymbol{A}^2 + ... + \beta_{n-1}(t)\boldsymbol{A}^{n-1} = \sum_{i=0}^{n-1} \beta_i(t)\boldsymbol{A}^i, \tag{2.49}$$

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

$$\boldsymbol{V}\boldsymbol{\beta} = \boldsymbol{\eta}, \tag{2.50}$$

for the parameter vector $\boldsymbol{\beta} = [\beta_0(t), \beta_1(t), ..., \beta_{n-1}(t)]^T$, given vector of modes $\boldsymbol{\eta} = [\eta_1, \eta_2, ..., \eta_m]^T$ such that

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

and the confluent Vandermonde matrix $\boldsymbol{V} = [V_1, V_2, ..., V_m]^T$ such that

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

The proof of this expansion is somewhat extensive, but a detailed discussion can be found in [Chen, 1998]. Basically, the expansion in (2.49) is a direct application of the Cayley-Hamilton theorem [Atiyah, 2018] and the linear system in (2.50) is a result of the Sylvester's matrix theorem [Horn and Johnson, 2012].

From the expansion presented in Theorem 2.5, 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.

Example 2.4. Consider, for the sake of illustration, the same State-Space model as the one simulated in Fig. 2.3, but considering only the first two states (since they are independent of the others). Its matrix $\boldsymbol{A}$ is shown below:

$$\boldsymbol{A} = \begin{bmatrix} -5.93 & 0 \\ 0.83 & -4.70 \end{bmatrix}. \tag{2.51}$$

It is easy to see that $\boldsymbol{\lambda} = [-5.93, -4.70]$, since $\boldsymbol{A}$ is diagonal. The Sylvester expansion for the exponential of this matrix has the parameter vector $\boldsymbol{\beta} = [\beta_0(t), \beta_1(t)]$ which solves the system

$$\begin{bmatrix} 1 & -5.93 \\ 1 & -4.70 \end{bmatrix} \begin{bmatrix} \beta_0(t) \\ \beta_1(t) \end{bmatrix} = \begin{bmatrix} e^{-5.93t} \\ e^{-4.70t} \end{bmatrix}. \tag{2.52}$$

The solution to this system is $\boldsymbol{\beta} = [4.82e^{-4.7t} - 3.82e^{-5.93t},\ 0.813e^{-4.7t} - 0.813e^{-5.93t}]$. Hence, state-transition matrix is calculated through the expansion:

$$\begin{aligned} e^{\boldsymbol{A}t} &= \beta_0(t)\boldsymbol{I} + \beta_1(t)\boldsymbol{A} \\ &= 4.82e^{-4.7t} - 3.82e^{-5.93t} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + 0.813e^{-4.7t} - 0.813e^{-5.93t} \begin{bmatrix} -5.93 & 0 \\ 0.83 & -4.70 \end{bmatrix}. \\ &= \begin{bmatrix} e^{-5.93t} & 0 \\ 0.675e^{-4.7t} - 0.675e^{-5.93t} & e^{-4.7t} \end{bmatrix} \end{aligned} \tag{2.53}$$

The simulation in Fig 2.5 shows the evolution of each element of the state-transition matrix from Example 2.4, 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 from the Lagrange formula that the time evolution of each state is given by a row-wise weighted sum of these elements. The weights, in this case, 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.

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 2.5, this case introduces a slightly different interpretation of the modes contributions to the natural response.

Theorem 2.6. *Consider the same expansion defined in Theorem 2.5. 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:*

$$\begin{cases} \beta_0 + \alpha\beta_1 + \alpha^2\beta_2 + ... + \alpha^{n-1}\beta_{n-1} = e^{\alpha t}cos(\omega t) \\ 0 + \omega\beta_1 + \omega^2\beta_2 + ... + \omega^{n-1}\beta_{n-1} = e^{\alpha t}sin(\omega t) \end{cases}. \tag{2.54}$$

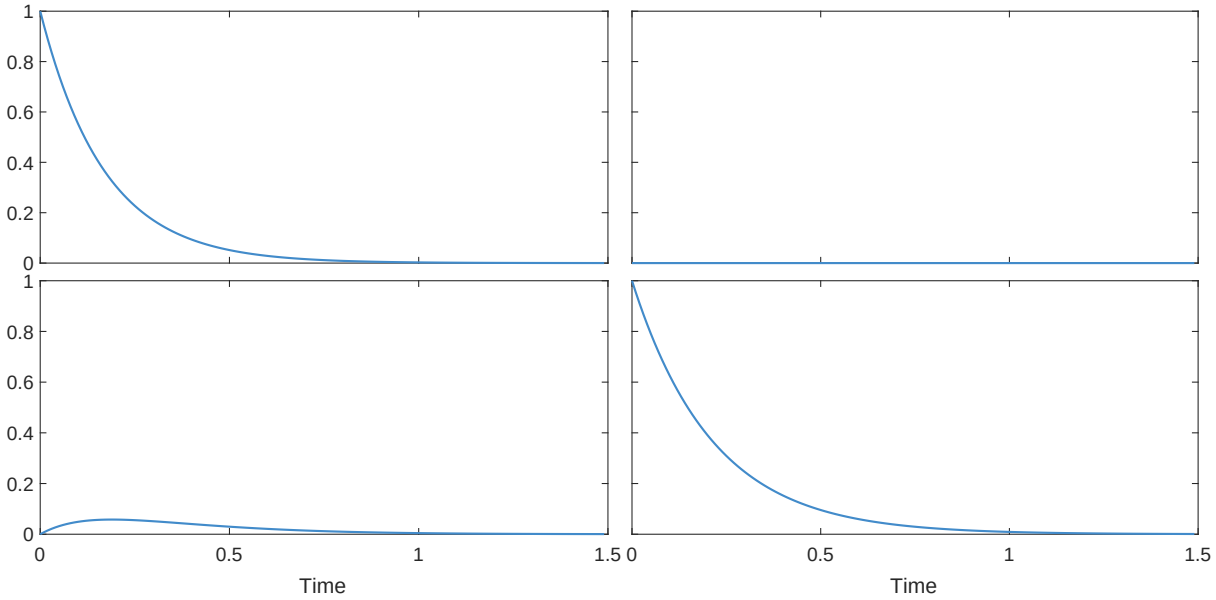


Figure 2.5: Simulation of the state-transition matrix $e^{\boldsymbol{A}t}$ from Example 2.4 for $t \in [0, 1.5]$.

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:

$$\begin{cases} \beta_0 + \lambda_c\beta_1 + \lambda_c^2\beta_2 + ... + \lambda_c^{n-1}\beta_{n-1} = e^{\lambda_c t} \\ \beta_0 + \lambda_c'\beta_1 + (\lambda_c')^2\beta_2 + ... + (\lambda_c')^{n-1}\beta_{n-1} = e^{\lambda_c' t} \end{cases}. \tag{2.55}$$

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:

$$\begin{cases} \beta_0 + 2\alpha\beta_1 + 2\alpha^2\beta_2 + ... + 2\alpha^{n-1}\beta_{n-1} = 2e^{\alpha t}cos(\omega t) \\ 0 + 2j\omega\beta_1 + 2j\omega^2\beta_2 + ... + 2j\omega^{n-1}\beta_{n-1} = 2je^{\alpha t}sin(\omega t) \end{cases}. \tag{2.56}$$

Finally, dividing the first row by 2 and the second row by $2j$ results in (2.54). □

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.

Example 2.5. Consider, again for the sake of illustration, the following example. A toy system is described by the matrix

$$\boldsymbol{A} = \begin{bmatrix} -0.1 & 0.5 \\ -0.5 & -0.1 \end{bmatrix}. \tag{2.57}$$

The eigenvalues of this matrix are given as $\lambda, \lambda' = -0.1 \pm j0.5$. The Sylvester expansion for the exponential of this matrix has the parameter vector $\boldsymbol{\beta} = [\beta_0(t), \beta_1(t)]$ which solves the system

$$\begin{bmatrix} 1 & -0.1 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} \beta_0(t) \\ \beta_1(t) \end{bmatrix} = \begin{bmatrix} e^{-0.1t}cos(0.5t) \\ e^{-0.1t}sin(0.5t) \end{bmatrix}. \tag{2.58}$$

The solution to this system is $\boldsymbol{\beta} = [e^{-0.1t}cos(0.5t) + 0.2e^{-0.1t}sin(0.5t),\ 2e^{-0.1t}sin(0.5t)$. Hence, state-transition matrix is calculated through the expansion:

$$\begin{aligned} e^{\boldsymbol{A}t} &= \beta_0(t)\boldsymbol{I} + \beta_1(t)\boldsymbol{A} \\ &= e^{-0.1t}cos(0.5t) + 0.2e^{-0.1t}sin(0.5t)\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} + 2e^{-0.1t}sin(0.5t)\begin{bmatrix}-0.1 & 0.5\\ -0.5 & -0.1\end{bmatrix}. \\ &= \begin{bmatrix} e^{-0.1t}cos(0.5t) & e^{-0.1t}sin(0.5t) \\ -e^{-0.1t}sin(0.5t) & e^{-0.1t}cos(0.5t)\end{bmatrix} \end{aligned} \tag{2.59}$$

The elements of the resulting state-transition matrix are 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.

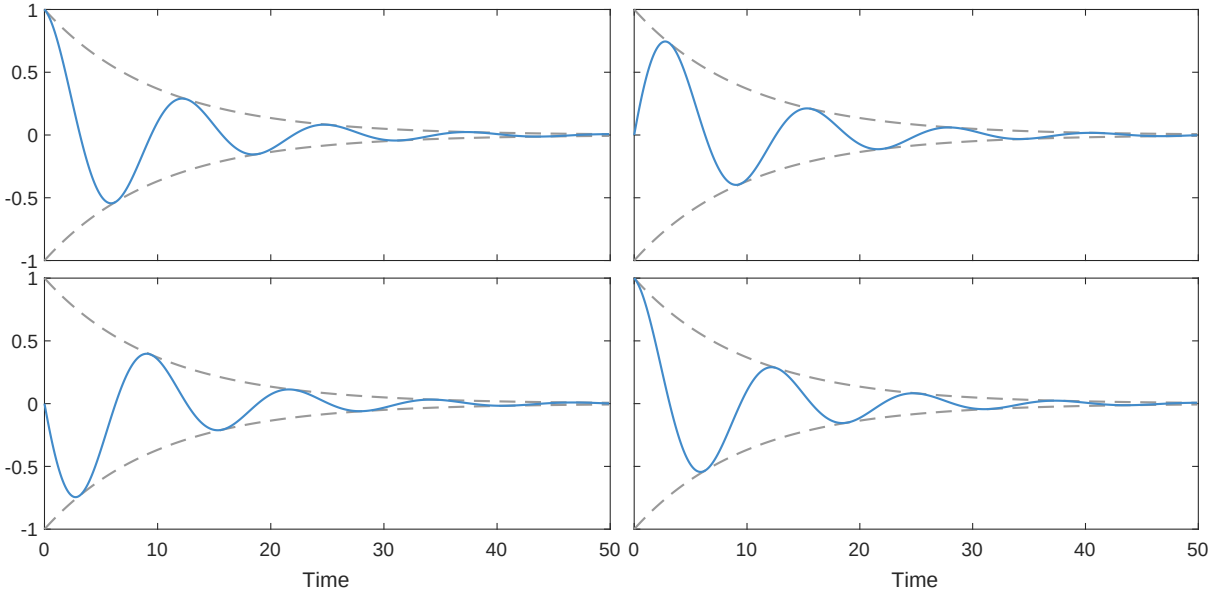


Figure 2.6: Simulation of the state-transition matrix $e^{\boldsymbol{A}t}$ from Example 2.5 for $t \in [0, 50]$.

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, focused on the natural response $\boldsymbol{x}_n(t)$, whereas the forced response $\boldsymbol{x}_f(t)$ was deliberately omitted. Now, the analysis focus on the opposite case. Consider the two systems described by the matrices $(\boldsymbol{A}^{(1)}, b^{(1)})$ and $(\boldsymbol{A}^{(2)}, b^{(2)})$ given below. These matrices are more complete descriptions of the systems from Examples 2.4 and 2.5.

$$\boldsymbol{A}^{(1)} = \begin{bmatrix}-5.93 & 0\\ 0.83 & -4.70\end{bmatrix} \quad b^{(1)} = \begin{bmatrix}3.81\\ -1.09\end{bmatrix}; \qquad \boldsymbol{A}^{(2)} = \begin{bmatrix}-0.1 & 0.5\\ -0.5 & -0.1\end{bmatrix} \quad b^{(2)} = \begin{bmatrix}1\\ 1\end{bmatrix}. \tag{2.60}$$

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 the first state of both systems are shown in Fig. 2.7. Since $\boldsymbol{A}^{(1)}$ is diagonal and $\boldsymbol{A}^{(2)}$ has a pair of complex conjugate eigenvalues, it is possible to consider that those are characteristic responses of aperiodic and pseudo-periodic modes, respectively, to an unitary step input signal.

The unitary step is the default input signal used to analyze the forced response of dynamical systems. This is due to the fact that any continuous signal can be approximated by a sequence of step signals. When discussing the time evolution of systems given forcing input signals,

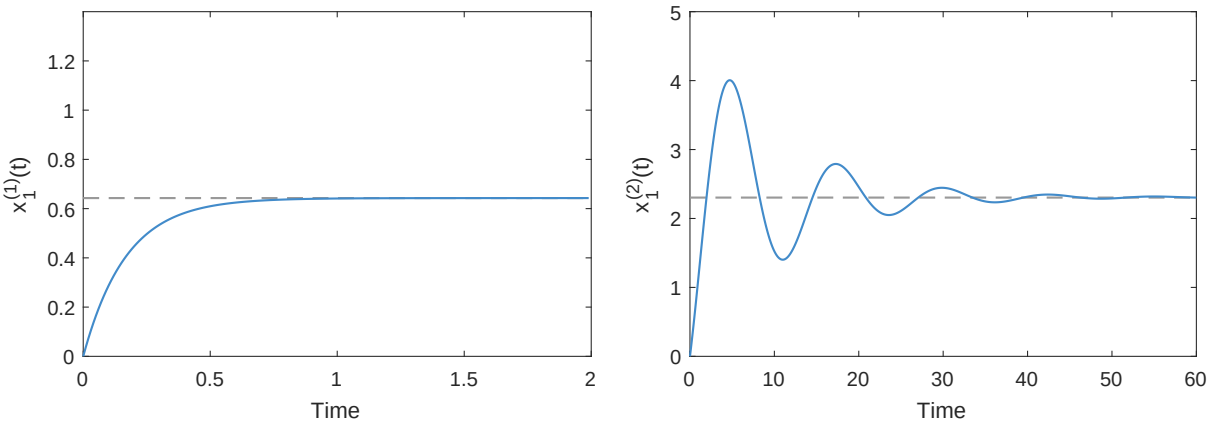


Figure 2.7: One-state forced response of the two systems from (2.60).

the response can be separated into a transient regime and a subsequent steady-state regime. The transient part of the response corresponds to the dynamical evolution from the previous state equilibrium to the next steady-state point. The steady-state point, as already mentioned, corresponds to the point in which the derivatives of the states are zero, which represents the discharge of the initial energy given by the forcing input. These regimes can be characterized in respect to the specifications given below.

Definition 2.6. (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

$$\tau = -\frac{1}{\lambda}. \tag{2.61}$$

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

$$\tau = -\frac{1}{\alpha} \quad ; \quad \omega_n = \sqrt{\alpha^2 + \omega^2} \quad ; \quad \zeta = -\frac{\alpha}{\omega_n}. \tag{2.62}$$

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 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 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.

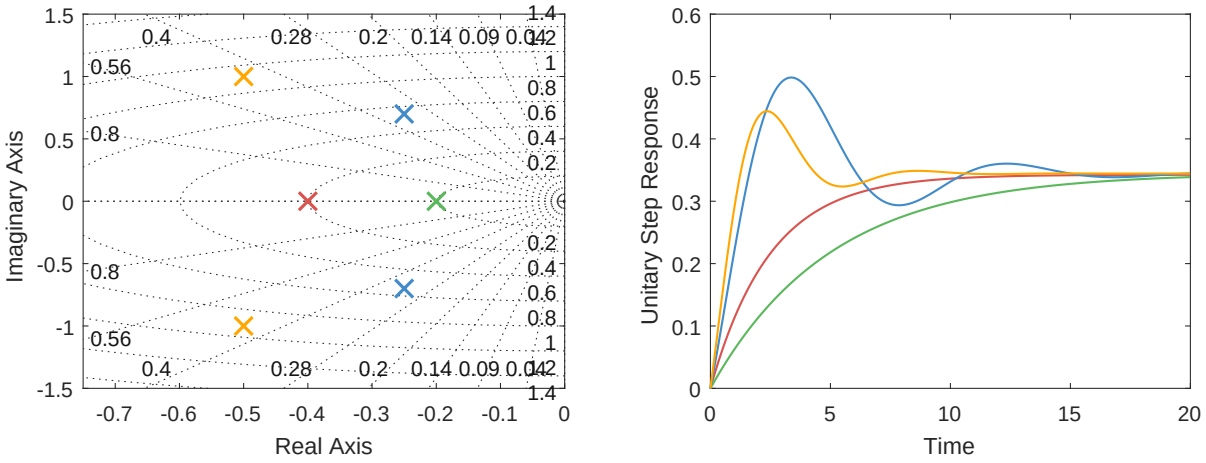


Figure 2.8: Eigenvalues of a system and the forced response associated with their modes.

2.4 Similarity Transformations

A State-Space representation can be interpreted through a system of coordinates. A state, in this context, represents a point 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 representation to the same system. This is the motivation for this section.

First of all, consider a brief reflection about the geometrical interpretation of a State-Space model. Let any state-vector at an arbitrary time t be $\boldsymbol{x}(t)$, defined in the $\mathbb{R}^2$ space. Naturally, it is possible to associate to this vector an orthonormal basis $\boldsymbol{I} \in \mathbb{R}^2$, the 2-dimensional identity matrix. There is, however, the possibility to associate any other arbitrary basis to represent a state-vector and visualize both states through this new perspective. When the basis is not orthogonal, a change in the state-vector in a direction parallel to a component of the basis, i.e., a change in only one element of this vector, also produces a change in other directions if observed through the original orthonormal basis. This is an interesting result, since it allows for the use of a single direction to represent the simultaneous evolution of both states.

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 2.7. *(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:*

$$\begin{cases} \dot{\boldsymbol{z}}(t) = \tilde{\boldsymbol{A}}\boldsymbol{z}(t) + \tilde{\boldsymbol{B}}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \tilde{\boldsymbol{C}}\boldsymbol{z}(t) + \tilde{\boldsymbol{D}}\boldsymbol{u}(t) \end{cases}, \tag{2.63}$$

where:

$$\tilde{\boldsymbol{A}} = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1} \quad ; \quad \tilde{\boldsymbol{B}} = \boldsymbol{P}\boldsymbol{B} \quad ; \quad \tilde{\boldsymbol{C}} = \boldsymbol{C}\boldsymbol{P}^{-1} \quad ; \quad \tilde{\boldsymbol{D}} = \boldsymbol{D}. \tag{2.64}$$

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:

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{P}^{-1}\dot{\boldsymbol{z}}(t) &= \boldsymbol{A}\boldsymbol{P}^{-1}\boldsymbol{z}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{I}\dot{\boldsymbol{z}}(t) &= \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1}\boldsymbol{z}(t) + \boldsymbol{P}\boldsymbol{B}\boldsymbol{u}(t) \\ \dot{\boldsymbol{z}}(t) &= \tilde{\boldsymbol{A}}\boldsymbol{z}(t) + \tilde{\boldsymbol{B}}\boldsymbol{u}(t) \end{aligned} \tag{2.65}$$

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

$$\begin{aligned} \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \\ &= \boldsymbol{C}\boldsymbol{P}^{-1}\boldsymbol{y}(t) + \boldsymbol{D}\boldsymbol{u}(t). \\ &= \tilde{\boldsymbol{C}}\boldsymbol{z}(t) + \tilde{\boldsymbol{D}}\boldsymbol{u}(t) \end{aligned} \tag{2.66}$$

□

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 2.8. *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 matrix $\tilde{\boldsymbol{A}} = \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^{-1}$. In this case, $\boldsymbol{A}$ and $\tilde{\boldsymbol{A}}$ shares 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:

$$\begin{aligned} \boldsymbol{P}^{-1}\tilde{\boldsymbol{A}}\boldsymbol{P}\boldsymbol{v} &= \boldsymbol{\lambda}\boldsymbol{v} \\ \tilde{\boldsymbol{A}}\boldsymbol{P}\boldsymbol{v} &= \boldsymbol{\lambda}\boldsymbol{P}\boldsymbol{v} \end{aligned} \tag{2.67}$$

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

It is clear from previous results that, since both the matrices shares the same set of eigenvalues, the original and transformed model provides the same dynamical responses and, as shown later, the same general dynamic properties. Therefore, the similarity transformation consists in a method to produce new State-Spaces representations that emphasizes some geometrical perspective of the model, hopefully helping to analyze a specific property, without actually changing the relationship between the original model and 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 2.9. *(Diagonalization) Consider a n-dimensional system in State-Space form represented by the matrices $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D})$. Let matrix $\boldsymbol{A}$ have 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 $\boldsymbol{A}$, will result in a 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 $\boldsymbol{A}$:

$$
\begin{aligned}
\boldsymbol{\lambda}\boldsymbol{v} &= \boldsymbol{A}\boldsymbol{v} \\
\begin{bmatrix} \lambda_1\boldsymbol{v}_1 & \lambda_2\boldsymbol{v}_2 & \cdots & \lambda_n\boldsymbol{v}_n \end{bmatrix} &= \begin{bmatrix} \boldsymbol{A}\boldsymbol{v}_1 & \boldsymbol{A}\boldsymbol{v}_2 & \cdots & \boldsymbol{A}\boldsymbol{v}_n \end{bmatrix} \\
\begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} &= \boldsymbol{A} \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix} \\
\begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} &= \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix}^{-1} \boldsymbol{A} \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{bmatrix} \\
\boldsymbol{\Lambda} &= \boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V}
\end{aligned}
\qquad . \tag{2.68}
$$

Thus, $\boldsymbol{\Lambda}$ is a diagonal matrix whose elements are the eigenvalues of matrix $\boldsymbol{A}$. □

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. This result can be easily extended to the case where the eigenvalues are distinct conjugate complex pairs.

In this diagonalization procedure, the elements of the resulting matrix $\boldsymbol{\Lambda}$ are the very own eigenvalues of the original matrix $\boldsymbol{A}$. 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 as:

$$
\begin{aligned}
e^{\boldsymbol{\Lambda}t} = \sum_{k=0}^{\infty} \frac{A^k t^k}{k!} = \sum_{k=0}^{\infty} \frac{t^k}{k!} \begin{bmatrix} \lambda_1^k & 0 & \cdots & 0 \\ 0 & \lambda_2^k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n^k \end{bmatrix} &= \begin{bmatrix} \sum_{k=0}^{\infty} \frac{t^k \lambda_1^k}{k!} & 0 & \cdots & 0 \\ 0 & \sum_{k=0}^{\infty} \frac{t^k \lambda_2^k}{k!} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sum_{k=0}^{\infty} \frac{t^k \lambda_n^k}{k!} \end{bmatrix} \\
&= \begin{bmatrix} e^{\lambda_1 t} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2 t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\lambda_n t} \end{bmatrix} .
\end{aligned}
\tag{2.69}
$$

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 $\boldsymbol{J}$, in which there are decoupled states between blocks. 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 the dynamical system analysis to a practical perspective of control theory, it is necessary to define and discuss some important properties system and their models. These are definitions concerning the stability, controllability and observability of dynamical systems. Although those properties are established from the dynamical model of the system, they can be directly related to questions about control and measurement instrumentation applied to the control system. .

Stability

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 2.7. (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., $\boldsymbol{x}(t) \to \boldsymbol{0}$ as $t \to \infty$. This result is very intuitive from Theorem 2.4, since the vanishing of the natural response implies that $e^{\boldsymbol{A}t} \to \boldsymbol{0}$ 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, as shown in the following theorem.

Theorem 2.10. *(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 $Re[\lambda_i] < 0, \ \forall i \in [1, 2, ..., n]$.*

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 decays exponentially, as this theorem shows. 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. 2.9 for three different poles configurations, given an unitary step.

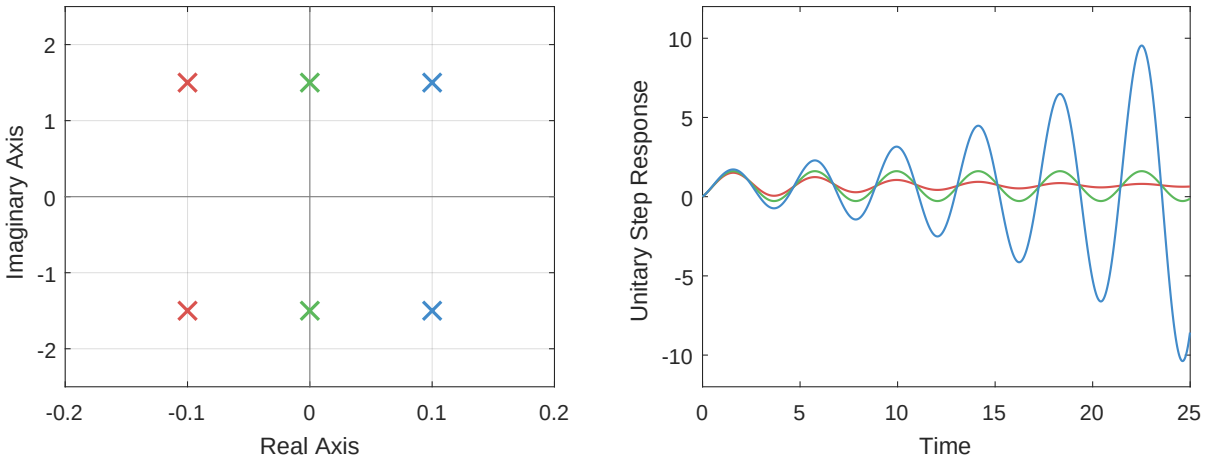


Figure 2.9: Stability of forced responses given the positions of the system poles.

Controllability

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 drives the system to any arbitrary point in the space, in a finite time interval. 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.

Definition 2.8. (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 2.11. *(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:*

$$\boldsymbol{\mathcal{C}} = \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \boldsymbol{A}^2\boldsymbol{B} & \cdots & \boldsymbol{A}^{n-1}\boldsymbol{B} \end{bmatrix}. \tag{2.70}$$

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:

$$\boldsymbol{x}_f(t) = \int_0^t e^{\boldsymbol{A}(t-\tau)}\boldsymbol{B}\boldsymbol{u}(\tau)d\tau. \tag{2.71}$$

From the Cayley-Hamilton theorem, 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 $\boldsymbol{A}^i$, $i \in [0, 1, ..., n-1]$. Using this theorem to substitute the matrix exponential at (2.71), and letting $\tau_2 = t - \tau$ for easier manipulation, the forced response can be represented as:

$$\begin{aligned}\boldsymbol{x}_f(t) &= \int_0^t \left(\sum_{i=0}^{n-1}\beta_i(\tau_2)\boldsymbol{A}^i\right)\boldsymbol{B}\boldsymbol{u}(t-\tau_2)d\tau_2 = \sum_{i=0}^{n-1}\left(\boldsymbol{A}^i\boldsymbol{B}\right)\int_0^t \beta_i(\tau_2)\boldsymbol{u}(t-\tau_2)d\tau_2 \\ &= \sum_{i=0}^{n-1}\left(\boldsymbol{A}^i\boldsymbol{B}\right)\tilde{\beta}_i(\boldsymbol{u}, t),\end{aligned} \tag{2.72}$$

where $\boldsymbol{A}^i\boldsymbol{B}$ are the columns of the matrix $\boldsymbol{\mathcal{C}}$ and $\tilde{\beta}_i(\boldsymbol{u}, t)$, comprising the integral term, is a function that depends only on the input signal $\boldsymbol{u}(t)$ and time t. This result implies that the forced 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 it spans the entire n-dimensional space, i.e., the entire state space, and thus any desirable state vector 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.

Observability

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. Thus, it is necessary to know if it is possible to reconstruct $\boldsymbol{x}(t)$ through $\boldsymbol{y}(t)$.

Definition 2.9. 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 an observability matrix.

Theorem 2.12. *(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:*

$$\boldsymbol{\mathcal{O}} = \begin{bmatrix}\boldsymbol{C} & \boldsymbol{C}\boldsymbol{A} & \boldsymbol{C}\boldsymbol{A}^2 & \cdots & \boldsymbol{C}\boldsymbol{A}^{n-1}\end{bmatrix}^T. \tag{2.73}$$

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, observability and their duality, together with the conditions to fulfill these properties, were first introduced by [Kalman, 1960b]. 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 $\boldsymbol{C}$ 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. In the case which these additional devices are not feasible, one is restricted to control and/or observe only a subspace 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. 2.10.

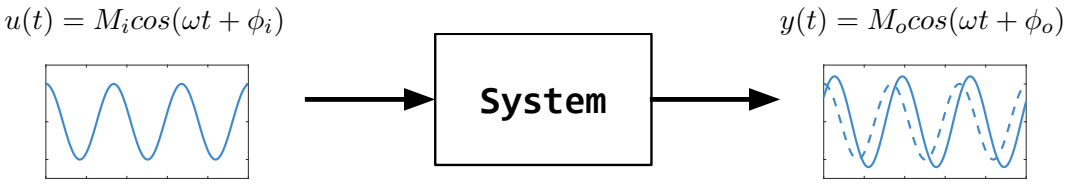


Figure 2.10: Illustration of the steady-state response of LTI systems to sinusoidal inputs.

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:

$$g(t) = \frac{y(t)}{u(t)} = \frac{M_o\angle\phi_o}{M_i\angle\phi_i} = M_g\angle\phi_g, \tag{2.74}$$

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. 2.11, for a transfer function obtained by applying the transformation of Theorem 2.1 to the system from (2.51):

$$\boldsymbol{G}(s) = \begin{bmatrix} \dfrac{3.81}{s + 5.92} & \dfrac{-1.09s - 3.33}{s^2 + 10.62s + 27.84} \end{bmatrix}. \tag{2.75}$$

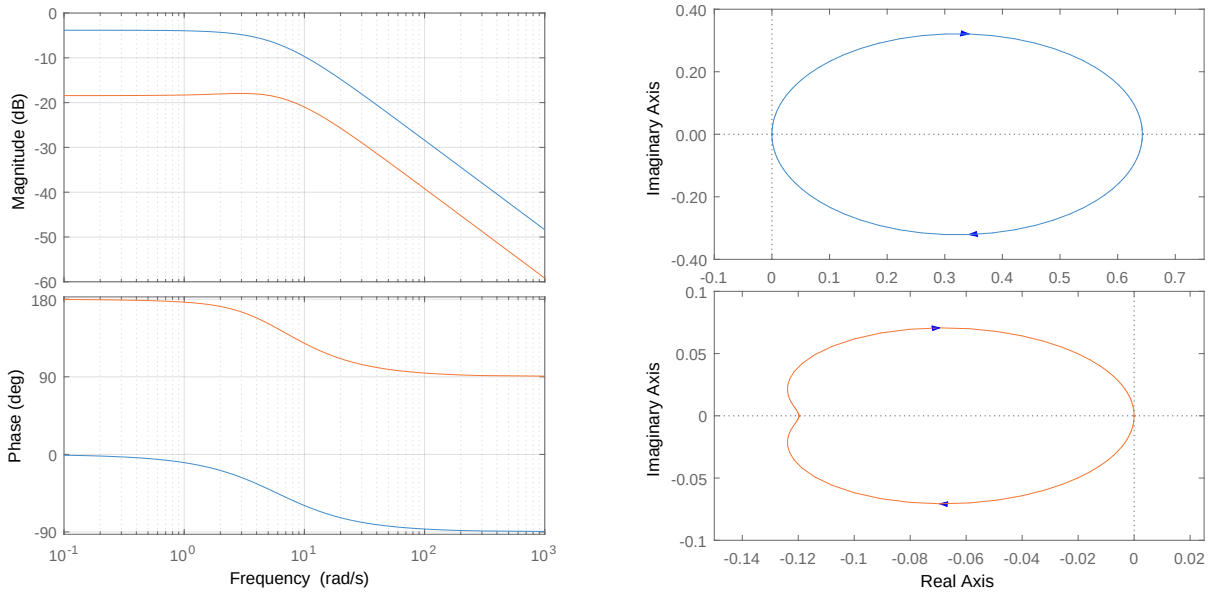


Figure 2.11: Visualization of the two-states system in (2.75) in Bode plots (left) and Nyquist diagrams (right). The blue and orange lines represents the states x_1 and x_2, respectively.

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.

Chapter 3

Controller Synthesis

This chapter discusses the general results and properties for the design of 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 on the system output. For this reason, the results in this section focus on this formulation explicitly through the state-equations, where the output-equations are made implicit.

3.1 State Feedback Controllers

In modern control theory, advances in computer performance and the maturing of State-Space model analysis have made the design of controllers using State-Space models feasible for real-world applications. This is usually desirable since 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 very popular, feedback controller used in those settings is the *Full-State Feedback Controller*, defined below.

Definition 3.1. Full-State Linear Feedback Given a linear system in State-Space representation, an input action $\boldsymbol{u}(t)$ is calculated by the linear control law $\pi(\cdot)$ through state-feedback as:

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

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*.

A block diagram of the closed-loop system for the state-feedback controller is shown at Fig. 3.1. In this schematic each blocks represents a function or operation that is applied to the denoted signals, which themselves are represented by the arrows. The direction of the arrows indicates whether the signal variable is an operand or a result of an operation.

Notice that the control law of the full-state feedback is both linear and time-invariant, which makes the analysis of the closed-loop system similar to the one used in open-loop configurations. Of course, this is a specific 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:

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

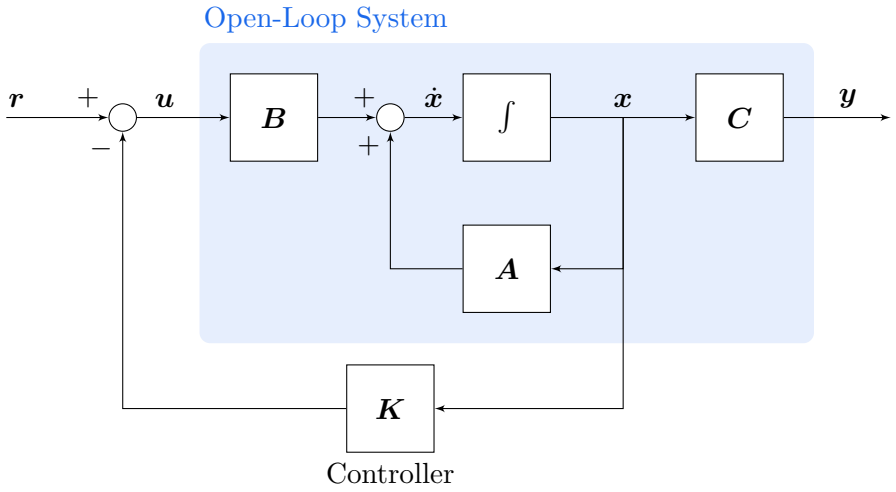


Figure 3.1: Block diagram of a state-feedback closed-loop system.

Clearly, 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 variable is now the reference signal $\boldsymbol{r}(t)$. Therefore, an appropriate choice of $\boldsymbol{K}$ allows to modify the behavior of the closed-loop system such that a desirable behavior is obtained. 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 [Chen, 1998]. An equivalent result can be provided for MIMO systems, but the result is cumbersome and does not highlight the main results for the state-feedback. Using the canonical form just presented, it is possible to discuss an important 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 be placed arbitrarily in the complex plane, as long as complex conjugate eigenvalues are assigned in pairs.

Proof. Suppose 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 2.8 it is known that $\boldsymbol{A}_{cl}$ and $\tilde{\boldsymbol{A}}_{cl}$ share the same set of eigenvalues, and, consequently, the same characteristic equations. Consider this characteristic equation as:

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

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

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

Substituting 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 that yield the desirable eigenvalues. Since $\tilde{\boldsymbol{A}}_{cl}$ and $\boldsymbol{A}_{cl}$ shares the same set of eigenvalues, it can be concluded that it is possible to assign the system poles directly through matrix $\tilde{\boldsymbol{K}}$. □

Notice 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). The 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 $\boldsymbol{K}$. The result is preserved for MIMO systems, although the design of $\boldsymbol{K}$ is not as straightforward, since it is not unique for a desired set of eigenvalues in this case [Moore, 1975].

Using the parameters from Definition 2.6, the positions of the closed-loop system poles can be designed 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. Algorithm 1 summarizes a simple procedure of designing an appropriate feedback gain matrix, given desirable pole positions.

Albeit being a simple formula, Pole-Placement can be used in several applications to yield controllers capable of achieve performance requirements. Of course, the designer must take into account that, although the eigenvalue assignment allows for the whole complex plane, a careless choice of eigenvalues could result in controllers requesting 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 the objective that the device is expected to fulfill. There exists two modes of operation that the controller may require for the system: regulation and reference tracking. 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.

Regulator Problem

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 2.10). 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. However, it is common to have linear systems that are linearized versions of nonlinear models, using the approximation from Theorem 2.2. In these cases, the regulator actually has 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.

Example 3.1. For the sake of illustration, considers the first system of (2.60), repeated here:

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

It is easy to verify that this system is controllable, since its controllability matrix is full-rank. In this case, consider the gains $K_1 = [-0.993, 1.221]$, $K_2 = [0.782, 0.549]$ and $K_3 = [0.81, 5.20]$ to yield regulator controllers. The correspondent closed-loop systems $\boldsymbol{A}_{cl}^{(i)}$ for each i-th controller are given as:

$$\boldsymbol{A}_{cl}^{(1)} = \begin{bmatrix} -2.144 & -4.651 \\ -0.257 & -3.355 \end{bmatrix}; \quad \boldsymbol{A}_{cl}^{(2)} = \begin{bmatrix} -8.906 & -2.092 \\ 1.692 & -4.093 \end{bmatrix}; \quad \boldsymbol{A}_{cl}^{(3)} = \begin{bmatrix} -9.014 & -19.812 \\ 1.723 & 1.014 \end{bmatrix}. \tag{3.12}$$

Since the open-loop system was obtained by the linearized model in (2.39) 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 $t \to \infty$.

The responses of the three closed-loop systems calculated in Example 3.1 are shown in Fig. 3.2. The simulations assumed initial state $\Delta\boldsymbol{x}(t) = \boldsymbol{0} - \boldsymbol{x}_o$, and the response and input signal were vertically corrected by $\boldsymbol{x}_o$ and $\boldsymbol{u}_o$, respectively. Notice that, as expected, the feedback gains $\boldsymbol{K}_i$ can effectively change the system behavior, even causing pseudo-periodic responses.

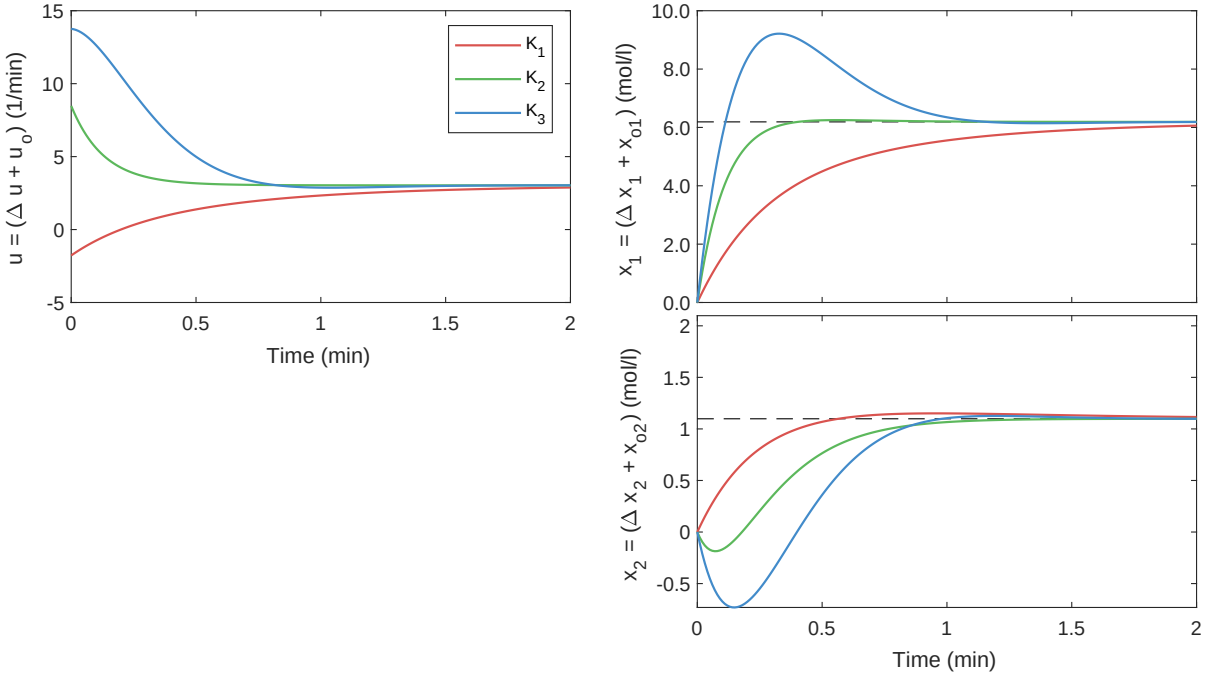


Figure 3.2: Simulation of three closed-loop systems from Example 3.1 showing the input signal (left) and state responses (right).

Reference Tracking Problem

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 perform the 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:

$$G(s) = \frac{Y(s)}{R(s)} = \frac{\beta_1 s^{n-1} + \beta_2 s^{n-2} + \cdots + \beta_{n-1} s + \beta_n}{s^n + \tilde{\alpha}_1 s^{n-1} + \tilde{\alpha}_2 s^{n-2} + \cdots + \tilde{\alpha}_{n-1} s + \tilde{\alpha}_n}. \tag{3.13}$$

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:

$$G(0)F = 1 \Rightarrow F = \frac{\tilde{\alpha}_n}{\beta_n}, \tag{3.14}$$

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 design of F becomes more complex since it would be clearly time-varying. 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* $\boldsymbol{F}$ to adapt the reference as $\tilde{\boldsymbol{r}}(t) = \boldsymbol{F}\boldsymbol{r}(t)$, resulting in the following closed-loop state equation and equivalent solution:

$$\begin{array}{ll} \textbf{State Equation:} & \textbf{Lagrange solution:} \\ \dot{\boldsymbol{x}}(t) = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\,\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{F}\boldsymbol{r}(t) & \boldsymbol{x}(t) = e^{(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K})t}\boldsymbol{x}(t_0) + \int_{t_0}^{t} e^{(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K})(t-\tau)}\boldsymbol{B}\boldsymbol{F}\boldsymbol{r}(\tau)d\tau \end{array} \tag{3.15}$$

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 has not been included 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 benefit 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) Consider a State-space system and augmented state $\boldsymbol{x}_a : \mathbb{R} \to \mathbb{R}^p$ defined by:

$$\boldsymbol{x}_a(t) = \int_0^t \boldsymbol{r}(\tau) - \boldsymbol{y}(\tau)d\tau \Longrightarrow \dot{\boldsymbol{x}}_a(t) = \boldsymbol{r}(t) - \boldsymbol{y}(t). \tag{3.16}$$

A robust tracking (or servo) controller, defined by the gain $\tilde{\boldsymbol{K}} = \begin{bmatrix}\boldsymbol{K} & \boldsymbol{K}_a\end{bmatrix}$, for $\boldsymbol{K} \in \mathbb{R}^{r\times n}$ and $\boldsymbol{K}_a \in \mathbb{R}^{r\times p}$, operates on the following augmented version of the original system:

$$\begin{cases} \begin{bmatrix}\dot{\boldsymbol{x}}(t)\\ \dot{\boldsymbol{x}}_a(t)\end{bmatrix} &= \begin{bmatrix}\boldsymbol{A}-\boldsymbol{B}\boldsymbol{K} & -\boldsymbol{B}\boldsymbol{K}_a\\ -\boldsymbol{C} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}(t)\\ \boldsymbol{x}_a(t)\end{bmatrix} + \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{I}_p\end{bmatrix}\boldsymbol{r}(t)\\ \boldsymbol{y}(t) &= \begin{bmatrix}\boldsymbol{C} & \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}(t)\\ \boldsymbol{x}_a(t)\end{bmatrix}\end{cases} \tag{3.17}$$

or, equivalently:

$$\begin{cases}\dot{\tilde{\boldsymbol{x}}}(t) = \tilde{\boldsymbol{A}}_{cl}\tilde{\boldsymbol{x}}(t) + \tilde{\boldsymbol{B}}\boldsymbol{r}(t)\\ \boldsymbol{y}(t) = \tilde{\boldsymbol{C}}\tilde{\boldsymbol{x}}(t)\end{cases}. \tag{3.18}$$

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 in Fig. 3.3 illustrates how an integrator can be included to the block diagram of the control loop.

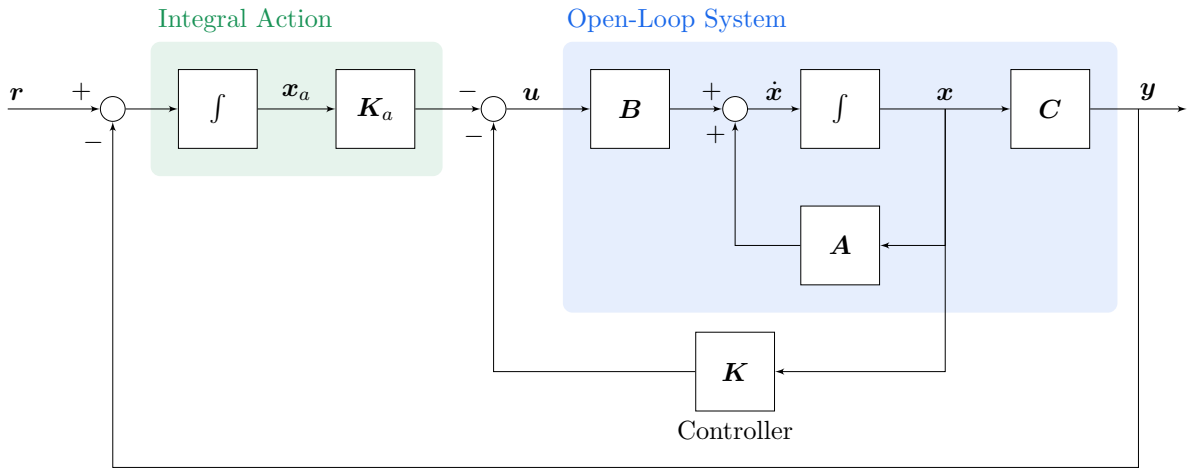


Figure 3.3: Block diagram of a state-feedback closed-loop system with integral action.

Notice that state-equation for closed-loop of this new augmented system is resulting from

$$\tilde{\boldsymbol{A}}_{cl} = \begin{bmatrix}\boldsymbol{A} & \boldsymbol{0}\\ -\boldsymbol{C} & \boldsymbol{0}\end{bmatrix} - \begin{bmatrix}\boldsymbol{B}\boldsymbol{K} & \boldsymbol{B}\boldsymbol{K}_a\\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix} = \begin{bmatrix}\boldsymbol{A} & \boldsymbol{0}\\ -\boldsymbol{C} & \boldsymbol{0}\end{bmatrix} - \begin{bmatrix}\boldsymbol{B}\\ \boldsymbol{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{K} & \boldsymbol{K}_a\end{bmatrix} = \tilde{\boldsymbol{A}} - \tilde{\boldsymbol{B}}\tilde{\boldsymbol{K}}. \tag{3.19}$$

In this case, it must be discussed whether the new gain $\tilde{\boldsymbol{K}}$ still preserves the eigenvalue assignment property of a state-feedback gain defined on the original State-Space model.

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 new controllability

matrix $\tilde{\boldsymbol{\mathcal{C}}} \in \mathbb{R}^{(n+p)\times nr}$ is calculated as:

$$\begin{aligned}\tilde{\boldsymbol{\mathcal{C}}} &= \begin{bmatrix} B & \boldsymbol{A}B & \boldsymbol{A}^2B & \boldsymbol{A}^3B & \cdots & \boldsymbol{A}^{n-1}B \\ 0 & -CB & -C\boldsymbol{A}B & -C\boldsymbol{A}^2B & \cdots & -C\boldsymbol{A}^{n-2}B \end{bmatrix} \\ &= \begin{bmatrix} 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & -\alpha_1(\alpha_1^2-\alpha_2)+\alpha_2\alpha_1-\alpha_3 & \cdots & \Delta_1(\alpha_1,...,\alpha_n) \\ 0 & 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & \cdots & \Delta_2(\alpha_1,...,\alpha_n) \\ 0 & 0 & 1 & -\alpha_1 & \cdots & \Delta_1(\alpha_1,...,\alpha_n) \\ 0 & 0 & 0 & 1 & \cdots & \Delta_3(\alpha_1,...,\alpha_n) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \Delta_n(\alpha_1,...,\alpha_n) \\ 0 & -\beta_1 & -\beta_1\alpha_1-\beta_2 & -\beta_1(\alpha_1^2-\alpha_2)+\beta_2\alpha_1-\beta_3 & \cdots & \Delta_{n+1}(\alpha_1,...,\alpha_n) \end{bmatrix},\end{aligned} \tag{3.20}$$

where $\Delta_i(\alpha_1,...,\alpha_n)$ is a polynomial used to save space in the equations. By inspection of this matrix, it is possible to discover a pattern between the rows. Since elementary operations between the rows $r_1, r_2, ..., r_n$ do not 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:

$$\tilde{\boldsymbol{\mathcal{C}}} = \begin{bmatrix} 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & -\alpha_1(\alpha_1^2-\alpha_2)+\alpha_2\alpha_1-\alpha_3 & \cdots & \Delta_1(\alpha_1,...,\alpha_n) \\ 0 & 1 & -\alpha_1 & -\alpha_1^2-\alpha_2 & \cdots & \Delta_2(\alpha_1,...,\alpha_n) \\ 0 & 0 & 1 & -\alpha_1 & \cdots & \Delta_3(\alpha_1,...,\alpha_n) \\ 0 & 0 & 0 & 1 & \cdots & \Delta_4(\alpha_1,...,\alpha_n) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \Delta_n(\alpha_1,...,\alpha_n) \\ 0 & 0 & 0 & 0 & \cdots & \beta_n \end{bmatrix}. \tag{3.21}$$

Since $G(s)$ 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. From this it can be concluded that the augmented system $(\tilde{\boldsymbol{A}}, \tilde{B})$ is controllable and, from Theorem 3.2, the eigenvalues of $\tilde{\boldsymbol{A}}$ can be assigned anywhere in the complex plane. □

Although the result was only shown for the SISO case, the main intuition still holds in the MIMO case, but the equations become cumbersome. 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 robust tracking and disturbance rejection can be taken from the fact that the first row of (3.19) has the same form of a basic regulator, except from the term $\boldsymbol{B}\boldsymbol{K}_a$ which is a linear function of $\boldsymbol{x}_a(t)$. Because of this, a control action will always be requested whenever $\boldsymbol{x}_a(t) \neq \boldsymbol{0}$, i.e., when there is an error between the reference and the output signal. When $\boldsymbol{x}_a(t) = \boldsymbol{0}$, the equation reduces to a simple regulator. A more quantitative analysis on tracking State-Space controllers can be found in [Franklin et al., 2018].

Example 3.2. 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 and has no zeros at $s = 0$. Thus, a robust tracking controller can be designed by state-feedback. Consider the 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 correspondent closed-loop systems $tilde\boldsymbol{A}_{cl}^{(i)}$ for each i-th controller are given as:

$$\tilde{\boldsymbol{A}}_{cl}^{(1)} = \begin{bmatrix} -9.707 & 4.65 & -95.25 \\ 1.92 & -6.04 & 27.45 \\ 0 & -1.00 & 0 \end{bmatrix}; \quad \tilde{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\tilde{\boldsymbol{A}}_{cl}^{(2)} = \begin{bmatrix} -2.94 & 2.09 & -95.25 \\ -0.02 & -5.30 & 27.46 \\ 0 & -1.00 & 0 \end{bmatrix}; \quad \tilde{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{3.22}$$

$$\tilde{\boldsymbol{A}}_{cl}^{(3)} = \begin{bmatrix} -2.84 & 19.81 & -190.50 \\ -0.06 & -10.41 & 54.91 \\ 0 & -1.00 & 0 \end{bmatrix}; \quad \tilde{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

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.

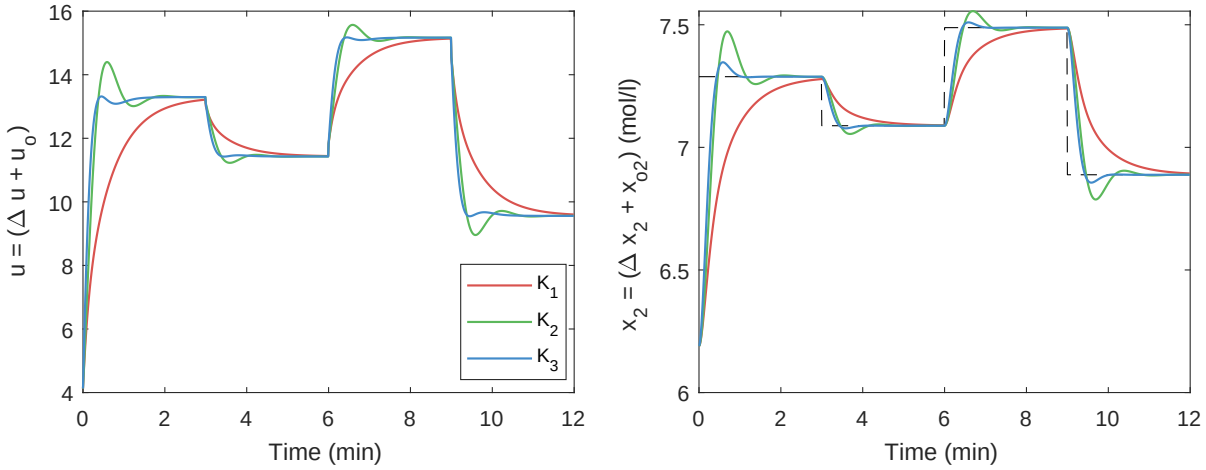


Figure 3.4: Simulation of three closed-loop systems showing the input signal (left) and state response (right) for reference tracking. The reference signal is indicated by the black dashed line.

3.3 Deterministic State Observers

Until now, the state feedback was discussed from the perspective that the device has direct access to the real value of all the states of the system. This assumption is unrealistic, since the states are only observed through the output signal $\boldsymbol{y}(t)$, 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. 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 the states from the observations.

A device that determines 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 several possible techniques, a 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 an observer gain $\boldsymbol{L} \in \mathbb{R}^{n \times p}$, the estimated state-vector $\hat{\boldsymbol{x}}(t)$ is represented by the observer system:

$$\dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{A}\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{L}\left(\boldsymbol{y}(t) - \boldsymbol{C}\hat{\boldsymbol{x}}(t)\right), \tag{3.23}$$

or, equivalently:

$$\dot{\hat{\boldsymbol{x}}}(t) = \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{L}\boldsymbol{y}(t). \tag{3.24}$$

The observer system works as a parallel system that is simulated alongside 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 (3.24):

$$\begin{aligned}
\dot{\boldsymbol{e}} &= \boldsymbol{x} - \hat{\boldsymbol{x}} \\
&= \left(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}\right) - \left(\left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\hat{\boldsymbol{x}} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{L}\boldsymbol{C}\boldsymbol{x}\right) \\
&= \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\boldsymbol{x} - \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\hat{\boldsymbol{x}} \\
&= \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\left(\boldsymbol{x} - \hat{\boldsymbol{x}}\right) \\
&= \left(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}\right)\boldsymbol{e}
\end{aligned}, \tag{3.25}$$

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, since it implies that the natural response of $\dot{\boldsymbol{e}}$ vanishes and there is no forcing input action to yield forcing response to this variable (Theorem 2.10). Therefore, a specific gain $\boldsymbol{L}$ could be designed to ensure this condition and allows the observer to asymptotically track the state-vector with a desired behavior. The following theorem relates this statement with the choice of a gain $\boldsymbol{L}$.

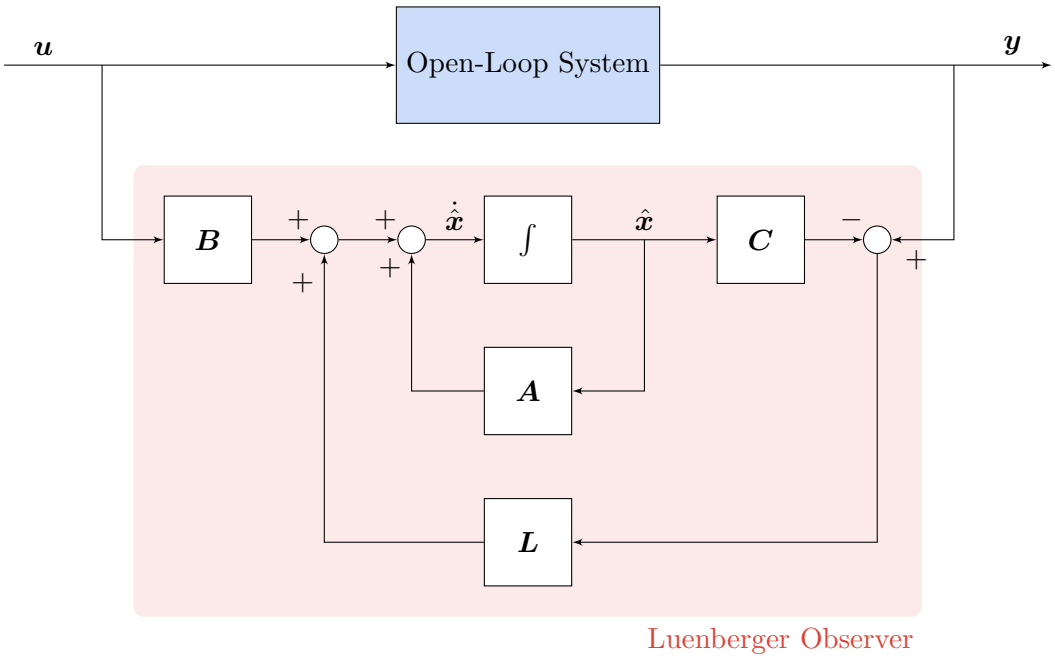


Figure 3.5: Block diagram of a State-Space system with Luenberger observer.

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 $\boldsymbol{A}_{obs} = \boldsymbol{A} - \boldsymbol{LC}$ can arbitrarily be assigned anywhere in the complex plane, as long as 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 [Chen, 1998], 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 $\boldsymbol{K}$ can also place the eigenvalues of the matrix $(\tilde{\boldsymbol{A}}_{obs})^T = \boldsymbol{A} - \boldsymbol{K}^T\boldsymbol{C}$. Therefore, making $\boldsymbol{L} = \boldsymbol{K}^T$ proves the theorem. □

The procedure stated in this proof highlights the similarities between existing closed-loop observers, such as the Luenberger observer, and closed-loop controllers, such 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 an observer gain such that each eigenvalue of $\boldsymbol{A}_{obs}$ is on the left-half side of the complex plane.

The only reason to develop an observer is to allow 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:

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) - \boldsymbol{B}\boldsymbol{K}\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{r}(t) \\ \dot{\hat{\boldsymbol{x}}}(t) = (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C} - \boldsymbol{B}\boldsymbol{K})\,\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{r}(t) + \boldsymbol{L}\boldsymbol{C}\boldsymbol{x}(t) \end{cases} . \tag{3.26}$$

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, states that design of these two gains are independent.

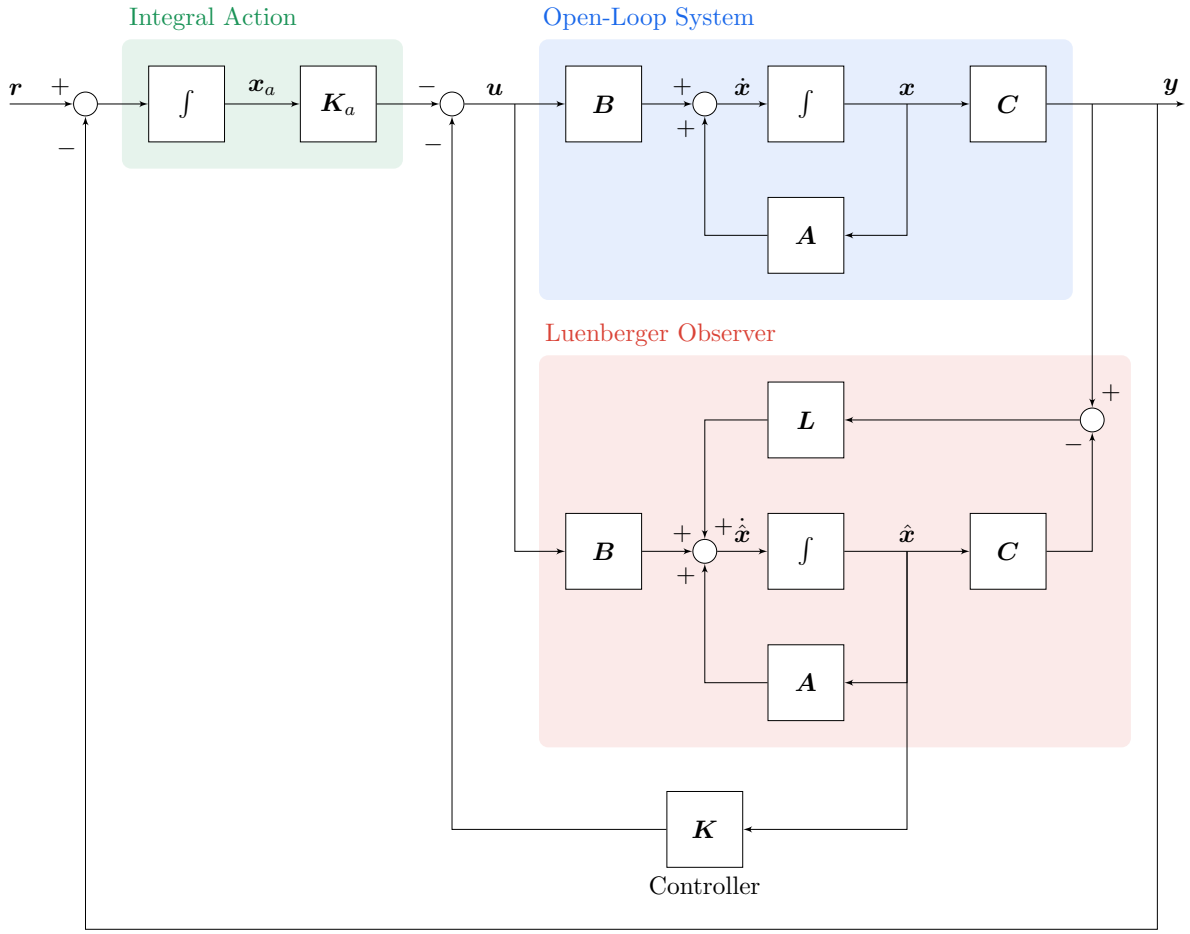


Figure 3.6: Block diagram of a state-feedback closed-loop system with integral action and Luenberger observer.

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* $(\boldsymbol{A}-\boldsymbol{BK})$ *are independent from those of* $(\boldsymbol{A}-\boldsymbol{LC})$*.*

Proof. Consider a feedback from estimated states as defined in (3.26). The controller-estimator system can be rewritten as a single state equation:

$$\begin{bmatrix}\dot{\boldsymbol{x}}\\ \dot{\hat{\boldsymbol{x}}}\end{bmatrix} = \underbrace{\begin{bmatrix}\boldsymbol{A} & -\boldsymbol{BK}\\ \boldsymbol{LC} & \boldsymbol{A}-\boldsymbol{LC}-\boldsymbol{BK}\end{bmatrix}}_{\tilde{\boldsymbol{A}}}\begin{bmatrix}\boldsymbol{x}\\ \hat{\boldsymbol{x}}\end{bmatrix} + \underbrace{\begin{bmatrix}\boldsymbol{B}\\ \boldsymbol{B}\end{bmatrix}}_{\tilde{\boldsymbol{B}}}\boldsymbol{r}. \tag{3.27}$$

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

$$\underbrace{\begin{bmatrix}\boldsymbol{I} & 0\\ \boldsymbol{I} & -\boldsymbol{I}\end{bmatrix}}_{\boldsymbol{P}}\begin{bmatrix}\boldsymbol{x}\\ \hat{\boldsymbol{x}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{x}-\hat{\boldsymbol{x}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{e}\end{bmatrix}. \tag{3.28}$$

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

$$\begin{bmatrix}\dot{\boldsymbol{x}}\\ \dot{\boldsymbol{e}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{A}-\boldsymbol{BK} & -\boldsymbol{BK}\\ \boldsymbol{0} & \boldsymbol{A}-\boldsymbol{LC}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{e}\end{bmatrix} + \begin{bmatrix}\boldsymbol{B}\\ \boldsymbol{0}\end{bmatrix}\boldsymbol{r}. \tag{3.29}$$

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

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 systems allows for a full characterization of its behavior, but the real system will exhibit a different response due to these limitations and, essentially, due to unmodelled 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.

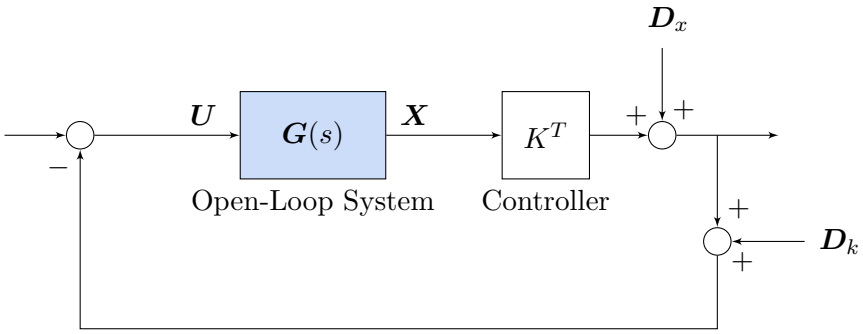


Figure 3.7: Simplified block diagram of a perturbed state-feedback closed-loop.

Using block diagram algebra and Theorem 2.1, it is possible to associate a forward-path transfer function $\boldsymbol{G}_f(s)$ given as [Anderson and Moore, 1990]:

$$\boldsymbol{G}_f(s) = -\boldsymbol{K}^T G(s) = -\boldsymbol{K}^T \left(s\boldsymbol{I} - \boldsymbol{A}\right)^{-1} B. \tag{3.30}$$

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:*

$$T(s) = \frac{KG(s)}{1 + KG(s)}. \tag{3.31}$$

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 (3.30). 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 [Nise, 2015]. The introduction of these stability evaluation techniques may seem redundant, given that the closed-loop BIBO stability can still be characterized from Theorem 2.10. 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:

$$GM = \frac{1}{|T(j\omega_{pc})|}, \tag{3.32}$$

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:

$$PM = \angle T(j\omega_{gc}), \tag{3.33}$$

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 $s = 0$).

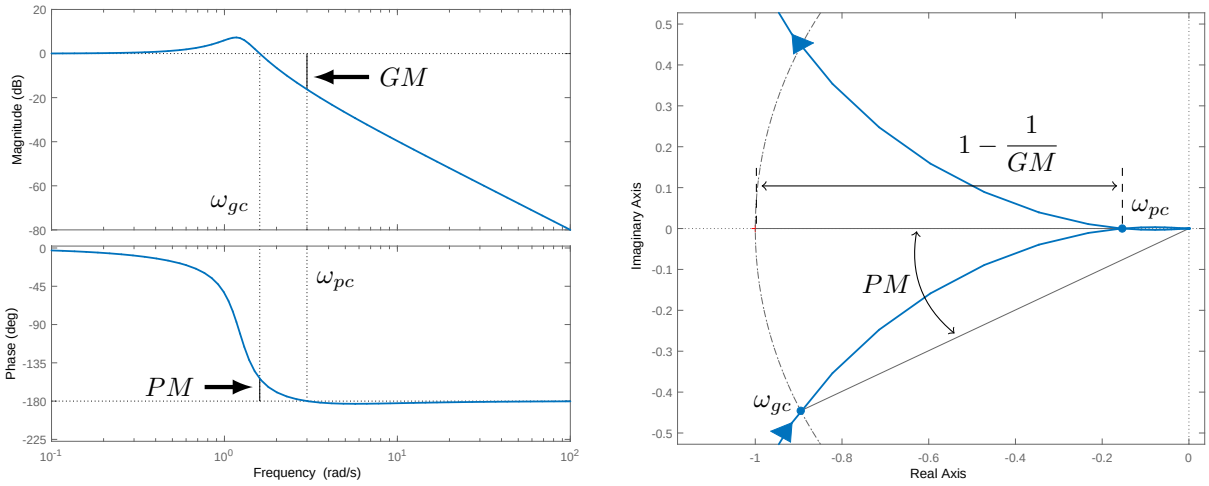


Figure 3.8: Stability margins visualizations given Bode plots (left) and Nyquist diagrams (right) of closed-loop dynamical systems.

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. From this motivation, the use of high-gain controllers have been a working example of how to achieve robust control [Khalil, 2005, Mahmoud and Khalil, 2002], but is usually challenging for the instrumentation in real-world operations. 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.

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 optimizing a cost function that produces an optimal sequence of control actions to achieve the desired control objective, under some conditions. The chapter starts by discussing a general formulation of the optimal control problem and then specializes the 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 estimation is presented. Finally, the main stability and robustness properties of the controllers derived in this chapter are accessed.

4.1 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, is rather time-consuming and demands the designer to consider what are the best pole locations given some hard specifications. In addition to that, the design techniques previously presented becomes more involving when applied to MIMO systems. Thus, a more general and automatic procedure to control synthesis is desirable.

With this motivation, the concept of *Optimal Control* [Anderson and Moore, 1990, Kirk, 1998, Liberzon, 2012, Bertsekas, 2017] was introduced as an alternative strategy for controlling dynamical systems. Optimal controllers determines the necessary action by minimizing a cost function (or maximizing a reward function). These formulations produces feedback controllers that autonomously requires optimal input signals given a desired objective and possible restrictions, begin easily generalized to different systems.

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

Note that this optimization can be converted to maximizing a function $V(\cdot)$ by evaluating $V(\cdot) = -J(\cdot)$. Thus, this document will only refers to optimization as minimizing some cost function. The problem of finding an optimal control law, or optimal control policy, $\pi^*(\cdot)$ depends

on the choice of the cost function $J(\cdot)$, and several optimization techniques can be used to determine the value of $\boldsymbol{u}(t)$ that achieve its minimum. This problem differs from standard optimization problems because it is dependent of time, there can be a decision variable and it is constrained by the dynamics of the model and by the optimal policy action. Solutions to this optimization can be obtained from several methods, however, this work specifically focus on the concept of Dynamic Programming, first introduced by [Bellman, 1954].

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 Regulator* is defined as any controller whose optimal policy over a time interval $t \in [t_0, T]$ minimizes the cost functional:

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

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 optimizing for a sequence of control actions $[\boldsymbol{u}(t_0), \cdots, \boldsymbol{u}(T)]$ that can cause a state trajectory $[\boldsymbol{x}(t_0), \cdots, \boldsymbol{x}(T)]$, which is optimal given the loss functions. As will be shown later, this is a very feasible strategy for controllable linear systems. In an optimization notation, this problem can also be presented as the following *program*:

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

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

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 $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)$, restated as:*

$$V(\boldsymbol{x}, \boldsymbol{u}, t_0) = \int_{t_0}^{T} l(\boldsymbol{x}, \boldsymbol{u}, \tau) d\tau + l_f(\boldsymbol{x}(T)) \tag{4.4}$$

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 $V(\cdot)$ is equivalent to determining the solution of the Hamilton-Jacobi equation, which is given by the partial differential equation:

$$\frac{\partial V^*}{\partial t} = -\min_{u(t)} \left[l(\boldsymbol{x}, \boldsymbol{u}, t) + \left[\frac{\partial V^*}{\partial x} \right]^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \right] \tag{4.5}$$

and the boundary condition:

$$V^*(\boldsymbol{x}, T) = l_f(\boldsymbol{x}(T)). \tag{4.6}$$

Proof. First of all, consider the cost functional $V(\cdot)$ from (4.4). Minimizing this functional with respect to the control inputs $\boldsymbol{u}(t_0), \cdots, \boldsymbol{u}(T)$ consists of 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 as:

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

Notice that this is a recursive formula, since the optimal cost $V^*(\boldsymbol{x}, t)$ depends on $V^*(\boldsymbol{x}, t_r)$, which itself is an optimal cost for the future time $t_r \geq t$. 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:

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

where $\mathcal{O}(\delta t)^2$ denotes high order terms. Since the terms $V^*(\boldsymbol{x}, t)$ and $(\partial V^*/\partial t)\delta t$ do not depend on $\boldsymbol{u}(t)$, they can be disregarded 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, letting $\delta t \to 0$ yields:

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

To prove 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 at $t = T$ to the beginning of the horizon at $t = t_0$. This results from the fact that the last action $u(T)$ depends only on $l_f(\cdot)$, so it can be directly calculated for a desired final state $\boldsymbol{x}(T)$ and used to calculate the remaining optimal costs backward in time. This recursive property, directly evidenced in (4.8), is a statement of the Bellman's Principle of Optimality [Bellman, 1954] and, for this reason, the Hamilton-Jacobi equation in the context of optimal control theory is known as the Hamilton-Jacobi-Bellman (HJB) equation. An illustration of this principle is shown in Fig. 4.1

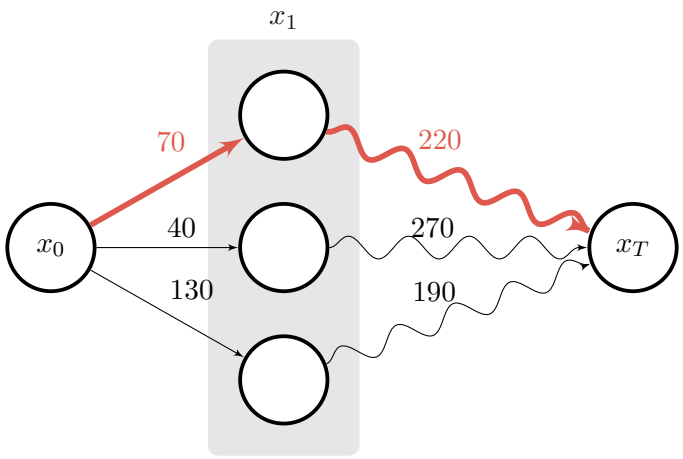


Figure 4.1: Illustration of the Bellman's Principle of Optimality for a system with discrete set of states for a discrete time evolution. Each column represents a time instance and each node represents a possible state. The straight lines represents state transitions given an action with associated costs, 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 (LQ) Controllers

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 Controllers. Since these controllers are realizations of state-feedback controllers, this section discuss the two possible control operations presented in the last chapter, namely regulation and reference tracking.

Regulation

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 a major breakthrough in the development of optimal control theory [Bryson, 1996], and has been a foundation for more complex systems ever since. From

an 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. From a mathematical point of view, the cost function has the advantage of being a quadratic function of the state and input signals, which is desirable since quadratic optimization problems have been widely studied in the literature [Boyd and Vandenberghe, 2004]. 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 by the LQR cost function defined in Definition 4.3 for a linear system. 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 deviation $\lambda \in \mathbb{R}$ in the parameters $(\boldsymbol{x},\ \boldsymbol{u}(t))$ will result in a greater value of the cost. 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 that:

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

or, simply:

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

which directly 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 $V^*(\boldsymbol{x}, t) = \boldsymbol{x}^T(t)\boldsymbol{P}(t)\boldsymbol{x}(t)$. □

This theorem shows that the fact that the cost function $V(\boldsymbol{x}, \boldsymbol{u}, t)$ is quadratic directly implies that the optimal cost $V^*(\boldsymbol{x}, t)$ is also quadratic. Therefore, it is possible to define the optimal action $\boldsymbol{u}^*(t)$ that produce this optimal cost by evaluating its relationship with 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 (4.5) 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 its derivative with respect to the control action and equating it to 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:

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

Thus, substituting 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 action can be calculated analytically. Several others optimal controllers may offer performance improvements, but usually depends on iterative optimization procedures, which can be computationally expensive to calculate for high-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 [Davison and Maki, 1973, Opanuga et al., 2015].

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:

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

This formulation indicates that the Linear Quadratic Regulator solution follows the exact same form of a general state-feedback regulator, represented by the linear but time-variant feedback system $\boldsymbol{A}_{cl}(t) = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}(t))$. 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 an open-loop optimizer, since the optimization solution itself does not depend on the state signal, but only on the system dynamical model.

Reference Tracking

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.

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 turns the tracking problem into a regulation problem. In this case, the optimal controller will try to regulate towards a zero-state for $\boldsymbol{x}_a(t)$, yielding a zero error between the output $\boldsymbol{y}(t)$ and reference signal $\boldsymbol{r}(t)$. The new p diagonal entries of the augmented matrix $\tilde{\boldsymbol{Q}}$ can be interpreted as weights penalizing the state deviation from the reference signal. The solution for this controller is the same as the one derived in Theorem 4.3, with $\boldsymbol{A} = \tilde{\boldsymbol{A}}$ and $\boldsymbol{B} = \tilde{\boldsymbol{B}}$. The resulting control action is equal to:

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

Infinite-Horizon LQ Controllers

To conclude the section, it is also worth mentioning 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 $\boldsymbol{P}$ is the solution of the matrix algebraic Riccati equation:

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

A detailed proof of this result can be found in [Anderson and Moore, 1990]. 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

The state-vector $\boldsymbol{x}(t)$ is a necessary information to calculate $\boldsymbol{u}(t)$ in state-feedback controllers, but may also be inaccessible in practice. In Chapter 3, a deterministic observer, namely the Luenberger observer, was derived to deal with this problem. Despite being possible to utilize the Luenberger observer together with optimal controllers, this method suffers from the same design limitations of the Pole-Placement method, 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.

General State Estimators

First of all, consider a system perturbed by disturbances in the state-response and in the measurements, respectively $\boldsymbol{w} : \mathbb{R} \to \mathbb{R}^n$ and $\boldsymbol{v} : \mathbb{R} \to \mathbb{R}^p$. This configuration is illustrated at Fig. 4.2. If these disturbances are assumed to be zero-mean Gaussian variables, i.e., $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{kf})$ and $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_{kf})$ 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. For the sake of simplicity, the discussion is focused on the discrete-time case.

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

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

for the initial condition $\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0)$.

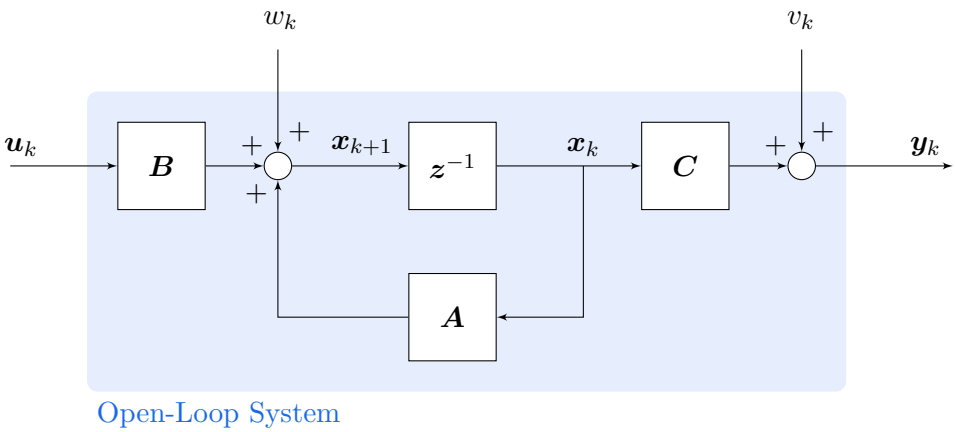


Figure 4.2: Block diagram of a stochastic discrete State-Space system.

With this formulation, the noise variables can actually represent both the effect of external disturbances and the model uncertainty. This is important because dynamical models, even those developed according to a first-principle methodology, are not the true representation of the actual dynamical system or process. The motivation for a discussion of only a discrete State-Space model is to facilitate the interpretation of the statistical properties of the model.

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 variables are assumed to have a Gaussian distribution, both variables $\boldsymbol{x}_k \sim \mathcal{N}(\boldsymbol{\mu}_{x_k}, \boldsymbol{\Sigma}_{x_k})$ and $\boldsymbol{y}_k \sim \mathcal{N}(\boldsymbol{\mu}_{y_k}, \boldsymbol{\Sigma}_{y_k})$ are also Gaussian, where $\boldsymbol{\mu}_{x_k}, \boldsymbol{\mu}_{y_k}$ are their respective mean vectors and $\boldsymbol{\Sigma}_{x_k}, \boldsymbol{\Sigma}_{y_k}$ are the respective covariance matrices. This is because the Gaussian distribution is closed to linear operations. For this reason, each state and output trajectory over a discrete time interval is a collection of time-indexed random variables following the Gaussian distribution, i.e., a stochastic process known as a Gaussian Process [Rasmussen and Williams, 2006].

Definition 4.6. (Gaussian Process) Given a Gaussian process defined as a collection of random variables:

$$\boldsymbol{X} = \{\boldsymbol{x}_k \sim \mathcal{N}(\mu_{\boldsymbol{x}_k}, \Sigma_{\boldsymbol{x}_k}); k \in [0, 1, \cdots, K]\}\,. \tag{4.33}$$

In the case that the variables represents states of a causal system, the joint probability of the random variables can be modeled as:

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

The defined joint probability $p(\boldsymbol{x}_0, \cdots, \boldsymbol{x}_K)$ is interpreted as the probability of occurring a specific state trajectory. The conditional probability $p(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, \cdots, \boldsymbol{x}_0)$ states that the probability associated with a random state variable $\boldsymbol{x}_k$ can be conditioned on the state trajectory $(\boldsymbol{x}_{k-1}, \cdots, \boldsymbol{x}_0)$ from the previous time instants. These two probabilities can be related through the well-known *Chain Rule of Probability*, which, in the context of stochastic processes, is a realization of the *Markov Property* [Stratonovich, 1968]. A nice property of Gaussian distributions, in this context, is that the joint and condition distribution of Gaussian random variables are still Gaussian [Ross, 2010].

Kalman Filter

The dynamic response $\boldsymbol{x}_{k+1}$ of a system for a time instant $k+1$, given its trajectory until the actual state $\boldsymbol{x}_k$, 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 the an actual state stores enough information from the state trajectory until that point, that is possible to determine the next state response directly through its probability. Therefore, the conditional probability could be restated as:

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

A process that exhibits this property is known as a First-Order Markov process. With this formulation, it is always possible to marginalize the probability distribution of a random variable $\boldsymbol{x}_{k+1}$ given the initial state $\boldsymbol{x}_0$, which are related through the 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 Gauss-Markov Process, and using the model from Definition 4.5 it is possible to describe a stochastic dynamical system as the following probabilistic State-Space model.

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

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

for the initial condition given by $p(\boldsymbol{x}_0)$. Furthermore, since the noises $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are zero-mean Gaussian noises, these distributions are modeled as the Normal conditional distributions:

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

A Hidden-Markov Model is a statistical interpretation of dynamical systems that is well-known in the literature [Barber, 2012]. 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 $\boldsymbol{x}_k$ actual value is never known (since it is a latent variable), but can be observed through the emission variable $\boldsymbol{y}_k$, under the assumption that system model is perfectly known. This raises the problem of state estimation as a common inference problem known as the *filtering*, which consists on determining statistically the actual value of $\boldsymbol{x}_k$ given the observations $\boldsymbol{y}_0, \cdots, \boldsymbol{y}_k$ and control actions $\boldsymbol{u}_0, \cdots, \boldsymbol{u}_k$. Thus, the 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 which 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.

There are several approaches to solve the filtering and other related problems in statistical state estimation [Särkkä, 2013], but the most popular method has been the Kalman filter, presented in the seminal paper [Kalman, 1960a]. The discrete formulation for a linear Kalman Filter is given below.

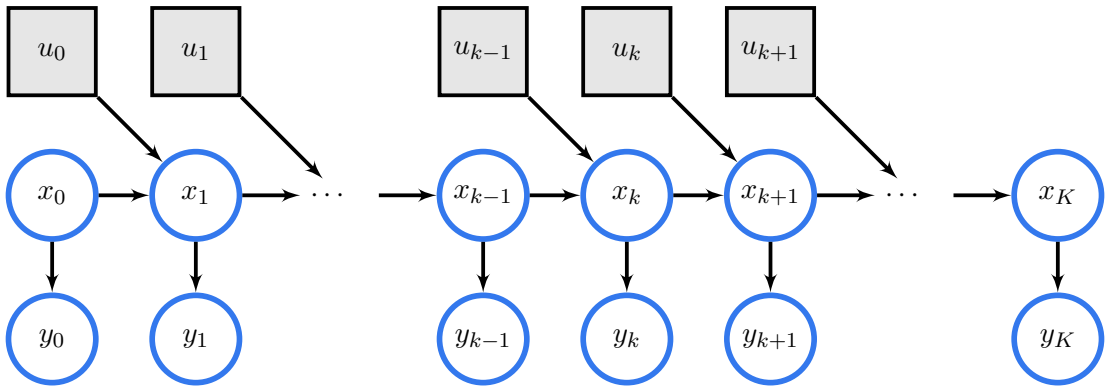


Figure 4.3: A Controlled Hidden Markov Model for a discrete-time system, where the shaded boxes indicates that the inputs are not random variables.

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

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

where $\bar{x}_k$*, which denotes the mean of* $\boldsymbol{x}_k$*, and covariance* $\boldsymbol{P}_k$ *are recursively computed through the prediction and update steps below.*

- *Prediction Step:*

$$\begin{aligned} \hat{\bar{\boldsymbol{x}}}_k &= \boldsymbol{A}\bar{\boldsymbol{x}}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}, \\ \hat{\boldsymbol{P}}_k &= \boldsymbol{A}\boldsymbol{P}_{k-1}\boldsymbol{A}^T + \boldsymbol{Q}_{\boldsymbol{kf}}. \end{aligned} \tag{4.41}$$

- *Update Step:*

$$\begin{aligned} \boldsymbol{K}_k &= \hat{\boldsymbol{P}}_k\boldsymbol{C}^T(\boldsymbol{R}_{kf} + \boldsymbol{C}\hat{\boldsymbol{P}}_k\boldsymbol{C}^T)^{-1}, \\ \bar{\boldsymbol{x}}_k &= \hat{\bar{\boldsymbol{x}}}_k + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k), \\ \boldsymbol{P}_k &= \hat{\boldsymbol{P}}_k - \boldsymbol{K}_k(\boldsymbol{R}_{kf} + \boldsymbol{C}\bar{\boldsymbol{P}}_k\boldsymbol{C}^T)\boldsymbol{K}_k^T. \end{aligned} \tag{4.42}$$

Proof. Consider the transition and emission distributions given as:

$$\begin{cases} p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{A}\bar{\boldsymbol{x}}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}, \boldsymbol{Q}_{kf}) \\ p(\boldsymbol{y}_k|\boldsymbol{x}_k) \quad = \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{C}\bar{\boldsymbol{x}}_k, \boldsymbol{R}_{kf}) \end{cases}. \tag{4.43}$$

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

$$\begin{aligned} p(\boldsymbol{x}_k, \boldsymbol{x}_{k-1}|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) &= p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})p(\boldsymbol{x}_{k-1}|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) \\ &= \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{A}\bar{\boldsymbol{x}}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}, \boldsymbol{Q}_{kf})\mathcal{N}(\boldsymbol{x}_{k-1}|\bar{\boldsymbol{x}}_{k-1}, \boldsymbol{P}_{k-1}) \\ &= \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_{k-1} \\ \boldsymbol{x}_k\end{bmatrix} \middle| \begin{bmatrix}\bar{\boldsymbol{x}}_{k-1} \\ \boldsymbol{A}\bar{\boldsymbol{x}}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}\end{bmatrix}, \begin{bmatrix}\boldsymbol{P}_{k-1} & \boldsymbol{P}_{k-1}\boldsymbol{A}^T \\ \boldsymbol{A}\boldsymbol{P}_{k-1} & \boldsymbol{A}\boldsymbol{P}_{k-1}\boldsymbol{A}^T + \boldsymbol{Q}_{kf}\end{bmatrix}\right). \end{aligned} \tag{4.44}$$

Marginalizing the joint distribution, it is possible to obtain the distribution of $\boldsymbol{x}_k$ conditioned

in the measurement history as:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) = \mathcal{N}(\underbrace{\boldsymbol{A}\bar{\boldsymbol{x}}_{k-1} + \boldsymbol{B}\boldsymbol{u}_{k-1}}_{\hat{\bar{\boldsymbol{x}}}_k}, \underbrace{\boldsymbol{A}\boldsymbol{P}_{k-1}\boldsymbol{A}^T + \boldsymbol{Q}_{kf}}_{\hat{\boldsymbol{P}}_k}), \tag{4.45}$$

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, which applying Lemma A.1 results in:

$$\begin{aligned} p(\boldsymbol{x}_k, \boldsymbol{y}_k|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) &= p(\boldsymbol{y}_k|\boldsymbol{x}_k)p(\boldsymbol{x}_k|\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) \\ &= \mathcal{N}(\boldsymbol{y}_k|\boldsymbol{C}\bar{\boldsymbol{x}}_k, \boldsymbol{R}_{kf})\mathcal{N}(\boldsymbol{x}_k|\hat{\bar{\boldsymbol{x}}}_k, \hat{\boldsymbol{P}}_k) \\ &= \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}_k \\ \boldsymbol{y}_k\end{bmatrix} \middle| \begin{bmatrix}\hat{\bar{\boldsymbol{x}}}_k \\ \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k\end{bmatrix}, \begin{bmatrix}\hat{\boldsymbol{P}}_k & \hat{\boldsymbol{P}}_k\boldsymbol{C}^T \\ \boldsymbol{C}\hat{\boldsymbol{P}}_k & \boldsymbol{C}\hat{\boldsymbol{P}}_k\boldsymbol{C}^T + \boldsymbol{R}_{kf}\end{bmatrix}\right). \end{aligned} \tag{4.46}$$

Therefore, using Lemma A.2, the filtering distribution can be obtained as:

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k, \boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_{k-1}) = \mathcal{N}(\boldsymbol{x}_k|\bar{\boldsymbol{x}}_k, \boldsymbol{P}_k), \tag{4.47}$$

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

$$\begin{cases} \bar{\boldsymbol{x}}_k = \hat{\bar{\boldsymbol{x}}}_k + \hat{\boldsymbol{P}}_k\boldsymbol{C}^T(\boldsymbol{R}_{kf} + \boldsymbol{C}\hat{\boldsymbol{P}}_k\boldsymbol{C}^T)^{-1}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k) \\ \boldsymbol{P}_k = \hat{\boldsymbol{P}}_k - \hat{\boldsymbol{P}}_k\boldsymbol{C}^T(\boldsymbol{R}_{kf} + \boldsymbol{C}\hat{\boldsymbol{P}}_k\boldsymbol{C}^T)^{-1}\boldsymbol{C}\hat{\boldsymbol{P}}_k \end{cases}. \tag{4.48}$$

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

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

$$\bar{\boldsymbol{x}}_k = \hat{\bar{\boldsymbol{x}}}_k + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k), \tag{4.49}$$

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 $\boldsymbol{K}_k$:

$$\begin{aligned} \bar{\boldsymbol{x}}_k &= \boldsymbol{A}\bar{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\hat{\bar{\boldsymbol{x}}}_k) \\ &= (\boldsymbol{A} - \boldsymbol{K}_k\boldsymbol{C})\,\bar{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{K}_k\boldsymbol{y}_k \end{aligned}. \tag{4.50}$$

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.

Controller-Estimator Duality

In addition to that, it can be shown that the recursive steps that solves the filtering problem are also responsible for iteratively minimizing the trace of the covariance matrix $\boldsymbol{P}_k$, which is given as:

$$\boldsymbol{P}_k = \mathbb{E}\left[(\boldsymbol{x}_k - \bar{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \bar{\boldsymbol{x}}_k)^T\right]. \tag{4.51}$$

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 variable* $\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_{kf})$ *and measurement noise variable* $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_{kf})$*, 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* $\bar{\boldsymbol{x}}(t)$ *at time* t*, the error covariance:*

$$\boldsymbol{J}(\boldsymbol{x}, \bar{\boldsymbol{x}}, t) = \mathbb{E}\left\{[\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t)][\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t)]^T\right\} \tag{4.52}$$

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

$$\dot{\bar{\boldsymbol{x}}}(t) = \boldsymbol{A}\bar{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{K}_e(t)\left(\boldsymbol{y}(t) - \boldsymbol{C}\bar{\boldsymbol{x}}(t)\right), \tag{4.53}$$

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:*

$$\dot{\boldsymbol{P}}_e(t) = \boldsymbol{A}\boldsymbol{P}_e(t) + \boldsymbol{P}_e(t)\boldsymbol{A}^T - \boldsymbol{P}_e(t)\boldsymbol{C}^T\boldsymbol{R}_{kf}^{-1}\boldsymbol{C}\boldsymbol{P}_e(t) + \boldsymbol{Q}_{kf}, \tag{4.54}$$

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 [Crassidis and Junkins, 2011]. 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 an estimator considering $t_0 \to -\infty$, which, as in Theorem 4.4, results in a time-invariant filter with gain K_e.

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 an 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.

The control configuration that connects an 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. 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. A formal definition is given below.

Definition 4.8. (Linear Quadratic Gaussian) Consider a stochastic system in State-Space representation just as in Definition 4.5, rewritten below:

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{w}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{v}(t) \end{cases}, \tag{4.55}$$

whose estimated state-vector $\hat{\boldsymbol{x}}(t)$ is determined by a Kalman-Bucy filter and whose optimal 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:

$$\dot{\hat{\boldsymbol{x}}}(t) = [\boldsymbol{A} - \boldsymbol{K}_e(t)\boldsymbol{C} - \boldsymbol{B}\boldsymbol{K}(t)]\,\hat{\boldsymbol{x}}(t) + \boldsymbol{K}_e(t)\boldsymbol{y}(t), \tag{4.56}$$

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 solve the Riccati differential equations:

$$\begin{cases} -\dot{\boldsymbol{P}}(t) = \boldsymbol{A}^T\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{A} - \boldsymbol{P}(t)\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}(t) + \boldsymbol{Q} \\ \dot{\boldsymbol{P}}_e(t) = \boldsymbol{A}\boldsymbol{P}_e(t) + \boldsymbol{P}_e(t)\boldsymbol{A}^T - \boldsymbol{P}_e(t)\boldsymbol{C}^T\boldsymbol{R}_{kf}^{-1}\boldsymbol{C}\boldsymbol{P}_e(t) + \boldsymbol{Q}_{kf} \end{cases} \tag{4.57}$$

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

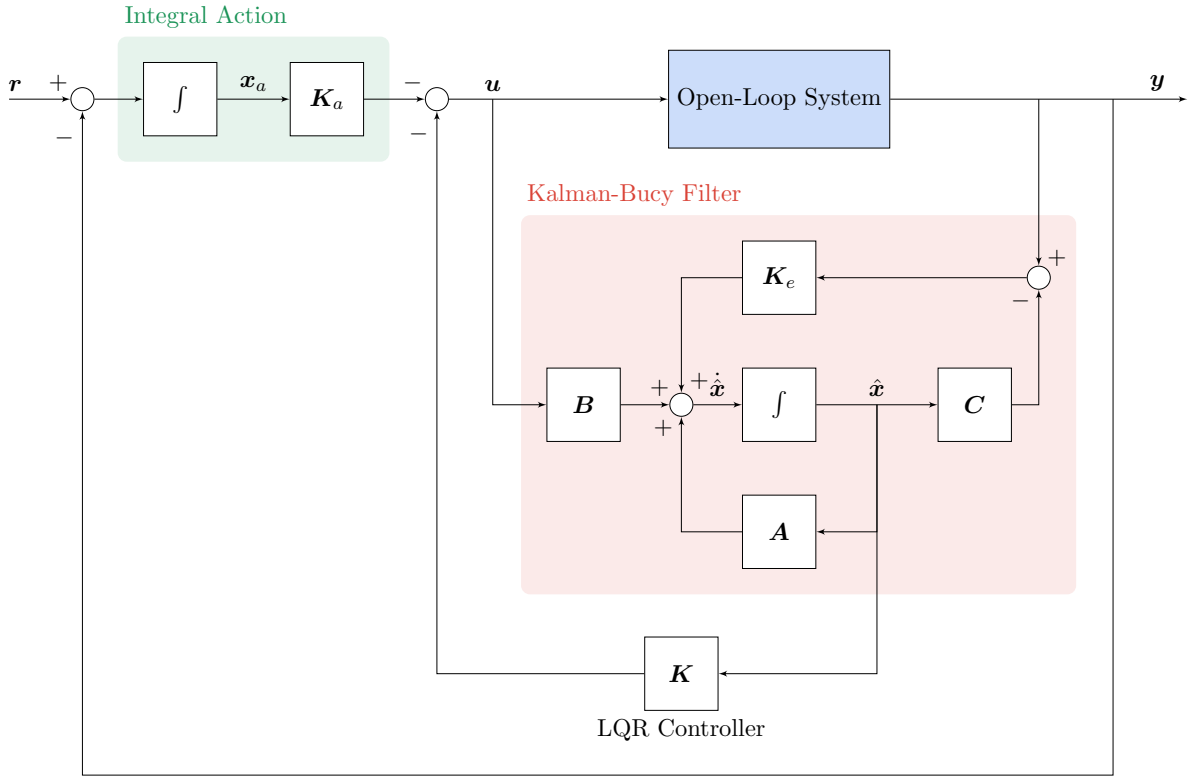


Figure 4.4: Block diagram of a Linear Quadratic Gaussian control configuration.

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:

$$\begin{cases} \begin{bmatrix} \dot{\hat{\boldsymbol{x}}}(t) \\ \dot{\hat{\boldsymbol{x}}}_a(t) \end{bmatrix} &= \begin{bmatrix} \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}(t) - \boldsymbol{K}_e(t)\boldsymbol{C} & -\boldsymbol{B}\boldsymbol{K}_a(t) \\ -\boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}(t) \\ \hat{\boldsymbol{x}}_a(t) \end{bmatrix} + \begin{bmatrix} \boldsymbol{K}_e(t) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{y}(t) \\ \boldsymbol{r}(t) \end{bmatrix} \\ \boldsymbol{y}(t) &= \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{x}_a(t) \end{bmatrix} \end{cases}. \tag{4.58}$$

Example 4.1. For the sake of illustration, consider the same system as in Example 3.1. 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:

$$\boldsymbol{Q}_{kf} = \begin{bmatrix} 0.001 & 0.001 \\ 0.001 & 0.001 \end{bmatrix} \quad \boldsymbol{R}_{kf} = 0.001. \tag{4.59}$$

A LQG controllers for reference tracking is obtained by solving the controller Ricatti differential equation for weights $\boldsymbol{Q} = \texttt{diag}(10, 10, 1)$ and $\boldsymbol{R} = 0.1$, considering the 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 from Example 4.1 are shown in Fig. 4.5. Notice that, since the integral action is now calculated over the observations, which are noisy, the controller produced a noisy input signal. The reference tracking was still achieved, but with very oscillatory behavior. A solution to this problem could be using the integral action as a feedback over an output signal obtained from the estimated states, which are probably smoother. Another solution would be to use a *smoothing filter* directly to the input signal to avoid this oscillatory action.

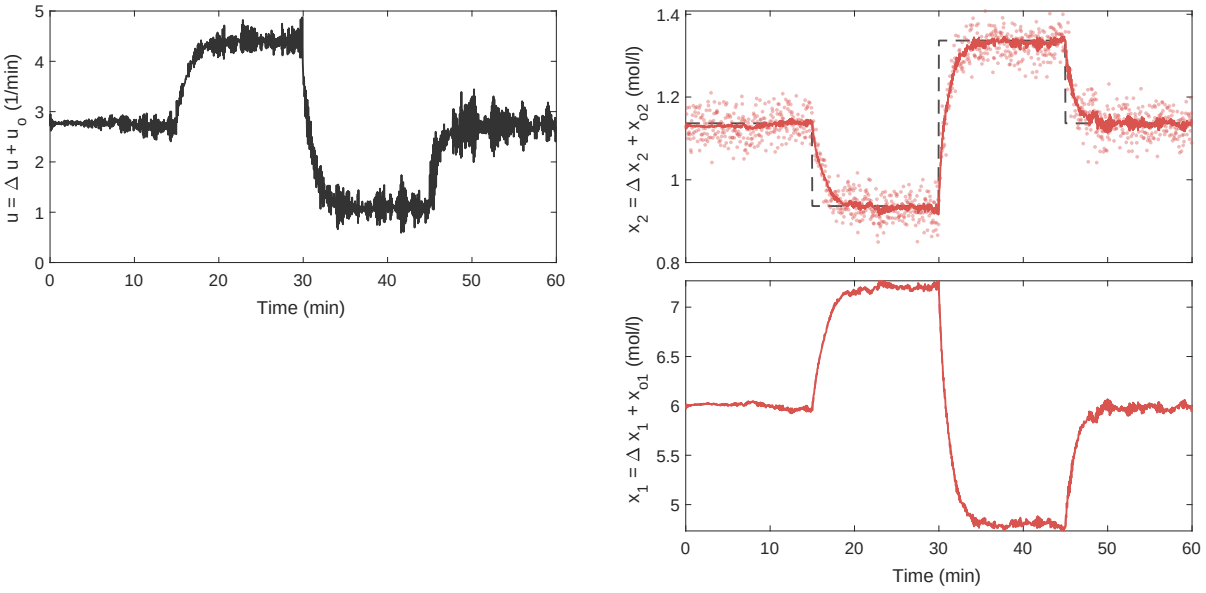


Figure 4.5: Simulation of the LQG controllers showing the input signal (left) and correspondent actual state responses (right). The dots denotes represents the measurements from the real system.

4.5 Stability and Robustness Analysis

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

Stability

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 an 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:

$$J(x, u, t_0) = \int_{t_0}^{\infty} u^2(t)dt. \tag{4.60}$$

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 under which the optimization process may result 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 generality, that $\boldsymbol{Q} = \boldsymbol{C}^T\boldsymbol{C}$, which is still positive definite. In this case $\boldsymbol{C} = \boldsymbol{Q}^{1/2}$, given that $\boldsymbol{Q}^{1/2}\boldsymbol{Q}^{1/2} = \boldsymbol{C}$, which is a nonsingular matrix and the observability assumption holds. Furthermore, consider the LQR cost function given an optimal input trajectory $\boldsymbol{u}^* : \mathbb{R} \to \mathbb{R}^r$:

$$J(\boldsymbol{x}, \boldsymbol{u}^*, t_0) = \int_{t_0}^{\infty} \left(\boldsymbol{x}^T(\boldsymbol{C}^T\boldsymbol{C})\boldsymbol{x} + (\boldsymbol{u}^*)^T\boldsymbol{R}\boldsymbol{u}^*\right) dt = \int_{t_0}^{\infty} \left(\boldsymbol{y}^T\boldsymbol{y} + (\boldsymbol{u}^*)^T\boldsymbol{R}\boldsymbol{u}^*\right) dt. \tag{4.61}$$

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 corresponding 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:

$$\boldsymbol{V}^*(\boldsymbol{x}, t_0) \leq \boldsymbol{x}^T(t_0)\boldsymbol{P}\boldsymbol{x}(t_0), \tag{4.62}$$

and, therefore, (4.61) 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 defined through the assignment of the eigenvalues, the LQR controller guarantees a stable closed-loop system directly, as long as the system is observable, even when the open-loop system is unstable. Since the BIBO stability (Definition 2.7) 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.

Robustness

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 $PM \geq 60°$ [Anderson and Moore, 1990]. 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 an unit circle centered in $s = 0$ makes with the point $s = -1$ is exactly 60°, which establishes the lower bound on the gain phase.

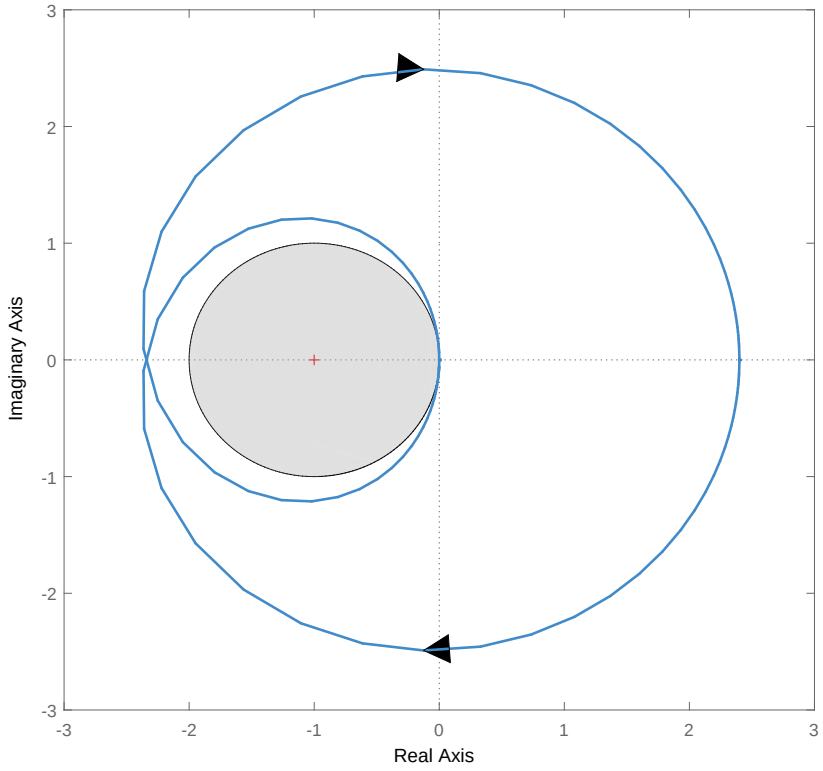


Figure 4.6: Nyquist diagram of a single state system with an open-loop unstable pole but which is closed-loop stable. The gain calculated by a LQR stabilizing controller never crosses the unit circle centered at $s = -1 + j0$.

Of course, there is no physical support of saying that a controller device exhibits an 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 [Doyle, 1978] that an optimal controller-estimator configuration, i.e., the Linear Quadratic Gaussian, has no guaranteed margins.

Chapter 5

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, 2018] software as the framework to compute such simulations. The results obtained are presented and discussed in the next chapter.

5.1 Non-Isothermal Continuous Stirred Tank

The system used for the experiments was the non-isothermal Continuous Stirred Tank Reactor (CSTR) presented by [Klatt and Engell, 1998]. This system is a realization of a class of processes that characterize a wide range of industrial applications, hence being considered a classical benchmark for reactor systems. In particular, the system proposed for the experiment is 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. A schematic of this system is shown in Fig. 5.1.

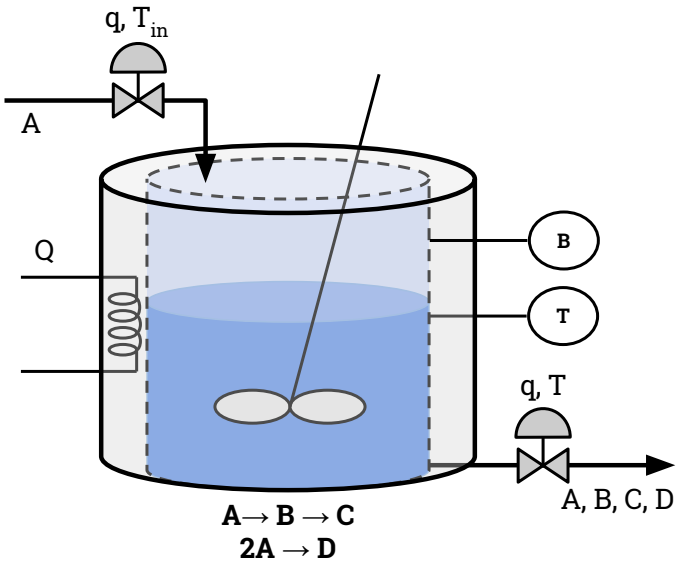


Figure 5.1: The non-isothermal continuous stirred tank reactor proposed.

This system is comprised by a tank containing a dilute solution of cyclopentadiene ($\mathbf{C}_5\mathbf{H}_6$), which suffers reactions together with water molecules and together with the side-products of these reactions. The reaction scheme follows the same Van de Vusse scheme explored earlier in

this report:

$$\begin{cases} \overbrace{\mathbf{C_5H_6}}^{A} & \overset{K_1(T)}{\longrightarrow} & \overbrace{\mathbf{C_5H_7OH}}^{B} & \overset{K_2(T)}{\longrightarrow} & \overbrace{\mathbf{C_5H_8(OH)_2}}^{C} \\ \overbrace{\mathbf{2C_5H_6}}^{2A} & \overset{K_3(T)}{\longrightarrow} & \overbrace{\mathbf{C_10H_12}}^{D} \end{cases}. \tag{5.1}$$

Since the reaction is non-isothermal, the kinetics rates $K_1(T)$, $K_2(T)$ and $K_3(T)$ are function of the temperature inside the tank, and they are assumed to follow the Arrhenius equation:

$$K_i(T) = K_{i0} e^{\frac{-E_i/R}{T+273.15}} \tag{5.2}$$

In order to control the temperature, ensuring some safety constraints to the process, a coolant system is coupled to the tank reactor to perform heat exchange by conduction. This scenario also 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}.

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

$$\begin{cases} \dfrac{d\rho_A}{dt} &= q(\rho_{in}^{(A)} - \rho_A) - \left(K_1(T)\rho_A + K_3(T)\rho_A^2\right) \\ \dfrac{d\rho_B}{dt} &= -q\rho_B + K_1(T)\rho_A - K_2(T)\rho_B \\ \dfrac{dT}{dt} &= q(T_{in} - T) + \dfrac{k_W A_r}{\varrho C_p V_r}(T_C - T) \\ & \quad - \dfrac{1}{\varrho C_p}\left(K_1(T)\rho_A \Delta H_{AB} + K_2(T)\rho_B \Delta H_{BC} + K_1(T)\rho_A^2 \Delta H_{AC}\right) \\ \dfrac{dT_C}{dt} &= \dfrac{1}{m_K C_{pK}} Q + \dfrac{k_W A_r}{m_K C_{pK}}(T - T_C) \end{cases}. \tag{5.3}$$

The meaning and value of each system variable is shown in Table 5.1. Additionally, the real system has some constraints in respect to the manipulated variables, namely:

$$5hr^{-1} \leq F \leq 35h^{-1}; \quad -8500\frac{kJ}{h} \leq Q \leq 0\frac{kJ}{h}. \tag{5.4}$$

5.2 Simulation

The process and all the control structures were simulated using the MATLAB software. In the first part of the experiments, the dynamics of the process were analyzed in respect to the mathematical structure of the models and through the simulation results.

In respect to the control, the second part of the experiments focused on the design of controllers using the optimal control techniques. The controllers were developed and analyzed in the same order that they were presented in this document:

1. Linear Quadratic Regulator, assuming a nonzero initial state and perfect measurement of the states;

2. Linear Quadratic Regulator with Integral Action, for a non-constant reference signals and assuming perfect measurement of the states;

3. Linear Quadratic Gaussian, assuming a nonzero initial state and reconstruction of the state-vector through observations from a stochastic linear system;

System Variable	Symbol	Value	Unit
Concentration A in the reactor solution	ρ_A	–	mol/l
Concentration A in the reactor solution	ρ_B	–	mol/l
Temperature inside the reactor	T	–	^{o}C
Temperature of the coolant jacket	T_C	–	^{o}C
Inflow concentration of A	$\rho_{in}^{(A)}$	5.1	mol/l
Inflow liquid temperature	T_{in}	130	^{o}C
Nominal kinetic rate for $A \to B$	K_{10}	$(1.287) \times 10^{12}$	h^{-1}
Nominal kinetic rate for $B \to D$	K_{20}	$(1.287) \times 10^{12}$	h^{-1}
Nominal kinetic rate for $2A \to D$	K_{30}	$(9.043) \times 10^{9}$	l mol^{-1}h
Activation energy for $A \to B$	E_1/R	9758.3	-
Activation energy for $B \to C$	E_2/R	9758.3	-
Activation energy for $2A \to D$	E_3/R	8560.0	-
Enthalpy for $A \to B$	ΔH_{AB}	4.2	kJ/mol
Enthalpy for $B \to C$	ΔH_{BC}	-11.0	kJ/mol
Enthalpy for $2A \to D$	ΔH_{AD}	-41.85	kJ/mol
Total density of the reactor solution	ϱ	$(0.9342) \times 10^{-4}$	kg/l
Heat capacity of the solution	C_p	3.01	$kJ\ kg^{-1}K$
Heat capacity of the coolant	C_{pK}	2.0	$kJ\ kg^{-1}K^{-1}$
Surface area of the tank reactor	A_R	0.215	m^2
Volume of the tank reactor	V_R	10.01	m^3
Coolant mass	m_K	5.0	kg
Heat transfer coefficient	k_W	4032.0	$kJ\ h^{-1}m^2K$

Table 5.1: Model parameters and variables nomenclature.

4. Linear Quadratic Gaussian with Integral Action, for a non-constant reference and assuming a reconstruction of the state-vector through observations from a stochastic linear system;

The simulation results are mainly analyzed through graphical visualizations of the data obtained. The experiments were focused on validating the modeling and control procedures for the system presented, instead of comparing and ranking the controllers based on the most performing one. For the sake of completeness, however, some metrics are used to evaluate each controller performance and relate them to the visualizations. The metrics used were the *Integral of Squared Error* (ISE), *Integral of Absolute Error* (IAE), *Integral of Time-Weighted Squared Error* (ITSE) and *Integral of Time-Weighted Absolute Error* (ITAE), which are given as:

$$\begin{aligned} \text{IAE} &= \int_{t_0}^{T} |\boldsymbol{r}(\tau) - \boldsymbol{y}(\tau)| d\tau; \qquad \text{ITAE} = \int_{t_0}^{T} \tau |\boldsymbol{r}(\tau) - \boldsymbol{y}(\tau)| d\tau; \\ \text{ISE} &= \int_{t_0}^{T} \left[\boldsymbol{r}(\tau) - \boldsymbol{y}(\tau)\right]^2 d\tau; \qquad \text{ITSE} = \int_{t_0}^{T} \tau \left[\boldsymbol{r}(\tau) - \boldsymbol{y}(\tau)\right]^2 d\tau. \end{aligned} \tag{5.5}$$

Chapter 6

Results and Discussion

This chapter presents the results of the experiments described in the last chapter. The sections starts by presenting the simulation of the dynamical model and analyzing its properties. After that, several optimal controllers are formulated and simulated in closed-loop conditions, both for regulation and tracking. The control simulations are grouped accordingly to the control objective that they impose to the system.

6.1 Dynamical Analysis

Nonlinear model simulation

The behavior of the process is accessed directly by simulating the nonlinear model presented at (5.3). The main goal is to observe how the system responds to variations in each of its manipulated variables. In the first simulation, shown in Fig. 6.1, the process is started at the steady-state condition, and then a pulse is applied to the liquid inflow to the tank.

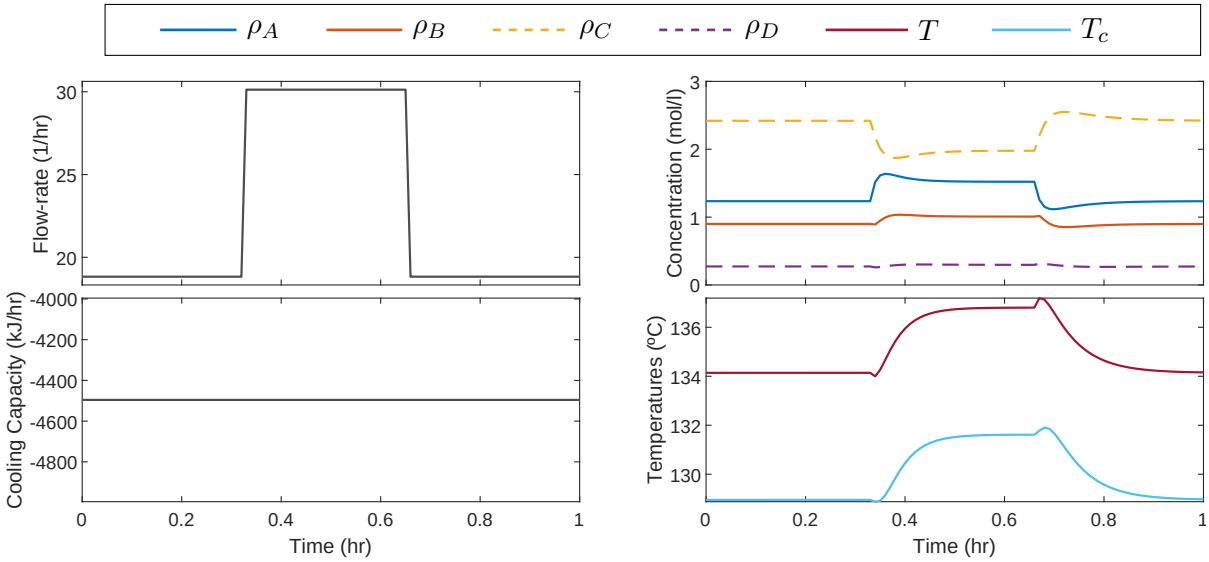


Figure 6.1: Non-linear simulation showing the manipulated variables (left) and correspondent response of the system variables (right) for a pulse change in the input flow-rate.

The simulation shows that both compounds A and B grows together when there is an increase in the input flow-rate, in which case the first exhibits a greater response than the other. Of course, this conclusion is direct from the fact that the input carries concentrations of this very same chemical. In respect to the temperatures, notice that, since the process is non-isothermal,

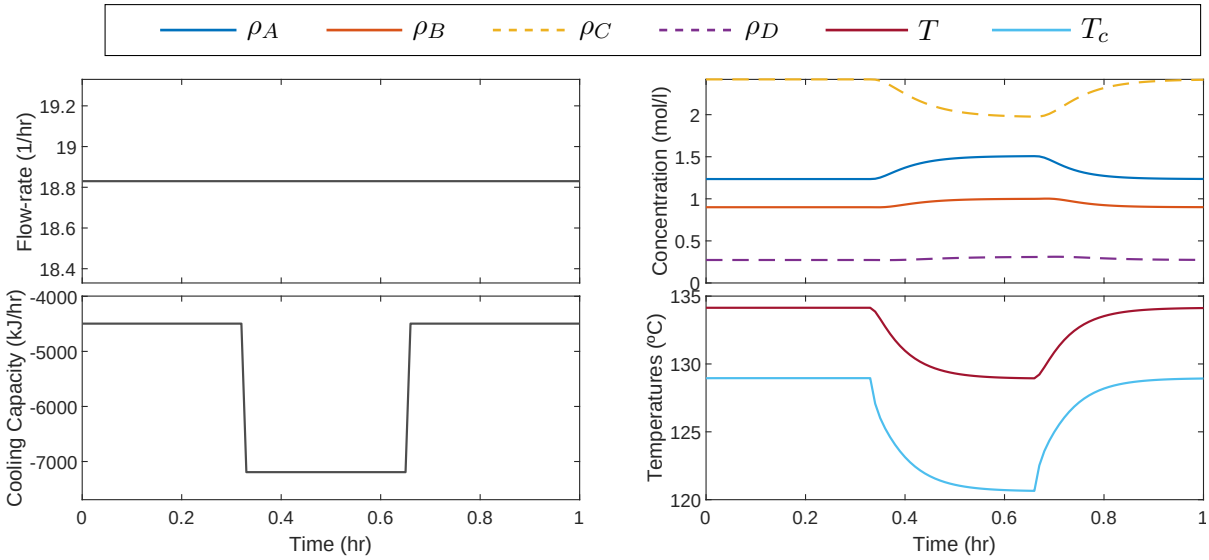


Figure 6.2: Non-linear simulation showing the manipulated variables (left) and correspondent response of the system variables (right) for a pulse change in the cooling capacity.

the temperatures raised together with the increase in the reagents concentrations. Given that the temperature of the liquid inflow is actually lower than the tank temperature at the steady-state, it is possible to relate the increase in the temperature directly to the occurrence of the chemical reactions. This fact characterizes the process as an exothermic process, as it can be also noted from the first-principles model.

The second simulation was performed similarly, but applying a negative pulse to the jacket cooling capacity. The result is shown in Fig. 6.2. As expected, the change in the cooling capacity affects the temperatures inside the reactor tank and inside the cooling jacket. The visualization also shows that the change in the temperature slightly changed the concentration of the substances inside the reactor. Therefore, it is possible to consider an indirect effect of this manipulated variable to these system variables.

Now, the goal was to visualize if it is possible to increase the concentration of the chemicals through the reactions, while maintaining the tank temperature close to the steady-state value. For this reason, the last simulation considers a pulse to increase the input flow-rate while another pulse increases the magnitude of the cooling capacity. The result of this simulation is shown at Fig. 6.3. Note that the tank temperature remained restricted to values closer to the steady-state point, whereas the concentrations were increased by the flow-rate input. Thus, the possibility of increasing the production of a chemical without breaking safety constraints holds.

Linearized model realization and simulation

The discussion throughout the document focused on control design methods for linear systems. For this reason, the nonlinear model can not directly be used to develop the controllers, and a linearization (Theorem 2.2) of this model is required. First of all, consider the following choice for the state, input and output variables:

$$\boldsymbol{x}(t) = \begin{bmatrix} \rho_A(t) \\ \rho_B(t) \\ T(t) \\ T_c(t) \end{bmatrix}; \quad \boldsymbol{u}(t) = \begin{bmatrix} q(t) \\ Q(t) \end{bmatrix}; \quad \boldsymbol{y}(t) = \begin{bmatrix} \rho_B(t) \\ T(t) \end{bmatrix}. \tag{6.1}$$

By evaluating the derivative of each differential state equation in relation to each state

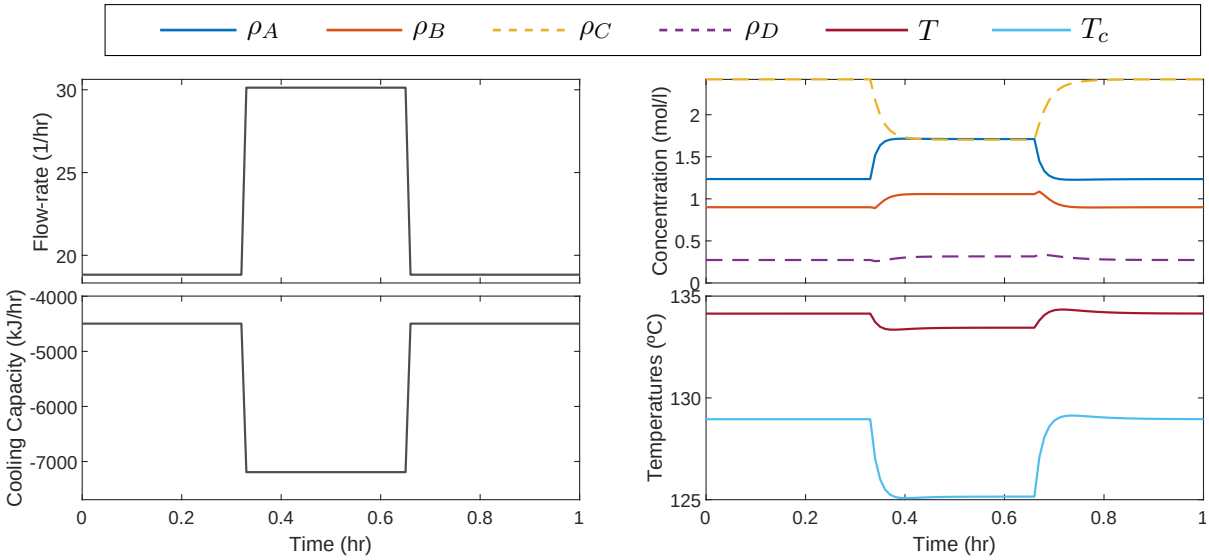


Figure 6.3: Non-linear simulation showing the manipulated variables (left) and correspondent response of the system variables (right) for a change in the input flow-rate and cooling capacity.

variable, the obtained Jacobian is

$$\boldsymbol{A} = \begin{bmatrix} -u_{1o} - K_1(x_{3o}) - 2K_3(x_{3o})x_{1o} & 0 & \dfrac{K_1(x_{3o})}{dx_{3o}}x_{1e} + \dfrac{K_3(x_{3o})}{dx_{3o}}x_{1o}^2 & 0 \\ K_1(x_{3o}) & -u_{1o} - K_2(x_{3o}) & \dfrac{K_1(x_{3o})}{dx_{3o}}x_{1e}\dfrac{K_2(x_{3o})}{dx_{3o}}x_{2e} & 0 \\ \dfrac{\partial H(x_{1o},x_{2o},x_{3o})}{\partial x_{1o}} & \dfrac{\partial H(x_{1o},x_{2o},x_{3o})}{\partial x_{2o}} & -u_{1e} - \alpha + \dfrac{\partial H(x_{1o},x_{2o},x_{3o})}{\partial x_{3o}} & \alpha \\ 0 & 0 & \beta & -\beta \end{bmatrix}, \tag{6.2}$$

where the parameters α and β are given by

$$\alpha = \frac{k_W A_r}{\varrho C_p V_r}; \quad \beta = \frac{k_W A_r}{m_K C_{pK}}, \tag{6.3}$$

and the derivatives are calculated as

$$\begin{aligned} \frac{dK_i}{dx_{3o}} &= \frac{E_1/R}{(x_{3o}+273.15)^2} K_{i0} e^{\frac{-E_i/R}{x_{3o}+273.15}}; \\ \frac{\partial H}{\partial x_{1o}} &= \frac{-1}{m_K C_{pK}} \left(K_1(x_{3o})\Delta H_{AB} + 2K_3(x_{3o})x_{1o}\Delta H_{AD}\right); \\ \frac{\partial H}{\partial x_{2o}} &= \frac{-1}{m_K C_{pK}} \left(K_2(x_{3o})\Delta H_{BC}\right); \\ \frac{\partial H}{\partial x_{3o}} &= \frac{-1}{m_K C_{pK}} \left(\frac{dK_1}{dx_{3o}}x_{1o}\Delta H_{AB} + \frac{dK_2}{dx_{3o}}x_{2o}\Delta H_{BC} + \frac{dK_3}{dx_{3o}}x_{1o}^2\Delta H_{AC}\right). \end{aligned} \tag{6.4}$$

Similarly, the derivative of each differential equation in relation to each input variable is

evaluated, which results in the simpler Jacobian

$$\boldsymbol{B} = \begin{bmatrix} \rho_{in}^{(A)} - x_{1o} & 0 \\ -x_{2o} & 0 \\ T_{in} - x_{3o} & 0 \\ 0 & \dfrac{1}{m_K C_{pK}} \end{bmatrix}. \tag{6.5}$$

Regarding the matrices $\boldsymbol{C}$ and $\boldsymbol{D}$, recall that only the variables ρ_B and T are measured in the process. The measurement, however, is assumed to be perfect, so that these matrices can be defined as

$$\boldsymbol{C} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}; \quad \boldsymbol{D} = \boldsymbol{0}. \tag{6.6}$$

The linear model presented in (6.2) is an approximation of the nonlinear model for any point in the state space, given that it is a steady-state point. To obtain feasible linearization points, the differential equation system is evaluated for all points where $\dot{\rho_A} = \dot{\rho_B} = \dot{T} = \dot{T}_c = 0$. To solve this system, the input variables are selected to be the interval defined by the physical variable restrictions. The result is shown in Fig. 6.4 for the two measured variables. In this work, it is assumed the steady-state point given by $\boldsymbol{x}_o = [1.235, 0.90, 134.14, 128.95]^T$ and $\boldsymbol{x}_o = [18.83, -4495.7]^T$, indicated in the figure by a white cross.

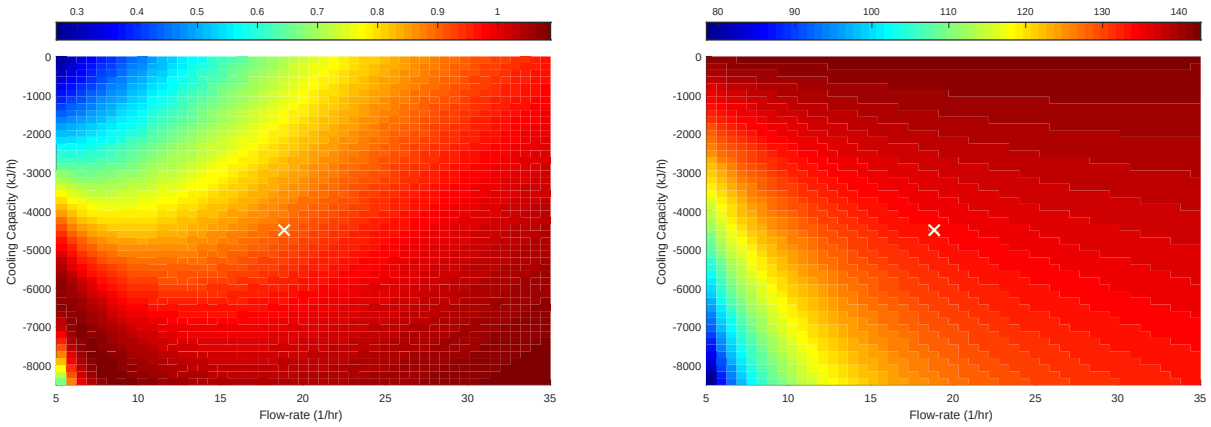


Figure 6.4: Visualization of the steady-state values for the linearized model.

Using the steady-state values just mentioned, a realization for the linearized model is obtained:

$$\begin{cases} \boldsymbol{A} = \begin{bmatrix} -86.0962 & 0 & -4.2010 & 0 \\ 50.6146 & -69.4446 & 0.9958 & 0 \\ 172.2263 & 197.9985 & -36.7597 & 30.7977 \\ 0 & 0 & 86.6880 & -86.6880 \end{bmatrix} & \boldsymbol{B} = \begin{bmatrix} 3.865 & 0 \\ -0.900 & 0 \\ -4.140 & 0 \\ 0 & 0.100 \end{bmatrix} \\ \boldsymbol{C} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} & \boldsymbol{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \end{cases}. \tag{6.7}$$

In order to demonstrate that the linear model is a valid approximation of the nonlinear model, both systems were simulated in parallel for the same input signal. The simulation, as shown in Fig. 6.5, starts at the steady-state values and then apply a sinusoidal change in the input to drag the system away from this initial point. As expected, the results shows that the linearized model is an accurate representation of the system, except when it is far from the steady-state values.

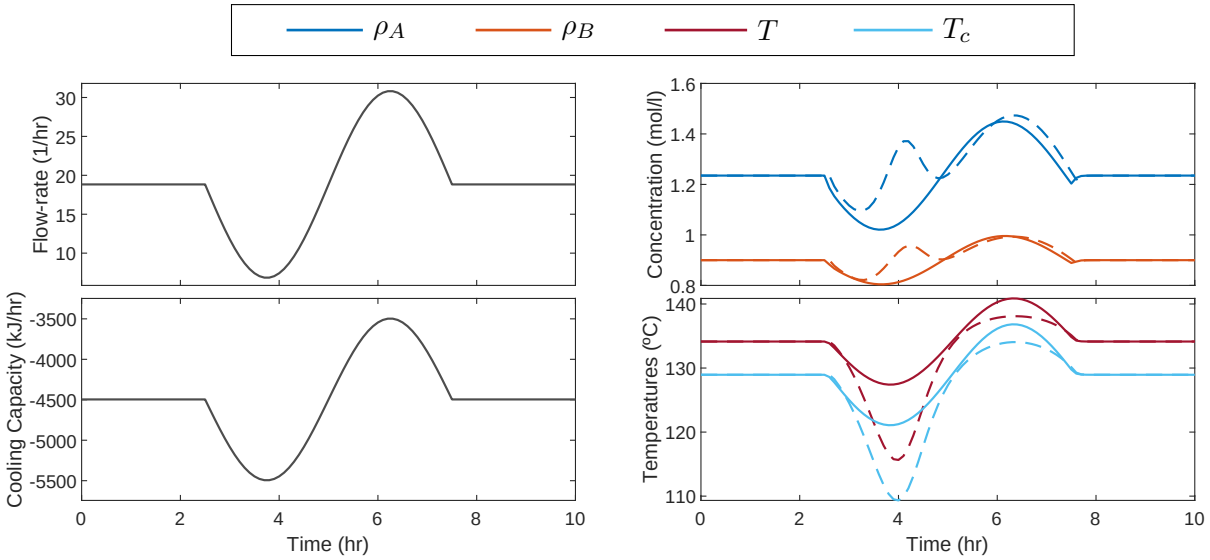


Figure 6.5: Simulation of both the nonlinear and linearized State-Space models. The nonlinear response is represented by the dashed lines.

Linearized model properties

Considering a State-Space realization, it is possible to access the system properties. First of all, the BIBO stability of the system was determined through the poles location in the complex plane. In the case of (6.7), the eigenvalues of $\boldsymbol{A}$ were calculated as $\boldsymbol{\lambda} = [-16.79, -54.84, -86.33, -121.01]$, which are all in right-hand side of the plane, i.e., $\text{Re}[\lambda_i]$, $\forall i \in [1, 4]$. It is possible to conclude that, in the light of this model, the system is stable.

The next property to be accessed consists in the controllability of the system. Using the definition, the controllability matrix of this model was calculated as:

$$\mathcal{C} = \begin{bmatrix} 3.8650 & 0 & -315.3696 & 0 & 24465 & -12.9 & 2271 \\ -0.9000 & 0 & 254.0027 & 0 & -32964 & 3.1 & -1250 \\ -4.1400 & 0 & 639.6409 & 3.0798 & -38589 & -380.2 & 43720 \\ 0 & 0.1000 & -358.8883 & -8.6688 & 86561 & 1018.5 & -121250 \end{bmatrix}. \tag{6.8}$$

This matrix has a row rank equal to 4, which is exactly the number of state variables. Therefore, it is concluded that this system is indeed controllable, meaning that it is possible to drive the system to any position in the space using a specific input signal. It is worth mentioning that this matrix exhibits some entries with high values, which can become numerically unstable in the case of augmenting the system.

Moreover, the last property of the system to be accessed concerns the observability of the system. Similarly to the controllability case, the observability matrix was calculated as:

$$\mathcal{O} = \begin{bmatrix} 0 & 1.0000 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 \\ 50.6146 & -69.4446 & 0.9958 & 0 \\ 172.2263 & 197.9985 & -36.7597 & 30.7977 \\ -7701.1 & 5019.7 & -318.4 & 30.7 \\ -11137 & -21028 & 3495 & -3802 \\ 862270 & -411630 & 51710 & -1246 \\ 496400 & 2152200 & -432200 & 437200 \end{bmatrix}. \tag{6.9}$$

This matrix has a column rank equal to 4, which, again, is exactly the number of state variables. Thus, it is possible to conclude that the system is observable, meaning that the information of any state can be reconstructed given only the outputs.

6.2 Optimal Control

Regulation

The results for the controllers arrangements are now presented. The first controller simulated was the Linear Quadratic Regulator (LQR), as shown in Fig. 6.6. The simulation started with initial condition $x_1(0) = x_2(0) = 0$ (or, equivalently, $\Delta x_1(0) = -x_{1o}$ and $\Delta x_2(0) = -x_{2o}$) while maintaining the states x_3 and x_4 at the steady-state value. The goal of this simulation was to represent the scenario where the solution in the tank does not actually contain any concentration of the reagents or products. This scenario, thus, requires that the controller "starts" the process by providing the flow of reagent necessary to achieve the desirable steady-state concentration of the controlled product. The weights used, regarding the LQR cost function, were:

$$\boldsymbol{Q} = \begin{bmatrix} 0.25 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \quad ; \quad \boldsymbol{R} = \begin{bmatrix} 0.25 & 0 \\ 0 & 1.1e-5 \end{bmatrix}. \tag{6.10}$$

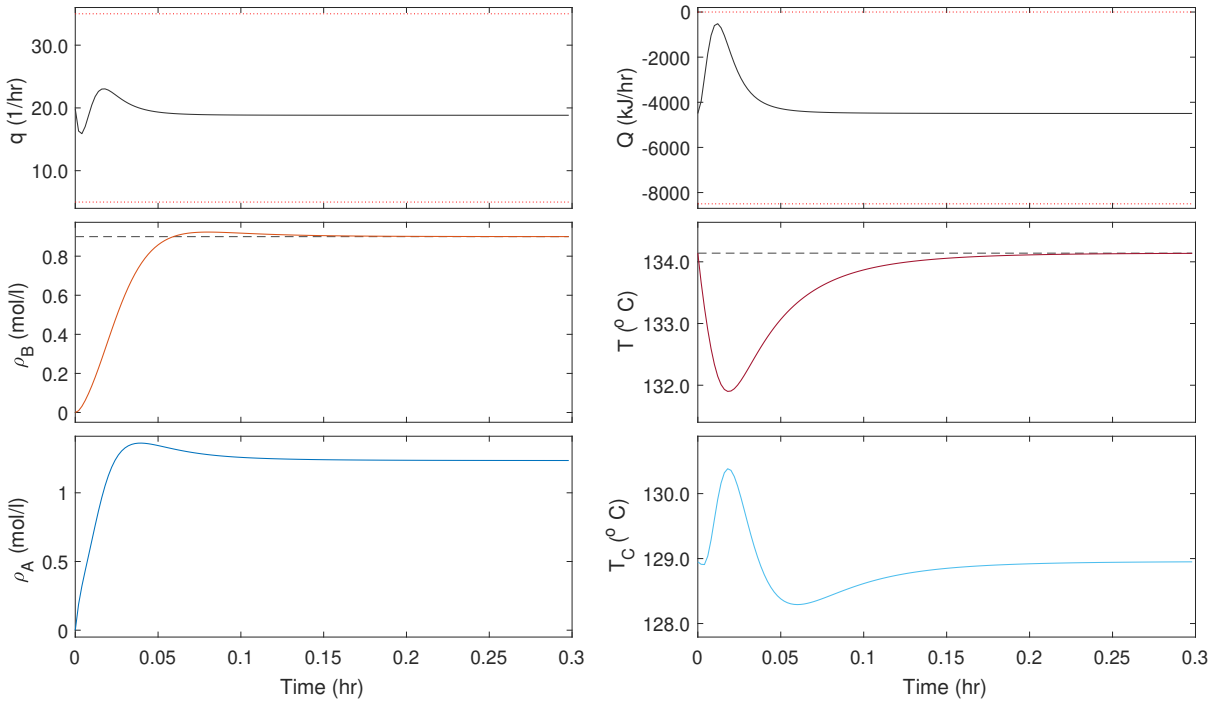


Figure 6.6: Simulation of the LQR_1 controller for "starting" the tank reactor to the steady-state point. The reference is indicated by the black dashed lines, and the manipulated variables restrictions are indicated by the red dotted lines.

Notice that the resulting reactions will change the temperature inside the tank, requiring also that the controller manipulates the cooling capacity in order to control the reactor temperature. As expected, the controller momentarily increases the input flow-rate in order to accelerate the production of compound B, returning to the steady-state value to achieve the desired reference. Notice that the input flow-rate does not grow further in the first minutes in order to avoid overshoots, which occurs in the results presented but with small magnitude. In respect to the reactor temperature, the controller was able to respond to the sudden change by decreasing the absolute value of the cooling capacity, causing the coolant to increase its temperature and, consequently, returns the reactor temperature to the reference. Therefore, the simulation shows that the proposed optimal controller is able to drive the process to a desired constant reference in an optimal manner, not causing overshoots and/or oscillations, in a small amount of time.

Further testing the capabilities of this controller, the same experiment was considered, but while subjecting the system to disturbances in some system variables. Specifically, disturbances were applied to the inflow concentration and temperature, $\rho_{in}^{(A)}$ and T_{in}, respectively, to represent the scenario where the fluid entering the tank is perturbed by previous processes or by instrumentation limitations. To simulate these cases, pulse signals of values $w_1 = \pm 0.5$ and $w_2 = \pm 2$ were applied to the nominal value of the inflow concentration and temperature, respectively. Each pulse starts at a time value that is an odd multiple of $\lfloor T/4 \rfloor$ and lasts for $0.02T$ hours, where T is the the total size of the control horizon. The results are shown in Fig. 6.7.

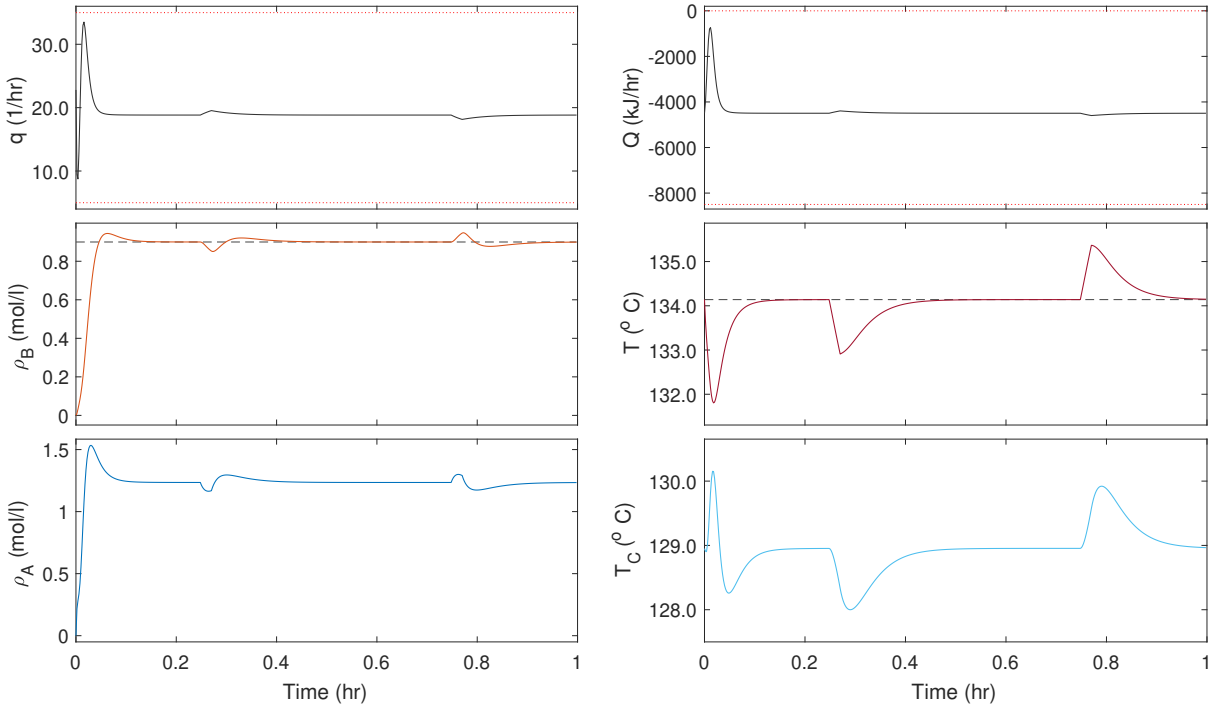


Figure 6.7: Simulation of the LQR_2 controller for "starting" the tank reactor to the steady-state point, but subjected to two disturbances pulses in the system variables.

First of all, it was observed that the control horizon had to be increased in order to allow the feedback controller to perceive the disturbance and corrects the system response accordingly. As a direct result of the larger horizon, the controller performed more aggressively at the first minutes. Despite of this, the input signals did not violated the defined constraints for the manipulated variables. It was observed that the regulator is capable of react to the disturbances and provide a corrective action to return to the steady-state point.

Until now, the experiments were done assuming that the optimal controller had direct access to the real system states. In a more realistic scenario, the states can only be reconstructed through noisy measurements. A Linear Quadratic Gaussian (LQG) controller is proposed to deal with this limitation, considering the same control objectives of the previous experiments and using the same weighting matrices as in (6.10). In the experiments it was assumed a process noise $\boldsymbol{w} : \mathbb{R} \to \mathbb{R}^4$ and measurement noise $\boldsymbol{z} : \mathbb{R} \to \mathbb{R}^2$ described by the Gaussian distributions:

$$\boldsymbol{w}(t) \sim \mathcal{N}\left(\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} \middle| \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \right) \quad \boldsymbol{z}(t) \sim \mathcal{N}\left(\begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} \middle| \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.0001 & 0 \\ 0 & 0.1 \end{bmatrix} \right) \tag{6.11}$$

The covariance matrix of the noise $\boldsymbol{w}(t)$ tries to represent the uncertainty of the non-linear model behavior in respect to the linearized model, while the covariance matrix of $\boldsymbol{z}(t)$ aims

to represent realistic limitation of measurement devices in respect to these variables. It is assumed that the Kalman-Bucy estimator can estimate these covariances fairly accurately, before optimizing each estimator gain $\boldsymbol{K}_e(t)$. The simulation results, for the scenario of starting the process, is shown in Fig. 6.8.

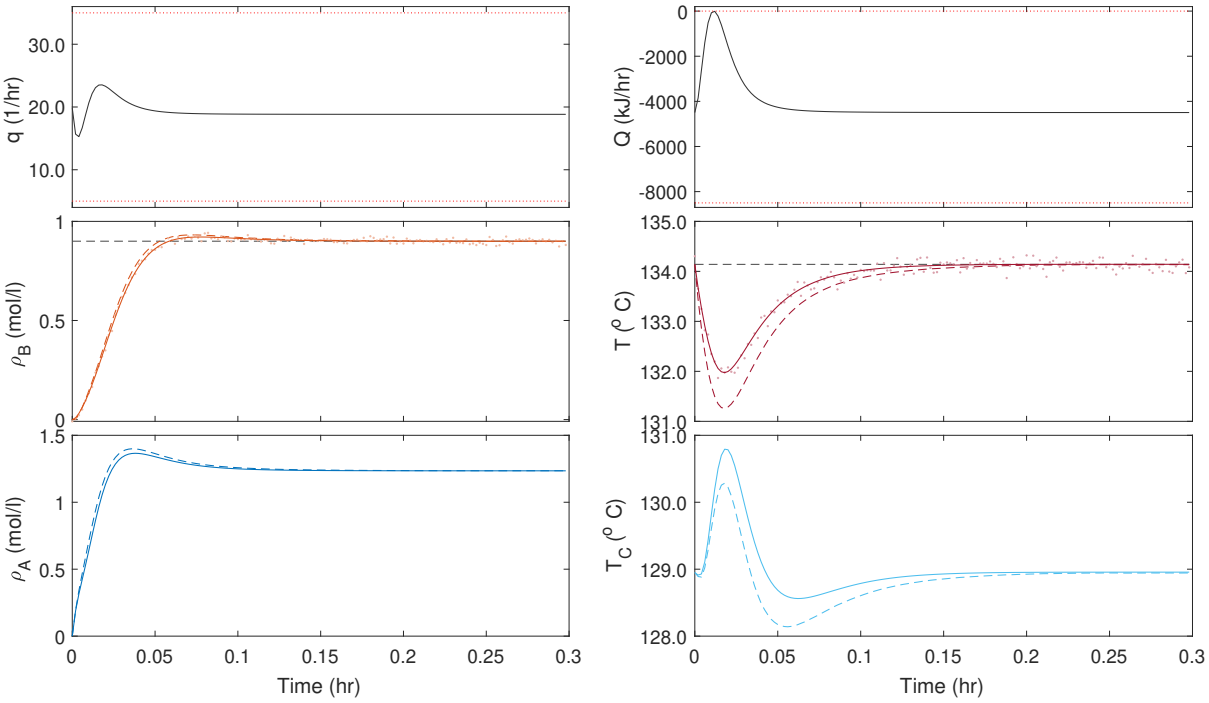


Figure 6.8: Simulation of the LQG_1 controller for "starting" the tank reactor to the steady-state point. The dots represents the measurements of the output variables, and the colored dashed lines represents the state estimations.

The estimation of the covariance matrices gave the estimator enough information to accurately reconstruct the state vector from the observations. As it can be seen in the visualizations, this control-estimation operation is very similar to the one shown in Fig. 6.6, when the states were directly known. Thus, the LQG controller proved to be optimal for this more realistic situation. Furthermore, the same controller was also evaluated for the case where disturbances in the system variables are present. In this experiment, the same disturbance signal used in the LQR simulation is provided to the system. The results are shown in Fig. 6.9. The system responses are very similar. However, the corrective actions for the disturbances were smoother and are barely visible in the scale of the visualization. Finally, a summary of the results in this subsection are displayed at Table 6.1.

Controller	**ISE**	**IAE**	**ITSE**	**ITAE**
LQR_1	$\begin{bmatrix} 7.2678 \\ 82.9199 \end{bmatrix}$	$\begin{bmatrix} 12.2719 \\ 60.6988 \end{bmatrix}$	$\begin{bmatrix} 0.0015 \\ 0.1045 \end{bmatrix}$	$\begin{bmatrix} 0.2592 \\ 2.8347 \end{bmatrix}$
LQG_1	$\begin{bmatrix} 7.2765 \\ 67.6334 \end{bmatrix}$	$\begin{bmatrix} 12.0436 \\ 48.4748 \end{bmatrix}$	$\begin{bmatrix} 0.0015 \\ 0.0596 \end{bmatrix}$	$\begin{bmatrix} 0.2299 \\ 1.7535 \end{bmatrix}$
LQR_2	$\begin{bmatrix} 7.2713 \\ 137.0858 \end{bmatrix}$	$\begin{bmatrix} 14.7600 \\ 134.5661 \end{bmatrix}$	$\begin{bmatrix} 0.0270 \\ 24.5280 \end{bmatrix}$	$\begin{bmatrix} 1.9614 \\ 51.1356 \end{bmatrix}$
LQG_2	$\begin{bmatrix} 7.3481 \\ 155.0745 \end{bmatrix}$	$\begin{bmatrix} 15.1579 \\ 160.7632 \end{bmatrix}$	$\begin{bmatrix} 0.0270 \\ 32.2785 \end{bmatrix}$	$\begin{bmatrix} 2.0193 \\ 65.5359 \end{bmatrix}$

Table 6.1: Comparison between the LQR and LQG controllers obtained. The pair (LQR_2, LQG_2) comprises the simulations in which the system was subjected to disturbances.

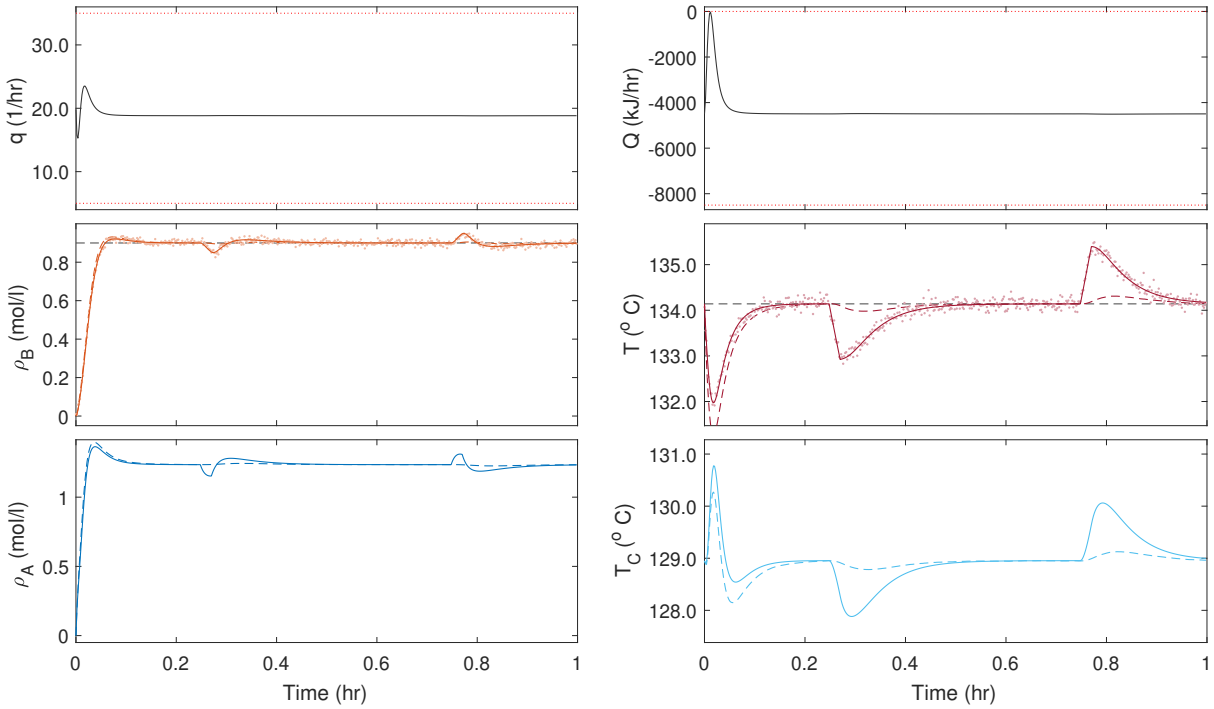


Figure 6.9: Simulation of the LQG_2 controller for "starting" the tank reactor to the steady-state point, but subjected to two disturbances pulses in the system variables.

Tracking

Now, the experiments for controllers that track non-constant references is presented. As in the previous subsection, the first controller simulated consists in the Linear Quadratic Regulator with Integral Action (LQRI), also known as Linear Quadratic Servo (LQ-Servo). The simulations starts with initial states equal to the steady-state values. The weights used were the same from (6.10), but including the weights for the augmented states:

$$\boldsymbol{Q} = \begin{bmatrix} 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 \end{bmatrix} \quad ; \quad \boldsymbol{R} = \begin{bmatrix} 0.16 & 0 \\ 0 & 1e-6 \end{bmatrix}. \tag{6.12}$$

The proposed scenario for this controller is to increase and then decrease the amount of compound B that is produced, while maintaining the reactor temperature at a constant value. For the concentration control, the reference signal is comprised by a sequence of step functions which varies from approximately 12% of the operation value, which is a realistic set-point change in such processes. The results of the simulation are shown in Fig. 6.10.

After the introduction of the integral action, two more weights were needed to be selected for the controller optimization. The MIMO case implies a trade-off between reaching the reference for the concentration and for the temperature, in the case that they do not represent a valid steady-state. Therefore, the choices of weights to the integral action has to take in account the fact that an aggressive set-point tracking may result in oscillatory, or even unstable, systems. To avoid these situations, the weights were selected to cause a smooth tracking, and the control horizon was expanded to allow the controller to operate for more hours.

Despite the mentioned difficulties, the controller was able to drive the system as close as possible to the reference, while still obeying the manipulated variables constraints. Notice

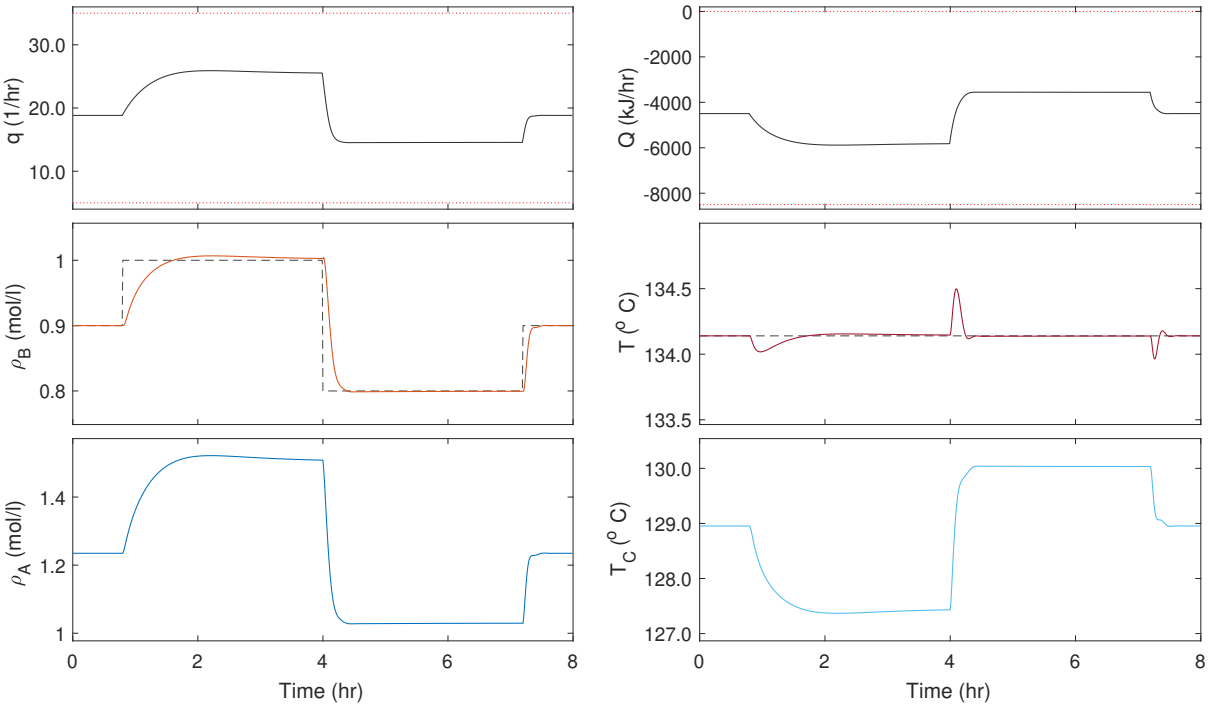


Figure 6.10: Simulation of the $LQRI_1$ for reference tracking.

that, in the moments were the reference changes, the change in the concentration resulted in a momentarily change in the temperature, which is promptly corrected by the feedback action of the controller. This change, however, is very small and the temperature remained within an interval of 0.5^oC around the reference signal, which is an excellent result. Another nice response observed in the visualization comprises in the change in the coolant temperature, which follows closely the shape of the change in the cooling capacity. In the last simulations, this variable evolved in a similar manner to the reactor temperature, but, in this case, the first was directly manipulated to allow for the latter to stay at the steady-state value.

Following the same case of the previous subsection, the same controller was used to simulate a closed-loop operation in the case when disturbances to the real system are occurring. In this case, a pulse signal is applied to the same variables as in the last case, but for each time instant multiple of $\lfloor T/4 \rfloor$ with a duration of $0.02T$, which is a longer step in this case since T is greater. The results are presented at Fig. 6.11. The first interesting phenomena to notice is that the system responded more aggressively to the disturbances that occurred between the time interval $t \in [0.8, 4]$ for which $r(t) = 1$, when compared to the disturbances occurring in the interval $t \in [4, 7.2]$ for which $r(t) = 0.8$. This is expected from the fact that the system has non-linear terms associated with the concentration ρ_A, which is greater at the first interval. The controller, however, was able to correct the disturbances and drive the system back to the reference.

In the direction for a more realistic scenario, the same reference tracking just proposed was used together with a Linear Quadratic Gaussian, augmented with integral action, to perform the feedback over the estimated states. In this case, a different weighting for the augmented states were modified to still result in a similar operation to the one in Fig. 6.6 and 6.7.

$$\boldsymbol{Q} = \begin{bmatrix} 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.1 \end{bmatrix} \quad ; \quad \boldsymbol{R} = \begin{bmatrix} 1 & 0 \\ 0 & 1.1e-5 \end{bmatrix}. \tag{6.13}$$

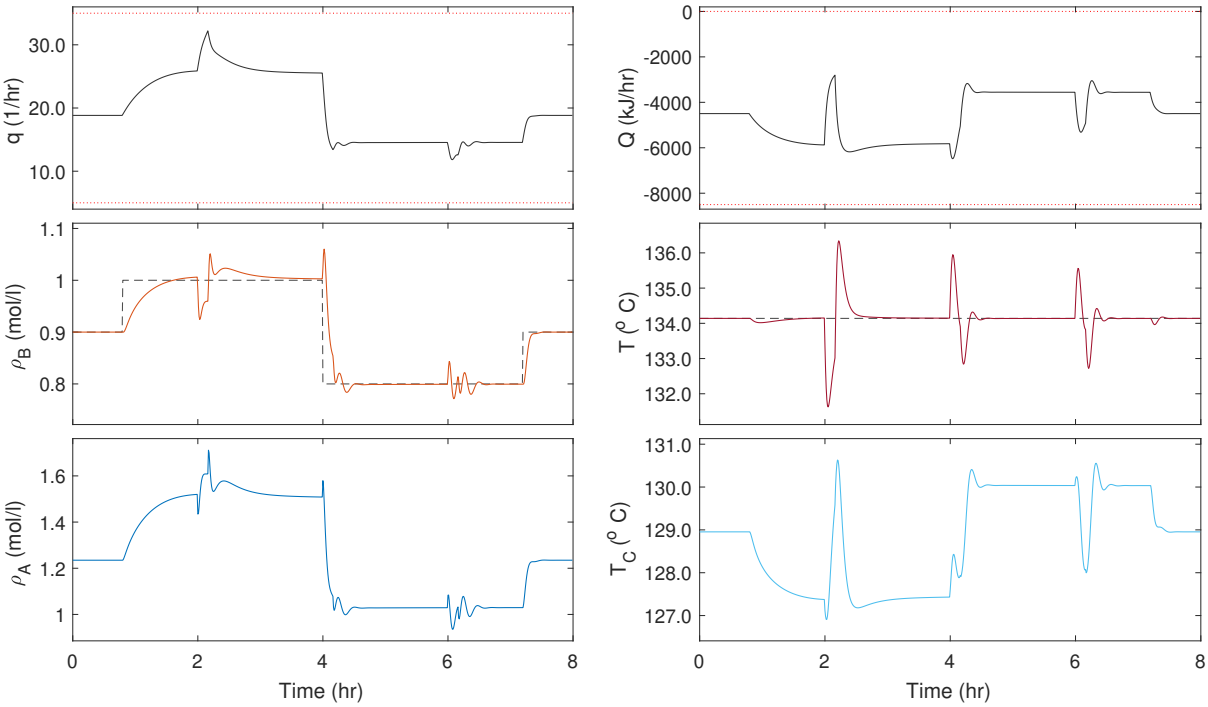


Figure 6.11: Simulation of the $LQRI_2$ controller for reference tracking, but subjected to three pulse of disturbances in the system variables.

In this case, the process and measurement noises used, $\boldsymbol{w} : \mathbb{R} \to \mathbb{R}^4$ and $\boldsymbol{z} : \mathbb{R} \to \mathbb{R}^2$, respectively, are described by the Gaussian distributions:

$$\boldsymbol{w}(t) \sim \mathcal{N}\left(\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} \middle| \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.4 & 0 & 0 & 0 \\ 0 & 0.4 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.5 \end{bmatrix}\right) \quad \boldsymbol{z}(t) \sim \mathcal{N}\left(\begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} \middle| \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.0001 & 0 \\ 0 & 0.01 \end{bmatrix}\right). \tag{6.14}$$

The greater values for the covariance $\boldsymbol{Q}_k$ of the covariance noise reflects the fact that the estimator becomes more uncertain about the system when it follows a reference signal far from the steady-state value. The resulting operation is displayed at Fig. 6.12.

The first difference noted was the fact that the weights for the augmented states for the LQG controller actually needed a value of greater magnitude than those used for the LQR case to be able to yield the same results. There are two important phenomena to emphasize in this result. First of all, the input signals requested from the controller are not smooth, resulting in a oscillatory behavior in the actual response of the system (and in the Kalman-Bucy filter state estimations). This is resulting for the fact, as noted in Example 4.1, that the integral action is calculated using the measurements of the system output, which are noisy. In addition to this, the Kalman-Bucy filter was not able to effectively reconstruct the real states in the cases for the references that are far from the linearization steady-state point. It is visible that the estimates have an offset when the system reaches the steady-state after tracking those references. In contrast, the estimate is far accurate when the real response is close to the linearization steady-state point, as evidenced in the temperature responses. This phenomena is resulting from the fact that the Kalman-Bucy filter is modeled from a linearized realization of the non-linear system, exhibiting the same limitations as were observed in Fig. 2.3.

The two problems just mentioned are somehow connected. A possible solution for the oscillatory input could be to use the state estimates and the linearized State-Space model to produce an output estimate; and then use it to calculate the integral action. Since the matrix $\boldsymbol{Q}_k$,

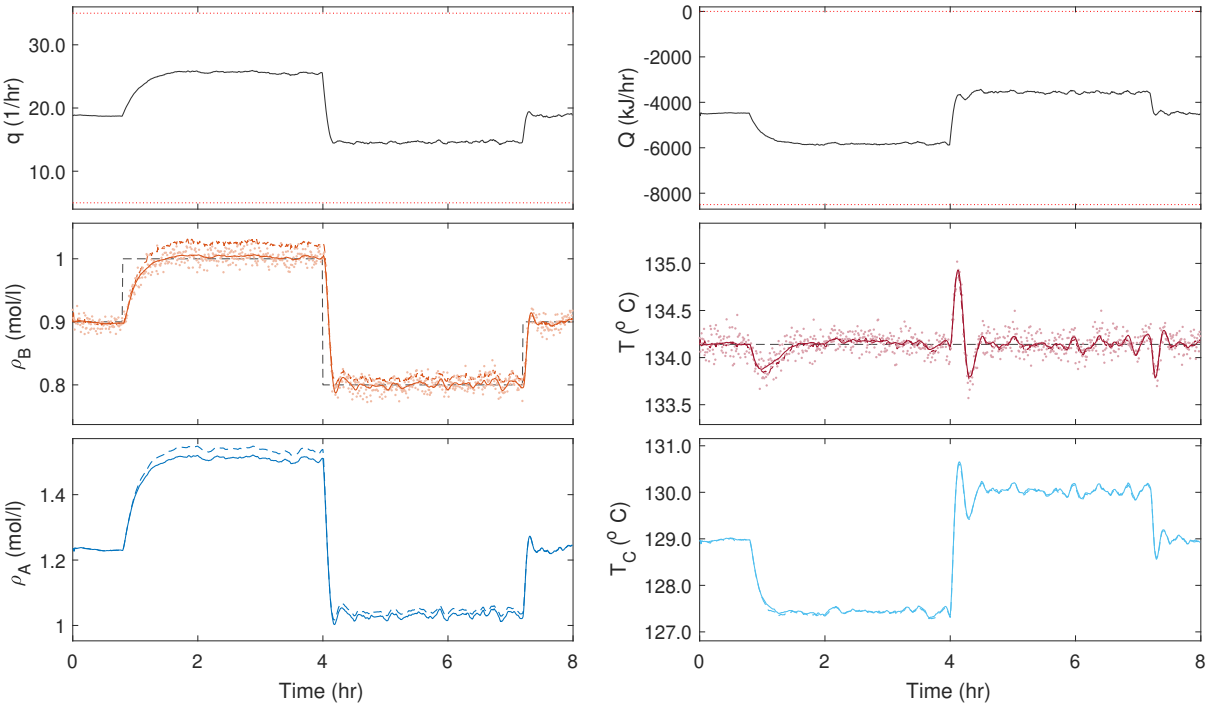


Figure 6.12: Simulation of the $LQGI_1$ controller for reference tracking.

estimated from the noise covariances, can be underestimated so that the Kalman-Bucy model is considered less uncertain and the estimations are less oscillatory. However, the limitation of the linear Kalman-Bucy filter will result in a poor reference tracking, in respect to the non-linear system response, if the estimated states are used in the integral action. A feasible solution, in this case, would be to explore more performing state estimators for non-linear systems.

For the sake of completeness, the same tracking objective was simulated for the case of disturbances actuating on the real system, just as in the previous cases, as shown in Fig. 6.13. The resulting control shows a more aggressive response in respect to the previous case, but it was still able to correct the disturbance and track the system to the reference. This fact proves the feasibility of using controller-estimator configurations for reference tracking control. Finally, a summary of the results in this subsection are displayed at Table 6.2.

Controller	**ISE**	**IAE**	**ITSE**	**ITAE**
$LQRI_1$	$\begin{bmatrix}0.5929\\2.0197\end{bmatrix}$	$\begin{bmatrix}7.4300\\15.1220\end{bmatrix}$	$\begin{bmatrix}9.0789\\33.7962\end{bmatrix}$	$\begin{bmatrix}23.1565\\49.4736\end{bmatrix}$
$LQGI_1$	$\begin{bmatrix}0.5060\\11.5765\end{bmatrix}$	$\begin{bmatrix}7.0710\\47.5308\end{bmatrix}$	$\begin{bmatrix}8.1233\\214.0017\end{bmatrix}$	$\begin{bmatrix}25.0444\\189.6191\end{bmatrix}$
$LQRI_2$	$\begin{bmatrix}0.7860\\167.4390\end{bmatrix}$	$\begin{bmatrix}10.1643\\128.4992\end{bmatrix}$	$\begin{bmatrix}11.6943\\2059.6345\end{bmatrix}$	$\begin{bmatrix}32.7671\\450.4048\end{bmatrix}$
$LQGI_2$	$\begin{bmatrix}0.8902\\687.4735\end{bmatrix}$	$\begin{bmatrix}12.6788\\322.6335\end{bmatrix}$	$\begin{bmatrix}17.1311\\11317.1011\end{bmatrix}$	$\begin{bmatrix}50.5231\\1274.8359\end{bmatrix}$

Table 6.2: Comparison between the LQRI and LQGI controllers. The pair $(LQRI_2, LQGI_2)$ comprises the simulations in which the system was subjected to disturbances.

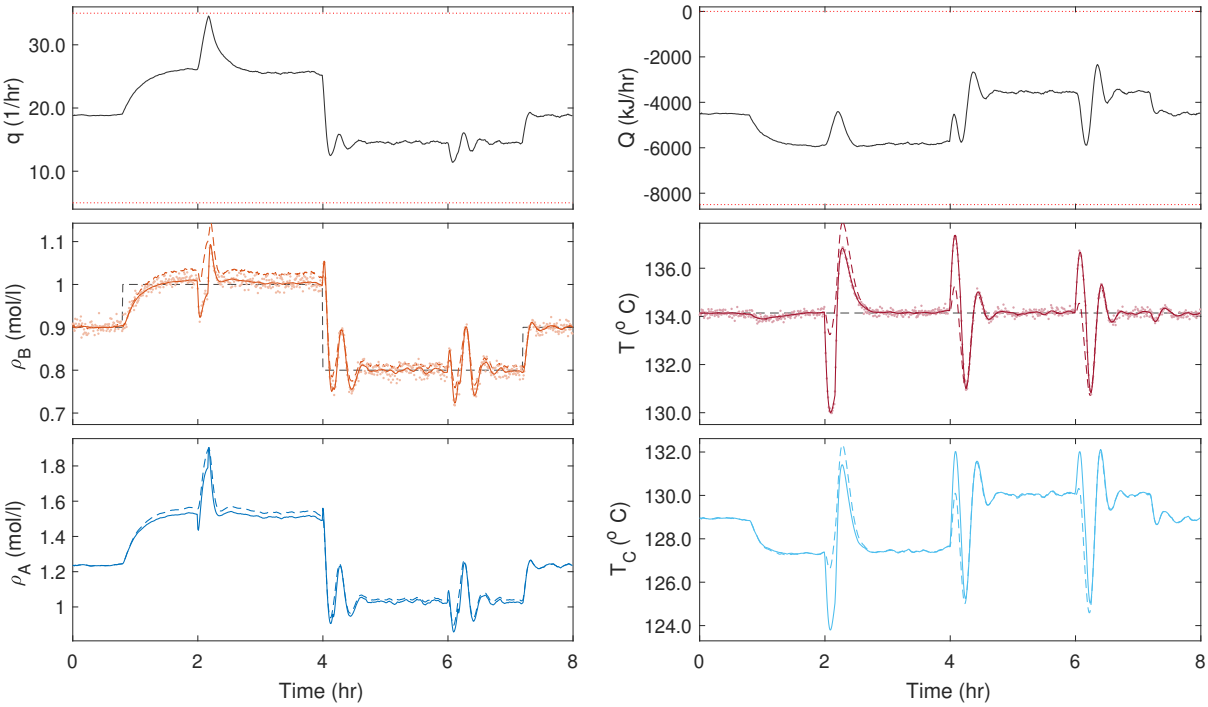


Figure 6.13: Simulation of the $LQGI_2$ controller for reference tracking, but subjected to three pulse of disturbances in the system variables.

Chapter 7

Conclusion

This work has presented a self-contained mathematical framework to model and control chemical reactor systems, focusing on optimal control techniques. In this document, several results were presented and discussed in order to provide a complete understanding of the concepts and methods explored. The discussed methods were applied to a given model of a challenging real-world chemical reactor, which consists of a non-isothermal process following the Van de Vusse reaction scheme. The experiments have shown that optimal control is a feasible solution when the controller has direct access to the internal states of the system, but performed worse when the states are not accessible but estimated. The control simulations, however, still produced practical results that are realistic in respect to real plant of such processes.

The use of optimal controllers greatly reduces the complexity of the controller synthesis procedure, while ensuring an optimal operation. The methods presented can still be expanded to more complex formulations, which would allow for better performance in more adverse cases. Specifically, more advanced state estimation techniques can be considered to mitigate the problem of reference tracking, determining the integral action through an estimate of the controlled variables.

This work also emphasizes the possibility of formulating more advances techniques, within the scope of optimal control, to achieve better results. As future work, there is the necessity to decouple the control and estimator sample time from the continuous-time in which the real system operates, providing a more realistic configuration of feedback control loops. Additionally, non-linear state estimation techniques should be explored to solve the linearized Kalman filter limitation, and constrained optimal controllers should be explored to mathematically ensure that the hard constraints of the system are being respected.

Bibliography

[Anderson and Moore, 1990] Anderson, D. O. B. and Moore, J. B. (1990). *Optimal Control*. Prentice-Hall.

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

[Atkins and de Paula, 2011] Atkins, P. and de Paula, J. (2011). *Physical Chemistry for the Life Sciences*. W. H. Freeman, 2nd edition.

[Barber, 2012] Barber, D. (2012). *Bayesian reasoning and machine learning*. Cambridge University Press.

[Bellman, 1954] Bellman, R. (1954). The theory of dynamic programming. *Bull. Amer. Math. Soc.*, 60(6):503–515.

[Bergman et al., 2017] Bergman, T. L., Lavine, A. S., Incropera, F. P., and DeWitt, D. P. (2017). *Fundamentals of Heat and Mass Transfer*. Wiley, 8th edition.

[Bertsekas, 2017] Bertsekas, D. P. (2017). *Dynamic Programming and Optimal Control*, volume I. Athena Scientific, 4 edition.

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

[Boyd and Vandenberghe, 2004] Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

[Bryson, 1996] Bryson, A. E. (1996). Optimal control-1950 to 1985. *IEEE Control Systems Magazine*, 16(3):26–33.

[Cairano et al., 2014] Cairano, S. D., Bernardini, D., Bemporad, A., and Kolmanovsky, I. V. (2014). Stochastic mpc with learning for driver-predictive vehicle control and its application to hev energy management. *IEEE Transactions on Control Systems Technology*, 22(3):1018–1031.

[Chen, 1998] Chen, C.-T. (1998). *Linear System Theory and Design*. Oxford University Press.

[Crassidis and Junkins, 2011] Crassidis, J. L. and Junkins, J. L. (2011). *Optimal Estimation of Dynamic Systems*. Chapman & Hall/CRC, 2nd edition.

[Davison and Maki, 1973] Davison, E. and Maki, M. (1973). The numerical solution of the matrix riccati differential equation. *IEEE Transactions on Automatic Control*, 18(1):71–73.

[Doyle, 1978] Doyle, J. C. (1978). Guaranteed margins for lqg regulators. *IEEE Transactions on automatic Control*, 23(4):756–757.

[Eren et al., 2017] Eren, U., Prach, A., Koçer, B. B., Raković, S. V., Kayacan, E., and Açıkmeşe, B. (2017). Model predictive control in aerospace systems: Current state and opportunities. *Journal of Guidance, Control, and Dynamics*, 40(7):1541–1566.

[Falcone et al., 2007] Falcone, P., Borrelli, F., Asgari, J., Tseng, H. E., and Hrovat, D. (2007). Predictive active steering control for autonomous vehicle systems. *IEEE Transactions on Control Systems Technology*, 15(3):566–580.

[Franklin et al., 2018] Franklin, G. F., Powell, J. D., Emami-Naeini, A., and Powell, J. D. (2018). *Feedback control of dynamic systems*. Pearson, 8th edition.

[Goerzen et al., 2009] Goerzen, C., Kong, Z., and Mettler, B. (2009). A survey of motion planning algorithms from the perspective of autonomous uav guidance. *Journal of Intelligent and Robotic Systems*, 57(1):65.

[González et al., 2016] González, D., Pérez, J., Milanés, V., and Nashashibi, F. (2016). A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):1135–1145.

[Gupta et al., 2012] Gupta, V. K., Ali, I., Saleh, T. A., Nayak, A., and Agarwal, S. (2012). Chemical treatment technologies for waste-water recycling—an overview. *RSC Adv.*, 2:6380–6388.

[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.

[Holenda et al., 2008] Holenda, B., Domokos, E., Rédey, A., and Fazakas, J. (2008). Dissolved oxygen control of the activated sludge wastewater treatment process using model predictive control. *Computers & Chemical Engineering*, 32(6):1270 – 1278.

[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, 1960a] Kalman, R. E. (1960a). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45.

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

[Khalil, 2005] Khalil, H. K. (2005). A note on the robustness of high-gain-observer-based controllers to unmodeled actuator and sensor dynamics. *Automatica*, 41(10):1821 – 1824.

[Kirk, 1998] Kirk, D. E. (1998). *Optimal Control Theory*. Prentice-Hall.

[Klatt and Engell, 1998] Klatt, K.-U. and Engell, S. (1998). Gain-scheduling trajectory control of a continuous stirred tank reactor. *Computers & Chemical Engineering*, 22(4-5):491–502.

[Liberzon, 2012] Liberzon, D. (2012). *Calculus of Variations and Optimal Control Theory*. Princeton University Press.

[Lucia et al., 2013] Lucia, S., Finkler, T., and Engell, S. (2013). Multi-stage nonlinear model predictive control applied to a semi-batch polymerization reactor under uncertainty. *Journal of Process Control*, 23(9):1306 – 1319.

[Mahmoud and Khalil, 2002] Mahmoud, M. S. and Khalil, H. K. (2002). Robustness of high-gain observer-based nonlinear controllers to unmodeled actuators and sensors. *Automatica*, 38(2):361 – 369.

[MATLAB, 2018] MATLAB (2018). *version 9.5.0 (R2018b)*. The MathWorks Inc., Natick, Massachusetts.

[Mnih et al., 2015] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, 518:529.

[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.

[Moore, 1975] Moore, B. C. (1975). On the flexibility offered by state feedback in multivariable systems beyond closed loop eigenvalue assignment. In *1975 IEEE Conference on Decision and Control including the 14th Symposium on Adaptive Processes*, pages 207–214.

[Mulas et al., 2015] Mulas, M., Tronci, S., Corona, F., Haimi, H., Lindell, P., Heinonen, M., Vahala, R., and Baratti, R. (2015). Predictive control of an activated sludge process: An application to the viikinmäki wastewater treatment plant. *Journal of Process Control*, 35:89 – 100.

[Nise, 2015] Nise, N. S. (2015). *Control Systems Engineering*. Wiley, 7th edition.

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

[Opanuga et al., 2015] Opanuga, A. A., Edeki, S., Okagbue, H. I., and Akinlabi, G. (2015). A novel approach for solving quadratic riccati differential equations. *International Journal of Applied Engineering Research*, 10(11):29121–29126.

[Qin and Badgwell, 2003] Qin, S. and Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7):733 – 764.

[Rasmussen and Williams, 2006] Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.

[Ross, 2010] Ross, S. M. (2010). *A first course in probability*. Pearson.

[Särkkä, 2013] Särkkä, S. (2013). *Bayesian Filtering and Smoothing*. Cambridge University Press, New York, NY, USA.

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

[Stratonovich, 1968] Stratonovich, R. L. (1968). *Conditional Markov Processes and Their Application to the Theory of Optimal Control*. American Elsevier Publisher.

[Sutton and Barto, 2018] Sutton, R. and Barto, A. (2018). *Reinforcement Learning*. MIT Press, 2nd edition.

[Syrmos et al., 1997] Syrmos, V., Abdallah, C., Dorato, P., and Grigoriadis, K. (1997). Static output feedback—a survey. *Automatica*, 33(2):125 – 137.

[Tang et al., 2016] Tang, X., Rupp, B., Yang, Y., Edwards, T. D., Grover, M. A., and Bevan, M. A. (2016). Optimal feedback controlled assembly of perfect crystals. *ACS Nano*, 10(7):6791–6798. PMID: 27387146.

[Todorov and Jordan, 2002] Todorov, E. and Jordan, M. I. (2002). Optimal feedback control as a theory of motor coordination. *Nature Neuroscience*, 5(11):1226–1235.

[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.

[Wang and Su, 2015] Wang, L. and Su, J. (2015). Robust disturbance rejection control for attitude tracking of an aircraft. *IEEE Transactions on Control Systems Technology*, 23(6):2361–2368.

Appendix A

Properties of Gaussian Distributions

Definition A.1. (Gaussian Distribution) A random variable $\boldsymbol{x} \in \mathbb{R}^n$ has a Gaussian distribution with mean $\boldsymbol{\mu} \in \mathbb{R}^n$ and covariance $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$, denoted as $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, if its probability density is described as:

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right\}. \tag{A.1}$$

Lemma A.1. *(Joint distribution of Gaussians) Consider the random variables $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{y} \in \mathbb{R}^m$, which have the Gaussian distributions*

$$\begin{aligned} p(\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \\ p(\boldsymbol{y}|\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{u}, \boldsymbol{L}), \end{aligned} \tag{A.2}$$

then the distribution of $(\boldsymbol{x}, \boldsymbol{y})$ and the marginal distribution of $\boldsymbol{y}$ are given as:

$$\begin{aligned} p\left(\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix}\right) &= \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix} \middle| \begin{bmatrix}\boldsymbol{\mu}\\ \boldsymbol{A}\boldsymbol{\mu}+\boldsymbol{u}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma} & \boldsymbol{\Sigma}\boldsymbol{A}^T \\ \boldsymbol{A}\boldsymbol{\Sigma} & \boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}^T + \boldsymbol{L}\end{bmatrix}\right), \\ p(\boldsymbol{y}) &= \mathcal{N}\left(\boldsymbol{A}\boldsymbol{\mu} + u, \boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}^T + \boldsymbol{L}\right). \end{aligned} \tag{A.3}$$

Lemma A.2. *(Conditional distribution of Gaussians) Consider the random variables $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{y} \in \mathbb{R}^m$, which have the joint Gaussian distribution*

$$p\left(\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix} \middle| \begin{bmatrix}\boldsymbol{\mu}_x\\ \boldsymbol{\mu}_y\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{xx} & \boldsymbol{\Sigma}_{xy} \\ \boldsymbol{\Sigma}_{xy}^T & \boldsymbol{\Sigma}_{yy}\end{bmatrix}\right), \tag{A.4}$$

then the marginal and conditional distributions of $\boldsymbol{x}$ and $\boldsymbol{y}$ are given as:

$$\begin{aligned} p(\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_{xx}), \\ p(\boldsymbol{y}) &= \mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_{yy}), \\ p(\boldsymbol{x}|\boldsymbol{y}) &= \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_x + \boldsymbol{\Sigma}_{xy}\boldsymbol{\Sigma}_{yy}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_y), \boldsymbol{\Sigma}_{xx} - \boldsymbol{\Sigma}_{xy}\boldsymbol{\Sigma}_{yy}^{-1}\boldsymbol{\Sigma}_{xy}^T), \\ p(\boldsymbol{y}|\boldsymbol{x}) &= \mathcal{N}(\boldsymbol{y}|\boldsymbol{\mu}_y + \boldsymbol{\Sigma}_{xy}^T\boldsymbol{\Sigma}_{xx}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_x), \boldsymbol{\Sigma}_{yy} - \boldsymbol{\Sigma}_{xy}^T\boldsymbol{\Sigma}_{xx}^{-1}\boldsymbol{\Sigma}_{xy}). \end{aligned} \tag{A.5}$$