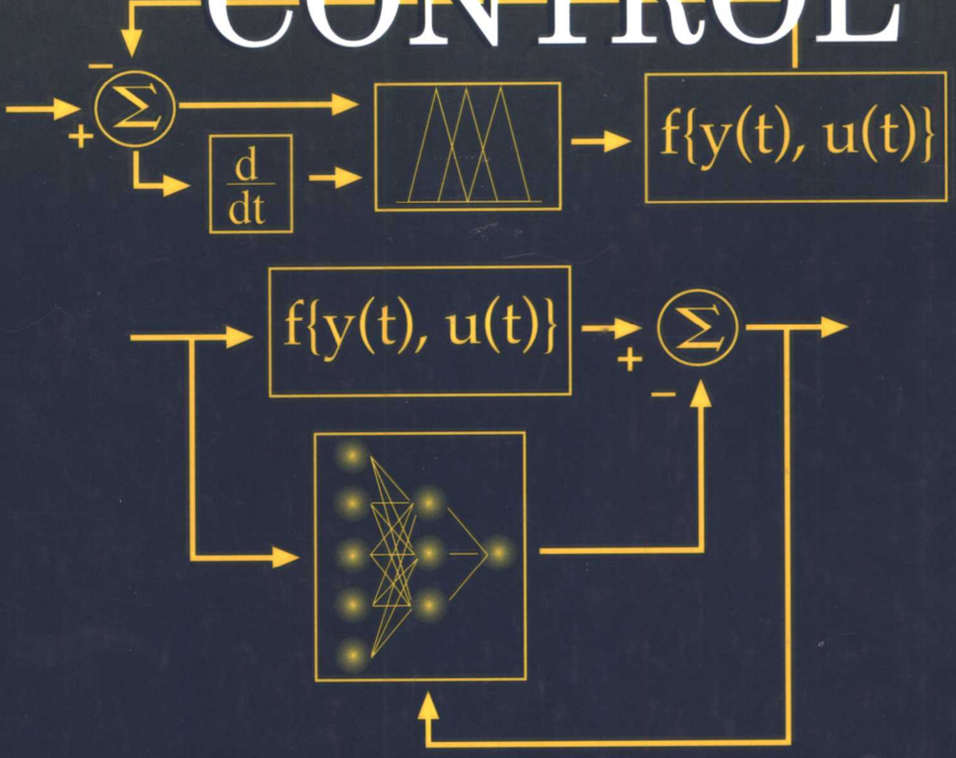
A First Course in FUZZY and NEURAL CONTROL



Hung T. Nguyen • Nadipuram R. Prasad
Carol L. Walker • Elbert A. Walker

CHAPMAN & HALL/CRC

A First Course in FUZZY and NEURAL CONTROL

Hung T. Nguyen • Nadipuram R. Prasad
Carol L. Walker • Elbert A. Walker

CHAPMAN & HALL/CRC

A CRC Press Company
Boca Raton London New York Washington, D.C.

Library of Congress Cataloging-in-Publication Data

A first course in fuzzy and neural control / Hung T. Nguyen ... [et al.].
p. cm.
Includes bibliographical references and index.
ISBN 1-58488-244-1
1. Soft computing. 2. Neural networks (Computer science) 3. Fuzzy systems. 4. Control theory. I. Nguyen, Hung T., 1944-

QA76.9.S63 .F57 2002
006.3—dc21 2002031314

Direct all inquiries to CRC Press LLC, 2000 N.W. Corporate Blvd., Boca Raton, Florida 33431.

Visit the CRC Press Web site at www.crcpress.com

International Standard Book Number 1-58488-244-1
Library of Congress Card Number 2002031314
Printed in the United States of America 1 2 3 4 5 6 7 8 9 0
Printed on acid-free paper

Preface

Soft computing approaches in decision making have become increasingly popular in many disciplines. This is evident from the vast number of technical papers appearing in journals and conference proceedings in all areas of engineering, manufacturing, sciences, medicine, and business. Soft computing is a rapidly evolving field that combines knowledge, techniques, and methodologies from various sources, using techniques from neural networks, fuzzy set theory, and approximate reasoning, and using optimization methods such as genetic algorithms. The integration of these and other methodologies forms the core of soft computing.

The motivation to adopt *soft computing*, as opposed to *hard computing*, is based strictly on the tolerance for imprecision and the ability to make decisions under uncertainty. Soft computing is goal driven — the methods used in finding a path to a solution do not matter as much as the fact that one is moving toward the goal in a reasonable amount of time at a reasonable cost. While soft computing has applications in a wide variety of fields, we will restrict our discussion primarily to the use of soft computing methods and techniques in control theory.

Over the past several years, courses in fuzzy logic, artificial neural networks, and genetic algorithms have been offered at New Mexico State University when a group of students wanted to use such approaches in their graduate research. These courses were all aimed at meeting the special needs of students in the context of their research objectives. We felt the need to introduce a formal curriculum so students from all disciplines could benefit, and with the establishment of The Rio Grande Institute for Soft Computing at New Mexico State University, we introduced a course entitled "Fundamentals of Soft Computing I" during the spring 2000 semester. This book is an outgrowth of the material developed for that course.

We have a two-fold objective in this text. Our first objective is to emphasize that both fuzzy and neural control technologies are firmly based upon the principles of classical control theory. All of these technologies involve knowledge of the basic characteristics of system response from the viewpoint of stability, and knowledge of the parameters that affect system stability. For example, the concept of state variables is fundamental to the understanding of whether or not a system is controllable and/or observable, and of how key system variables can be monitored and controlled to obtain desired system performance.

To help meet the first objective, we provide the reader a broad flavor of what classical control theory involves, and we present in some depth the mechanics of implementing classical control techniques. It is not our intent to cover classical methods in great detail as much as to provide the reader with a firm understanding of the principles that govern system behavior and control. As an outcome of this presentation, the type of information needed to implement classical control techniques and some of the limitations of classical control techniques should become obvious to the reader.

Our second objective is to present sufficient background in both fuzzy and neural control so that further studies can be pursued in advanced soft computing methodologies. The emphasis in this presentation is to demonstrate the ease with which system control can be achieved in the absence of an analytical mathematical model. The benefits of a model-free methodology in comparison with a model-based methodology for control are made clear. Again, it is our intent to bring to the reader the fundamental mechanics of both fuzzy and neural control technologies and to demonstrate clearly how such methodologies can be implemented for nonlinear system control.

This text, *A First Course in Fuzzy and Neural Control*, is intended to address all the material needed to motivate students towards further studies in soft computing. Our intent is not to overwhelm students with unnecessary material, either from a mathematical or engineering perspective, but to provide balance between the mathematics and engineering aspects of fuzzy and neural network-based approaches. In fact, we strongly recommend that students acquire the mathematical foundations and knowledge of standard control systems before taking a course in soft computing methods.

Chapter 1 provides the fundamental ideas of control theory through simple examples. Our goal is to show the consequences of systems that either do or do not have feedback, and to provide insights into controller design concepts. From these examples it should become clear that systems can be controlled if they exhibit the two properties of *controllability* and *observability*.

Chapter 2 provides a background of classical control methodologies, including state-variable approaches, that form the basis for control systems design. We discuss state-variable and output feedback as the primary motivation for designing controllers via pole-placement for systems that are inherently unstable. We extend these classical control concepts to the design of conventional *Proportional-Integral* (PI), *Proportional-Derivative* (PD), and *Proportional-Integral-Derivative* (PID) controllers. Chapter 2 includes a discussion of stability and classical methods of determining stability of nonlinear systems.

Chapter 3 introduces mathematical notions used in linguistic rule-based control. In this context, several basic examples are discussed that lay the mathematical foundations of fuzzy set theory. We introduce linguistic rules -- methods for inferencing based on the mathematical theory of fuzzy sets. This chapter emphasizes the logical aspects of reasoning needed for intelligent control and decision support systems.

In Chapter 4, we present an introduction to fuzzy control, describing the general methodology of fuzzy control and some of the main approaches. We discuss the design of fuzzy controllers as well as issues of stability in fuzzy control. We give examples illustrating the solution of control problems using fuzzy logic.

Chapter 5 discusses the fundamentals of artificial neural networks that are used in control systems. In this chapter, we briefly discuss the motivation for neural networks and the potential impact on control system performance. In this context, several basic examples are discussed that lay the mathematical foundations of artificial neural networks. Basic neural network architectures, including single- and multi-layer perceptrons, are discussed. Again, while our objective is to introduce some basic techniques in soft computing, we focus more on the rationale for the use of neural networks rather than providing an exhaustive survey and list of architectures.

In Chapter 6, we lay down the essentials of neural control and demonstrate how to use neural networks in control applications. Through examples, we provide a step-by-step approach for neural network-based control systems design.

In Chapter 7, we discuss the hybridization of fuzzy logic-based approaches with neural network-based approaches to achieve robust control. Several examples provide the basis for discussion. The main approach is adaptive neuro-fuzzy inference systems (ANFIS).

Chapter 8 presents several examples of fuzzy controllers, neural network controllers, and hybrid fuzzy-neural network controllers in industrial applications. We demonstrate the design procedure in a step-by-step manner. Chapters 1 through 8 can easily be covered in one semester. We recommend that a minimum of two projects be assigned during the semester, one in fuzzy control and one in neural or neuro-fuzzy control.

Throughout this book, the significance of simulation is emphasized. We strongly urge the reader to become familiar with an appropriate computing environment for such simulations. In this book, we present MATLAB® simulation models in many examples to help in the design, simulation, and analysis of control system performance. MATLAB can be utilized interactively to design and test prototype controllers. The related program, Simulink®, provides a convenient means for simulating the dynamic behavior of control systems.

We thank the students in the Spring 2000 class whose enthusiastic responses encouraged us to complete this text. We give special thanks to Murali Siddaiah and Habib Gassoumi, former Ph.D. students of Ram Prasad, who kindly permitted us to share with you results from their dissertations that occur as examples in Chapters 6 and 8. We thank Chin-Teng Lin and C. S. George Lee who gave us permission to use a system identification example from their book *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems.*

Much of the material discussed in this text was prepared while Ram Prasad spent a year at the NASA/Jet Propulsion Laboratory between August 2001 and August 2002, as a NASA Faculty Fellow. For this, he is extremely thankful to Anil Thakoor of the Bio-Inspired Technologies and Systems Group, for his constant support, encouragement, and the freedom given to explore both

application-oriented technologies and revolutionary new technology development.

We thank our editor, Bob Stern, project coordinator, Jamie B. Sigal, and project editor Marsha Hecht for their assistance and encouragement. And we give a very personal thank you to Ai Khuyen Prasad for her utmost patience and good will.

Hung T. Nguyen
Nadipuram R. Prasad
Carol L. Walker
Elbert A. Walker

Las Cruces, New Mexico
July 2002

Contents

Chapter 1

A PRELUDE TO CONTROL THEORY

In this opening chapter, we present fundamental ideas of control theory through simple examples. These fundamental ideas apply no matter what mathematical or engineering techniques are employed to solve the control problem. The examples clearly identify the concepts underlying open-loop and closed-loop control systems. The need for feedback is recognized as an important component in controlling or regulating system performance. In the next chapter, we will present examples of classical modern control theory systems that rely on *mathematical models*, and in the remainder of this book, we explore possible alternatives to a rigid mathematical model approach. These alternative approaches — fuzzy, neural, and combinations of these — provide alternative designs for autonomous intelligent control systems.

1.1 An ancient control system

Although modern control theory relies on mathematical models for its implementation, control systems were invented long before mathematical tools were available for developing such models. An amazing control system invented about 2000 years ago by Hero of Alexandria, a device for the opening and closing of temple doors — is still viewed as a control system marvel. Figure 1.1 illustrates the basic idea of his vision. The device was actuated whenever the ruler and his entourage arrived to ascend the temple steps. The actuation consisted of lighting a fire upon a sealed altar enclosing a column of air. As the air temperature in the sealed altar increased, the expanding hot air created airflow from the altar into a sealed vessel directly below. The increase in air pressure created inside the vessel pushed out the water contained in this vessel. This water was collected in a bucket. As the bucket became heavier, it descended and turned the door spindles by means of ropes, causing the counterweights to rise. The left spindle rotated in the clockwise direction and the right spindle in the counter-

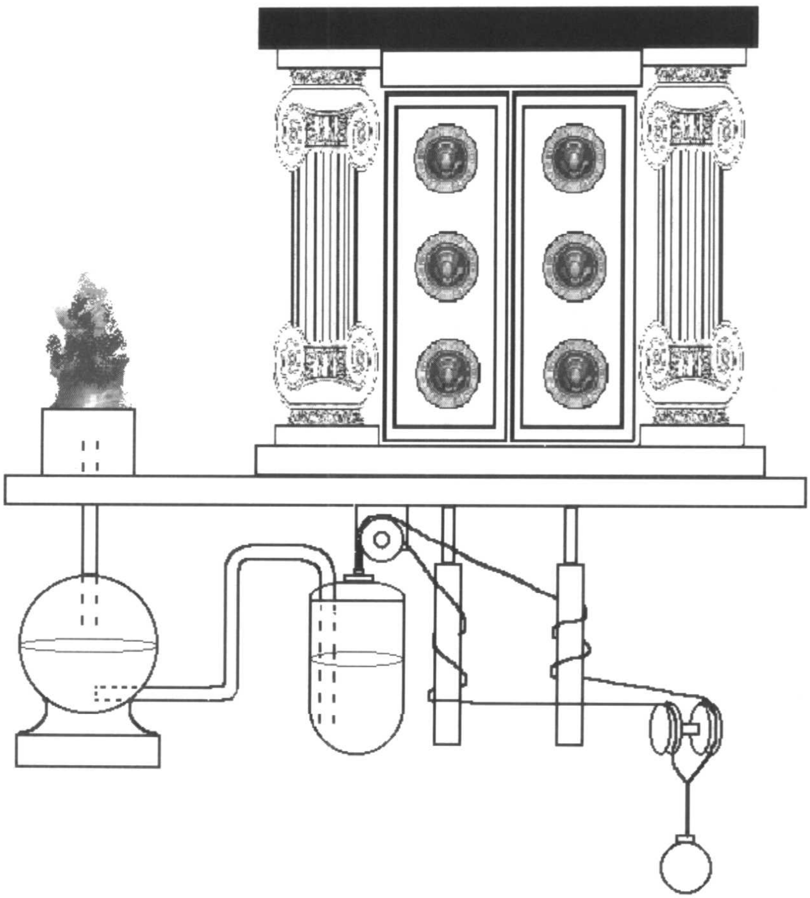


Figure 1.1. Hero's automatic temple doors

clockwise direction, thus opening the temple doors. The bucket, being heavier than the counterweight, would keep the temple doors open as long as the fire upon the altar was kept burning. Dousing the fire with cold water caused the temple doors to close.[1] As the air in the altar cooled, the contracting cool air in the altar created a suction to extract hot air from the sealed vessel. The resulting pressure drop caused the water from the bucket to be siphoned back into the sealed vessel. Thus, the bucket became lighter, and the counterweight

[1] Here, there is a question on how slow or how fast the temple doors closed after dousing out the fire. This is an important consideration, and a knowledge of the exponential decay in temperature of the air column inside the altar holds the answer. Naturally then, to give a theatrical appearance, Hero could have had copper tubes that carried the air column in close contact with the heating and cooling surface. This would make the temperature rise quickly at the time of opening the doors and drop quickly when closing the doors.

being heavier, moved down, thereby closing the door. This system was kept in total secret, thus creating a mystic environment of superiority and power of the Olympian Gods and contributing to the success of the Greek Empire.

1.2 Examples of control problems

One goal of classical science is to understand the behavior of motion of physical systems. In control theory, rather than just to understand such behavior, the object is to force a system to behave the way we want. Control is, roughly speaking, a means to force desired behaviors. The term **control**, as used here, refers generally to an instrument (possibly a human operator) or a set of instruments used to operate, regulate, or guide a machine or vehicle or some other system. The device that executes the control function is called the **controller**, and the system for which some property is to be controlled is called the **plant**. By a **control system** we mean the plant and the controller, together with the

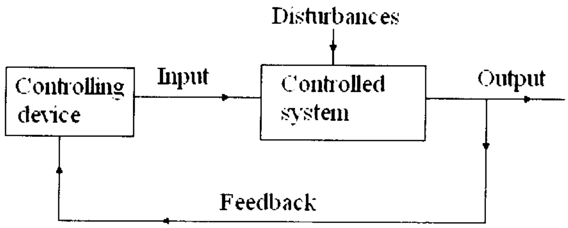


Figure 1.2. Control system

communication between them. The examples in this section include manual and automatic control systems and combinations of these. Figure 1.2 illustrates the basic components of a typical control system. The controlling device produces the necessary input to the controlled system. The output of the controlled system, in the presence of unknown disturbances acting on the plant, acts as a feedback for the controlling device to generate the appropriate input.

1.2.1 Open-loop control systems

Consider a system that is driven by a human — a car or a bicycle for example. If the human did not make observations of the environment, then it would be impossible for the "system" to be controlled or driven in a safe and secure manner. Failure to observe the motion or movement of the system could have catastrophic results. Stated alternatively, if there is no feedback regarding the system's behavior, then the performance of the system is governed by how well the operator can maneuver the system without making any observations of the behavior of the system. Control systems operating without feedback regarding the system's behavior are known as **open-loop control systems**. In other words, an open-loop control system is one where the control inputs are chosen without regard to the actual system outputs. The performance of such systems can only be guaranteed if the task remains the same for all time and can be duplicated repeatedly by a specific set of inputs.

Example 1.1 (Traffic light) To control the flow of traffic on city streets, a traffic engineer may preset a fixed time interval for a traffic light to turn green, yellow, and red. In this example, the environment around the street intersection is the plant. Traffic engineers are interested in controlling some specified plant

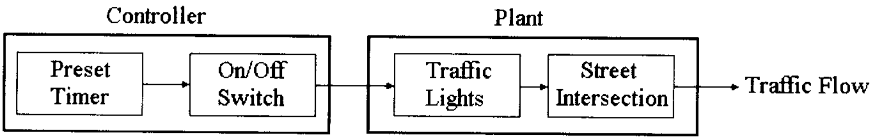


Figure 1.3. Traffic light, open-loop control

output, here the traffic flow. The preset timer and on/off switch for the traffic light comprise the controller. Since the traffic lights operate according to a preset interval of time, without taking into account the plant output (the timing is unaltered regardless of the traffic flow), this control system is an open-loop control system. A pictorial representation of the control design, called a ***block diagram***, is shown in Figure 1.3.

Example 1.2 (Toaster) A toaster can be set for producing the desired darkness of toasted bread. The "darkness" setting allows a timer to time out and switch off the power to the heating coils. The toaster is the plant, and the

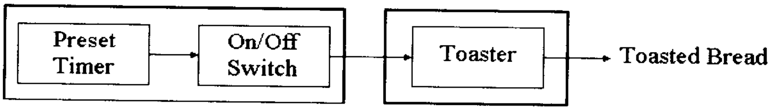


Figure 1.4. Standard toaster

timing mechanism is the controller. The toaster by itself is unable to determine the darkness of the toasted bread in order to adjust automatically the length of time that the coils are energized. Since the darkness of the toasted bread does not have any influence on the length of time heat is applied, there is no feedback in such a system. This system, illustrated in Figure 1.4, is therefore an open-loop control system.

Example 1.3 (Automatic sprinkler system) An automatic home sprinkler system is operated by presetting the times at which the sprinkler turns on and off. The sprinkler system is the plant, and the automatic timer is the controller.

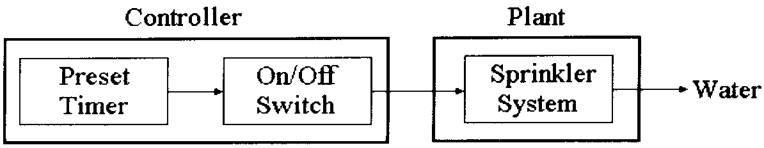


Figure 1.5. Automatic sprinkler system

There is no automatic feedback that allows the sprinkler system to modify the timed sequence based on whether it is raining, or if the soil is dry or too wet. The block diagram in Figure 1.5 illustrates an open-loop control system.

Example 1.4 (Conventional oven) With most conventional ovens, the cooking time is prescribed by a human. Here, the oven is the plant and the controller is the thermostat. By itself, the oven does not have any knowledge of the food

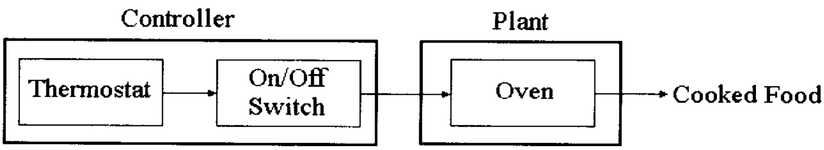


Figure 1.6. Conventional oven

condition, so it does not shut itself off when the food is done. This is, therefore, an open-loop control system. Without human interaction the food would most definitely become inedible. This is typical of the outcome of almost all open-loop control problems.

From the examples discussed in this section, it should become clear that some feedback is necessary in order for controllers to determine the amount of correction, if any, needed to achieve a desired outcome. In the case of the toaster, for example, if an observation was made regarding the degree of darkness of the toasted bread, then the timer could be adjusted so that the desired darkness could be obtained. Similar observations can be made regarding the performance of the controller in the other examples discussed.

1.2.2 Closed-loop control systems

Closed-loop systems, or **feedback control systems**, are systems where the behavior of the system is observed by some sensory device, and the observations are fed back so that a comparison can be made about how well the system is behaving in relation to some desired performance. Such comparisons of the performance allow the system to be controlled or maneuvered to the desired final state. The fundamental objective in closed-loop systems is to make the actual response of a system equal to the desired response.

Example 1.5 (Traffic light) To control the traffic flow in a more efficient

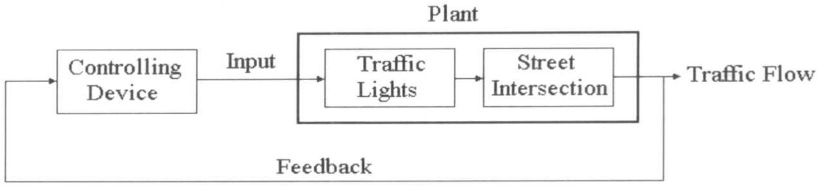


Figure 1.7. Traffic light feedback control

manner than in the example of the open-loop traffic light control described in Example 1.1, we could design a controller that does take into account the traffic flow (i.e., plant output). In this case, the new control system is referred to as a closed-loop system since the control strategy uses feedback information. The block diagram of this design is shown in Figure 1.7.

Example 1.6 (Flush tank) Suppose water flows into a flush tank through a supply valve, and the goal is to keep the water in the tank at a given level.

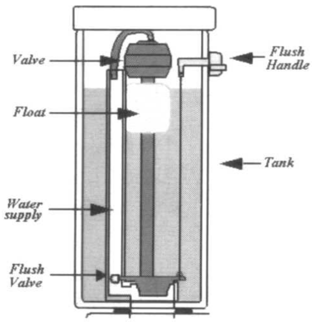


Figure 1.8. (a) Flush tank with float

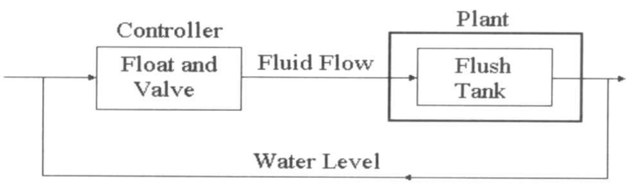


Figure 1.8. (b) Control system diagram for flush tank with float

One control system solving this problem uses a float that opens and closes the supply valve. As the water in the tank rises, the float rises and slowly begins to close the supply valve. When the water reaches the preset level, the supply valve closes shut completely. In this example, the float acts as the observer that provides feedback regarding the water level. This feedback is compared with the desired level, which is the final position of the float (see Figures 1.8 (a) and (b)).

Example 1.7 (Fluid level) Consider a manually controlled closed-loop system for regulating the level of fluid in a tank (see Figures 1.9 (a) and 1.9 (b)).

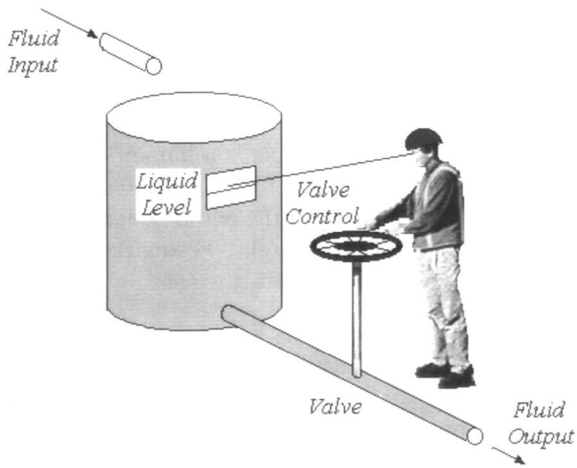


Figure 1.9. (a) Human maintaining fluid level

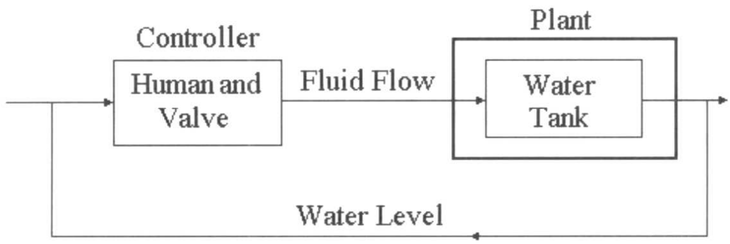


Figure 1.9. (b) Diagram of control system for maintaining fluid level

Fluid input is provided to the tank from a source that you can assume is continuous-time and time-varying. This means that the flow rate of fluid input can change with time. The fluid enters a tank in which there is an outlet for fluid output. The outlet is controlled by a valve, that can be opened or closed to control the flow rate of fluid output. The objective in this control scheme is to maintain a desired level of fluid in the tank by opening or closing the valve controlling the output. Such opening and closing operations either increase or decrease the fluid output flow rate to compensate for variations in the fluid input flow rate.