REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE DE SCIENCES ET TECHNOLOGIES
DEPARTEMENT DE GENIE ELECTRIQUE
N° :

DOMAINE : SCIENCES ET TECHNOLOGIE
FILIERE : Automatique
OPTION : Automatique et Systèmes

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par:

BOUNAB Aymen Abdelmalik
NADIR Badr Eddine

Intitulé

Control of Quadruple Tank Process Using Fuzzy Controllers Based on PSO Algorithm

Soutenu devant le jury composé de:

Dr OUBABAS Hocine	Université Mohamed Boudiaf - M'sila	Président
Dr AKKA Ali	Université Mohamed Boudiaf - M'sila	Rapporteur
Dr BARKAT Said	Université Mohamed Boudiaf - M'sila	Co-Rapporteur
Dr HAMMOUCHE Abdelaziz	Université Mohamed Boudiaf - M'sila	Examinateur

Année universitaire : 2019/ 2020

Acknowledgement

First of all, we would like to thank "Allah" for giving us the strength and courage to carry out this work.

We would like to express our appreciation for our supervisors Dr. Akka Ali and Pr. Barkat Said for their constant support and for providing guidance and feedback throughout this project.

We would like to express our deep gratitude to all our teachers from Electrical Engineering Department those inspired us during these years.

We would like thank all the people who have helped us from near or far.

Finally, we express our deep gratitude to our families, especially our fathers, mothers, brothers, sisters, as well as to all our relatives for their encouragement and support.

Dedicate

We dedicate this work to our dear parents, who have supported us with everything they have throughout our journey; without them we will never get to where we are.

To our brothers and sisters who have been always there for us.

To all our families.

To our friends who have studied with us during these years of specialty.

Abstract

Most industrial processes are multi-variable systems of a complex nature. Studying these systems via benchmark systems, such as the quadruple tank process, seems to be an effective solution. The quadruple tank process is a multi-input-multi-output process with minimum and non- minimum phase behavior. In this work, we investigated how to control the quadruple tank process using a fuzzy controller optimized by a PSO algorithm. Indeed, this study necessarily starts with a modeling phase on which a classical PI-based controller has been designed. In order to better control the behavior of the studied system, a fuzzy control is considered in the second part of this work. Finally, the above-mentioned controllers are optimized using the PSO algorithm and then applied to the quadruple tank process in order to boost its performance in its both operating phases.

ملخص

معظم العمليات الصناعية هي أنظمة متعددة المتغيرات ذات طبيعة معقدة. ان دراسة هذه الأنظمة من خلال أنظمة معيارية، مثل نظام الخزان الرباعي، تبدو حلاً فعالاً. نظام الخزان الرباعي هو نظام متعدد المداخل والمخارج فهو يتميز بتغير سلوكه بين حالتين الحالة الدنيا و الحالة القصوى. في هذه الدرسة سنرى كيفية التحكم في نظام الخزان الرباعي باستخدام وحدة تحكم ضبابية محسّنة بواسطة خوارزمية PSO. في الواقع ، تبدأ هذه الدراسة بمرحلة نمذجة ضرورية على اساسها تم تصميم وحدة تحكم كلاسيكية قائمة على PI. من أجل التحكم بشكل أفضل في سلوك النظام المدروس، تم اللجوء الى التحكم الضبابي في الجزء الثاني من هذا العمل. أخيرًا، تم تحسين طرق التحكم المذكورة أعلاه باستخدام خوارزمية PSO ثم تطبيقها على نظام الخزان الرباعي من أجل تعزيز أدائه في كلا حالتي التشغيل.

Résumé

La plupart des processus industriels sont des systèmes multi-variables de nature complexe. L’étude de ces systèmes via des systèmes benchmarks, tel que le processus à quatre réservoirs, semble être une solution efficace. Le processus à quatre réservoirs est un processus multi-entrées-multi-sorties présentant à la fois un comportement de phase minimum et un comportement de phase non minimum. Dans ce travail, nous sommes penchés sur la façon de contrôler le processus à quatre réservoirs à l'aide d'un contrôleur flou optimisé par un algorithme PSO. En effet, cette étude commence nécessairement par une phase de modélisation sur laquelle une commande classique à base de régulateurs PI a été conçue. Afin de mieux contrôler le comportement du système étudié, un réglage flou est considéré dans la deuxième partie de ce travail. Enfin, les régulateurs susmentionnés sont optimisés à l’aide de l'algorithme PSO puis appliqués sur le processus à quatre réservoirs afin de booster ces performances dans ses deux phases de fonctionnement.

Table of Contents

Chapter II

Fuzzy Control of Quadruple Tank Process

Chapter III

PSO Based Quadruple Tank Process Control

List of Figures

List of Tables

General Introduction

Most of the industrial systems face many control issues because they are nonlinear and have numerous manipulated and controlled variables with a large number of interacting control loops. The majority of the large and complex industrial processes are naturally Multi input Multi Output (MIMO) systems [1], such as the helicopter model and the active magnetic bearing process and the Quadruple Tank Process.

When dealing with MIMO systems, engineers faced many problems such as interaction, dominance and non-linearity etc. So, they decided to use the benchmark systems to find a way to solve these kinds of problems. The benchmark systems are useful for MIMO systems study because they allow to illustrate several interesting multivariable phenomena and also to develop new control techniques.

The Quadruple Tank Process (QTP) is a laboratory system, consisting of four interconnected water tanks and two pumps and two valves to separate the liquid [4]. The main objective is to independently control the water levels of two lower tanks using an appropriate control method.

The quadruple tank process has been sufficiently studied in the control literature to clarify different concepts in multivariate control, in particular the limitations of right half-plane zeros [6]. The main feature of the quadruple tank process is the flexibility in positioning one of its multivariate zeros on either half of the plane. Modeling is one of the most important steps in designing a control system. Although the problems of nonlinear tanks have been widely addressed in classical system dynamics when designing intelligent control systems, the corresponding simulation model should reflect all the characteristics of the actual system to be controlled [8].

In industrial process control, over 95% of the control loops are based on PI/PID type. This is mainly attributed to its effectiveness and relatively simple structure [3]. Several empirical and based-model approaches have been developed for the PI synthesis, known for their popularity in industrial fields such as Ziegler-Nichols method and the pole-assignment design techniques. Actually, there are numerous works in the literature, that deal with the control of the quadruple tank process using PI controller [1][3].

However, linear control techniques are not efficient enough to guarantee the stability and performance of nonlinear MIMO systems. For this reason, many researchers have turned to nonlinear techniques including intelligent techniques. Smart techniques, also known as soft computing techniques, namely neural networks, fuzzy logic, etc. have been proved to be very successful in controlling multivariate systems.

Recently, fuzzy control has become one of the most active areas for research in the application of fuzzy set theory [4]. Fuzzy logic was introduced by the researcher Zadeh (1965). Recently, it has become a very dominant tool for the representation of terms and vague knowledge. It is native to the human capacity to decide and act intelligently despite the vagueness and uncertainty of available knowledge. Its use in the field of control was one of the first applications in industry with the work of Mamdani and Assilian (1975) [18]. Since then, the applications of fuzzy logic have multiplied to affect very diverse fields.

There are several works in the literature, which deal with the quadruple tank process control using fuzzy control [5][20]. The main problem that every engineer faces when dealing with FLC is the determining the scaling factors values. And since there is no specific method to calculate them, they must use their knowledge and expertise and keep trying until they reach the desired result. This is probably one reason pushing many researchers to use optimization methods in order to overtake this limitation.

In order to get more efficient results with fuzzy control, applying metaheuristic optimization seems to be an effective way to achieve this goal. Metaheuristics are generic algorithms, inspired by nature; they are designed for solving complex optimization problems. Among the most recent metaheuristics, Particle Swarm Optimization (PSO) is one of those based on the movement of flocks of birds and schools of fish. Various versions of PSO have been used in different areas of research such as control systems, harmonic isolation, robot control, and problem of the job shop scheduling [20]. Due to its simplicity of implementation, PSO algorithm will be adopted here to achieve an optimal design for fuzzy controllers of quadruple tank process.

The main objective of our project is the PSO based fuzzy control of a quadruple tank process operating in minimum or non-minimum phase. In order to achieve this objective, modeling and control of the quadruple tank process using traditional PI and fuzzy controllers is the first step to take. Subsequently, the particle swarm optimization method is employed to adjust the PI parameters as well as the parameters of the membership functions of the fuzzy controller aim at improving the performance of traditional controllers.

To better organize this manuscript, it will be divided into three main chapters.

In the first chapter, we will present an overview about multivariable systems. Then a special attention will be given to the Quadruple Tank Process (QTP), in which we will explain its principal and establish its mathematical model. Finally, PI controllers design for QTP will be detailed.

In the second chapter, the fuzzy logic control of the quadruple tank process will be presented. First, the theoretical background of fuzzy logic will be provided. After that, the design of fuzzy logic controllers for quadruple tank process will be explained. At the end, comprehensive simulation results will be illustrated and discussed.

In the first part of the last chapter, an overview about PSO will be provided. In its second part, PI and fuzzy controllers optimized by PSO algorithm will be presented. At the end, some simulation results will be presented and examined.

At the end, a general conclusion will be given including some potential feature works.

Chapter I

Quadruple Tank Process Modeling and its PI Based Control

I.1 Introduction

In many situations, to control any industrial system it is indispensable to know its working principle and its mathematical model based on some known physical laws. Most of These industrial processes are multivariable systems.

For this reason, multivariable control is taught today in most control programs including both linear and nonlinear design methods. Lately it has been a huge interest in emphasizing the limitations the process imposes on control designs. However, it is not easy to find educational examples to explain multivariable performance, limitations in feedback systems. There was only a handful of multivariable laboratory processes, available from instruments manufacturers. None of these illustrate the effect of multivariable zeros on the closed-loop control performance. Until a group of researchers at Lund Institute of Technology, Sweden, in 1996 developed a new laboratory process in order to illustrate the importance of multivariable zero location for control design. This process called the quadruple tank process [16].

Quadruple tank process is a benchmark system that is used for a laboratory purposes. It has been designed to illustrate many concepts of control such us the performance limitation due to zeros locations in multivariable control systems [4].

The main characteristic of the quadruple tank process is the flexibility in positioning of its zeros on the "S" plane simply by changing the valves [6]. Therefore, the multivariable tank system becomes very interest in education and research fields.

It is well known that this kind of system is an interconnected system, which means that its control variables are effects on each other. This leads to interactions between its control loops, which makes the control task not easy.

But before talking about the control problems, first we are going to present in the first part of this chapter an overview about multivariable systems. Then a special attention will be given to the Quadruple Tank Process (QTP), in which we will explain its principal and establish its mathematical model.

I.2 Overview about Multivariable Systems

Systems can be classified into various types based on the number of inputs and outputs like SISO (Single input Single Output) and MIMO (Multi Input Multi Output). A SISO is a simple system with a single input and single output while MIMO system is a system with more than one input and output [7].

In dealing with multivariable systems, some extra concepts are important:

- **Grouping and pairing:** One or more inputs may be related to each controlled variable or group of them. The choice of pairings influences:
  - ***Interaction:*** The manipulated variables related to one controlled variable may affect the others.
  - ***Dominance:*** If the corresponding attached manipulated variable effect is greater than the others, the coupling presents dominance.
- **Conditioning:** Refers to the different “gains” a multivariable process may shows according to the combination (direction) of inputs. Then the process is ill conditioned and it may be difficult to control, if they are significatively different [5].

I.2.1 Process Variables

Studying the behavior of a process is to analyze the involved variables and their relationship [8].

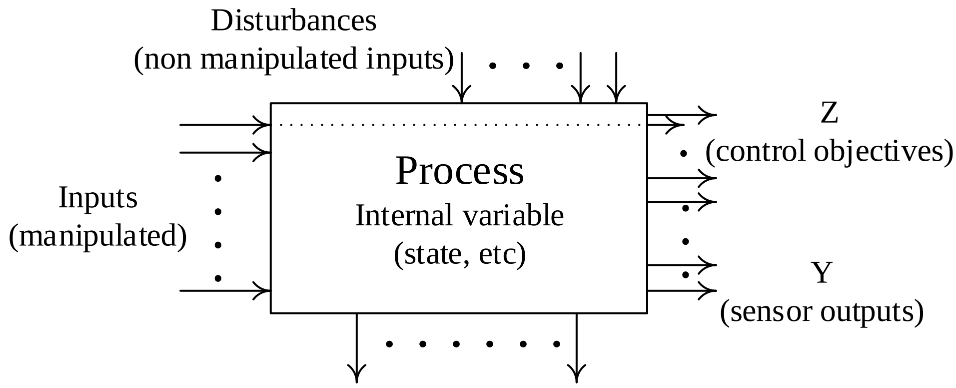


Figure (I.1): Process variables [5]

As shown in figure (I.1) these variables can be:

- **External** or **inputs**, being determined by other processes or the environment, acting on the process and considered as:

  - ***Manipulated*** or ***control variables***, *u*, if they are used to influence the dynamics of the process. Actuators will amplify the control commands to suitable power levels to modify plant's behavior.
  - ***Disturbance***, if they are uncontrollable outputs of other subsystems.
- **Internal**, being dependent on the process inputs, system structure and parameters. They can be classified as:
  - ***Outputs*** or ***measured variables***, *y*, if they are sensed and provide information about the process evolution.
  - ***Controlled variables***, *z*, if the control goals are based on them. They can be outputs or not, depending on the sensor's availability and placement.
  - ***State variables***, *x*, are a minimum set of internal variables allowing the computation of any other internal variable if the inputs are known.

I.2.2 Linearity

This is the most important simplification. A process is said to be linear if it accomplishes the following linearity principles:

- Proportionality.
- Causality.
- The response is proportional to the input. That is, if for a given input, $u(t)$ the response is $y(t)$, for an input $\alpha u(t)$ the response is $\alpha y(t)$, $\alpha \epsilon R$.
- The effect of various inputs is additive. That is, if for a given input, $u_1(t)$ the response is $y_1(t)$, and for an input, $u_2(t)$ the response is $y_2(t)$, for an input $u(t)=\alpha_1 u_1(t)+ \alpha_2 u_2(t)$ the response is $y(t)=\alpha_1 y_1(t)+ \alpha_2 y_2(t)$.

Linearization is an approximation technique allowing the representation of the nonlinear behavior of a process by an approximate linear model.

The linearization approach is used to consider the relationship between incremental variables around an equilibrium state, but it requires continuity and differentiability in the nonlinearities. Although this technique is not always applicable, in many cases it provides good insight into the process behavior and can be used in the design of a suitable controller [8].

I.2.3 Time invariance

This is another practical assumption, implying that parameters and functions appearing in the process model do not change with time. Usually, the process behavior changes with time because some parameters, assumed to be constant, slowly vary with time. Another cause of apparent time-variation is nonlinearity: changing the operating point changes the approximate linear behavior. Slow, un-modeled, nonlinear dynamics also may manifest that way, even without operating point changes [8].

I.3 Control Aims

Referring to a multivariable system where there are *p-controlled* variables, expressed as a vector *y(t),* and *m-manipulated* variables, *u(t),* and also considering a *d-dimensional* vector reference, *r(t),* can be stated at a different levels, as described below.

Ideal control: If the relationship between the process variables can be expressed by:

$$y = Gu + G_d d \tag{I.1}$$

where G and G_d are general operators, the ideal control action, leading to $y=r$ would be:

$$u = G^{-1}(r - G_d d) \tag{I.2}$$

This control could be perfect, but it is ideal and not achievable. There are many reasons for this, among them:

1. The operator G is usually not invertible.
2. Even if G were invertible, the resulting actions may be physically unfeasible.
3. The disturbance d may be un-measurable.
4. The process and disturbance models (G and G_d) are not perfectly known.

Possible control: For the control problem to be feasible, some performance requirement from the ideal one must be relaxed so that they are compatible with the constraints in the available actuator powers and measurements. For instance, high-speed reference tracking and low control action are not simultaneously achievable. High attenuation of un-measurable disturbance and high tolerance to modeling errors are also incompatible. Introducing additional sensors, actuators or control variables in a more complex *control structure* may improve the performance possibilities of the control system. Ideal control would be achieved under unlimited actuator power and *full information* sensing.

Optimal control: If the requirements are formulated as the minimization of a cost index or the maximization of a performance index, the resulting controller is called *"optimal"* one. But optimality from a mathematical viewpoint does not mean the best from a user view-point, unless user requirements are properly translated into optimization parameters [8].

Practical control: The controllers above are theoretical controllers. They are based on models and ideal performance. Other than the issues above, practical controllers should consider that:

- The models represent an approximate behavior of the actual process (sometimes very coarse) and this behavior may change,
- Formal control design specifications represent user requirements only approximately and partially,
- The process operation should be *robust* against moderate changes in the operating conditions, requirements and disturbance,
- The implementation of the controller is constrained to resource's availability.

Thus, practical controllers should prove to the end-users that they can consistently operate the process in an *"automatic"* way without the continuous surveillance of the operator. The complexity of the controller, the ease of parameter tuning, the interpretability of the different control actions, or its cost advantages are issues to be considered when selecting a control strategy [8].

I.4 Modes of Operation

The dynamic behavior of non-linear processes may be quite different depending on the operating conditions regarding loads, disturbances and references. But any controlled process may operate in a variety of situations such as starting-up/shutting-down, transferring from some operating conditions to new ones, under constraints (alarms, emergency) or under the guidance of the operator. All this means different modes of operation requiring different control strategies. In some cases, the same controller but with different parameters will be appropriate, but in some others new control structures or even no control at all may be needed [8].

In fact, any automatic control system should have the option of operating in at least two modes of operation:

- ***Manual control:*** if the manipulated variables are determined by the operator,
- ***Automatic control:*** where the manipulated variables are governed by the controller. This can be done in two basic settings:
  - **Open-loop control:** There is no feedback from the process and the manipulated variables are determined by the control system based on the information provided by the operator or input measurements,

– **Closed-loop control:** The controller determines the manipulated variable based on the references and goals introduced by the operator and the measurements from the process.

I.5 Description and Modeling of Quadruple Tank Process

Process models are tools that help for simulating the behavior of the controlled process and for analyzing its properties and evaluating the goals achievement, by designing the control system [8].

I.5.1 Quadruple Tank Process Description

The system under study is presented in Figure (I.2) and its schematic diagram is shown in Figure (I.2).It consists of four interconnected tanks and two pumps.

Figure (I.2): Quadruple Tank Process (QTP) [26]

The process inputs are $v_1$and v_2 (input voltages to pumps, (0-10 V)) and the outputs are h_1 and h_2 (are voltages from level measurement devices (0-10V)). The objective is to control the level of the lower two tanks with inlet flow rates. The output of each pump is split into two using a three-way valve. Thus, each pump output goes to two tanks, one lower and another upper, diagonally opposite and the ratio of the split up is controlled by the position of the valve. The position of the two valves determines whether the system can be appropriately placed either in the minimum phase or in the non-minimum phase. Let the parameter γ_i be determined by how the valves are set. If the parameter γ_1 is the ratio of the valve for the first tank then the parameter (1- γ_1) will be the valve ratio for the fourth tank. The parameters γ_1, $\gamma_2 \in [0, 1]$ are determined from how the valves are set prior to an experiment. The voltage applied to pump is v_i and the corresponding flow is $k_i v_i$ (k_i: Level measurement devices constant). The flow to tank 1 is $\gamma_1 k_1 v_1$ and the flow to tank 4 is $(1-\gamma_1)k_1 v_1$ and similarly for tank 2 and tank 3 [2].

For the figure (I.3), note that $\gamma_{a,b}$ are $\gamma_{1,2}$ respectively, and $q_{a,b}$ are the flow rate of pump (1,2)

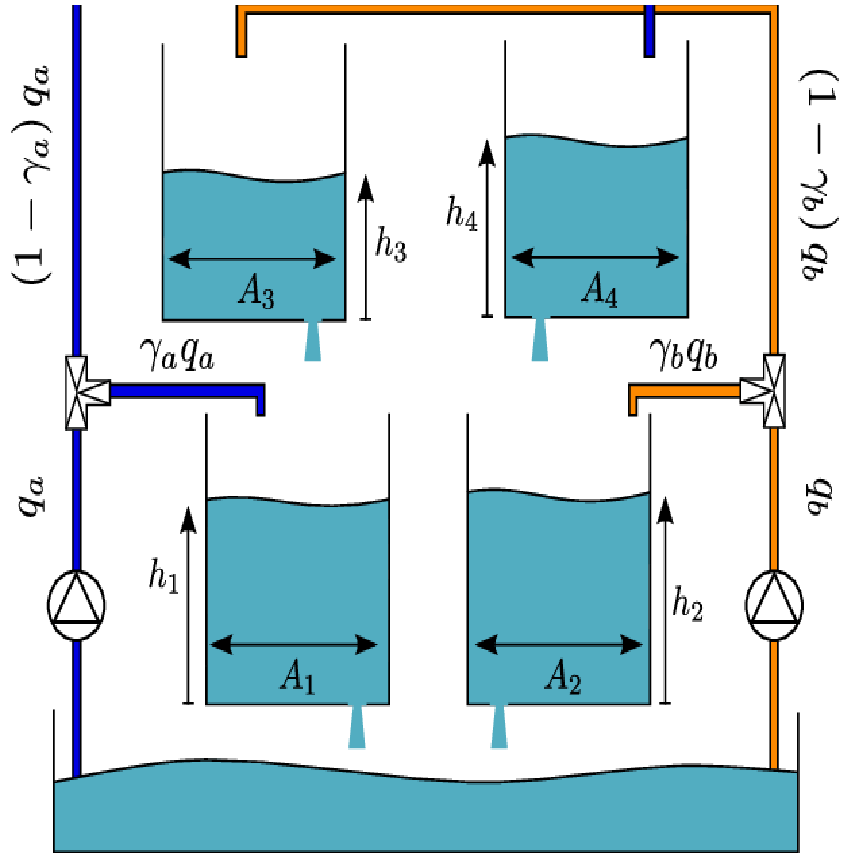


Figure (I.3): Schematic diagram of the quadruple tank process

I.5.2 Quadruple Tank Process Modeling

For each tank i=1,…,4, the mathematical modeling is done by consideration of mass balance equation and Bernoulli's law .

- Relationship between water level and the volume flow rate

The volume flow rate *Q* of a fluid is defined to be the volume of fluid that is passing through a given cross sectional area per unit time [5]. This is expressed as:

$$Q = \frac{dV}{dt} \qquad \text{(I.3)}$$

Where *V* is the tank volume.

Since the tanks are in a cylindrical shape ($V_i = A_i h_i$) then the equation (I.3) becomes:

$$Q_i = \frac{dh_i}{dt} A_i \tag{I.4}$$

Where A_i is the cross-sectional area of tank i, and h_i is the water level.

- Relationship between the inlet flow from the upper tank and the pump, and the outflow from the lower tank

This relationship can be proven by using the mass balance principle [7].The mass balance principle states that all mass is not created nor destroyed, it remains constant.

$$\left\{\begin{matrix}\text{Rate of accumulation} \\ \text{of mass in system}\end{matrix}\right\} = \left\{\begin{matrix}\text{Mass flow rate into} \\ \text{the system}\end{matrix}\right\} - \left\{\begin{matrix}\text{Mass flow rate out} \\ \text{of the system}\end{matrix}\right\}$$

This can be translated by the following equation:

$$\frac{dm_{acc}}{dt} = \frac{dm_{\mathrm{in}}}{dt} - \frac{dm_{out}}{dt} \tag{I.5}$$

The accumulated mass m_{acc} , mass at inlet m_{in} and mass at outlet m_{out} are expressed as follows :

$$\begin{aligned} m_{acc} &= V_i \rho_i \\ m_{in} &= Q_{in} \rho_{in} \\ m_{out} &= Q_{out} \rho_{out} \end{aligned} \tag{I.6}$$

Where ρ_i, ρ_{in}, ρ_{out} are the volumetric masses.

By replacing (I.6) into (I.5), it yields:

$$\frac{dV_i}{dt} \rho_i = \frac{dQ_{\mathrm{in}}}{dt} \rho_{in} - \frac{dQ_{out}}{dt} \rho_{out} \tag{I.7}$$

By considering $\rho_i = \rho_{in} = \rho_{out}$, the equation (I.7) will be simplified to the following one:

$$A_i \frac{dh_i}{dt} = \frac{dQ_{\mathrm{in}}}{dt} - \frac{dQ_{out}}{dt} \tag{I.8}$$

From figure (I.3),one can note that the inlet and outlet flow are proportional with the liquid speed and the cross-sectional area (a_i) of the outlet hole found in the bottom of the tank; hence:

$$\begin{aligned} Q_{in} &= a_i v_{in} \\ Q_{out} &= a_i v_{out} \end{aligned} \tag{I.9}$$

where V_{in}, V_{out} are the flow speeds.

Relationship between pressure and speed

The relationship between pressure and speed can be explained using Bernoulli principle.

Bernoulli principle

This principle states that the relationship between pressure and speed is inverse, which means that the more pressure, the lower the speed. Let us look at a tube containing static water. The water molecules collide with the pipe wall, which creates pressure on the pipe, but after the flow of water, its speed obstruct the random movement of water molecules to make them move in one direction and thus less collision with the wall of the pipe, which means the pressure decreases.

In addition, Bernoulli gave us an equation that relates the pressure, speed, and height of any two points (1 and 2) in a steady-flowing liquid out of density ρ [6]. Referring to figure (I.4), Bernoulli's equation is written as follows:

$$P_1 + \frac{1}{2}\rho v_1^2 + \rho g h_1 = P_2 + \frac{1}{2}\rho v_2^2 + \rho g h_2 \quad \text{(I.10)}$$

where, P_1, P_2 are liquid pressures; v_1,v_2 are liquid speed; h_1,h_2 are liquid levels and g is the gravity acceleration.

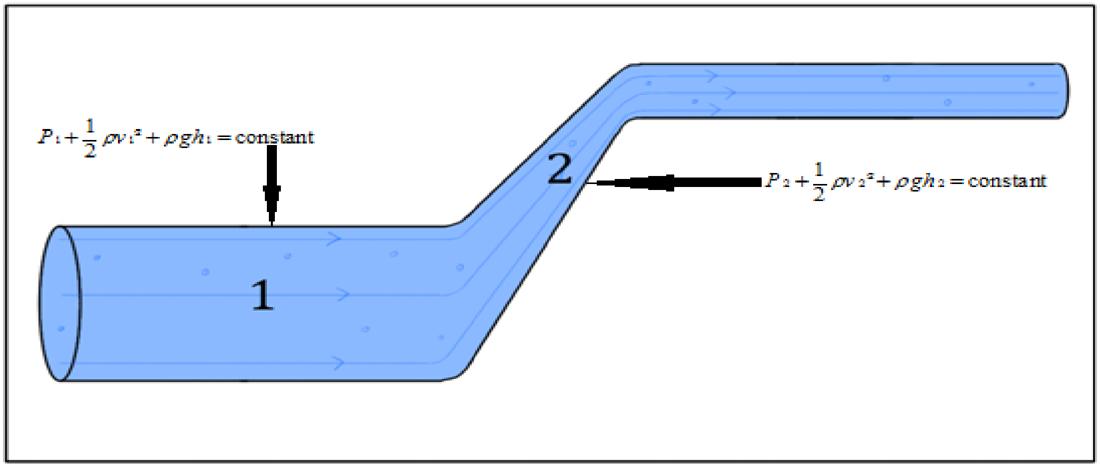


Figure (I.4): Diagram shows one particular choice of two points (1 and 2) in the fluid

By considering $P_1 = P_2 = P_{atm}$, $h_1 = 0$ and $v_2 << v_1$ then:

$$v_1 = \sqrt{2gh_2} \quad \text{(I.11)}$$

The equation (I.9) becomes,

$$Q_i = a_i \sqrt{2gh_i} \tag{I.12}$$

The two lower tanks have two inputs and one output, thus,

$$\frac{dh_i}{dt} = -\frac{a_i}{A_i}\sqrt{2gh_i} + \frac{a_{in}}{A_i}\sqrt{2gh_{in}} + \frac{\gamma_i k_i}{A_i} v_i \quad i = 1,2 \tag{I.13}$$

And the two upper tanks have one input and one output,

$$\frac{dh_i}{dt} = -\frac{a_i}{A_i}\sqrt{2gh_i} + \frac{(1-\gamma_j)k_j}{A_i} v_j \quad i = 3,4 \text{ ; j=1,2} \tag{I.14}$$

Finally, the complete model of the quadruple tank process is as follows:

$$\begin{aligned}
\frac{dh_1}{dt} &= -\frac{a_1}{A_1}\sqrt{2gh_1} + \frac{a_3}{A_1}\sqrt{2gh_3} + \frac{\gamma_1}{A_1} k_1 v_1 \\
\frac{dh_2}{dt} &= -\frac{a_2}{A_2}\sqrt{2gh_2} + \frac{a_4}{A_2}\sqrt{2gh_4} + \frac{\gamma_2}{A_2} k_2 v_2 \\
\frac{dh_3}{dt} &= -\frac{a_3}{A_3}\sqrt{2gh_3} + \frac{(1-\gamma_2)}{A_3} k_2 v_2 \\
\frac{dh_4}{dt} &= -\frac{a_4}{A_4}\sqrt{2gh_4} + \frac{(1-\gamma_1)}{A_4} k_1 v_1
\end{aligned} \tag{I.15}$$

As illustrative example, the parameter values of the QTP laboratory process are given in the Table (I.1).

Table (I.1): Parameter values of the QTP process [3]

Symbol	Designation	Value
A_1, A_3	Cross-sectional area of tanks 1 and 3	28cm²
A_2, A_4	Cross-sectional area of tanks 2 and 4	32cm²
a_1, a_3	Cross-sectional areas of the outlet holes	0.071cm²
a_2, a_4	Cross-sectional areas of the outlet holes	0.057cm²
g	Gravity acceleration	981cm/s²
$k_{1,2}$	Level measurement devices constant	0.5

I.6 Quadruple Tank Process Control

The QTP model given by (I.15) can be rewritten in the following form:

$$\frac{dh}{dt} = f(h_1,...,h_n,v_1,v_2) = f(h,v),\ n = 4 \tag{I.16}$$

With,

$$h = [h_1,\ h_2,\ h_3,\ h_4]^T,\ f = [f_1,\ f_2,\ f_3,\ f_4]^T,\ \frac{dh_i}{dt} = f_i,\ i = 1,...,4 \text{ and } v = [v_1,\ v_2]^T$$

The lower two tanks levels are chosen to be the system outputs resulting in the following output vector:

$$y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} k_c & 0 & 0 & 0 \\ 0 & k_c & 0 & 0 \end{pmatrix} h \tag{I.17}$$

The equation (I.16) is nonlinear because of the square root terms that appear in (I.15), which makes the linear controller design difficult. To overcome the difficulty the linearization is required.

I.6.1 QTP Model Linearization

The system linearization can be performed by using Taylor series expansion based on the knowledgement of the system equilibrium point.

Assume that the system (I.16) operates on its equilibrium point h_e while it is driven by its equilibrium input v_e.

On the equilibrium point the following differential equation is satisfied:

$$\dot{h}_e = f(h_{1e},...,h_{ne};v_{1e},v_{2e}) = f(h_e,v_e) = 0 \tag{I.18}$$

Assume that the deviation of the nonlinear system is in the neighborhood of the equilibrium point, that is

$$\begin{aligned} h &= h_e + \Delta h \\ v &= v_e + \Delta v \end{aligned} \tag{I.19}$$

Where Δh and Δv represent small quantities.

For the system deviation in close proximity to the equilibrium point, we have:

$$\dot{h}_e + \dot{\Delta h} = f(h_e + \Delta h, v_e + \Delta v) \quad \text{(I.20)}$$

The right-hand side can be expanded into a Taylor series about the equilibrium point, which produces:

$$\dot{h}_e + \dot{\Delta h} = f(h_e, v_e) + \frac{df(h_e, v_e)}{dh}\Delta h + \frac{df(h_e, v_e)}{dv}\Delta v + \text{higher order terms} \quad \text{(I.21)}$$

By canceling higher-order terms, it results:

$$\dot{\Delta h} = \frac{df(h_e, v_e)}{dh}\Delta h + \frac{df(h_e, v_e)}{dv}\Delta v \quad \text{(I.22)}$$

Equation (I.21) represents a linear, time-invariant system since the derivative of Δh is linear combinations of the deviation state variable Δh and the deviation input Δv.

The partial derivatives in (I.22) represent the Jacobian matrices given by:

$$A = \frac{df(h_e, v_e)}{dh} = \begin{pmatrix} \frac{\partial f_1}{\partial h_1} & \frac{\partial f_1}{\partial h_2} & \frac{\partial f_1}{\partial h_3} & \frac{\partial f_1}{\partial h_4} \\ \frac{\partial f_2}{\partial h_1} & \frac{\partial f_2}{\partial h_2} & \frac{\partial f_2}{\partial h_3} & \frac{\partial f_2}{\partial h_4} \\ \frac{\partial f_3}{\partial h_1} & \frac{\partial f_3}{\partial h_2} & \frac{\partial f_3}{\partial h_3} & \frac{\partial f_3}{\partial h_4} \\ \frac{\partial f_4}{\partial h_1} & \frac{\partial f_4}{\partial h_2} & \frac{\partial f_4}{\partial h_3} & \frac{\partial f_4}{\partial h_4} \end{pmatrix} \text{ and } B = \frac{df(h_e, v_e)}{dv} = \begin{pmatrix} \frac{\partial f_1}{\partial v_1} & \frac{\partial f_1}{\partial v_2} \\ \frac{\partial f_2}{\partial v_1} & \frac{\partial f_2}{\partial v_2} \\ \frac{\partial f_3}{\partial v_1} & \frac{\partial f_3}{\partial v_2} \\ \frac{\partial f_4}{\partial v_1} & \frac{\partial f_4}{\partial v_2} \end{pmatrix} \quad \text{(I.23)}$$

A and B are the Jacobian matrices of f with respect to h and v, respectively evaluated at the equilibrium point (h_e, v_e) .In this case, the linearized system can be represented as:

$$\dot{\Delta h} = A\Delta h + B\Delta v \quad \text{(I.24)}$$

Equation (I.24) is called the Jacobian linearization of the original nonlinear system (I.16) about the equilibrium point (h_e, v_e).

Similarly, the output of a nonlinear system satisfies a nonlinear algebraic equation, that is:

$$y = g(h_1, h_2, \ldots.., h_4, v_1, v_2) = g(h, v) \quad \text{(I.25)}$$

With, $y = [y_1, y_2]^T$ and $g = [g_1, g_2]^T$

This equation can also be linearized by expanding its right-hand side into a Taylor series about the equilibrium point (h_e, v_e) .This leads to:

$$y_e + \Delta y = g(h_e, v_e) + \frac{dg(h_e, v_e)}{dh}\Delta h + \frac{dg(h_e, v_e)}{du}\Delta v + \text{higher order terms} \qquad (I.26)$$

Note that y_e cancels the term $g(h_e, v_e)$. By neglecting higher-order terms, the linearized part of the output equation is given by:

$$\Delta y = C\Delta h + D\Delta v \qquad (I.27)$$

The matrices C and D are the Jacobian matrices that satisfy:

$$C = \frac{dg(h_e, v_e)}{dh} = \begin{pmatrix} \frac{\partial g_1}{\partial h_1} & \frac{\partial g_1}{\partial h_2} & \frac{\partial g_1}{\partial h_3} & \frac{\partial g_1}{\partial h_4} \\ \frac{\partial g_2}{\partial h_1} & \frac{\partial g_2}{\partial h_2} & \frac{\partial g_2}{\partial h_3} & \frac{\partial g_2}{\partial h_4} \end{pmatrix} \text{and } D = \frac{dg(h_e, v_e)}{dv} = \begin{pmatrix} \frac{\partial g_1}{\partial v_1} & \frac{\partial g_1}{\partial v_2} \\ \frac{\partial g_2}{\partial v_1} & \frac{\partial g_2}{\partial v_2} \end{pmatrix} \qquad (I.28)$$

By applying (I.23) and (I.28), the Jacobian matrices are given below:

$$A = \begin{pmatrix} -\frac{1}{T_1} & 0 & \frac{A_3}{A_1 T_3} & 0 \\ 0 & -\frac{1}{T_2} & 0 & \frac{A_4}{A_2 T_4} \\ 0 & 0 & -\frac{1}{T_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{T_4} \end{pmatrix} \text{and } B = \begin{pmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2)k_2}{A_3} \\ \frac{(1-\gamma_1)k_1}{A_4} & 0 \end{pmatrix} \qquad (I.29)$$

$$C = \begin{pmatrix} k_c & 0 & 0 & 0 \\ 0 & k_c & 0 & 0 \end{pmatrix} \text{and D=0} \qquad (I.30)$$

I.6.2 QTP State Space Model

State space method generates a matrix from the system differential equations by making each order of the derivatives into a variable. The state space representation serves as an alternative to transfer function representation of a system so that a SISO or a MIMO process can be treated equally. The state-space representation is best suited both for the theoretical treatment of control systems and for numerical calculations. The determination of the system response in the homogeneous case with the initial condition x (t_0) is very simple. For these many advantages the state space representation is carried out for the quadruple tank process. The linearized state space equation of a quadruple tank process is given as:

$$\Delta \dot{h} = \begin{pmatrix} -\frac{1}{T_1} & 0 & \frac{A_3}{A_1 T_3} & 0 \\ 0 & -\frac{1}{T_2} & 0 & \frac{A_4}{A_2 T_4} \\ 0 & 0 & -\frac{1}{T_3} & 0 \\ 0 & 0 & 0 & -\frac{1}{T_4} \end{pmatrix} \Delta h + \begin{pmatrix} \frac{\gamma_1 k_1}{A_1} & 0 \\ 0 & \frac{\gamma_2 k_2}{A_2} \\ 0 & \frac{(1-\gamma_2) k_2}{A_3} \\ \frac{(1-\gamma_1) k_1}{A_4} & 0 \end{pmatrix} \Delta v \tag{I.31}$$

$$\Delta y = \begin{pmatrix} k_c & 0 & 0 & 0 \\ 0 & k_c & 0 & 0 \end{pmatrix} \Delta h$$

With $T_i = \frac{A_i}{a_i}\sqrt{\frac{2h_i^0}{g}}, i = 1,\ldots,4$

The equation above gives the state space model of the QTP, which is obtained from the developed mathematical model.

The model and control of the quadruple-tank process are studied at two operating points P-: at which the system will be shown to have minimum-phase characteristics, and P+ :at which it will be shown to have non-minimum-phase characteristics. The chosen operating points correspond to the following parameter values [3]:

Table (I.2): Chosen operating points[3]

Parameters	P-	P+
(h_1^0, h_2^0) [cm]	(12.4, 12.7)	(12.6, 13.0)
(h_3^0, h_4^0) [cm]	(1.8, 1.4)	(4.8, 4.9)
(v_1^0, v_2^0) [V]	(3.00, 3.00)	(3.15, 3.15)
(k_1^0, k_2^0) $[cm^3/Vs]$	(3.33, 3.35)	(3.14, 3.29)
(γ_1^0, γ_2^0)	(0.70, 0.60)	(0.43, 0.34)

- For minimum phase, the QTP model is defined by the following matrices:

$$A_{\_} = \begin{pmatrix} -0.016 & 0 & 0.041 & 0 \\ 0 & -0.011 & 0 & 0.033 \\ 0 & 0 & -0.041 & 0 \\ 0 & 0 & 0 & -0.033 \end{pmatrix}, \; B_{\_} = \begin{pmatrix} 0.083 & 0 \\ 0 & 0.062 \\ 0 & 0.047 \\ 0.031 & 0 \end{pmatrix}, \; C_{\_} = \begin{pmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \end{pmatrix} \tag{I.32}$$

- For non-minimum phase, the QTP model is defined by the following matrices:

$$A_+ = \begin{pmatrix} -0.016 & 0 & 0.026 & 0 \\ 0 & -0.011 & 0 & 0.018 \\ 0 & 0 & -0.026 & 0 \\ 0 & 0 & 0 & -0.018 \end{pmatrix}, B_+ = \begin{pmatrix} 0.048 & 0 \\ 0 & 0.035 \\ 0 & 0.078 \\ 0.056 & 0 \end{pmatrix}, C_+ = \begin{pmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \end{pmatrix} \quad \text{(I.33)}$$

I.6.3 QTP Transfer Function Model

The transfer function method applies a Laplace transformation to the differential equations, which allows handling them as a single algebraic equation [10].The key advantage of transfer functions is in their compactness, which makes them suitable for frequency-domain analysis and stability studies. However, the transfer function approach suffers from neglecting the initial conditions. To determine the transfer matrix for the QTP the following formula is used:

$$G(s) = C(sI - A)^{-1}B \quad \text{(I.34)}$$

By replacing equations (I.28) and (I.29) into (I.34), it results:

$$G(s) = \begin{pmatrix} \dfrac{c_1\gamma_1}{(1+sT_1)} & \dfrac{c_1k_2(1-\gamma_2)}{k_1(1+sT_1)(1+sT_3)} \\ \dfrac{c_2k_1(1-\gamma_1)}{k_2(1+sT_2)(1+sT_4)} & \dfrac{c_2\gamma_2}{(1+sT_2)} \end{pmatrix} \quad \text{(I.35)}$$

Knowing that the ratio $\frac{k_1}{k_2}$ and $\frac{k_2}{k_1}$ are approximately equal to 1, the corresponding transfer matrix is:

$$G(s) = \begin{pmatrix} \dfrac{c_1\gamma_1}{(1+sT_1)} & \dfrac{c_1(1-\gamma_2)}{(1+sT_1)(1+sT_3)} \\ \dfrac{c_2(1-\gamma_1)}{(1+sT_2)(1+sT_4)} & \dfrac{c_2\gamma_2}{(1+sT_2)} \end{pmatrix} \quad \text{(I.36)}$$

Where $c_1 = \frac{T_1k_ck_1}{A_1}$ and $c_2 = \frac{T_2k_ck_2}{A_2}$.

- For minimum phase case, the transfer matrix becomes:

$$G_{-}(s)=\begin{pmatrix}\frac{2.6}{1+62s} & \frac{1.5}{(1+23s)(1+62s)}\\ \frac{1.4}{(1+30s)(1+90s)} & \frac{2.8}{1+90s}\end{pmatrix} \tag{I.37}$$

- For non-minimum phase, the transfer matrix becomes:

$$G_{+}(s)=\begin{pmatrix}\frac{1.5}{1+63s} & \frac{2.5}{(1+39s)(1+63s)}\\ \frac{2.5}{(1+56s)(1+91s)} & \frac{1.6}{1+91s}\end{pmatrix} \tag{I.38}$$

I.6.3.1 Interpretation of Multivariable Zero

The transfer matrix G has two zeros one of them is always in the left half of S-plane, but the other can be located either in left half or right half of S-plane based on the position of three way valves [10].

I.6.3.1.1 Zeros Location

The zeros of the transfer matrix in (I.36) are the zeros of the numerator polynomial of the rational function.

$$\det G(s)=\frac{c_1 c_2}{\gamma_1\gamma_2\prod_{i=1}^{4}(1+sT_i)}\left[(1+sT_3)(1+sT_4)-\frac{(1-\gamma_1)(1-\gamma_2)}{\gamma_1\gamma_2}\right] \tag{I.39}$$

The transfer matrix G thus has two finite zeros for $\gamma_1,\gamma_2\in(0,1)$. One of them is always in the left half-plane, but the other can be located either in the left or the right half-plane, as follows from the following root-locus argument. Let us introduce a parameter $\eta\in(0,\infty)$ as:

$$\eta=\frac{(1-\gamma_1)(1-\gamma_2)}{\gamma_1\gamma_2} \tag{I.40}$$

If η is small, the two zeros are close to $-\frac{1}{T_3}$ and $-\frac{1}{T_4}$, respectively. Furthermore, one zero tends to $-\infty$ and one zero tends to $+\infty$ as $\eta\to\infty$. If $\eta=1$ one zero is located at the origin. This case corresponds to $\gamma_1+\gamma_2=1$. It follows that the system is non-minimum phase for $0<\gamma_1+\gamma_2<1$ and minimum phase for $1<\gamma_1+\gamma_2<2$.

Recall that $\gamma_1+\gamma_2=1.30>1$ for P_ and $\gamma_1+\gamma_2=0.77<1$ for P+. Figure (I.5) shows the location of the two operating points P_ and P+. For operating points above the solid line the system is minimum phase and below it is non-minimum phase [3].

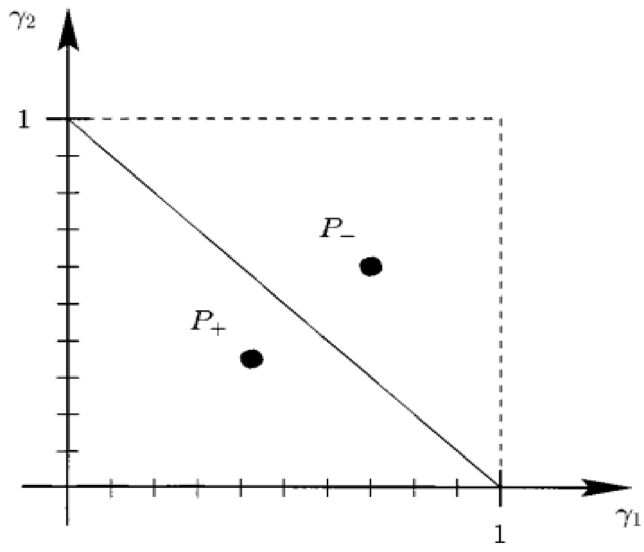


Figure (I.5): Depending on the values of the valve parameters γ_1 and γ_2 the system is minimum or non-minimum phase [3]

The multivariable zero being in the left or in right half-plane has a straightforward physical interpretation. Let q_i denote the flow through the pump *i* and assume that $q_1=q_2$.Then the sum of the flows to the upper tanks is $[2-(\gamma_1+\gamma_2)]q_1$ and the sum of the flows to the lower tanks is $(\gamma_1+\gamma_2)q_1$. Hence, the flow to the lower tanks is greater than the flow to the upper tanks if $\gamma_1+\gamma_2>1$, i.e., if the system is minimum phase. The flow to the lower tanks is smaller than the flow to the upper tanks if the system is non-minimum phase. It is intuitively easier to control y_1 with v_1 and y_2 with v_2 if most of the flows goes directly to the lower tanks. The control problem is particularly hard if the total flow going to the left tanks (Tanks 1 and 3) is equal to the total flow going to the right tanks (Tanks 2 and 4). This corresponds to $\gamma_1+\gamma_2=1$, i.e., a multivariable zero in the origin. There is thus an immediate connection between the zero location of the model and physical intuition of controlling the quadruple-tank process [3].

I.6.3.1.2 Zeros Direction

An important difference between scalar systems and multivariable systems is that not only the location of a multivariable zero is important but also their direction. We define the (output) direction of a zero z as a vector $\varphi\in\mathbb{R}^2$ of unit length such as $\varphi^T G(z)=0$. If φ is parallel to a unit vector, then the zero is only associated with one output. If this is not the case, then the effect of a right half-plane zero may be distributed between both outputs. For the transfer matrix G in (I.35), the zero direction is given by:

$$\begin{bmatrix}\varphi_1\\ \varphi_2\end{bmatrix}^T \begin{pmatrix} \frac{c_1\gamma_1}{(1+zT_1)} & \frac{c_1(1-\gamma_2)}{(1+zT_1)(1+zT_3)} \\ \frac{c_2(1-\gamma_1)}{(1+zT_2)(1+zT_4)} & \frac{c_2\gamma_2}{(1+zT_2)} \end{pmatrix} = \begin{bmatrix}0\\0\end{bmatrix}^T \tag{I.41}$$

Note that it follows from this equation that $\varphi_1, \varphi_2 \neq 0$, so the zero is never associated with only one output. If we solve (I.41) for γ_2 and simplify, it is easy to show that:

$$\frac{\varphi_1}{\varphi_2} = -\frac{1-\gamma_1}{\gamma_1}\frac{c_2(1+zT_1)}{c_1(1+zT_2)(1+zT_4)} \tag{I.42}$$

From this equation it is possible to conclude that if γ_1 is small, then z is mostly associated with the first output. If γ_1 is close to one, then z is mostly associated with the second output. Hence, for a given zero location, the relative size of γ_1 and γ_2 determines which output the right half-plane zero is related to [3].

I.6.3.2 Relative Gain Array

The relative gain array (RGA) was introduced by Bristol [9] as a measure of interaction in multivariable control systems. The RGA is defined as $\Lambda = G(0)G^{-T}(0)$,where the asterisk denotes the Schur product (element-by-element matrix multiplication) and -T inverse transpose. It is possible to show that the elements of each row and column of $\mathbf{\Lambda}$ sum up to one, so for a 2x2 system the RGA is determined by the scalar $\lambda = \Lambda_{11}$. The RGA is used as a tool mainly in the process industry to decide on control structure issues such as input–output pairing for decentralized controllers. McAvoy [11] proposed that one should strive for a pairing with $0.67 < \lambda < 1.5$. The system is particularly hard to control if $\lambda < 0$.

The RGA of the quadruple-tank process is given by the simple expression,

$$\lambda = \frac{\gamma_1\gamma_2}{\gamma_1+\gamma_2-1} \tag{I.43}$$

Note that the RGA is only depending on the valve settings and no other physical parameters of the process. The magnitude of λ increases as $\gamma_1+\gamma_2$ becomes close to one. There are no $\gamma_1, \gamma_2 \in (0,1)$ such that $\lambda \in (0,1)$. From (I.40) we see for instance that $\lambda < 0$ if $\gamma_1+\gamma_2 < 1$, which corresponds to the non-minimum-phase setting discussed previously.

If the valves of the quadruple-tank process are set such that $\gamma_1 + \gamma_2 < 1$, then the RGA analysis suggests that another input–output pairing for decentralized control should be chosen.

Let

$$\tilde{G} = \begin{pmatrix} G_{21} & G_{22} \\ G_{11} & G_{12} \end{pmatrix} \tag{I.44}$$

be the linearized model with y_1 and y_2 permuted. The RGA for $\tilde{G}$ is:

$$\tilde{\lambda} = \frac{(1-\gamma_1)(1-\gamma_2)}{1-\gamma_1-\gamma_2} \tag{I.45}$$

Hence, if $\gamma_1 + \gamma_2 < 1$ then $\tilde{\lambda} > 0$ so a decentralized control structure corresponding to $\tilde{G}$ is preferable according to the RGA. This is an intuitive from physical considerations.

We sum up the physical interpretations of the zero as well as of the RGA and give the corresponding values for the real process. One zero of G is always in the left half-plane. The other can be located anywhere on the real axis. The location is determined by how the valves corresponding to the parameters γ_1 and γ_2 are adjusted. If $\gamma_1 + \gamma_2 \in (0,1)$ then the zero is in the right half-plane, while if $\gamma_1 + \gamma_2 \in (1,2)$ the zero is in the left half-plane. The relative size of γ_1 and γ_2 gives the zero direction γ_1/γ_2, i.e., tells which output the effect of the zero is associated to. If γ_1/γ_2 is small, then the zero is associated with output one and vice versa. The RGA of the quadruple-tank process is given by $\frac{\gamma_1\gamma_2}{\gamma_1+\gamma_2-1}$ and is thus only dependent on the valve settings.

The zeros of two operating points P_ and P+ are given in Table (I.3).

Table (I.3): Zeros of the two operating points P_ and P+

	P-	P+
Zeros	(-0.06, -0.018)	(-0.057, 0.013)

Hence, G+ has a zero in the right half-plane, which deteriorates the performance considerably. The direction of the zero z = 0.013 is given by $\frac{\varphi_1}{\varphi_2} = -0.85$ .The RGA for P_ is given by $\lambda = 1.4$ and for P+ by $\lambda = -0.64$. RGA analysis indicates that the non-minimum-phase system is harder to control than the minimum-phase system. For the system $\tilde{G}$ (G with permuted outputs) the RGA is $\tilde{\lambda} = -0.4$ for the minimum-phase setting and $\tilde{\lambda} = 1.64$ for the non-minimum-phase setting [3].

I.7 Controller Design

For the quadruple tank process, two PI controllers are needed to be designed because we have two inputs and two outputs as shown in Figure (I.6).The proportional-integral control mode is used in applications where load disturbances occur frequently and set point changes are infrequent. It is also used when load changes are slow to allow enough time to elapse before it is necessary for the integral function to aid the proportional operation [11].

The mathematical formula of each controller is:

$$C_j(s) = k_{pj}(1 + \frac{1}{T_{ij}s}), \quad j=1,2 \tag{I.46}$$

Where k_{pj} is the proportional gain and T_{ij} is the integral time constant.

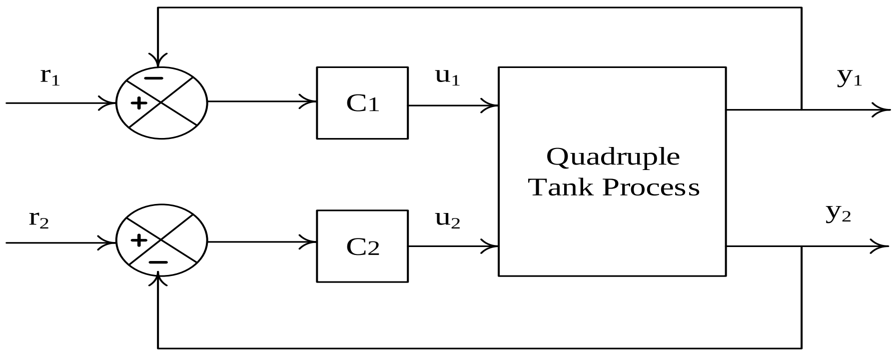


Figure (I.6): Decentralized control structure with two PI controllers of QTP system

The objective is to find each controller parameters k_p and k_i $(k_i = \frac{k_p}{T_i})$ that helps the system to reach the set-point by using of tuning parameters rules such as Ziegler-Nichols and Pole-assignment design techniques.

I.7.1 Pole-Assignment Design Techniques

In the general case of using Pole-Assignment Design Techniques, a transfer function is given by,

$$G(s) = \frac{b}{s+a} \tag{I.47}$$

For the quadruple tank process we used the following transfer functions:

$$\frac{h_1}{u_1} = G_{11}(s) = \frac{2.6}{\frac{1}{62} + s}$$
$$\frac{h_2}{u_2} = G_{22}(s) = \frac{2.8}{\frac{1}{90} + s} \tag{I.48}$$

The feedback PI control system is illustrated in Figure (I.7), where R(s), E(s), U(s) and Y(s) denote the Laplace transforms of the reference signal, error signal, control signal and output signal, respectively.

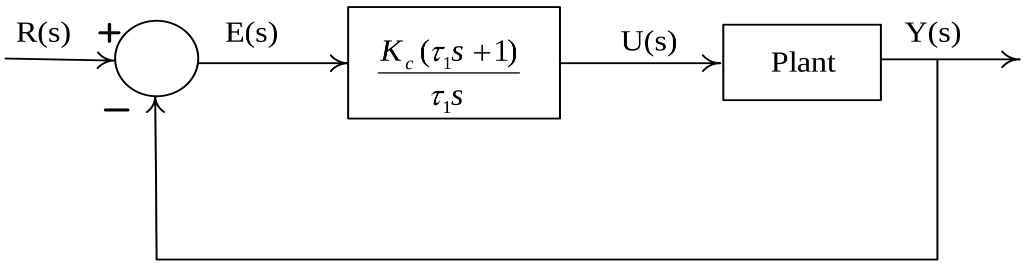


Figure (I.7): Block diagram of PI control system

To make the design simpler, the PI controllers are written in the following transfer functions form:

$$C_1(s) = \frac{c_1 s + c_0}{s}$$
$$C_2(s) = \frac{c_2 s + c_0^{'}}{s} \tag{I.49}$$

Where $k_{p_1} = c_1$, $T_{i_1} = \frac{c_1}{c_0}$ and $k_{p_2} = c_2$, $T_{i2} = \frac{c_2}{c_0^{'}}$

We will first find the coefficients c_1, c_0 and $c_2, c_0^{'}$ based on the model (I.48), then we convert these coefficients into the standard PI controller parameters k_p and T_i.

The closed-loop transfer function from the reference signal to the output signal is expressed as:

$$\frac{Y_1(S)}{U_1(s)} = \frac{2.6(c_1 s + c_0)}{s(s + \frac{1}{62}) + 2.6(c_1 s + c_0)}$$
$$\frac{Y_2(S)}{U_2(s)} = \frac{2.8(c_2 s + c_0^{'})}{s(s + \frac{1}{90}) + 2.8(c_2 s + c_0^{'})} \tag{I.50}$$

The closed-loop poles of the feedback system are the solutions of the polynomial equations with respect to s, that are:

$$s(s+\frac{1}{62})+2.6(c_1 s+c_0)=0$$
$$s(s+\frac{1}{90})+2.8(c_2 s+c_0^{'})=0 \tag{I.51}$$

Equation (I.51) is called a closed-loop characteristic equation. It is well understood that the locations of the closed-loop poles determine the closed-loop stability as well as its response speed to reference signal and disturbance rejection.

Since the model parameters *a* and *b* are given, the free parameters in (I.51) are the controller parameters c_1, c_0 and $c_2, c_0^{'}$. To find the controller parameters c_1, c_0 and $c_2, c_0^{'}$, the following polynomial equations are set:

$$s(s+\frac{1}{62})+2.6(c_1 s+c_0)=s^2+2\xi_1\omega_{n_1}s+\omega_{n_1}^2$$
$$s(s+\frac{1}{90})+2.8(c_2 s+c_0^{'})=s^2+2\xi_2\omega_{n_2}s+\omega_{n_2}^2 \tag{I.52}$$

Where the left-hand side of equations (I.52) is the characteristic polynomial that determines the actual closed-loop poles with the controller and the right-hand side is the desired closed-loop characteristic polynomial that determines the desired closed-loop poles. By equating these two polynomials, the actual closed-loop poles are assigned to the desired closed-loop poles. This controller design technique is called pole-assignment controller design [14].

Now, we compare the coefficients of the polynomial equations (I.52) on both sides. With the deliberately chosen model structure and the controller structure, the coefficients for s^2 on both sides of (I.52) are equal to 1. Equating the coefficient of s on the left-hand side to the one on the right-hand side gives:

$$\frac{1}{62}+2.6c_1=2\xi\omega_{n_1}$$
$$\frac{1}{90}+2.8c_2=2\xi\omega_{n_2} \tag{I.53}$$

The same procedure is applied to the constant term, leading to:

$$2.6c_0=\omega_{n_1}^2$$
$$2.8c_0^{'}=\omega_{n_2}^2 \tag{I.54}$$

Solving (I.53),(I.54) gives:

$$c_1 = \frac{2\xi_1\omega_{n_1} - \frac{1}{62}}{2.6} \quad \text{and} \quad c_2 = \frac{2\xi_2\omega_{n_2} - \frac{1}{90}}{2.8} \qquad \text{(I.55)}$$
$$c_0 = \frac{\omega_{n_1}^2}{2.6} \qquad c_0^{'} = \frac{\omega_n^2}{2.8}$$

The relationships between $c_1, c_0, c_2, c_0^{'}$ and the PI controller parameters k_p, T_i, are found as:

$$k_{c_1} = c_1 = \frac{2\xi_1\omega_{n_1} - \frac{1}{62}}{2.6} \quad \text{and} \quad k_{c_2} = c_2 = \frac{2\xi_2\omega_{n_2} - \frac{1}{90}}{2.8} \qquad \text{(I.56)}$$
$$T_{i_1} = \frac{c_1}{c_0} = \frac{2\xi_1\omega_{n_1} - \frac{1}{62}}{\omega_{n_1}^2} \qquad T_{i_2} = \frac{c_2}{c_0^{'}} = \frac{2\xi_2\omega_{n_2} - \frac{1}{90}}{\omega_{n_2}^2}$$

I.7.2 Ziegler-Nichols Frequency Response Method

The Ziegler-Nichols frequency response method is a closed-loop tuning method. This method first determines the point where the Nyquist curve of the plant *G(s)* intersects the negative real axis. It can be obtained experimentally in the following way: turn the integral and differential actions off and set the controller to be in the proportional mode only and close the loop. Slowly increase the proportional gain k_p until a periodic oscillation in the output is observed. This critical value of k_p is called the *ultimate gain* (k_u). The resulting period of oscillation is referred to as the *ultimate period* (p_u). Based on k_u and p_u, the Ziegler-Nichols frequency response method gives the following simple formulas for setting PID controller parameters [8]:

Table (I.4): Ziegler–Nichols tuning rule based on ultimate gain and ultimate period values

Type of controller	k_p	T_i	T_d
P	$0.5k_u$	∞	0
PI	$0.45k_u$	$p_u/1.2\,p_u$	0
PID	$0.6k_u$	$0.5p_u$	$0.125p_u$

Ziegler–Nichols tuning rules can, of course, be applied to plants whose dynamics are known.
We applied this method to find the PI controllers parameters for non-minimum phase based on the following transfer function,

$$\frac{h_1}{u_2} = G_{12}(s) = \frac{2.5}{\left(\frac{1}{39}+s\right)\left(\frac{1}{63}+s\right)}$$
$$\frac{h_2}{u_1} = G_{21}(s) = \frac{2.5}{\left(\frac{1}{56}+s\right)\left(\frac{1}{91}+s\right)} \qquad \text{(I.57)}$$

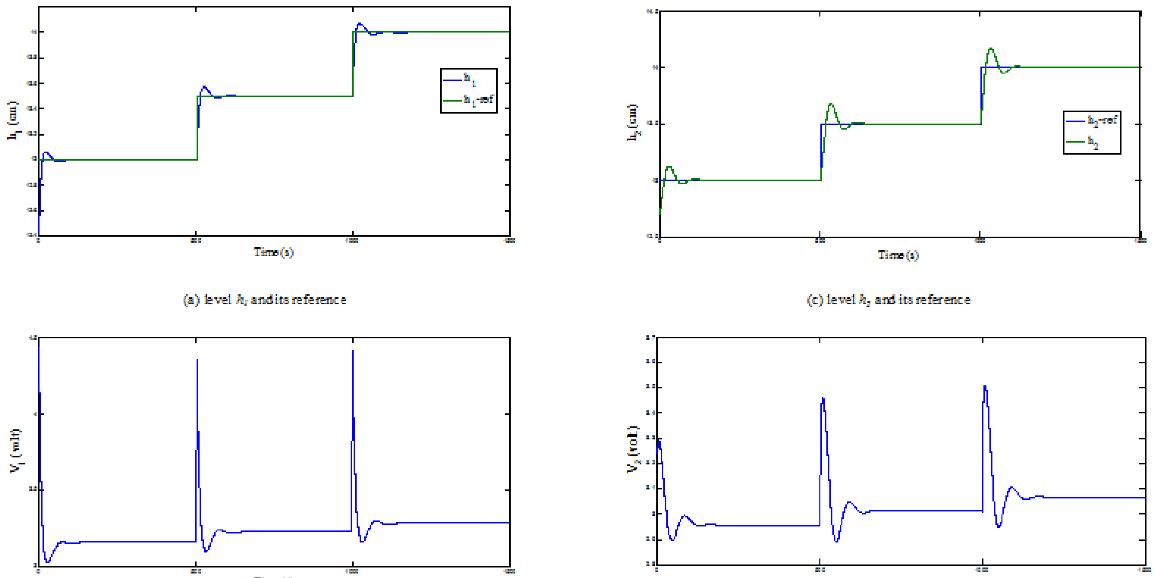
I.8 Simulation Results

In this section, simulations are conducted to validate the performance of the designed PI controllers applied on the quadruple-tank process. Separate simulations are performed for both minimum-phase and non-minimum phase systems.

I.8.1 Simulation Results for Minimum Phase

Pole-assignment method is used to find the following controller parameters for minimum-phase model: $k_{p1} = 3,\ k_{i1} = 0.33$ and $k_{p2} = 1.4,\ k_{i2} = 0.125$.

The closed loop responses of water levels in tanks 1 and 2 for minimum phase with PI controllers of QTP system are shown in Figure (I.8). In this figure the levels references value are step changed from 13 cm to 14 cm.



(a) level h_1 and its reference (c) level h_2 and its reference

(b) first controller output v_1 (d) second controller output v_2

Figure (I.8): Minimum phase performances of QTP controlled by PI controllers.

I.8.2 Simulation Results for Non-Minimum Phase

Ziegler–Nichols is used to find the following controller parameters for non-minimum-phase model: $k_{p1} = 2.5,\ k_{i1} = 0.12$ and $k_{p2} = 1.12,\ k_{i2} = 0.08$.

The closed loop responses of water levels in tanks 1 and 2 for non-minimum phase with PI controllers of QTP system are shown in Figure (I.9). In this figure the levels references value are step changed from 13 cm to 14 cm for h_1 and from 13.5 cm to 14.5 cm for h_2.

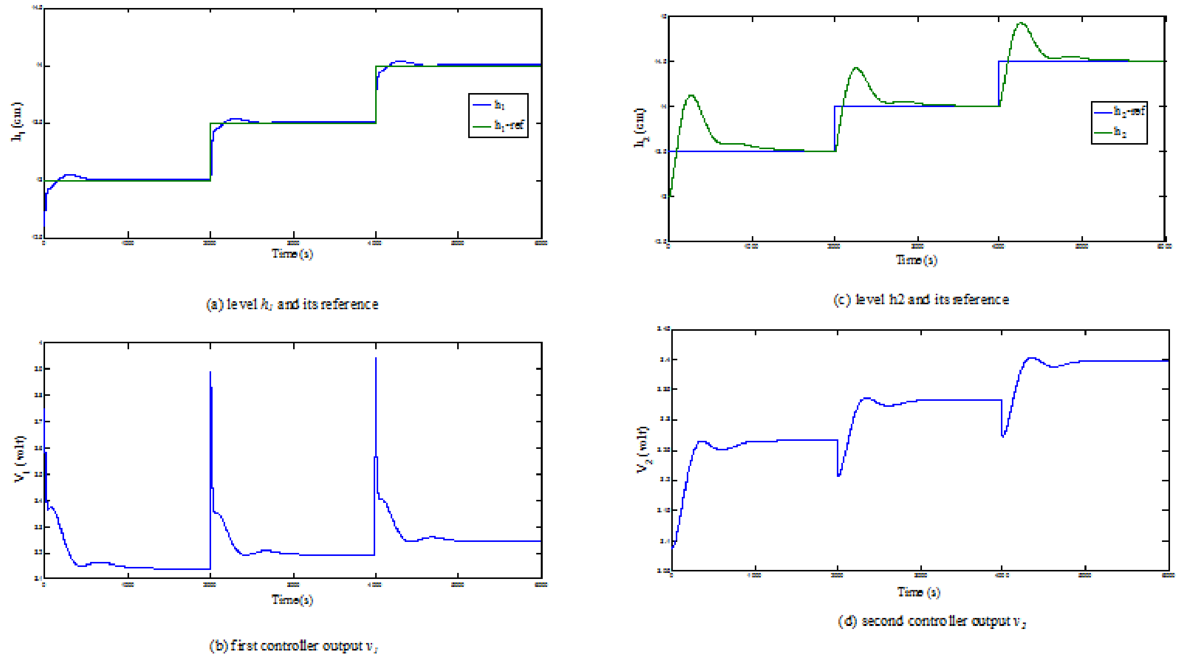


Figure (I.9) Non-minimum phase performances of QTP controlled by PI controllers

I.8.3 Discussion of Results

This section discusses the simulation studies carried out on the quadruple-tank process.

I.8.3.1 Minimum Phase Case

To test the robustness of the PI based control system, the tank 1 and tank 2 water levels setting points are changed between the following values 13 cm, 13.5 cm, and 14 cm.

Figure (I.8) shows the simulation results obtained by conventional PI algorithm applied on quadruple tank process operating in minimum phase. It can be seen from this figure that the water levels track their desired reference signals during steady state regime. Also, the input voltages stay within reasonable ranges. However, due to the inherent interactions between thanks, oscillations appear in the outputs responses.

I.8.3.2 Non-Minimum Phase Case

For the non-minimum phase operating condition, the adopted set points change between the following values 13 cm, 13.5 cm, and 14 cm.

Figure (I.9) shows the simulation results of the thanks water levels and input voltages to the corresponding pumps. From this figure, we can see that the curves of the liquid levels superimpose on their references after a noticeable transient regimes.

The simulated outputs of the quadruple tank process controlled by two PI controllers in non-minimum phase shows that the tank 1 and 2 responses are not good enough. The controllers faced difficulties to take action and the performance of the controllers are not satisfactory. Figure (I.9) clearly shows that there is a strong interaction between the tanks' levels, which impacts in its turn the obtained tanks responses.

From the obtained simulation results, the PI responses show acceptable performances in tracking but still not enough. Indeed, they present overshoots, and they are not so good in particular for non-minimum phase case in terms of setting time and precision. Indeed, the response and settling times for the non-minimum phase process are longer than those of minimum phase. Actually, there is a clear difference in the process behavior for both phases, where the PI controllers handle better the system in the minimum phase than the case of non-minimum phase.

I.9 Conclusion

The objective of this chapter was to control a quadruple tank process in its both minimum and non-minimum phases. The synthesis of such control algorithm requires an analytical description of the controlled system by an appropriate mathematical model. This point is achieved by defining a state space model as well as transfer functions for the QTP system, which makes possible the PI controllers design.

The obtained simulation results clearly show that the conventional control is able to control the QTP system. However, to boost up the control performances a modern nonlinear control method seems to be indispensable. For this reason, the next chapter will be devoted to fuzzy control of the QTP system.

Chapter II

Fuzzy Control of Quadruple Tank Process

II.1 Introduction

Fuzzy logic is a well-established mathematical theory that has become an essential technique in many areas including control engineering, pattern recognition, signal processing, and energy management in wireless sensor networks and robotics [12].

Fuzzy logic can be seen as an extension of classical or binary logic. However, its concept is much closer in spirit to human thinking and natural language than the traditional logical systems. Basically, it provides an effective means of capturing the approximate, inexact nature of the real world.

It can be used in both modeling and control of dynamic systems. The fuzzy logic system is a kind of expert knowledge based system that contains the encoded rule base derived from human experience and from theoretical and practical understanding of the dynamics of controlled system. The main advantage of fuzzy logic based control scheme is the ability to control complex dynamic systems without precise knowledge of their mathematical models [12].

The essential part of the fuzzy logic controller (FLC) is a set of linguistic control rules related by the dual concepts of fuzzy implication and the compositional rule of inference. In summary, the FLC provides an algorithm, which can convert the linguistic control strategy based on expert knowledge into an automatic control strategy [13].

In this chapter, fuzzy logic control of the quadruple tank process will be presented. First, the theoretical background of fuzzy logic will be provided. After that, the design of fuzzy logic controllers for quadruple tank process will be explained. At the end, simulation results will be illustrated and discussed.

II.2 Fuzzy Logic Theory

One of the characteristics of human reasoning is that it is generally based on imprecise or even incomplete data. In fact, the knowledge we have on any system is generally uncertain, either because we have doubts about its validity or we have difficulty expressing it clearly.

It is therefore necessary to think and develop a new type of approximate reasoning, which will allow mathematical treatment of the imprecise and the uncertain. The first to have underlined these possibilities of development is Lotfi A. Zadeh who from 1965introduced the theory of fuzzy logic [12].It is a technique for handling imprecise and uncertain knowledge. It makes it possible to take into account linguistic variables whose values are words or expressions of natural language, such as fast, slow, large, small, etc.

II.2.1 Fuzzy Sets

The most commonly used range of values of membership functions is the unit interval [0, 1]. In this case, each membership function maps elements of a given universal set ***X***, which is always a crisp set, into real numbers in [0, 1]. Two distinct notations are most commonly employed in the literature to denote membership functions. In one of them, the membership function of a fuzzy set ***A*** is denoted by μ_A [24], that is:

$$\mu_A : X \rightarrow [0,1] \tag{II.1}$$

A fuzzy set may be viewed as a generalization of the concept of an ordinary set whose membership function only takes two values (0,l). Thus a fuzzy set ***A*** in ***X*** may be represented as a set of ordered pairs of a generic element ***X*** and its grade of membership function:

$$A = \{(u, \mu_A(u) \mid u \in X)\} \tag{II.2}$$

When ***X*** is continuous, a fuzzy set ***A*** can be written concisely as:

$$A = \int_X \frac{\mu_A(u)}{u} \tag{II.3}$$

When ***X*** is discrete, a fuzzy set ***A*** is represented as:

$$A = \sum_{i=1}^{n} \mu_A(u_i)/u_i \tag{II.4}$$

II.2.2 Set Theoretic Operations

Let *A* and *B* be two fuzzy sets in *X*; μ_A and μ_B represents their membership functions, respectively. The set theoretic operations of union, intersection and complement for fuzzy sets are defined through their membership functions [12]. More specifically, see the following,

$$\mu_{A\cup B}(u) = \max\{\mu_A(u), \mu_B(u)\} \quad \text{(II.5)}$$

And:

$$\mu_{A\cap B}(u) = \min\{\mu_A(u), \mu_B(u)\} \quad \text{(II.6)}$$

And:

$$\mu_{\bar{A}}(u) = 1 - \mu_A(u) \quad \text{(II.7)}$$

II.3 Fuzzy Logic System Architecture

Fuzzy logic system (or, simply, fuzzy system, FS) architecture includes three parts as shown in figure (II.1) [13]:

1. Fuzzification is a block that transforms the crisp input values to fuzzy ones;
2. Fuzzy inference system (FIS) is a block that evaluates which control rules are appropriate at each time by using the fuzzy if-then rule base created by the user;
3. Defuzzification is a block that converters the fuzzy set into a crisp output value.

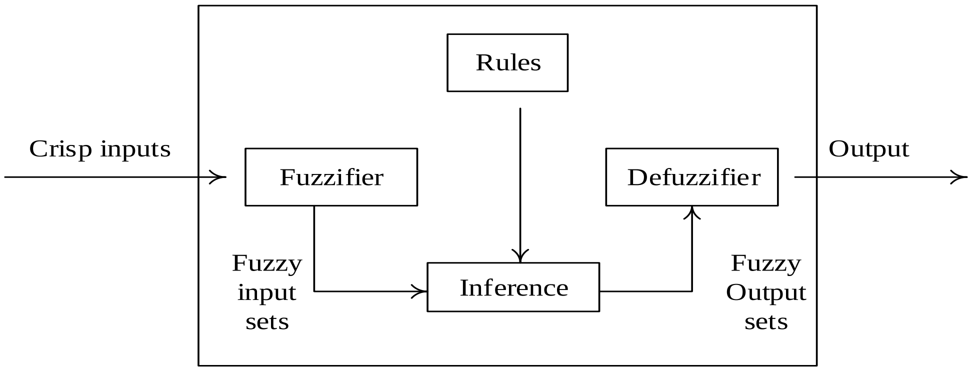


Figure (II.1): Structure of fuzzy logic System

II.3.1 Fuzzification

Fuzzification is defined as the process of transforming a crisp set to a fuzzy set or a fuzzy set to fuzzier set. Basically, this operation translates accurate crisp input values into linguistic variables.

During fuzzification, a fuzzy logic system receives inputs, and analyzes them according to user-defined membership functions. A membership function is a graphical representation of the degree of belonging of an element to the fuzzy set. The FS assigns each input a grade from 0 to 1 based on how well it fits into each membership function. Generally used membership functions are triangular, trapezoidal, and Gaussian, which are defined by the following expressions:

Triangular function is expressed by:

$$\mu(x) = \begin{cases} \dfrac{x-a}{b-a} & a < x \le b \\ \dfrac{c-x}{c-b} & b < x \le c \\ 0 & \text{otherwise} \end{cases} \tag{II.8}$$

Trapezoidal function is given by:

$$\mu(x) = \begin{cases} \dfrac{x-a}{b-a} & a < x \le b \\ 1 & b < x \le c \\ \dfrac{d-x}{d-c} & c < x \le d \\ 0 & \text{otherwise} \end{cases} \tag{II.9}$$

Gaussian function is defined by:

$$\mu(x) = e^{\left(\left(\frac{x-m}{\delta}\right)^2\right)} \quad , \quad -\infty < x < \infty \tag{II.10}$$

Each fuzzy system input can have several membership functions. Each of them is defined by a name called a label (linguistic term) such as NL (negative large), NM (negative medium), NS (negative small), ZR (zero), PS (positive small), and PM (positive medium).

II.3.2 Fuzzy Rule Base and Inference Engine

FS uses a reasoning process composed of IF...THEN rules, each providing an outcome. A rule is activated, if an input condition satisfies the IF part of the rule statement. This results in an output based on the THEN part of the rule statement. In a fuzzy logic system, many rules may exist, corresponding to one or more IF conditions. A rule may also have several input conditions, which are logically linked in either an AND or an OR relationship to activate the rule’s outcome.

In the case of MISO (multiple input single output), they are characterized as a collection of rules of the following form (case of two inputs and one output):

$$
\begin{aligned}
&R_1\text{: if x is } A_1, \ldots \text{ and y is } B_1 \text{ then z is } C_1 \\
&R_2\text{: if x is } A_2, \ldots \text{ and y is } B_2 \text{ then z is } C_2 \\
&\ldots \\
&R_n\text{: if x is } A_n, \ldots \text{ and y is } B_n \text{ then z is } C_n
\end{aligned} \tag{II.11}
$$

where x, y and z are linguistic variables representing the FS inputs and output, respectively. A_i, B_i and C_i are linguistic values of the variables x, y and z in the universe of discourse U,V and W, respectively with i = 1, 2, … , n.

Fuzzy inference is the combination of inputs, output membership functions and fuzzy rules resulting in fuzzy output. There are two main types of Fuzzy Inference Systems (FIS): the *Mamdani-type* and the *Takagi-Sugeno-Kang-type*.

II.3.2.1 Mamdani Fuzzy Inference Systems

The most commonly used fuzzy inference technique is the so-called *Mamdani method*, which was proposed first by *Mamdani* and *Assilian* [13]. In a Mamdani system, the output of each rule is a fuzzy set. Since Mamdani systems have more intuitive and easier to understand rule bases, they are well-suited to expert system applications where the rules are created from human expert knowledge.

The inference process of a Mamdani system is summarized in figure (II.2).

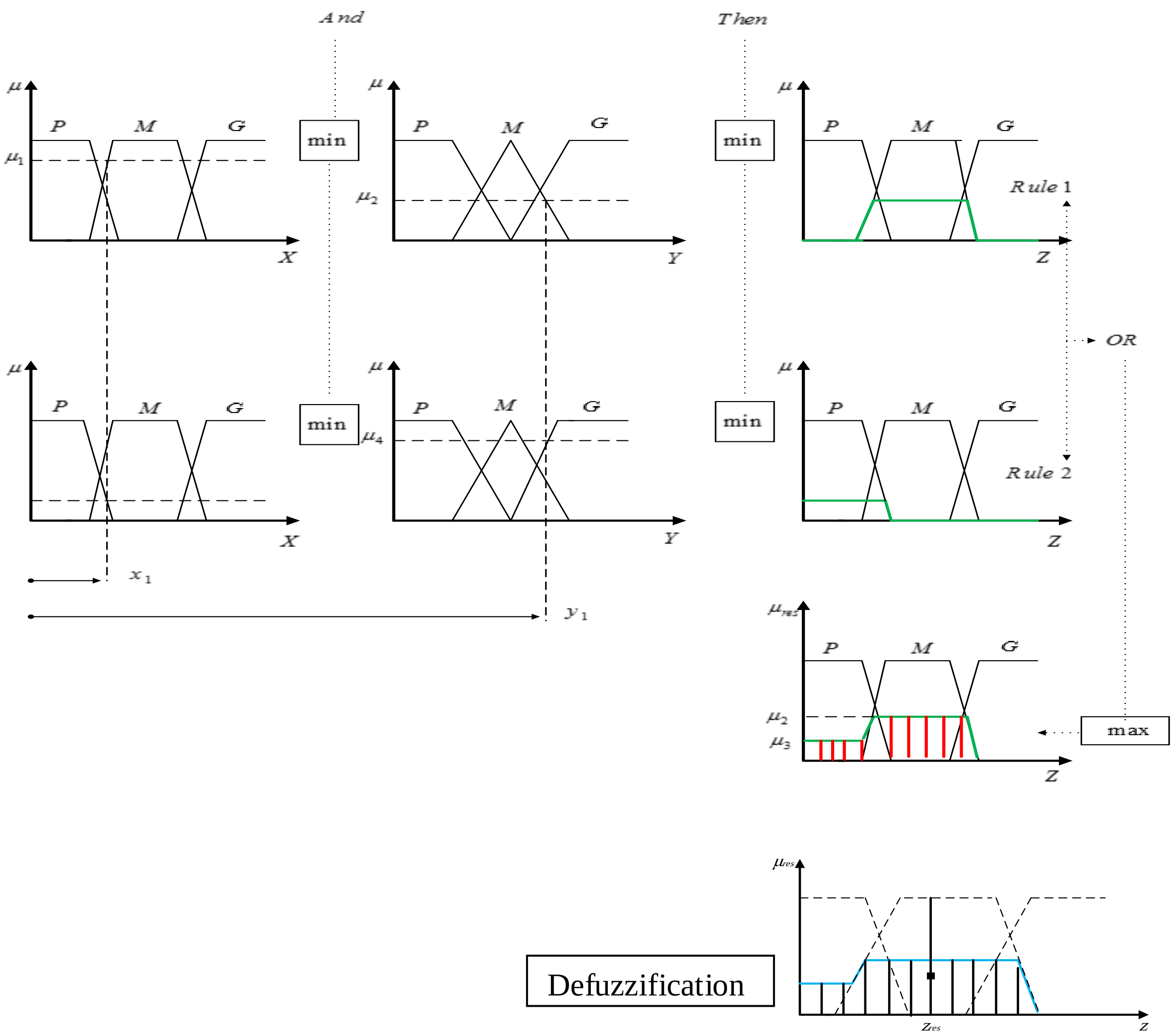


Figure (II.2): Mamdani's Min–Max inference method

In Mamdani's model the fuzzy implication is modeled by Mamdani's minimum operator, the conjunction operator is *min*, the t-norm from compositional rule is *min* and for the aggregation of the rules the *max* operator is used [13].

The output of each rule is a fuzzy set derived from the output membership function and the implication method of the FIS. These output fuzzy sets are combined into a single fuzzy set using the aggregation method of the FIS. Then, to compute a final crisp output value, the combined output fuzzy set is defuzzified using one of the *Defuzzification Methods.*

II.3.2.2 Takagi-Sugeno-Kang Fuzzy Inference System

The principle of Takagi-Sugeno-Kang fuzzy inference, also referred to as Sugeno fuzzy inference, lies in the fact that the conclusion of each rule does not belong to the symbolic domain, but is defined in numerical form as a linear combination of the inputs.

The defuzzification process for a Sugeno system is more computationally efficient compared to that of a Mamdani system, since it uses a weighted average or weighted sum of a few data points rather than compute a centroid of a two-dimensional area [13].

Each rule in a Sugeno system operates as shown in figure (II.3), which shows a two-input system with input values x and y. In this case, the conclusion of each rule of the Sugeno system including a collection of fuzzy IF–THEN rules is in the following form:

$$R^{(l)}\text{:If x is } F_1^l \text{ and y is } F_2^l \text{ Then } z^l = a^l x + b^l y + c^l \quad (II.12)$$

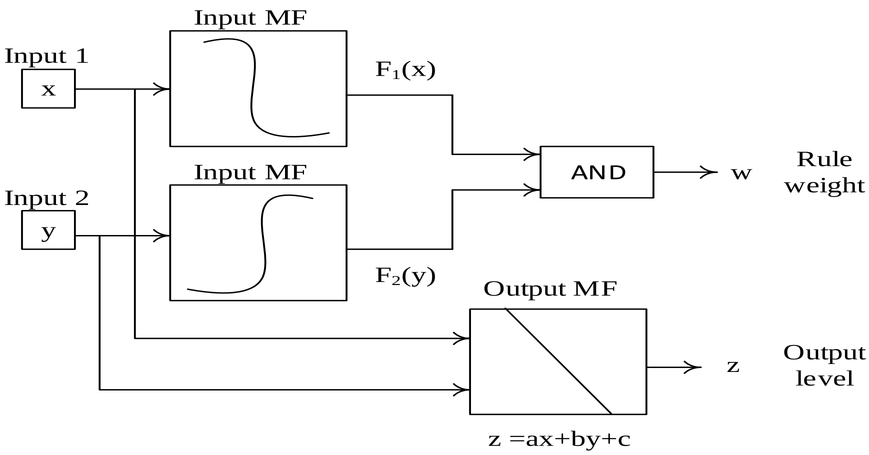


Figure (II.3): Diagram shows rule operations in Sugeno system

Each rule generates two values:

- z^l **:** Rule output level, which is either a constant value or a linear function of the input values:

Here, x and y are the values of input 1 and input 2, respectively, and a^l, b^l, and c^l are constant coefficients. For a zero-order Sugeno system, z^l is a constant ($a^l = b^l = 0$).

- w^l **:** Rule firing strength derived from the rule antecedent

Here, F_1 and $F_2$are the membership functions for inputs 1 and 2, respectively.

The output of each rule is the weighted output level, which is the product of w^l and z^l.

The easiest way to visualize first-order Sugeno systems (a^l and b^l are non-zero) is to think of each rule as defining the location of a moving singleton. That is, the singleton output spikes can move around in a linear fashion within the output space, depending on the input values. The rule firing strength then defines the size of the singleton spike [13].

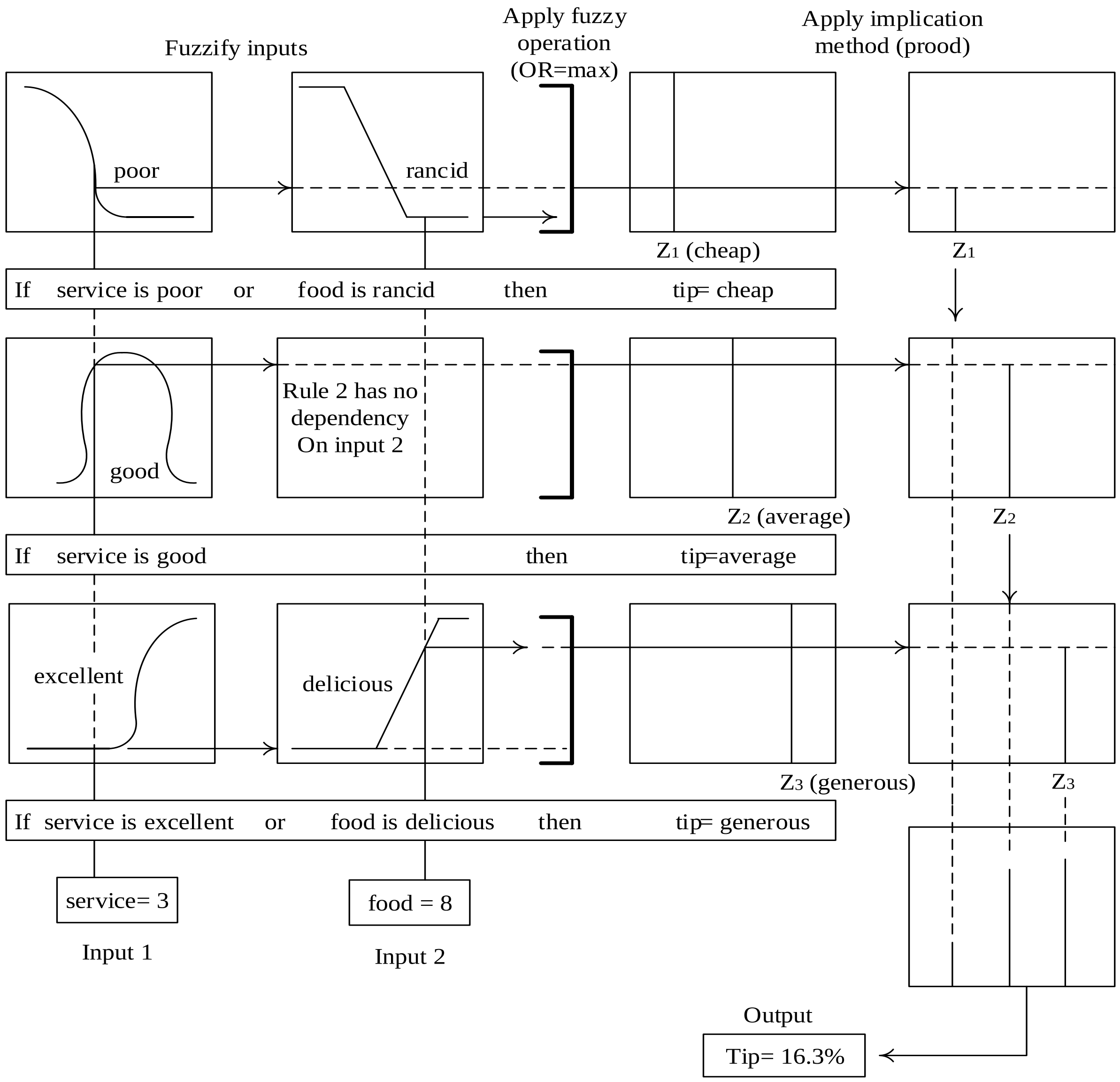


Figure (II.4): Fuzzy inference process for a Sugeno system

Sugeno systems always use product implication and sum aggregation, as shown in figure (II.4).

II.3.2.3 Comparison of Mamdani and Sugeno Inference Systems

Table (II.1) sums up the advantages and drawbacks of PI and FLC control approaches.

Table (II.1): Comparison between different Fuzzy Inference Systems [13].

Fuzzy Inference System	Advantages	Disadvantages
Mamdani	- Intuitive - Well-suited to human input - More interpretable rule base - Have widespread acceptance - Suitable for MISO and MIMO systems	- Less degree of freedom in the design.
Sugeno	- Computationally efficient - Work well with linear techniques, such as PID control - Work well with optimization and adaptive techniques - Guarantee output surface continuity - Well-suited to mathematical analysis	- Suitable only for MISO systems.

II.3.3 Defuzzification

To make that the output fuzzy set available to real applications, a defuzzification process is needed. So, the defuzzification is defined as the process of converting a fuzzy set into a crisp set or to convert a fuzzy member into a crisp member. The commonly used defuzzification techniques are Mean of Maximum method, Center of Gravity method and the Height method.

II.3.3.1 Mean of Maximum (MOM) Method

The Mean of Maximum (MOM) defuzzification method computes the average of those fuzzy outputs that have the highest degrees. In the case of a discrete universe, this defuzzification output z^* can be expressed as:

$$z^* = \sum_{i=1}^{k} \frac{z_i}{k} \qquad \text{(II.13)}$$

Where z_i is the control action whose membership functions reach the maximum, and k is the number of such control actions.

II.3.3.2 Center of Gravity (COG) Method

In the Center of Gravity method (COG), the center of the gravity of the area bounded by the membership function curve is computed to be the crisp value of the fuzzy quantity. The COG output z^* can be represented as:

$$z^* = \frac{\sum_{i=1}^{n} z_i \mu(z_i)}{\sum_{i=1}^{n} z_i} \qquad \text{(II.14)}$$

Where z_i represents a sample element of the scaled output fuzzy sets, and n represents the number of the samples.

II.3.3.3 Height Method (HM)

This method can be divided into two steps. First, the consequent membership function μ_Z can be converted into a crisp consequent c_i, where c_i is the center of gravity of μ_Z. Then the COG method is applied to the rules with crisp consequents, which can be expressed as

$$z^* = \frac{\sum_{i=1}^{n} z_i c_i}{\sum_{i=1}^{n} z_i} \qquad \text{(II.15)}$$

II.4 Types of FLC controller

In the last decade, there was an increasing interest in combining fuzzy concepts with classic control techniques. The main objectives for such a hybrid implementation were that fuzzy logic issues such as uncertainty (or unknown variations in plant parameters and structure) can be dealt with more effectively, hence improving the robustness of the control system [14].

II.4.1 PI Fuzzy Logic Controller (PI-FLC)

The purpose of this section is to present in more details the structure chosen for fuzzy controllers of PI type used in this work. This controller is represented in figure (II.5).

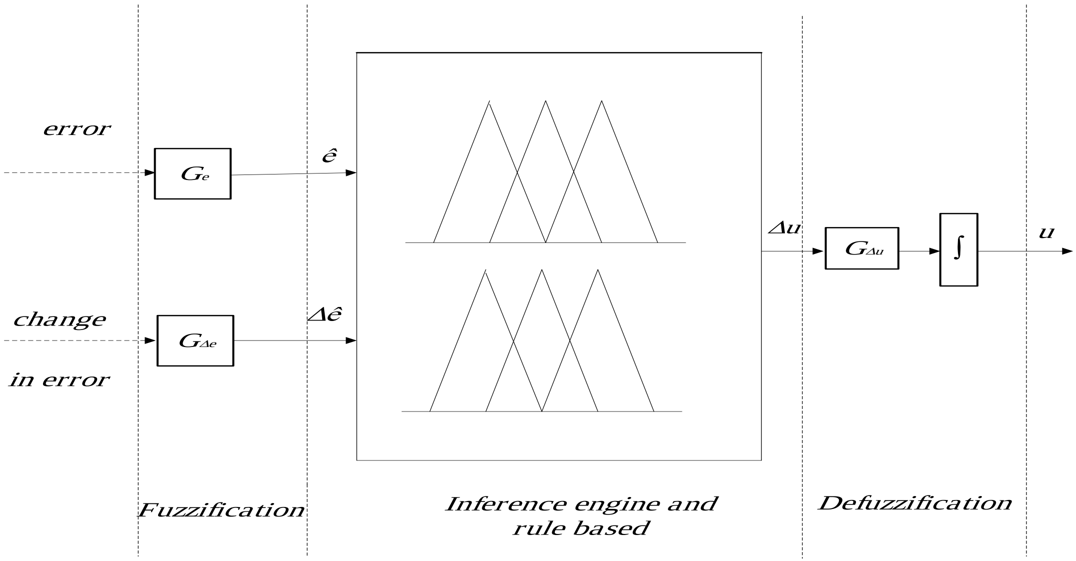


Figure (II.5): PI Fuzzy Controller block diagram [14]

Two inputs are processed, the error e and the change in error Δe for a single output u . The two inputs are normalized by means of normalization factors, G_e for the error and $G_{\Delta e}$ for the change in error. A normalization factor $G_{\Delta u}$ is used to normalize the output. The interval [−1, +1] is the universe of discourse of membership functions. Thus, the normalization factors make it possible to define the normalized range of variation of the inputs and the factor in denormalization defines the factor at the output of the fuzzy-PI controller. These elements make it possible to act globally on the control surface by widening or reducing the universe of the discourse of control quantities.[14]

II.4.1.1 Fuzzy Control Law Definition

This law is based on the error and its variation ($u = f\ (e, \Delta e)$). Consequently, the activation of all the associated decision rules gives the variation of the necessary output u.

The general formula of the control law is giving below [24][25],

$$u_{k+1} = u_k + G_{\Delta u}\Delta u_{k+1} \tag{II.16}$$

Where,

$G_{\Delta u}$: Gain associated to the controller output u_{k+1}

Δu_{k+1} :Change in the controller output (command).

The normalization of error e and change in error Δe are given as,

$$\begin{aligned} \hat{e} &= G_e e \\ \Delta\hat{e} &= G_{\Delta e}\Delta e \end{aligned} \tag{II.17}$$

Where, G_e and $G_{\Delta e}$ represents scaling factors (normalization parameters). To find a suitable response, these factors should be appropriately chosen.

II.4.2 Fuzzy-PID controller

The PID control algorithm is expressed as a discrete control by:

$$u(k) = K_P \left\{ e(k) + \frac{T}{T_i} \sum_{j=0}^{K} e(j) + \frac{T_d}{T} \left[e(k) - e(k-1) \right] \right\} \tag{II.18}$$

It has been very difficult to properly adjust the parameters of the PID controllers, as many industrial systems models do not take into account phenomena such as delays, and non linearities. For this reason, many researchers have put forward many new PID control strategies based on intelligent algorithms. One of the most interesting methods is based on the concepts of fuzzy logic. Figure (II.6) shows the control diagram of the system using a PID controller based on a fuzzy system, where the parameter values of the PID controller are adjusted by a fuzzy system.[14]

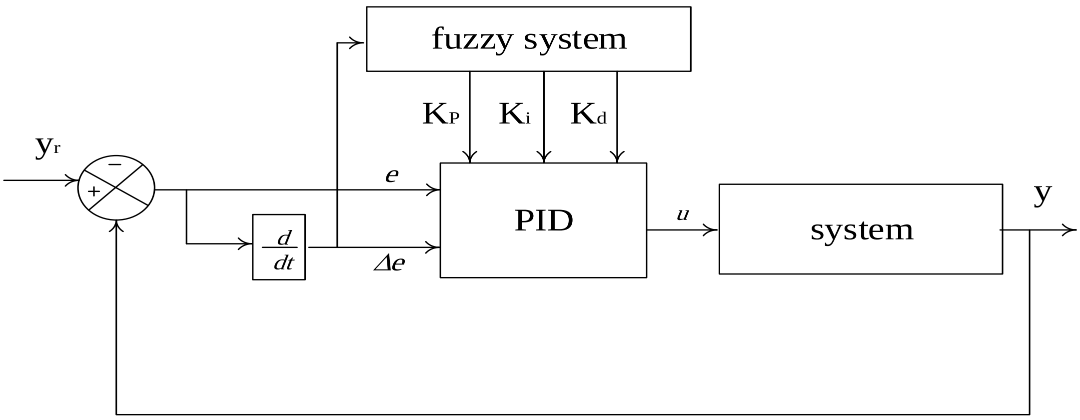


Figure (II.6): PD-Fuzzy controller [14]

Three separate fuzzy control tables are used to define the three parameters of the PID controller; these can be seen in tables from (II.2) to (II.4).

Table (II.2): Rule based of K_p for fuzzy-PID controller

		Δ*e*						
		NB	NM	NS	Z	PS	PM	PB
e	NB	PB	PB	PM	PM	PS	Z	Z
	NM	PB	PB	PM	PS	PS	Z	NS
	NS	PM	PM	PM	PS	Z	NS	NS
	Z	PM	PM	PS	Z	NS	NM	NM
	PS	PS	PS	Z	NS	NS	NM	NM
	PM	PS	Z	NS	NM	NM	NM	NB
	PB	Z	Z	NM	NM	NM	NB	NB

Table (II.3): Rule based of K_i for fuzzy-PID controller.

		Δ*e*						
		NB	NM	NS	Z	PS	PM	PB
e	NB	NB	NB	NM	NM	NS	Z	Z
	NM	NB	NB	NM	NS	NS	Z	Z
	NS	NB	NM	NS	NS	Z	PS	PS
	Z	NM	NM	NS	Z	PS	PM	PM
	PS	NM	NS	Z	PS	PS	PM	PB
	PM	Z	Z	PS	PS	PM	PB	PB
	PB	Z	Z	PS	PM	PM	PB	PB

Table (II.4): Rule based of $K_d$for fuzzy-PID controller

		Δ*e*						
		NB	NM	NS	Z	PS	PM	PB
e	NB	NS	NB	NB	NB	NS	PS	PS
	NM	NS	NB	NM	NM	NS	Z	PS
	NS	NS	NS	NM	NS	NS	Z	Z
	Z	NS	NS	NS	NS	NS	Z	Z
	PS	Z	Z	Z	Z	Z	Z	Z
	PM	NS	PS	PS	PS	PS	PB	PB
	PB	PM	PM	PM	PS	PS	PB	PB

II.8.1.3 Fuzzy-PID controller with integral action

A fuzzy-PD controller with a conventional *I* controller can be obtained by placing a fuzzy-PD controller in parallel with the conventional integral control device. The structure of the fuzzy-PD with classic controller *I* is shown in figure (II.7).[14]

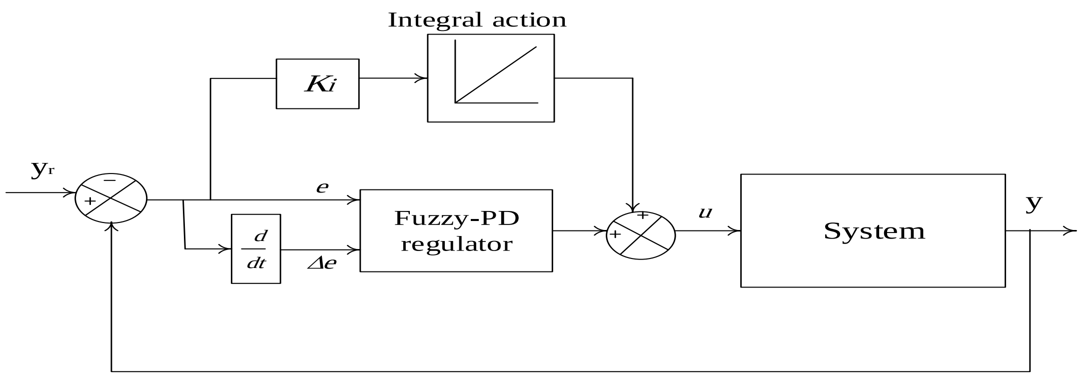


Figure (II.7): Fuzzy-PD controller with integral action

II.5 Fuzzy Controller Designing for Quadruple Tank Process

The structure of FLC of quadruple tank process is presented in figure (II.8).

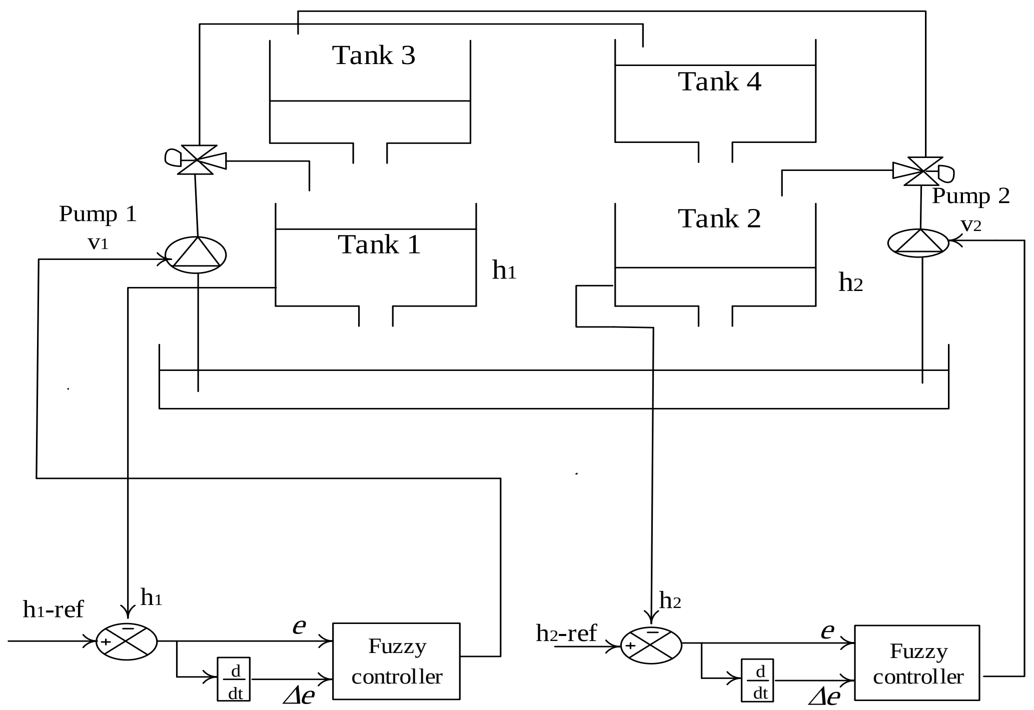


Figure (II.8): Fuzzy Control structure for Quadruple Tank Process [14]

Basic steps for the FLC designing, as shown in Figure (II.8), comprises of three principal components: fuzzification interface, decision making and a defuzzification interface[12].

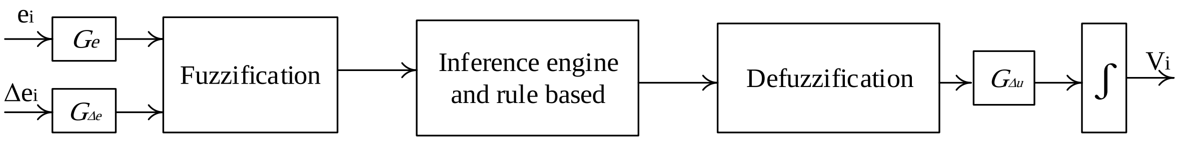


Figure (II.9): Internal configuration of a fuzzy logic controller

In the present work the error (e) and change in error (Δe) of level outputs (h_1 and h_2) are taken as inputs and the pumps voltages (v_1,v_2) are the controller outputs.

II.5.1 Fuzzification

Variables e_i and Δe_i are selected as the input variables, where e_i is the error between the reference water level ($h_{i\text{-}ref}$) and actual water level (h_i) of the tank i, Δe_i is the change in error in the sampling interval. The output variable is the reference voltage (v_i) where (i=1, 2). These variables are transformed to linguistic variables using seven fuzzy membership function named negative big (NB), negative medium (NM), negative small (NS), zero (Z), positive small (PS), positive medium (PM), positive big (PB).

Gaussian membership functions are adopted for input variables e_i, Δe_i, and output variables Δv_i. The selected Gaussian membership functions are given by:

$$\begin{aligned}
Gaussien(e_i,c_i,\sigma_i) &= e^{-\frac{1}{2}\left(\frac{e_i-c_i}{\sigma_i}\right)^2} \\
Gaussien(\Delta e_i,c_i,\sigma_i) &= e^{-\frac{1}{2}\left(\frac{\Delta e_i-c_i}{\sigma_i}\right)^2} \\
Gaussien(\Delta v_i,c_i,\sigma_i) &= e^{-\frac{1}{2}\left(\frac{\Delta v_i-c_i}{\sigma_i}\right)^2}
\end{aligned} \tag{II.19}$$

A Gaussian membership function is completely determined by c_i and σ_i. The parameter c_i represents the membership functions center and σ_i determines the MFs width. The adopted membership functions for error, change in error, and controller output are displayed in Figure (II.10) and their centers of gravity are given in table (II.5).

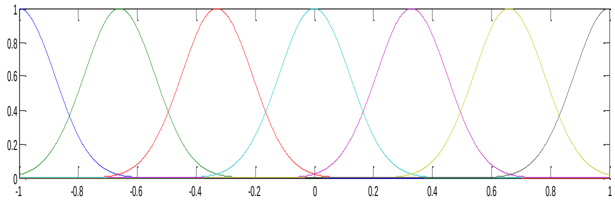


Figure (II.10): Gaussian membership function

Table (II.5): Centers of gravity of Gaussian membership functions for QTP [14]

Membership function	Center of gravity
NB	-1
NM	-0.66
NS	-0.33
ZE	0
PB	1
PS	0.66
PB	0.33

All the membership functions are set within a normalized range [-1,1]. Choosing of scale factors is the most important problem when working on designing FLC. This scale factors play an extremely important role because they are the ones that determine the dynamic performance of the control.

II.5.2 Fuzzy Inference Engine

The decision stage treats the input data and computes the controller output using 49 rules that are enumerated in Table (II.6).

Table (II.6): Rule Table of Fuzzy Logic controller for one loop of QTP

Δe \ e	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NB	NM	NS	NS	ZE	PM
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NM	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PS	PM	PB	PB
PB	ZE	PS	PS	PM	PB	PB	PB

II.5.3 Deffuzification

In this work, the centroid method is adopted; it is defined by the following algebraic expression:

$$dv_i = \frac{\sum \mu_j c_j}{\sum \mu_j}, \ \text{i=1,2} \qquad \text{(II.20)}$$

where c_j is the center of gravity.

Centers of gravity of Gaussian membership functions are given in Table (II.6).

II.6 Simulation Results and Discussions

In this section, simulations are conducted to validate the performance of the designed fuzzy controllers applied on the quadruple-tank process. Separate simulations are performed for both minimum-phase and non-minimum phase systems. The fuzzy control performances are compared between those obtained with classical PI control. The FLCs parameters are indicated in the Table (II.7).

Table (II.7): FLCs parameters [14]

	Tank 1 water level FLC	Tank 2 water level FLC
Minimum-phase	e=0.1 de=0.1 du=7	e=0.1 de=0.1 du=7
Non-minimum phase	e=0.001 de=0.01 du=10	e=0.001 de=0.01 du=10

II.6.1 Simulation Results for Minimum Phase

The closed loop responses of water levels in tanks 1 and 2 for minimum phase with fuzzy controllers of QTP system are shown in Figure (II.11). In this figure the levels references value are step changed from 13 cm to 14.5 cm.

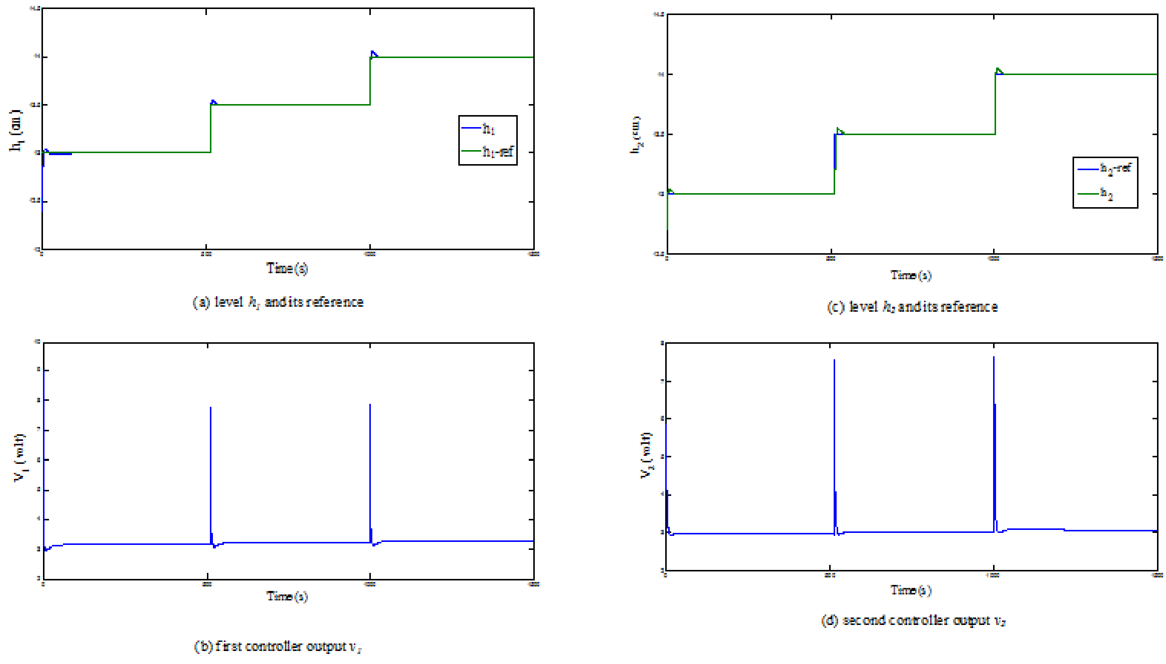


Figure (II.11): Minimum phase performances of QTP controlled by FLC

II.6.2 Simulation Results for Non-Minimum Phase

The closed loop responses of water levels in tanks 1 and 2 for non-minimum phase with fuzzy controllers of QTP system are shown in Figure (II.12). In this figure the levels references value are step changed from 13 cm to 14.5 cm.

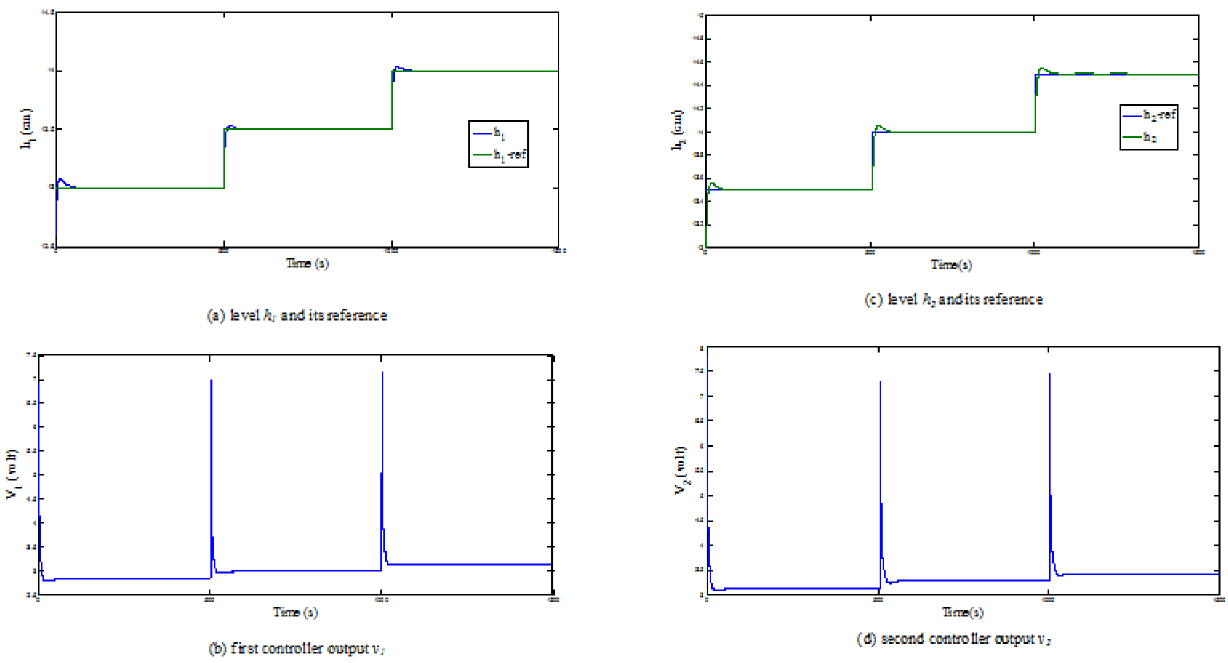


Figure (II.12): Non-minimum phase performances of QTP controlled by FLC

II.6.3 Discussion of Results

This part discusses the simulation studies carried out on the quadruple-tank process.

II.6.3.1 Minimum Phase Case

To test the robustness of the system, the tank 1 and tank 2 water levels setting points are changed between the following values 13 cm, 13.5 cm, and 14 cm.

Figure (II.11) shows the simulated outputs of the quadruple tank process controlled by FLC and operating in minimum phase. The tanks levels settle down with no oscillations in very small amount of time but with small overshoots. At each water level reference step variation, a tiny peak overshoot is produced due to its corresponding input voltage rapid variation. Apart from these starting regimes, both water levels follow perfectly their references.

II.6.3.2 Non-Minimum Phase Case

In this case, the tank water levels setting points changed from 13 cm at the beginning, then 13.5 cm to 14 cm at the end.

As illustrated in Figure (II.12), the simulation results representing the outputs of the quadruple tank process controlled by FLC and operating in non-minimum phase show that both tanks water levels take more time to regains their tracking points than the minimum phase case.

These results show that the FLC yields results superior to those obtained by conventional control algorithms in terms of overshooting amplitude and response time. In contrast to PI control, where the mathematical modeling phase is more than indispensible, the FLC controlling nonlinear MIMO QTP system seemed to be the best option to handle this limitation.

On the other hand, the FLC is more flexible than PI controller because it has more parameters (types and membership functions parameters in the fuzzification and defazzifcation modules) to select in order to shape its control surface. This makes it useful in case of nonlinear systems control. Admittedly, it is simple to implement FLC, but it is relatively difficult to tune its huge number of parameters. Another point, FLC is easy to understand because of its rule-bases structure, but only in case of small number of linguistic variables.

II.7 Conclusion

The main objective of this chapter was to design suitable fuzzy controllers capable to improve the quadruple tank process performances involving both minimum and non-minimum phases. The obtained simulation results clearly show the potential advantages of using fuzzy controller for a quadruple tank process compared to the conventional PI controllers case.

The main advantage of fuzzy systems is their ability to control a given system without the need of its mathematical model. However there is no formal method to choose the parameters of a fuzzy controller. The next chapter, will present an optimization technique to adjust this parameters.

Chapter III

PSO Based Quadruple Tank Process Control

III.1 Introduction

The last two decades have seen interesting developments in the field of liquid level control in many industrial applications. Therefore, many researchers have employed experimental laboratory systems, which have the same control problems of the liquid level in the industrial process like nonlinearity, constrained variables, and large time delay in order to evaluate the developed control strategies for controlling liquid levels. The real systems have actually some constraints and the control design is not an easy task. To solve this type of problem, various works have recently used the intelligent optimization algorithms such as Particle Swarm Optimization (PSO) algorithm, Genetic Algorithm (GA), and ant colony optimization [22].

PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles [23]. It is well proven that PSO can get better results in a faster way compared with other methods. Another reason that PSO is attractive is that there are few parameters to adjust. One version, with slight variations, works well in a wide variety of applications.

In this chapter, first, an overview about PSO will be provided. After that PSO based PI and FLC controllers will be presented and applied on the quadruple tank process. At the end, comprehensive simulation results will be illustrated and discussed.

III.2 Overview about PSO

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling. In past several years, PSO has been successfully applied in many research and application areas such as reactive power and voltage control by a Japanese electric utility [18]. Particle swarm optimization has also been used in conjunction with a back-propagation algorithm to train a neural network as a state-of-charge estimator for a battery pack for electric vehicle use [17].

III.2.1Principe of Particle Swarm Optimization Algorithm

Suppose the following scenario: a group of birds are randomly searching food in an area, in which only one piece of food exist. All birds do not know where the food is, but they know how far it is. So, the best and effective strategy to find the food is to follow the bird, which is nearest to it.

PSO learned from the previous scenario and used it to solve optimization problems. In PSO, each single solution, called particle, is a "bird" in the search space. All of particles have fitness values, which are evaluated by the fitness function to be optimized, and have velocities, which direct the flying of the particles. The particles fly through the problem space by following the current optimum particles [21].

Each particle keeps track of its coordinates in the problem space, which are associated with the best solution (P_{best}) it has achieved so far. Another “best” value called (P_{Gbest}) that is obtained so far by any particle in the population [16].

III.2.2Mathematical Concept

Mathematically, the particles are manipulated according to the following equations:

$$v_i(k+1) = w v_i(k) + c_1 r_1 [p_{ibest}(k) - x_i(k)] + c_2 r_2 [p_{Gbest}(k) - x_i(k)] \quad \text{(III.1)}$$

$$x_i(k+1) = x_i(k) + v_i(k+1) \quad \text{(III.2)}$$

where $x_i(k)$ is the position of the ith particle, $v_i(k+1)$ is the velocity of the ith particle, w is the inertia weight, which has been added in order to improve the convergence of the algorithm and to limit the velocity vector $V_i$to the maximum values. The inertia factor w varying between $[w_{\min}, w_{\max}]$, $p_{ibest}(k)$ represents the local best position of the ith particle, $p_{Gbest}(k)$ is the global best position in the swarm, c_1 and c_2 are positive constants, which define the learning factors of the velocity.

Figure (III.1) shows the search mechanism of PSO in multidimensional search space [16].

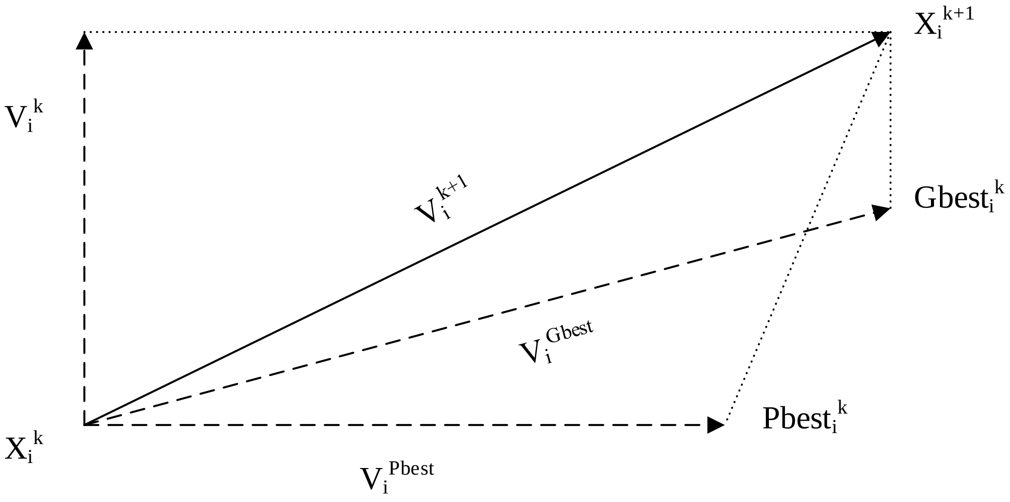


Figure (III.1): PSO search mechanism in multi-dimentional search space [19]

The PSO concept consists of, at each time step, changing the velocity (accelerating) each particle toward its P_{best} and G_{best} locations (global version of PSO). Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward P_{best} and G_{best} locations.

The process for implementing the global version of PSO is as follows [19]:

1) Initialize a population of particles with random positions and velocities in the problem space;
2) For each particle, evaluate the desired optimization fitness function;
3) Compare particle's fitness evaluation with particle's P_{best}. If the current value is better than P_{best}, then set P_{best} value equal to the current value, and the P_{best} location equal to the current location;
4) Compare fitness evaluation with the population's overall previous best. If the current value is better than G_{best}, then reset G_{best} to the current particle's array index and value;
5) Change the velocity and position of the particles according to equations (III.1) and (III.2), respectively;
6) Loop to step (2) until a criterion is met; usually a sufficiently good fitness or a maximum number of iterations (generations) are the most adopted criteria.

Not that if the velocity of each particle exceeds V_{max}, which is a parameter specified by the user, then the velocity should be limited to V_{max}. Actually, V_{max} determines the resolution with which regions between the present position and the target position are searched. If V_{max} is too high, particles might fly past good solutions. The maximum velocity V_{max} serves as a constraint to control the global exploration ability of a particle swarm. Indeed, a larger V_{max} facilitates global exploration, while a smaller V_{max} encourages local exploitation. [19].

III.2.3 Inertia Weight

The concept of an inertia weight was developed to better control the exploration and exploitation. Suitable selection of the inertia weight provides a balance between global and local exploration and exploitation, and results in less iteration on average to find a sufficiently optimal solution. As originally developed, *w* often is decreased linearly from about 0.9 to 0.4 during a run [21].

After some experience with the inertia weight, it was found that although the maximum velocity factor V_{max} couldn't always be eliminated, the particle swarm algorithm works well if V_{max} is set to the value of the dynamic range of each variable. Thus, the need to think about how to set V_{max} each time the particle swarm algorithm is used is eliminated [21].

III.2.4 Construction Factor

The work did by Clerc in [20] indicate that use of a construction factor may be necessary to insure convergence of the particle swarm algorithm. A simplified method of incorporating it appears in equation (III.3), where *k* is a function of c_1 and $c_2$as reflected in equation (III.4).

$$v_{id} = k[v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id})] \qquad \text{(III.3)}$$

$$k = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|}, \text{where } \varphi = c_1 + c_2,\ \varphi > 4 \qquad \text{(III.4)}$$

III.2.5 Particle Swarm Optimization Coding Steps

The process for implementing the PSO is as follows:

1. Set PSO parameters: $w_{\min}, w_{\max}$, c_1 and c_2
2. Initialize population of particles having velocities *V* and positions *X*
3. Set iteration *k* = 1
4. Calculate fitness of particles $F_i^k = f(X_i^k), \forall i$ and find the index of the best particle *b*
5. Select $P_{ibest}(k) = X_i(k), \forall i$ and $P_{Gbest}(k) = X_b(k)$
6. $w = w_{\max} - \frac{k(w_{\max} - w_{\min})}{Maxite}$
7. Update velocity and position of particles using equation (III.1) and (III.2)
8. Evaluate fitness $F_i^{k+1} = f(X_i^{k+1}), \forall i$ and find the index of the best particle *b1*
9. Update P_{best} of population $\forall i$

   If $F_i^{k+1} < F_i^k$ then $P_{ibest}(k+1) = X_i(k+1)$ else $P_{ibest}(k+1) = P_{ibest}(k)$

10. Update P_{Gbest} of population

If $F_{b1}^{k+1} < F_b^k$ then $P_{Gbest}(k+1) = P_{b1best}(k)$ and set $b=b1$ else $P_{Gbest}(k+1) = P_{Gbest}(k)$

11. If $k<$ Maxite then $k=k+1$ and goto step 6 else goto step 12
12. Print optimum solution as $P_{Gbest}(k)$

The most commonly used parameters of PSO algorithm are considered as follows:

- Inertia weight: 0.9 to 0.4
- Acceleration factors (c_1 and c_2): 2 to 2.05
- Population size: 10 to 100
- Maximum iteration (Maxite): 500 to 10000
- Initial velocity: 10 % of position

A detailed flowchart of PSO considering the above steps is shown in Figure (III.2).

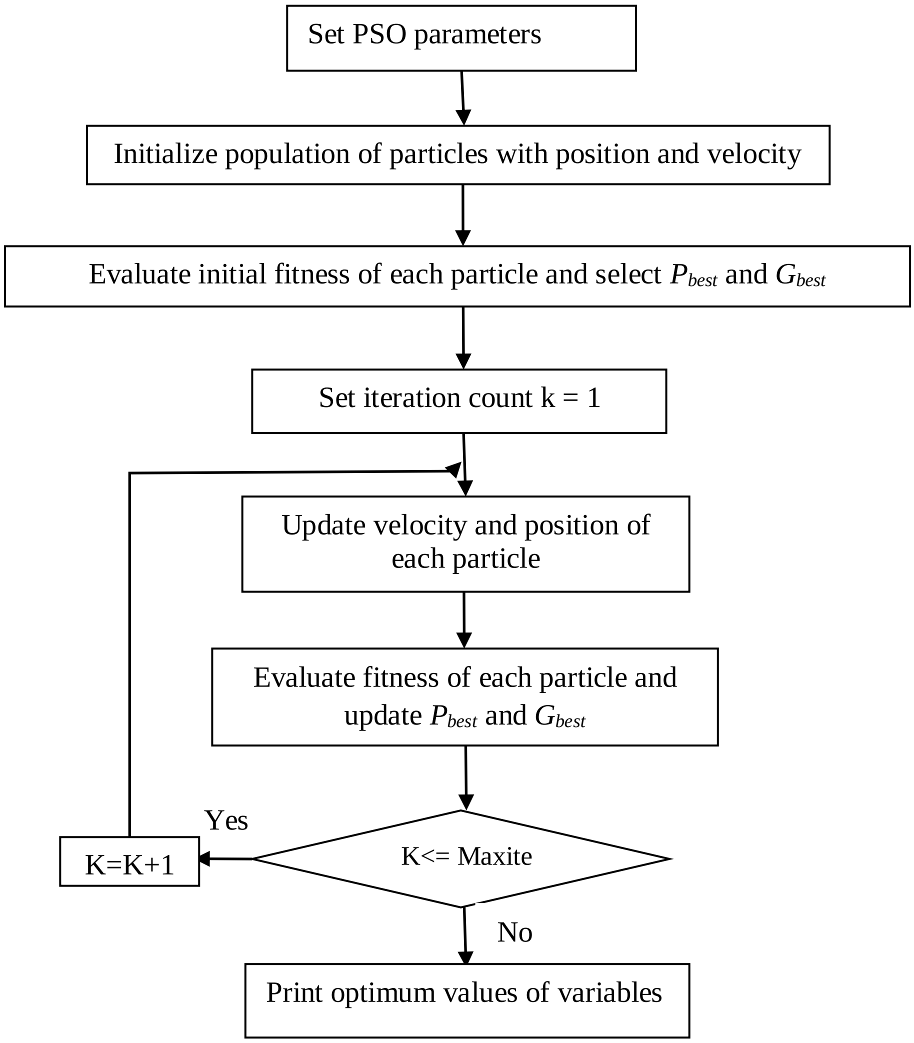


Figure (III.2): Flowchart of PSO [19]

III.3 PSO for Quadruple Tank Process

In this section, we explained how to optimize both PI controller and fuzzy controller parameters using PSO algorithm, and discuss the obtained results.

III.3.1 PI Controller Optimized by PSO

The proposed PSO algorithm is applied to optimize the control law of the quadruple tank process (Figure (III.3)). Indeed, the PSO algorithm is used to adjust the parameters of the two PI controllers. Table (III.1) shows the parameters adopted for the PSO algorithm. The chosen objective function is given by equation (III.5).

$$f = \int_0^{\infty} e(t)^2 \, dt \tag{III.5}$$

where *e* represents the error between the measured water level and its reference.

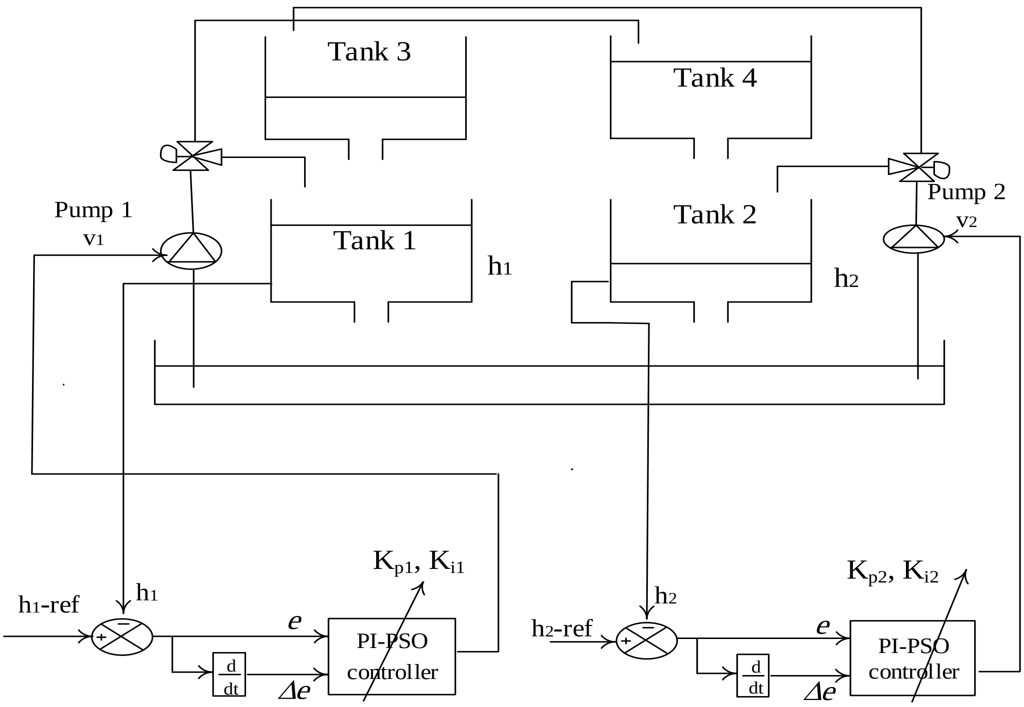


Figure (III.3): PI-PSO based quadruple tank process control [14]

Table (III.1) : PSO parameters [14]

Parameter	Value
Number of particles (N)	40
c_1	1
c_2	1
w_{min}	0.25
w_{max}	1.05

The optimal PIs gains are grouped together in Table (III.2).

Table (III.2): Parameters optimized by PSO

PI_1 controller parameters optimized by PSO for the minimum phase	k_{p1}=2.21, k_{i1}=0.045
PI_2 controller parameters optimized by PSO for the minimum phase	k_{p2}=1.63, k_{i2}=0.08
PI_1 controller parameters optimized by PSO for the non-minimum phase	k_{p1}=3.65, k_{i1}=0.347
PI_2 controller parameters optimized by PSO for the non-minimum phase	k_{p2}= -0.08, k_{i2}= -0.007

III.3.1.1 Simulation Results

In this section, simulations are conducted to validate the performance of the optimized PI controller using PSO algorithm applied on the quadruple tank process. Separate simulations are performed for both minimum-phase and non-minimum phase systems.

III.3.1.1.1 Simulation Results for Minimum Phase

The closed loop responses of water levels in tanks 1 and 2 for minimum phase optimized PI controller of QTP system are shown in Figure (III.4). In this figure, the levels references value are step changed from 13 cm to13.5 cm then to 14 cm.

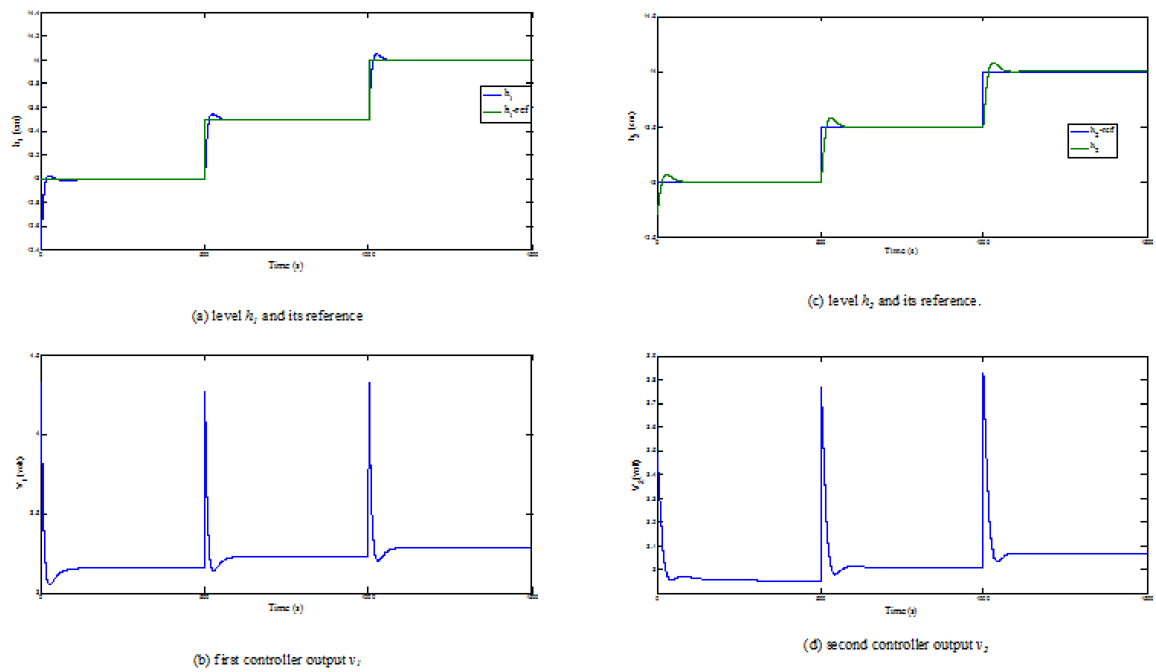
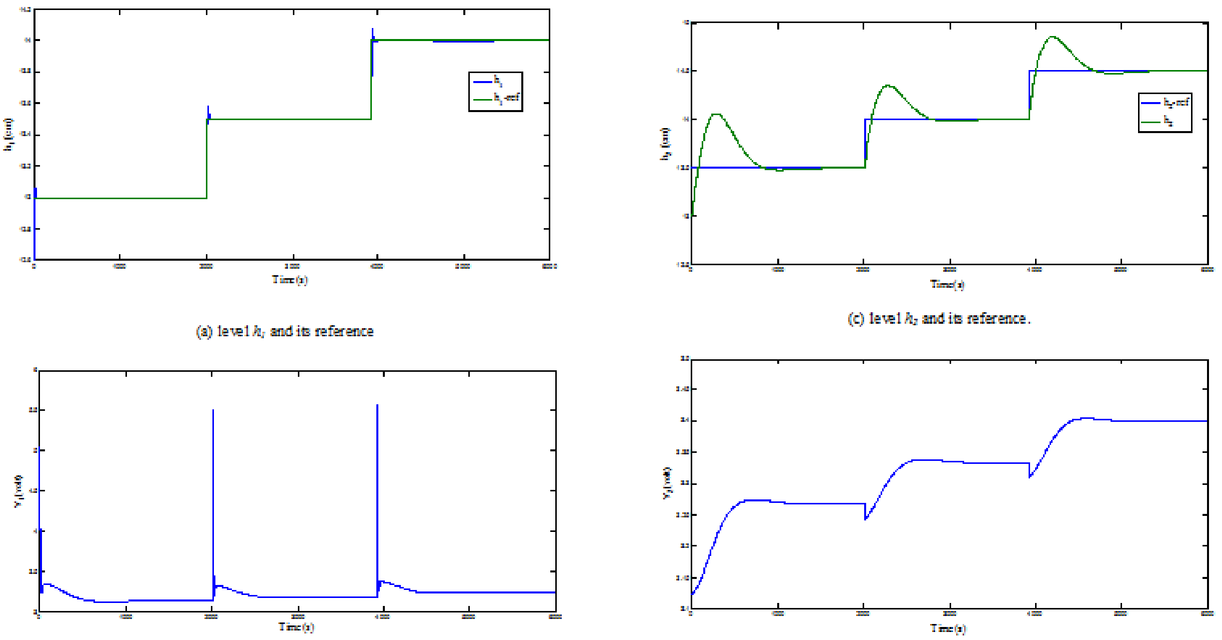


Figure (III.4): response of QTP operating in minimum phase and controlled by optimized PI controllers using PSO algorithm

III.3.1.1.2 Simulation Results for non-minimum phase

The closed loop responses of water levels in tanks 1 and 2 for non-minimum phase with optimized PI controllers of QTP system are shown in Figure (III.5). In this figure, the levels references values are step changed from 13 cm to13.5 cm then to 14 cm for h_1 and from 13.5 cm to 14 cm then to 14.5 cm for h_2.



(a) level h_1 and its reference (c) level h_2 and its reference.

(b) first controller output v_1 (d) second controller output v_2

Figure (III.5): response of QTP operating in non-minimum phase and controlled by optimized PI controllers using PSO algorithm

III.3.1.2Discussion of Results

This section discusses the simulation studies carried out on the quadruple-tank process.

III.3.1.2.1 Minimum Phase Case

Figure (III.4) shows the obtained results using the optimized PI controllers using PSO in the presence of disturbances. It easy to notice that the water levels responses are slow with overshoots for both considered outputs.

III.3.1.2.2 Non-Minimum Phase Case

The results of figure (III.5) show that the system responses are markedly improved, especially in terms of response time. Indeed, the response times have been reduced compared to the case of a non-optimized controller.

Also, from previous results, it is possible to realize that the use of particle swarm optimization algorithm has positive consequences on response time reduction and tracking accuracy even in the presence of perturbations, which leads without doubt to an improvement in the system dynamic regimes.

III.3.2 Fuzzy Controller Optimized by PSO

The particle swarm optimization algorithm is used to adjust the parameters of the membership functions of the fuzzy controller to improve its performance. The structure of FLC of quadruple tank process is presented in figure (III.6). The Gaussian membership functions used are given by equation (III.6).

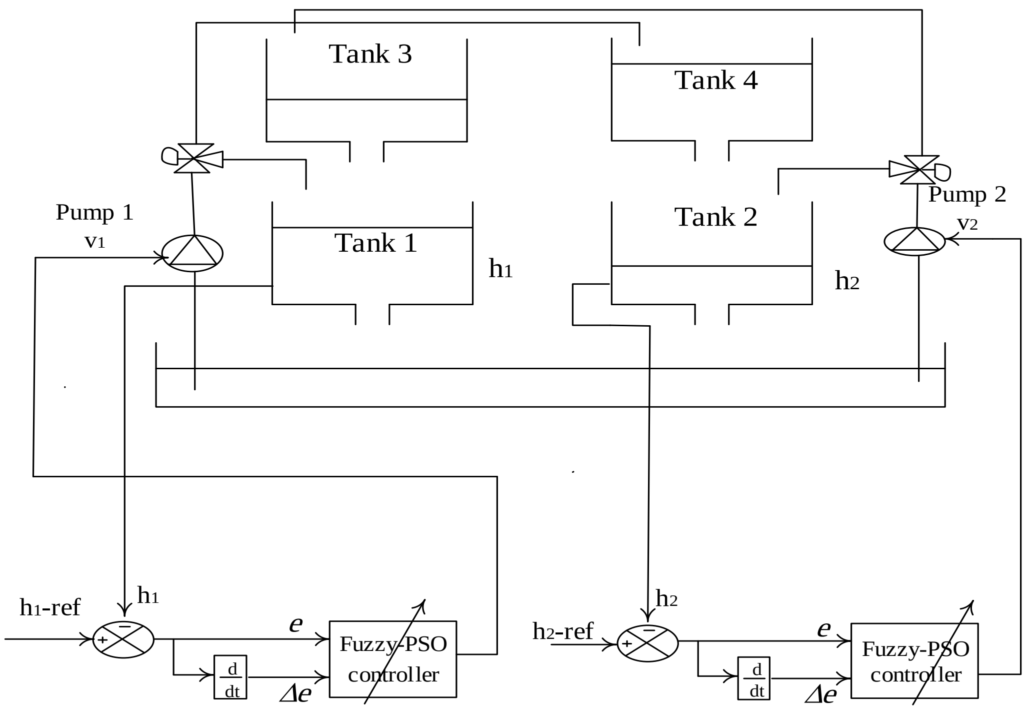


Figure (III.6): Fuzzy-PSO based quadruple tank process control [14]

$$Gaussien(e_i,c_i,\sigma_i) = e^{-\frac{1}{2}\left(\frac{e_i - c_i}{\sigma_i}\right)^2}$$
$$Gaussien(\Delta e_i,c_i,\sigma_i) = e^{-\frac{1}{2}\left(\frac{\Delta e_i - c_i}{\sigma_i}\right)^2} \qquad \text{(III.6)}$$
$$Gaussien(\Delta v_i,c_i,\sigma_i) = e^{-\frac{1}{2}\left(\frac{\Delta v_i - c_i}{\sigma_i}\right)^2}$$

The fuzzy controller optimization steps using PSO can be summarized as follows:

Step 1: Initialization of each individual.

Step 2: Calculate the parameters σ_i, c_i (the centers and the widths of the Gaussian membership functions for the inputs and outputs).

Step 3: Calculate the objective function (eq. III.5).

Step 4: Update the speed and position and iteration.

Step 5: Update p_{best} and g_{best}.

Step 6: Go to step 2 until the stop criterion is met.

Step 7: The fuzzy controller generated by the last g_{best} is the optimal fuzzy controller.

Table (III.3) shows the adopted parameters for the PSO algorithm.

Table (III.3): PSO Parameters [14]

Parameter	Value
Numberof particules (N)	20
c_1	1
c_2	1
w_{min}	0.3
w_{max}	1.1

III.3.2.1 Simulation Results

In this section, simulations are conducted to validate the performance of the optimized fuzzy controller using PSO algorithm applied on the quadruple tank process. Separate simulations are performed for both minimum-phase and non-minimum phase systems.

III.3.2.1.1 Simulation Results for Minimum Phase

Figure (III.7) shows the membership functions of the first controller representing the error, change in error, and controller output in the case of minimum phase.

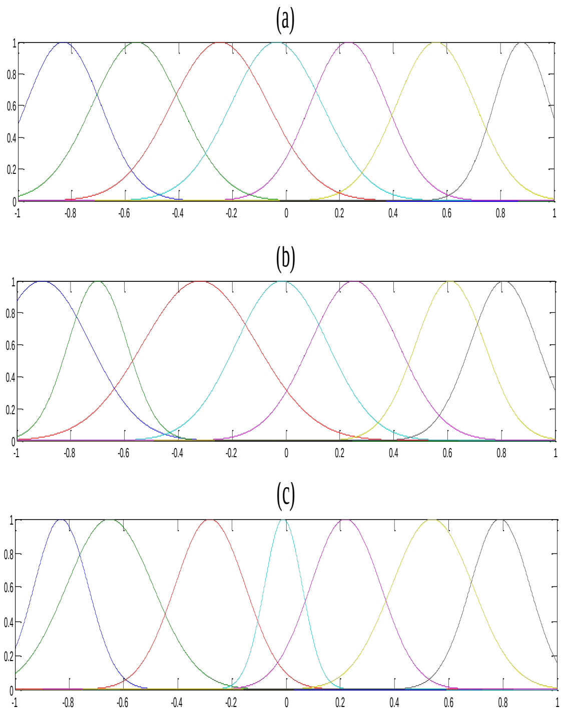


Figure (III.7): Membership functions of the first FLC optimized by PSO in case of minimum phase: a) e_1, b) Δe_1, c) ΔV_1

Figure (III.8) shows the membership functions of the second controller representing the error, change in error, and controller output in the case of minimum phase.

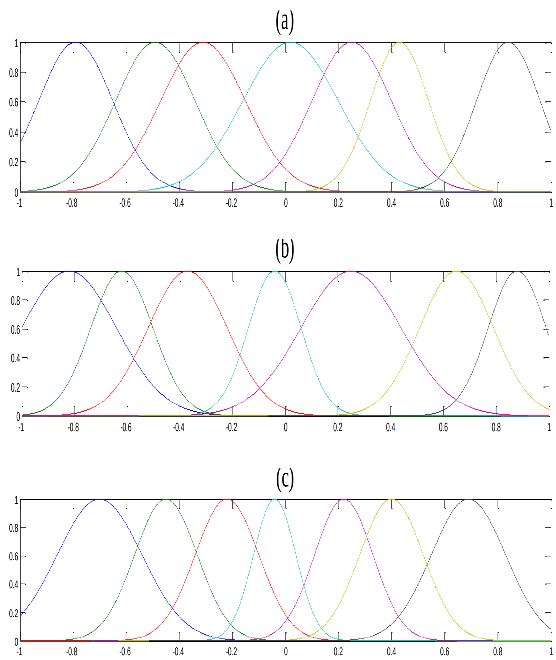


Figure (III.8): Membership functions of the second FLC optimized by PSO in case of minimum phase: a) e_2, b) Δe_2, c) ΔV_2

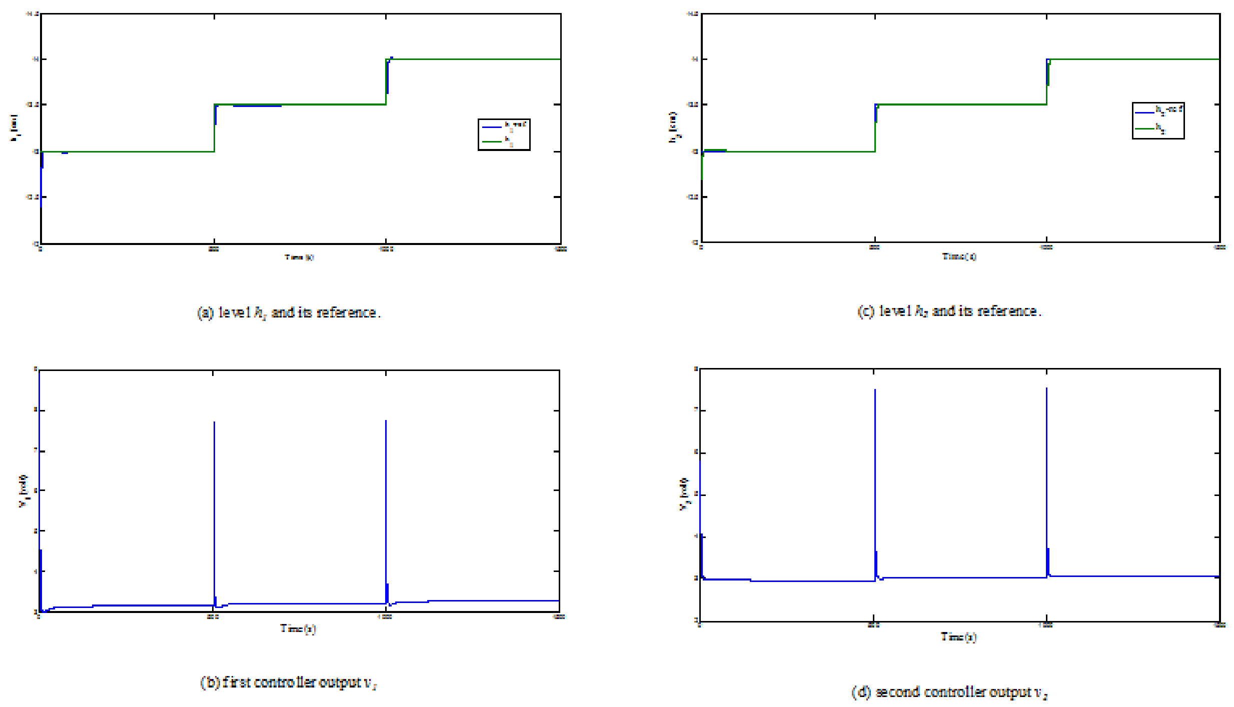


Figure (III.9): responses of QTP operating in minimum phase and controlled by optimized fuzzy controller using PSO algorithm

III.3.2.1.2 Simulation Results for non-minimum phase

Figure (III.10) shows the membership functions of the first controller representing the error, change in error, and controller output in the case of non-minimum phase.

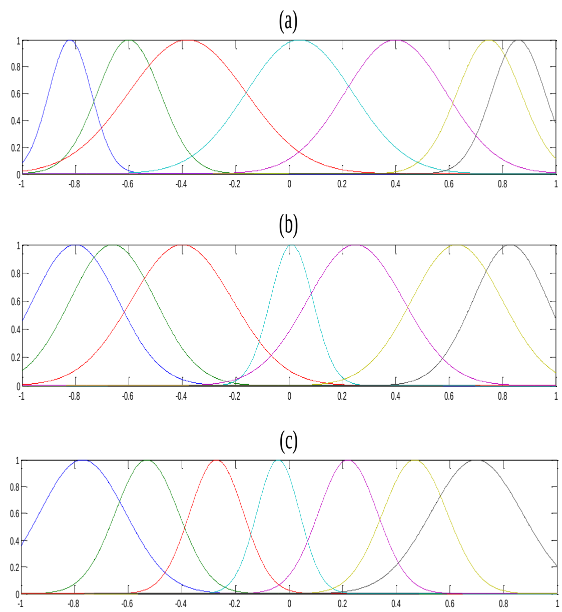


Figure (III.10): The membership functions of the first Fuzzy controller optimized by PSO of the non-minimum phase case: a) e_1, b) Δe_1, c) ΔV_1

Figure (III.11) shows the membership functions of the second controller representing the error, change in error, and controller output in the case of non-minimum phase.

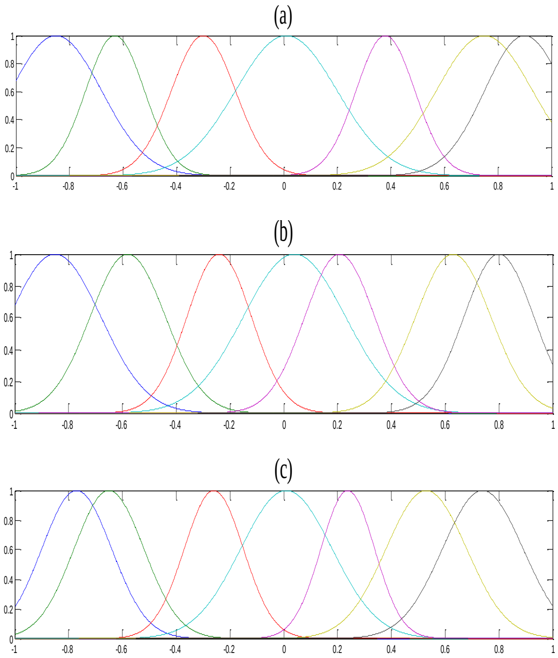


Figure (III.11): The membership functions of the first fuzzy controllers optimized by PSO of the non-minimum phase case: a) e_2, b) Δe_2, c) ΔV_2

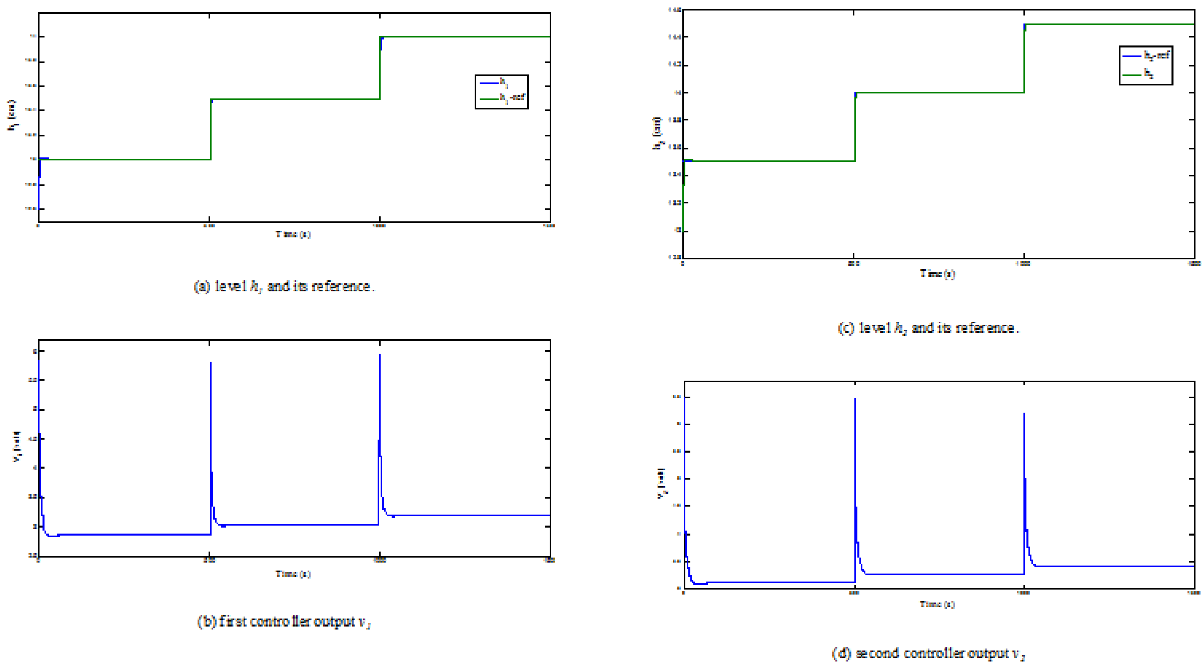


Figure (III.12): responses of QTP operating in non-minimum phase and controlled by optimized fuzzy controllers using PSO algorithm

III.3.2.2Discussion of Results

This section discusses the simulation studies carried out on the quadruple-tank process controlled by FLCs optimized by PSO.

III.3.2.2.1 Minimum Phase Case

The results illustrated in figure (III.11) show rapid convergences of the system outputs towards their references confirming a good tracking even in the presence of disturbances.

In addition, these results corroborate that the fuzzy controller optimized by PSO presents not only a good tracking but also it guarantees the system stability with overshoot reduction.

III.3.2.2.2 Non-Minimum Phase Case

The results of figure (III.12) show a rapid tracking of the system outputs towards their references. This can be seen as the ability of the FLC generated commands in forcing the system to follow its predefined set-points.

III.4 Performance criteria

Modern complex control systems are usually evaluated by performance criteria. Error and time are very important factors, which must be considered simultaneously. The notion of a performance index is very important in the design of the controller. Four performance criteria are generally considered.

III.4.1 ISE Criterion (Integral of the Squared Error)

The ISE criterion is expressed by:

$$ISE = \int_0^{\infty} e(t)^2 \, dt \qquad \text{(III.7)}$$

This criterion is relatively insensitive to weak errors compared to strong ones. Consequently, it often leads to a response with little overshoot but with a fairly long destabilization.

III.4.2 IAE Criterion (Integral of the Absolute Value Error)

The IAE criterion is expressed by:

$$IAE = \int_0^{\infty} |e(t)| \, dt \qquad \text{(III.8)}$$

This criterion gives more weight to small errors.

III.4.3 ITAE Criterion (Integral of the Time-Weighted Absolute Error)

The ITAE criterion is expressed by:

$$ITAE = \int_0^{\infty} t \, |e(t)| \, dt \qquad \text{(III.9)}$$

This criterion favors a low static error at the cost of an initial overshoot, which can be significant since it occurs for low values of *t*.

III.4.4 ITSE Criterion (Integral of the Time-Weighted Squared Error)

The ITSE criterion is expressed by:

$$ITSE = \int_0^{\infty} t\,|e(t)|^2\,dt \tag{III.10}$$

ITAE and ITSE have an additional function error time multiplier, which emphasizes errors of long durations, and therefore these criteria are most often applied in systems requiring a rapid stabilization time.

III.5 Comparative Study

The most commonly used performance indices as a design tool to assess regulators are ISE and IAE. The values of the performance indices obtained for the PI controller, PI controller optimized by PSO and fuzzy controller optimized by PSO are collated in Table (III.4).

According to the results gathered in Table (III.4), it is clear that the performance indices ISE and IAE have the highest values in case of PI based control. These values decrease significantly when PSO based PI control is applied. The same observations can be drawn for fuzzy control. Indeed, the performance indices ISE and IAE of FLC-PSO record the lowest value showing that the FLC-PSO is better than traditional FLC and PI controllers.

Tableau (III.4): Comparison of performance indices between controllers.

Controller	performance indices	Minimum phase		Non-minimum phase	
		h_1	h_2	h_1	h_2
PI	*ISE*	2.97	5.18	8.96	134.3
PI-PSO		2.11	3.08	7.33	81.3
FLC		1.77	1.83	6.31	7.91
FLC-PSO		1.13	1.26	3.83	3.97
PI	*IAE*	8.96	10.7	47.2	497.1
PI-PSO		4.37	5.17	6.51	195.2
FLC		2.67	3.56	5.12	6.11
FLC-PSO		1.37	1.62	3.26	3.81

III.6 Conclusion

The main idea of this chapter was to apply a PSO based optimization algorithm for the PI and fuzzy controllers in order to enhance the control performances of the quadruple tank process.

In the optimized version of the adopted PI controller, the PSO algorithm is used to adjust its parameters so that a predefined objective function is minimized, while for fuzzy controller, the particle swarm optimization algorithm is used to adjust the parameters of the membership functions to minimize the same objective function.

The obtained simulation results clearly show the positive effect of using PSO algorithm on the quadruple tank process performances. Indeed, the resulting optimized controllers can achieve good tracking performances and guarantee the stability of the system with reduced overshoots as well.

Hence, FLC-PSO has better results than the PI-PSO in term of response time and set-point tracking.

General Conclusion

The work done throughout this dissertation has focused on controlling a quadruple tank process through the use of an optimized PI and fuzzy controllers.

Our aim in the first chapter was to control the quadruple tank process in its both minimum and non-minimum phases. This requires an analytical description of the controlled system by an appropriate mathematical model. The linearization of its equations around an operating point has helped us to define a state space model as well as transfer functions for the QTP system, which makes possible the PI controllers design.

In the second chapter, the design of suitable fuzzy controllers for the quadruple tank process was accomplished. For this, the basics necessary to understand the methods based on fuzzy logic were presented first. Among the advantages of fuzzy systems is their ability to control a given system independently of knowledge of its mathematical model. However there is no formal method to choose the parameters of a fuzzy controller.

In order to improve the previous controllers performances by an appropriate choice of their parameters, the third part of this work was reserved to the optimization of PI and fuzzy controllers using PSO algorithm. This algorithm is used to optimize the gains of PI controllers and to adjust the parameters of the membership functions of fuzzy controllers.

According to the obtained simulation results, the QTP system has less robust performances when controlled by PI regulators compared to the case when fuzzy regulators are used. Additionally, as expected, the optimized versions of PI and fuzzy controllers show better performances when compared with traditional controllers.

Of course the methods presented in this work have given remarkable and concordant results, but other avenues remain to be explored. We can cite among others:

- Optimization of the proposed regulators using other optimization algorithms such as: fireworks algorithm and bee algorithm,
- Practical implementation of the proposed control techniques,
- Nonlinear control of the quadruple tank system.

References

[1]. P. J. Indira, N. Vignesh, R. Sunandha, P. ArunPrasath, and D. Saravanan, "Design and Implementation of a Self Tuning Regulator for Quadruple Tank System," Journal of Innovative Research and Solution, vol. 1, no. 1, 2014, pp. 264-9.

[2]. C. Ramadevi and V. Vijayan, "Design of Decoupled PI Controller for Quadruple Tank System," International Journal of Science and Research, vol. 3, no. 5, 2014, pp. 318-23.

[3]. J. K. Henrik, "The Quadruple-Tank Process: A Multivariable Laboratory Process with an Adjustable Zero," IEEE Transactions on Control Systems Technology, vol.8, no. 3, 2000, pp. 456-465.

[4]. J. Jayaprakash, T. SenthilRajan, and T. HarishBabu. "Analysis of Modelling Methods of Quadruple Tank System," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 3, no. 8, 2014, pp. 11552-11565

[5]. A. Pedro and S. Antonio, "Multivariable Control Systems: an Engineering Approach," Springer Science & Business Media, 2006.

[6]. R. S. Mani Malar, "Modeling of Quadruple Tank System Computing Techniques," European Journal of Scientific Research, vol. 29, no. 2, 2009, pp.249-264.

[7]. V. D. Angeline and N. Devarajan, "Design of Decentralized PI Controller Using Model Reference Adaptive Control for Quadruple Tank Process," International Journal of Engineering and Technology, vol. 5, no. 6, 2014, pp. 5057-5066.

[8]. B. Terry LM. "Industrial Automated Systems: Instrumentation and Motion Control," Cengage Learning, 2010.

[9]. E. Bristol, "On a New Measure of Interaction for Multivariable Process Control,"IEEE Trans. Automat. Contr., vol. 11, 1966, p. 133.

[10]. W. Liuping, S. Chai, D. Yoo, Lu Gan, and Ki Ng, "PID and Predictive Control of Electrical Drives and Power Converters Using MATLAB/Simulink," John Wiley & Sons, 2015.

[11]. D. Aniruddha, M.-T. Ho, and S. P. Bhattacharyya, "Structure and Synthesis of PID Controllers, "Springer Science & Business Media, 2013.

[12]. G. J. Klir, and B. Yuan, "Fuzzy Sets and Fuzzy Logic: Theory and Applications," Possibility Theory versus Probab. Theory, vol. 32, no. 2, 1996, pp. 207-208.

[13]. I. Ion. "A Mamdani Type Fuzzy Logic Controller. Fuzzy Logic: Controls, Concepts, Theories and Applications," 2012, pp. 325-350.

[14]. Akka Ali, “Commande par l’utilisation des techniques du soft computing et leurs optimisations: Application à la commande d’un quadruple réservoir,” Ph.D. Thesis, USTO University, Oran, 2018.

[15]. W. Dongyun and F. Fan, "Parameters Tuning of Fuzzy Controller for Rotated Pendulum Based on Improved Particle Swarm Optimization," IEEE International Conference on Computational Intelligence and Software Engineering, 2009, pp. 1-5.

[16]. A. Thamallah, A. Sakly, and F. M'Sahli, "A new constrained PSO for fuzzy predictive control of Quadruple-Tank process," Measurement, vol. 136, 2019, pp.93-104.

[17]. T. Dong-Ping and L. Nai-Qian. "Fuzzy Particle Swarm Optimization Algorithm," IEEE International Joint Conference on Artificial Intelligence, 2009. pp. 263-267.

[18]. S. Yuhui, et al. "Particle Swarm Optimization: Developments, Applications and Resources," IEEE Proceedings of the 2001 congress on evolutionary computation, 2001, pp. 81-86.

[19]. A. M. Nabab, "Particle Swarm Optimization: Algorithm and its Codes in Matlab,"https://www.researchgate.net/publication/297245624_Particle_Swarm_Optimization_Algorithm_and_its_Codes_in_MATLAB

[20]. M. Clerc and J. Kennedy, "The Particle Swarm – Explosion, Stability, and Convergence in a Multidimensional Complex Space," IEEE Transactions on Evolutionary Computation, vol. 6, no. 1, 2002, pp. 58–73.

[21]. J. Robinson, S. Sinton, and Y. Rahmat-Samii, "Particle Swarm, Genetic Algorithm, and Their Hybrids: Optimization of a Profiled Corrugated Horn Antenna," In Antennas and Propagation Society International Symposium, IEEE Press, New York, vol. 1, 2002, pp. 314–317.

[22]. D. Swagatam, A. Abraham, and A. Konar, "Particle Swarm Optimization and Differential Evolution Algorithms: Technical Analysis, Applications and Hybridization Perspectives," In Advances of Computational Intelligence in Industrial Systems, Springer, Berlin, Heidelberg, 2008, pp. 1-38.

[23]. O. Chao, and W. Lin. "Comparison between PSO and GA for Parameters Optimization of PID Controller," IEEE International Conference on Mechatronics and Automation, 2006, pp. 2471-2475.

[24]. K. Hachemi and B. Mazari, "Etude comparative des régulateurs PID et flou: autopilotage d'un moteur synchrone à aimants permanents," Damascus University Journal, Vol. 21, No. 2, 2005, pp. 43-64.

[25]. P. Borne, J. Rozinoer, J. Y. Dieulot, and L. Dubois, "Introduction à la commande floue," Edition Technip, Paris, 1998.