Optimizing the fuzzy PI controller for quadruple tank system using multi-objective Jaya algorithm

Nguyen Anh Tuan[1,2,*], **Ho Pham Huy Anh**[1]

[1]Ho Chi Minh City University of Technology (HCMUT), VNU-HCM
[2]Sai Gon University (SGU)
* Corresponding author: nguyenanhtuan.sdh21@hcmut.edu.vn

Received: 1 March 2024; Accepted: 15 March 2024

Abstract:

This article proposes the Jaya optimization algorithm for the multi-objective optimization of parameters for a Fuzzy Proportional-Integral (Fuzzy PI) controller, referred to as MOJaya. The purpose is to control the water level of a quadruple-tank system (QTS). The controlled plant is a nonlinear and uncertain Multiple-Input Multiple-Output (MIMO) system, which is affected by various factors such as parameter uncertainties, external disturbances, and significant time delays. These factors have a considerable impact on the stability of the control system. The parameters of the fuzzy controller and the normalization coefficients of the Fuzzy PI controller are optimized offline using the multi-objective Jaya optimization (MOJaya) algorithm. To demonstrate the effectiveness of the proposed algorithm, it was compared with the Fuzzy PI and PI-MOJaya algorithms. The simulation results indicate that the proposed algorithm outperforms the other algorithms in terms of control quality.

Keywords: Quadruple-tank system (QTS); Fuzzy PI controller; PI-MOJaya controller; Multi-objective Jaya optimization (MOJaya); Optimal parameters; Water level.

1. Introduction

The quadruple-tank system in the laboratory is representative of a highly nonlinear MIMO system with strong interactions between the input voltages and output fluid levels, adding to the nonlinearity and complexity of control. Therefore, controlling this system has attracted numerous research efforts, ranging from classical to modern algorithms. Initially, the robust proportional-integral-derivative controller (PID) [1] controlled the water level of a quadruple-tank system. Next, the Fractional-Order SMC algorithm [2] controlled the water level of a Modified QTS, demonstrating fast convergence and reduced chattering compared to conventional SMC. In [3], the authors proposed a predictive control algorithm based on the system's past trajectory data to control the water levels of the four tanks. Similarly, using the predictive control algorithm [4], the authors designed a constrained model predictive controller using a linearized state-space model of the QTS.

The machine learning algorithms being extensively researched and explored owing to their high effectiveness are as follows. First, the emotional learning-based controller algorithm [5] is proposed, followed by the machine learning-based QTS model algorithm [6], which is applied to efficiently control the QTS. Furthermore, researchers have increasingly applied optimization algorithms inspired by biological phenomena owing to their high effectiveness. First, a Genetic Algorithm (GA) is used to optimize the parameters of the modified active disturbance rejection controller [7] for controlling the water level of the QTS. The results show a quick response and good noise rejection. Another study [8] proposed using a GA to optimize the parameters of the PI controller for controlling a coupled tank system, achieving an output response without overshooting. Additionally, a study [9] suggested applying the PSO algorithm to optimize the parameters of the PI controller for controlling the QTS, resulting in a reduced overshoot compared to manual tuning. To effectively address multi-objective problems, evolutionary algorithms for multi-objective optimization have been developed. A study [10] proposed the Multi-objective Particle Swarm Optimization (MOPSO) algorithm to assist the PSO algorithm in effectively solving multi-objective problems demonstrated through the Pareto front. Subsequently, a study [11] applied this MOPSO algorithm to optimize the parameters of the PI controller for controlling the water level of the QTS. Studies [12], [13] proposed the MOPSO algorithm to optimize the T-S Fuzzy controller parameters to find suitable control rules. These studies demonstrate the application of the proposed algorithm to control MIMO nonlinear systems. However, the simulation results still exhibited overshoot and steady-state errors. Another simple yet effective algorithm developed by Rao in 2016 is the Jaya optimization algorithm [14]. This algorithm only requires determining general parameters, such as the population size and number of iterations, without specifying algorithm-specific parameters that would affect its effectiveness, unlike some other optimization algorithms. Rao further enhanced this algorithm by dividing the total population into sub-population groups to search for the best solution in different areas of the search space [15], [16]. The study [16] applied the Jaya algorithm to optimize the parameters of a new adaptive multilayer fuzzy controller for effectively controlling the QTS. Additionally, to address multi-objective optimization problems, Rao developed and applied the Multi-Objective Optimization Jaya algorithm (MOJaya) [17], [18], [19], which simultaneously handles multiple objectives based on priority principles and crowding distance evaluation. Subsequently, a study [20] demonstrated the effective application of the MOJaya algorithm to optimize the power flow in solving the optimal power flow problem. Furthermore, [21] applied the MOJaya algorithm to effectively control the gait of a biped robot to mimic human-like motion. Finally, the particle swarm optimization (PSO) - gravitational search algorithm (GSA) optimization algorithm is applied to optimize the parameters of the Fuzzy PI controller for controlling a brushless DC motor [22]. The study shows its effectiveness compared to the Fuzzy PI controller and the conventional PI controller. The Fuzzy PI control algorithm is effective for systems with relatively fast dynamics. We believe that combining this algorithm with multi-objective optimization algorithms can effectively control MIMO nonlinear systems. Inspired by the strengths of the MOJaya algorithm in solving multi-objective problems, this study proposes the MOJaya algorithm for controlling the liquid level of the QTS system.

Contributions of this study:

- Successful application of the multi-objective optimization algorithm MOJaya to optimize the parameters of the Fuzzy PI controller for controlling the QTS system, which is representative of a highly nonlinear, intensely interactive, and difficult-to-control MIMO system.
- Through simulation results, the control quality of the MOJaya optimization algorithm is evaluated to be more effective than the conventional Fuzzy PI and PI-MOJaya algorithms in controlling the QTS system.

The structure of this study consists of five parts. An introduction is presented in Section 1. Section 2 describes the computational model and object description. The proposed controller is then presented in Section 3. Section 4 presents the simulation results, and Section 5 concludes the paper.

2. Modeling and description of the control plant

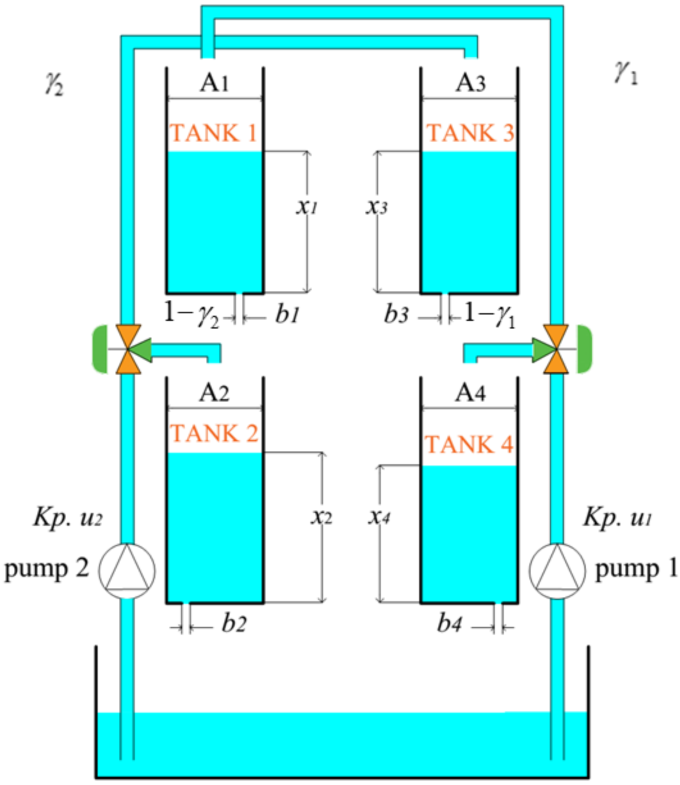


Figure 1. Quadruple-tank system (QTS).

Figure 1 illustrates a nonlinear MIMO system with time delays based on the quadruple-tank system (QTS) model by Quanser. The parameters are described in Table 1 [16]. The two inputs of the system are the control voltage of pump 1 u_1 and the control voltage of pump 2 u_2, and the two or four outputs are the water levels of tanks 1, 2, 3, and 4 corresponding to x_1, x_2, x_3 and x_4. Pump 1 directly controls the water level of Tank 1, while the drain pipe of Tank 1 influences the water level of Tank 2. On the other hand, pump 2 directly controls the water level of tank 3, while the drain pipe of tank 3 influences the water level of tank 4. Additionally, there are cross-coupling effects between motor 1 and Tank 4 and between motor 2 and Tank 2. This mutual interaction further enhances the system's nonlinearity. The mathematical equations of the system are as follows [16]:

$$\begin{cases} \dfrac{dx_1}{dt} = \dfrac{K\gamma_1 u_1(t)}{A_1} - \dfrac{b_1 C\sqrt{2gx_1}}{A_1} \\ \dfrac{dx_2}{dt} = \dfrac{K(1-\gamma_2)u_2(t)}{A_2} + \dfrac{b_1 C\sqrt{2gx_1}}{A_2} - \dfrac{b_2 C\sqrt{2gx_2}}{A_2} \\ \dfrac{dx_3}{dt} = \dfrac{K\gamma_2 u_2(t)}{A_3} - \dfrac{b_3 C\sqrt{2gx_3}}{A_3} \\ \dfrac{dx_4}{dt} = \dfrac{K(1-\gamma_1)u_1(t)}{A_4} + \dfrac{b_3 C\sqrt{2gx_3}}{A_4} - \dfrac{b_4 C\sqrt{2gx_4}}{A_4} \end{cases} \quad (1)$$

Where u_1, u_2 representing the control voltages of pump 1 and pump 2, respectively, and x_1, x_2, x_3 and x_4 meaning the water levels of tanks 1, 2, 3, and 4, Table 1 presents the definitions and values of the system parameters.

Table 1. Physical meaning and values used in simulation.

Symbols	Physical meaning	Value	Unit
A_1	The surface area of the inner tank 1	16.619	cm^2
A_2	The surface area of the inner tank 2	16.619	cm^2

Symbols	Physical meaning	Value	Unit
A_3	The surface area of the inner tank 3	16.619	cm^2
A_4	The surface area of the inner tank 4	16.619	cm^2
b_1	The cross-sectional area of the drain pipe in Tank 1	1	cm^2
b_2	The cross-sectional area of the drain pipe in Tank 2	1	cm^2
b_3	The cross-sectional area of the drain pipe in Tank 3	1	cm^2
b_4	The cross-sectional area of the drain pipe in Tank 4	1	cm^2
C	The discharge coefficient of the drain pipe	0.8	
g	Gravitation	981	cm/s^2
K_p	Pump flow value	50	$cm^3/(s.V)$
γ_1	The flow ratio between Tank 1 and Tank 4	90	%
γ_2	The flow ratio between Tank 2 and Tank 3	90	%

The problem addressed in this paper is the design of a water level controller for tanks 2 and 4 to follow their respective reference signals.

3. The proposed controller

From Equation (1), the state equation can be rewritten as follows:

$$\begin{cases} \dot{x} = f(x) + g(u) \\ y = h(x) \end{cases} \tag{2}$$

Select the outputs:

$$\begin{cases} y_1 = x_2 \\ y_2 = x_4 \end{cases} \tag{3}$$

A controller is designed such that the two outputs x_2, and x_4 follow the reference signals r_1 and r_2 respectively. The proposed controller in Figure 2 is a Fuzzy PI controller with the parameters of the Fuzzy PI controller optimized by the MOJaya algorithm. The proposed controller consists of two Fuzzy PI controllers. Fuzzy PI Controller 1 was designed to control the water level of Tank 2, and Fuzzy PI Controller 2 was designed to control the water level of Tank 4. The block diagram of the proposed controller is illustrated in Figure 2. Fuzzy PI Controller 1: The error signals e_1 and de_1 are used as the two inputs of Fuzzy PI Controller 1, and the output is du_1. The normalization factors Ke_1, Kde_1, and the parameters of the input and output membership functions of the fuzzy controller (Fuzzy 1) were optimized using the MOJaya algorithm. With the optimized parameters, the fuzzy PI Controller 1 will control the water level of Tank 2 according to the desired signal r_1. The fuzzy PI Controller 2 is designed similarly to Fuzzy PI Controller 1, where the error signals e_2 and de_2 are used as the two inputs of Fuzzy PI Controller 2, and the output is du_2. The normalization factors Ke_2, Kde_2, and the parameters of the input and output membership functions of the fuzzy controller (Fuzzy 2) were optimized using the MOJaya algorithm. With the optimized parameters, fuzzy PI Controller 2 controls the water level of Tank 4 according to the desired signal r_2.

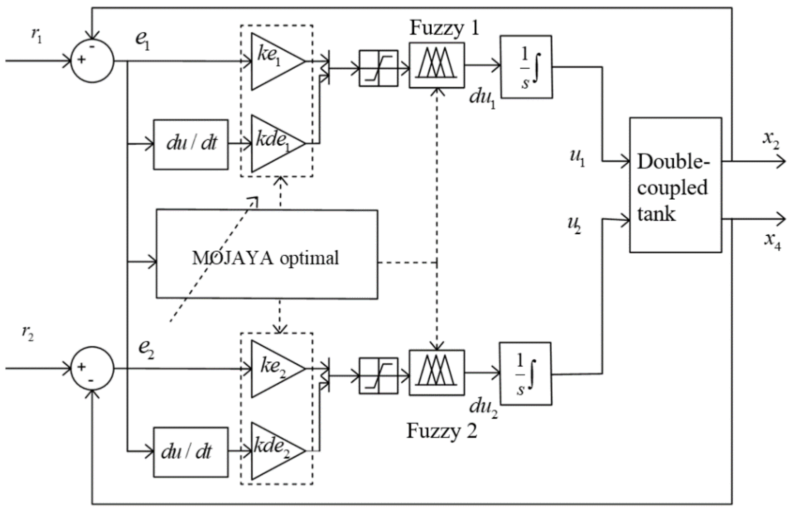


Figure 2. Proposed Fuzzy PI controller with parameters optimized by MOJaya algorithm.

3.1. Fuzzy PI controllers and problem formulation

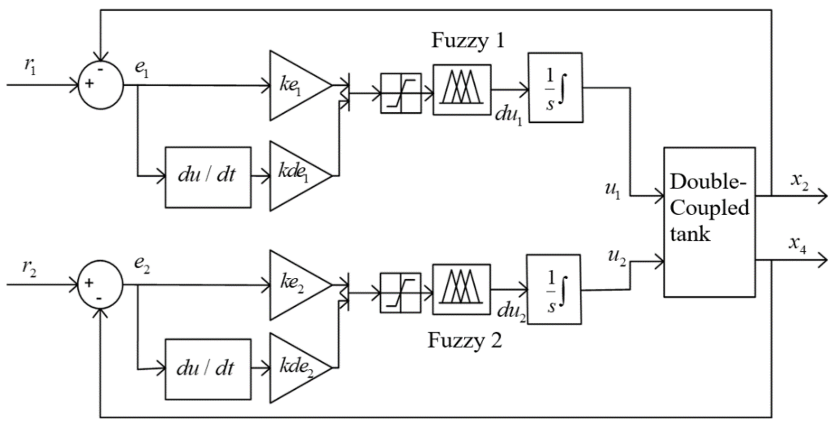
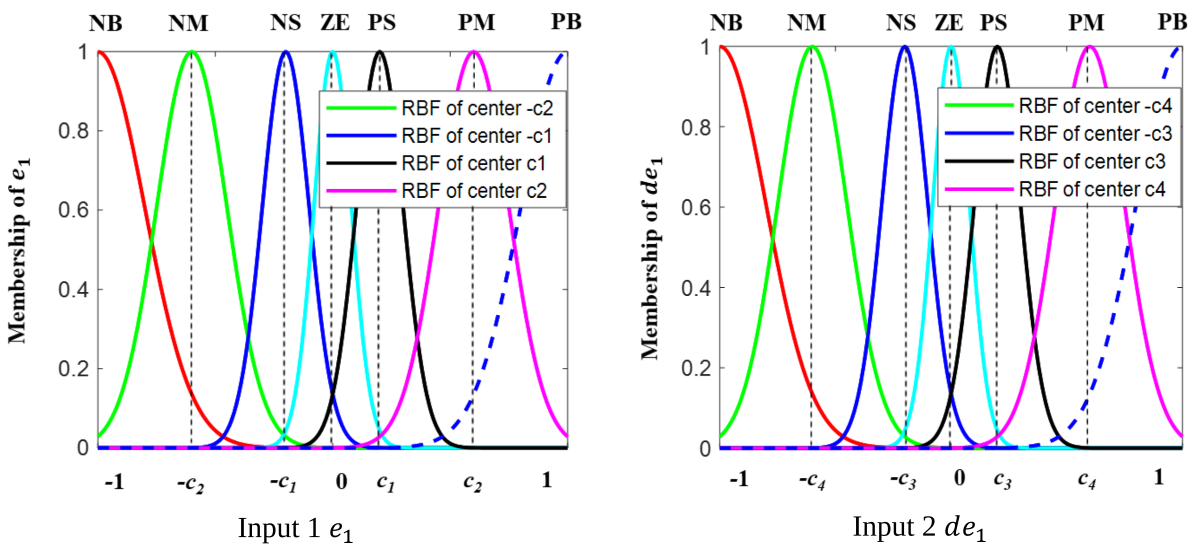
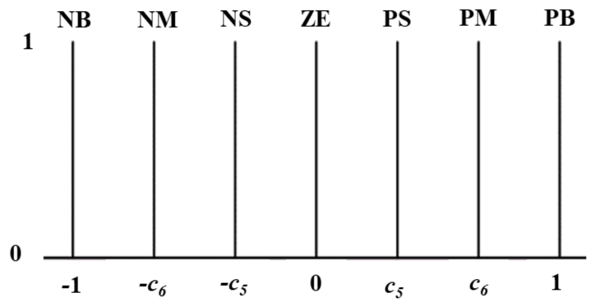


Figure 3. Simulation Diagram of Fuzzy PI Controllers for QTS System.

Figure 3 depicts the simulation diagram of the Fuzzy PI controllers controlling the QTS system. The controller consists of two identical Fuzzy PI controllers, namely Fuzzy PI 1 and Fuzzy PI 2. Fuzzy PI Controller 1 consists of Fuzzy Set 1 with two inputs e_1 and de_1, in the form of Gaussian membership functions with centers c_1-c_4, and one output du_1, in the form of a line with weights c_5 and c_6 as depicted in Figure 4.





Membership function of du_1

Figure 4. Linguistic values of Fuzzy PI controller 1 before optimization.

Table 2. Fuzzy rules of Fuzzy PI Controller 1.

du_1		de_1						
		NB	NM	NS	ZE	PS	PM	PB
e_1	NB	NB	NB	NB	NB	NM	NS	ZE
	NM	NB	NB	NB	NM	NS	ZE	PS
	NS	NB	NB	NM	NS	ZE	PS	PM
	ZE	NB	NM	NS	ZE	PS	PM	PB
	PS	NM	NS	ZE	PS	PM	PB	PB
	PM	NS	ZE	PS	PM	PB	PB	PB
	PB	ZE	PS	PM	PB	PB	PB	PB

The fuzzy rule base of Fuzzy PI Controller 1 is presented in Table 2. The values NB, NM, NS, ZE, PS, PM, and PB represent the linguistic terms for big negative, medium negative, small negative, zero, small positive, medium positive, and big positive.

Fuzzy PI Controller 2: This controller is designed similarly to Fuzzy PI Controller 1. Fuzzy set 2 has two inputs e_2, and de_2, and one output du_2. The PI parameters were Ke_2 and Kde_2. The membership function weights for the input variables are c_7 - c_{10} and those for the output variable are c_{11} and c_{12}.

The calculation of selecting the parameters for Fuzzy PI Controller 1 and Fuzzy PI Controller 2 to achieve the desired control quality is a highly complex task. There are various approaches for parameter selection. This paper proposes the Jaya multi-objective optimization algorithm, presented in Section 3.2, as a recommended method.

3.2. The Jaya optimization algorithm and MOJaya algorithm

The Jaya optimization algorithm

The Jaya optimization algorithm [16] was developed by Rao in 2016 based on the concept of searching for the best solution among individuals in a population. Individuals tend to move towards better solutions and avoid worse solutions. This is advantageous compared to other optimization algorithms. Moreover, unlike other algorithms, the Jaya algorithm does not require specific parameters that require careful selection. It only relies on general parameters such as population size, number of design variables, and maximum number of iterations. The individuals of the solution are moved by updating their positions according to Equation (4).

$$X'_{j,k,i} = X_{j,k,i} + r_{1,j,i}(X_{j,best,i} - |X_{j,k,i}|) - r_{2,j,i}(X_{j,worst,i} - |X_{j,k,i}|) \quad (4)$$

Where, j =1,2,..,m are the number of design variables; =1,2,..,n are the number of candidate solutions, and i is the iteration.

Where is the value of the *j*th variable for the *k*th individual in the *i*th iteration, $X_{j,best,i}$ is the value of the *j*th variable for the best individual, $X_{j,worst,i}$ is the value of the *j*th variable for the worst individual, and $r_{1,j,i}$ and $r_{2,j,i}$ are two random numbers for the *j*th variable in the *i*th iteration within the range [0,1]. $f(x)$ is the objective function to be minimized (or maximized). For a given individual, if the new solution after the movement is better than the previous solution before the action, then this new solution is accepted, and the corresponding individual is updated. Conversely, if the new solution is inferior, it is discarded, and the current solution is maintained.

The proposed multi-objective Jaya optimization algorithm (MOJaya)

The MOJaya optimization algorithm was proposed to solve multi-objective problems. This method was introduced in 2016 by Rao [19]. The algorithm also updates the positions according to Equation (4), but this update is based on multiple objective functions. The detailed steps of this algorithm are presented in [19]. A flowchart of this algorithm is illustrated in Figure 5.

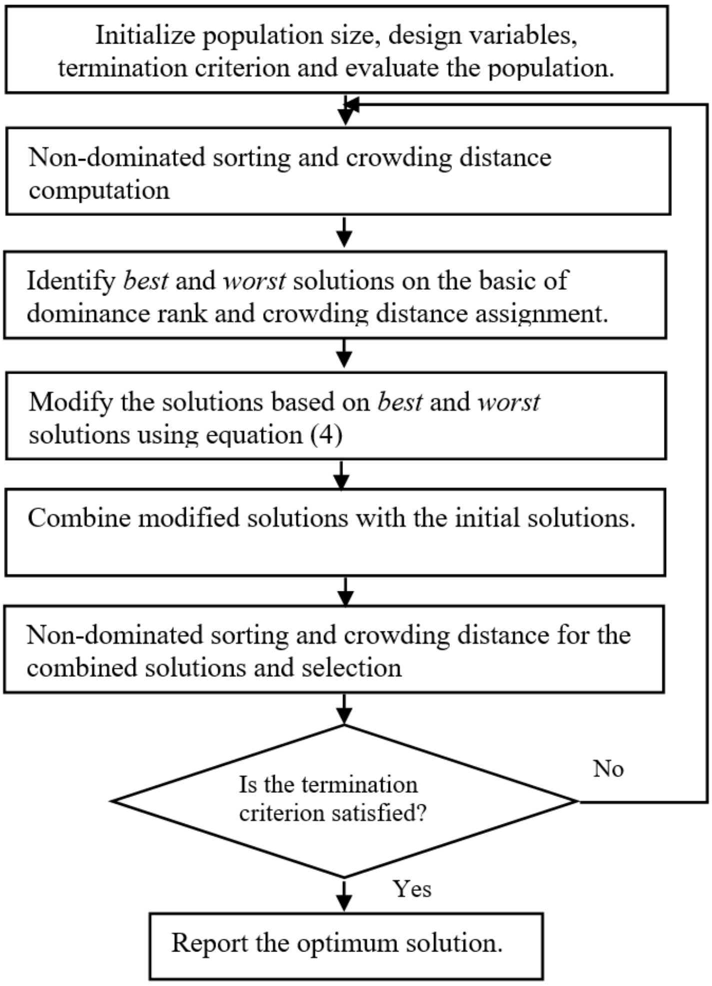


Figure 5. Flowchart of MOJaya algorithm.

Applying the MOJaya optimization algorithm to the QTS system

The parameters of the Fuzzy PI controller consist of 16 values: 12 weights c_1-c_6, c_7- c_{12} of the input and output membership functions of both Fuzzy 1 and Fuzzy 2 controllers, and four normalization factors Ke_1, Kde_1 of Fuzzy PI Controller 1, and Ke_2, Kde_2 of Fuzzy PI Controller 2.

The widths of the Gaussian membership functions for the inputs of both Fuzzy PI controllers 1 and 2 were chosen to be fixed as specified in Table 3.

Table 3. Widths of the Gaussian membership functions for the inputs of Fuzzy PI controllers 1 and 2.

	σ_{NB}	σ_{NM}	σ_{NS}	σ_{ZE}	σ_{PS}	σ_{PM}	σ_{PB}
Widths	0.2	0.15	0.1	0.08	0.1	0.15	0.2

The following parameters were selected for the MOJaya algorithm:

The number of design variables j = 16, population size k = 50, and iteration i = 20.

The upper bound (UB) and lower bound (LB) limits for the parameters to be optimized are selected under the following constraints:

c_1- c_{12}: UB = 1, LB = 0. $c_2 > c_1$, $c_4 > c_3$, $c_6 > c_5$, $c_8 > c_7$, $c_{10} > c_9$, $c_{12} > c_{11}$ to ensure the order of the membership function values of the fuzzy set, as depicted in Figure 4.

Ke_1 and Ke_2: UB = 0.9, LB = 0.01.

Kde_1 and Kde_2: UB = 1.2, LB = 0.01.

The objective function is selected based on the Integral of Time multiplied by Squared Error (ITSE) criterion as follows:

$$f_1 = \int_0^\infty e_1^2(t)dt + \rho \int_0^\infty u_1^2(t)dt \qquad (5)$$

$$f_2 = \int_0^\infty e_2^2(t)dt + \rho \int_0^\infty u_2^2(t)dt \qquad (6)$$

Where ρ is chosen as ρ = 1, $e_1 = r_1 - x_2$, $e_2 = r_2 - x_4$.

The MOJaya algorithm aims to find an optimal solution by minimizing the two objective functions f_1 and f_2 corresponding to the water level outputs x_2 and x_4 of the QTS system. In this case, the errors between the water level outputs x_2 and x_4 and their respective reference signals r_1 and r_2 are minimized.

The results of the MOJaya optimization process are represented by a Pareto plot, which is a set of optimal solutions. From there, the best set of optimal solutions is selected to achieve the minimum values of the objective functions f_1 and f_2 in Equations (5) and (6).

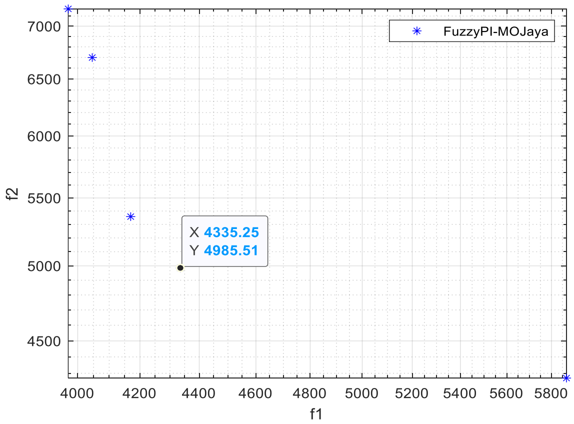


Figure 6. The Pareto chart of the MOJaya algorithm.

The set of four optimal solutions found in the Pareto chart in Figure 6 after 20 generations, where the best optimal solution is selected, has values of f_1 = 4335.25 and f_2 = 4985.51. The corresponding values of the variables for this solution are presented in Table 4, Table 5, and Figure 7.

Table 4. Optimal results of Fuzzy PI controller 1 obtained by MOJaya algorithm.

Centers	c_1	c_2	c_3	c_4	c_5	c_6
	0.234	0.867	0.444	0.513	0	0.984
Normalization factors	Ke_1	Kde_1				
	0.112	0.915				

Table 5. Optimal results of Fuzzy PI controller 2 obtained by MOJaya algorithm.

Centers	c_7	c_8	c_9	c_{10}	c_{11}	c_{12}
	0.111	1	0.855	1	0.201	0.491
Normalization factors	Ke_2	Kde_2				
	0.01	0.085				

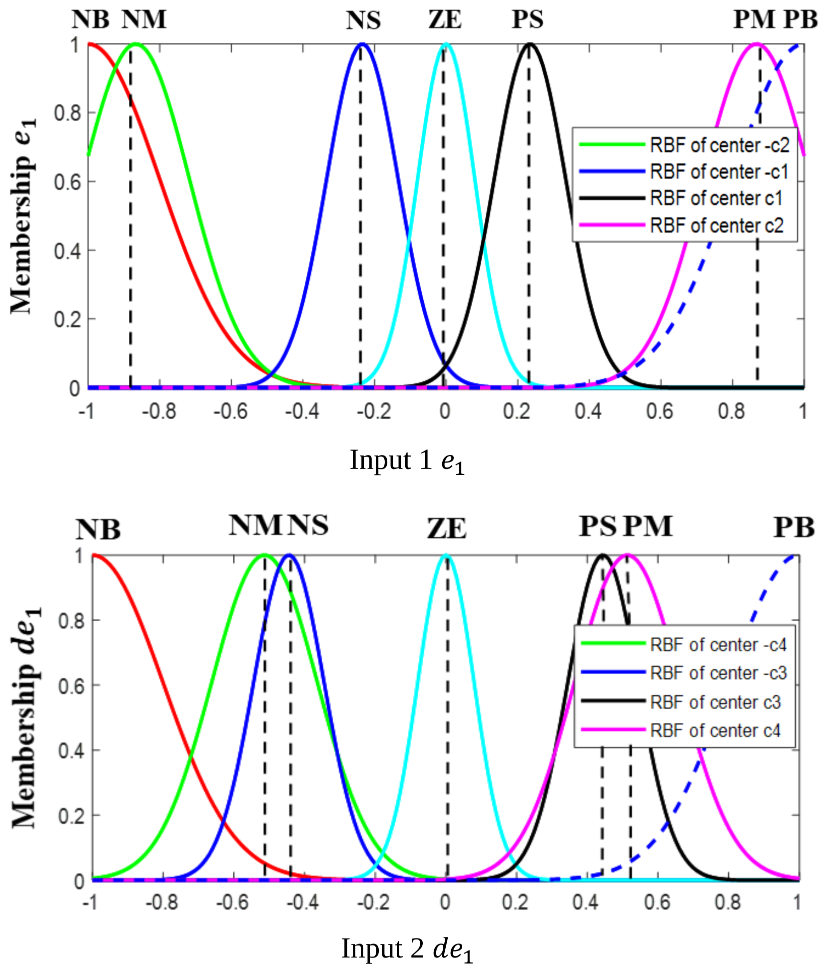


Figure 7. Linguistic values of Fuzzy PI Controller 1 after optimization.

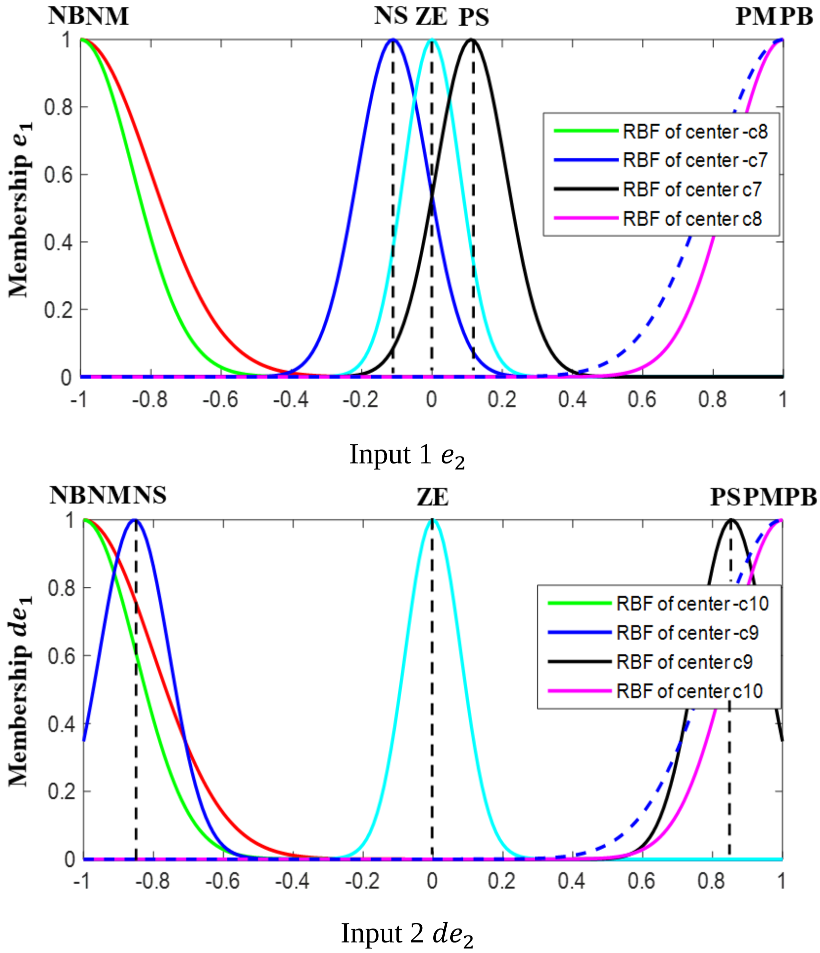


Figure 8. Linguistic values of Fuzzy PI Controller 2 after optimization.

After optimization, the values of the centers of the Gaussian membership functions for both fuzzy sets change, as shown in Figure 7 and 8, while still satisfying the initial constraints mentioned above. These parameters are used for simulation in Section 4.

4. Simulation results

Perform the simulation of the Fuzzy PI controller using the parameters optimized by the MOJaya algorithm in Section 3 to control the QTS system. The results are compared with those of the PI-MOJaya and traditional Fuzzy PI algorithms.

4.1. The simulation results compare the three algorithms

Simulation results were used to compare the three controller algorithms. They include PI-MOJaya, Traditional Fuzzy PI, and Fuzzy PI controllers with parameters optimized using the MOJaya algorithm, as depicted in Figure 9 with varying reference change signals. The simulation results in Figure 9 indicate that the PI-MOJaya algorithm achieves a rapid response with zero steady-state error but exhibits a higher overshoot. In contrast, the traditional Fuzzy PI algorithm shows a slower response but without overshoot, reaching a steady state after approximately 60s with zero steady-state error. Additionally, the Fuzzy PI-MOJaya algorithm achieves a faster response than the Fuzzy PI algorithm, reaching steady state after approximately 50s with zero steady-state error and zero overshoot. The control quality of the algorithms is compared in Table 6. Because the PI-MOJaya algorithm exhibits overshoot, the quality indices are not compared in this table. This comparison shows that the Fuzzy PI-MOJaya algorithm outperforms the traditional Fuzzy PI algorithm in all quality indices.

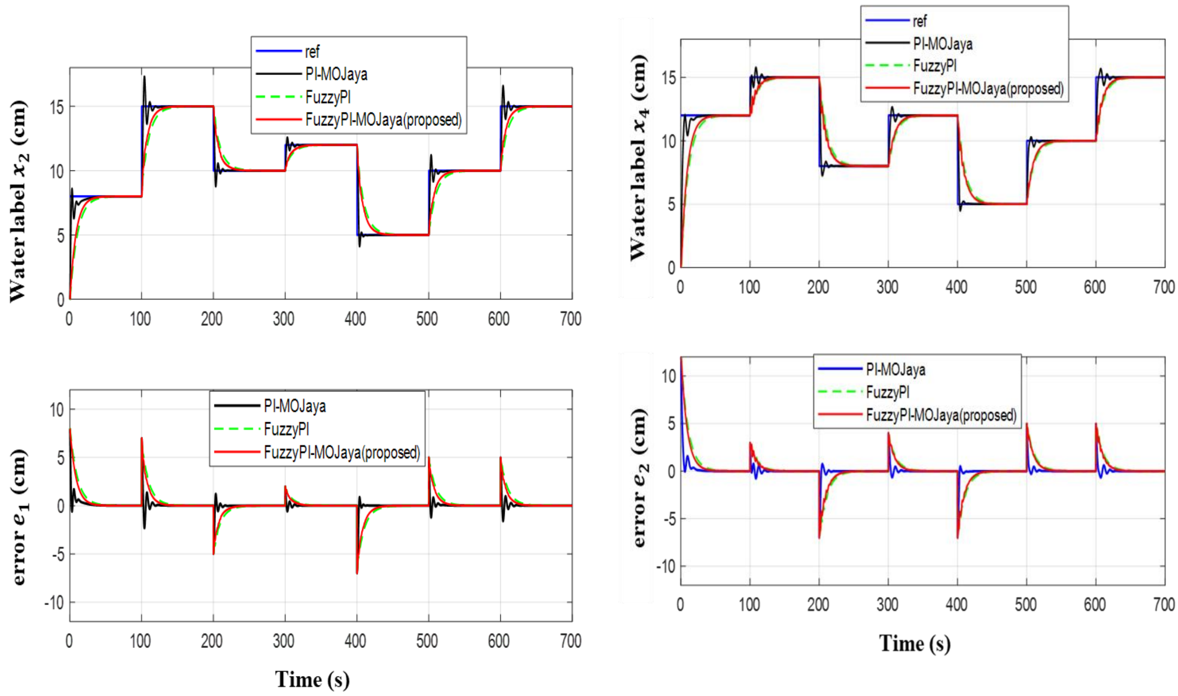


Figure 9. Response results of outputs and of the QTS system controlled by Fuzzy PI-MOJaya, PI-MOJaya algorithm, and traditional Fuzzy PI.

Table 6. Comparison of control quality indices among algorithms.

	x_2		x_4	
	Fuzzy PI	**Fuzzy PI-MOJaya**	**Fuzzy PI**	**Fuzzy PI-MOJaya**
IAE	383.3	298.2	457.4	390.8
ISE	1175	894.7	1729	1478
ITAE	1.072*10^5	8.183*10^4	1.232*10^5	1.051*10^5
ITSE	4937	4700	5586	5346

4.2. The simulation results of the proposed MOJaya algorithm with an untrained reference signal are as follows

Optimal results of the Fuzzy PI controller parameters using the proposed MOJaya algorithm with untrained reference signals. The maximum reference signal levels of and increase from 15 cm to 25 cm (an increase of 66.6 %) and from 15 cm to 20 cm (a rise of 33.3 %), respectively, compared to the trained reference signals, as shown in Figure 10.

Figure 10 presents the response results of outputs and of the QTS system using the Fuzzy PI controller with Gaussian membership functions and parameters optimized using the MOJaya algorithm. Despite the untrained and significantly broader reference signals, the response was satisfactory, with zero steady-state error and overshoot. The simulations above show that the proposed algorithm provides good control quality compared with other algorithms for controlling highly nonlinear MIMO QTS systems.

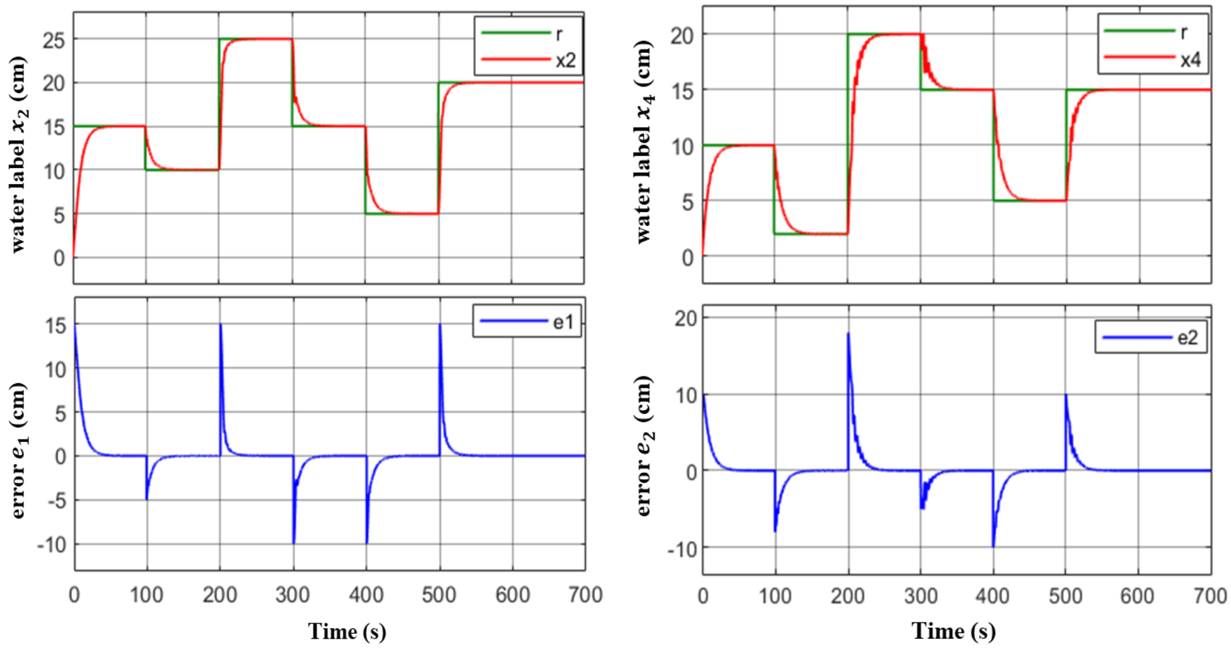


Figure 10. Response results of outputs x_2 and x_4 of the QTS system controlled by Fuzzy PI-MOJaya with untrained reference signals.

5. Conclusion

This paper introduces the MOJaya multi-objective optimization algorithm for optimizing the Fuzzy PI controller's parameters with the fuzzy set's inputs utilizing Gaussian membership functions for water level control in the MIMO QTS system, which exhibits high nonlinearity and interaction characteristics. The parameters of the Fuzzy PI controller are optimized offline using the MOJaya multi-objective optimization algorithm. Simulation experiments were conducted and compared with the PI-MOJaya algorithm and conventional Fuzzy PI algorithm to demonstrate the effectiveness of the proposed algorithm. The simulation results show that the proposed algorithms achieve good output responses according to the reference signal, indicating that the MOJaya optimal algorithms find optimal solutions for optimizing the parameters of the Fuzzy PI controller for controlling the MIMO QTS system. Therefore, this algorithm can be applied to control other nonlinear MIMO systems.

Acknowledgment

This study was funded by Saigon University (SGU) under grant number CSB2022-53. The authors acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for supporting this study.

References

[1] C. M. Ionescu, A. Maxim, C. Copot, and R. D. Keyser, “Robust PID auto-tuning for the quadruple tank system,” *IFAC-PapersOnLine*, vol. 49, no. 7, pp. 919-924, 2016, doi: 10.1016/j.ifacol.2016.07.313.

[2] S. Sutha , P. Lakshmi, and S. Sankaranarayanan, “Fractional-order sliding mode controller design for a modified quadruple tank process via multi-level switching,” *Comput. Electr. Eng.*, vol. 45, pp. 10-21, Jul. 2015, doi: 10.1016/j.compeleceng.2015.04.012.

[3] J. R. Salvador, D. M. de la Peña, T. Alamo, and A. Bemporad, “Data-based predictive control via direct weight optimization,” *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 356-361, 2018, doi: 10.1016/j.ifacol.2018.11.059.

[4] Z. Zidane, “Constrained model predictive control for the quadruple-tank process,” *Int. Adv. Res. Eng. J.*, vol. 3, no. 3, pp. 175-181, 2019, doi: 10.35860/iarej.535552

[5] B. Debnath and S. J. Mija, “Emotional learning based controller for quadruple tank system - An improved stimuli design for multiple set-point tracking,” *IEEE Trans. Ind. Electron.*, vol. 68, no. 11, pp. 11296-11308, Nov. 2021, doi: 10.1109/TIE.2020.3038083.

[6] A. M, “Modeling of quadruple tank system using machine learning algorithms,” *Res. Sq.*, 2023, doi: 10.21203/rs.3.rs-1884689/v1.

[7] Z. S. Hashim et al., “Robust liquid level control of quadruple tank system: A nonlinear model-free approach,” *Actuators*, vol. 12, no.3, 2023, Art. no. 119, doi: 10.3390/act12030119.

[8] M. Chew, F. Juwono, and W. K. Wong, “GA-based optimization for multivariable level control system: A case study of multi-tank system,” *Eng. J.*, vol. 26, no. 5, pp. 25-41, May 2022, doi: 10.4186/ej.2022.26.5.25.

[9] N. Z. Abidin, S. Sahlan, and N. A. Wahab, “Optimization tuning of PI controller of quadruple tank process,” in *2013 Aust. Control Conf.*, Fremantle, WA, Australia, 2013, doi: 10.1109/AUCC.2013.6697294.

[10] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, “Handling multiple objectives with particle swarm optimization,”*IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 256-279, Jun. 2004, doi: 10.1109/TEVC.2004.826067.

[11] H.V.H. Ayala and L. Coelho, “Controller tuning using multiobjective particle swarm optimization applied to a quadruple-tank process,” *ABCM Symp. Ser. Mechatron.*, vol. 4, pp. 72-79, 2010.

[12] T. Ali, S. Anis, and M. Faouzi, “Design of multivariate predictive controller based on new constrained multi-objective PSO and T-S Fuzzy modeling approach,”. in *2015 7th Int. Conf. Model., Identif. Control*, Sousse, Tunisia, 2015, doi: 10.1109/ICMIC.2015.7409381.

[13] A. Thamallah, A. Sakly, and F. M'Sahli, “Constrained multiobjective PSO and T-S fuzzy models for predictive control,” *Turk. J. Electr. Eng. Comput. Sci.*, vol. 26, no. 6, pp. 3239-3257, 2018, doi: 10.3906/elk-1803-14.

[14] R. V. Rao, “Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems,” *Int. J. Ind. Eng. Comput.*, vol. 7, no. 1, pp. 19-34, 2016, doi: 10.5267/j.ijiec.2015.8.004.

[15] R. V. Rao and R. B. Pawar, “Self-adaptive multi-population Rao algorithms for engineering design optimization,” *Appl. Artif. Intell.* vo. 34, no. 3, pp. 187-250, 2020, doi: 10.1080/08839514.2020.1712789.

[16] C. V. Kien, N. N. Son, and H. P. H. Anh, “Adaptive MIMO Fuzzy Controller for Double Coupled Tank System optimizing by Jaya Algorithm,”. in *2020 5th Int. Conf. on Green Technol. and Sustain. Dev.*, Ho Chi Minh City, Vietnam, 2020, doi: 10.1109/GTSD50082.2020.9303140.

[17] R. V. Rao, D. P. Rai, J. Ramkumar, and J. Balic, “A new multi-objective Jaya algorithm for optimization of modern machining processes,” *Adv. Product. Eng. Magnage.*, vol. 11, no. 4, pp. 271-286, Dec. 2016, doi: 10.14743/apem2016.4.226.

[18] R. V. Rao, D. P. Rai, and J. Balic, “Optimization of abrasive waterjet machining process using multi-objective Jaya algorithm,”. *Marter. Proc.*, vol. 5, no. 2, part 1, pp. 4930-4938, 2018, doi: 10.1016/j.matpr.2017.12.070.

[19] R. V. Rao, and H. S. Keesari, “Rao algorithms for multi-objective optimization of selected thermodynamic cycles,” *Eng. Comput.*, vol. 37, pp. 3409-3437, Oct. 2021, doi: 10.1007/s00366-020-01008-9.

[20] F. Berrouk et al., “A new multi-objective Jaya algorithm for solving the optimal power flow problem,” *J. Electr. Syst.* vol. 14, no. 3, pp. 165-181, 2018.

[21] H. T. Thien, C. V. Kien, and H. P. H. Anh, "Optimized gait planning of biped robot using multi-objective Jaya algorithm," Int. J. Adv. Robot. Syst., vol. 17, no. 6, 2020, doi:10.1177/1729881420976344.

[22] Song, B., et al., "Design of fuzzy PI controller for brushless DC motor based on PSO–GSA algorithm," *Syst. Sci. Control Eng.*, vol. 8, no.1, pp. 67-77, 2020, doi: 10.1080/21642583.2020.1723144