



Real-time implementation of QFT, GA, and BFTPSO controller for pH neutralization system

Aparna V.* • D. N. Jamal

Department of Electronics and Instrumentation Engineering, B. S. Abdur Rahman Crescent Institute of Science & Technology, Chennai, India

Received 05 27 2020; accepted 05 17 2021
Available 06 30 2021

Abstract: The control of a pH process is complex because of severe nonlinearities in its behavior. A continuous pH neutralization process is usually represented as a first-order plus dead time system, but its gain varies for different operating points. Therefore, a conventional linear controller cannot be used, and the pH system was thus represented as a linear state-space model around an equilibrium point. This linear model was then used to compute the PID controller gains using robust and optimization techniques like quantitative feedback theory, bacterial foraging technique-based particle swarm optimization algorithm, and genetic algorithm. The corresponding controller gains resulting from the three algorithms were used to control the pH using a reconfigurable I/O device, NI myRIO-1900. Finally, the output time domain specifications and the servo and regulatory responses, resulting from the three algorithms, were compared in simulation and in real-time to deduce the appropriate tuning algorithm for this system.

Keywords: pH Control, PID Controller, Genetic Algorithm, Hybrid Bacterial Foraging Technique Particle Swarm Optimization, Quantitative Feedback Theory, Robust Control

*Corresponding author.

E-mail address: vaparna06@gmail.com (Aparna V.).

Peer Review under the responsibility of Universidad Nacional Autónoma de México.

1. Introduction

The pH neutralization setup is widely used in the pH treatment of wastewater or runoff water from pharmaceutical or biotech organizations and construction sites (Aparna, 2014). By and large, the pH neutralization process plays a significant role in several industries, as it is necessary to neutralize the pH of the industrial fluid wastes before they are discharged into the atmosphere. The principal purpose of controlling a pH neutralization system is to maintain the pH value of the discharged effluent as per the environmental safety regulation of 6 to 8. A highly acidic or alkaline pH would be fatal to the ecological surroundings. However, control of a pH neutralization system is highly challenging and complicated because of the inherent nonlinearity in the system (Abdullah et al., 2012; Bharathi et al., 2007; Ibrahim & Murray-Smith, 2007; Saji & Sasi, 2011; Shinsky, 1973). Classical control methods prove to be beneficial only for linear processes and for analytical testing. For real-time control, the performance of the designed controller should be able to bypass the effects of nonlinearities, modeling errors and uncertainties, dead time, interaction effect, and dynamic behavior in the pH neutralization process (Bequette, 1991; Faanes & Skogestad, 2004).

The most widely used controller for this type of system is continuous Proportional-Integral-Derivative (PID) control, and experiments and studies have been carried out to figure out the appropriate tuning method for the PID controller. Saji and Sasi (2011) employed particle swarm optimization and Ant colony optimization algorithm to tune the PID controller parameters. They achieved setpoint tracking, robustness, and sensitivity from their nonlinear controller (Saji & Sasi, 2011). Bingi et al. (2016) preferred particle swarm optimization algorithm over Ziegler Nichols method to tune PID controller for a pH neutralization plant (Bingi et al., 2016). Tadeo et al. (2000) used a loop shaping approach by combining robust and graphical loop shaping to achieve robustness and reduce the effect of model uncertainty in the controller performance (Tadeo et al., 2000). Shabani et al. (2010) employed quantitative feedback theory-based controller to attain setpoint tracking and disturbance rejection despite inherent model uncertainties (Shabani et al., 2010). Ali (2001) compared the performance of a standard PI controller with a globally linearizing control algorithm and gain scheduling strategy in effectively controlling a complex pH process over a wide range of operating conditions without constant retuning (Ali, 2001).

In another work, Bingi et al. (2018, 2020) designed a setpoint weighted fractional-order PID tuned using an accelerated particle swarm optimization algorithm to achieve superior setpoint tracking in a pH neutralization plant (Bingi et al., 2018, 2020).

To assuage this need for effective control of the pH neutralization system and achieve desired disturbance rejection, setpoint tracking, sensitivity, and robustness, several optimizations, self-tuning, intelligent, robust, and model-based approaches have been proposed (Barve & Nataraj, 1999; Ibrahim, 2010; Moore, 1995).

In this paper, sections 2 and 3 describe the pH neutralization setup and its mathematical model and analysis. Section 4 explains the various tuning algorithms adopted to design the controller, and sections 5 and 6 present and analyze the response of the controlled system with different tuning algorithms in simulation. Section 7 describes the real-time implementation of the controllers in the system. Section 8 presents the experimental results; finally, the suitable controller for the proposed process based on the simulation and experimental results is deduced.

2. pH neutralization system

The pH neutralization process controls or maintains the pH of a solution such that it remains a neutral solution.

The solution whose pH is to be monitored is fed into the water tank at the top. This solution from the water tank is supplied to the process tank through a solenoid valve (SV). Depending on the initial pH value of the solution and based on whether the solution is to be neutralized or made acidic/alkaline, alkaline (0.1 N Sodium Hydroxide solution) and acidic solution (0.1 N Hydrochloric acid solution) are added to the process tank through Pump 1 & 2. A stirrer mixes the contents of the process tank before its pH is measured. The pH probe and transmitter measure the pH value of the solution in the process tank and convert it into (0-5) V range. The measured pH value is the controller input, and the controller varies the strokes per minute of the pumps, changing the inflow of acidic and alkaline solution to the process tank based on the desired pH value to be maintained in the process tank. Once the pH value is controlled or retained at the desired value, the contents of the process tank are emptied into the collecting container. Three hand valves, HV1, HV2, and HV3, are provided to drain the contents of the alkaline tank, collecting tank, and acid tank, respectively. Figure 1 portrays the schematic diagram of the pH neutralization setup.

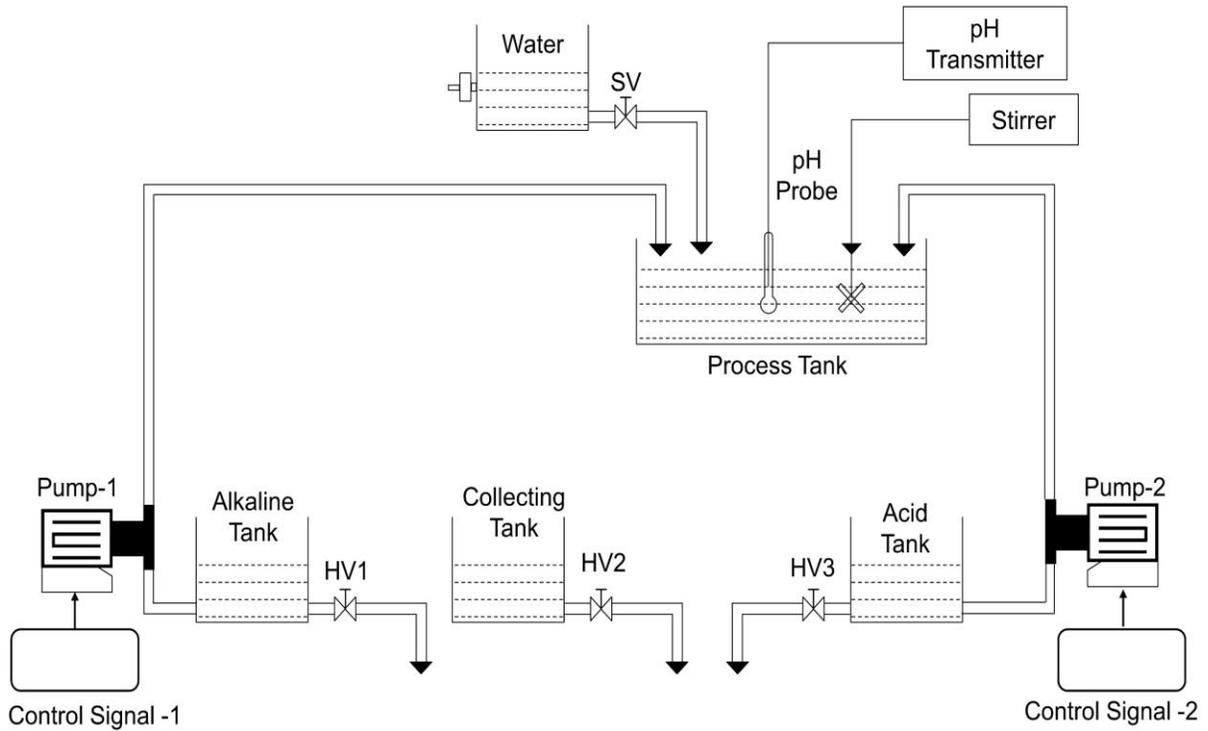


Figure 1. Schematic diagram of pH system.

The pH system is a Two-Input One-Output (TISO) system, where the concentration of the non-reacting acid solution and alkaline solution in the tanks (x_{c_a} , x_{c_b}) are the states and pH of the solution is the controlled variable of the system. Pumps 1 and 2 act as the final control elements and strokes per minute (SPM) of pumps 1 and 2, and inflow rates of acid and alkaline solutions to the reactor tank (Fw_a and Fw_b) are the variables manipulated by the controller.

3. Empirical modeling

3.1. System modeling

The general representation of a pH system is given by (Alina & Madalina, 2016; Bharathi et al., 2007; Biagiola et al., 2016; Darab et al., 2012; Elameen et al., 2014; Rodriguez & Loparo, 2004),

$$\frac{dx_{c_a}}{dt} = \frac{\{Fw_a Co_a - (Fw_a + Fw_b) x_{c_a}\}}{C} \quad (1)$$

$$\frac{dx_{c_b}}{dt} = \frac{\{Fw_b Co_b - (Fw_a + Fw_b) x_{c_b}\}}{C} \quad (2)$$

$$Y = -\log_{10} \left\{ \sqrt{\frac{(x_{c_b} - x_{c_a})^2 + 4 \cdot E_w}{4}} - \frac{(x_{c_b} - x_{c_a})}{2} \right\} V \quad (3)$$

State-space models derived from the above equations are,

$$A = \begin{pmatrix} \frac{-(Fw_{as} + Fw_{bs})}{C} & 0 \\ 0 & \frac{-(Fw_{as} + Fw_{bs})}{C} \end{pmatrix} \quad (4)$$

$$B_{acidic} = \begin{pmatrix} \frac{-1}{C} \\ \frac{Co_{bs}}{C} \end{pmatrix} \quad (5)$$

$$B_{basic} = \begin{pmatrix} \frac{Co_{as}}{C} \\ \frac{-1}{C} \end{pmatrix} \quad (6)$$

$$C = (3 - (8 \cdot 10^{-pH_s}) \quad 3 + (8 \cdot 10^{-pH_s})) \quad (7)$$

$$D = 0 \quad (8)$$

The various parameters of the process were assumed, as stated in the following Table 1,

Depending on the required pH value from the process, either the acid solution or alkaline solution was added to the process tank. For instance, if the process solution is acidic, an

alkaline solution was added to neutralize it and vice versa. State-space models for both the cases, stated in Eqs. 4 to 8, were derived from Eqs. 1 to 3. Then the various operating points obtained from the open-loop I/O characteristics (Table 2) were substituted to compute the state-space model and the corresponding transfer function, as shown in Table 3. Open-loop I/O characteristics from the process (Figure 2) were noted by varying one manipulated variable and maintaining the other variable constant. Initially, the inflow rate of the acidic solution was maintained at a constant value ($FW_a = 7.2$ L/h). Then, for various inflow rates of the alkaline solution (FW_b), the pH value of the solution in the process tank was recorded. Similarly, FW_b was maintained constant, and FW_a was varied, and the corresponding pH value was recorded.

Table 1. Operating data for the pH process.

Parameter	Description	Value
R	The radius of the process tank	8 cm
H	The maximum height of the process tank	20 cm
C	The capacity of the process tank	4.02 L
Co_a, Co_b	The concentration of acid and alkaline solution	0.1 N
FW_a, FW_b	The flow rate of acid and alkaline solution to the process tank	12 L/h
E_w	The equilibrium constant for water dissociation	10^{-14}
XC_a, XC_b	The concentration of the non-reacting acid solution and alkaline solution in tanks	

Table 2. Operating points obtained from I/O characteristics.

Case	Operating Points
Case 1 FW_b Vs. pH	$FW_{as} = 7.2$ L/h $FW_{bs} = 7.2$ L/h $C = 4.02$ L $Co_{bs} = 1.23 \times 10^{-10}$ mol/L $pH_s = 9.91$
Case 2 FW_a Vs. pH	$FW_{as} = 7.2$ L/h $FW_{bs} = 7.2$ L/h $C = 4.02$ L $Co_{as} = 1.7 \times 10^{-10}$ mol/L $pH_s = 9.77$

From Table 3, it can be seen that, mathematically, the state space and transfer function model of both the cases are similar. Therefore, either case can represent the pH neutralization system in simulation.

Table 3. State-space and transfer function model of the system.

Case	State Space Model	Transfer Function
FW_b Vs. pH	$A = \begin{bmatrix} -3.581 & 0 \\ 0 & -3.581 \end{bmatrix}$ $B = \begin{bmatrix} -0.2487 \\ 3.059e - 11 \end{bmatrix}$ $C = [3 \quad 3]$ $D = 0$	$\frac{-0.7461s - 2.672}{s^2 + 7.162s + 12.82}$
FW_a Vs. pH	$A = \begin{bmatrix} -3.581 & 0 \\ 0 & -3.581 \end{bmatrix}$ $B = \begin{bmatrix} 4.228e - 11 \\ -0.2487 \end{bmatrix}$ $C = [3 \quad 3]$ $D = 0$	$\frac{-0.7461s - 2.672}{s^2 + 7.162s + 12.82}$

3.3. Stability analysis

Eigenvalues and frequency response of the system were used to analyze the system's stability. The system was completely stable since all its eigenvalues were negative, and the phase margin and the gain margin were positive values. Furthermore, the phase margin was higher than the gain margin, as stated in Table 4.

Table 4. Stability analysis of pH system using eigenvalues and Bode plot.

Eigenvalue	Bode Plot Analysis
0.0920	Gain Margin = 4.8000
-0.2522	Phase Margin = Inf

4. Controller tuning methods

For the proposed system, the PID controller was implemented to maintain the pH at a desired acidic, alkaline, or neutral value. Three tuning algorithms, a classic optimization algorithm, genetic algorithm (GA), a metaheuristic optimization algorithm, Bacterial Foraging Technique based Particle Swarm Optimization algorithm (BFTPSO), and a robust technique, quantitative feedback theory (QFT), were chosen to tune the PID controller for this nonlinear process.

4.1. Genetic algorithm

A population of solutions is continuously evolved in a genetic algorithm by modifying the bits of the solution based on a fitness function to obtain the optimal output from the optimization problem. The three most common operators for evolving the population of solutions are Selection, Crossover, and Mutation.

A standard genetic algorithm involves the following steps,

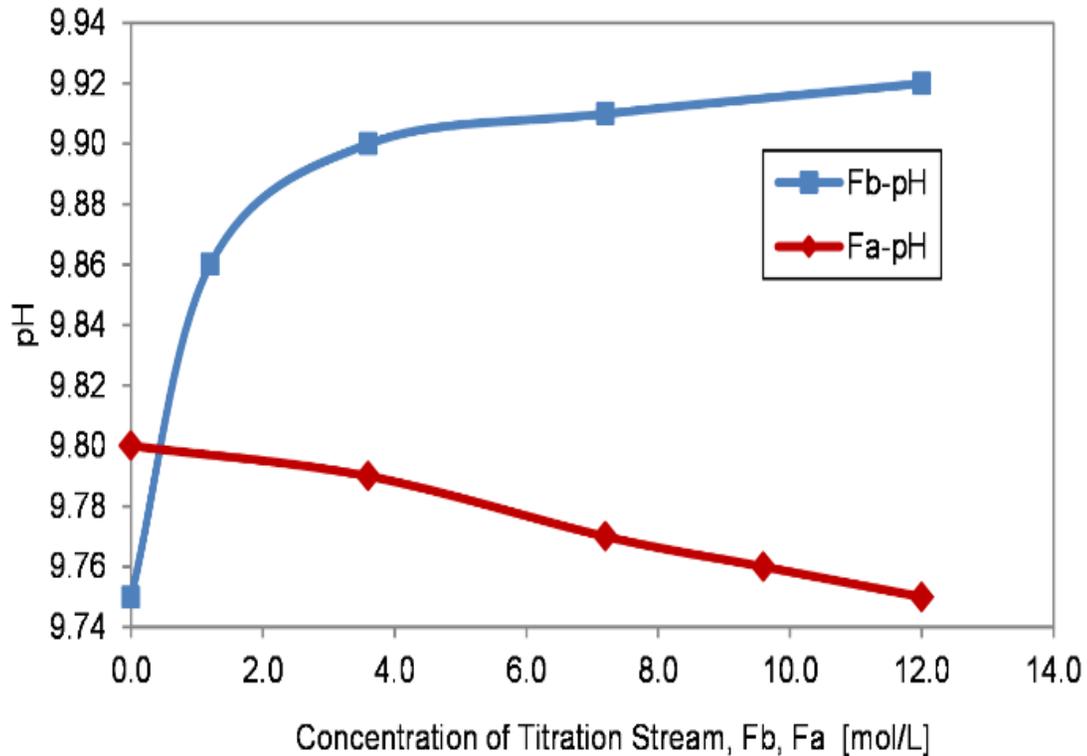


Figure 2. Process open-loop I/O characteristics.

Step 1–Generation of an initial population of ‘n’ chromosomes.

Step 2–Calculation of the fitness function $f(x)$ for each chromosome (x) in the population.

Step 3–Repetition of the following steps ‘i’, ‘j’, and ‘k’ until ‘n’ offspring are generated from ‘n’ chromosomes,

- i) Creation of a fitting parent population depending on higher/lower fitness function value. Usually, two-parent chromosomes are selected to apply other GA operators to it. However, in elitism selection, one best parent chromosome is kept until the next generation. Also, a parent chromosome may not be selected at all, or it may be selected more than once based on its fitness value.
- j) Generation of several offspring at randomly chosen points of the parent population based on the crossover rate. Crossover facilitates the offspring to inherit unique features and the common features of the parent chromosomes.
- k) Mutation of the offspring at randomly chosen points depending on the mutation probability. If the mutation probability is greater than the mutated chromosome, then the mutated is used in the new population. Else it is discarded.

Step 4–Replacement of the existing population with the newly created population.

Step 5–Repetition from Step 2 again.

In this work, Integral Time Weighted Absolute Error (ITAE) was regarded as the optimization function for the algorithm, and the controller parameters were computed for the most minimal ITAE value. One iteration completed one generation of the algorithm, and optimization was carried out for several generations until suitable controller gains that reduced the error in the system were attained. Sensitivity analysis was done for different population sizes, crossover, and mutation probabilities. The best results for this system were obtained in 300 iterations for a single-point crossover method with a probability of 0.2 and with a uniform mutation probability of 0.05. An initial population size of 100 and a combination of elitism and stochastic uniform selection methods were employed to implement the algorithm.

4.2. Bacterial foraging technique-based particle swarm optimization algorithm

Bacterial Foraging Technique (BFT) and Particle Swarm Optimization Algorithm (PSO) are powerful optimization techniques. Bacterial Foraging Technique-based Particle

Swarm Optimization Algorithm (BFTPSO) employs both BFT and PSO algorithms. It minimizes looping in local minima solutions, and the random velocity of a particle in BFT is improved by incorporating the PSO algorithm into it (Hassan et al., 2020).

A standard BFTPSO algorithm involves the following steps,

Step 1–Initialization of random locations for the bacteria.

Step 2–Assignment of random global best location and fitness value for each bacterium. The assumed fitness value should usually be significant.

Step 3–Initialization of all the variables to be used in a BFT-PSO algorithm, namely, bacteria population size, the maximum number of steps, width and height of the repellent, width and depth of the attractant, number of chemotactic, reproduction, and elimination-dispersal steps, probability of elimination-dispersal.

Usually, the number of chemotactic steps should be greater than the number of reproduction steps, which should be greater than the number of elimination-dispersal steps.

Step 4–Implementation of chemotaxis activity by repeating the following steps 'i', 'j,' and 'k' for each bacterium in the search space until the maximum number of chemotactic steps is reached,

i) Updation of the velocity and position of bacterium based on the PSO equation.

j) Computation of the fitness function and the cell-to-cell attraction of each bacterium.

k) Implementation of swimming activity until the maximum number of swimming steps is reached.

a) For each swimming step, if,
(computed fitness function < initial random fitness function)

&

(current swimming step < maximum number of swimming steps), then,

Initial fitness function = computed fitness function

Current bacteria position = updated bacteria position

Swim step count++

b) Else, if,

(current fitness function < global fitness function), then,

Current bacteria position = global best position

Step 5–Implementation of reproduction activity until the maximum number of reproduction steps is reached by sorting the bacteria in the population based on the increasing fitness function values. The weakest bacteria shall be neglected, and the best bacteria shall be split and placed in the same position.

Step 6–Implementation of elimination-dispersal activity until the maximum number of elimination-dispersal steps is reached,

a) For each bacterium in the search space, if, (randomly generated elimination-dispersal probability < initialized elimination-dispersal probability), then,

Current bacteria position = updated bacteria position

Step 7–Termination of the algorithm.

In this work, system overshoot and ITAE were regarded as the optimization function for the BFTPSO algorithm. The minimum optimization function was determined with an initial bacteria population of 10 and with 20 chemotactic steps. The number of reproduction steps and elimination-dispersal events were assumed to be 4 each, and the probability of elimination/dispersion of bacteria was assumed to be 0.25. Just as in GA, a sensitivity analysis was carried out to fix these initial values. Finally, the algorithm was implemented for 300 iterations to determine the controller gains corresponding to the minimum optimization function.

4.3. Quantitative feedback theory

A vast number of parameters must be controlled in real-time plants. Unfortunately, the relationship between such parameters is often specified by a system model that is not entirely accurate. In order to achieve the desired control, the process dynamics must be precisely understood, which is not always possible. Thus, the system's performance specifications would undoubtedly be far from what is expected because of this misrepresentation of the actual model of the system. As a result, the controller design must be made independent of the process model. Quantitative Feedback Theory (QFT) is a controller tuning technique that facilitates robustness in the performance specifications despite uncertainties in the process model.

A standard QFT algorithm involves the following steps,

Step 1–Initialization of the nominal transfer function and the uncertainty in the model variables.

Step 2–Generation of templates of the plant model.

Step 3–Initialization of the preferred performance specifications.

Step 4–Generation of bounds for the performance specifications.

Step 5–Generation of the controller design.

Step 6–Computation of the controller gain values.

In QFT-based tuning, the templates of the plant model for $\pm 10\%$ of model uncertainties were generated, and the desired performance specifications, viz., robust stability, setpoint tracking, and input disturbance rejection, from the control loop of the plant model were chosen. A margin value higher than one was used, and the bounds of the preferred performance.

specifications were generated at different frequencies to design the controller by adding gain, integrator, and differentiator.

Since the theory behind GA, BFTPSO, and QFT has already been explained by several researchers, it is not replicated in this paper. However, the detailed explanation regarding the PID controller tuning in different applications using these algorithms can also be referred to from some of the author's previous publications (Aparna et al., 2017; Shajahan et al., 2018, 2019).

5. Simulation results

PID controller was tuned using optimization and robust algorithms, GA, BFTPSO, and QFT, to deduce the efficient controller for the proposed nonlinear system. ITAE was preferred as the optimization function over other indices because it provides a realistic estimation of the settling time, overshoots, and undershoots in the process response.

As stated in Table 5, the resulting PID controller gains were used to control the system for neutralization, acidic and alkaline control. Furthermore, the controllers were tested with different pH setpoints and external input disturbance to analyze its servo and regulatory capability.

Since the process gain is negative, the PID controller should bring the positive error value of the process to zero. This is possible by increasing the controller output, which eventually decreases the process output. For this purpose, negative K_p and K_i values were obtained for the controller. Additionally, even though the K_d value is negative from the K_i value, it still yielded a stable controller design.

5.1. Servo action

Variable setpoints were provided to assess the controller's ability to track setpoints. A setpoint of 4 was chosen for acidic control, 7 pH was chosen for neutralization, and a pH value of 9 was chosen for alkaline control.

The corresponding results are depicted in Figures 3 to 5. Figure 3 illustrates the servo response of the system when the controller was tuned using the BFTPSO algorithm. Similarly, Figures 4 and 5 show the servo response of the system when the controller was tuned using the GA and the QFT algorithm, respectively.

5.2. Servo and regulatory action

The controllers' ability to track setpoints while also rejecting external input disturbance was tested. Initially, a step input of 4 and a disturbance of +25% of the set value were applied during acidic control. Similarly, a step input of 7 and a disturbance of +14% of the set value were applied during neutralization. Finally, a step input of 9 and a disturbance of +11% of the set value were applied during alkaline control.

The resulting servo and regulatory response of the controllers are depicted in Figures 6 to 8. Figure 6 illustrates the servo and regulatory response of the system when the controller was tuned using the BFTPSO algorithm. Similarly, Figures 7 and 8 show the servo and regulatory response of the system when the controller was tuned using GA and QFT algorithm, respectively.

Table 5. Controller gains tuned using GA, BFT-PSO, and QFT.

Algorithm	Neutral	Alkaline Control	Acidic Control
GA	$K_p=-5.4976$	$K_p=-5.6272$	$K_p=-1.0514$
	$K_i=-6.5547$	$K_i=-6.3353$	$K_i=-3.9196$
	$K_d=0.1346$	$K_d=4.8895$	$K_d=2.3094$
BFTPSO	$K_p=-2.3118$	$K_p=-2.3118$	$K_p=-2.3118$
	$K_i=-4.7060$	$K_i=-4.7060$	$K_i=-4.7060$
	$K_d=0.3716$	$K_d=0.3716$	$K_d=0.3716$
QFT	$K_p=-0.9538$	$K_p=-0.9538$	$K_p=-0.9538$
	$K_i=-2$	$K_i=-2$	$K_i=-2$
	$K_d=1$	$K_d=1$	$K_d=1$

6. Discussions

The effective controller among the three was deduced by comparing the three controllers' setpoint tracking capability and servo and regulatory action. In addition, the time domain specifications of the servo response in all three control cases were also compared, as shown in Table 6.

It can be seen from Figures 3 to 5 and Figures 6 to 8 that all the controllers could track the setpoints and reject the input disturbance quickly.

However, it can be seen from the results stated in Table 6 that, with neutralization, although the BFTPSO controller provided the quickest control action, it generated a response with large overshoots and undershoots. Similarly, the QFT controller generated no overshoots in the response. However, the settling time was more than its counterparts because of slower control action, and it also generated substantial undershoots in the response. Thus, the GA controller was ideal for neutralization as it generated a quick control action with the fastest settling time, with no overshoots and negligible undershoots and error.

The behavior of the controllers for alkaline and acidic pH control was similar, as with neutralization. The BFTPSO controller generated the fastest control action with large overshoots and undershoots. On the other hand, the QFT controller generated the slowest control action with no overshoots and large undershoots in the response. Meanwhile, the GA controller generated a response with significant undershoots in alkaline control and significant overshoots in acidic control. Nonetheless, the GA controller was preferred over the others, as it produced a quick control action with the fastest settling time and lesser overshoots/undershoots.

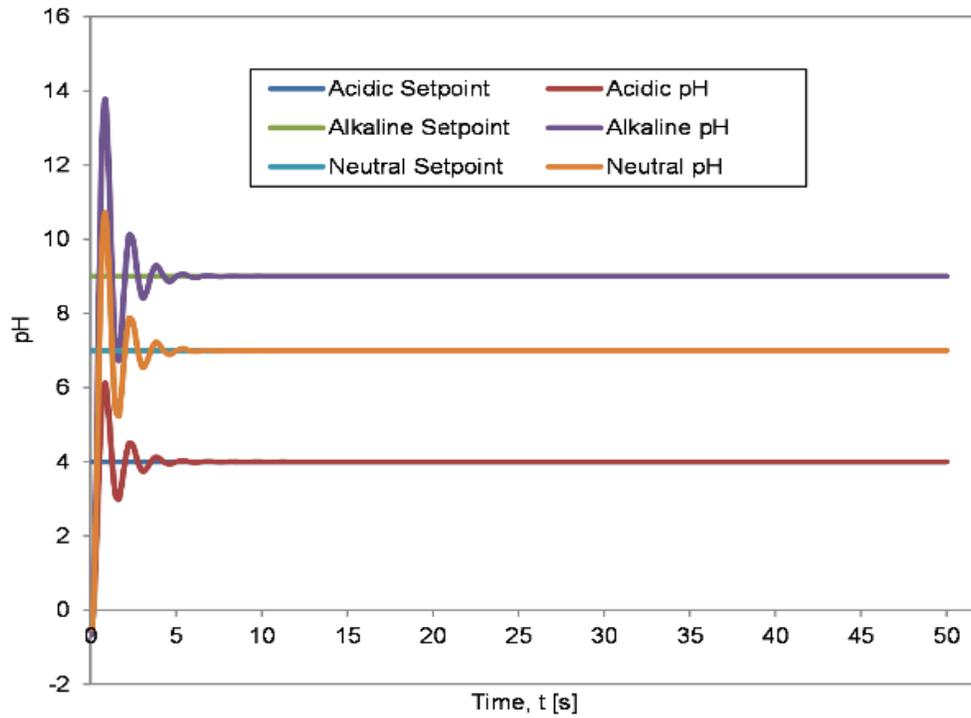


Figure 3. Servo action of BFTPSO algorithm tuned PID.

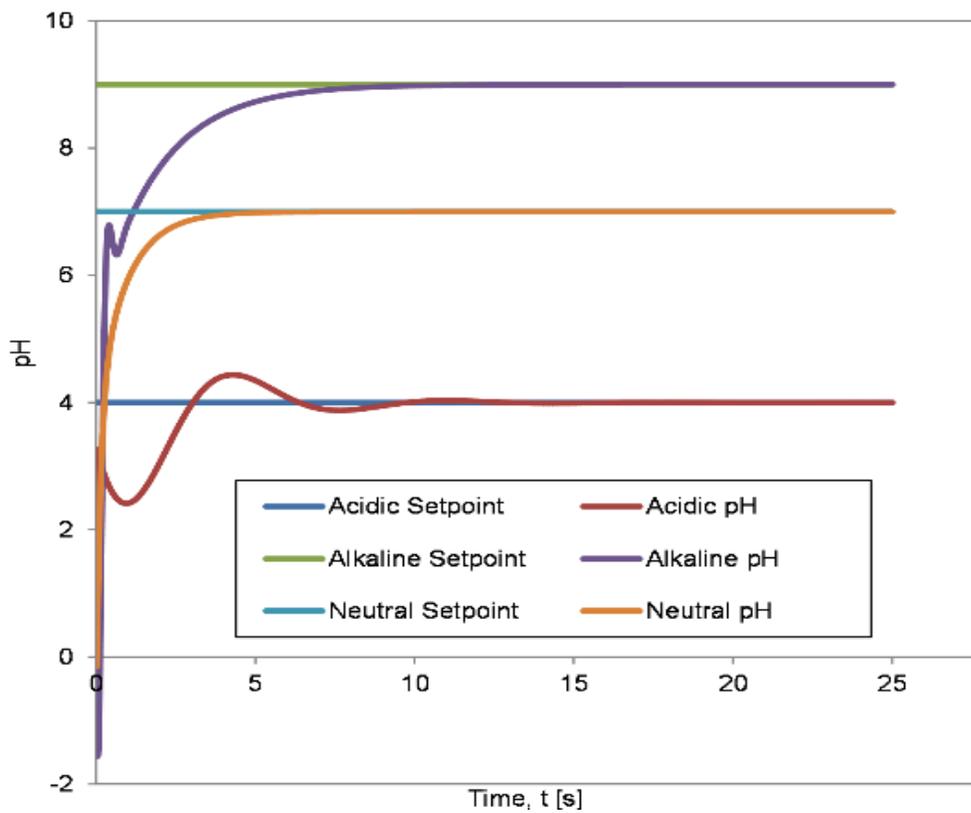


Figure 4. Servo action of genetic algorithm optimized PID.

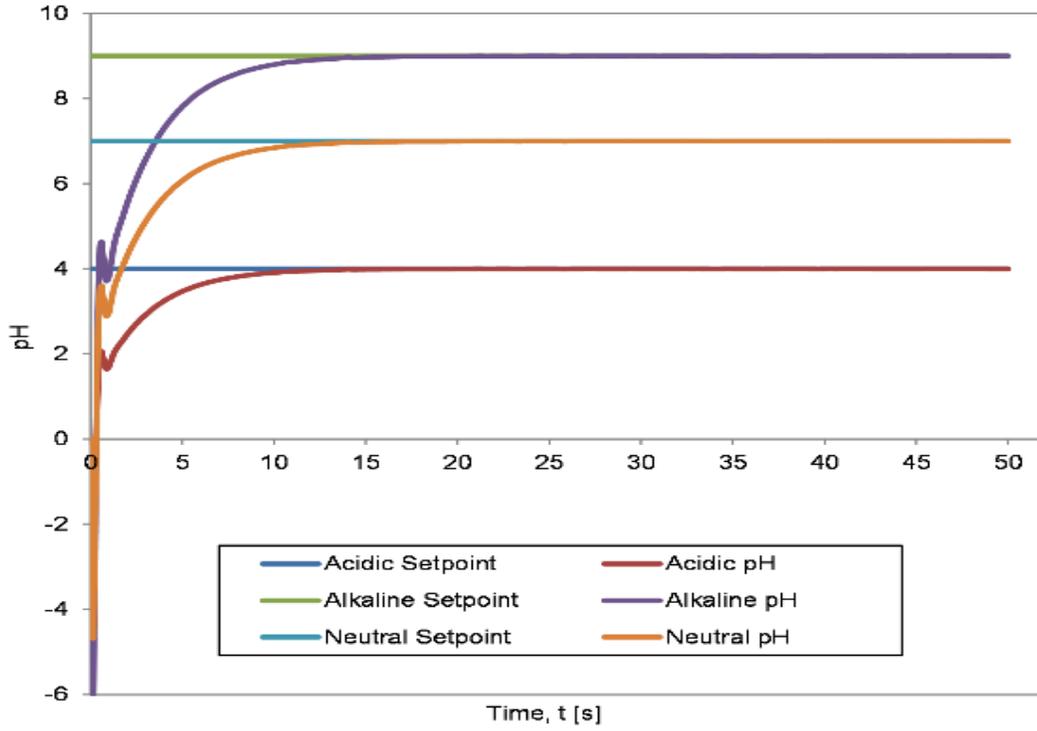


Figure 5. Servo action of Quantitative Feedback Theory tuned PID.

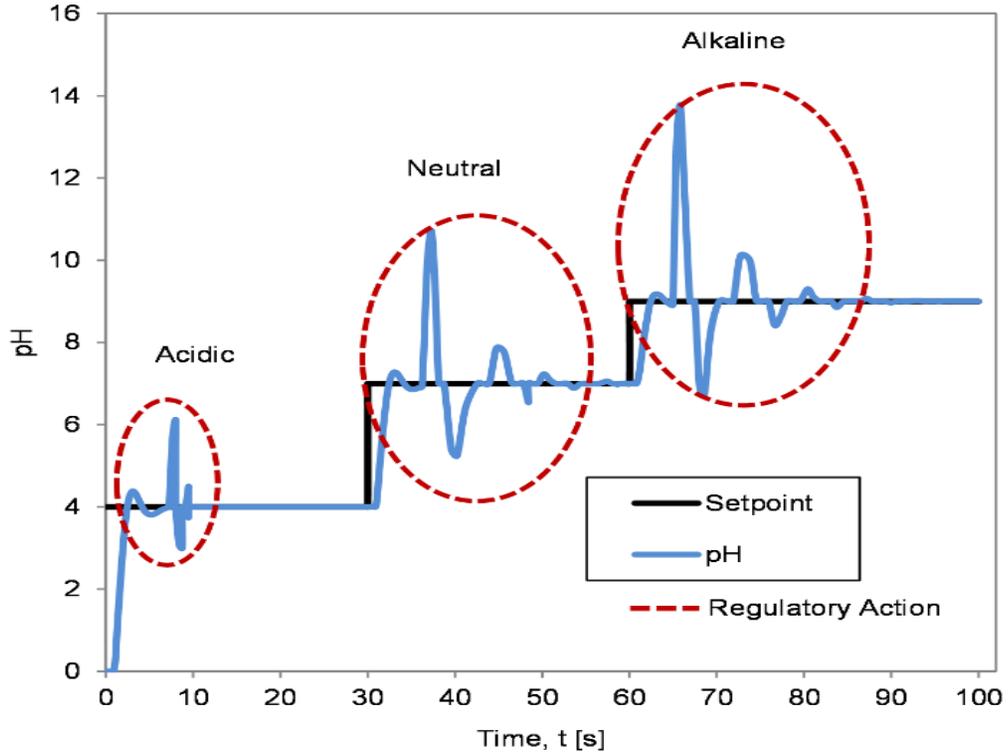


Figure 6. Servo and regulatory action of BFTPSO algorithm tuned PID.

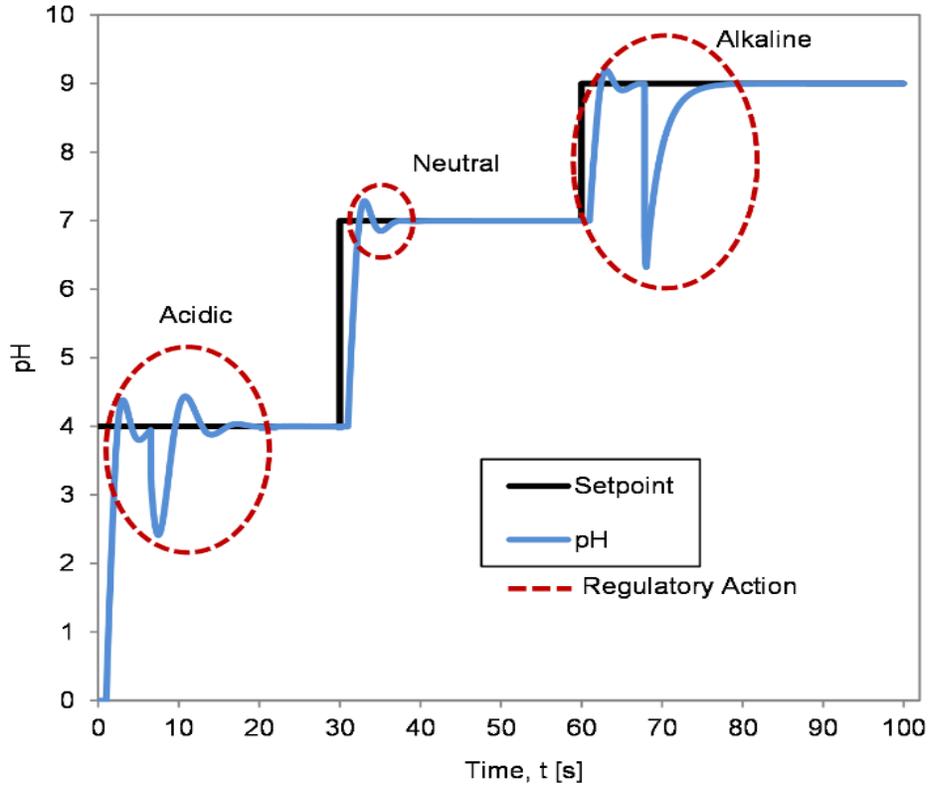


Figure 7. Servo and regulatory action of the genetic algorithm optimized PID.

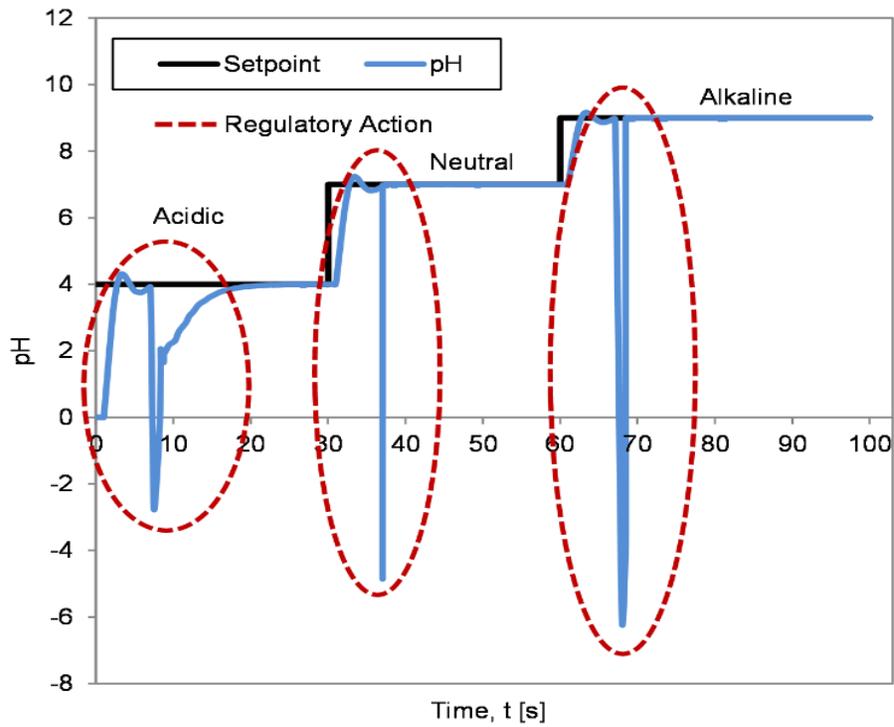


Figure 8. Servo and regulatory of Quantitative Feedback Theory tuned PID.

Thus, the controller tuned using GA produced the desired output response with faster setpoint tracking and fewer overshoots/undershoots for all three types of pH control than the QFT or the BFTPSO algorithm.

Table 6. Step response characteristics for pH control.

pH Control	Characteristics	BFTPSO	GA	QFT
Neutral	ITAE	6.09	10.86	-
	Rise Time	0.25	1.34	5.44
	Settling Time	3.99	2.88	8.91
	Overshoot	53.01	0	0.02
	Undershoot	6.98	2.20	66.65
Alkaline Control	ITAE	7.84	13.55	-
	Rise Time	0.25	2.55	5.44
	Settling Time	3.99	5.47	8.91
	Overshoot	53.01	0	0.02
	Undershoot	6.98	17.29	66.65
Acidic Control	ITAE	3.48	2.80	-
	Rise Time	0.25	2.55	5.44
	Settling Time	6.12	8.60	8.91
	Overshoot	53.01	10.95	0.02
	Undershoot	6.98	0	66.65

7. Real-time implementation

The offline controller gains obtained from the three algorithms, stated in Table 5, were used to implement real-time control of the pH of the reactor tank fluid using NI myRIO-1900 (Figure 9 (a)). NI myRIO-1900 is a reconfigurable input/output embedded device with a Xilinx Zynq-7010 processor comprising a Real-Time (RT) processor and a Field Programmable Gate Array (FPGA) processor for higher resolution, sensitivity and faster response times in real-time applications.

In this system, the pH probe and transmitter were calibrated before use, and the output of the pH transmitter ranging from -420 mV to 420 mV was scaled to 0.5 V to 3 V using a signal conditioning circuit comprising IC CA3140, IC TL081, and IC 555 timer, as shown in Figure 9 (b). Then, the output of the signal conditioning circuit was fed as the input to the RIO, and the output of the RIO was used to manipulate the speed of the pump and thereby the flow rate of the alkaline or acid solution to the mixing tank in the system to maintain the pH value, as explained in Figure 10.

The output of the signal conditioning circuit was fed as the input in any of the referential single-ended or differential-ended analog input channels of the RIO. The corresponding

analog output from RIO was converted to 4-20 mA and fed to the pump to manipulate the speed of the pump and thereby the flow rate of the alkaline or acid solution to the mixing tank in the system. Then the control loop was developed in LabVIEW software. The controller configuration and gains were varied in the Front Panel and Block Diagram (Dixit & Jain, 2016). After changing the controller gains, the corresponding results were recorded.

8. Experimental results and discussions

The controller gains in NI myRIO-1900 were varied based on different algorithms like GA, BFTPSO, and QFT, and the accuracy and effectiveness of these algorithms in controlling the pH of the fluid in the reactor tank was found out by analyzing the time domain specifications of the servo capability of the controllers obtained for two test cases of neutralization,

Case 1 - Neutralization of acidic solution (pH = 5)

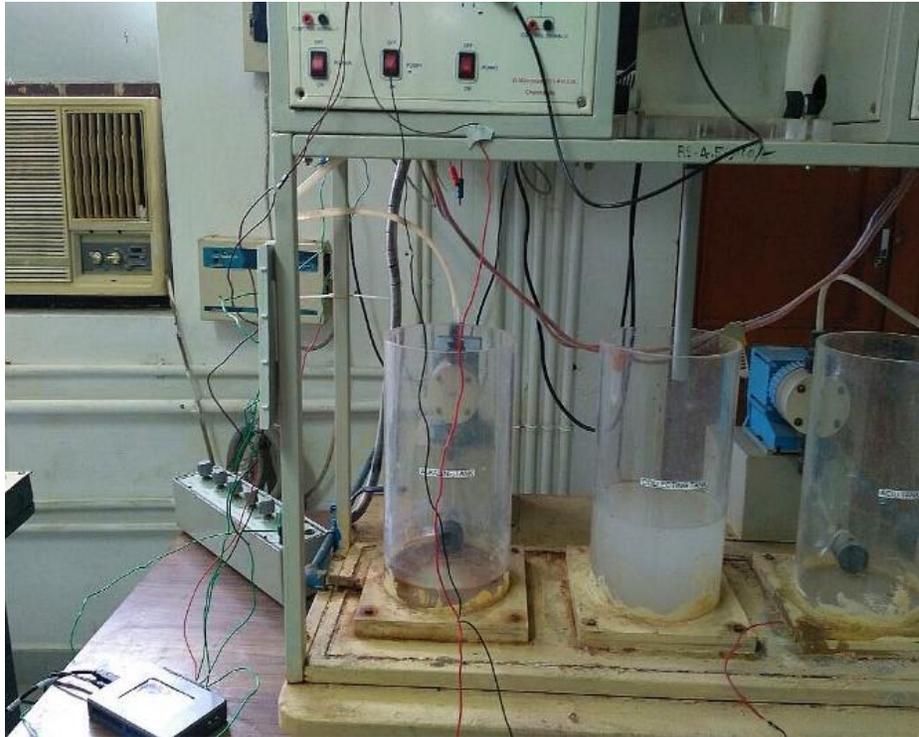
Case 2 - Neutralization of alkaline solution solution (pH = 8)

In both cases, the neutralized solution had a pH value of 6.5. Figures 11 to 13 depict the servo response of the system when the controller was tuned using BFTPSO, GA, and QFT algorithm, respectively.

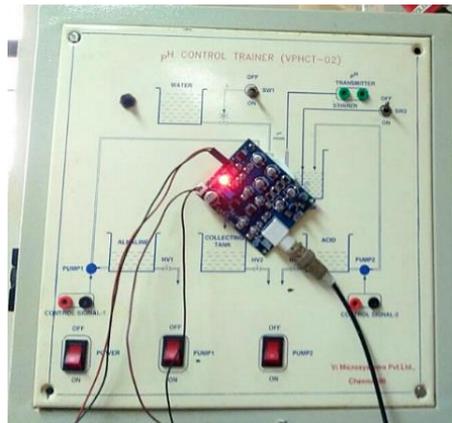
The experimental results indicated in Figures 11 to 13 and Table 7 show that BFTPSO and QFT algorithms generated responses with large settling time. In contrast, GA produced the best response among the three algorithms by generating the desired output response with negligible overshoot, undershoot, and faster setpoint tracking for the neutralization process than QFT or BFTPSO controllers.

Table 7. Step response characteristics for NI myRIO-1900 output response.

Neutralization	Characteristics	BFTPSO	GA	QFT
Acidic To Neutral	Settling Time (minutes)	22	15	22
	Overshoot	0.05	-	-
	Undershoot	-	-	-
Basic to Neutral	Offset	0.17	0.18	0.18
	Settling Time (minutes)	21	12	21
	Overshoot	-	-	-
Neutral	Undershoot	0.017	-	-
	Offset	0.19	0.18	0.13



(a). Laboratory setup of pH neutralization system with NI myRIO-1900.



(b). pH signal conditioning circuit comprising IC CA3140, IC TL081, and IC 555.

Figure 9. Control of pH neutralization system using NI myRIO-1900 and pH signal conditioning circuit.

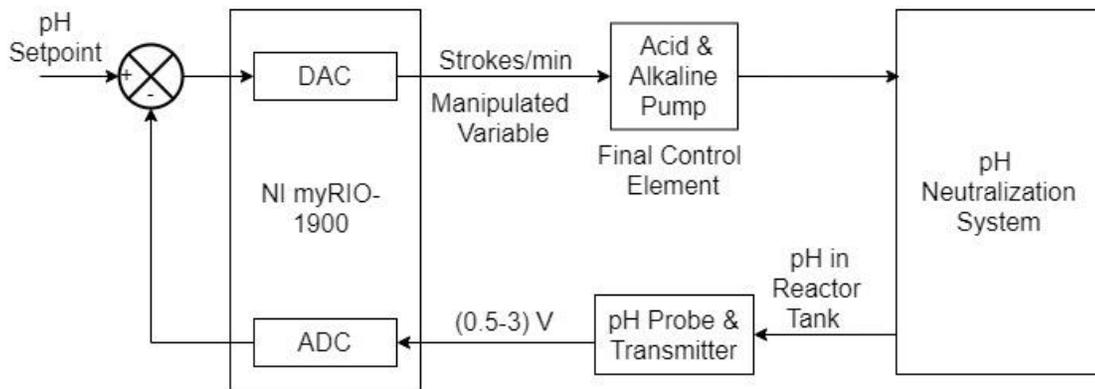


Figure 10. Block diagram for pH Control using NI myRIO-1900.

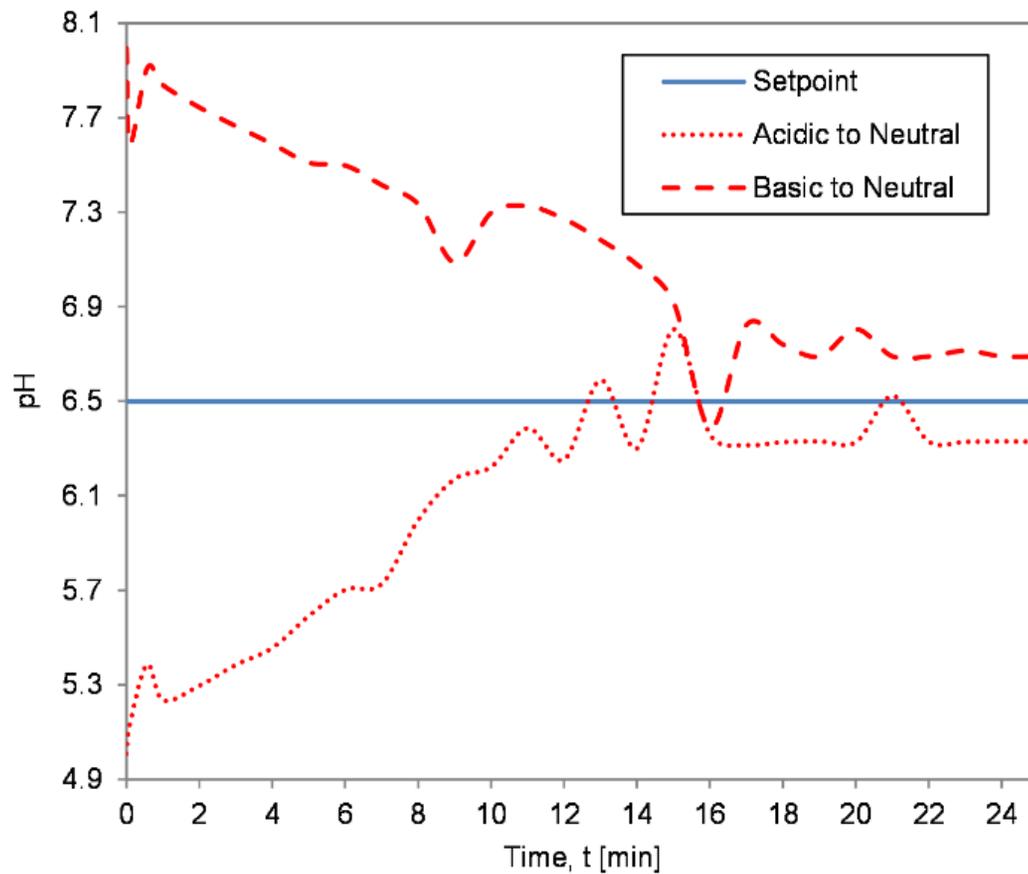


Figure 11. Response of BFTPSO PID controller using NI myRIO-1900.

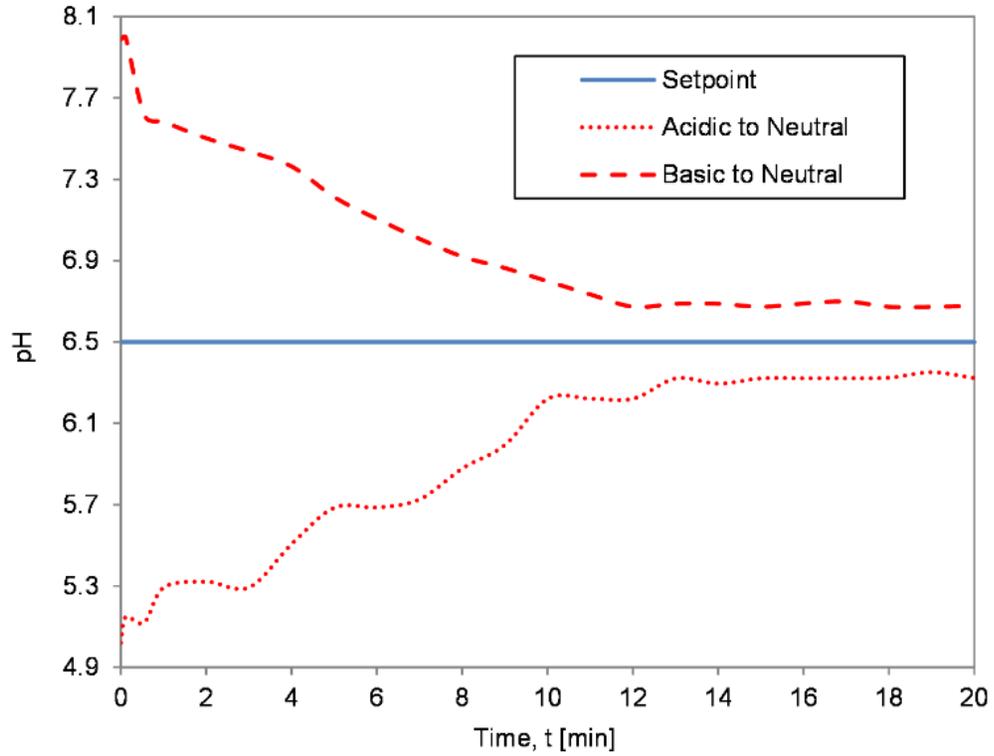


Figure 12. Response of GA PID controller using NI myRIO-1900.

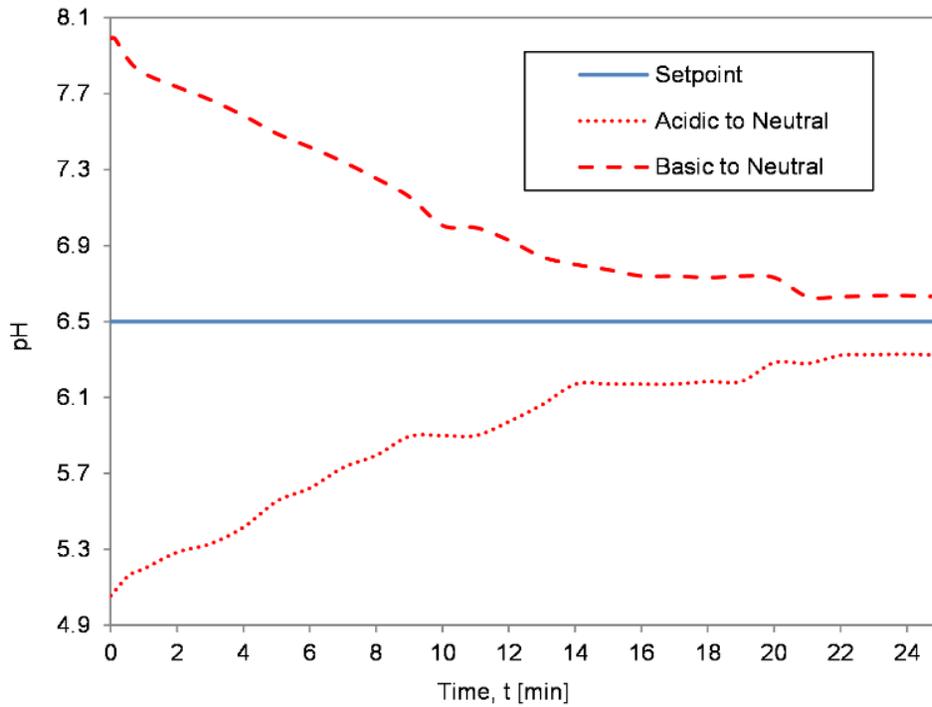


Figure 13. Response of QFT PID controller using NI myRIO-1900.

9. Conclusions

In this project, experimental data was acquired from a highly nonlinear multiple input single output pH neutralization system, and linear system models were developed around an equilibrium point. Popular robust and metaheuristic optimization algorithms, like, Quantitative Feedback Theory, Bacterial Foraging Technique based Particle Swarm Optimization Algorithm, and Genetic Algorithm were chosen to deduce the appropriate values of the gain parameters for implementing a PID controller. These controllers were then implemented in real-time to control the pH of the fluid in the neutralization system using a signal conditioning circuit and reconfigurable I/O device. The step response characteristics of the output response from the three algorithms were compared in simulation and experiment, and the genetic algorithm was found to be a suitable algorithm for the real-time control of this system. Furthermore, the genetic algorithm produced output with quicker settling times and negligible overshoot and undershoot, than the QFT or the BFTPSO algorithm.

Acknowledgments

The author would like to thank the Department of Chemistry for their technical assistance.

References

- Abdullah, N. H. S., Karsiti, M. N., & Ibrahim, R. (2012). A review of pH neutralization process control. *2012 4th International Conference on Intelligent and Advanced Systems (ICIAS2012)*, 2, 594–598. <https://doi.org/10.1109/ICIAS.2012.6306084>
- Ali, E. (2001). pH Control Using PI Control Algorithms with Automatic Tuning Method. *Chemical Engineering Research and Design*, 79(5), 611–620. <https://doi.org/10.1205/02638760152424398>
- Alina, B., & Mădălina, C. (June). Internal model control for wastewater ph neutralization process. In *2016 8th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)* (pp. 1-6). IEEE. <https://doi.org/10.1109/ECAI.2016.7861072>
- Aparna, V. (2014). Development of automated pH monitoring & control system through USB Data Acquisition. *2014 6th IEEE Power India International Conference (PIICON)*, 1–6. <https://doi.org/10.1109/34084POWERI.2014.7117602>
- Aparna, V., Banu, U. S., & Yasmin, S. (2017). PI controller design using quantitative feedback theory for a flow process. *2017 IEEE International Conference on Technological Innovations in Communication, Control and Automation (TICCA)*, 2018, 31–35. <https://doi.org/10.1109/TICCA.2017.8344596>
- Barve, J. J., & Nataraj, P. S. V. (1999). Synthesis of a robust pH controller using QFT. *Proceedings of IEEE TENCON '98. IEEE Region 10 International Conference on Global Connectivity in Energy, Computer, Communication and Control (Cat. No.98CH36229)*, 1, 214–217. <https://doi.org/10.1109/TENCON.1998.797124>
- Bequette, B. W. (1991). Nonlinear control of chemical processes: a review. *Industrial & Engineering Chemistry Research*, 30(7), 1391–1413. <https://doi.org/10.1021/ie00055a001>
- Bharathi, N., Shanmugam, J., & Rangaswamy, T. R. (2007). *Intelligent Control of pH in a Neutralization Process. Asian Journal of Information Technology*, 6(6), 667–673.
- Biagiola, S. I., Agamennoni, O. E., & Figueroa, J. L. (2016). Robust control of wiener systems: application to a ph neutralization process. *Brazilian Journal of Chemical Engineering*, 33(1), 145–153. <https://doi.org/10.1590/0104-6632.20160331s00002846>
- Bingi, K., Ibrahim, R., Karsiti, M. N., Chung, T. D., & Hassan, S. M. (2016). Optimal PID control of pH neutralization plant. *2016 2nd IEEE International Symposium on Robotics and Manufacturing Automation (ROMA)*, 1–6. <https://doi.org/10.1109/ROMA.2016.7847812>
- Bingi, K., Ibrahim, R., Karsiti, M. N., & Hassan, S. M. (2018). Fractional Order Set-point Weighted PID Controller for pH Neutralization Process Using Accelerated PSO Algorithm. *Arabian Journal for Science and Engineering*, 43(6), 2687–2701. <https://doi.org/10.1007/s13369-017-2740-7>
- Bingi, K., Ibrahim, R., Karsiti, M. N., Hassan, S. M., & Harindran, V. R. (2020). *Fractional-order Systems and PID Controllers*. Springer International Publishing. https://doi.org/10.1007/978-3-030-33934-0_2
- Darab, C., Hodrea, R., Crisan, R., & Nascu, I. (2012). Modeling and internal model control strategy of pH neutralization process. *2012 20th Telecommunications Forum (TELFOR)*, 1579–1582. <https://doi.org/10.1109/TELFOR.2012.6419524>

- Dixit, S. A., & Jain, A. (2016). Implementation of PPC-SSR as final control element and interfacing of PLC with LabVIEW using Modbus in two tank non interacting level control system. *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, 1–6. <https://doi.org/10.1109/ICPEICES.2016.7853095>
- Elameen, M. H. E., Karsiti, M. N., & Ibrahim, R. (2014). Nonlinear model feedback linearization control strategy of a pH neutralization process. *2014 5th International Conference on Intelligent and Advanced Systems (ICIAS), 1*, 1–5. <https://doi.org/10.1109/ICIAS.2014.6869488>
- Faanes, A., & Skogestad, S. (2004). pH-neutralization: integrated process and control design. *Computers & Chemical Engineering*, 28(8), 1475–1487. <https://doi.org/10.1016/j.compchemeng.2003.11.001>
- Hassan, S. M., Ibrahim, R., Saad, N., Bingi, K., & Asirvadam, V. S. (2020). Hybrid PID Based Predictive Control Strategies for WirelessHART Networked Control Systems (Vol. 293). Springer Nature. https://doi.org/10.1007/978-3-030-47737-0_5
- Ibrahim, R., & Murray-Smith, D. J. (2007). Design, implementation and performance evaluation of a fuzzy control system for a pH neutralisation process pilot plant. *2007 International Conference on Intelligent and Advanced Systems*, 1001–1006. <https://doi.org/10.1109/ICIAS.2007.4658536>
- Ibrahim, R. (2008). *Practical modelling and control implementation studies on a pH neutralization process pilot plant* (Doctoral dissertation, University of Glasgow).
- Moore, R. L. (1995). *Environmental Protection by the Neutralization of Wastewater Using PH Control*. Isa.
- Shajahan, M. M., Jamal, D. N., & Aparna, V. (2019). Controller design using quantitative feedback theory for thermal power plant process *Case studies in thermal engineering*, 14, 100441. <https://doi.org/10.1016/j.csite.2019.100441>
- Shajahan, M. M., Jamal, D. N., Aparna, V., & Khan, M. A. (2018). Control of electric power generation of thermal power plant in TamilNadu. *Case studies in thermal engineering*, 12, 728-735. <https://doi.org/10.1016/j.csite.2018.08.008>
- Rodriguez, J. L., & Loparo, K. A. (2004). Modeling and identification of pH processes. *Proceedings of the 2004 American Control Conference*, 6, 5483–5488. <https://doi.org/10.1109/ACC.2004.249190>
- Saji, K. S., & Sasi Kumar, M. (2011). A novel controller tuning of pH neutralization process. *2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies*, 712–717. <https://doi.org/10.1109/ICSCCN.2011.6024643>
- Shabani, R., Sedigh, A. K., & Salahshoor, K. (2010). Robust control of a pH neutralization process plant using QFT. *ICCAS 2010*, 497–500. <https://doi.org/10.1109/ICCAS.2010.5669969>
- Shinsky, F. G. (1973). *pH and pION Control in Process and Waste Streams (Environmental Science & Technology)*. Wiley.
- Tadeo, F., Lopez, O. P., & Alvarez, T. (2000). Control of neutralization processes by robust loop shaping. *IEEE Transactions on Control Systems Technology*, 8(2), 236–246. <https://doi.org/10.1109/87.826795>