

# Optimized and Self-Organized Fuzzy Logic Controller for pH Neutralization Process

**Parikshit Kishor Singh, Surekha Bhanot**

Dept. of Electronics & Instrumentation, BITS Pilani, Pilani Campus, India

*E-mail: parikshit\_singh@pilani.bits-pilani.ac.in, surekha@pilani.bits-pilani.ac.in*

**Hare Krishna Mohanta**

Dept. of Chemical Engineering, BITS Pilani, Pilani Campus, India

*E-mail: harekrishna@pilani.bits-pilani.ac.in*

**Abstract**— To conform to strict environmental safety regulations, pH control is used in many industrial applications. For this purpose modern process industries are increasingly relying on intelligent and adaptive control strategies. On one hand intelligent control strategies try to imitate human way of thinking and decision making using artificial intelligence (AI) based techniques such as fuzzy logic whereas on the other hand adaptive mechanism ensures adjusting of the controller parameters. A self-organized fuzzy logic controller (SOFLC) is intelligent in nature and adapts its performance to meet the figure of merit. This paper presents an optimized SOFLC for pH control using performance correction table. The fuzzy adaptation mechanism basically involves a penalty for the output membership functions if the controller performance is poor. The evolutionary genetic algorithm (GA) is used for optimization of input-output scaling factors of the conventional fuzzy logic controller (FLC) as well as elements of the fuzzy performance correction table. The resulting optimized SOFLC is compared with optimized FLC for servo and regulatory control. Comparison indicate superior performance of SOFLC over FLC in terms of much reduced integral of squared error (ISE), maximum overshoot and undershoot, and increased speed of response.

**Index Terms**— pH; Neutralization Process; Intelligent Control; Fuzzy; Self-Organizing; Adaptive; Optimization; Genetic Algorithm

## I. Introduction

pH control plays an important role in many industrial process applications such as wastewater treatment in paper and pulp plants, boiler feedwater treatment in thermal power plants, biopharmaceutical plants, food processing plants, and various other chemical processing plants. However, highly nonlinear titration curve with extreme sensitivity near to and extreme insensitivity away from the neutral pH of 7, and time

varying parameters makes control of pH a very challenging problem. For these reasons, pH control is often regarded as a benchmark for modeling and control of highly nonlinear processes. In recent years the industrial application of advanced pH control techniques for the process industries has become more demanding, mainly due to the increasing complexity of the processes as well as due to stringent requirements of product quality and conformance with strict environmental and safety regulations. Therefore the process industries require more reliable, accurate, robust, and efficient pH control systems for the operation of process plants. In order to fulfill the above requirements there is a continuing need for research to develop advanced pH control schemes, many of which depend on accurate process model. The new trend in terms of advanced control technology is increasingly towards the use of a control approach known as an “intelligent” control strategy. Intelligent control can be described as a control approach or solution that tries to imitate important characteristics of the human brain and way of thinking and decision making. It is also a term that is commonly used to describe control schemes that are based on AI techniques such as neural networks, fuzzy logic, GAs, and their combinations.

The problem of pH control of neutralization process can be divided into two parts. The first part of the problem being the dynamic modeling of the pH neutralization process, whereas the second part being the design of appropriate control schemes based on the process model. Early works on pH control were based on development of the dynamic behavior of the pH process using the neutralization curve and its subsequent use in feedback, feedforward and ratio based adaptive control techniques [1]. The neutralization curves however are static in nature and subjected to variations if acid-base concentrations vary or buffering occurs. To overcome this difficulty the pH process is dynamically characterized as a first-order process with varying gain with or without dead time [2], [3]. Although time varying gain gave a better modeling alternative than static model, this model too was difficult to adopt, particularly in case of random

variations in various process inputs. A more fundamental and rigorous approach for the development of the dynamic process model involves material balance on selective ions, equilibrium constants and electroneutrality equation [4], [5]. The associated model has been used by many researchers as a platform for subsequent investigations and forms the basis to introduce new and improved forms of dynamic modeling and adaptive control techniques using the concept of reaction invariant and strong acid equivalent [6], [7], [8], [9]. In addition to the first-principle based dynamic models, the other approaches have used the development of black-box type data-driven dynamic model based on AI techniques [10], [11], [12]. The main purpose in using AI techniques is to handle the inherent nonlinearity and time-varying nature of the pH neutralization process without the complex mathematical model, thereby improving its performance.

The advent of fuzzy set theory brought a phenomenal change in analysis of complex systems, decision processes and computational intelligence [13], [14]. The most visible paradigm shift was use of natural linguistic variables in place of numerical variables which resulted in "from computing with numbers to computing with words" [15]. The ascent was however not without resistance. During its nascent stage the term "fuzzy logic" gave an impression of vague logic and faced tremendous criticism from few leading scientific intellectuals. In reality, the term "fuzzy logic" refers to the fact that the logic involved can deal with lexical definition of inputs, in contrast to binary logic which accepts only "true" or "false". Therefore, the term "fuzzy" in fuzzy logic applies to the imprecision in the data and not in the logic [16]. The fuzzy set theory laid the foundation of fuzzy control. The major boost to industrial application of fuzzy logic came with implementation of Mamdani fuzzy inference system based intelligent controller which provided a coherent strategy to combine linguistic control rules based on fuzzy theory [17]. Further to make the intelligent controller adaptive in nature, the self-organizing control strategies are applied [18], [19]. The extensive application of fuzzy logic to conventional control techniques such as proportional-integral-derivative control, sliding mode control, and adaptive control, resulted in improved performance for the hybrid controller over their conventional counterparts [20], [21], [22]. Apart from control applications, fuzzy logic has also been extensively used in system modeling identification [23], [24].

To optimize fuzzy inference system, various techniques are used. One among them is self-organizing technique in which the basic idea is to replace the poor performing rule with a better performing rule. Other techniques include altering fuzzy input-output scaling factors and membership functions with the help of evolutionary algorithms such as GA, a search techniques based on the mechanism of natural selection and natural genetics [25]. GAs perform evolutionary

search within a defined search space as per "survival of fittest" principle proposed by Charles Darwin [26]. Over a number of years GAs have been extensively used in various optimization problems that are not well suited for standard optimization algorithms [27], [28], [29].

This research article is organized as follows. Section II describes the dynamic modeling of pH neutralization process. Section III introduces the design perspective of FLC. Section IV shows implementation of SOFLC. Section V gives methodology for genetic optimization of fuzzy inference system (FIS). Section VI details result and discussion on performance of FLC and SOFLC. Section VII presents the conclusion.

## II. Dynamic Modeling of pH Neutralization Process

The pH neutralization process takes place in continuous stirred tank reactor (CSTR) with perfect mixing and constant volume. The CSTR has two influent streams: the hydrochloric acid as titration stream (feed A) and the sodium hydroxide as process stream (feed B), and one effluent stream. The flow characteristics of pump A and pump B are assumed to be linear and identical.

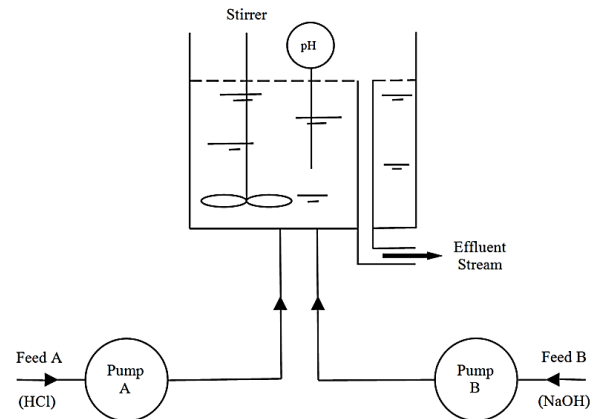


Fig. 1: pH neutralization process

The dynamic model of pH neutralization process involves material balances on selective ions, equilibrium relationship, and electroneutrality equation. Based on principle of material balances, the process mixing dynamics may be described as follows:

$$V \frac{dx_a}{dt} = F_a C_a - (F_a + F_b) x_a \quad (1)$$

$$V \frac{dx_b}{dt} = F_b C_b - (F_a + F_b) x_b \quad (2)$$

where  $V$  is the maximum volume of the CSTR (1.9 L);  $C_a$  is the concentration (0.05 mol/L) and  $F_a$  is the flow rate (0 to 6.23 mL/s i.e. 0 to 100%) of titration stream A;

$C_b$  is the concentration (0.05 mol/L) and  $F_b$  is the flow rate (0 to 6.23 mL/s i.e. 0 to 100%) of process stream B;  $(F_a + F_b)$  is the flow rate of the effluent stream;  $x_a$  is the concentration of acid component (chloride ion,  $Cl^-$ ) in the effluent stream (in mol/L);  $x_b$  is the concentration of base component (sodium ion,  $Na^+$ ) in the effluent stream (in mol/L).

The equilibrium relationship for water is given as

$$K_w = [H^+] [OH^-] \tag{3}$$

where  $K_w$  is the dissociation constant of water ( $10^{-14}$ ).

From the electroneutrality condition, we have

$$[Na^+] + [H^+] = [Cl^-] + [OH^-] \tag{4}$$

All of the  $Cl^-$  ion comes from the HCl and all of the  $Na^+$  ion comes from the NaOH. Using (3) and (4), we have

$$[H^+]^2 - (x_a - x_b) [H^+] - K_w = 0 \tag{5}$$

From the definition of  $pH = -\log_{10}[H^+]$ , the pH titration curve for a strong acid-strong base is given by

$$pH = -\log_{10} \left( \frac{x}{2} + \sqrt{\frac{x^2}{4} + K_w} \right) \tag{6}$$

where

$$x = (x_a - x_b) \tag{7}$$

The block diagram implementation of dynamic pH neutralization process model is shown in Fig. 2.

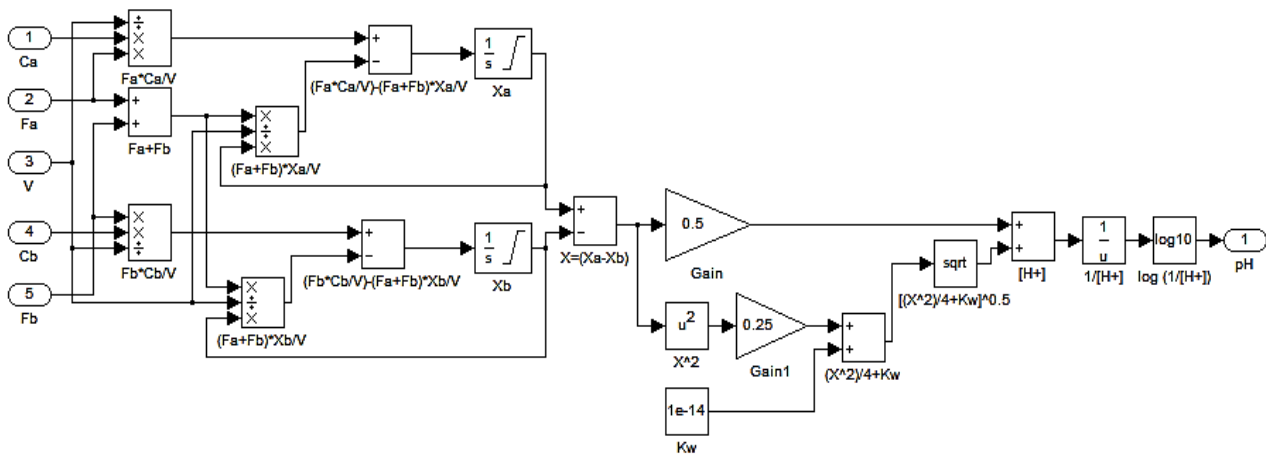


Fig. 2: Dynamic pH neutralization process model for strong acid-strong base

### III. Design of FLC

The double input-single output FLC structure is based on computationally efficient zero-order Sugeno FIS [30]. The FIS consists of an input fuzzification stage, a fuzzy rule processing stage, and an output defuzzification stage. The input fuzzifier stage first determines the belonging of input to membership functions and its membership degree (a number between 0 and 1) to them. Since the antecedent of rule has more than one part, the AND fuzzy algebraic product operator is applied to obtain the firing strength for that rule. The output level of each rule is weighted by the above firing strength. The final defuzzified output of the FIS is the weighted average of all rule outputs.

The input variables used for the Sugeno FIS are error  $e(k) = pH_{SP} - pH(k)$  i.e. the difference between the desired setpoint ( $pH_{SP}$ ) and measured value of controlled variable  $pH(k)$ , and change in error  $ce(k) = e(k) - e(k-1) = pH(k-1) - pH(k)$  i.e. the difference between the error at the present instant  $k$  and

previous instant ( $k-1$ ). The output variable for Sugeno FIS is  $co(k)$ . Also another output variable  $co1(k) = F_a(k-1) - F_a(k)$  is used as minus of the change in manipulated variable i.e. minus of the change in acid flow rate of feed A. The relation between  $co(k)$  and  $co1(k)$  is shown in Fig. 6. The input variables are expressed in pH and output variables are expressed in %.

The linguistic variables for Sugeno FIS inputs ( $x_1 = e$  and  $x_2 = ce$ ) and output ( $y = co$ ) are negative large (NL), negative medium (NM), negative small (NS), zero (ZE), positive small (PS), positive medium (PM), and positive large (PL). The definitions of the membership functions for input fuzzy sets are given in (8), (9), (10), (11), (12) and (13) where  $i = 1, 2$ , and  $m, n = 1, 2, 3, 4, 5, 6, 7$  represents NL, NM, NS, ZE, PS, PM, PL respectively. Similarly the definition of the membership functions for output fuzzy set are given in (14) and (15) where  $q = 1, 2, 3, 4, 5, 6, 7$  represents NL, NM, NS, ZE, PS, PM, PL respectively. The scaling factors  $k_1, k_2, k_3$  are real numbers. The universe of discourse for input fuzzy set is  $x_i \in [-6, 6]$  and output fuzzy set is  $y \in [0, 1]$ . For  $k_1 = k_2 = k_3 = 1$ , the shape of

the membership functions for normalized inputs ( $e^*$ ,  $ce^*$ ) and output ( $co^*$ ) are shown in Fig. 3 and Fig. 4 respectively.

For  $x_i \leq -(4 - m)k_i/3$  and  $m = 1$ , we have

$$\mu_{im}(x_i) = 1 \tag{8}$$

For  $-(5 - m)k_i/3 \leq x_i \leq -(4 - m)k_i/3$  and  $m = 2, 3, 4, 5, 6, 7$ , we have

$$\mu_{im}(x_i) = \frac{1}{k_i}[3x_i + (5 - m)k_i] \tag{9}$$

For  $x_i \geq -(3 - m)k_i/3$  and  $m = 1, 2, 3, 4, 5, 6$ , we have

$$\mu_{im}(x_i) = 0 \tag{10}$$

For  $x_i \leq -(5 - m)k_i/3$  and  $m = 2, 3, 4, 5, 6, 7$ , we have

$$\mu_{im}(x_i) = 0 \tag{11}$$

For  $-(4 - m)k_i/3 \leq x_i \leq -(3 - m)k_i/3$  and  $m = 1, 2, 3, 4, 5, 6$ , we have

$$\mu_{im}(x_i) = -\frac{1}{k_i}[3x_i + (3 - m)k_i] \tag{12}$$

For  $x_i \geq -(4 - m)k_i/3$  and  $m = 7$ , we have

$$\mu_{im}(x_i) = 1 \tag{13}$$

For  $y = (q - 1)k_3/6$  and  $q = 1, 2, 3, 4, 5, 6, 7$ , we have

$$\mu_q(y) = 1 \tag{14}$$

For  $y \neq (q - 1)k_3/6$  and  $q = 1, 2, 3, 4, 5, 6, 7$ , we have

$$\mu_q(y) = 0 \tag{15}$$

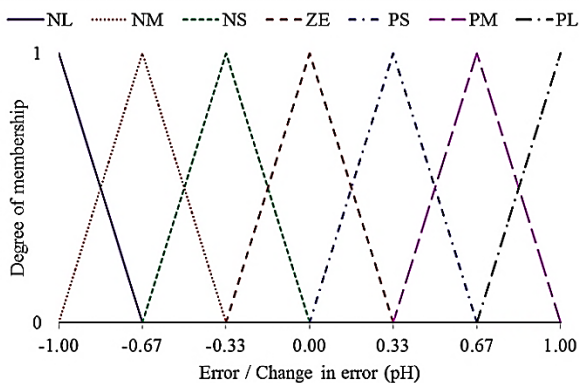


Fig. 3: Normalized membership functions for error and change in error

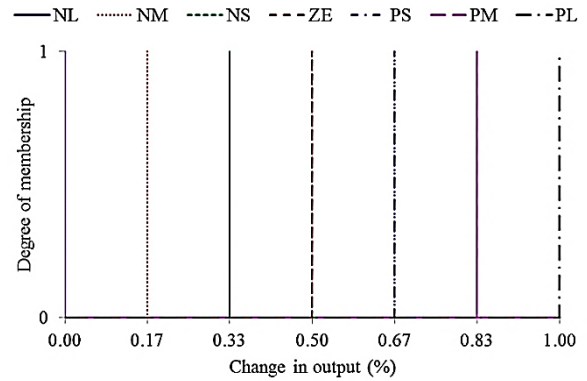


Fig. 4: Normalized membership functions for change in output

Table 1: Fuzzy Rule Base

e	ce						
	NL	NM	NS	ZE	PS	PM	PL
NL	MF <sub>11</sub>	MF <sub>12</sub>	MF <sub>13</sub>	MF <sub>14</sub>	MF <sub>15</sub>	MF <sub>16</sub>	MF <sub>17</sub>
NM	MF <sub>21</sub>	MF <sub>22</sub>	MF <sub>23</sub>	MF <sub>24</sub>	MF <sub>25</sub>	MF <sub>26</sub>	MF <sub>27</sub>
NS	MF <sub>31</sub>	MF <sub>32</sub>	MF <sub>33</sub>	MF <sub>34</sub>	MF <sub>35</sub>	MF <sub>36</sub>	MF <sub>37</sub>
ZE	MF <sub>41</sub>	MF <sub>42</sub>	MF <sub>43</sub>	MF <sub>44</sub>	MF <sub>45</sub>	MF <sub>46</sub>	MF <sub>47</sub>
PS	MF <sub>51</sub>	MF <sub>52</sub>	MF <sub>53</sub>	MF <sub>54</sub>	MF <sub>55</sub>	MF <sub>56</sub>	MF <sub>57</sub>
PM	MF <sub>61</sub>	MF <sub>62</sub>	MF <sub>63</sub>	MF <sub>64</sub>	MF <sub>65</sub>	MF <sub>66</sub>	MF <sub>67</sub>
PL	MF <sub>71</sub>	MF <sub>72</sub>	MF <sub>73</sub>	MF <sub>74</sub>	MF <sub>75</sub>	MF <sub>76</sub>	MF <sub>77</sub>

MF<sub>11</sub>, MF<sub>12</sub>, MF<sub>13</sub>, MF<sub>14</sub>, MF<sub>21</sub>, MF<sub>22</sub>, MF<sub>23</sub>, MF<sub>31</sub>, MF<sub>32</sub>, MF<sub>41</sub>  $\equiv$  NL; MF<sub>15</sub>, MF<sub>24</sub>, MF<sub>33</sub>, MF<sub>42</sub>, MF<sub>51</sub>  $\equiv$  NM; MF<sub>16</sub>, MF<sub>25</sub>, MF<sub>34</sub>, MF<sub>43</sub>, MF<sub>52</sub>, MF<sub>61</sub>  $\equiv$  NS; MF<sub>17</sub>, MF<sub>26</sub>, MF<sub>35</sub>, MF<sub>44</sub>, MF<sub>53</sub>, MF<sub>62</sub>, MF<sub>71</sub>  $\equiv$  ZE; MF<sub>27</sub>, MF<sub>36</sub>, MF<sub>45</sub>, MF<sub>54</sub>, MF<sub>63</sub>, MF<sub>72</sub>  $\equiv$  PS; MF<sub>37</sub>, MF<sub>46</sub>, MF<sub>55</sub>, MF<sub>64</sub>, MF<sub>73</sub>  $\equiv$  PM; MF<sub>47</sub>, MF<sub>56</sub>, MF<sub>57</sub>, MF<sub>65</sub>, MF<sub>66</sub>, MF<sub>67</sub>, MF<sub>74</sub>, MF<sub>75</sub>, MF<sub>76</sub>, MF<sub>77</sub>  $\equiv$  PL.

To frame fuzzy rule (FR) base, let us suppose  $x_1 = NL$  and  $x_2 = NL$ . Here  $x_1 = NL$  implies that  $pH(k) \gg pH_{SP}$  i.e. the present pH is much away from the setpoint and  $x_2 = NL$  implies that  $pH(k) \gg pH(k-1)$  i.e. the pH response has tendency to move away from the setpoint. Therefore to bring pH back to the setpoint we must increase the acid flow rate by large amount. Similarly, let us suppose next  $x_1 = NL$  and  $x_2 = PL$ . Here  $x_1 = NL$  implies that  $pH(k) \gg pH_{SP}$  i.e. the present pH is much away from the setpoint and  $x_2 = PL$  implies that  $pH(k) \ll pH(k-1)$  i.e. the pH response has tendency to move towards the setpoint. Therefore to bring pH back to the setpoint we must keep the acid flow rate unchanged. The complete fuzzy rule base for FLC is shown in Table I. The individual fuzzy rule structure is represented by

$$FR_z: \text{ IF } x_1 \text{ is } m \text{ AND } x_2 \text{ is } n \text{ THEN } y \text{ is } MF_{mn}$$

where  $z = (7m + n - 7)$ , and  $m, n = 1, 2, 3, 4, 5, 6, 7$  represents NL, NM, NS, ZE, PS, PM, PL respectively.

The defuzzified output of the FLC ( $y_{FC}$ ) is given by (16).

$$y_{FC}(x_1, x_2) = \frac{\sum_{n=1}^7 \sum_{m=1}^7 \{[\min(\mu_{1m}(x_1), \mu_{2n}(x_2))] MF_{mn}\}}{\sum_{n=1}^7 \sum_{m=1}^7 \{\min(\mu_{1m}(x_1), \mu_{2n}(x_2))\}} \quad (16)$$

#### IV. Design of SOFLC

The FLC for inherently nonlinear pH neutralization process can be optimized for a given operating condition. However the controller needs to be reoptimized if the operating condition changes. This necessitates development of adaptive controller which have adjustable parameters and a mechanism for adjusting the parameters [31]. To make the FLC performance adaptive, the self-organizing technique is incorporated using performance correction table (PC) shown in Table II where  $k$  is a positive scaling factor. The self-organizing strategy uses non-zero penalties for non-performance and zero penalties for meeting performance. We know if  $-6 < x_1 < -2.5$  and  $-6 < x_2 < -2.5$ , then the pH response is moving away from the setpoint and to bring pH back to the setpoint we must increase the acid flow rate. Therefore we can impose a non-zero penalty on each output membership functions which will result in more increase in acid flow rate. Similarly if  $-6 < x_1 < -2.5$  and  $0.5 < x_2 < 6$ , then we will impose zero penalties on each output membership functions since the pH response is already moving towards the setpoint.

The self-organizing mechanism updates FLC output membership as per following rule.

If  $-6 < x_1 < -2.5$  and  $-6 < x_2 < -2.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{11} \times 10^{-5}$$

If  $-6 < x_1 < -2.5$  and  $-2.5 < x_2 < -1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{12} \times 10^{-5}$$

If  $-6 < x_1 < -2.5$  and  $-1.5 < x_2 < -0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{13} \times 10^{-5}$$

If  $-6 < x_1 < -2.5$  and  $-0.5 < x_2 < 0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{14} \times 10^{-5}$$

If  $-2.5 < x_1 < -1.5$  and  $-6 < x_2 < -2.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{21} \times 10^{-5}$$

If  $-2.5 < x_1 < -1.5$  and  $-2.5 < x_2 < -1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{22} \times 10^{-5}$$

If  $-2.5 < x_1 < -1.5$  and  $-1.5 < x_2 < -0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{23} \times 10^{-5}$$

If  $-2.5 < x_1 < -1.5$  and  $-0.5 < x_2 < 0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{24} \times 10^{-5}$$

If  $-1.5 < x_1 < -0.5$  and  $-6 < x_2 < -2.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{31} \times 10^{-5}$$

If  $-1.5 < x_1 < -0.5$  and  $-2.5 < x_2 < -1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{32} \times 10^{-5}$$

If  $-1.5 < x_1 < -0.5$  and  $-1.5 < x_2 < -0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{33} \times 10^{-5}$$

If  $-1.5 < x_1 < -0.5$  and  $-0.5 < x_2 < 0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{34} \times 10^{-5}$$

If  $-0.5 < x_1 < 0.5$  and  $-6 < x_2 < -2.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{41} \times 10^{-5}$$

If  $-0.5 < x_1 < 0.5$  and  $-2.5 < x_2 < -1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{42} \times 10^{-5}$$

If  $-0.5 < x_1 < 0.5$  and  $-1.5 < x_2 < -0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{43} \times 10^{-5}$$

If  $-0.5 < x_1 < 0.5$  and  $0.5 < x_2 < 1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{45} \times 10^{-5}$$

If  $-0.5 < x_1 < 0.5$  and  $1.5 < x_2 < 2.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{46} \times 10^{-5}$$

If  $-0.5 < x_1 < 0.5$  and  $2.5 < x_2 < 6$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{47} \times 10^{-5}$$

If  $0.5 < x_1 < 1.5$  and  $-0.5 < x_2 < 0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{54} \times 10^{-5}$$

If  $0.5 < x_1 < 1.5$  and  $0.5 < x_2 < 1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{55} \times 10^{-5}$$

If  $0.5 < x_1 < 1.5$  and  $1.5 < x_2 < 2.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{56} \times 10^{-5}$$

If  $0.5 < x_1 < 1.5$  and  $2.5 < x_2 < 6$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{57} \times 10^{-5}$$

If  $1.5 < x_1 < 2.5$  and  $-0.5 < x_2 < 0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{64} \times 10^{-5}$$

If  $1.5 < x_1 < 2.5$  and  $0.5 < x_2 < 1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{65} \times 10^{-5}$$

If  $1.5 < x_1 < 2.5$  and  $1.5 < x_2 < 2.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{66} \times 10^{-5}$$

If  $1.5 < x_1 < 2.5$  and  $2.5 < x_2 < 6$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{67} \times 10^{-5}$$

If  $2.5 < x_1 < 6$  and  $-0.5 < x_2 < 0.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{74} \times 10^{-5}$$

If  $2.5 < x_1 < 6$  and  $0.5 < x_2 < 1.5$ , then

$$MF_{nn}(\text{new}) = MF_{nn}(\text{old}) + PC_{75} \times 10^{-5}$$

If  $2.5 < x_1 < 6$  and  $1.5 < x_2 < 2.5$ , then

$$MF_{nm}(\text{new}) = MF_{nm}(\text{old}) + PC_{76} \times 10^{-5}$$

If  $2.5 < x_1 < 6$  and  $2.5 < x_2 < 6$ , then

$$MF_{nm}(\text{new}) = MF_{nm}(\text{old}) + PC_{77} \times 10^{-5}$$

where  $m, n = 1, 2, 3, 4, 5, 6, 7$ , and  $PC_{mn}$  are the elements of fuzzy performance correction table as shown in Table II.

Table 2: Performance Correction Table

e	ce						
	-6 to -2.5	-2.5 to -1.5	-1.5 to -0.5	-0.5 to 0.5	0.5 to 1.5	1.5 to 2.5	2.5 to 6
-6 to -2.5	-3k	-3k	-3k	-3k	0	0	0
-2.5 to -1.5	-3k	-2k	-2k	-2k	0	0	0
-1.5 to -0.5	-2k	-2k	-1k	-1k	0	0	0
-0.5 to 0.5	-2k	-1k	-1k	0	1k	1k	2k
0.5 to 1.5	0	0	0	1k	1k	2k	2k
1.5 to 2.5	0	0	0	2k	2k	2k	3k
2.5 to 6	0	0	0	3k	3k	3k	3k

V. FIS Optimization using GA

The flowchart for GA optimization of FIS, conventional or self-organized, is shown in Fig. 5 [32]. To optimize the FIS, the scaling factors  $k_1, k_2$ , and  $k_3$  are chosen to proportionately scale the vertices of membership functions for normalized variables  $e^*, ce^*$ , and  $co^*$  respectively as given in (8), (9), (10), (11), (12), (13), (14) and (15). To optimize the performance correction table, the scaling factor  $k$  is chosen.

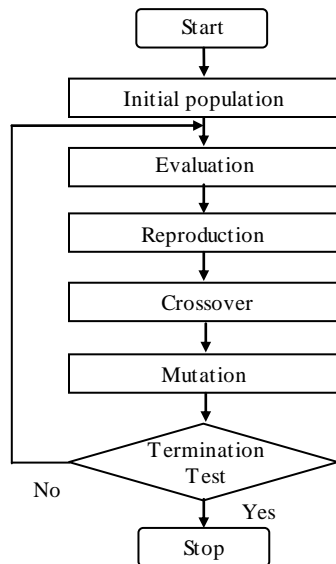


Fig. 5: Flowchart for GA optimization

GA optimization starts with an initial population of individuals of type ‘double’ and size ‘20’, generated at random within the initial population range representing minimum and maximum values for optimization variables. Each individual in the population represents a

potential solution to the optimization problem under consideration. The individuals evolve through successive iterations, called generations. During each generation, each individual in the population is evaluated using integral of squared errors (ISE) fitness function. The evaluated fitness values of the individuals are ranked in an increasing order such that the best elite individual having minimum fitness value has the rank as 1, the next elite individual fit has the rank as 2, and so on. The rank fitness scaling function is used to assign scaled values to the individuals inversely proportional to square root of their rank. Based on the assigned scaled values by the fitness scaling function, the selection function is used to select parents for next generation. The stochastic uniform selection function represents a line in which each parent corresponds to a section of the line of length proportional to its scaled value. The GA moves along the line in steps of equal size and, at each step, the GA allocates a parent to the section it occupies. Using current generation individuals, the next generation population is created through genetic operators namely reproduction, crossover and mutation. The reproduction operation has the elite children count as 2. Besides elite children, the GA uses current generation parents to create next generation crossover and mutation children. The GA creates the crossover children by combining a pair of parents whereas the mutation children are created by applying random changes to a single parent. The GA uses scattered crossover with fraction as 0.8 to create 14 crossover children. The remaining 4 next generation children are created using Gaussian mutation with both scale and shrink as 0.01. The procedure continues until the termination condition is satisfied. The GA optimization parameters are given in Table III. Unless otherwise indicated, the values are true for servo and regulatory operations of both FLC and SOFLC.

**VI. Simulation Results and Discussions**

The schematic block diagram for implementation and optimization of SOFLC for pH neutralization process is shown in Fig. 6. The output membership functions for variable ‘co’ are placed within the range  $[0, k_3]$  for conventional FLC, as given in (14) and (15). The output membership functions for variable ‘col’ are placed within the range  $[-40k_3, 40k_3]$ . The performance of

optimized "intelligent" controller, conventional as well as self-organized, are compared for servo and regulatory operations based on ISE, maximum overshoot and undershoot, settling and rejection time (for  $\pm 2\%$  pH error band) using MATLAB simulation. To calculate ISE, all the errors with magnitude greater than or equal to 0.01 are considered since further smaller errors will have negligible contribution.

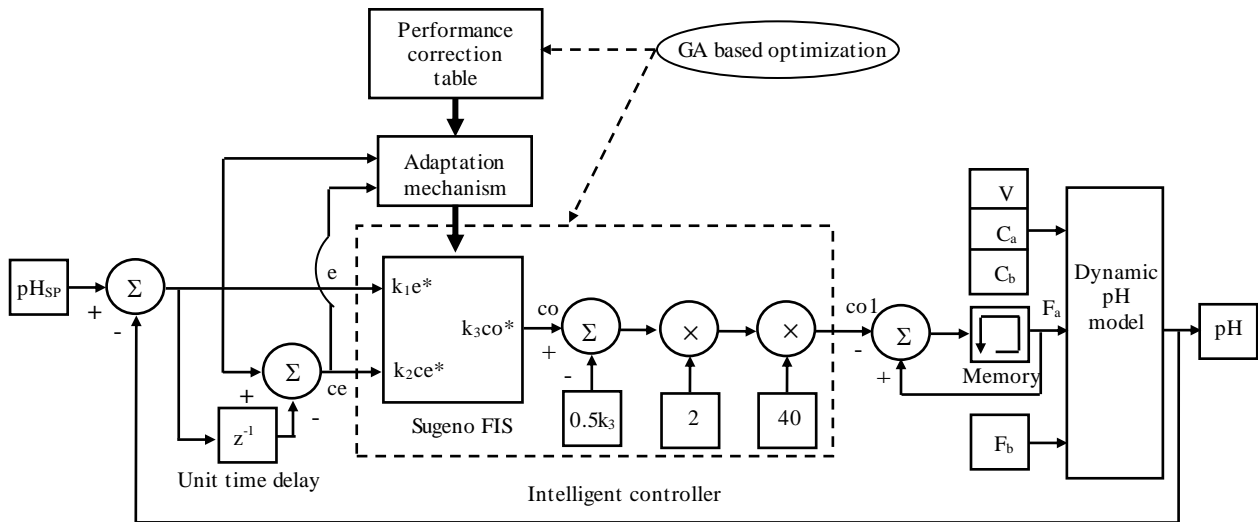


Fig. 6: Optimized and self-organized fuzzy logic based pH control

Table 3: GA Optimization Parameters

Parameters	Descriptions
Variables	$(k_1, k_2, k_3)$ FLC $(k_1, k_2, k_3, k)$ SOFLC
Population	Type: Double vector Size: 20 Creation function: Uniform Initial range: [0 0 0; 4.5 4.5 0.03] FLC Servo [0 0 0; 4.5 4.5 0.05] FLC Regulatory [0 0 0 0; 4.5 1.5 0.02 0.05] SOFLC Servo [0 0 0 0; 4.5 4.5 0.06 0.08] SOFLC Regulatory
Fitness scaling	Function: Rank
Selection	Function: Stochastic uniform
Reproduction	Elite count: 2 Crossover fraction: 0.8
Mutation	Function: Gaussian Scale: 0.01 Shrink: 0.01
Crossover	Function: Scattered
Migration	Direction: Forward Fraction: 0.2 Interval: 20
Constraint parameters	Initial penalty: 10 Penalty factor: 100
Stopping criteria	Generations: 100 Stall generations: 50 Function tolerance: $10^{-6}$

**6.1 Servo Response**

For servo operation, the pH setpoint ( $pH_{SP}$ ) variation is shown below for various sampling instants (k).

For  $1 \leq k \leq 600$ , we have  $pH_{SP}(k) = 7$

For  $601 \leq k \leq 1200$ , we have  $pH_{SP}(k) = 8$

For  $1201 \leq k \leq 1800$ , we have  $pH_{SP}(k) = 9$

For  $1801 \leq k \leq 2400$ , we have  $pH_{SP}(k) = 8$

For  $2401 \leq k \leq 3000$ , we have  $pH_{SP}(k) = 7$

For  $3001 \leq k \leq 3600$ , we have  $pH_{SP}(k) = 6$

For  $3601 \leq k \leq 4200$ , we have  $pH_{SP}(k) = 5$

For  $4201 \leq k \leq 4800$ , we have  $pH_{SP}(k) = 6$

For  $4801 \leq k \leq 5400$ , we have  $pH_{SP}(k) = 7$

The process variable  $F_b$  is kept constant at 1.75 mL/s. The self-organization adaptation loop is set to run for 10 iterations. The plot of best and mean values of ISE as a result of GA optimization process for FLC and SOFLC are shown in Fig. 7 and Fig. 8. The final mean fitness values are closer to best fitness values which indicate good GA convergence. The resulting values of scaling factors  $k_1, k_2, k_3$ , and  $k$  are given in Table IV. The performance characteristic of self-organization technique is shown in Fig. 9. The variations of



controlled and manipulated variables for servo response for selected duration are shown in Fig. 10 and 11 respectively.

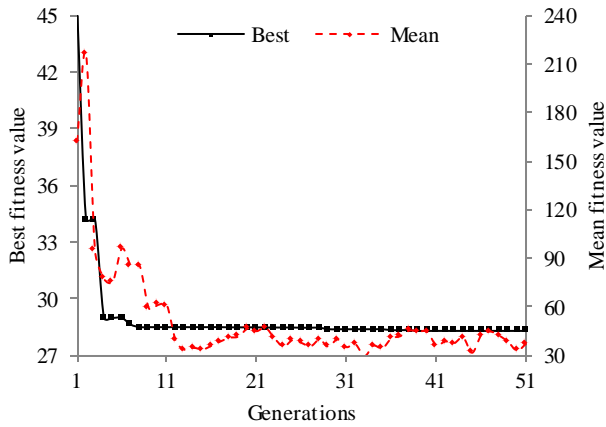


Fig. 7: GA optimization for FLC (servo response)

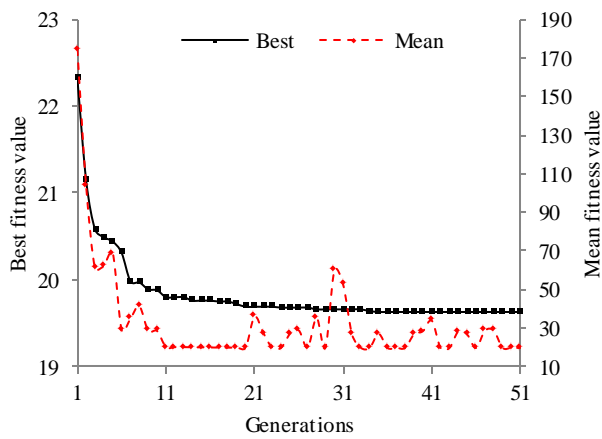


Fig. 8: GA optimization for SOFLC (servo response)

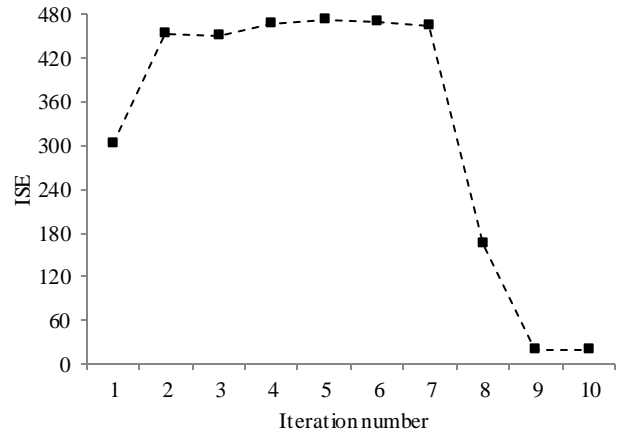


Fig. 9: Self-organizing performance characteristic for servo response

Table 4: Comparison of Servo Response

Parameters	FLC	SOFLC
k1	4.74203	3.56715
k2	0.85630	0.49444
k3	0.01113	0.01576
k	0	0.04646
ISE	28.2985	19.6285
Maximum undershoot* <sup>1</sup>	0.87818	0.27198
Maximum overshoot* <sup>2</sup>	0.87896	0.27122
Settling time* <sup>1</sup>	28 samples	15 samples
Settling time* <sup>2</sup>	28 samples	15 samples

\*<sup>1</sup> For pH<sub>SP</sub> changes from 9 to 8. \*<sup>2</sup> For pH<sub>SP</sub> changes from 5 to 6.

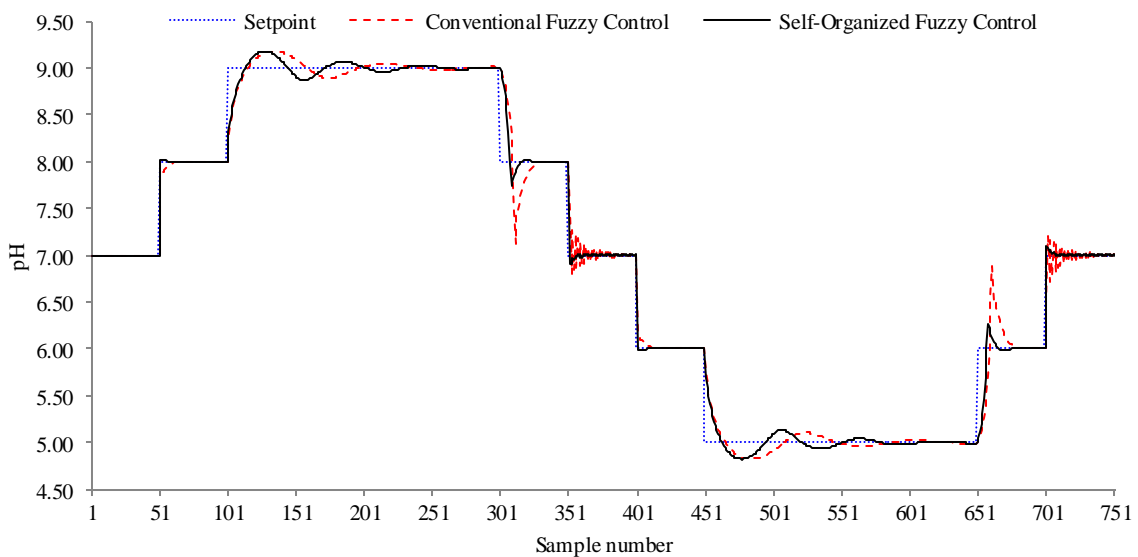


Fig. 10: Controlled variable variations for servo response for selected duration



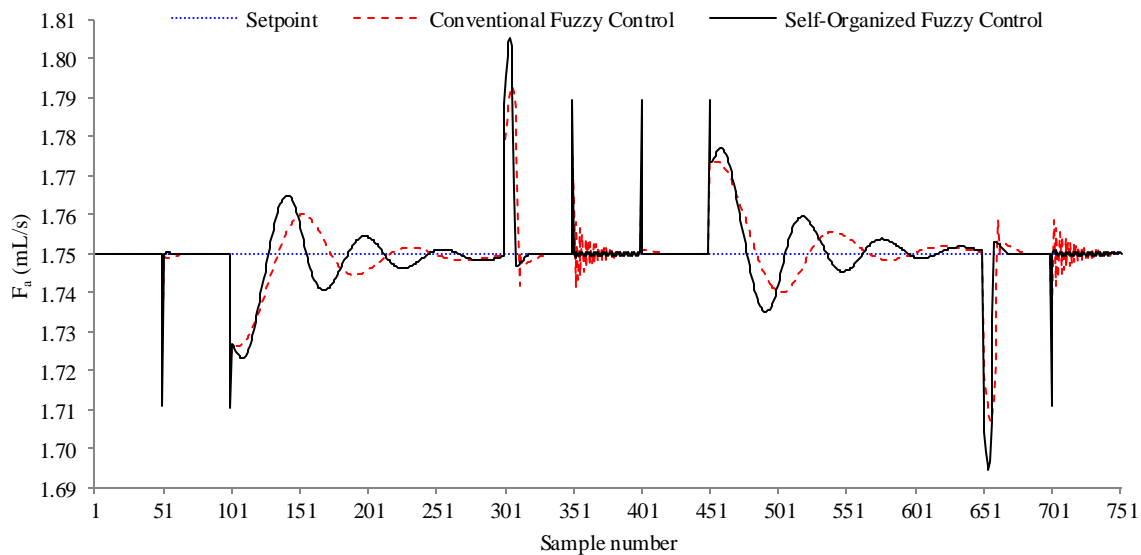


Fig. 11: Manipulated variable variations for servo response for selected duration

In comparison with FLC, the SOFLC demonstrates superior servo performance as indicated by values in Table IV. The self-organization mechanism reduces the ISE by almost 30%. Both controllers shows worst performance when the  $pH_{SP}$  changes from 9 to 8 and from 5 to 6. However SOFLC again outperforms FLC by reducing maximum undershoot and overshoot by 69%. Finally the SOFLC reduces the settling time by 46% in comparison with FLC.

The self-organization mechanism essentially modifies the output membership functions  $M_{33}, M_{34}, M_{43}, M_{44}, M_{45}, M_{54},$  and  $M_{55}$  as indicated in Table VI.

### 6.2 Regulatory Response

For regulatory operation, the process variable  $F_b$  (in mL/s) variation is shown below for various sampling instants ( $k$ ).

- For  $1 \leq k \leq 600$ , we have  $F_b(k) = 1.75$
- For  $601 \leq k \leq 1200$ , we have  $F_b(k) = 1.8375$
- For  $1201 \leq k \leq 1800$ , we have  $F_b(k) = 1.75$
- For  $1801 \leq k \leq 2400$ , we have  $F_b(k) = 1.6625$
- For  $2401 \leq k \leq 3000$ , we have  $F_b(k) = 1.75$
- For  $3001 \leq k \leq 3600$ , we have  $F_b(k) = 1.925$
- For  $3601 \leq k \leq 4200$ , we have  $F_b(k) = 1.75$
- For  $4201 \leq k \leq 4800$ , we have  $F_b(k) = 1.575$
- For  $4801 \leq k \leq 5400$ , we have  $F_b(k) = 1.75$

The setpoint  $pH_{SP}$  is kept constant at 7. The self-organization adaptation loop is set to run for 10 iterations. The plot of best and mean values of ISE as a

result of GA optimization process for FLC and SOFLC are shown in Fig. 12 and Fig. 13. The final mean fitness values are closer to best fitness values which again indicate good GA convergence for the regulation. The resulting values of scaling factors  $k_1, k_2, k_3,$  and  $k$  are given in Table IV. The performance characteristic of self-organization technique is shown in Fig. 14. The variations of controlled and manipulated variables for regulatory response for selected duration are shown in Fig. 15 and 16 respectively.

In comparison with FLC, the SOFLC demonstrates superior regulatory performance as indicated by values in Table IV. The self-organization mechanism reduces the ISE by almost 30%. Both controllers show almost equal maximum undershoot and overshoot when process is subjected to disturbance of  $\pm 10\%$ . Finally the SOFLC reduces the settling time by more than 60% in comparison with FLC.

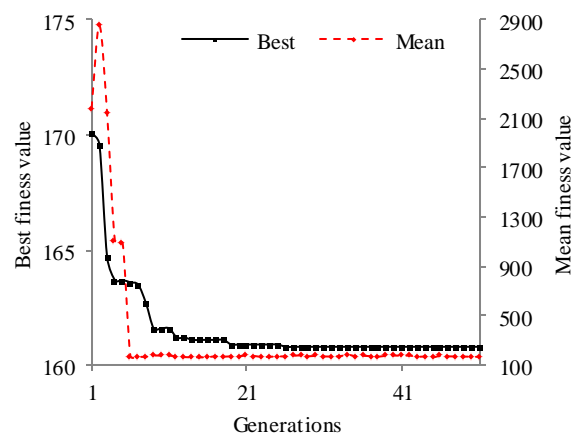


Fig. 12: GA optimization for FLC (regulatory response)

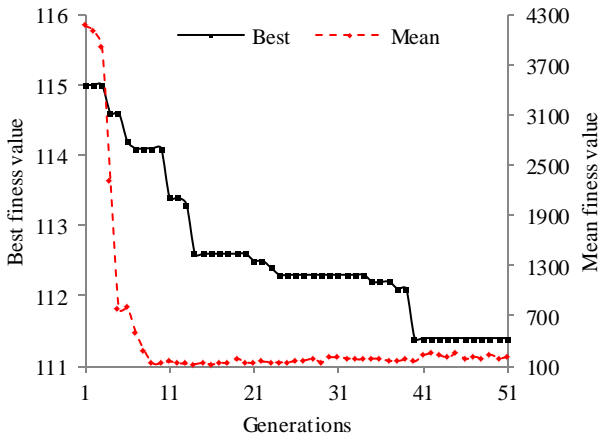


Fig. 13: GA optimization for SOFLC (regulatory response)

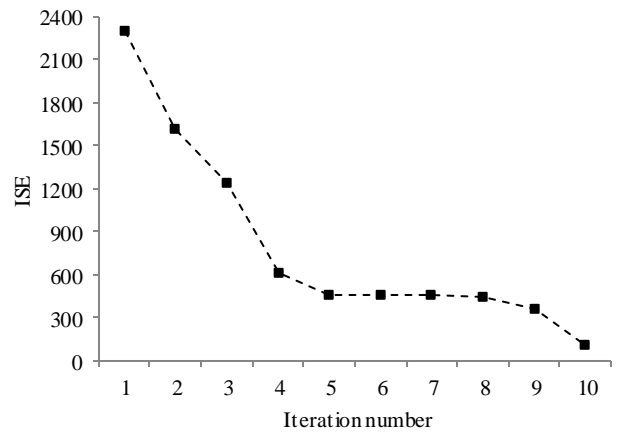


Fig. 14: Self-organizing performance characteristics for regulatory response

Table 5: Comparison of Regulatory Response

Parameters	FLC	SOFLC
k1	4.45597	1.95117
k2	3.95831	3.19950
k3	0.03984	0.03792
k	0	0.07132
ISE	160.782	111.377
Maximum undershoot* <sup>1</sup>	1.94240	1.82714
Maximum overshoot* <sup>2</sup>	1.94209	1.82710
Rejection time* <sup>1</sup>	111 samples	44 samples
Rejection time* <sup>2</sup>	114 samples	33 samples

\*<sup>1</sup> FOR F<sub>b</sub> CHANGES BY -10%. \*<sup>2</sup> FOR F<sub>b</sub> CHANGES BY +10%.

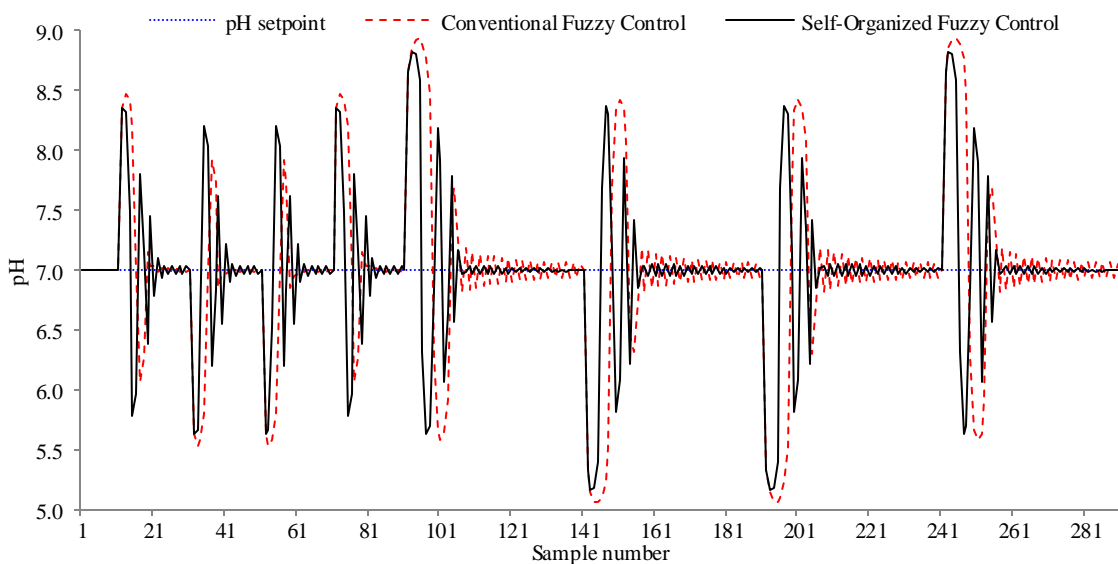


Fig. 15: Controlled variable variations for regulatory response for selected duration

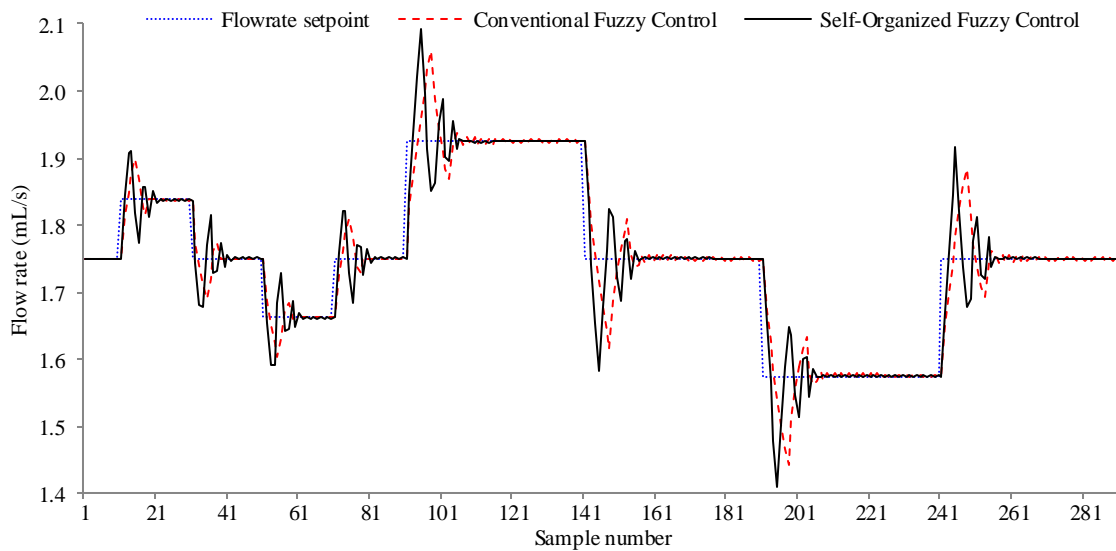


Fig. 16: Manipulated variable variations for regulatory response for selected duration

Table 6: Modification of Fuzzy Rule Base

e	ce							
	NL	NM	NS	ZE	PS	PM	PL	
NL	0	0	0	0	0.00189	0.00379	0.00557	FLC Servo
	0	0	0	0	0.00677	0.01355	0.01992	FLC Regulatory
	0	0	0	0	0.00268	0.00536	0.00788	SOFLC Servo* <sup>1</sup>
	0	0	0	0	0.00268	0.00536	0.00788	SOFLC Servo* <sup>2</sup>
	0	0	0	0	0.00645	0.01289	0.01896	SOFLC Regulatory* <sup>1</sup>
	0	0	0	0	0.00645	0.01289	0.01896	SOFLC Regulatory* <sup>2</sup>
NM	0	0	0	0.00189	0.00379	0.00557	0.00735	FLC Servo
	0	0	0	0.00677	0.01355	0.01992	0.02629	FLC Regulatory
	0	0	0	0.00268	0.00536	0.00788	0.01040	SOFLC Servo* <sup>1</sup>
	0	0	0	0.00268	0.00536	0.00788	0.01040	SOFLC Servo* <sup>2</sup>
	0	0	0	0.00645	0.01289	0.01896	0.02503	SOFLC Regulatory* <sup>1</sup>
	0.00000	0.00000	0.00000	0.00637	0.01289	0.01896	0.02503	SOFLC Regulatory* <sup>2</sup>
NS	0	0	0.00189	0.00379	0.00557	0.00735	0.00924	FLC Servo
	0	0	0.00677	0.01355	0.01992	0.02629	0.03307	FLC Regulatory
	0	0	0.00268	0.00536	0.00788	0.01040	0.01308	SOFLC Servo* <sup>1</sup>
	0	0	0.00402	0.00534	0.00788	0.01040	0.01308	SOFLC Servo* <sup>2</sup>
	0	0	0.00645	0.01289	0.01896	0.02503	0.03147	SOFLC Regulatory* <sup>1</sup>
	0.00000	0.00000	0.00916	0.01297	0.01900	0.02503	0.03147	SOFLC Regulatory* <sup>2</sup>
ZE	0	0.00189	0.00379	0.00557	0.00735	0.00924	0.01113	FLC Servo
	0	0.00677	0.01355	0.01992	0.02629	0.03307	0.03984	FLC Regulatory
	0	0.00268	0.00536	0.00788	0.01040	0.01308	0.01576	SOFLC Servo* <sup>1</sup>
	0	0.00268	0.00690	0.00788	0.00885	0.01308	0.01576	SOFLC Servo* <sup>2</sup>
	0	0.00645	0.01289	0.01896	0.02503	0.03147	0.03792	SOFLC Regulatory* <sup>1</sup>
	0	0.00645	0.01653	0.01897	0.02142	0.03148	0.03792	SOFLC Regulatory* <sup>2</sup>
PS	0.00189	0.00379	0.00557	0.00735	0.00924	0.01113	0.01113	FLC Servo
	0.00677	0.01355	0.01992	0.02629	0.03307	0.03984	0.03984	FLC Regulatory
	0.00268	0.00536	0.00788	0.01040	0.01308	0.01576	0.01576	SOFLC Servo* <sup>1</sup>

PS	0.00268	0.00536	0.00788	0.01042	0.01176	0.01576	0.01576	SOFLC Servo* <sup>2</sup>
	0.00645	0.01289	0.01896	0.02503	0.03147	0.03792	0.03792	SOFLC Regulatory* <sup>1</sup>
	0.00645	0.01289	0.01891	0.02496	0.02872	0.03792	0.03792	SOFLC Regulatory* <sup>2</sup>
PM	0.00379	0.00557	0.00735	0.00924	0.01113	0.01113	0.01113	FLC Servo
	0.01355	0.01992	0.02629	0.03307	0.03984	0.03984	0.03984	FLC Regulatory
	0.00536	0.00788	0.01040	0.01308	0.01576	0.01576	0.01576	SOFLC Servo* <sup>1</sup>
	0.00536	0.00788	0.01040	0.01308	0.01576	0.01576	0.01576	SOFLC Servo* <sup>2</sup>
	0.01289	0.01896	0.02503	0.03147	0.03792	0.03792	0.03792	SOFLC Regulatory* <sup>1</sup>
	0.01289	0.01896	0.02503	0.03155	0.03792	0.03792	0.03792	SOFLC Regulatory* <sup>2</sup>
PL	0.00557	0.00735	0.00924	0.01113	0.01113	0.01113	0.01113	FLC Servo
	0.01992	0.02629	0.03307	0.03984	0.03984	0.03984	0.03984	FLC Regulatory
	0.00788	0.01040	0.01308	0.01576	0.01576	0.01576	0.01576	SOFLC Servo* <sup>1</sup>
	0.00788	0.01040	0.01308	0.01576	0.01576	0.01576	0.01576	SOFLC Servo* <sup>2</sup>
	0.01896	0.02503	0.03147	0.03792	0.03792	0.03792	0.03792	SOFLC Regulatory* <sup>1</sup>
	0.01896	0.02503	0.03147	0.03792	0.03792	0.03792	0.03792	SOFLC Regulatory* <sup>2</sup>

\*<sup>1</sup> - Before adaptation mechanism

\*<sup>2</sup> - After adaptation mechanism

The self-organization mechanism essentially modifies the output membership functions  $M_{21}$ ,  $M_{22}$ ,  $M_{23}$ ,  $M_{24}$ ,  $M_{31}$ ,  $M_{32}$ ,  $M_{33}$ ,  $M_{34}$ ,  $M_{35}$ ,  $M_{42}$ ,  $M_{43}$ ,  $M_{44}$ ,  $M_{45}$ ,  $M_{46}$ ,  $M_{53}$ ,  $M_{54}$ ,  $M_{55}$ ,  $M_{56}$ ,  $M_{57}$ ,  $M_{64}$ ,  $M_{65}$ ,  $M_{66}$ , and  $M_{67}$  as indicated in Table VI.

## VII. Conclusion

In this paper, the intelligent controller for pH neutralization process is designed by incorporating the self-organizing mechanism in the conventional FLC to make it performance adaptive. The SOFLC adaptation mechanism modifies the output membership function positions using performance correction table. In order to have optimal performance, parameters of both FLC as well as SOFLC are optimized using GA. In case of servo response, the ISE and maximum undershoot and overshoot for SOFLC are decreased in comparison with FLC, by approximately 30% and 69% respectively. Also, the settling time for servo response is reduced by 46% for SOFLC in comparison with FLC. Similarly, in case of regulatory response, the ISE and the rejection time for SOFLC are less by approximately 30% and 50-60% respectively, in comparison with FLC.

## References

- [1] Shinskey F G. Process-Control Systems: Application / Design / Adjustment. McGraw-Hill, USA, 1979.
- [2] Mellichamp D A, Coughanowr D R, Koppel L B. Characterization and gain identification of time varying flow processes. A.I.Ch.E. J., 1966, 1 (12):75-82.
- [3] Mellichamp D A, Coughanowr D R, and Koppel L B. Identification and adaptation in control loops with time varying gain. AIChE, 1966, 1 (12):83-89.
- [4] McAvoy T J, Hsu E, Lowenthal S. Dynamics of pH in controlled stirred tank reactor. Ind. Eng. Chem. Process Des. Develop., 1972, 1 (11):68-70.
- [5] McAvoy T J, Hsu E, Lowenthal S. Time optimal and Ziegler-Nichols control. Ind. Eng. Chem. Process Des. Develop., 1972, 1 (11): 71-78.
- [6] Gustafsson T K, Waller K V. Dynamic modeling and reaction invariant control of pH. Chemical Engineering Science, 1983, 3 (38):389-398.
- [7] Gustafsson T K. An experimental study of a class of algorithms for adaptive pH control. Chemical Engineering Science, 1985, 5 (40):827-837.
- [8] Wright R A, Kravaris C. Nonlinear control of pH processes using strong acid equivalent. Ind. Eng. Chem. Process Des. Develop., 1991, 7 (30):1561-1572.
- [9] Wright R A, Soroush M, Kravaris C. Strong acid equivalent control of pH processes: An experimental study. Ind. Eng. Chem. Process Des. Develop., 1991, 11 (30):2437-2444.
- [10] Bhat N, McAvoy T J. Use of neural nets for dynamic modeling and control of chemical process system. Proceedings of the 1989 American Control Conference. Pittsburg, USA: Institute of Electrical and Electronics Engineers, 1989. 1342-1348.
- [11] Draeger A, Ranke H, Engell S. Neural network based model predictive control of a continuous neutralization reactor. Proceedings of the third IEEE conference on Control Applications. Strathclyde University, Glasgow: Institute of

- Electrical and Electronics Engineers, 1994. 427-432.
- [12] Zhang J, Morris J. Neuro-fuzzy networks for modelling and model-based control. Proceedings of the IEE Colloquium on Neural and Fuzzy systems: Design, Hardware and Applications. London, UK: The Institution of Engineering and Technology, 1997. 6/1-6/4.
- [13] Zadeh L A. Fuzzy sets. *Information and Control*, 1965, 3 (8):338-353.
- [14] Zadeh L A. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans. on Systems, Man, and Cybernetics*, 1973, 1 (3):28-44.
- [15] Zadeh L A. From computing with numbers to computing with words – From manipulation of measurements to manipulation of perceptions. *Int. J. Appl. Math. Comput. Sci.*, 2002, 3 (12):307-324.
- [16] Zadeh L A. Is there a need for fuzzy logic? *Information Sciences*, 2008, 13 (178):2751-2779.
- [17] Mamdani E H, Assilian S. An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man-Machine Studies*, 1975, 1 (7):1-13.
- [18] Procyk T J, Mamdani E H. A linguistic self-organizing process controller. *Automatica*, 1979, 1 (15):15-30.
- [19] Mamdani E H. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Trans. On Computers*, 1976, 12 (26):1182-1191.
- [20] Erenoglu I, Eksin I, Yesil E, et al. An intelligent hybrid fuzzy PID controller. Proceedings of the 20th European Conference on Modeling and Simulation. Bonn, Germany: Bonn-Rhein-Sieg University of Applied Sciences, 2006.
- [21] Daroogheh N. High gain adaptive control of a neutralization process pH. Proceedings of Chinese Control and Decision Conference. Guilin, People's Republic of China: Institute of Electrical and Electronics Engineers, 2009. 3477-3480.
- [22] Saji K S, Sasi K M. Fuzzy sliding mode control for a pH process. Proceedings of IEEE International Conference on Communication Control and Computing Technologies. Ramanathapuram, India: Institute of Electrical and Electronics Engineers, 2010. 276-281.
- [23] Takagi T, Sugeno M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Systems, Man, and Cybernetics*, 1985, 1 (15):116-132.
- [24] Jang J–S R, Sun C–T. Neuro-fuzzy modeling and control. *Proc. of the IEEE*, 1995, 3 (83):378-406.
- [25] Holland J H. *Adaptation in natural and artificial systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT, Cambridge, 1992.
- [26] Goldberg D E. *Genetic Algorithms in Search, Optimization, & Machine Learning*. Pearson, 1989.
- [27] Wang P, Kwok D P. Optimal fuzzy PID control based on genetic algorithm. Proceedings of the 1992 International Conference on Industrial Electronics, Control, Instrumentation, and Automation. San Diego, California: Institute of Electrical and Electronics Engineers, 1992. 977-981.
- [28] Karr C L, Gentry E J. Fuzzy control of pH using genetic algorithms. *IEEE Trans. Fuzzy Systems*, 1993, 1 (1):46-53.
- [29] Bagis A. Determination of the PID controller parameters by modified genetic algorithm for improved performance. *Journal of Information Science and Engineering*, 2007, 5 (23):1469-1480.
- [30] *Fuzzy Logic Toolbox User's Guide*, The MathWorks Inc., USA, <http://www.mathworks.in>.
- [31] Astrom K J, Wittenmark B. *Adaptive control*. Dover Publications, New York, 2008.
- [32] *Global Optimization Toolbox User's Guide*, The MathWorks Inc., USA, <http://www.mathworks.in>

#### Authors' Profiles



**Parikshit Kishor Singh** was born in Kolkata, India in 1977. He received the B.E. (Electronics Engineering) and M.Tech. (Electronics and Instrumentation) degrees from Sardar Vallabhbhai National Institute of Technology (SVNIT) Surat and National Institute of Technology (NIT) Warangal, in 2003 and 2005, respectively. In 2005, he joined the Vignan's Institute of Information Technology (VIIT) Vishakhapatnam as Assistant Professor, under Early Faculty Induction Programme (EFIP) of All India Council of Technical Education (AICTE). In 2007, he joined Birla Institute of Technology and Science (BITS) Pilani as Assistant Lecture. From 2009 onwards, he is designated as Lecturer at BITS Pilani. From 2008 onwards, he is pursuing his doctoral research in the area of applications of artificial intelligence in process control.

**Surekha Bhanot** was born in Patiala, India in 1957. She received the B.E. (Mechanical Engineering), M.Phil. (Instrumentation) and Ph.D. degrees from Birla Institute of Technology and Science (BITS) Pilani, BITS Pilani and Indian Institute of Technology (IIT)



Roorkee, in 1979, 1983 and 1995, respectively. In 1979, she joined BITS Pilani as Teaching Assistant. In 1983, she joined Thapar Institute of Engineering and Technology (TIET) Patiala. In 2002, she joined BITS Pilani as Associate Professor. From 2006 onwards, she is designated as Professor at BITS Pilani. Her research area of interest includes artificial intelligence applications in process modeling and control, biosensor design and applications, and biomedical signal processing and related instrumentation.



**Hare Krishna Mohanta** was born in Odisha, India in 1972. He received the B.E. (Chemical Engineering), M.Tech. (Chemical Engineering) and Ph.D. degrees from NIT Rourkela, IIT Kanpur and BITS Pilani, in 1995, 1998 and 2006, respectively. In 1995, he joined as Lecturer in Indira Gandhi Institute of Technology. In 1995, he joined as a graduate engineer trainee in Indian Rare Earths Limited. In 1998, he joined as senior project associate in Petroleum laboratory of IIT Kanpur. In 1998, he joined BITS Pilani as Assistant Lecturer. In 2000, he got designated as Lecturer at BITS Pilani. From 2006 onwards, he is designated as Assistant Professor at BITS Pilani. His research area of interest includes advanced process control, process monitoring and control, and applied wavelet analysis.

**How to cite this paper:** Parikshit Kishor Singh, Surekha Bhanot, Hare Krishna Mohanta, "Optimized and Self-Organized Fuzzy Logic Controller for pH Neutralization Process", *International Journal of Intelligent Systems and Applications (IJISA)*, vol.5, no.12, pp.99-112, 2013. DOI: 10.5815/ijisa.2013.12.09