International Journal of Power Electronics and Drive Systems (IJPEDS)
Vol. 16, No. 1, March 2025, pp. 212~224
ISSN: 2088-8694, DOI: 10.11591/ijpeds.v16.i1.pp212-224

Comparing multi-control algorithms for complex nonlinear system: An embedded programmable logic control applications

Sochima Vincent Egoigwe[1], Asogwa Tochukwu Chijindu[2], Lois Onyejere Nwobodo[3], Onuigbo Chika Martha[4], Frank Ekene Ozioko[2], Ozor Godwin Odozo[3], Ebere Uzoka Chidi[5]
[1]Department of Mechatronics Engineering, Faculty of Engineering, University of Nigeria Nsukka, Enugu, Nigeria
[2]Department of Computer Science, Faculty of Applied and Physical Science, Enugu State University of Science and Technology, Enugu, Nigeria
[3]Department of Computer Engineering, Faculty of Engineering, Enugu State University of Science and Technology, Enugu, Nigeria
[4]Department of Electrical Electronics Engineering, Faculty of Engineering, Enugu State University of Science and Technology, Enugu, Nigeria
[5]Department of Electrical Electronics Engineering, Faculty of Engineering, University of Nigeria Nsukka, Enugu, Nigeria

Article Info

Article history:

Received Apr 3, 2024
Revised Aug 14, 2024
Accepted Aug 29, 2024

Keywords:

ANN
Control system
GA
PID
Thermodynamic set-points

ABSTRACT

This paper examines the impact of multiple control algorithms, such as genetic algorithm (GA), artificial neural network (ANN), and proportional integral derivative (PID), on programmable logic controller (PLC) performance during a nonlinear thermodynamic process. The ANN was trained with data that modeled the thermodynamic process and then generated the control algorithm. GA was improved by applying the counter-premature algorithm (CPA) to address issues of pre-mature convergence, while the PID presents the current algorithm used to optimize the PLC in the existing testbed. Experimental evaluation of these models against the process set-points showed that all the algorithms were able to reject disturbance and follow the reference set points under different step changes, but each algorithm experienced different internal behaviors while trying to reject disturbance. Lastly, the result showed that while the improved GA was better than the PID, with a recorded slight overshoot due to the uncertainties of the thermodynamic process, the ANN achieved better control performance in terms of system stability than the other counterpart algorithms.

Corresponding Author:

Asogwa Tochukwu Chijindu
Department of Computer Science, Faculty of Applied and Physical Science
Enugu State University of Science and Technology
Agbani, Enugu State, Nigeria
Email: tochukwu.asogwa@esut.edu.ng

1. INTRODUCTION

All over the world, programmable logic controllers (PLCs) have dominated process design and basic process control systems [1]. The idea is to minimize the risk of accidents during the technical process, as research [2] revealed that there is no plant all over the world that is 100% risk-free. The process design has to do with the correct specification of engineering components such as valves, pressure transmitters, actuators, sensors, alarms, etc. in line with the standards of industrial automation [3], while the process control system is the application of PLCs or other control systems for the monitoring and adjustment of process inputs to give the desired output [4] and [5]. Recently, PLCs have gained more attention due to recent advancements in their features, such as wireless control access, larger memory, better processing speed, and programming flexibility [6]. In addition, it has been applied for real-time control operations, which according to Ulagwu-Echefu *et al.* [7], are in great demand in today's industrial settings. However, while the PLC has continued to offer promising solutions to optimize industrial automation, there is still a need for it to offer optimal control, especially in complex nonlinear situations [8]. Process control has become heterogeneous in nature, with multiple parameters, time-invariant constraints, set-points, multi-logic sequences, and all requiring approximation in a short time [9], thus presenting complex control issues for the conventional PLC and hence presenting the need for advanced control system (ACS).

Airikka [10] defines ACS as the application of classical control techniques with the ability to perform complex computations from process modeling, parameter estimation, performance criterion optimization, multivariable, and back-propagation-based control characteristics into the basic process control system to enhance performance. In the context of PLC, these ACS can be applied to optimize the performance through automatic parameter tuning [11], which can be done using external computing machines (ECM) or the application of advanced process control algorithms (APCA) [12]. While these two methods can improve PLC, Tarnawski *et al*. [12] identified APCA as offering better results for safety integrity, quality assurance, and economy when compared to ECM.

Many works of literature on APCA have been presented, recommending various APCA techniques to improve constraint approximation for nonlinear systems. For instance, a study [13] applied the Ziegler Nicholas tuning technique or constraint approximation of a nonlinear system, while a study [14] compared the Ziegler Nicholas, internal model control (IMC), and Shams-IMC techniques, respectively, and reported Shams-IMC as more consistent than the others. However, a study [15] revealed the PID cannot be reliable for the control of multiple variables, which are time-invariant, and suggested an adaptive solution in a bounded least square optimizer solver [16] to address integration issues such as compatibility and function overhead. Study [17] applied fuzzy logic for improved servo motor control, while [18] improved fuzzy logic with Takagi-Sugeono and applied it to optimize the S7-300/400 PLC system. In the same vein, the study [19] applied normalization acceleration-based fuzzy inference engine adjustment to control the scaling input and output coefficients of the PLC, while the study [20] improved the fuzzy control system using a particle swarm algorithm and achieved a better control solution when compared with the results obtained in [19].

The data-driven approach (DDA) [21] was also used for the optimization of PLC performance. DDA can be classified into two approaches, which are statistical and artificial intelligence (AI). The statistical approach used a mathematically inspired heuristic model to solve the optimization problems of the plant, while the AI employed smart optimization approaches and machine learning (ML) algorithms to learn the behavior of the plant and perform control operations. Literature has focused recently on AI. approach for optimization of PLCs, using mostly ML algorithms. For example, Bayesian optimization was used by Mohamed *et al*. [22] to tune the parameters of the cascade controller and achieve robustness to disturbance and optimal tracking performance, while Bhargav [23] applied a back-propagation-based neural network for error detection, fault tolerance, and optimization of PLC. While these studies recorded significant control success, they were not tested considering very complex nonlinear chemical processes with multiple control constraints, which leaves a gap. In [24], a predictive control algorithm developed with a model predictive controller (MPC) was used to optimize the function of SCL500-PLC. The MPC was improved with Nesterov's forest gradient [25] and then embedded into the PLC using structured test programming and tested experimentally on multiple connected tank systems. The MPC was compared with the PID, and the results showed that the MPC has better constraint approximation features. In the study [26], a gradient algorithm was applied to optimize MPC and then integrated into Festo-programmable logic controllers (PLC). An experimental result was performed on a nonlinear process. The results were compared against standard PLC, and it was observed that the improved MPC-based PLC was better. Despite the success, it is still not clear whether the system is effective when tested in a complex nonlinear system with multiple constraints.

In another approach, Zhao *et al*. [27] adopt the Koopman subspace model and a multi-parameter quadratic programming approach to solve the constraints optimization problem in a chilled water plant control. In addition, piecewise-affine control laws and active constraints sets were determined using data-driven partition of disturbance space to reduce power consumption in the chiller plant, while achieving optimal operation irrespective of constraints violations, while Arturo *et al*. [28] applied artificial neural network (ANN) to improve the performance of Allen-Bradley PLC operations during water level control. The study trained a feed-forward neural network algorithm with data from the plant to generate a control model, which was integrated into the PLC using RSLogix 5000. The performance showed the ANN was able to improve the constraint approximation efficiency of PLC when compared with traditional PLC. ML-inspired discrete-time controller was proposed in [29], using PID and neural networks to develop an advanced control algorithm for the approximations of single input and output discrete-time nonlinear systems. In the study, the neural network was trained with the data from the plant. The error between the input and output was minimized using adaptation rules of the PID, with three neurons used as the P-I-D inputs.

The model generated after evaluation and justifying the success was recommended for nonlinear control systems. In the same vein, Lee and Jang [30] trained 1000 data points for the mass spring damper system (MSDS) using a neural network and long short-term memory (LSTM). The models were respectively applied for the self-tuning of PID and control of MSDS. The result showed that both models achieved good performance, but the LSTM was recommended due to its predictive characteristics. In [31], DDA was used for control performance assessment of PID performance. Comparative ML algorithms such as decision tree, extra trees, Adaboast, support vector machine (SVM), among others, were trained with data collected from signal-based closed-looped process systems. The result showed that the SVM achieved better performance efficiency after comparative analysis with other models. [32] used black box multi objective optimization (BBMA) and reinforcement learning (RL) to tune PID. The RL was used to minimize multi-step convergence and facilitate the tuning of the PID, while the BBMA developed with particle swarm (PS) [33] and genetic algorithm (GA) [34] was used to tune the PID, respectively, and comparatively analyzed through simulation experiments. The result showed that the RL achieved faster self-tune when compared to the rest.

Overall, the literature has shown that several approaches have been applied over the years for the optimization of PLC and have all had significant success in the approximation of nonlinear systems; however, it is still not clear which algorithm achieved the best performance. Secondly, majority of the results were not tested considering complex nonlinear process with multiple constraints requiring approximation within a short time, while some of the results, despite their success, require validation through the real-world testbed method. Based on these gaps, the following contribution will be made in this paper. Based on these gaps, the following contribution will be made in this paper: i) A mathematical formulation of the nonlinear problem in a thermodynamic process will be presented; ii) Three notable control algorithms (ANN, PID and GA) will be developed using and integrated to optimize plc respectively. The effectiveness of each algorithm will be accessed experimentally under the considered nonlinear problem; iii) An improved ga will be applied to address issues of pre-mature convergence which has continuously hindered success performance of ga using counter premature algorithm (CPA); and iv) Recommendation will be made of engineers on the choice of the best control algorithm for approximation of complex nonlinear constraints in technical process.

2. METHOD

The methodology used for the study began with the mathematical modeling of a nonlinear thermodynamic process of two connected reactor tanks during an irreversible exothermic reaction. To improve the PLC applied for the system approximation, three control algorithms which are PID, ANN and GA were developed and each integrated separately on the PLC. The GA was also improved with CPA to address issues of pre-mature covariance and improve the approximation process. The three models were integrated into PLC and then experimentally validated under nonlinear conditions. Results obtained from each test are comparative analyzed to identify the most suitable control solution to optimize PLC and maintain stability of the thermodynamic process in real time.

2.1. The nonlinear thermodynamic process

A complex chemical process was described by Li [35] as a dynamic behavior of two connected reactor tank during an irreversible exothermic reaction, which is controlled with water coolant. The flow rate for both reactors are given as F_{j1} and F_{j2}while temperatures of the two reactors are T_{j1} and T_{j2}. The chemical process is modeled with the assumption according to Li [35] that $V_{j1} = V_{j2} = V_j, V_1 = V_2 = V$, $F_o = F_2 = F$ and $F_1 = F + F_R$ as the differential equation which presents the rate of concentration in (1) and (2) respectively for the two reactors and temperatures changes in (3) and (4)and the volumetric flow rate of the chemical process presented in the (5) and (6) respectively.

$$\frac{dC_{A1}}{dt} = \frac{F_0}{V} C_{A0} - \frac{F + F_R}{V} C_{A1} + \frac{F_R}{V} C_{A2} - aC_{A1^{e^{-E/RT_1}}} \tag{1}$$

$$\frac{dC_{A2}}{dt} = \frac{F + F_R}{V} C_{A1} - \frac{F + F_R}{V} C_{A2} - aC_{A2^{e^{-E/RT_2}}} \tag{2}$$

$$\frac{dT_1}{dt} = \frac{F_o}{V} T_o - \frac{F + F_R}{V} T_1 + \frac{F_R}{V} T_2 - \frac{an}{pc_p} C_{A1^{e^{-E/RT_1}}} - \frac{UA}{pc_{pV}} (T_1 - T_{j1}) \tag{3}$$

$$\frac{dT_2}{dt} = \frac{F_o}{V} T_1 - \frac{F + F_R}{V} T_1 + \frac{F_R}{V} T_2 - \frac{an}{pc_p} C_{A2^{e^{-E/R2}}} - \frac{UA}{pc_{pV}} (T_2 - T_{j2}) \tag{4}$$

$$\frac{dT_{J10}}{dt} = \frac{F_{j1}}{V} \left(T_1 - T_{j1}\right) - \frac{UA}{p_j c_j V_j} (T_1 - T_{j1}) \tag{5}$$

$$\frac{dT_{J20}}{dt} = \frac{F_{j2}}{V} \left(T_2 - T_{j2}\right) - \frac{UA}{p_j c_j V_j} (T_2 - T_{j2}) \tag{6}$$

From the model of the thermodynamic behavior considering the three key attributes which are temperature, concentration, and volume of mixture in the reactors, the ideas are to control C_{AN}, T_{N1} and T_{N2} through the variation of C_{AF}, T_{j10} and T_{j20}. The variation between the input temperatureT_F and the controlled temperature values T_F^d is the error as $T_F - T_F^d$. Let the variation between the input and control concentrations be given as $x_{11}C_{A2} - C_{A2,}^d x_{12} f_2$, $and\ that\ of\ the\ temperature\ variation\ be\ given\ as\ x_{21} = T_2 - T_2^d$, $x_{22} = T_{J2} - T_{J2}^d$, $x_{31} = T_1 - T_1^d$, $x_{32} = T_{J1} - T_{J1}^d$. The change in the two reactors in (1)-(6) can be presented as in (7).

$$\left.\begin{aligned} &x_{11} = b_{11}x_{12}, x_{12} = b_{12}u_1, y_1 = x_{11} \\ &x_{21} = b_{21}x_{22}, \phi_{21} + \phi x_{31}. \\ &x_{22} = b_{22}u_2 + \phi_{22} \\ &y_2 = x_{21} \end{aligned}\right\} \tag{7}$$

Where $x_{21} = b_{31}x_{32}, \phi_{31} + \psi w, x_{32}. x_{32} = b_{32}u_3 + \phi_{32}; y_2 =; b_{11} = 1, b_{12} = 1; b_{21} = \frac{UA}{pc_{pV}}, b_{22=} \frac{F_{J2}}{V_j}$; $b_{31} = \frac{UA}{pc_{pV}}, b_{32} = \frac{F_{ji}}{V_j}; \psi = \frac{F_o}{V}, \emptyset = \frac{F + F_R}{V}, w = T_o - T_o^d$ and $u_1 = \frac{F + F_R}{V^2} c_{A0} - f_4;\ u_2 = T_{j20} - T_{j20}^d;\ u_3 = T_{j10} - T_{j10}^d;\ C_{A1} = \frac{V}{F+F_R}\left(x_{12} + \frac{F + F_R}{V}(x_{11} + c_{A2}^d) + a(x_{11} + C_{A2}^d) e^{-(E/R(x_{21} + T_2^d))}\right.$

$$\emptyset_{21} = \frac{F + F_R}{V} T_1^d + \frac{F + F_R}{V}(x_{21} + T_2^d) - \frac{an}{pc_p}; ((x_{11} + C_{A2}^d) e^{-(E/R(x_{21} + T_2^d))} - \frac{UA}{pc_p V}(x_{21} + T_2^d - T_{j2}^d)$$

$$\emptyset_{22} = \frac{F_{J2}}{V}\left(T_{20}^d - x_{22} - T_{j2}^d\right) + \frac{UA}{p_j c_j V_j}(x_{21} + T_2^d - x_{22} - T_{j2}^d)$$

$$\emptyset_{31} = \frac{F_o}{V} T_o^d - \frac{F+F_R}{V}(x_{31} + T_1^d) - \frac{an}{pc_p} C_{A1^{e^{-(E/R(x_{21} + T_1^d))}}} - \frac{F_R}{V}(x_{21} + T_2^d) - \frac{UA}{pc_p V}(x_{31 + T_1^d - T_{j1}^d})$$

$$\emptyset_{32=} \frac{F_{j1}}{V_j}\left(T_{j10}^d - x_{32} - T_{j1}^d\right) + \frac{UA}{p_j c_j V_j}(x_{31} + T_1^d - x_{32} - T_{j1}^d)$$

$f_1 = \frac{F+F_R}{V} C_{A1} + \frac{F_R}{V} C_{A2} - aC_{A1^{e^{-(E/RT1))}}}$; $f_2 = \frac{F+F_R}{V} C_{A1} + \frac{F_R}{V} C_{A2} - aC_{A2^{e^{-(E/RT2))}}}$; $f_3 = \frac{F+F_R}{V} T_1 + \frac{F_R}{V} T_2 - aC_{A1^{e^{-(E/RT1))}}} - \frac{UA}{pc_p V}(T_2 - T_{j2}); f_4 = \frac{F+F_R}{V} f_1 - \frac{F + F_R}{V} + ae^{-(E/RT1))} * f_2 - a\frac{E}{RT_2^2} C_{A2^{e^{-(E/RT2))}}} f_3$

The (1) to (7) were presented with the objective of covering the system output to zero. This was achieved using the uncertain parameters in (4) and (5) to develop control algorithms that will be programmed in the PLC to control the complex reactors. The objective function is to use the flow rate (F_j) of coolant as input to stability of the reactor as the controlled concentration in (8); controlled temperature in (9) and the control temperature change of the coolant as a result of thermodynamics within the reactor and then the difference between the coolant and its initial temperature presented in the (10).

$$(dC_A)/dt = F/V * (C_{A0} - C_A) - a * C_A^{e^{-\frac{E}{RT}}} \tag{8}$$

$$(dT)/dt = F/V * (T_0 - T) - an/(pc\_p) * C_A^{e^{-\frac{E}{RT}}} - UA/(pc_{pV}) * (T - T_j) \tag{9}$$

$$(dT_j)/dt = F_j/V_j * (T_{j0} - T_j) + UA/(p_j * c_j * V_j) * (T - T_j) \tag{10}$$

2.2. Basics of the PLC

The PLC operated based on the cyclic scanning method in which its operating system monitors the timer and the collected data from the input module to check the status of all input devices. The processor used the application software based on the workflow of the ACS algorithm programmed using the ladder logic method, to instruct and adjust the PLC control parameters to match the desired output, based on internal computations and then write the data into the output module, and the scan cycle continues. The power supply ensures the regulated power low into the entire system, via the conversion of the incoming alternating current into direct current. The input module connects the sensors and transmitters to the central processing which use the optimization algorithm programmed using ladder logic, structured text, or function block method to adjust the PLC control parameters to match the desired output and then used to control other output devices. The ethernet is the communication section of the PLC which is used to interface other computers for the monitoring and analysis of the technical process. The PLC programming specifications are in Table 1.

Table 1. The PLC specifications

Parameters	Values	Parameters	Values
Current	4-20 mA	Input port	3
Program memory with run mode	12289 bytes	Output port	3
Program memory without run mode	16384 bytes	Communication interface	RS485
Data memory	10241 bytes	Power supply	220-24 V/DC
Backup memory	100 hrs	Analogue adjustment	2
Speed of computation	2 at 200 MHz	Floating point	Yes

2.3. Advance control algorithm

In this section, AI-inspired control algorithms are proposed and presented to optimize control performance of PLC. Popular algorithms such as proportional integral derivate, genetic algorithm, and neural network algorithms are proposed respectively to facilitate tuning of the PLC control system. This performance will be comparatively analyzed the best selected for system integrated to optimize technical processes.

2.3.1. Genetic algorithm

Genetic algorithm (GA) is a random search used in solving complex optimization problems [36] like nonlinear parameter approximation in chemical processes. Jayachitra and Vinodha [37] added that the GA employed the rules of probability transition to handle generalized population of chromosomes which evolved through a series of iterations generations, pioneered by fitness tests, cross over, and mutation. GA takes four simple steps which are the population generalization, fitness selection, crossover, and mutation approach respectively to arrive at the optimum solution, and when the result does not converge, the output is feedback for another fitness test. Parameters used for the GA updates and computations are in Table 2; while the pseudocode is in Algorithm 1.

The GA in Algorithm1, presents the traditional GA [38] for the optimization of PLC, however, this algorithm suffers among many limitations the issues of pre-mature convergence. This usually occurs when there is not enough search space for the algorithm to explore. It can also happen when there is not enough diverse between the mutation and crossover operation of the chromosomes or if the size of the chromosomes is very small [39]. To address this issue, the study proposed a novel counter-premature algorithm.

Table 2. Parameters of the GA

Parameter	Values	Parameter	Values
Population size	8000	Cross over operator	Due point with probability (P = 0.8)
Representation	Mixed binary real	Mutation operator	uniform
Initialization	Random	Probability	0.01
Scale factor	(5, 20)	Proportional coeff.	0, 10 xmax ($\lvert u_{min}\rvert, \lvert u_{max}\rvert$)

Algorithm 1. GA pseudocode

1) Start
2) Initiate the random population size of the variables in the exothermic reaction = 8000
3) Set a reference standard for temperature and concentration
4) Perform computation test with reference standards using the fitness model
5) Get new offspring
6) Generate new population
7) Crossover sample
8) Mutation
9) Do until
10) Best offspring is determined
11) Generate best PLC control functions
12) Return
13) End

This CPA is tailored towards optimizing the traditional GA (in Algorithm1) to address issues of pre-mature convergence associated with GA, which might impact its reliability as a PLC optimizer. The algorithm begins by optimizing the population size of the chromosomes using a multiple population algorithm, as referenced in [40]. Initially, the population size is denoted as P, and the desired increase in population is represented as ΔP. The new population size is determined by adding ΔP to the initial population, resulting in P+ΔP. Next, the technique adjusts the crossover and mutation operations using a probability function that generates values between 0 and 1. The output of this probability function is utilized to adapt the crossover and mutation rates, enabling the control of population diversity. To select the best outcome, the least square algorithm (LSA) referenced as [41] was applied.

LSA evaluates the probability function's output and identifies the best outcome, designated as N. This best outcome is recommended as the foundation for the next generation. To determine the Pareto-optimal solutions, a multi-objective evolution algorithm referenced as [42] was adopted to facilitate the discovery of solutions that simultaneously optimize multiple objectives while maintaining diversity. Furthermore, a local search operator was used to refine the best outcome obtained thus far. Through fine-tuning the algorithm's parameters, this local search operation aims to enhance the quality of the solution. Lastly, the refined best outcome serves as a basis for generating new offspring in the subsequent generation, thereby continuing the optimization process. In summary, the technique optimizes the population size, adjusts crossover and mutation rates using a probability function, selects the best outcome using LSA, applies multi-objective evolution to determine Pareto-optimal solutions, refines the best outcome through a local search operator, and generates new offspring. These steps collectively aim to improve the algorithm's performance and facilitate the discovery of optimal solutions for the traditional GA in Algorithm 1. The proposed CPA was presented as Algorithm 2. Algorithm 1 presents the traditional GA, while Algorithm 2 presents the proposed CPA. Collectively other algorithms were integrated as an improved GA for the optimization PLC for enhanced control of nonlinear in continuous stir tank reactor (CSTR) plant. The proposed GA was reported as Algorithm 3.

2.3.2. Neural network algorithm

To solve the control system problem of CSTR, feed-forward neural network (FFNN) [43], [44] was applied to optimize the PLC. The neural network was adopted from [45] and used to control the PLC. The neural network is a branch of ML that is inspired by the behavior of the human brain. The neurons have weights, biases, and activation functions. The neurons were configured considering the number of control parameters to determine the input and form of the network. The activation function was used to trigger the neurons to give output within the desired range based on the activation function type. In this case, the type considered is the tangent hyperbolic function, which produces output features within the range of -1 and 1 and is connected at the hidden layers of the neurons, and then the purelin activation function, which is connected at the output of the neurons. The reason for the multiple activation functions is to ensure variation of nonlinearities, which helps improve the training process. The neural network was trained with data collected from the CSTR model at a steady state using Table 3. The data contain CSTR behavior parameters such as inlet flow rate of reactants A and B, concentration rate of products A and B, coolant temperature, inflow temperature, and coolant flow rate. The output target value is concentration of B which is the outlet. The training process was done with gradient descent-based training algorithm [46].

During the training of the neural network with parameters in Table 4, mean square error (MSE) and Regression (R) were respectively used to measure the performance of the control laws. The MSE was used to measure the error that occurred during the training process, with the target value of zero and the R value of 1. The performance was validated using tenfold cross-validation technique, and the results are presented in Table 5. The result reported an average MSE of 0.03033e-10 and an R of 0.97614. The implications of the training result showed that the FFNN correctly learned the plant features and was also able to control dynamics correctly. The output produced with the FFNN training is the reference control law in Algorithm 4.

2.3.3. PID control function

The PID is one of the most used control functions of PLC optimization. The PID is made up of the integration of three mathematical functions which are the proportional, integral, and derivative functions respectively to form the control law. Each function compensates and helps adjust the gain of the other until a good approximation function is achieved for the plant constraints. The proportional function is presented using (11).

$$P = K_{P.error\ (t)} \tag{11}$$

Where Kp is the proportional gain; the integral term is presented (12).

$$I = K_i \int_0^t error(t)dt \tag{12}$$

Where $K_I = \frac{K_p}{T_I}$ is the integral gain, T_I is the integral time constant. The derivative function was presented as (13).

$$D = K_D \frac{derror(t)}{dt} \tag{13}$$

Where $K_D = \frac{K_p}{T_D}$ is the derivative gain. The relationship between the (8)-(10) was used to develop the PID controller as in (14).

$$G = K_P\left(1 + \frac{1 + T_I.T_D s^2}{T_I}\right) = K_p\left(1 + \frac{1}{T_I} + T_D s\right) \quad (14)$$

Algorithm 2. The CPA pseudocode
1) Start
2) Optimize chromosomes with multiple population algorithms
3) Let the initial population size be P, and the desired increase in population size be ΔP
4) The new population size is determined as P + ΔP
5) Adjusting crossover and mutation
6) The algorithm adjusts the crossover and mutation rates using a probability function between 0 and 1
7) The output of the probability function is used to regulate the crossover and mutation operations
8) This adjustment aims to influence the population diversity
9) Selecting the best outcome with LSA as (n)
10) Determine the pareto-optimal solutions with multi-objective evolution algorithm while maintaining diversity
11) Refining the best outcome with local search operator
12) Fine tune the algorithm for new offspring generation
13) Recommend the offspring
14) Return

Algorithm 3. Proposed GA
1) Start the control algorithm for the CSTR
2) Optimize the control parameters using a CPA in algorithm 2
3) Set the initial population size as P, and determine the desired increase in population size as ΔP
4) Calculate the new population size as P + ΔP
5) Adjust the crossover and mutation rates within the GA to enhance the exploration and exploitation capabilities of the algorithm
6) Use a probability function between 0 and 1 to regulate the crossover and mutation operations, aiming to influence the population diversity and improve the quality of the solutions
7) Select the best outcome using the LSA and designate it as N
8) Utilize a multi-objective evolution algorithm to determine the Pareto-optimal solutions while maintaining diversity among the solutions
9) Employ a local search operator to refine the best outcome obtained so far, aiming to further improve its quality and convergence properties
10) Fine-tune the algorithm's parameters and control settings to enhance the generation of new offspring
11) Recommend the offspring, which represents the next generation of control actions or set-points for the CSTR
12) Return to continue the iterations of the GA, iterating through steps 2-11 to further optimize the control of the CSTR

Algorithm 4. FFNN control function
1) Start
2) Load CSTR data at steady state
3) Split into training and test (80:20)
4) System identification as nonlinear auto regressive moving average
5) Configure neural network architecture
6) Activate neurons with tanh function at the input layer
7) Start gradient descent algorithm
8) Set MSE target ≈ 0
9) Start training neurons
10) Activate neurons with purelin activation function at the output layer
11) If MSE ≈ 0
12) Stop training
13) Generate reference neuro control function
14) Else
15) Adjust neurons
16) Do until
17) MSE ≈ 0
18) Apply step (12)
19) Stop training
20) End

Table 3. Steady state parameters of the CSTR

Parameters	Unit	Value	Parameters	Unit	Value
Volumetric flow rate	m^3/h	1.00000	Boltzmann's ideal gas constant	kcal/kgmol	1.98590
Reactor Volume	m^3	1.00000	Reaction heat	kcal/kgmol	-5960
Pre-exponential non-thermal factor	1/h	3.5562e+08	Capacity of heat density	m^3k	470.30
Activation energy	kcal/kgmol	11851.4	Heat transfer	kcal/k*h	145.101
Set point (T and C)	K and mol/m^3	311 and 11	Boltzmann's ideal gas constant	kcal/kgmol	1.98590

Table 4. FFNN training parameters

Parameter	Values	Parameter	Values	Parameter	Values	Parameter	Values
Hidden layer	7	Training samples	8000	Delay output	2	Max. interval (s)	20
Interval (s)	0.2	Max. plant input	3	Max. output	3	Min. interval (s)	5
Delay input	2	Min. plant input	0	Min. output	3	Training epochs	200

Table 5. Training and valid

S/N	MSE	Regression
1	0.002845e-10	0.9729
2	0.005423e-10	0.9752
3	0.004535e-10	0.9832
4	0.024165e-10	0.9539
5	0.048345e-10	0.9809
6	0.030245e-10	0.9817
7	0.04532e-10	0.9811
8	0.05287e-10	0.9837
9	0.03412e-10	0.9749
10	0.05542e-10	0.9739
Avg	0.03033e-10	0.97614

3. RESULTS AND DISCUSSION

The models of the CSTR and the three ACS algorithms used for the optimization of the PLC were tested in an experimental test made of Siemens PLC, laptop installed with Studo500 software, human machine interface, somatic manager software. The parameters in Tables 1-3 were used for the programming with the reference temperature set-point changes from 311-313(K) and concentration at 10-10.3 ($kgmol/m^3$). The performance of the plant was changed via the introduction of step change at various instances of the technical process, while the control algorithms were monitored considering overshoot and response time as the try to adapt and follow the reference set-point and perform disturbance rejection. The batch reactor presents the dynamic behavior of the two connected tanks whose concentration were modelled in the (1) and (2), temperature dynamic modeled in (3) and (4), and then volumetric flow rates of the flow modelled in the (5) and (6). To control the system, the variations between input and controlled variables are defined as errors $(x_1, x_{12}, x_{21}, x_{22}, x_{31}, x_{32})$. These errors are related to the input and controlled concentrations $(C_{AF}\ , C_{A2}^d)$, input and controlled temperatures $\left(T_F, T_F^d, T_{J2}, T_{J2}^d, T_1, T_1^d, T_{J1}, T_{J1}^d\right)$. The experimental setup used for the data monitoring of the batch reactor was presented in Figure 1.

Figure 1. The experimental setup

The experimental setup was used to monitor the thermodynamic process of the batch reactor plants. The objective was to use its uncertain parameters as in (4) as input to each of the control algorithms and then improve the PLC for better control of complex reactors. Table 3 was used for the testing parameters, in two different tests, while the result of test 1 comparative response of the algorithms (PID, Improved GA, ANN) using error temperature from the reactor as in (7) as input to control the plant and produce the stabilized controller output result in Figure 2 which was produced from (8) and also the controlled temperature for the three algorithms as modelled in (9) and reported in Figure 3. These control outcomes were achieved from the injected coolant in (10) which also produce the result in Figure 4.

Figures 2-4 present the result of the plant test with the three ACS developed to optimize the performance of the PLC. Figure 2 shows the control concentration of the plant which was achieved due to the temperature control response in Figure 3, using the coolant in Figure 4. The result showed that the three ACS all followed the reference set-point to control the variation in concentration in (1) and temperature variation in (2). The PID functions each approximated the control parameters and sum up the three computed outputs as the control function as modeled in (14) to approximate the plant. The improved GA in algorithm (3) on the other hand collects the population size of the plant using the CPA algorithm to optimize the population and address pre-mature convergence problem, then apply fitness test to generate new samples which converge and control the plants after series of mutation and crossover.

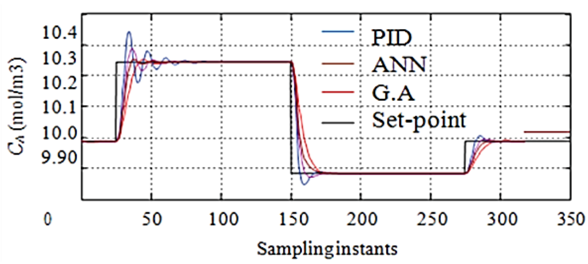

Figure 2. Controlled concentration

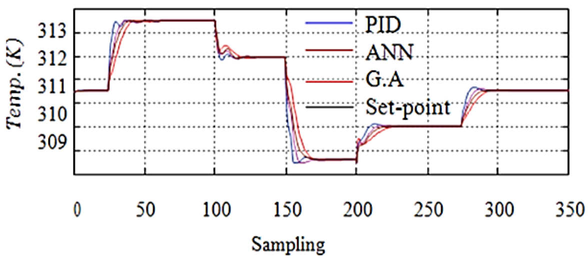

Figure 3. Controlled temperature

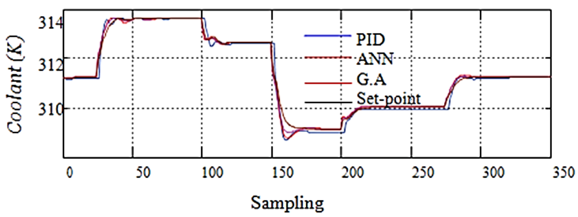

Figure 4. Coolant temperature

From the result, it was observed that the PID experiences overshot, while that of the GA was slight. The reason for the overshot in the GA despite the optimization with the CPA algorithm was due to the uncertain characteristics of the thermodynamic process which changes with time and may not be captured by the GA. Furthermore, the ANN was able to control the overshoot and achieved a better steady state with the plant. Similarly, at the point of step change of the various instances, the PID and G.A overshoots (see Table 6 for overshoot percentage), while trying to reject disturbance, also the ANN perfectly follows the reference set-point and controls the plant with a limited overshoot of 1.12%. Another test 2 was performed, setting the initialize temperature and concentration of the plant at 310.5-313.5(K) and concentration at 10-10.25 ($\mathrm{kgmol/m^3}$), while the temperature instances were varying at various steps of the technical process, to give room for the evaluation of the ACS algorithm used to optimize the PID. The results were presented in Figures 5-7, while more analysis was presented in Table 7.

From the result of the test result, it was observed that the variation of temperature, in the various instances affects the concentration of the plant, while the PID, GA, and ANN algorithms try to reject disturbance on the plant. The steady state was achieved via the injection of the coolant into the reactor. From the outcome, it was observed that the control algorithms followed a similar trend in the first test, with the PID and G.A experiencing overshoot, while ANN overshoot was minimal as in Table 6, also is the comparative response of test 2 in Table 7.

Tables 6 and 7 present the comparative analysis of the control algorithms tested on the nonlinear plant. The result showed that the FFNN-PLC achieved a better control response considering the overshot and response time to disturbance rejection when compared with the GA and PID counterparts. The reason was due to the intelligence of the neurons which understand the plant behavior and use the reference to track the set points. The overshot and delay experienced by the GA was due to the tie it takes to collect the chromosomes, perform fitness, and mutation until the desired control response is experienced. These results in delays in the plant, likewise the case of the PID where its individual P-I-D mathematical functions act on the constraints to reject disturbance and control the plant.

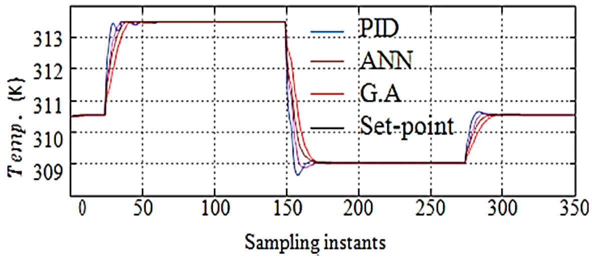

Figure 5. Controlled temperature

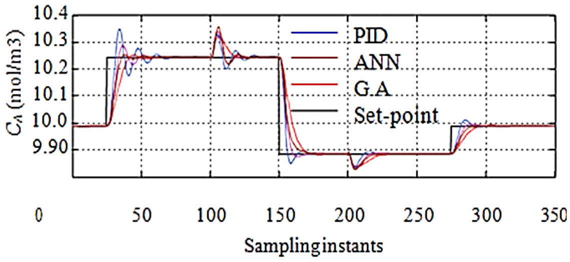

Figure 6. Controlled concentration

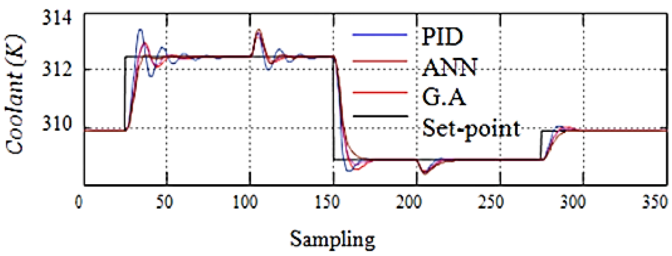

Figure 7. Coolant temperature

Table 6. Test 1 comparative response

Control laws	Overshoot (%)	Settling time (samples)
PID- PLC	31.25	23.13
GA- PLC	12.15	35.00
FFNN-PLC	1.12	8.75

Table 7. Test comparative response

Control laws	Overshoot (%)	Settling time (samples)
PID- PLC	35.17	27.00
GA- PLC	22.31	41.00
FFNN-PLC	3.00	10.30

- Data availability

The data used for this work is available at kaggle.com/datasets/eddardd/continuous-stirred-tank-reactor-domain-adaptation.

4. CONCLUSION

This research has successfully evaluated the impact of various control algorithms on the PLC and tested during complex technical processes, where multiple constraints are controlled. From the review of literatures, it was observed that many works have been presented which optimized PLC performance during control of nonlinear system, however this paper focused on extending this PLC functionality to more complex nonlinear systems, considering G.A, PID, and ANN respectively as the control algorithms tested separately on a complex thermodynamic process. From the result, it was observed that all three algorithms tried to follow the reference set-points and control the plant; however, their behavior during this process varies considering settling time and overshoot. The PID experiences overshoot and hence not recommended for the control of multi-variant dynamic systems, the improved GA recorded good control performance with limited overshoot but suffers delay training time The ANN achieved better response to disturbance identification and rejection when compared to the PID and the improved GA. Overall, it can be deduced that optimizing PLC it neural network control algorithm will provide the needed adaptive control functionality for the approximation of complex nonlinear systems.

REFERENCES

[1] H.C. Inyama and Chukwudi A., “A Survey of Controller Design Methods for A Robot Manipulator in Harsh Environments,” *European Journal of Engineering and Technology*, vol. 3, no. 3, pp. 301-329, 2015.
[2] B. Swapan, “Basics of hazard, risk ranking and safety systems,” *Plant Hazard analysis and Safety Instrumentation System*, pp. 1-81, 2017.
[3] L. Fortuna, A. Buscarino, “Automatic Control and System Theory and Advanced Applications - Volume 2,” *Inventions*, vol. 9, no. 1, 2024, doi:10.3390/inventions9010005.
[4] L. Suo, J. Ren, Z. Zhao, C. Zhai, “Study on the Nonlinear Dynamics of the Continuous Stirred Tank Reactors,” *Processes*, vol. 8, no. 11, 2020, doi*:* 10.3390/pr8111436.
[5] K. Pablo, S. Nilay, and L. Daniel, “PLC implementation of real-time embedded MPC algorithm based on linear input/output models,” *IFAC Paper Online*, vol. 53, no. 2, pp. 6987-6992, 2020.
[6] M. Ibrahim, R. Elhafiz, “Security Assessment of Industrial Control System Applying Reinforcement Learning,” *Processes*, vol. 12, no. 4, 2024, doi:10.3390/pr12040801.
[7] A. Ulagwu-Echefu, I.I. Eneh, U.C. Ebere, “Mitigating the effect of latency constraints on industrial process control monitoring over wireless using predictive approach,” *International Journal of Research and Innovation in Social Science*, vol. 5, no. 1, pp.82-87, 2022.
[8] P. Oscar and A. Luis, “Integration of design and NMPC based control for chemical processes under uncertainty: An MPC-based framework,” *Computer and Chemical Engineering*, vol. 162, no. 107815, 2022, doi: 10.1016/j.compchemeng.2022.107815.
[9] Y. Liu, J. Hu, and Y. Li, “Quantized Formation Control of Heterogeneous Nonlinear Multi-Agent Systems with Switching Topology,” *Journal of Systems Science and Complexity*, vol. 36, no. 1, pp. 2382–2397, 2023, doi:10.1007/s11424-023-2387-2.
[10] P. Airikka, “Advanced control methods for industrial process control,” *Computing and Control Engineering*, vol. 15, no. 3, pp. 18-23, 2004, doi: 10.1049/cce:20040303.
[11] I. Muresan, I. Birs, C. Ionescu, E. Dulf, and R. De Keyser, “A Review of Recent Developments in Autotuning Methods for Fractional-Order Controllers,” *Fractal and Fractional*, vol. 6, no.1, 2022, doi: 10.3390/fractalfract6010037.
[12] J. Tarnawski, P. Kudełka, and P. Korzeniowski, “Advanced Control With PLC—Code Generator for a MPC Controller Implementation and Cooperation With External Computational Server for Dealing With Multidimensionality, Constraints and LMI Based Robustness,” *IEEE Access*, vol. 10, pp. 10597-10617, 2022, doi: 10.1109/ACCESS.2022.3142054.
[13] C. Hang, K. Åström, and Q. Wang, “Relay feedback auto-tuning of process controllers–a tutorial review,” *Journal of process control*, vol. 12, no. 1, pp. 143–162, 2002.
[14] E. Priyanka, C. Maheswari, and B. Meenakshipriya, “Parameter monitoring and control during petrol transportation using PLC based PID controller,” *Journal of Applied Research and Technology*, vol. 14, no. 2, 2016, 10.1016/j.jart.2016.03.004.
[15] Y. Mfoumboulou and R. Tzoneva, “Development of a Model Reference Digital Adaptive Control Algorithm for a Linearized Model of a Nonlinear Process,” *International Journal of Applied Engineering Research*, vol. 13, no. 23, pp. 16662-16675, 2018.
[16] N. Saraf and A. Bemporad, “Fast model predictive control based Online input/output models and bounded variable least squares,” in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 1919–1924, doi: 10.1109/CDC.2017.8263930.
[17] D. Lulia and I. Sergiu, “A fuzzy PLC control system for a servomechanism,” *IFAC Proceedings Volumes*, vol. 43, no. 22, pp. 69-74, 2010, doi: 10.3182/20100929-3-RO-4017.00013.
[18] J. Jiri, K. Jiri, and P. Miroslav, “Implementation of fuzzy logic control based on PLC,” in *ETFA2011*, 2011, pp. 1-8, doi: 10.1109/ETFA.2011.6059049.
[19] K. Onur, Y. Engin, G. Mujde, and E. Ibrahim, “Implementation of a new self-tuning fuzzy PID controller on PLC,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 12, no. 2, pp. 1-10, 2005
[20] L. Jeydson, “Fuzzy logic control with PSO tuning,” in *Fuzzy Systems - Theory and Application*, by C. Volosencu, 2021,doi: 10.5772/intechopen.96297.
[21] M. Khosravi, *et al*., “Performance-Driven Cascade Controller Tuning with Bayesian Optimization,” *IEEE Transactions on Industrial Electronics*, 2021, doi: 10.1109/TIE.2021.3050356.
[22] A. Mohamed, K. Hamdy, M. Hatem, E. Elattar, H. Abdel, and A. Dina, “Data Driven Prognostics from Machine Learning to Deep Learning: A survey,” Research Squre, 2022, doi: 10.21203/rs.3.rs-1952441/v1.
[23] J. Bhargav, “Application of Neural Network On PLC-based Automation Systems For Better Fault Tolerance And Error Detection,” Thesis, Auburn University, 2019.
[24] G. Valencia-Palomo, R. Hilton, and A. Rossiter, “Predictive control implementation in a PLC using the IEC 1131.3 programming standard,” in *2009 European Control Conference (ECC)*, 2009, pp. 1317-1322, doi: 10.23919/ECC.2009.7074588.
[25] M. Pereira, D. Limon, P. Munoz, and T. Alamo, “MPC implementation in a PLC based on Nesterov's fast gradient method,” in *2015 23rd Mediterranean Conference on Control and Automation (MED)*, 2015, pp. 371-376, doi: 10.1109/MED.2015.7158777.
[26] B. Kaepernick and K. Graichen, “PLC implementation of a nonlinear model predictive controller,” in *IFAC Proceedings Volumes*, Cape Town, South Africa, 2014, pp. 1892-1897, doi: 10.3182/20140824-6-ZA-1003.00911.
[27] Z. Zhao, Y. Li, T. Salsbury, and J. House, “Global Self-Optimizing Control With Data-Driven Optimal Selection of Controlled Variables With Application to Chiller Plant,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 2, p. 021008, 2018, doi: 10.1115/1.4052395.
[28] V. Arturo, R. Oscar, R. Germán, V. Guillermo, “Implementation of an intelligent algorithm in a PLC,” in *Conference: 6to.Congreso Internacionalen Ciencias Computacionales CiComp 2013*, Ensenada, México, 2013.
[29] A. Tamer and S. Khaled, “Model Following Control of SISO Nonlinear Systems using PID Neural Networks,” *International Journal of Computer Applications*, vol. 140, no. 10, pp. 12-18, 2016.
[30] Y. Lee and D. Jang, “Optimization of Neural Network-Based Self-Tuning PID Controllers for Second Order Mechanical Systems,” *Applied Sciences*, vol. 11, no. 17, p. 8002, 2021, doi: 10.3390/app11178002.
[31] P. Patryk, T. Thanh, C. Jacek, N. Pawel, K. Tomasz, and G. Bogdan, “Application of Machine Learning to Performance Assessment for a class of PID-based Control Systems,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 7, pp. 1-30, 2021, doi: 10.1109/TSMC.2023.3244714.
[32] P. Ashwad and H. Bipin, “Online tuning of PID controller using black box multi-objective optimization and reinforcement learning,” *IFAC-Papers OnLine*, vol. 51, no. 32, pp. 844-849, 2018, doi: 10.1016/j.ifacol.2018.11.440.
[33] R. Eberhart and Y. Shi, “Particle Swarm Optimization: Developments, Applications and Resources,” in *Proceedings of the Congress on Evolutionary Computation*, vol. 1, pp. 81-86, 2001, doi: 10.1109/CEC.2001.934374.

[34] H. Holland, *Adaptation in natural and artificial systems*, University of Michigan Press: Ann Arbor, 1975.
[35] Li. D.-J, "Adaptive neural network control for continuous stirred tank reactor process," *IFAC Proceedings*, vol. 46, no. 20, 171-175, 2013, doi: 10.3182/20130902-3-CN-3020.00148.
[36] C. Yogesh, "Design and Implementation of Intelligent Controller For A Continuous Stirred Tank Reactor System Using Genetic Algorithm," *International Journal of Advances in Engineering & Technology*, vol. 6, no. 1, 2013.
[37] A. Jayachitra and R. Vinodha, "Genetic Algorithm Based PID Controller Tuning Approach for Continuous Stirred Tank Reactor," *Advances in Artificial Intelligence*, vol. 1, 2014, doi: 10.1155/2014/791230.
[38] M. Kotyrba, E. Volna, H. Habiballa, and J. Czyz, "The Influence of Genetic Algorithms on Learning Possibilities of Artificial Neural Networks," *Computers*, vol. 11, no. 5, 2022, doi:10.3390/computers11050070.
[39] E. Simona, "Mechanisms to Avoid the Premature Convergence of Genetic Algorithms" *Petroleum-Gas University of Ploiesti Bulletin, Mathematics-Informatics-Physics Series*, vol. 61, no. 1, pp. 87–96; 2009.
[40] X. Shi, W. Long, Y. Li, D. Deng, "Multi-population genetic algorithm with ER network for solving flexible job shop scheduling problems," *PloS one*,2020, doi: 10.1371/journal.pone.0233759.
[41] R. Olympia, F. Stefta, and M. Paprzycki, "The influence of population size on the genetic algorithm performance in the case of cultivation process,"in *2013 Federated Conference on Computer Science and Information Systems*, 2013, pp. 371–376.
[42] S. Sharma and C. Vijay. "A Comprehensive Review on Multi-objective Optimization Techniques: Past, Present and Future," *Archives of Computational Methods in Engineering*, vol. 29, 2022, doi: 10.1007/s11831-022-09778-9.
[43] L.O. Nwobodo and H.C. Inyama, "Neuro-Fuzzy Model for Strategic Intellectual Property Cost Management," *International Journal of Computer Applications Technology and Research*, vol. 4, no. 7, pp. 574-578, 2015.
[44] U.C Ebere M.C. Harbor, I.I. Eneh, "Precision Control of Autonomous Vehicle Under Slip Using Artificial Neural Network," *International Journal of Research and Innovation in Applied Science (IJRIAS)*, vol. 6, no. 9, pp. 51-55, 2021.
[45] T.C Asogwa, "Improving the intelligent control of magnetic levitation ball using artificial neural network," *International Journal of Engineering and Computer Science*, vol. 8, no. 6, pp.24679-24685, 2019.
[46] A. Alexander, MIT 6.S191 (2020): Introduction to Deep Learning, 2020. [Online]. Available: https://www.youtube.com/watch?v=njKP3FqW3Sk

BIOGRAPHIES OF AUTHORS

Sochima Vincent Egoigwe holds a B.Eng. in Electronics Engineering from the University of Nigeria, Nsukka, Enugu State, an M.Eng. from Enugu State University of Science and Technology (ESUT) in Electrical and Electronic Engineering, and a Ph.D. at Enugu State University of Science and Technology (ESUT). He is duly registered as an engineer. He can be contacted at email: sochima.egoigwe@unn.edu.ng.

Asogwa Tochukwu Chijindu is a former head of the Department of Computer Science, at Enugu State University of Science and Technology (ESUT). He holds a B.Eng. in Computer Science and Engineering, ESUT in 2001, a master's degree in computer science in 2011, and a Ph.D. in Computer Science in 2021. He has so many years of academic experience. He is a member of Computer Professionals of Nigeria (CPN). His research interest is on artificial intelligence. He has so many international and local publications. He can be contacted at email: tochukwu.asogwa@esut.edu.ng.

Lois Onyejere Nwobodo is a seasoned computer scientist and engineer, with a wealth of experience in both industry and academics. She obtained her bachelor's degree in computer engineering from Enugu State University of Science and Technology (ESUT), She went to Nnamdi Azikiwe University (NAU), where she got her Ph.D. and M.Eng. in Electrical/Electronics and Computer Engineering and majored in Computer and Control Systems. Her wealth of experience has been made manifest in numerous journal publications and conference proceedings she has written. She is duly registered professionally as both a computer scientist and computer engineer. She can be contacted at email: lois.nwobodo@esut.edu.ng.

Onuigbo Chika Martha obtained her B.Eng., M.Eng., and Ph.D. in Electrical and Electronics Engineering from Enugu State University of Science and Technology, Nigeria. She is currently a senior lecturer in Communication Electronic Engineering at the same university. She is equally an examiner and facilitator in Computer Science within the Faculty of Science at the National Open University of Nigeria (Enugu Study Centre). She has made many publications to both local and international journals. She can be contacted at email: onuigboch@gmail.com.

Frank Ekene Ozioko holds Ph.D. and a senior lecturer in the Computer Science Department at ESUT. His main research interests include artificial intelligence, the internet of things, computational intelligence, and autonomous systems. He has so many international and local publications. He has attended many conferences where he presented papers. He can be contacted at email: ekene.ozioko@esut.edu.ng.

Ozor Godwin Odozo holds Ph.D. and a senior lecturer in the Department of Computer Engineering, at Enugu State University of Science and Technology. His research interests include cyber-physical systems, software engineering, internet of things, control system engineering, energy, and artificial intelligence. He has published in several peer-reviewed and indexed journals both locally and internationally. He has attended many conferences where he presented papers. He can be contacted at email: ozor.godwin@esut.edu.ng.

Ebere Uzoka Chidi holds a PGD (Electrical/Electronic), M.Eng. (Control and Instrumentation) and currently doing his Ph.D. at the Department of Electrical Electronics Engineering, University of Nigeria, Nsukka. He has published in several indexed journals both locally and internationally. His research interests include control systems, robots, deep learning, internet of things, and power systems. He can be contacted at email: uzoka.ebere22@acespedunn.edu.ng.