Performance Evaluation of An Optimized PID Controller for Networked Control System Using Ant Colony Optimization

CHUKWUEBUKA O. EKE-NNODU[1], CHIDIEBELE C. UDEZE[2], CHUKWUDI CHUKWUDOZIE[3], MMASOM I. NDEFO[4], HENRY I. UGWU[5]

[1, 2, 3, 4, 5] *Electronic Engineering Department, University of Nigeria Nsukka, Nigeria*

Abstract-* The potentials of controlling systems remotely led researchers to develop systems comprising of networks and control systems. Such systems have been known by various names, but it is lately referred to as Networked Control System (NCS). The inherent challenges with communication networks such as latency, packet drop/loss, congestion etc. creates major challenges for such system especially in the transmission of control signal from one component to another in real time applications. Proportional Integral Derivative (PID) controller is used to estimate factors which can cause delay during data transfer among system components. This work compensates this delay with the use of Ant Colony Optimization (ACO) technique to improve the parameters of the PID controller such that performance of the NCS will be improved. The work will evaluate the performance of the NCS after ACO technique has been used to tune the parameters of PID and compare it with the performance of NCS when fuzzy tuned PID controller and PID controller are used. The performance obtained by simulation shows that the performance of the ACO technique performed better than the fuzzy tuned PID controller and PID controller. The simulation of the performance was done using Matlab/Simulink 2016b and TrueTime toolbox.

Indexed Terms-* Ant Colony Optimization (ACO), Networked Control System (NCS), Optimization, Proportional Integral Derivative (PID).

I. INTRODUCTION

As a result of the quest to remotely control systems in real time, researcher's interest has been channelled to this area. This interest dates back to 1980s when such systems were referred to as "integrated communication and control networks" [1]. Over the years the name has been changed a few times but lately it is referred to Networked Control System (NCS). It can also be described as a system whose control loops are closed through a communication network such that both control and feedback signals can be exchanged among system components such as sensors, actuators, controllers, plants [2]. This system has advantages such as efficient cost reduction, installation ease, easy reconfiguration (flexibility), easy of upgrade to the system, reduced maintenance time and cost. NCS can be easily altered if some new system parts need to be introduced or damaged system parts needs to be replaced without causing any major change to the overall system structure, as a result of these advantages NCS has found a wide range of application in smart industries, autonomous traffic systems, tele-operations etc. In as much as NCS has the advantages some challenges have been encountered. These challenges are mainly because of the network-imposed resource constraints that conflicts with the communication network requirements of the networked control systems and as result of these challenges the performance and stability of the control system can be greatly impaired. These challenges include limited bandwidth, network induced delay, congestion, packet drop/loss, sampling, quantization and data collision.

Research works have proposed ways to improve the performance of the system by solving or managing these challenges. Heuristic optimization techniques have been proposed as one of such ways by various researchers. These optimization techniques have found applications in controllers of various systems which helps improve the performance of such systems [3].

For instance, Elamin A.Y et al in [4] used PSO algorithm to optimally tune the parameters of the Generalized Predictive Controller (GPC) when handling transmission delays in NCS. PSO algorithm was also used by the authors of [5] as a tuning tool in the implementation of PID controller in NCS. Hamidi J in [6] proposed that by selecting appropriate weighting matrices for designing optimal controller using PSO, we obtain a more sufficient and robust controller than when Genetic algorithm is used in designing the controller.

Authors of [7] worked on SGA (Search Group Algorithm) to improve its ability to solve optimization problem. They employed fuzzy logic in creating an adaptive parameter control for improving the quality of solution a basic SGA can provide. Just like other researchers Y. Tang et al in [8] proposed a novel method of tackling the network induced delays for Multiple Input Multiple Output Networked Control System (MIMO NCS). They introduced a new smith predictive fuzzy immune PID algorithm that helps reduce the negative effect that network induced delay has on NCS in general, but they focused on multiple input multiple output networked control system. The authors demonstrated the efficiency of the proposed algorithm using numerical examples with Ethernet protocol. Delay in each data transmission will always vary due to the size of data or the conditions of the network during a transmission. The varying network induced delay in communication channel between the sensor to controller and controller to actuator severely impacts the NCS. The authors of [9] proposed an idea of compensating the varying communication delay in the communication channels by using the Thiran approximation technique to design a sliding surface. A discrete time sliding mode control law was derived using the proposed surface to compensate the fractional time stability of the closed loop system. The authors used their simulation results to support their proposed strategy.

ACO algorithm has also shown great promise, hence its wide range of application. It has been applied in solving problems of traveling salesmen, graph coloring, routing of vehicles and scheduling. Tawfeek M et al in [10] applied ACO in task scheduling in cloud computing. A scheduling policy was developed based on ACO algorithm and compared against scheduling policies like First Come First Serve (FCFS) and Round Robin (RR). M. J Blondin et al in [11] used ACO algorithm in designing an optimal tuning method for an Automatic Voltage Regulator (AVR) system. This tuning method combined ACO algorithm and Nelder-Mead method in tuning the PID controller for the AVR system and the results obtained shows higher computational efficiency to other optimization algorithms. H. El-Sayed Ahmed Ibrahim et al, authors of [12] agreed that the PID controller is simple and most effective for different kinds of engineering control applications. However, this controller faces the challenge of adequate tuning. The conventional Ziegler-Nichols tuning method has shown poor performance when tuning a controller for a nonlinear system. The authors used Genetic Algorithm and ACO to tune the PID controller for the Switched Reluctance Motor (SRM) because of its high nonlinear characteristics and the results showed that the performance of such a system was greatly improved when compared with Ziegler-Nichols tuning method. ACO was used to optimize the speed control of a DC motor in [13]. Tri Kuntoro Priyambodo et al in their work [14] used ACO to optimize the vertical moving control of an Unmanned Aerial Vehicle (UAV) system by fine-tuning a PID controller.

The remainder of this work is partitioned as follows; section 2 briefly describes the ACO algorithm and how it can be applied to a PID controller. Section 3 shows the block diagram of the proposed NCS system, and the parameters used in simulating the system. We also show the pictorial diagram of the model in a Matlab/Simulink environment and the results of the different PID controlled NCS and section 4 will show the performance analysis of the ACO tuned PID controlled system and fuzzy logic tuned PID and PID controlled system. Conclusion and recommendation for future works will be done in section 5.

II. ACO ALGORITHM AND TUNING OF PID CONTROLLER

Most heuristic algorithms are based on the natural behavior of living organisms. Inventors of these algorithms studied the behavior of these natural organisms and were able to create algorithms which are mostly used for optimization purposes. Macro Dorigo, the inventor of the Ant Colony Optimization

(ACO) algorithm [15] was inspired by the behavior of actual ants. He observed that ants are social insects that exist in colonies and as such were able to locate the easiest pathway between their habitat and food sources. The ants can communicate their path amongst themselves by dropping a chemical substance called pheromone. Pheromone is an odorous chemical substance that real ants deposit and can perceive while walking about. Ants can trace their best path by the quantity of pheromone along a particular path. When a path is favorable, a lot of ants tend to follow that path thereby dropping more pheromone whereas if a path is not favorable ants avoid that path. This causes the quantity of pheromone on that path to reduce.

The inventor of ACO algorithm having studied the behavior of the ants used it in creating the ACO algorithm which has helped in finding shortest paths and solving optimization problems. ACO algorithm can efficiently search the best solution to an optimization problem owing to the self-organizing behavior that ants exhibit in choosing a path. The artificial ants in this ACO algorithm are equipped with extra capabilities that enable solving optimization problems quite easy. The finite size of the artificial ants is one of such capabilities. It is easier to find optimum solution to an optimization problem when several ants of finite size work together. The ants do not interact directly but indirectly during the quest for a solution by adding pheromones to the environment. Just like natural ants, the amount of pheromone deposit is proportional to the quality of the move the ants have made hence more pheromone means a better solution was found. The pheromone deposit is proportional to quality of solution as per local pheromone updating rule. Once iteration is complete the amount of pheromone is updated again as per global updating rule. The execution of the ACO algorithm is best explained by the flow chart below.

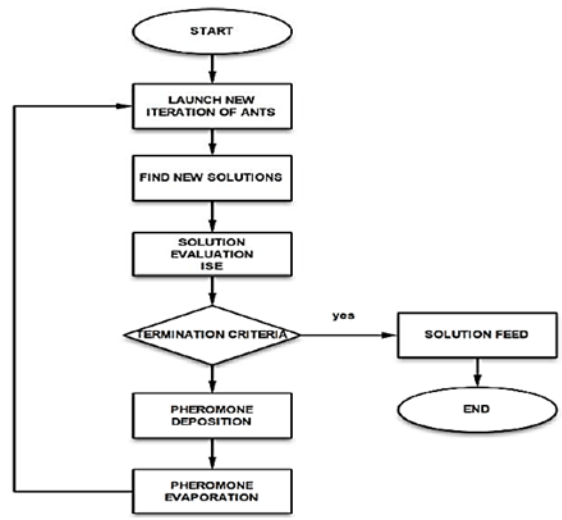

Fig 1: A flowchart for ant algorithm [13]

The solutions developed by all ants are updated in the global pheromone through the evaporation phase and reinforcement phase.

Evaporation phase: This is where a fraction of the pheromone evaporates meaning that the optimal solution is not found in that path.

Reinforcement phase: Where each ant deposits an amount of pheromone which is proportional to the fitness. This process is repeated until stopping criteria is met. This happens because an optimal solution has been found in that iteration.

According to literature, Proportional Integral Derivative (PID) controller is widely used in feedback control system because of the simplicity in the structure, its reliability, performance and cost [16]. The PID controller continuously calculates the error value as the difference between the calculated process variable (PV) and the target set points (SP). The controller tries to eliminate the error value by adjusting its parameters and other variables. The ACO algorithm helps in adjusting the parameters by picking the optimal values of the parameters.

PID controller is generally represented by equation below

$$G_C[s] = K_p + K_i/s + K_d s \quad (1)$$

Where the proportional, integral and derivative gains are represented by K_p, K_i, and K_d, respectively.

Therefore, the transfer function of the PID can be written as:

$$G_C[s] = K_p|1 + 1/|T_i s| + T_d s| \quad (2)$$

where the integral and derivative time constants are respectively represented by T_i and T_d. These terms are related as:

$$T_i = \frac{K_p}{K_i} \text{ hence } K_i = \frac{K_p}{T_i} \quad (3)$$

$$T_d = \frac{K_d}{K_P} \text{ hence } K_d = K_p * T_d \quad (4)$$

Substituting the values of K_i and K_d in equation (1) gives equation (2)

PID controller is expressed in its discrete-time equivalent as:

$$u(k) = K_P e(k) + K_i . T_s \sum_{i=1}^{n} |e(i)| + \frac{K_d}{T_s} \Delta e(k) \quad (5)$$

The control signal which is the input is represented with $u(k)$, while the error which results due to the difference between reference and the process output is represented by $e(k)$. The controller sampling period is represented by T_s. The error which results due to the difference between the reference and the process output can be approximated as:

$$\Delta e(k) = e(k) - e(k-1) \quad (6)$$

The measure by which a PID controller samples the error fed from a process is greatly dependent on the manipulation of the parameters K_P, K_i and K_d, thereby establishing optimum values of these parameters is important.

Tuning PID parameters can be done through conventional methods and intelligent optimization algorithms. To tune the parameters of the PID controller, optimal values of K_p, K_i and K_d have to be obtained from the many potential values between the minimum and maximum values. The potential values between minimum and maximum values are equally distributed into three parts between the upper and lower bounds. Finding the optimal values that give a good performance is the problem which is why intelligent optimization algorithms are developed and used. This work proposes the use of ACO algorithm. The ants tour the combination of values in the three parts which are arranged in three columns namely K_p, K_i and K_d where each potential value is represented by a node. The node toured by the ant is selected as the value of the parameter for each set of parameters. Ants select the next node to visit through a stochastic mechanism when building a solution. The probability of selecting the J^th^ node of the I^th^ parameter is given by equation 7

$$P_{ij}^{k} = \frac{\tau_{ij}^{a} * \eta_{ij}^{\beta}}{\sum_{l=1}^{n} \tau_{ij}^{a} * \eta_{ij}^{\beta}} \quad (7)$$

Where the parameter is represented by i= 1, 2, 3 and node represented by j= 1,2,3…..n. τ_{ij} is the pheromone. The parameters α and β control the relative importance of the pheromone versus the heuristics information η_{ij}
η_{ij} = 1/nodes ij where i=1,2,3 and j=1,2,3…..n

The optimal values gotten from tuning the controller produces a control signal that is sent to the actuator and the actuator applies the commands in the signal to the plant. One benefit of this algorithm is that after an iteration, the search space is reduced, which makes it easier for the artificial ants to cover the search space. The search space is referred to as FSS (Feasible Search Space) which is reduced by β and uses previous iteration information. All ants randomly search for the best solution to a given problem within the FSS at the beginning of the first iteration and are compared with the old ant colony solution that was generated at the initialization stage. After that, the pheromone amount is changed. The new ant colony is formed in the solution phase using Equation (8) and (9) are based on the best solution from the old ant colony. After which, the two colonies' best solutions are compared. An optimal solution is found as ants find their routes in the limited space. Assuming the number of ants being associated with m random initialization vectors is x^k, k =1, 2, 3, ...m, the solution vector for each ant is modified using the expression

$$X_t^{k(new)} = X_t^{k(old)} \pm \quad (8)$$
$$(t = 1, 2, \ldots, I)$$

Where $X_t^{k(new)}$ is the solution vector of the k^th^ ant at iteration t, $X_t^{k(old)}$ is the result gotten from a former iteration, and α is a vector given arbitrarily to find the length of jump. Ant vector $X_t^{k(new)}$ obtained at t^{th} iteration in (8) is found by using the value of same ant obtained from former iteration. The positive (+) sign in equation (8) is applied when point X_t^k is to the left of the best solution on the x coordinate axis and the negative (-) sign is applied when point X_t^k is to the right of the best solution on the same axis. The focus of search is defined by equation (9)

$$\overline{x}_t = x_t^{best} + \left(x_t^{best} * 0.01\right) \quad (9)$$

If $f\left(\overline{x}_t^{best}\right) \leq f\left(x_t^{best}\right)$, (+) sign is used in equation (8) otherwise, (-) sign is used. The (±) sign defines the direction of the search to reach the global optimum. The length of the jump is defined by the alpha (α) value, and it will be decreased slightly in attempt not to cross the global optimum. As the ants generated in the old colony are removed at the end of each cycle, a new ant colony is formed. Quantity of pheromone (τ) is reduced to simulate the evaporation process of real ant colonies using equation (10) in the pheromone update phase. After reducing of the quantity of pheromone, it is updated using (11). Quantity of pheromone only intensifies around the best values. This procedure is repeated till the desired number of iterations is completed.

$$\tau_t = 0.1 * \tau_{t-1} \quad (10)$$

$$\tau_t = \tau_{t-1} + 0.01 * f\left(x_{t-1}^{best}\right) \quad (11)$$

III. PROPOSED NCS SYSTEM WITH OPTIMIZED PID CONTROLLER

The system is designed and built using the TrueTime toolbox and Matlab/Simulink software. The ACO algorithm was written in Matlab .m file. The block diagram of the proposed system is shown in figure 2 below.

Figure 3 shows the Simulink design of the proposed system where the PID controller is tuned. The proposed system is divided into three nodes namely network node, controller node, plant node which encompasses the sensor and actuator. The sensor node collects data for the plant at periodic intervals configured at the sensor trigger block and forwards the collected data through the network block to the controller node. The computed signal from the ACO tuned PID controller is forwarded back to the actuator node through the network node. The actuator executes the computed signal on the plant and tries to attain the desired reference signal introduced at the controller node. Periodic and non-periodic signal are produced by the interference node which acts as the traffic in the network. The traffic in a network needs to be properly managed because it can cause serious problems in the network and in this case the system.

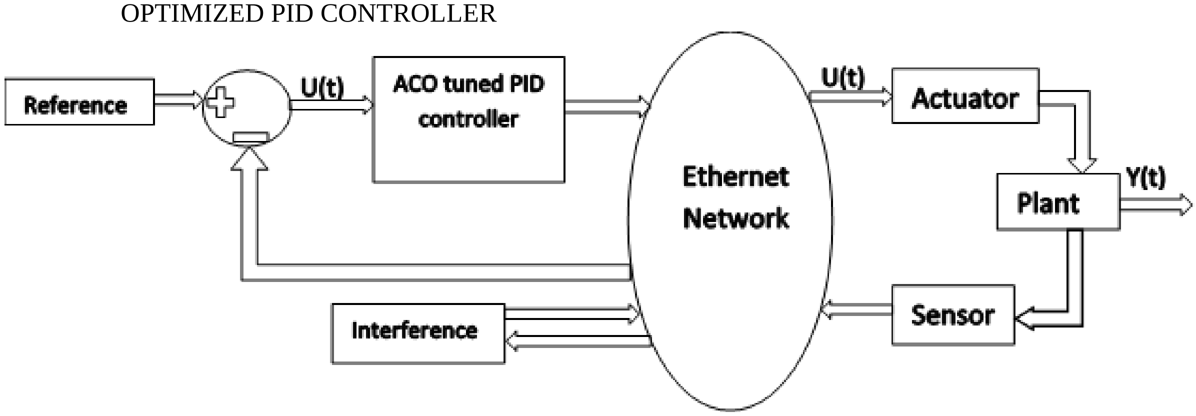

Fig 2: Block diagram of proposed system

The interference node introduces traffic of 10%. The goto block is used to interconnect some blocks so the Simulink design is not messy. The parameters used in the ACO algorithm are as follows; maximum iteration-10, number of ants-60, α=0.8, β=0.2, evaporation rate is 0.7, number of parameters=3 (K_p, K_i and K_d), number of nodes for each parameter is 1000. The proposed system is designed with Simulink and one of the three plant transfer function used in [17] is adopted in this work. The transfer function of the plant used is

$$H(S) = \frac{1}{(s+1)(s+1)}$$

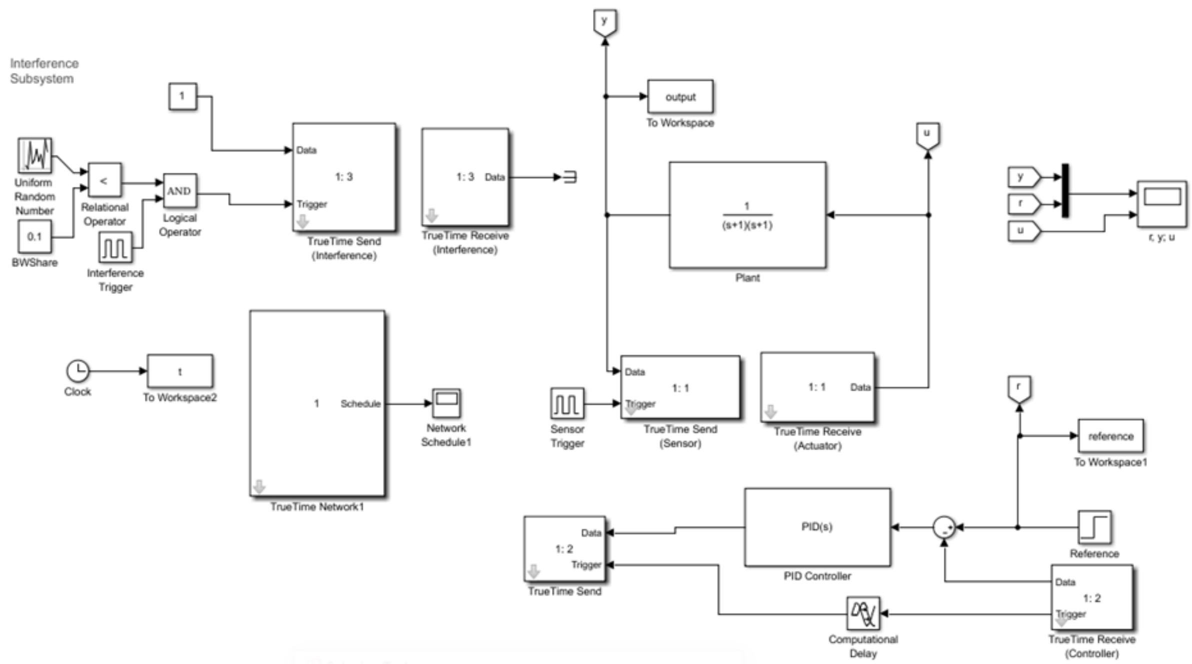

Fig 3: Simulink diagram of the proposed system

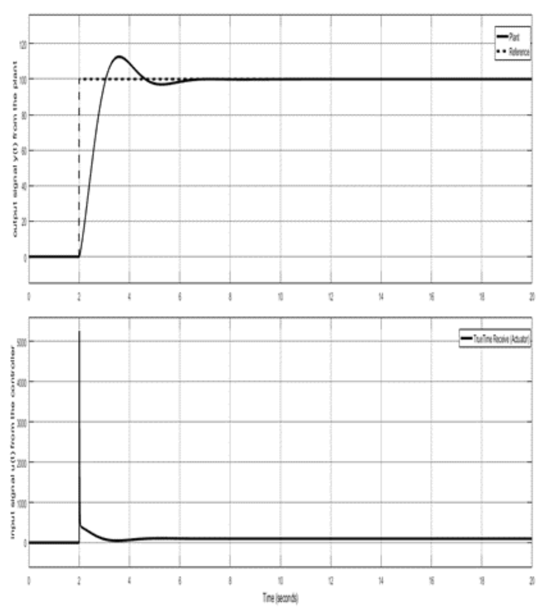

Fig 4: Simulation output of the proposed system

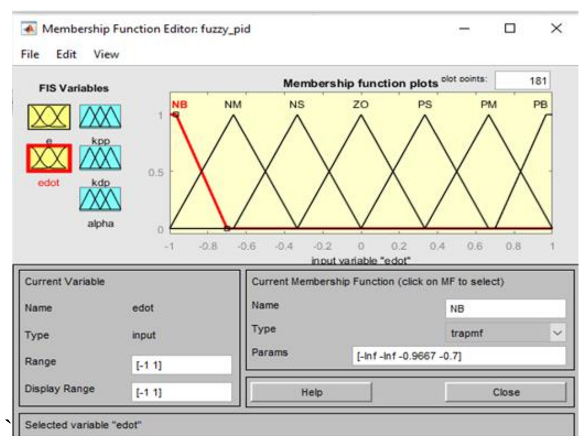

Fig 5: Membership function of Fuzzy PID

For the fuzzy-PID controller, the e and edot are assigned the input variable with a variation range of (-1,1). The memberships functions are 7 namely NB, NM, NS, ZO, PS, PM, PB. This uses triangular membership function for the output variables. Below are figures showing the membership function, fuzzy logic designer and fuzzy rules for the fuzzy tuned PID controller.

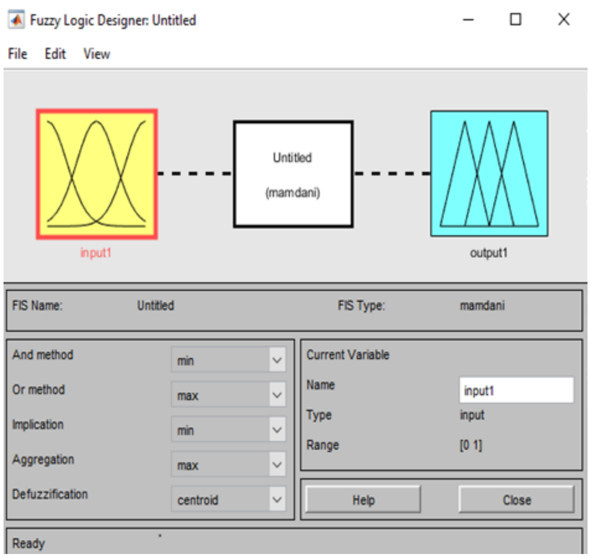

Fig 6: Fuzzy Logic designer

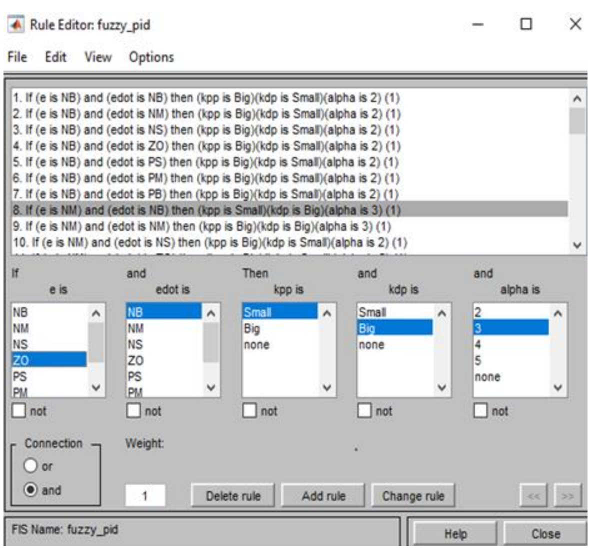

Fig 7: Fuzzy PID rule

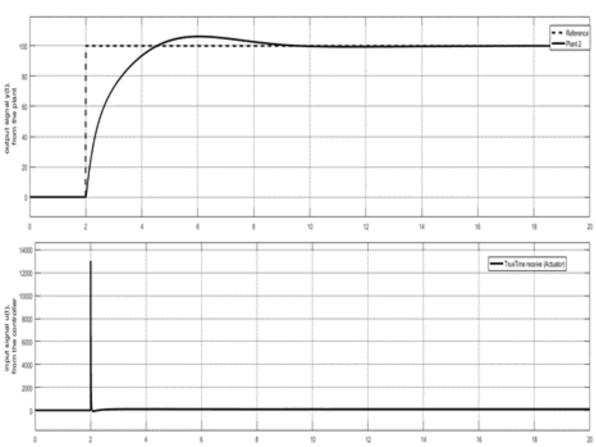

Fig 8: Simulation output of fuzzy-PID system

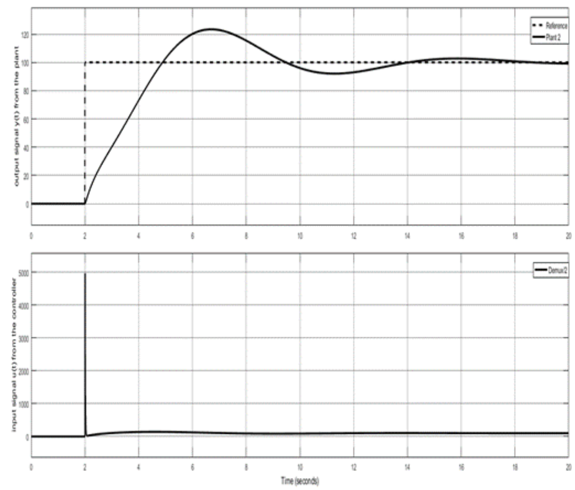

Fig 9: Simulation output of PID system

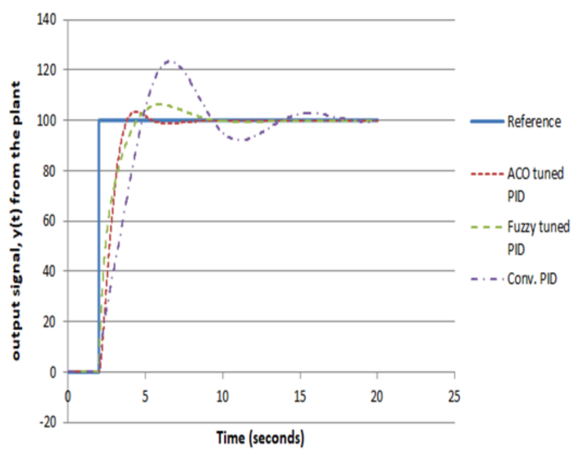

Fig 10: Comparison of the outputs from the different Networked control System's

IV. PERFORMANCE ANALYSIS

The figure 4, 8 and 9 shows the simulation output of the ACO optimized PID controlled system, fuzzy-PID controlled system and PID controlled system. This work validates the used of ACO algorithm in tuning PID controller by comparing it with the work by [17]. The authors of the work showed that tuning the PID controller of NCS using optimal fuzzy logic compensated for the delay induced by the FlexRay network. The optimal fuzzy logic PID controlled NCS performed better than PID controlled NCS. The figure below shows comparison of the simulation output, that when ACO tunes the PID controller, the performance of the NCS is better than that of [17].

Table 1: Performance of the proposed system against an existing system

		Overshoot	Undershoot	Settling time
Proposed System	ACO tuned PID NCS	103	98	8s
Existing System	Fuzzy PID NCS	107	99	9s
	Conventional PID NCS	124	92	18s

CONCLUSION

This research work investigated the use of tuning PID controller with ACO algorithm and the result showed that the NCS performed better. The result from this research shows that when ACO tunes the PID controller it compensates for the delay introduce by the network by reducing the computational delay of the controller. This makes the system more efficient and beneficial for use in our industries.

Further works can be done on eliminating delay totally for the system, especially from the network. This Networked control system is quite a new concept and works can be aimed at proposing a new theory that will consider the effect the network has on the system which was not considered in the already existing control theory.

REFERENCES

[1] Y. B. Zhao, X. M. Sun, J. Zhang, and P. Shi, “Networked Control Systems: The Communication Basics and Control Methodologies,” Math. Probl. Eng., vol. 2015, 2015.

[2] X. M. Zhang, Q. L. Han, and X. Yu, “Survey on Recent Advances in Networked Control Systems,” IEEE Trans. Ind. Informatics, vol. 12, no. 5, pp. 1740–1752, 2016.

[3] D. Pal and S. K. Bhagat, “Optimization based integrated fuzzy logic PID controller for networked controlled systems (NCSs),” Univers. J. Electr. Electron. Eng., vol. 6, no. 5, pp. 31–41, 2019.

[4] Y. Elamin, N. A. M. Subha, N. Hamzah, and A. Ahmad, “A particle swarm optimization based predictive controller for delay compensation in networked control systems,” Commun. Comput. Inf. Sci., vol. 752, pp. 417–431, 2017.

[5] R. Pahlevi, M. A. Murti, and E. Susanto, “The implementation of PID using particle swarm optimization algorithm on Networked Control System,” Proc. - Int. Conf. Ind. Autom. Inf. Commun. Technol. IAICT 2014, no. August, pp. 35–38, 2014.

[6] J. Hamidi, “Control System Design Using Particle Swarm Optimization (PSO),” Int. J. Soft Comput. Eng., vol. 1, no. 6, pp. 116–119, 2012.

[7] S. F. H. Noorbin and A. Alfi, “Adaptive parameter control of search group algorithm using fuzzy logic applied to networked control systems,” Soft Comput., vol. 22, no. 23, pp. 7939–7960, 2018.

[8] Y. Tang, F. Du, Y. Cui, and Y. Zhang, “New Smith predictive fuzzy immune PID control algorithm for MIMO networked control systems,” Eurasip J. Wirel. Commun. Netw., vol. 2018, no. 1, 2018.

[9] D. H. Shah and A. J. Mehta, “Fractional delay compensated discrete-time SMC for networked control system,” Digit. Commun. Networks, vol. 3, no. 2, pp. 112–117, 2017.

[10] M. Tawfeek, A. El-sisi, A. Keshk, and F. Torkey, “Cloud Task Scheduling Based on Ant Colony Optimization,” 2014.

[11] M. J. Blondin, J. Sanchis, P. Sicard, and J. M. Herrero, “New optimal controller tuning method for an AVR system using a simplified Ant Colony Optimization with a new constrained Nelder–Mead algorithm,” Appl. Soft Comput. J., vol. 62, pp. 216–229, 2018.

[12] H. El-Sayed Ahmed Ibrahim, M. Said Sayed Ahmed, and K. Mohamed Awad, “Speed control of switched reluctance motor using genetic algorithm and ant colony based on optimizing PID controller,” ITM Web Conf., vol. 16, p. 01001, 2018.

[13] P. Prathyusha, B. T. Praise, and V. Suresh, “SPEED CONTROL OF DC MOTOR USING ANT,” Int. J. Adv. Res. Biol. Eng. Sci. Technol. (IJARBEST, vol. 2, no. 6, pp. 112–117.

[14] T. K. Priyambodo, A. Dharmawan, O. A. Dhewa, and N. A. S. Putro, “Optimizing control based on fine tune PID using ant colony logic for vertical moving control of UAV system,” AIP Conf. Proc., vol. 1755, no. July, 2016.

[15] C. L. Johansson, “Ant Colony Optimization Optimal Number of Ants Ant Colony Optimization - Optimal Number of Ants,” 2018.

[16] G. Chen, Z. Li, Z. Zhang, and S. Li, “An Improved ACO Algorithm Optimized Fuzzy PID Controller for Load Frequency Control in Multi Area Interconnected Power Systems,” IEEE Access, vol. 8, pp. 6429–6447, 2020.

[17] O. Akpeghagha, M. Maneke Igw, C. Chukwudozi, and C. Marcelinus, “Optimal Fuzzy PID Controller Design for Stability of FlexRay Bus in NCS,” J. Appl. Sci., vol. 20, no. 2, pp. 51–57, 2020.