Detection and Minimization of Time Delay in Distributed Control System

Otagburuagu, Richard Obinna[1], Eneh, Innocent.I[2]
Department of Electrical Electronic Engineering
Abia State Polytechnic Aba and Enugu state
University of Science and Technology, Agbani
richard.otagburuagu@abiastatepolytechnic.edu.ng
DOI:10.56201/ijemt.vol.11.no1.2025.pg123.138

Abstract
Multi-Source Communication Delays (MSCD) seriously impair the operation of Distributed Control Systems (DCS), which are essential for real-time process monitoring in industrial settings. These difficulties were revealed by a testbed analysis conducted at the Tiger Food Paste production facility. A Multi-Aware Particle Swarm Optimisation (M-PSO) technique has been developed to identify communication delays by optimising significant features such as routing speed and bandwidth in order to solve MSCD. The approach guarantees accurate delay detection in dynamic network conditions through enhancements of gradient descent and dynamic learning rate updates. Additionally, a Network Management Algorithm (NMA) has been adopted, which uses priority scheduling to reduce latency and enhance bandwidth usage. Significant improvements in delay detection and management were shown when the suggested solutions were evaluated against common QoS criteria, improving the DCS network's dependability and efficiency. With applicability in a variety of real-time control systems, this study offers a strong framework for resolving communication delays in industrial automation. Advanced machine learning models may be used in future research to increase flexibility to changing network circumstances.

Keywords: Distributed Control System (DCS); Multi-Source Communication Delays (MSCD); Multi-Aware Particle Swarm Optimization (M-PSO); Network Management Algorithm (NMA); Industrial Automation

1. INTRODUCTION

Time-delay is a common term in process control system engineering studies which is inevitable and has remained infinite in various dimensions of process design, resulting to system instability, performance degradation, control loop tuning difficulty, increased sensitivity to disturbance, among other problems. According to Xiaoyi et al. (2019), every dynamic system at some point has time varying steady state delay, and it cannot be eliminated even with the most sophisticated technology. Similarly, the high demand for products has equally resulted in the complex modelling of process designs to meet up with the production demand (Tan et al., 2017). These complexities in design have also presented various uncertain behaviours (Langerak and Polderman, 2016) which impact the system stability and presents the need for delay stability analysis.

Delay stability analysis has remained an active research area over the past decade (Liu et al., 2019; Alexandre, 2011; Feng et al., 2021; Liu et al., 2016; Manikandan and Kokil, 2020; Wang et al., 2020), with major emphasis of its impact on the stability of Distributed Control System (DCS).
DCS is a control system architecture where various Basic Process Control System (BPCS) components are interconnected over a communication network to exchange data and allow for a better industrial automation process (Ashraf and Jihong, 2015). These BPCS components include actuators, sensors, valves, controller, reactor plants, communication unit, etc and are characterized with dynamic behaviour which yields a finite dimensional approximation problem, with continuous time inequality constraints (Chongyang et al., 2021). For instance, the time it took the sensor to identify the input signal is sampling delay time; the time the controller processes the signal is the processing time, others are network delay time, synchronization time, intentional delay, slack time, transport time etc. Fridman (2014) classified the delay time into dead time delay, measurement time delay and communication time delay respectively. The dead time delay is the total time of the controller operation, communication delay presented latency constraints due to network components, while the measurement delay arises from the actuators and sensor sampling performance (Estrada et al., 2017).
Generally, delay time can be static or time variant (Chongyang et al., 2021). The static delay are known delay which are always considered during the modelling of the control systems, while the time varying delay are delay which originates and changes within the time horizon of the technical process and collectively these delay are stochastic and impact on the stability analysis of the DCS. Many studies have addressed the delay problems from various perspectives of research field which include communication and control system engineering. The communication delay was addressed using methods such as feedback approach, mathematical approach, sapling approach, wireless-hart, etc. (Abusayeed et al., 2012; Hamidreza et al., 2015; Maurice et al., 2012; Alexandre, 2011; Rahim et al., 2015). Similarly, the dead time delay was addressed in Joelianto et al. (2008) using Programmable Logic Controller (PLC)/Proportional Integral Differential (PID) (Inyama and Agbaraji, 2015). Salem (2015) applied Neuro-Fuzzy (NF) logic and Echo state neural. However, despite their success, there is need for a delay solution which intelligently addresses the problem of system instability.
According to Steven (2018), addressing the problem of delay often "requires the need to predict the past input". In most of the early cases, the state predictor method was employed for the system state observer (Thau, 1973), but recently, methods such as multiple intervals pseudo-spectral approximation scheme (Tang, 2019), time scaling transformation approach (Wu et al., 2020) and control parameterization schemes (Chongyang et al., 2021) were all applied for the prediction of delay time; but despite their success, Ge et al. (2023) argued that the stochastic nature of these delays has made it difficult for optimal management. This is because the delay is a composition of many factors from communication, process design and basic process control system, hence these three dimensions must be considered while addressing this problem to achieve a holistic solution. To this end, the paper developed a control solution, which focuses on the management of communication delay through the utilization of optimization algorithms to detect the problem and the other, applying priority scheduling scheme in the management.

2. RESEARCH METHODOLOGY

The methodology used for the research is the experimental, simulation and observation approach. The experimental approach was used to investigate the impact of MSCD on the DCS network. Another investigation was performed to assess the impact of internal delay, transport delay, input and output delay on a closed loop linear time invariant control system network through simulation approach. The data model of the delay was collected through observation and then analyzed considering Service Level Agreement standard for network component delay analysis. The findings showed that delay in DCS originates from two different sections of the network which are the communication section and the process design section. Each of these sections induces various forms of delay which collectively formulated the stochastic problem.
To solve this problem, first optimization techniques was adopted and applied to develop an algorithm which detects rising delay through network condition, while priority scheduling scheme-based network management algorithm was developed to manage the communication delay. Then, this model was integrated in the DCS and then used to manage stochastic delay on the network.

2.1 Data Acquisition

The data was acquired by investigating the Tiger Food Paste processing plant which is located in Anambra State, Nigeria. The company is equipped with communication and industrial control system distributed network used for the automation and monitoring of food pastes such as tomatoes among others. The process design test bed is made of the sensors for monitoring pressure, level, concentration, PH values and temperature during the technical process. Other components include PLC controller, alarms, multiple batch reactor plants and DC motors. Overall, these components made up various hierarchies of the DCS and are monitored remotely from one central station via the Supervisory Control Data Acquisition System (SCADA) network. Figure 1 were used to present some sections of the physical testbed.

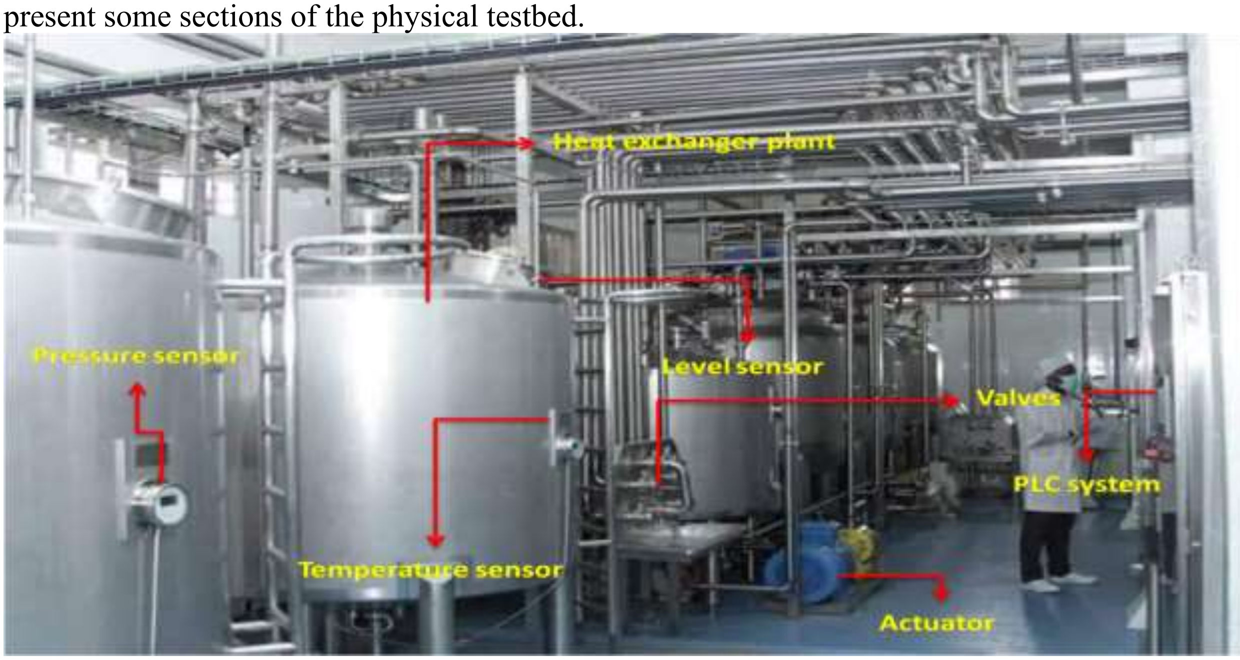

Figure 1: The Experimental testbed (Courtesy: Tiger Food Paste LTD.)

The experimental testbed used for this investigation, which involved a heat exchanger system, was shown on Figure 1. The components were labelled to show their functions in a process control setting. To measure the system's internal pressure, a pressure gauge is installed on the tank. By ensuring that pressure levels stay within specified bounds, this sensor delivers vital information for preserving operating safety and maximizing performance. This sensor, which is positioned at the tank's base, gauges the fluid or process medium's temperature. In order to preserve thermal stability and guarantee the heat exchanger's effective operation, temperature monitoring is crucial. By measuring the height of the liquid or medium inside the tank, the level sensor makes sure that the right volumes are kept for continuous operation. It aids in avoiding depletion or overflow, which might interfere with the process. Heat transfer between two fluids occurs in this crucial part of the system.

By recycling heat or cooling the process fluid as needed, the heat exchanger promotes energy efficiency. It is essential to the thermal management of the plant. In order to regulate the fluid flow inside the system, valves are positioned strategically. By limiting or permitting fluid passage, they govern the process and guarantee exact control over pressure, flow rate, and direction. The PLC is the system's automation brain. After processing the data from the sensors, it instructs actuators and other parts. By adapting to dynamic changes in the process environment, the PLC guarantees seamless operation. The actuator converts electrical signals from the PLC into mechanical movement to operate valves or other mechanical devices. It plays a crucial role in executing the automated responses necessary for controlling the system.

3. THE PROPOSED SOLUTION TO DELAY IN DSC

From the system investigation, it was observed that delay originates from two major sections of the DCS, which are the communication components which produces the communication delay and the process design components which produce transport delays.

3.1 The detection algorithm for communication delay using optimization technique

Today there are varieties of optimization algorithms which are specialized in solving various complex problems considering their parameters. Before adopting any, it is important to identify the system requirements which in this case are the communication variables such as delay time, bandwidth availability factor, routing speed and position of components. The position defines the source of data origination, while the delay time and speed are the overall time and routing speed of data traveling from each component source to destination, and the bandwidth availability factor is the ratio between the packet been transferred and the actual bandwidth capacity of the channel per seconds. The aim is to detect problem in the network channel considering these pre-defined parameters through fitness computation and update until communication delay is properly managed. To this end, Particle Swam Optimization (PSO) (Dorigo, 2008; Mubeen and Yadav, 2022; Ge et al., 2023) was adopted as the compatible optimization techniques to inspire the communication delay detection algorithm called Multi-aware (M-PSO).

First the problem is defined considering the components particle such as multiple reactor plants which have monitoring temperature and pressure sensors, valves, actuators and PLC control system which communicates within the DCS. These components are defined as x ; positions as p_i, speed of data transmission as S_n,packet transfer (d), time (t) and bandwidth availability factor bf. Collectively these network parameters were used to define the stochastic problem within the DCS. To identify this problem, fitness computation was performed which considers the channel quality from every component position using the model in Equation 1;

$$\textbf{Fitness}= \frac{1}{(1+t+bf*S)*(1+\sum_{i=1}^{n}(Distance_{(x)},i)^2} \qquad \textbf{(1)}$$

Where $1 + t + bf * S$ considered the network behaviors such as delay time, speed and bandwidth availability factor. When the values of these parameters are small, these implied that the network overall fitness is high and indicates poor quality which results to communication delay beyond the desired standard for DCS best practices. The term $(1 + \sum_{i=1}^{n}(Distance_x, i)^2$presents the position of the various components which interacts within the DCS network from $i - th$ position. However, to capture the complex relationship within these parameters as they communicate, nonlinearity was introduced to each variant parameters of the network so as to facilitate a more sophisticated fitness function which properly characterized the DCS as in Equation 2;

$$\boldsymbol{Fitness} = \frac{\alpha*speed^{-\beta}}{(\Upsilon*t+\mu*bf*+e)} \qquad \textbf{(2)}$$

Where -β is the nonlinear parameter introduced to the routing speed, Y presents the nonlinear function for the delay time, μ is the nonlinear for bandwidth availability factor, α is the fixed constant of the scaling factor through the fitness process, and e is the overall constant. These nonlinear parameters are updated using gradient descent (Ranjana and Vinay, 2023) and learning control function. The reason for the adoption of gradient descent as the optimization update algorithm of choice was to address the issues of local maxima posited by Kshitij and Chong (2020) as a common problem in optimization algorithms during fitness computation, especially in a complex search space like the case of the DCS. The update models for each of the nonlinear fitness components are presented in Equation 3-5;

$$\frac{dF}{d\beta} = -\frac{\alpha*speed^{-\beta}*In\,(speed)}{(\Upsilon*t+\mu*bf+e)^{\wedge}2} \qquad \textbf{(3)}$$

$$\frac{dF}{d\mu} = -\frac{\alpha*speed^{-\beta}*bf}{(\Upsilon*t+\mu*bf+e)^{\wedge}2} \qquad \textbf{(4)}$$

$$\frac{dF}{de} = -\frac{\alpha*speed^{-\beta}}{(\Upsilon*t+\mu*bf+e)^{\wedge}2} \qquad (5)$$

These Equations were used to adjust each of the nonlinear parameters until their optimal values are determined in the direction of a descent fitness function. This optimization process was also made possible through the learning rate (ŋ) which is also a gradient descent hyper-parameter which determines the learning step size for the updating process over time. The learning rates for each update of the nonlinear parameters were presented the Equation 6-8;

$$\boldsymbol{\beta_{update}} = \boldsymbol{\beta_{current}} - (ŋ) * \frac{dF}{d\beta} \qquad (6)$$

$$\boldsymbol{\Upsilon_{update}} = \boldsymbol{\Upsilon_{current}} - (ŋ) * \frac{dF}{d\Upsilon} \qquad \textbf{(7)}$$

$$\boldsymbol{\mu_{update}} = \boldsymbol{\mu_{current}} - (\eta) * \frac{\boldsymbol{dF}}{\boldsymbol{d\mu}} \quad \textbf{(8)}$$

The Equations 6 to Equation 8 are the learning rates which the nonlinear parameters in Equation 3-5 used for update fitness computation for the network speed, delay time, and bandwidth availability factor. This fitness computation produces output which are normalized to fall within the range of 0-1 using Equation 9

$$\boldsymbol{N_{fitness}} = \frac{\boldsymbol{V_{fitness}} - \boldsymbol{Lf_{value}}}{\boldsymbol{Mf_{value}} - \boldsymbol{Lf_{value}}} \quad \textbf{(9)}$$

Where Lf is the least fitness value obtained during the fitness computation process, Mf is the maximum fitness value, V presents each value. This Equation was used to compress the outcome of the fitness process to produce values within the range of 0-1 without affecting the relativity between the fitness variables. This ensures easy interpretation and programming of set-point for the communication delay management.

First the algorithm was initialized and then parameters were identified considering speed, position, time delay and bandwidth availability factor and used to define the network communication problem. Fitness test was performed considering nonlinearities of every individual particle with Equation 1; during the process the nonlinear parameters were updated with the Equation 2-4, using the learning rate or each nonlinear variables as posited with Equations 6-9. The outcome of the fitness was normalized using the model in Equation 9 while the result was analysed considering a defined standard of 0.6 as the reference for quality of service, while values below the standard implied arising poor network performance and presents the need for control measured. The pseudocode of the algorithm was presented as algorithm 1;

M-PSO Algorithm (Algorithm 1)

1. Start
2. Parameter initialization
3. Gather population of particles and network parameters
4. Define problem formulation and induce nonlinearity
5. Perform fitness and determine network condition
6. Update fitness with gradients descent and learning rate
7. Apply normalization to fitness score
8. For Updated fitness scoreCheck value and compared with reference value (0.6)
9. If Fitness score is < 0.6Then Activate Network Management Algorithm
10. Else
11. Return to step 3
12. End

3.2 The Network Management Algorithm (NMA) to minimize the detected delay

This section focuses on processing the outcome of the Algorithm 1 performance. In the M-PSO fitness computation was used to identify the behaviour of the network and then determine the rate of network issues such as congested bandwidth, poor routing speed, time delay which overall affects the network stability. In the network NMA the algorithm seeks to solve the identified problem of the network and is triggered when the fitness score is below 0.6, which is value that implied arising network communication delay.

To address the problem, priority scheduling scheme was utilized to schedule critical components of the process design for higher priority and allow throughput during poor network performance, while those with lower priority are queued until the network condition improved, then they are allowed throughput. This way quality of service was maintained and communication delay greatly reduced from the source. In the formulation of the NMA, all the components from the process designs were identified from the network domain expert, then their impacts on the network were considered for their respective prioritization. The key components of the network are temperature sensor, pressure sensor, colour sensor, concentration sensor, level sensor, actuator, valve, heat exchanger plant and the PLC controller, while their priority and signals were assigned according to Table 1 based on data collection from the DCS domain expert;

Table 1: Components classification in priority

S/N	Components classification	Class	Signals	Priority
1	Temperature sensors	A	ʆ	High
2	Concentration sensor	A	ʂ	High
3	Heat exchanger plant	C	ᶓ	N/A
4	Pressure sensor	C	Φ	Low
5	Level sensor	B	ϕ	Medium
6	PLC	A	Ψ	High
7	Valve	C	Σ	Low
8	Actuator	A	P	High
9	Color sensor	C	Ψ	Low

The Table 1 presents the data model used for the development of the NMA. The components in the table were assigned priority based on their level of impact on the technical process and then rule based was used to assign throughput to those of high priority when the network condition is degrading, to maintain quality of service. Why pressure sensor was assign to the low priority was because Guass Law already posited that temperature and pressure are directly proportional (Gurmukh and Marco, 2022), which implied similar role on the network analysis. The algorithm for the NMA is presented as Algorithm 2;

Pseudocode of NMA (Algorithm 2)

1. Start
2. Initialization and time control function
3. Set time (t) = 500ms
4. Generate population of all components
5. Identify all component signals
6. **%% Component classification and prioritization**
7. Assign class A, B, C, to components
8. Allow throughput for all signals

9. Check for fitness score
10. For output >=0.6 = true
11. Apply step 8
12. Else for step (10) =false
%% Scheduling of components
13. Queue all signals for B, C and assign to (t)
14. Activate time (t-1)
15. While t= 0ms
16. Apply step 9
17. Do until
18. Step (10) = true
19. Return to step (8)
20. End

First, the time control function and population of the component's particles were activated. The time was set to 5000ms and then components signal identified and assign classes A-C. While throughput is allowed for all classes of component signal, fitness test was used to check the network behaviour and when below 0.6 scale factor, the signal from class B and C are queued using the delay time as it counts down, while other components of class A is allowed throughput. As the time counts to zero, the fitness score is checked again and then the fitness score is above 0.6, then the queued signals are allowed throughput. Overall, the NMA used priority scheduling method to manage the network problem in the DCS and ensure that communication delay which arises due to the stochastic behaviour of the DCS is minimized to the lowest level**.**

3.3 The model for the Communication Delay Management (CDM) in DCS

The model for the management of communication delay was develop considering the M-PSO algorithm which was used for the detection of arising constraints of communication quality in DCS such as communication delay, packet drop, poor routing speed. Even though communication delay was identified as the major implication of poor quality of service in DCS during arising poor quality of service, however other constraints like packet dropout, poor routing speed, and congestions were also revealed as constraints of service quality in DCS. To manage the problem, the NMA algorithm 2 was developed which used fitness computation result of the algorithm 1 to detect the network condition and then applied priority scheduling scheme to manage the problem and allow throughput for component signal with higher priority, while the other component signal is queued until quality of service is restored to the network. Figure 3 presents the flow chart of the CDM.

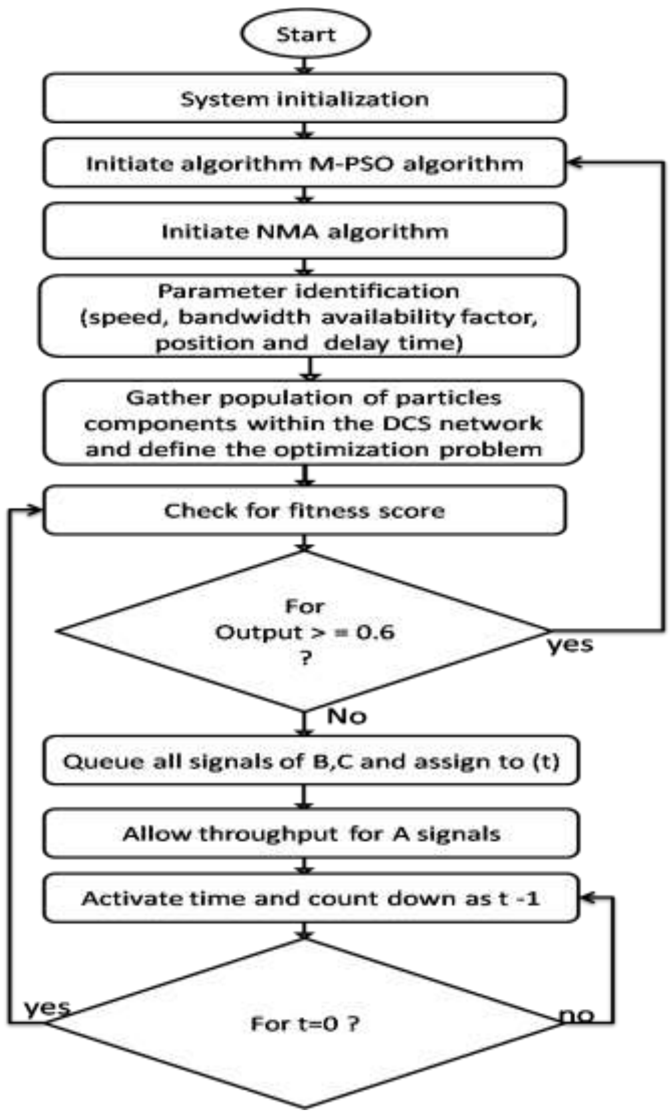

Figure 2: Flowchart of the model for managem hient of communication delay

The flowchart in Figure 2 illustrates the workflow of a time-controlled signal processing system. It begins with system initialization and setting a time delay of 5000 milliseconds. Next, the population of all system components is generated, and their signals are identified. Components are then categorized into classes A, B, and C. The system allows throughput for all signals and checks their fitness score. If the output fitness score meets or exceeds 0.6, the process continues; otherwise, signals from classes B and C are queued and assigned to the time delay. Only signals from class A are allowed to pass through while the system countdown activates, decrementing the timer (t - 1) until it reaches zero. Once the timer equals zero, the queued signals are reprocessed. This design ensures systematic signal prioritization and efficient throughput management based on fitness scores and timed processing.

4. SYSTEM IMPLEMENTATION

The system was implemented using Simulink in MATLAB programming language. The implementation was done in two different sections considering the PLC and Neural network control system. Figure 3 presented the Simulink block of the PLC based DCS used in the characterization. This was achieved using optimization toolbox, differential function, integral function and function blocks in Simulink. To achieve this, each of the blocks were selected from the Simulink library and then applied to configure the DCS. In Figure 4, the PLC was replaced with the neural network model trained as the machine learning based stochastic regularization approach for delay management for real-time technical process operation. The neural network implementation was done using neural network toolbox and the programming was applied to integrate the network NMA to ensure improved resource management and real-time data process to effectively manage delay.

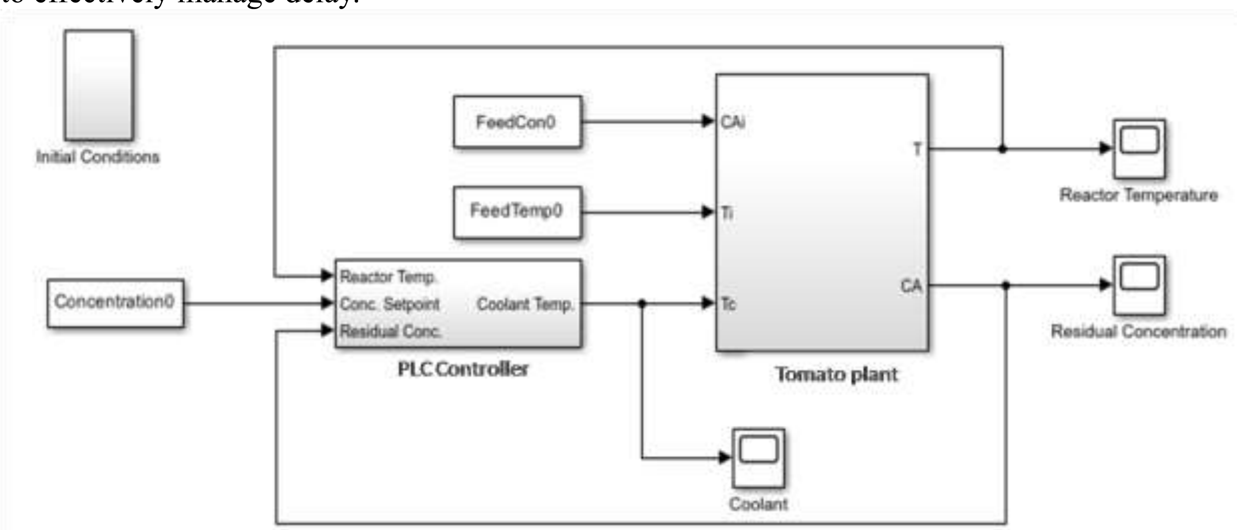

Figure 3: Simulink of the PLC control system

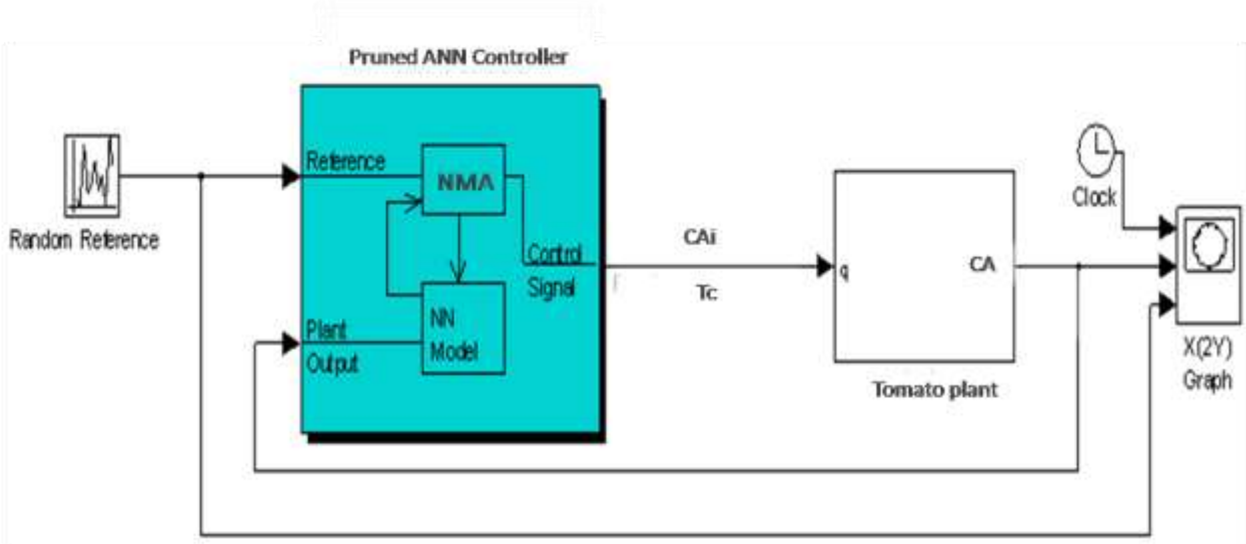

Figure 4: Simulink of the neural network control system

5. RESULT OF THE DCS WITH COMMUNICATION DELAY MANAGEMENT MODEL

This section presents the result of the CDM developed to address the impact of communication delay due to congested bandwidth utilization factor on the network. To achieve this, the algorithm 1 which utilized M-PSO to detect the network condition considering population of particles and then apply nonlinear fitness computation in Equation 1 to determine the network condition and when the utilization factor exceeds 0.6, the Algorithm 2 which depicts the NMA was activated to manage the problem through the queuing of less priority components signals and allow throughput for signals of priority A, and then continue to check until the utilization factor is returned to scale less than 0.6, then other components can interact. To evaluate the result the RRT in Figure 5 was utilized to determine communication delay with the new CDM during system input with disturbance.

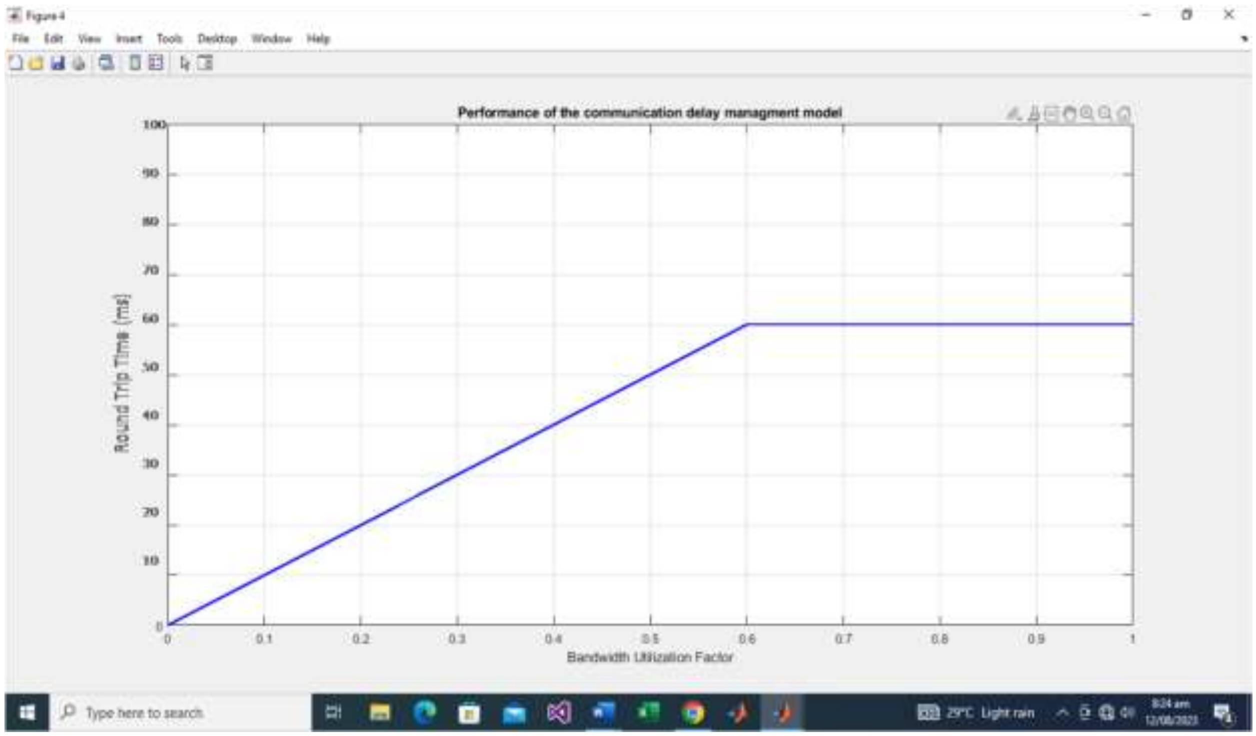

Figure 5: Result of RTT with CDM

Figure 5 posited the RTT of the network with CDM with input signal with disturbance was applied. The RTT which was measured using the delay model and from the result in Figure 5, it was observed that the time increases as the utilization factor increases, but at 0.6 utilization factor, it maintained a steady state. What is actually happening at this point was that the fitness computation in the M-PSO, detects the arising congestion due to the bf of 0.6 and then queued the component signal with less priority, while the key components signal of priority A as depicted in algorithm 2 was allowed throughput. This scheduling process resulted to the constant delay that was maintained until the network was restored to a better state. The average RTT of the network with CDM is 50.201ms, which when analysed considering the SLA standard; it is very food for a DCS.

6. CONCLUSION

This study presents a detailed examination of the impact of Multi-Source Communication Delay (MSCD) on Distributed Control System (DCS) networks, focusing on the delays originating from communication and process design components. A hybrid approach that included observation, simulation, and experimental analysis was used. An experimental study of a real-world testbed at the Tiger Food Paste processing factory yielded important information on the causes and consequences of delays in the DCS network.

In order to overcome these obstacles, a brand-new Multi-Aware Particle Swarm Optimisation (M-PSO) technique was created to identify communication delays. To guarantee reliable identification, nonlinear parameters for bandwidth, routing speed, and delay duration were included. Gradient descent and learning rate updates were used to improve the optimisation model and find the best fitness parameters for the network circumstances. To reduce delays, a Network Management Algorithm (NMA) was also suggested. It uses a priority scheduling technique to control bandwidth distribution and enhance network stability in general.

The results showed that the proposed strategy successfully identified and controlled DCS network delays, preserving communication effectiveness within the acceptable quality of service (QoS) range. In addition to improving the DCS's dependability, this novel method offered a scalable framework for dealing with stochastic delays in industrial automation and monitoring systems. Future research can look at combining the suggested optimisation algorithms with cutting-edge machine learning approaches to further enhance the predictive and adaptive capabilities of real-time industrial networks.

REFERENCES

Ashraf, F., & Jijong, W. (2015). Stability and time delay tolerance analysis approach for network control systems. *Mathematical Problems in Engineering*, Hindawi, ID: 812070.

Chongyang, L., Ryan, L., Kok, L., & Song, W. (2021). Optimal state delay control in nonlinear dynamic systems. *Elsevier*, 135, ID: 109981.

Dorigo, M. (2008). Particle swarm optimization. *Vol. 3*, 1486.

Estrada, S., Velasco, V., & Rodriguez, C. (2017). Prediction-based control for nonlinear systems with input delay. *Mathematical Problems in Engineering*, Hindawi, ID: 7415418, 1–14.

Feng, W., Xie, Y., Luo, F., Zhang, X., & Duan, W. (2021). Enhanced stability criteria of network-based load frequency control of power systems with time-varying delays. *Energies*, 14(5820). https://doi.org/10.3390/en14185820

Fridman, E. (2014). *Introduction to time-delay systems: Analysis and control*. Birkhäuser, Basel, Switzerland.

Ge, Y., Qigong, C., Ming, J., & Yiqing, H. (2023). Modeling of random delays in networked control systems. *Journal of Control Science and Engineering*, Hindawi, Article ID 383415, 9 pages. http://dx.doi.org/10.1155/2013/383415

Gurmukh, C., & Marco, C. (2022). Gas laws and clinical application. *National Library of Medicine*. Retrieved from http://www.ncbi.nlm.nih.gov

Inyama, H. C., & Agbaraji, C. (2015). A survey of controller design methods for a robot manipulator in harsh environments. *European Journal of Engineering and Technology*, 3(3), ISSN 2056-5860.

Joelianto, E., Sutarto, H. Y., & Wicaksono, A. (2008). Compensation of delay time uncertainties on industrial control ethernet networks using LMI-based robust H∞ PID controller. In *Wireless and Optical Communications Networks (WOCN '08), 5th IFIP International Conference on* (pp. 1–5).

Khabbaz, M. J., Fawaz, W. F., & Assi, C. M. (2012). Modelling and delay analysis of intermittently connected roadside communication networks. *IEEE Transactions on Vehicular Technology*. Retrieved from https://www.researchgate.net/publication/326589111

Kshitij, T., & Chong, N. Y. (2020). Multi-robot exploration for environmental monitoring. In *Fundamental Optimization Methods for Machine Learning*. https://www.sciencedirect.com/topics/engineering/gradient-descent

Langerak, R., & Polderman, J. (2016). Tools for stability of switching linear systems: Gain automata and delay compensation. In *Proceedings of the European Control Conference and IEEE Conference on Decision and Control (CDC-ECC)* (pp. 4867–4872).

Liu, K., Selivanov, A., & Fridman, E. (2019). Survey on time-delay approach to networked control. *Annual Reviews in Control*, 48, 57–79.

Liu, S., Liu, P., & Wang, X. (2016). Stability analysis and compensation of network-induced delays in communication-based system control: A survey. *ISA Transactions*. https://doi.org/10.1016/j.isatra.2016.09.022

Manikandan, S., & Kokil, P. (2020). Stability analysis of load frequency control system with constant communication delays. *International Federation of Automatic Control*. https://doi.org/10.1016/j.ifacol.2020.06.057

Mubeen, S., & Yadav, D. (2022). A review of particle swarm optimization (PSO) algorithm. *International Journal of Mechanical Engineering and Technology (IJMET)*, 13(7), 19–44. https://doi.org/10.17605/OSF.IO/KQ34H

Rahim, D., Tavassoli, B., & Beheshti, M. T. H. (2015). Improved stability analysis of nonlinear networked control systems over multiple communication links. *IEEE Transactions on Automatic Control*, 55(8), 1781–1796.

Ranjana, D., & Vinay, K. S. (2023). Statistical modeling in machine learning. In *Fundamental Optimization Methods for Machine Learning*. https://www.sciencedirect.com/topics/engineering/gradient-descent

Saifullah, A., Xu, Y., Lu, C., & Chen, Y. (2012). End-to-end communication delay analysis in industrial wireless networks. NSF through grants CNS-1035773 (CPS), CNS-1320921 (NeTS), CNS-1017701 (NeTS), and CNS-1144552 (NeTS).

Salem, A. (2015). Improvement of control system performance by modification of time delay. *International Journal of Advanced Computer Science and Applications*, 6(2), 181–185.

Seuret, A. (2011). Stability analysis of networked control systems with asynchronous sampling and input delay. In *Proceedings of the American Control Conference*. Retrieved from https://www.researchgate.net/publication/50514509

Shariatmadari, H., Iraji, S., Laya, A., Anjum, O., Jantti, R., Li, Z., & Wijting, C. (2015). Delay analysis of network architectures for machine-to-machine communications in LTE system. https://www.researchgate.net/publication/280878193

Steven, A. (2018). *Control theory tutorial*. Springer Briefs in Applied Sciences and Technology, 95–102.

Tan, C., Zhang, H., & Wong, S. (2017). Delay-dependent algebraic Riccati equation to stabilization of networked control systems: Continuous-time case. *IEEE Transactions on Cybernetics*, 48(10), 2783–2794.

Tang, X. (2019). Multiple interval pseudo-spectral approximation for nonlinear optimal control problems with time-varying delays. *Applied Mathematical Modeling*.

Thau, F. (1973). Observing the state of nonlinear dynamic systems. *International Journal of Control*, 17(3), 471–479.

Wang, Z., & Seiji, F. (2020). Control strategy for networked control systems with time delay and packet dropout using linear matrix inequalities. *EURASIP Journal on Wireless Communications and Networking*, 42. https://doi.org/10.1186/s13638-019-1556-4

Wu, D. (2019). A new computational approach for optimal control problems with multiple time delays. *Automatica*.

Xiaoyi, W., Yuqin, Z., Zhiyao, Z., Wei, W., & Wei, L. (2019). Time delay system control based on the integration of active disturbance rejection and modified twice optimal control. *IEEE Access*, ID: 130734, 1–11.