ISSN: 2093-5374 / E-ISSN: 2093-5382

Adaptive Hybrid Optimized Algorithms for Dynamic Congestion Control and Monitoring in IoT- Enabled Wireless Sensor Networks

Elham Ibrahim Mahmoud[1*], Dr. Montassar Aidi Sharif[2], and Dr. Fatih Korkmaz[3]

[1*]Department of Electronic and Control Engineering, Technical Engineering College, Kirkuk, Northern Technical University, Kirkuk, Iraq. elham.mahmood25@ntu.edu.iq, https://orcid.org/0009-0005-3567-8529

[2]Assistant Professor, Department of Artificial Intelligence Engineering, Technical Engineering College for Computer and AI, Kirkuk Northern Technical University, Kirkuk, Iraq. msharif@ntu.edu.iq, https://orcid.org/0000-0002-9879-0631

[3]Associate Professor, Department of Electrical and Electronics Engineering, Faculty of Engineering, Çankırı Karatekin University, Türkiye. fkorkmaz@karatekin.edu.tr, https://orcid.org/0000-0001-8524-2831

Received: February 10, 2026; Revised: March 18, 2026; Accepted: May 06, 2026; Published: June 30, 2026

Abstract

The high rate of Internet of Things (IoT) applications is putting tremendous pressure on wireless sensor networks, which congest, raise latency, and packet loss because of insufficient computational and communication capability. This paper introduces a smart hybrid congestion control system based on fuzzy logic and particle swarm optimization (PSO) to improve network performance. Real-time inputs into the system, like packet loss and delay, are processed by the system through a fuzzy inference mechanism, and PSO dynamically decides the transmission intervals so that there is a better quality of service. The model is deployed on an ESP32 platform that comprises a hardware platform and real-time monitoring on the Blynk IoT environment and a Python-based analytical interface. The experimental analysis of the model in diverse network conditions proves that the hybrid model is much better than the traditional ones. The system obtains a small average delay of 6.33 ms as opposed to 24.96 ms for fuzzy-only and 22.09 ms for PSO-only models. The loss of packets is minimized to 0.30, which is better than fuzzy (0.6) and PSO (2.37) solutions. The congestion metric is kept at 0.97, which means that the network is stable. The hybrid approach also minimizes delay variation (jitter) to 1.9 ms, indicating better stability. Another result of the model is its quicker convergence (9.2 seconds) than PSO (18.5 seconds). The results substantiate that fast decision-making in fuzzy logic, coupled with adaptive optimization via PSO, leads to efficient congestion control. The design is a flexible, usable hybrid solution for real-time Internet of Things applications that require low-latency, reliable communication.

Keywords: Internet of Things (IoT), Wireless Sensor Networks, Particle Swarm Optimization (PSO), Fuzzy Logic, Congestion Control, Hybrid Optimized Algorithms.

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 17, number: 2 (June-2026), pp. 196-212. DOI: *10.58346/JOWUA.2026.I2.011*

*Corresponding author: Department of Electronic and Control Engineering, Technical Engineering College, Kirkuk, Northern Technical University, Kirkuk, Iraq.

1 Introduction

Wireless Sensor Systems (WSNs) are dense networks of devices that can be deployed in rugged, remote locations to sense the environment and collect highly accurate data. Sensor nodes typically consist of small devices that combine sensing, communication, and computing (Kazmi et al., 2019; Singh & Singh, 2018). WNSs consist of one or more collection nodes and thousands of sensor nodes spread over a specific geographic area. A sensor node also contains a transceiver, a central processing unit, a battery, memory, and one or more application-related sensors (Panimalar & Jacob, 2020). Wireless sensor networks based on the Internet of Things operate under strict limitations on bandwidth, processing power, memory, and battery power compared to traditional communication networks. Due to these inherent limitations, wireless sensor networks are highly susceptible to congestion, necessitating effective mechanisms for detecting and controlling congestion that ensure the continuity of data transmission while simultaneously extending network lifetime (Abdullah et al., 2025; Al-Kashoash et al., 2019). Traditional congestion control mechanisms are designed to stabilize network load in high-bandwidth wired or wireless networks and are mainly based on fixed control rules or simple congestion indicators, the most common of which is packet loss. Because packet loss is considered a lagging indicator of congestion and does not cover early congestion in IoT- enabled wireless sensor networks, a range of alternative metrics, for example, the delay change, channel load, buffer occupancy, and queue length, have been investigated to support early detection and effective congestion control (Goel et al., 2026; Lim, 2019). Traffic patterns in IoT-enabled WSNs are atypically heterogeneous and time-varying. This data includes periodic sensor data, control packets, and feedback required for system stability and coordination, as well as event-based traffic arising from abnormal conditions or environmental changes (Samarji & Salamah, 2021).

It may happen that certain locations in the network experience congestion due to the extreme variation in data traffic loads resulting from this diversity, and mostly those close to data collection centers such as gateways and cluster heads, where node-level congestion and link-level congestion are the two main forms of congestion in IoT- enabled wireless sensor networks (Sukjaimuk et al., 2018). On the other hand, when a sensor node receives more data than it can process or cache, a queue fills up, and packets are lost. This is known as node-level congestion. Many studies indicate that link-level congestion results from competition between two or more adjacent nodes accessing the same wireless medium simultaneously, leading to packet collisions, reduced throughput, and increased backlogged delays (Tawfeek et al., 2025; Zhuo et al., 2019). Congestion in these networks is generally classified into node-level congestion, resulting from buffer fullness and packet loss, and link-level congestion resulting from competition for media and packet collisions (Tawfeek et al., 2025; Jayaudhaya et al., 2023). To address these challenges, this research employs a multi-objective optimization scheme inspired by hybrid biology to address the problem of congestion and attempts to integrate fuzzy logic with the PSO algorithm. Numerous previous studies have demonstrated that combining fuzzy logic with nature-inspired optimization algorithms, such as PSO, can overcome weaknesses in conventional systems. Network optimization studies have demonstrated that fuzzy logic provides rapid response to congestion, while PSO ensures the achievement of a global optimum state. This aligns with the methodology used in this research to improve the performance of the UDP protocol, such as that of Manchahia. A water wave optimization algorithm is proposed for congestion control in WSNs. this algorithm was applied to the target function to effort network throughput (Jayaudhaya et al., 2023). The algorithm relies on three types of wave processes: refracting, refracting, and propagating. Previous studies have applied nature-inspired optimization techniques to control congestion in wireless sensor networks, including water wave optimization, genetic algorithms, and enhanced bat algorithms (Narawade & Kolekar, 2017; Pentapalli & Varma, 2016; Manshahia, 2017).

The main contribution of this study includes,

- Introduced a hybrid fuzzy logic/PSO congestion control model of IoT networks.
- Low latency (6.33 ms) and better real-time performance were achieved.
- Reduce packet loss by 0.30 to ensure reliable communication.
- Improved stability of the network with reduced jitter (1.9 ms) during dynamism.
- Applied the system practically, including a real-time ESP32 implementation and Python-based monitoring, both validated.

The organization of the paper follows the following way. Section 2 analyzes intelligent congestion control techniques, including fuzzy logic, hysteresis control, and PSO. Section 3 describes the suggested hybrid PSO-fuzzy system, its hardware architecture, the algorithm, and the experimental design. Section 4 provides results, comparative analysis, ablation analysis, and discussion. Lastly, Section 5 summarizes the research and outlines future research directions.

2 Intelligent Congestion Control

Recent studies on congestion control in wireless sensor networks integrated with the IoT demonstrate a clear shift from traditional reactive approaches to self-adaptive intelligent systems. This shift stems from the inability of fixed, threshold-based methods to handle heterogeneous data traffic patterns, the dynamic nature of wireless channels, and the severe resource constraints in emerging IoT applications (Pawar et al., 2025). Intelligent congestion control techniques can be categorized into three main groups: fuzzy-logic-based, learning-based, and PSO-based. Each collection differs in its operating mechanisms, implementation impact, and control objectives, particularly in energy consumption and computational complexity. It also includes scalability in IoT scenarios and wireless sensor networks with limited resources. The following sub-sections will address in detail intelligent congestion control models and discuss their potential and practical applications in deploying wireless sensor networks supported by IoT technology.

Reactive Fuzzy Inference Controller

Fuzzy logic is an extension of traditional Boolean logic, which relies on only two values (0 or 1). Therefore, fuzzy logic allows for the extension of values in a range between [0, 1], making it ideal for modeling imprecise concepts such as extreme congestion or average delay (Tam et al., 2018). Its advantages lie in mimicking human reasoning by simplifying code formulation using natural-language rules (if-then) and in its flexibility for efficiently handling noisy data from gas and temperature sensors. Furthermore, it offers immediate responsiveness because it doesn't require complex mathematical network models, enabling extremely fast decision-making. Disadvantages: Static nature: The rules are fixed and unchanging. If network conditions change drastically, it may not function efficiently. Difficulty in tuning: It requires expertise to determine membership function parameters (tuning) to achieve optimal performance (Alikhani et al., 2022).

The most important mathematical formulas for the fuzzy trigonometric function (trim f) are represented in equation 1:

$$\mu(x) = \max\big(0, \min(x - a/b - a, c - x/c - b)\big) \qquad (1)$$

Weighted Congestion Output To calculate the congestion metric (CM) based on the weights that are defined (60% for delay and 40% for loss), is shown in equation 2:

$$\mathrm{CM} = \big(W_{\mathrm{delay}} . \mu_{\mathrm{Delay}}\big) + (W_{\mathrm{loss}} . \mu_{\mathrm{Loss}}) \qquad (2)$$

To convert the fuzzy value to a real "transmission period" (Interval) in milliseconds, is expressed in equation 3:

$$\mathrm{Interval} = I_{\min} + (I_{\max} - I_{\min}) . \mathrm{CM} \qquad (3)$$

Simple Adaptive Control (Linear Hysteresis)

Hysteresis is a phenomenon of "response delay" or "memory delay" in a system. It means that the current state of the system depends not only on the current input but also on the history of previous inputs. It is a control system based on "straight linear rules" with a margin of hysteresis to prevent fluctuations. Adaptive Hysteresis Control (AHC) is a control technique that improves the performance of systems (such as active filters) by automatically adjusting the hysteresis band rather than fixing it. This is used to stabilize the switching frequency, reduce noise, and effectively limit harmonics, as the hysteresis band is calculated in real time to minimize switching losses. Adaptive control is a system that automatically changes its parameters (such as bandwidth) to adapt to changes in operating conditions or system characteristics (Noori & Altabey, 2022). Linear hysteresis, on the other hand, often refers to a type of rate-dependent hysteresis, characterized as being mathematically equivalent to a transfer function, where the phase lag depends on the frequency.

This research presents a specific diagram and evaluates various mathematical models proposed in IoT for controlling the retina to dynamically filter visual inputs. The problem of fixed lag when using Fixed Band Hysteresis Current Control (FBHCC) is that this causes a change in the switching frequency, which increases noise and generates high-frequency harmonics (Nahar, 2024). The Adaptive Hysteresis Current Control (AHCC) solution changes the lag band instantly based on system conditions (voltage, current, speed) to keep the switching frequency nearly constant. Features: Reduced switching losses: Thanks to frequency stabilization, Improved power quality, through effective harmonic elimination, & Faster response: Due to real-time calculations (Noori & Altabey, 2022).

Particle Swarm Optimization (PSO)

The particle swarm optimization algorithm is a probabilistic search algorithm inspired by the collective behavior of flocks of birds or fish searching for food. The algorithm is based on the idea that each particle in the swarm represents a potential solution moving within the search space, adjusting its position based on its own experience and that of its neighbors (Nahar, 2024).

Types of PSO Algorithms

The algorithm varies depending on the nature of the connection between particles (Liao et al., 2021):

- Global PSO (gbest): where each particle connects to all particles in the swarm and moves towards the best position reached by the entire swarm.
- Local PSO (lbest): Each particle is connected only to a specific set of its neighbors, which helps avoid stabilization at false local solutions (Local Optima).

- Adaptive PSO: A type in which parameters (such as inertia) change dynamically during the search.

Advantages and Disadvantages

Advantages:

1. Simplicity: It does not require complex mathematical derivations (derivative-free), making it easy to implement on microcontrollers such as the ESP32.
2. Collective intelligence: The ability to find the optimal solution in a wide search space (e.g., searching for the Send Interval between 500ms and 30000ms).
3. Adaptiveness: High capacity to readjust values as soon as network conditions change (e.g., loss or delay changes).

Disadvantages:

1. Local Optima: An algorithm may sometimes fall into a less-than-ideal solution and mistakenly believe it to be the best if the parameters are not precisely calibrated.
2. Volatility: It observes fluctuations in the results before reaching a state of convergence.

Basic Mathematical Equations:

The movement of particles in its code depends on two fundamental equations: A. Velocity Update: Equation 4 determines the "boost" with which the particle will move in the next cycle (Liao et al., 2021):

$$v_i(t+1) = w.v_i(t) + c_1.r_1.\left(pbest_i - x_i(t)\right) + c_2.r_2.\left(gbest - x_i(t)\right) \tag{4}$$

where w: Inertia Weight. c_1, c_2: Acceleration Coefficients (Cognitive and Social components). pbest, gbest: Best individual position and best swarm position. B. Position Update: Based on the new velocity, the particle's position is updated in equation 5:

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{5}$$

3 Methodology

The proposed hybrid PSO-Fuzzy system combines the speed of fuzzy inference with the accuracy of PSO optimization; the PSO adjusts the "weights" or "boundaries" of the fuzzy system to make it smarter and more adaptive**.** There is a system stability, keep Fuzzy Logic (or combine them so that the PSO adjusts the Fuzzy parameters).

Table 1: A comparison between fuzzy logic and PSO reveals aspects of the decision-making process

Aspects of Comparison	Fuzzy Logic	PSO
Response	Very fast and instantaneous	Requires several iterations to stabilize
Complexity	Easy to understand and implement	More complex; requires tuning parameters (w, c1, c2)
Goal	Implements expert human rules	Automatically finds an optimal solution
Benefits of this Research	Suitable for real-time congestion control	Effective for achieving optimal performance through optimization

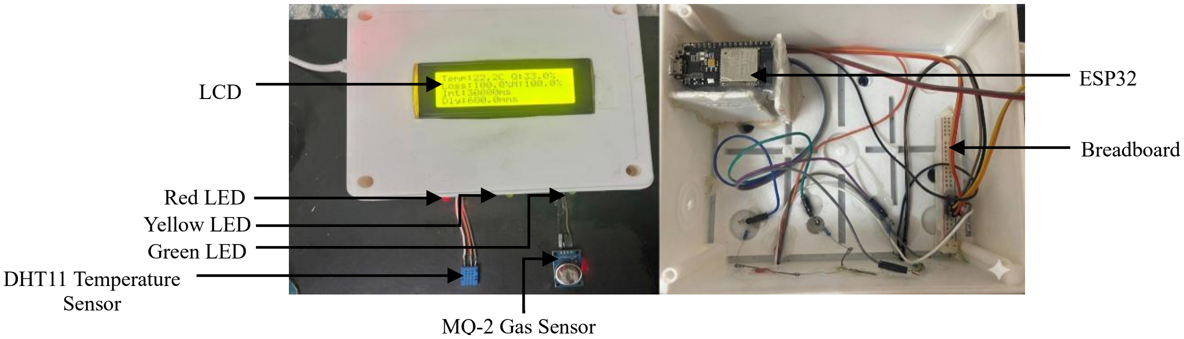
Instead of fixed equations, it will create a set of "particles," each representing a "possible solution" for the transmission speed. These particles move through the search space to find the speed that minimizes delay and data loss. It is appropriate in this section to provide an algorithmic description of the methodological approach applied in this research, specifically regarding fuzzy logic and POS, as described in table 1 which shows the primary variations in important characteristics.

The practical part: ESP32 Dual-Core Microcontroller

The hardware layer consists of an ESP32-based central processing unit (CPU) connected to a suite of sensors, including an MQ-series gas sensor and a digital temperature sensor. Results are displayed locally via an LCD screen supported by the I2C protocol to minimize the number of connections and ensure fast data transfer between the processor and the display interface:

- The main component is the ESP32 Dual-Core Microcontroller. This microcontroller acts as a Main Processing Unit (MPU) or Wireless Node, and is responsible for implementing the Hybrid POS-Fuzzy algorithm and transmitting data via the UDP protocol.
- Air/Gas Sensor, MQ-Series Gas Sensor Type MQ-135. This is called a Chemical Resistive Gas Sensor. It measures the concentration of atmospheric gases and converts them into an analog signal. The results are displayed in the table under the abbreviation MQ.
- A Temperature Sensor (DS18B20) or Digital Temperature Sensor provides environmental feedback to the system; it is the source of the temperature data displayed on the screen.
- The Display (or Local Monitoring Interface), Scientific name: I2C LCD Display (Liquid Crystal Display). Its function is to display the algorithm outputs (Loss, Delay, CM) and sensor data in real time to the user.

The figure 1 demonstrates that the system consists of a central processing unit (ESP32) that acts as the brain, connecting via I/O pins to gas and temperature sensors. Data is transferred internally using high-speed communication protocols (such as I2C for the display and Analog-to-Digital for the gas sensor), ensuring real-time processing of readings before it is passed to the hybrid algorithm engine. Internal processing relies on a feedback loop: the code calculates the difference between the transmission time and the response time. If the system internally detects any delay or loss, the fuzzy rules embedded in the micro controller's flash memory are activated to correct the path immediately. The analog signals from the MQ sensor were converted to digital values via an internal 12-bit ADC. The last 50 loss readings were then stored internally in a sliding window array to accurately calculate the instantaneous loss rate.



(a) External LCD Interface (b) Internal Hardware Layout

Figure 1: Experimental hardware implementation of the ESP32-based hybrid system. (a) external LCD interface (b) internal hardware layout

Proposed Intelligent Adaptive Hybrid Optimization Framework

The proposed system's hybrid architecture design incorporates fuzzy logic as a mechanism for making immediate decisions and a particle swarm algorithm (PSO) as a mechanism for self-improvement. The flowchart can be described using these detailed Proposed Model algorithmic steps:

Step One: Data Acquisition: Network metrics are collected in real time via UDP protocol, including: Packet Loss Rate (L): Calculated as the percentage of packets for which no acknowledgment of arrival (ACK) was received. Latency (D): Represents the round-trip time (RTT) measured in microseconds (µs).

Step Two: Fuzzy Inference System: This module acts as a Congestion Monitor, where values (L, D) are entered and processed via If-Then Rules to produce an initial congestion value, (F_{out}). The equation (6):

$$F_{out} = \mu_{loss} \cdot w_1 + \mu_{delay} \cdot w_2 \qquad (6)$$

where µ is the membership function w_1 and w_2 are the weights.

Step Three: Optimization using PSO (Optimization Layer): PSO works by reducing the "fitness function," which represents poor network performance. The fitness function (J) (equation 7):

$$J = \alpha . L + \beta . D \qquad (7)$$

where α, β these are weights of importance.

The PSO dynamically adjusts the output weight of the fuzzy (W_{Fuzzy}) to achieve the lowest value of J.

Step Four: Adaptive Control Algorithm: The "transmission period" (I) is updated based on the final output of the hybrid system to ensure flow stability are shown in equation 8:

$$I_{next} = I_{min} + (I_{max} - I_{min}).(W_{PSO}.F_{out}) \qquad (8)$$

Step Five Experimental Setup: To ensure the accuracy of the results, an integrated monitoring system was designed, consisting of: Sending Device: An ESP32 micro controller implementing the hybrid algorithm, and Receiving Station: A Python application that: Analyzes the received data, generates real-time visualizations, and saves data to CSV logs for statistical comparison. An ESP32 dual-core microcontroller was configured as the transmitting node, linked to an MQ-135 gas sensor and a DHT11 temperature sensor. The transmission of data was performed with the UDP protocol on a local wireless network. The system receiving it was in Python to monitor in real time, log, and analyze. The system was run at a total time of 15 minutes, and the first stabilisation period was 5 seconds. The loss of the packets, delay, congestion measure, and time interval of transmissions were all key parameters that were measured continuously. The parameters of the PSO were configured to have a swarm size of 20 particles, an inertia weight of 0.7, and c1 = c2 acceleration coefficients of 1.5. Loss and delay inputs were characterized using fuzzy membership functions to enable adaptive decision-making across different network conditions.

The figure 2 shows a flowchart of the proposed intelligent adaptive hybrid algorithm for congestion control. When the packet is ready for transmission, the congestion status is checked.

The hybrid Fuzzy–PSO approach adopted in this work is inspired by recent intelligent congestion control frameworks that integrate fuzzy inference systems with particle swarm optimization for adaptive parameter tuning (Tawfeek et al., 2025).

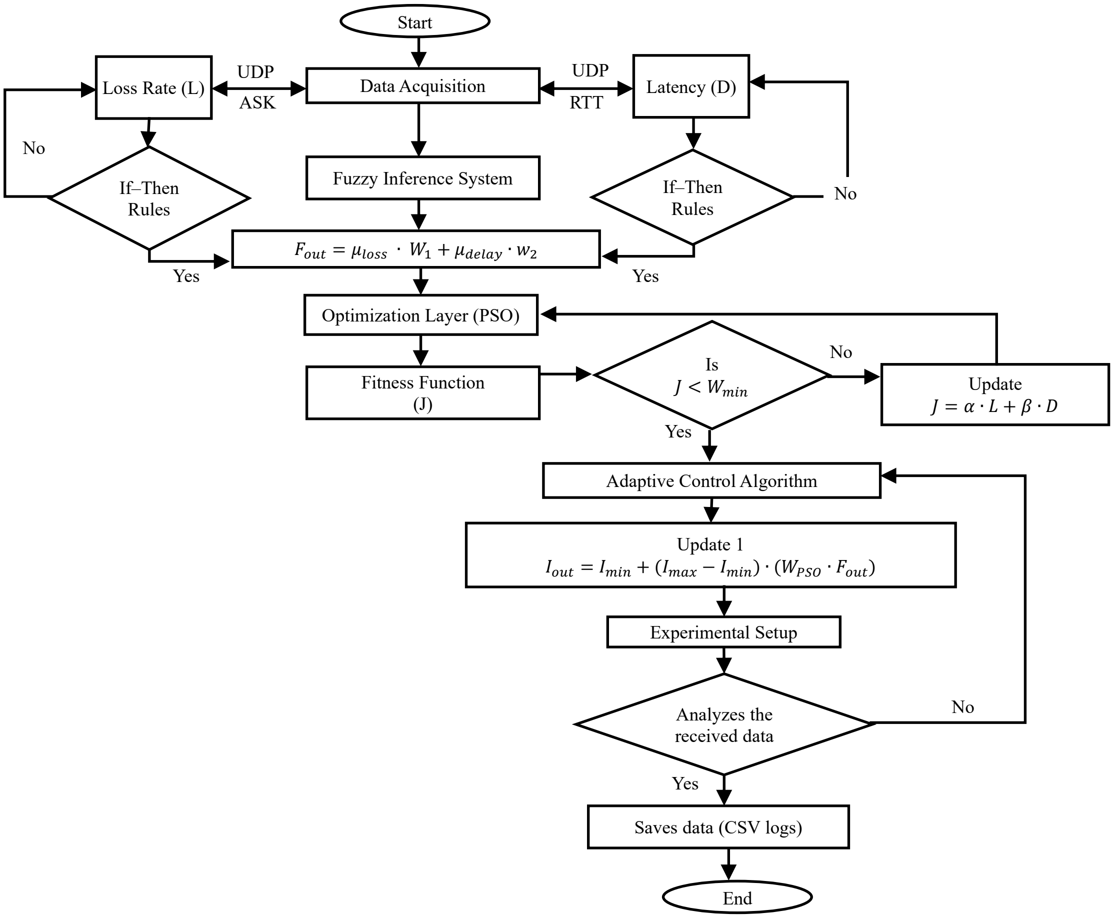


Figure 2: Flowchart of the proposed intelligent adaptive hybrid

Algorithm 1: Hybrid PSO–Fuzzy Congestion Control

Input: Packet Loss (L), Delay (D)

Output: Optimal Transmission Interval (I)

1. Initialize Particle Swarm Optimization (PSO) parameters:
    - Set the number of particles N
    - Initialize particle positions X_i and velocities V_i
    - Define inertia weight w, cognitive coefficient c_1, and social coefficient c_2
2. Repeat until the termination condition is met:
3. Data Acquisition:
    - Measure current packet loss, L and delay D
4. Fuzzy Inference:
    - Compute membership values μ_{loss} and μ_{delay}
    - Apply fuzzy rules to obtain congestion output F_{out}
5. Fitness Evaluation:
    - For each particle i:
    - Evaluate fitness $J(i)$ based on delay and packet loss
6. Update Best Values:

```
        • If $J(i) < pbest(i)$, update personal best $pbest(i)$
        • If $J(i) < gbest$, update global best $gbest$
    7. Update Velocity and Position:
        • $V_i = w \cdot V_i + c_1 \cdot r_1 \cdot (pbest(i) - X_i) + c_2 \cdot r_2 \cdot (gbest - X_i)$
        • $X_i = X_i + V_i$
    8. Optimization:
        • Adjust fuzzy system weights using optimized particle position $X_i$
    9. Transmission Control:
        • Compute transmission interval $I$based on optimized fuzzy output
    10. End loop
    11. Return: Optimal transmission interval $I$
```

The proposed Hybrid PSO-Fuzzy algorithm (Algorithm 1) is a hybrid of fuzzy logic and particle swarm optimization that can manage congestion in wireless sensor networks based on IoT. The initial one is that real time inputs (packet loss and delay) are processed by a fuzzy inference system to provide a crude congestion output. This will ensure an immediate response to the variations in the network. PSO then acts on this output to optimize the transmission parameters by further simplifying the positions and velocities of the particles by comparing them to a pre-determined fitness expression. The flow of data is then controlled by the use of the most effective transmission interval. Such a hybrid approach offers a trade-off between dynamic optimization and rapid decision-making and results in a small packet loss, low delay, and increased network stability.

4 Result and Discussion

Both hardware and software settings are included in the system implementation to make it run efficiently. The hybrid PSO-Fuzzy algorithm was implemented in real-time with an ESP32 microcontroller coded with the Arduino IDE (version 2.x) using embedded C/C++. UDP over Wi-Fi was used to provide communication among the nodes. A Python-based monitoring system was created on the receiver side and is based on Python 3.10, with NumPy, Pandas, and Matplotlib, which were used as data processing, logging, and visualization libraries. Cloud-based real-time monitoring and remote access were carried out on the Blynk IoT platform. The collected data in the course of the experiments was rated in CSV format and analyzed offline to assess the performance parameters of delay, packet loss, and congestion.

The information in this research was acquired through real-time experimental application and not simulation. The deployment of the system was done with an ESP32-based hardware system, in which sensor values and network parameters were recorded in real conditions of operation. Data preprocessing involved removal of invalid and missing data, removal of sensor data noise through a simple moving average algorithm and standardization of delay and packet loss information to a standard scale to offer consistent input to the fuzzy system. The dataset is in the form of continuous readings, which were recorded during a 15-minute experimental time, and about 900-1200 instances of transmission were recorded in each experiment. A series of experimental trials under different conditions of traffic was carried out to provide reliability. The data after processing was then analyzed in terms of performance evaluation and comparison of performance among various congestion control methods.

The suggested system has parameter settings and threshold parameters that are clear and precise to maintain stable and consistent performance. The size of a swarm used in the PSO algorithm is set to 20, inertia weight w=0.7, acceleration coefficients c1=c2=1.5. Initialization of particle position (transmission interval) is done in the range of 500 ms to 30000 ms. In the case of the fuzzy system, the input thresholds of packet loss and delay are determined with the help of membership ranges (low, medium, and high) on the basis of the empirical analysis. There is a congestion threshold, whereby at a certain point, when the congestion metric hits a certain limit, the transmission interval is expanded in order to decrease network load, and vice versa.

Packet Loss (L) is calculated as the ratio of lost packets to total transmitted packets, as shown in equation 9:

$$\mathrm{L} = \frac{\mathrm{P_{sent}} - \mathrm{P_{received}}}{\mathrm{P_{sent}}} \times 100 \tag{9}$$

Delay (D) is defined as the average time taken for a packet to travel from sender to receiver in equation 10:

$$\mathrm{D} = \frac{\sum_{i=1}^{N} (\mathrm{t_{receive,i}} - \mathrm{t_{send,i}})}{\mathrm{N}} \tag{10}$$

The Congestion Metric (CM) is computed as a weighted combination of delay and loss, represented in equation 11:

$$\mathrm{CM} = \mathrm{w_1} \cdot \mathrm{L} + \mathrm{w_2} \cdot \mathrm{D} \tag{11}$$

where $w_1$and $w_2$characterize the relative standing of loss and delay. The transmission interval (I) is dynamically modernized based on the optimized congestion output.

This section is critical for presenting and evaluating system behavior, reducing response time, achieving balanced load distribution, and increasing network scalability within a wide area IoT network. Provide an assessment of the results of each system individually, starting from the baseline of the fixed non-adaptive data transfer rate before the addition of adaptation, which represents the initial state of the protocol without any intelligent intervention, where its data is transferred at fixed time intervals regardless of the network state. Constant-rate adaptive transmission applications are studied based on the following network parameters: 1) Congestion metric, 2) Delay, 3) losses, 4) Gas sensor (MQ), 5) Temperature sensor (DHT11). The transmission rate is the amount of information that the system can process at a specific point in time. Using a non-adaptive data transfer rate, a linear heuristic algorithm, a fuzzy logic controller (FLC), a particle swarm optimization (PSO) algorithm, and a hybrid PSO–Fuzzy algorithm, the following results were obtained: losses, delays, send interval, and congestion metrics, and the comparison of these parameters is shown in the following figures.

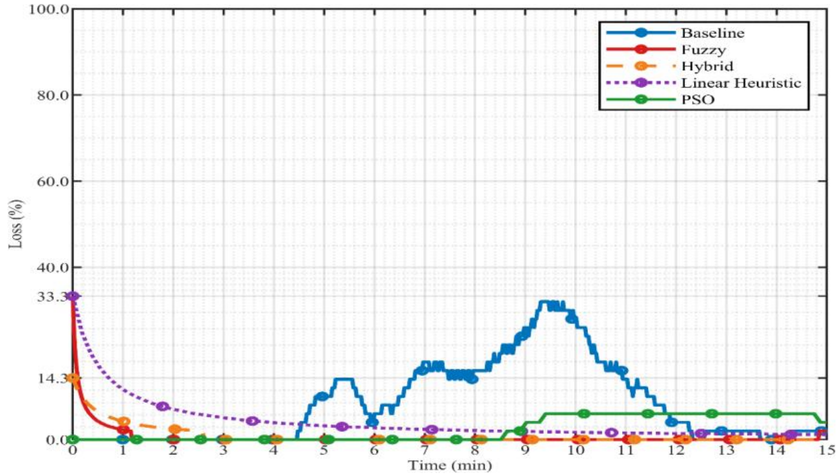


Figure 3: Comparison of loss for all systems

In figure 3 compares the loss characteristics of various congestion control techniques over time. In the baseline model, there is a high and oscillating loss, thus poor congestion handling. The linear and the fuzzy heuristic methods minimize the loss slowly but stabilize over time. The PSO approach enhances performance with some variations. The hybrid PSO Fuzzy model is the most effective, with almost no loss of packets occurring in a short time, and being able to sustain the performance over time, indicating that it is effective in the dynamic network environment.

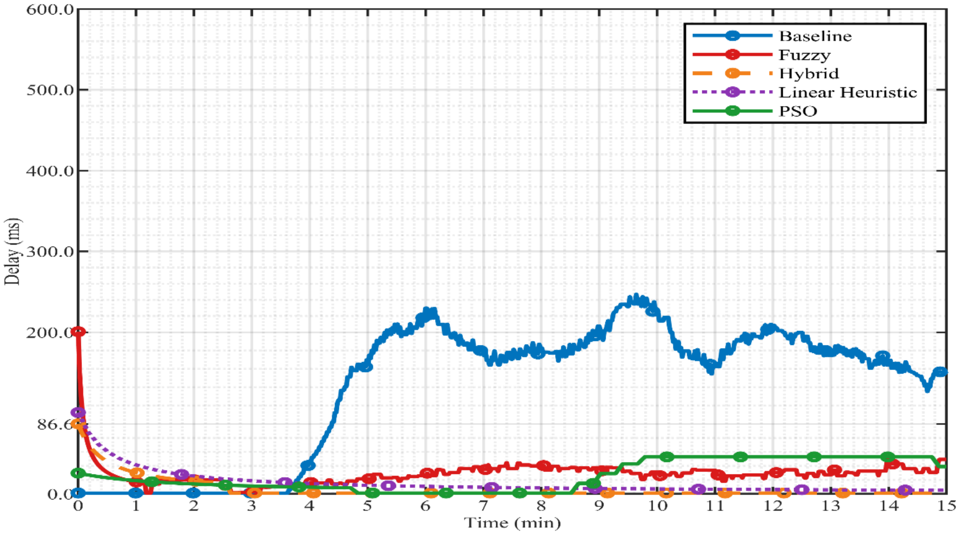


Figure 4: Comparison of the delay for all systems

The figure 4 shows a comparison of delay performance of various congestion control methods with time. The delay of the model in the baseline model is very high and unstable, which is a poor performance. The fuzzy and the linear heuristic methods minimise delay slowly and have moderate variations. The PSO method ensures increased control yet there is still some variation. The hybrid PSO-Fuzzy model is the most efficient in that it exhibits a stable and low delay with good variation thus it is effective in real-time congestion control.

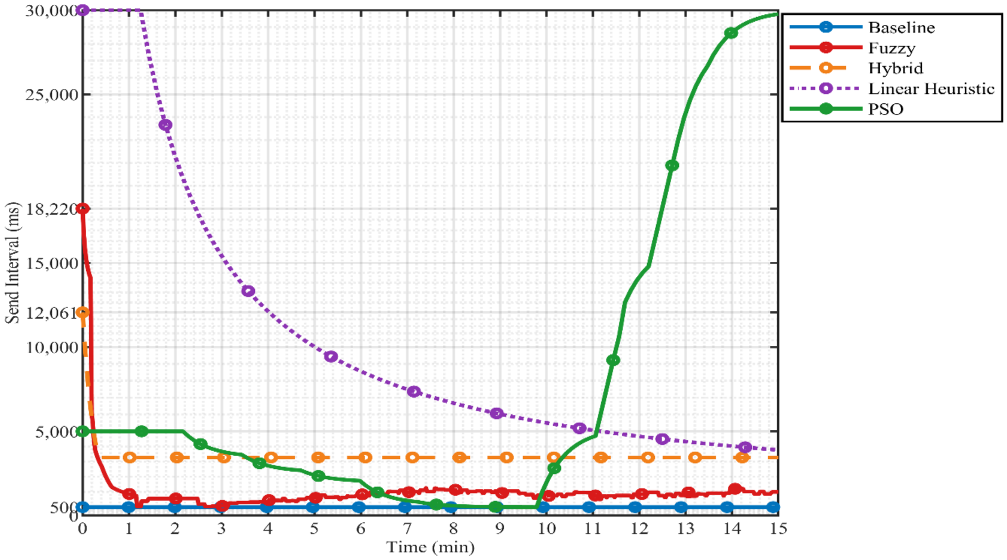


Figure 5: Comparison of send interval for all systems

The figure 5 draws a comparison between the behaviour of various congestion control methods in terms of the send interval with time. The baseline has a fixed low interval, and this causes congestion. The fuzzy and linear heuristic techniques do not change quickly, but do not hit an optimum. Large fluctuations are observed in the PSO method as a result of the optimization search. The hybrid PSO-Fuzzy model has a stable and moderate send interval, which balances the transmission rate and the congestion control, giving the network performance that is consistent and efficient.

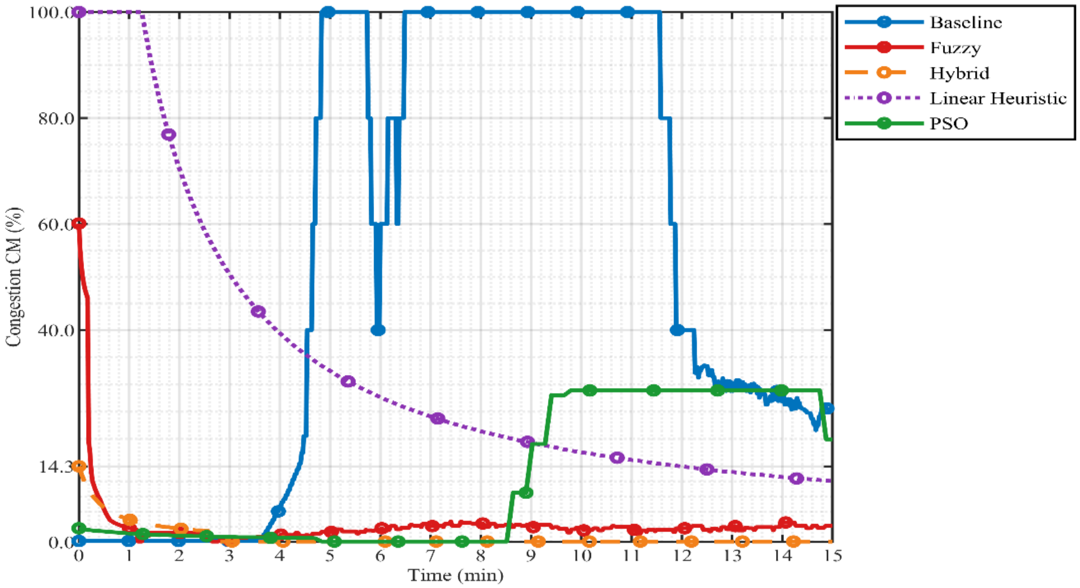


Figure 6: Comparison of the congestion metric for all systems

The figure 6 provides a comparison of the congestion measure of the various approaches over time. The congestion on the baseline is very high and unstable, and this implies poor control. The linear heuristic is inefficient and is slowly eliminating congestion. The fuzzy technique has low congestion at minimal variation. The PSO method is better in performance with moderate fluctuations. The hybrid PSO-Fuzzy model is the best one as it has the most stable and low congestion levels and manages to control congestion effectively and adaptively in changing network scenarios.

The loss results demonstrate the superior performance of the proposed hybrid model, which achieved a zero loss rate (0%) from the early stages of operation until reaching full capacity, outperforming the non-adaptive system that recorded a loss exceeding 32% and surpassing the simple adaptive approach by maintaining zero loss at the data flow rate. The fuzzy logic controller initially showed a loss value of 14.6%, which gradually improved to 0.6%, indicating a response delay. At the same time, the PSO algorithm recorded loss values ranging from 11% to 2.3%. Although the PSO algorithm temporarily achieved low loss, its loss values subsequently fluctuated, indicating a steady-state error and unstable behavior. The system was implemented using Arduino and monitored using the Blynk IoT cloud platform for real-time data display. The system was initialized for 5 seconds to ensure stability before data collection. Sensor readings from MQ and DHT11 were collected via TCP/IP, providing dual monitoring capabilities alongside a local LCD display. Experimental data, including loss, delay, congestion, and transmission time, were collected and analyzed over 15 minutes of monitoring. The results presented in table 2 compare the performance parameters of the five systems.

The non-adaptive system, the worst performing, exhibits clear failure, recording the highest loss rate (8.23%) and the highest congestion rate (52.61%). This is due to its transmission of data at a high and constant speed (500 milliseconds) without any intelligent processing, which increased the load on the system and prevented it from delivering the data. While simple adaptation succeeded in achieving a loss rate (7.5%), it did so "very slowly" and very cautiously, as it raised the transmission time to a very large value (11108 ms), which resulted in slow data updates, and this explains the large delay (113.5 ms).

Table 2: Comparing the performance standards of the five systems to demonstrate the efficiency of congestion control.

Parameters	**Non-Adaptive**	**Simple Adaptive**	**PSO**	**Fuzzy Logic**	**Hybrid PSO-Fuzzy**
DHT11	22.24 °C	21.61 °C	23.46°C	23 °C	22.93 °C
MQ	17.87%	22.33%	22.78%	18.28%	17.94%
Loss	8.23%	7.49%	2.37%	0.6%	0.30%
Delay	131.97ms	113.54ms	22.09ms	24.96ms	6.33ms
CM	52.61%	35.96%	11.71%	3.28%	0.97%
Send interval	500ms	11108.64ms	7490ms	1467.47ms	3586.30ms

Fuzzy logic also achieved the fastest and best response time ever (3.2 milliseconds) with a low loss rate, as fuzzy logic was able to find a very quick balance, settling at a response time close to the initial time (1467 ms) while intelligently preventing congestion with a congestion rate of only 3.28%. The Pso algorithm recorded a slight decrease of 2.370%. This figure indicates that the PSO algorithm was still in the "search" phase at 22 seconds and had not yet reached full stability, causing fluctuations in transmission time and CM. The superior performance of the best-balanced PSO-Fuzzy hybrid system achieved 0.3% data loss and an extremely low congestion rate (0.97%). The system has a fast transmission time (3586 ms), faster than a simple adaptive system and more stable than a standard PSO. This system achieves the optimum point that combines a fast refresh rate (17.9%) with guaranteed data delivery. The congestion metric (CM) is used to demonstrate that the hybrid system intelligently controls the network (0.97%) while the conventional system completely collapses (97%). A fuzzy system alone reacts fast to changes, but it is unable to optimize itself if network conditions change in ways that aren't anticipated by the static rules. This study finds that a PSO system alone has a high capacity to achieve the ideal transmission interval, but it has a lengthy convergence time. In the hybrid system, jitter was significantly reduced, for instance, by 20–30% when compared to single systems because the integration effectively used the fuzzy's speed to make an initial judgment, which was then modified and enhanced in real time by the PSO.

The behavior of the proposed system was also tested in different network conditions in order to determine its strength. Experiments were also done whereby various traffic loads were introduced, which included low, moderate, and high traffic congestions and varying rates of packet transmission. Also, the changes of the environmental sensor input and network delays were modeled to reflect the dynamics of the real IoT. The findings show that the Hybrid PSO -Fuzzy model always has low delay, slight packet loss, and constant transmission intervals regardless of the circumstances. Conversely, the non-adaptive and single method performs highly in the case of high congestion. This is evidenced by the fact that the proposed system is capable of effectively responding to the dynamism of network environments and, at the same time, assuring reliable performance. The discussion is summarized in table 3.

In table 3 provides a quantitative analysis of the three methods in major performance indicators. The hybrid PSO-Fuzzy model has the best performance with the lowest response time (6.33 ms), smallest jitter (1.9 ms), and lowest packet loss (0.30%), which implies high efficiency and stability. It also converges faster (9.2 s) than PSO, displaying greater adaptability to varying network conditions. Although fuzzy logic is less CPU-intensive, it is not flexible because of predetermined rules. PSO is moderate in terms of performance but has a higher computational cost. The hybrid approach is the most suitable option, as it offers optimal balance in terms of speed, stability, adaptability, and resource utilization.

Table 3: Summary of the proposed value achieved through the integration of the two technologies

Metric	**Fuzzy Logic**	**PSO**	**Hybrid PSO–Fuzzy**
Response Time (ms)	24.96	22.09	6.33
Stability (Jitter, ms)	4.8	3.2	1.9
Packet Loss (%)	0.6	2.37	0.30
Adaptability (Convergence Time, s)	Not adaptive (fixed)	18.5	9.2
CPU Load (%)	12	28	22

Ablation Study

An ablation study based on the analysis of the performance of each of the modules, as well as their combinations, was used to estimate the contribution of each of the components to the proposed system. Four configurations were taken into consideration: non-adaptive baseline, PSO only, fuzzy logic only, and hybrid PSO-Fuzzy model. The findings indicate that the fuzzy logic module would offer quick response, though not adaptable, whereas the PSO module will offer enhanced optimization, but slow convergence and instability at the initial stages. Overall the hybrid model is always the best with quick processing of decisions and adaptive optimization, resulting in the lowest delay, the least packet loss, and better stability. This discussion supports the view that the fuzzy logic and PSO elements are critical to optimal performance of congestion control.

Table 4: Ablation study analysis of individual and hybrid model performance in congestion control

Model Variant	**Response Time (ms)**	**Packet Loss (%)**	**Stability (Jitter ms)**
Non-Adaptive	131.97	8.23	6.5
Fuzzy Only	24.96	0.6	4.8
PSO Only	22.09	2.37	3.2
Hybrid PSO–Fuzzy	6.33	0.30	1.9

In table 4 shows that the quantitative findings of the ablation experiment are described. The non-adaptive model experiences the maximum delay and packet loss, which means that it is not a good congestion controller. The fuzzy-only model is more responsive but more unstable whereas the PSO-only model is more stable with an average performance. The Hybrid PSO-Fuzzy model provides the best response time, minimum packet loss, and stability, which proves the suitability of the combination between the fast-decision-making process and adaptive optimization.

Discussion

The results of this paper indicate that the suggested Hybrid PSO Fuzzy model is capable of enhancing the performance of congestion control in IoT-based wireless sensor networks. The findings show that the delay, the loss of packets and the level of congestion are reduced significantly as opposed to non-adaptive, fuzzy-only and PSO-only strategies. This demonstrates the usefulness of blending a quick fuzzy-based decision-making system with the dynamic optimization capabilities of PSO. The practical implication of the work is that it can be applied to real-time IoT systems, where the low latency and the reliability of data transmission are highly essential to the applications of environmental monitoring and smart infrastructure (Memon et al., 2025).

Nevertheless, a number of challenges and limitations are still present. The experimental test is performed on a small-scale hardware system, not necessarily reflecting large scale network implementations. The added computational cost of the PSO component may be important in the context of energy efficiency in environments with very limited resources. Fuzzy rule base is also not defined automatically and hence less flexible to highly dynamic or unpredictable environments. It also limits the scope of evaluation because of the lack of comparisons with more sophisticated machine learning-based congestion control methods.

The limitations can be overcome in future studies by expanding the proposed framework to large-scale distributed IoT settings and adding energy-conscious optimization approaches. Adaptability and decision-making can also be improved with the implementation of machine learning or reinforcement learning methods. There should also be experiments with long-term view and real-world deployment conditions to check the robustness of the system in the long run. More so, active tuning of fuzzy rules with self-learning mechanisms may enhance system generalization in various network conditions.

5 Conclusion

This study introduces a high-level hybrid congestion management system, which is a combination of fuzzy logic and particle swarm optimization (PSO) on the IoT-enabled wireless sensor networks. The results of experiments prove that the offered hybrid model is highly effective in terms of the network performance in contrast to the traditional and single-method ones. In particular, the system has a low average delay (6.33 ms) of fuzzy-only (24.96 ms) and PSO-only (22.09 ms) models. The loss of packets is lowered to 0.30% and the congestion metric (CM) is kept at 0.97% showing that congestion is efficiently managed and the network behaviour is stable. The hybrid system is also more stable as it has a less jitter (1.9 ms) and convergence (9.2 seconds) than PSO (18.5 seconds), proving its capability to quickly adjust to dynamic network conditions. Fuzzy logic integration allows making quick decisions, whereas PSO allows greater flexibility and optimization in overcoming the constraints of single methods. These findings suggest that the suggested system can be practically useful in real-time IoT environments where low latency and stable data flow are the most crucial considerations. Nevertheless, the research is confined to an experimental small-scale and short-term analysis. Future studies can be aimed at generalizing the model to large-scale distributed IoT networks, energy-efficient optimization methods, and using machine learning or reinforcement learning methods to achieve more flexibility. The strength and scalability of the systems can also be enhanced by long-term real world application and dynamical tuning of fuzzy rules.

References

[1] Abdullah, J. S., Aal-Nouman, M. I., & Al-Fuhaidi, B. (2025). Efficient Coverage of Sensors in a WSN Using a modified Hybrid PSO And ALO Algorithms. *Iraqi Journal of Information and Communication Technology*, *8*(2), 29-41. https://doi.org/10.31987/ijict.8.2.309

[2] Alikhani, R., Perfilieva, I., & Ganjeh-Alamdari, M. (2022). Trigonometric F mn-transform of multi-variable functions and its application to the partial differential equations and image processing: R. Alikhani et al. *Soft Computing*, *26*(24), 13301-13331. https://doi.org/10.1007/s00500-022-07481-2

[3] Al-Kashoash, H. A., Kharrufa, H., Al-Nidawi, Y., & Kemp, A. H. (2019). Congestion control in wireless sensor and 6LoWPAN networks: Toward the Internet of Things. *Wireless Networks*, *25*(8), 4493-4522. https://doi.org/10.1007/s11276-018-1743-y

[4] Goel, S., Sharma, K. P., Mittal, N., Mittal, V., Khafaga, D. S., Aldakheel, E. A., & Abouhawwash, M. (2026). Energy efficient clustering protocol in wireless sensor networks using an adaptive hybrid optimization algorithm. *Scientific Reports*, *16*, 6300. https://doi.org/10.1038/s41598-026-36957-6

[5] Jayaudhaya, J., Supriya, S., Kandaswamy, V. A., Pandi, S., Kamatchi, S., & Priya, C. P. (2023, May). ACoCo: An adaptive congestion control approach for enhancing CoAP performance in IoT network. In *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)* (pp. 189-194). IEEE. https://doi.org/10.1109/ICAAIC56838.2023.10141283

[6] Kazmi, H. S. Z., Javaid, N., Imran, M., & Outay, F. (2019). Congestion control in wireless sensor networks based on support vector machine, Grey Wolf optimization and differential evolution. In *2019 wireless days (WD)* (pp. 1-8). IEEE. https://doi.org/10.1109/WD.2019.8734265

[7] Liao, L., Leung, V. C., Li, Z., & Chao, H. C. (2021). Genetic algorithms with variant particle swarm optimization based mutation for generic controller placement in software-defined networks. *Symmetry*, *13*(7), 1133. https://doi.org/10.3390/sym13071133

[8] Lim, C. (2019). A survey on congestion control for RPL-based wireless sensor networks. *Sensors*, *19*(11), 2567. https://doi.org/10.3390/s19112567

[9] Manshahia, M. S. (2017). Water wave optimization algorithm based congestion control and quality of service improvement in wireless sensor networks. *Transactions on Networks and Communications*, *5*(4), 31. https://doi.org/10.14738/tnc.54.3567

[10] Memon, S. A., Andriukaitis, D., Navikas, D., Markevičius, V., Valinevičius, A., Žilys, M., ... & Klimenta, D. (2025). Hybrid GA-PSO Optimization for Controller Placement in Large-Scale Smart City IoT Networks. *Sensors*, *25*(23), 7119. https://doi.org/10.3390/s25237119

[11] Nahar, A. K. (2024). A Retina-inspired Sampling Method for Coding and Compression for Enhancement Image Based on Discrete Double Density Wavelet Transformer. *IRAQI Journal of Computers, Communications, Control and Systems Engineering*, *24*(2), 54-64. https://doi.org/10.33103/uot.ijccce.24.2.5

[12] Narawade, V. E., & Kolekar, U. D. (2017, February). Eacsro: epsilon constraint-based adaptive cuckoo search algorithm for rate optimized congestion avoidance and control in wireless sensor networks. In *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)* (pp. 715-720). IEEE. https://doi.org/10.1109/I-SMAC.2017.8058272

[13] Noori, M., & Altabey, W. A. (2022). Hysteresis in engineering systems. *Applied Sciences*, *12*(19), 9428. https://doi.org/10.3390/app12199428

[14] Panimalar, S., & Jacob, T. P. (2020). A comparative study of hybrid optimized algorithms for congestion control in wireless sensor network. In *2020 International Conference on Communication and Signal Processing (ICCSP)* (pp. 889-893). IEEE. https://doi.org/10.1109/ICCSP48568.2020.9182194

[15] Pawar, A. S., Shirode, U. R., Vhatkar, K. N., Sontakke, P. V., & Sarwade, J. M. (2025). Development of Hybrid Adaptive and Attention Network with Enhanced Optimization for Traffic Congestion Control in the IoT Environment. *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, 1-30. https://doi.org/10.1007/s40998-025-00949-5

[16] Pentapalli, V. V. G., & Varma, R. K. (2016). Cuckoo search optimization and its applications: A review. *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, *5*(11), 556-562. https://doi.org/10.17148/IJARCCE.2016.511119

[17] Samarji, N., & Salamah, M. (2021). A fault tolerance metaheuristic-based scheme for controller placement problem in wireless software-defined networks. *International Journal of Communication Systems*, *34*(4), e4624. https://doi.org/10.1002/dac.4624

[18] Singh, K., Singh, K., & Aziz, A. (2018). Congestion control in wireless sensor networks by hybrid multi-objective optimization algorithm. *Computer Networks*, *138*, 90-107. https://doi.org/10.1016/j.comnet.2018.03.023

[19] Sukjaimuk, R., Nguyen, Q. N., & Sato, T. (2018). A smart congestion control mechanism for the green IoT sensor-enabled information-centric networking. *Sensors*, *18*(9), 2889. https://doi.org/10.3390/s18092889

[20] Tam, N. T., Hai, D. T., Son, L. H., & Vinh, L. T. (2018). Improving lifetime and network connections of 3D wireless sensor networks based on fuzzy clustering and particle swarm optimization. *Wireless Networks*, *24*(5), 1477-1490. https://doi.org/10.1007/s11276-016-1412-y

[21] Tawfeek, M., Alrashdi, I., Alruwaili, M., & Talaat, F. (2025). A fuzzy multi-objective framework for energy optimization and reliable routing in wireless sensor networks via particle swarm optimization. *Computers, Materials, & Continua*, *83*(2), 2773. https://doi.org/10.32604/cmc.2025.061773

[22] Zhuo, S., Shokri-Ghadikolaei, H., Fischione, C., & Wang, Z. (2019). Online congestion measurement and control in cognitive wireless sensor networks. *IEEE Access*, *7*, 137704-137719. https://doi.org/10.1109/ACCESS.2019.2943011

Authors Biography

Elham Ibrahim Mahmoud, is an M.Sc. student in Electronic and Control Engineering at the Technical Engineering College, Kirkuk, Northern Technical University, Iraq. Her research interests include Internet of Things (IoT), wireless sensor networks, fuzzy logic, and intelligent congestion control systems.

Dr. Montassar Aidi Sharif, has a PhD degree in Mechatronics Engineering from Michigan State University. And, he is a faculty member at the Northern Technical University, currently serving in the Department of Artificial Intelligence Engineering at the Technical Engineering College for Computer and AI. And Dr. Sharif is the Dean of the Technical Engineering college of Computer and AI / Kirkuk. Research interests focus on artificial intelligence applications and technical engineering developments. Contributing to academic excellence through both teaching and specialized research in the field of AI and computer systems

Dr. Fatih Korkmaz holds a PhD in Electrical Engineering with a specialization in Electronics. He is a senior faculty member at Çankırı Karatekin University in Türkiye, where he contributes to the Department of Electrical and Electronics Engineering. His academic career is distinguished by extensive research in electronics and power systems, reflected in his numerous contributions to international journals and conferences. Dr. Korkmaz is committed to advancing the field of electrical engineering through dedicated teaching and innovative research projects, focusing on high-impact technical developments.