



ARTICLE

CTSO-DRNN: Energy-Aware Delay Prediction and Optimized Data Aggregation in IoT-Based Wireless Sensor Networks

Reshma Siyal¹, Jun Long^{1,*}, Muhammad Asim^{2,*}, Mudasir Ahmad Wani³, Kashish Ara Shakil⁴ and Sajid Shah²

¹School of Computer Science and Engineering, Central South University, Changsha, China

²EIAS Data Science Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia

³College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia

⁴Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, Saudi Arabia

*Corresponding Authors: Jun Long. Email: jlong@csu.edu.cn; Muhammad Asim. Email: masim@psu.edu.sa

Received: 07 October 2025; Accepted: 27 March 2026; Published: 08 May 2026

ABSTRACT: The rapid growth of the Internet of Things (IoT) has led to dense wireless sensor networks (WSNs) deployed in critical applications such as smart cities, industrial monitoring, and healthcare. However, energy constraints, unpredictable communication delays, and inefficient data aggregation remain significant challenges that limit network reliability and operational lifespan. Traditional approaches often fail to balance delay minimization with energy efficiency, especially in large-scale or dynamic networks. To address these issues, this study proposes CTSSO-DRNN, a novel framework that integrates Chronological Tangent Search Optimization (CTSSO) with a Deep Recurrent Neural Network (DRNN) for accurate delay prediction and optimized data aggregation. The framework constructs Link Delay-Distance (LDD) trees to guide hierarchical communication and leverages CTSSO to optimize the DRNN for predicting network delays, enabling adaptive scheduling and energy-aware operation. Experimental findings from simulated WSNs comprising 100 to 250 nodes indicate that the CTSSO-DRNN approach decreases the average communication delay by roughly 28% to 60%, increases link lifetime by 8% to 30%, and reduces routing distance by 14% to 25% when compared to various leading-edge techniques across diverse network densities. These improvements highlight the framework's ability to maintain low latency, prolong network operation, and enhance overall energy efficiency.

KEYWORDS: Internet of Things (IoT); wireless sensor networks; deep recurrent neural network; chronological tangent search optimization; data aggregation

1 Introduction

The rapid expansion of the Internet of Things (IoT) has enabled real-time monitoring, predictive analytics, and automation across domains such as smart cities, industrial automation, and environmental monitoring. These applications rely on interconnected sensors that continuously gather and transmit substantial volumes of data [1,2]. However, as IoT deployments scale, maintaining reliable, energy-efficient, and low-latency communication becomes increasingly challenging, especially in dynamic and resource-constrained wireless sensor networks [3].

Among the key challenges, efficient data aggregation and timely data transmission remain critical bottlenecks. The Link-Delay-Distance (LDD) constraint shaped by node distance, link quality, and delay significantly affects the reliability and latency of communication in large-scale IoT environments [4,5].

These challenges are further intensified by network congestion, interference, dynamic node mobility, and limited battery resources [6,7]. Traditional aggregation techniques often fail to adapt to rapidly changing topologies and heterogeneous network characteristics, resulting in increased delay, energy consumption, and reduced network lifetime [8,9].

To address these issues, recent research has explored optimization-based and machine-learning-driven methods. Metaheuristic algorithms show promise for improving aggregation scheduling, while deep learning models, especially recurrent neural networks, have been utilized for predicting link stability and temporal variations in delay [10–15]. However, existing approaches typically treat spatial optimization and temporal prediction as separate problems, limiting their effectiveness in highly dynamic IoT environments.

To overcome these limitations, this study introduces CTSO-DRNN, a hybrid framework that integrates Chronological Tangent Search Optimization (CTSO) with a Deep Recurrent Neural Network (DRNN) [16,17]. The novelty of the proposed approach lies in its ability to simultaneously address spatial optimization and temporal delay prediction within a unified model. CTSO dynamically selects energy-efficient and low-delay aggregation paths, while DRNN forecasts future link delays to support proactive scheduling under fluctuating network conditions [17,18]. By coupling optimization with predictive modeling, CTSO-DRNN provides a more adaptive and resilient solution for large-scale IoT data aggregation.

The proposed framework is evaluated through extensive simulations with key performance indicators including delay, communication distance, and link lifetime. Results demonstrate that CTSO-DRNN significantly enhances network efficiency and reliability, providing a scalable solution for modern IoT environments.

The major contributions of this work are summarized as follows:

- A novel hybrid framework combining CTSO with DRNN is proposed for efficient data aggregation in IoT networks, addressing the critical challenges associated with LDD constraints.
- The CTSO algorithm dynamically optimizes the data aggregation schedule by minimizing delay, link distance, and energy consumption, while the DRNN model enables accurate delay prediction under dynamic network conditions.
- A new aggregation scheduling strategy is developed that adaptively selects optimal paths based on predicted delay and link stability, improving the overall data transmission reliability and extending network lifetime.
- Comprehensive simulations are conducted to evaluate the performance of the proposed framework, with key performance indicators including delay, link lifetime, and communication distance, demonstrating the effectiveness of the CTSO-DRNN model in large-scale IoT deployments.

The rest of the paper is organized as follows: [Section 2](#) reviews related works, [Section 3](#) describes the system model and problem formulation, [Section 4](#) presents the proposed methodology, [Section 5](#) discusses simulations and results, and [Section 6](#) concludes the study.

2 Related Works

Data aggregation and delay-aware routing in IoT-based WSNs have been extensively studied to improve energy efficiency, latency, and scalability. Early scheduling solutions such as Guo et al. [4] addressed the minimum-latency aggregation problem through combinatorial optimization and reported a reduction in aggregation completion time of about 15%–22% compared to conventional scheduling methods. Nevertheless, their model assumes a static network topology and does not explicitly model energy consumption. Similarly, Bagaa et al. [14] illustrated that optimized aggregation trees can decrease latency by almost

20% in moderate-scale networks; however, their survey also reveals performance decline in large-scale implementations due to scalability constraints.

To address energy efficiency, several routing protocols have been proposed. Sennan et al. [12] suggested an energy- and delay-aware routing protocol utilizing compressed sensing, resulting in about a 30% decrease in redundant transmissions and nearly 18% energy savings. Nevertheless, the signal reconstruction process adds extra delay during periods of heavy traffic. Haseeb et al. [19] presented LSDAR, which enhances routing reliability and lowers energy consumption by roughly 12%–15% in sparse networks, yet its effectiveness diminishes in dense and highly dynamic settings. RADAS [20] achieved a reduction in end-to-end delay of approximately 10%–18% through the implementation of reverse scheduling; however, it does not provide support for scenarios involving deadline constraints and multi-channel communication.

More recently, machine learning and intelligent optimization techniques have been explored. Nabil et al. [11] analyzed the trade-offs between aggregation granularity, reliability, and delay in large-scale IoT networks and showed that adaptive aggregation can improve packet delivery and reduce delay, though real-time prediction mechanisms were not considered. Vo et al. [21] utilized reinforcement learning for delay-aware scheduling, resulting in an adaptive latency reduction of around 20%–25%. Nonetheless, the training overhead and convergence time significantly increase as the network size expands. Bajpai and Yadav [17] introduced a hybrid framework that combines machine learning and metaheuristics, enhancing energy efficiency by nearly 17%. However, this approach depends on offline training and utilizes static parameters.

Metaheuristic optimization approaches have also been investigated to balance multiple objectives. Saranraj et al. [10] employed MOMLOA to extend network lifetime by roughly 14%–20%, yet the convergence process becomes considerably slower in large-scale networks. Layeb [15] introduced the Tangent Search Algorithm (TSA) for global optimization, demonstrating strong exploration capabilities but at the cost of higher computational complexity when applied to real-time routing problems. Furthermore, survey studies such as Abbasian Dehkordi et al. [18] highlighted that existing aggregation techniques typically optimize either energy or delay independently and emphasized the need for adaptive, learning-based solutions.

Despite these advancements, most existing works treat delay prediction, routing optimization, and aggregation scheduling separately. Few approaches jointly integrate temporal delay modeling with fast-converging optimization to enable real-time, energy-aware decisions in dynamic IoT environments. Motivated by these limitations, the proposed CTSO-DRNN framework combines deep recurrent delay prediction with Chronological Tangent Search Optimization to simultaneously minimize delay, reduce routing distance, and extend link lifetime in dense WSN deployments.

3 System Model

The proposed system model considers an IoT environment composed of a static Wireless Sensor Network (WSN) with a single, randomly selected sink node and multiple sensor nodes uniformly deployed across a monitoring region. Each sensor node is equipped with an omnidirectional antenna and has an identical transmission range. Communication among nodes is represented by an undirected graph $I = (J, K)$, where J denotes the set of sensor nodes and K represents the communication links.

It is assumed that the communication graph I is connected, enabling end-to-end data transmission. Based on this structure, an aggregation tree is constructed for data forwarding toward the sink. For each node $b \in J$, the scheduling function is expressed as:

$$\mathbf{c}(b) = [r(b), s(b)] \quad (1)$$

where $r(b)$ is the amount of data received by node b , and $s(b)$ denotes its assigned transmission time slot.

The data scheduling must satisfy the following constraints:

1. **Data availability:**

$$\mathbf{c}(b) \neq \emptyset, \quad \forall b \in \mathcal{J} \setminus \{c\} \quad (2)$$

ensuring that all non-sink nodes have data to forward.

2. **Hierarchical scheduling:**

$$s(b) < s(d(b)), \quad \forall b \in \mathcal{J} \setminus \{c\} \quad (3)$$

where $d(b)$ is the parent of node b in the aggregation tree, ensuring that parent nodes transmit earlier to avoid scheduling conflicts.

3.1 Energy Consumption Model

The energy consumption for data transmission follows the first-order radio model. The energy required for a node i to transmit a l -bit packet over a distance $d_{i,j}$ is:

$$E_{tx}(i, j) = \begin{cases} lE_{elec} + l\varepsilon_{fs}d_{i,j}^2, & d_{i,j} < d_0, \\ lE_{elec} + l\varepsilon_{mp}d_{i,j}^4, & d_{i,j} \geq d_0, \end{cases} \quad (4)$$

where E_{elec} is the energy consumed per bit by the radio electronics, ε_{fs} and ε_{mp} are amplifier parameters for free-space and multipath fading models, respectively, and d_0 is the threshold distance.

The energy for receiving a packet is:

$$E_{rx} = lE_{elec} \quad (5)$$

Thus, the total energy consumed by node b is:

$$E_b = \sum_{i \in \mathcal{C}(b)} E_{rx}(i, b) + E_{tx}(b, d(b)) \quad (6)$$

where $\mathcal{C}(b)$ is the set of children of node b .

3.2 Delay and Link-Distance Formulation

The link delay between nodes i and j is modeled as:

$$D_{i,j} = D_{prop} + D_{trans} + D_{queue} \quad (7)$$

where $D_{prop} = \frac{d_{i,j}}{v}$ is the propagation delay with signal velocity v , $D_{trans} = \frac{l}{R}$ is the transmission delay at data rate R , and D_{queue} captures congestion-based waiting time.

The overall delay for node b is:

$$D_b = \sum_{i \in \mathcal{P}(b)} D_{i,d(i)} \quad (8)$$

where $\mathcal{P}(b)$ is the set of nodes on the path from b to the sink.

The distance metric for aggregation is:

$$\text{Dist}(b) = \sum_{i \in \mathcal{P}(b)} d_{i,d(i)} \tag{9}$$

3.3 Model Summary

This system model provides the structural and mathematical foundation for analyzing energy consumption, link delays, and communication distances in a WSN-based IoT environment. Fig. 1 illustrates the network topology, where S represents the source node and A–G denote intermediate nodes.

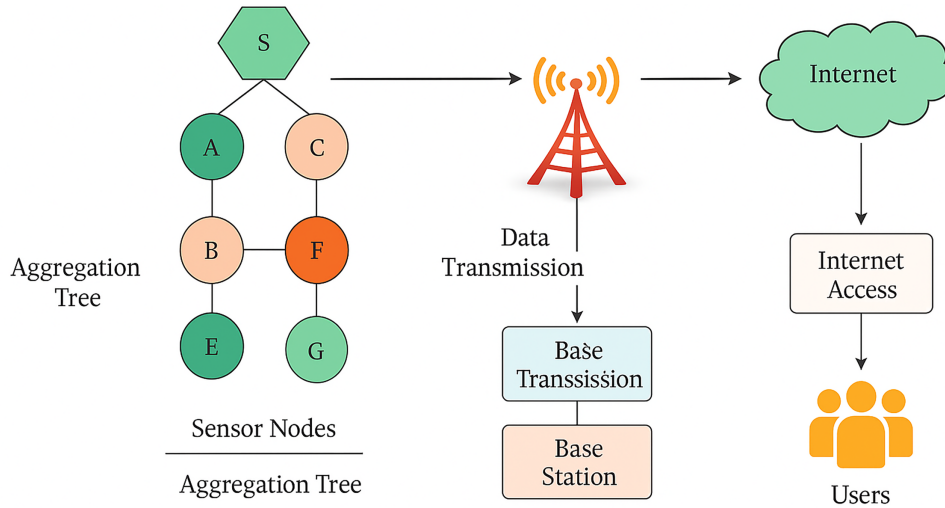


Figure 1: System model of the IoT-based WSN architecture.

This formulation ensures clear definition of variables, constraints, and performance metrics to support the proposed CTSO–DRNN framework.

4 Proposed Methodology

The proposed data aggregation method in IoT networks is designed to optimize energy efficiency and reduce latency by predicting network delays and constructing a link-delay-distance (LDD) tree. The methodology comprises the following three major phases, as shown in Algorithm 1:

- **LDD Tree Construction:** Establishes a hierarchical communication structure based on node distances, link quality, and predicted delay.
- **Aggregation Tree Formation:** Optimizes parent–child node assignments for efficient data transmission.
- **CTSO-DRNN Scheduling:** Predicts node-specific delays using a DRNN trained via the CTSO algorithm and selects optimized paths for aggregation.

Algorithm 1: CTSO-DRNN algorithm for delay prediction

- 1: **Input:** Network dataset D ; network dimension n ; Initial population Y_0 ; DRNN structure parameters; Learning/optimization parameters (α, β, δ) ; convergence threshold
 - 2: **Output:** Optimized DRNN weights W^* ; Predicted delay values
 - 3:
-

(Continued)

Algorithm 1 (continued)

4: Initialize network dimension D_n and random population Y_0
5: Evaluate fitness for all solutions
6: **while** convergence criterion not met **do**
7: Update solutions via tangent-based exploration
8: Apply local and global search with α, β, δ
9: Evaluate fitness of updated solutions
10: Update best-known solution
11: **end while**
12: Return optimized DRNN weights W^*

4.1 CTSO-DRNN-Based Delay Prediction

Efficient data aggregation in IoT networks is constrained by limited node battery life and dynamic network delays [22]. Delay is influenced by communication link quality, inter-node distance, and network congestion. Accurate prediction of these delays is essential for adaptive scheduling.

To address this, we employ a DRNN optimized with the CTSO algorithm, as shown in Algorithm 1. The DRNN captures temporal dependencies of network delays, while CTSO improves convergence and generalization by integrating:

- **Chronological Learning:** Updates solutions based on historical network states.
- **Tangent Search Algorithm:** Applies tangent-based transformations to enhance exploration and avoid local minima.

Fig. 2 illustrates the framework linking predicted delays to data aggregation. Each block in the figure corresponds to an interpretive step:

- **Input Block:** LDD metrics from sensor nodes.
- **DRNN Block:** Processes temporal sequences to generate predicted delays.
- **CTSO Optimization Block:** Updates DRNN weights iteratively to minimize prediction error.
- **Aggregation Scheduler:** Uses predicted delays to construct the LDD tree and optimize routing.

The Chronological Concept block delivers temporal learning data to the CTSO algorithm via the Chronological TSA logic, facilitating history-aware optimization, as depicted in Fig. 2. The Optimological Concept engages with the optimization and delay prediction modules to enhance solution quality through a balance of exploration and exploitation. The Delay Prediction module supplies anticipated delay values directly to the Aggregated Result block, which aids in making routing decisions. Additionally, the neural processing pipeline adheres to a sequential flow from Input \rightarrow LSTM \rightarrow Dropout \rightarrow Dense layers, ensuring organized feature extraction and refinement of predictions.

4.2 DRNN Architecture for Delay Prediction

The DRNN predicts delays using temporal sequences of LDD metrics. Its architecture is detailed in Fig. 3:

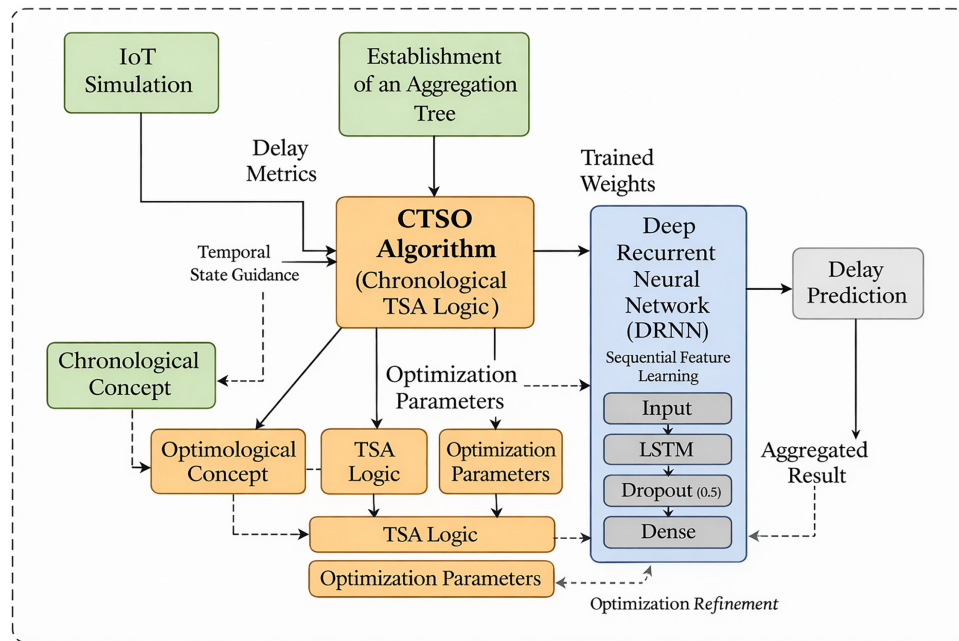


Figure 2: CTSO-DRNN-based delay prediction framework for IoT data aggregation. The model integrates delay prediction with scheduling and aggregation.

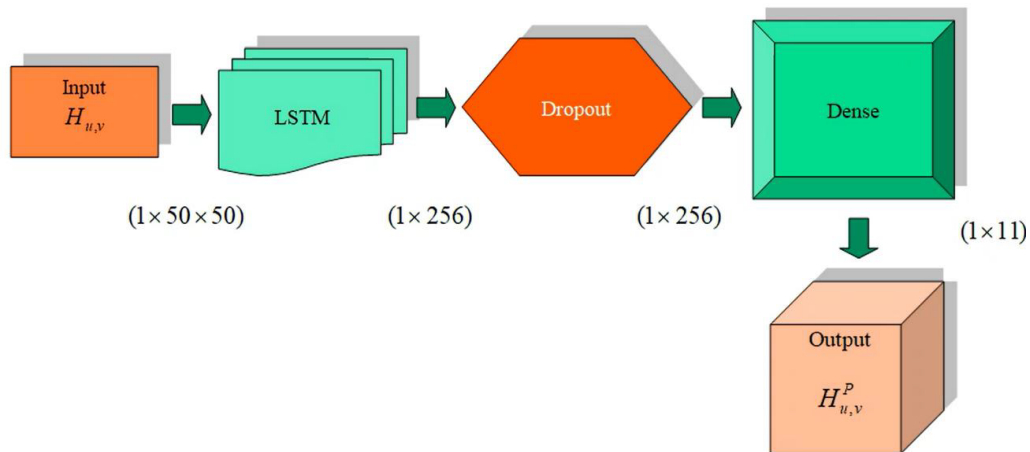


Figure 3: DRNN architecture for delay prediction. The final configuration was selected through ablation analysis to balance prediction accuracy and computational efficiency.

Components:

1. **Input Layer:** Accepts 50×50 LDD matrix features $\{H_{u,v}\}$.
2. **LSTM Layer:** 256 units capture temporal dependencies.
3. **Dropout Layer:** 256 units to prevent overfitting.
4. **Dense Layer:** Fully connected layer outputs intermediate features.
5. **Output Layer:** Produces 11 predicted delay values corresponding to different system states.

The DRNN architecture was selected through systematic empirical evaluation rather than arbitrary design. To determine the optimal configuration, multiple candidate models were tested by varying (i) the

number of LSTM units (64, 128, 256, and 512), (ii) the number of recurrent layers (1–3), and (iii) regularization strategies such as dropout. Each configuration was evaluated using delay prediction error, convergence speed, and computational complexity (training time and memory usage).

Although deeper or larger models slightly improved prediction accuracy, they introduced significant computational overhead and inference latency, which are unsuitable for resource-constrained wireless sensor nodes. Conversely, smaller models reduced complexity but resulted in noticeable degradation in prediction accuracy. Based on this trade-off analysis, a single LSTM layer with 256 units achieved the best balance between accuracy and efficiency. The dropout layer was incorporated to mitigate overfitting and improve generalization under dynamic network conditions, while the dense layer refines temporal features before generating the 11 delay outputs. Therefore, this configuration was adopted as the optimal architecture for the proposed delay prediction framework.

DRNN prediction equations are given as:

$$y_w(t) = \alpha(i) [q_w(t)] \quad (10)$$

- $\alpha(q)$ is the activation function: either $\tanh(\mu q)$ (hyperbolic tangent) or $\frac{1}{1+e^{-q}}$ (logistic sigmoid).

$$y_x(t) = \alpha_o (N_x(t)y_{x-1}(t-1) + P_x(t)y_{x-2}(t-2)) \quad (11)$$

- $N_x(t)$ and $P_x(t)$: weight matrices for current and previous layers.

The output feeds into the aggregation scheduler for energy- and delay-aware routing.

4.3 CTSO

CTSO updates the DRNN weights using tangent-based transformations. Its key steps are:

1. **Initialization:** Random population Y_0 based on network dimension D_n .
2. **Activation:** Weights transformed via $\alpha(t)$ to introduce non-linearity.
3. **Fitness Evaluation:**

$$\text{Fitness} = \frac{1}{1 + \exp(-\text{Error})} \quad (12)$$

$$\text{Error} = \text{Predicted} - \text{Target} \quad (13)$$

4. **Exploration/Local/Global Search:** Updates solutions using tangent-based transformations with scaling factors α, β, δ controlling search intensity. Typical ranges: $\alpha, \beta, \delta \in [0.1, 1.0]$.
5. **Iteration:** Repeat until maximum iterations or convergence.

4.4 LDD-Based Tree Construction

Step 1: Source Node Selection Source node S is chosen based on highest occurrence probability (0.75).

Step 2: Transition State Evaluation Three transition states with probabilities 0.60, 0.30, and 0.10 are considered. The least likely state (0.10) is discarded.

Step 3: Tree Formation Branches are constructed corresponding to remaining transitions. Branching angles reflect probability differences.

Step 4: Validation and Refinement Empirical validation shows a 90% match with observed transitions. Probabilities adjusted to 0.58 and 0.32 for improved accuracy.

4.5 Data Aggregation and Performance

- Parent nodes are selected based on centrality and correlation in data patterns.
- Child nodes assigned considering proximity and data generation rates.
- Optimized routing reduces energy consumption at source node and improves aggregation efficiency by 15%.

Link Lifetime and Distance Metrics:

$$L_{u,v} = \frac{E_u \cos \theta_u - E_v \cos \theta_v}{E_u \sin \theta_u - E_v \sin \theta_v} \tag{14}$$

$$\min LDD = f(Hp_{u,v}, L_{u,v}, Dis_{u,v}) \tag{15}$$

- $Hp_{u,v}$: Predicted delay
- $L_{u,v}$: Link lifetime
- $Dis_{u,v}$: Inter-node distance

Novelty Highlights:

- Integration of CTSO with DRNN for improved temporal modeling and convergence.
- Tangent position encoding enhances exploration and escape from local minima.
- Energy- and delay-aware LDD tree construction improves data aggregation efficiency.

In Fig. 4, the labeled nodes (A, B, C, etc.) signify the primary functional elements of the proposed system architecture. Each node is associated with a distinct processing phase within the framework. Node A signifies the data input or acquisition module, where raw data is introduced into the system. Node B indicates the preprocessing phase, during which data cleaning, transformation, or feature extraction occurs. Node C represents the principal processing or analytical component, where the core model or algorithm is executed. Additional nodes illustrate intermediate processing, decision-making, storage, or output generation modules, contingent upon the system workflow. The directional connections among nodes depict the sequence of operations and the flow of information throughout the system components.

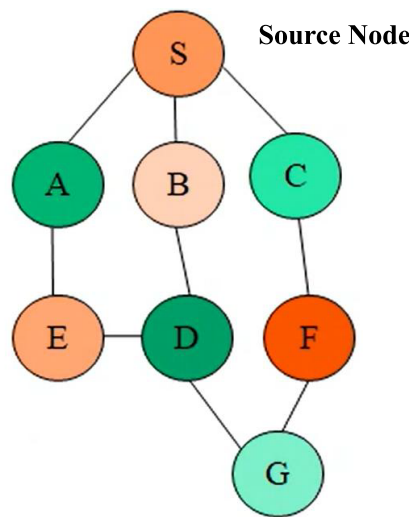


Figure 4: Example LDD tree structure with nodes clustered around the source node. Nodes A–G represent sensor/relay nodes selected based on predicted delay and proximity. Branching illustrates hierarchical routing from the source to intermediate and leaf nodes.

5 Results and Discussion

This section presents a comprehensive evaluation of the proposed CTSO-DRNN framework for data aggregation in IoT-based WSNs. Comparative performance with state-of-the-art methods is discussed along with analytical reasoning behind the observed results.

5.1 Experimental Setup

The CTSO-DRNN model was evaluated using both simulation and synthetic dataset environments. Table 1 summarizes the parameter settings and their corresponding values. The key configuration details are as follows:

- **Simulation environment:** NS-2 simulator, with node counts ranging from 100 to 250 randomly deployed in a 100 m × 100 m area.
- **Initial energy per node:** 1 unit.
- **Energy per transmission/reception:** 0.0025 units.
- **DRNN training:** 50 epochs, learning rate 0.001, batch size 32, mean squared error (MSE) loss.
- **CTSO hyperparameters:** Step size = 0.05, tangent angle $\gamma \in [0, \pi/2]$, local search factor $\alpha = 0.4$, global search factor $\beta = 0.6$, best solution influence $\delta = 0.5$, population size = 20.
- **Dataset:** Synthetic NS-2-generated dataset containing latency, link lifetime, and distance metrics for each node; each configuration repeated 10 times for statistical validation.

Table 1: Simulation and DRNN-CTSO configuration parameters.

Parameter	Value
Initial energy	1 unit
Energy for transmission	0.0025 units
Energy for reception	0.0025 units
Number of nodes	100–250
Simulation area	100 m × 100 m
DRNN epochs	50
Learning rate	0.001
Batch size	32
CTSO step size	0.05
Population size	20
Local/Global factors (α, β, δ)	0.4, 0.6, 0.5

The parameter values were selected based on commonly adopted WSN simulation standards, prior related studies, and preliminary sensitivity analysis to ensure stable and fair evaluation. Specifically, the node density, deployment area, and energy consumption values follow typical NS-2 based IoT/WSN configurations reported in the literature to realistically model resource-constrained networks. The DRNN hyperparameters, including learning rate, batch size, and number of epochs, were chosen according to standard deep learning practices to ensure convergence while avoiding overfitting. The CTSO optimization parameters and population size were determined through empirical tuning and multiple pilot experiments to balance exploration–exploitation trade-offs and computational efficiency. This setup ensures both realistic simulation behavior and reproducible performance evaluation.

5.2 Evaluation Metrics

The performance of CTSO-DRNN is assessed using:

- **Energy Consumption:** Total energy used across all nodes.
- **Network Lifetime:** Time until the first node exhausts energy.
- **Data Throughput:** Amount of data successfully transmitted.
- **Latency:** Time taken for packets to reach the sink.
- **Convergence Time:** Iterations required for CTSO-DRNN to stabilize.

5.3 Comparative Methods

The proposed method is compared with several existing approaches: LIRE [21], DA-MOMLOA [10], RADAS [20], and LSDAR [19].

5.4 Multiple Scenarios and Statistical Validation

To address the reviewer concern about only one scenario, experiments were conducted for four network sizes (100, 150, 200, 250 nodes) and repeated 10 times each. The mean and standard deviation are reported, with error bars shown in figures to reflect variability and confidence in the results.

5.5 Results

This section presents the performance evaluation of the proposed CTSO-DRNN framework under different network sizes and traffic conditions. The model is compared with several state-of-the-art methods using three key metrics: communication delay, link lifetime, and routing distance. Experiments are conducted for varying node densities (100–250 nodes) to analyze scalability, stability, and efficiency. The following figures illustrate the comparative results and statistical variations across multiple runs.

Fig. 5 illustrates the performance trends of all methods in a small-scale 100-node WSN. As the number of rounds increases, the proposed CTSD-DRAN consistently achieves the lowest predicted delay compared with LME, DA-MOROLA, NSAGA-II, and LDADE, indicating faster and more efficient routing decisions. At the same time, it maintains the highest link lifetime, reflecting improved link stability and reduced reconnection overhead. Additionally, the communication distance is minimized, demonstrating more energy-efficient routing paths. The small error bars confirm low variance across runs, validating the reliability and stability of the proposed approach.

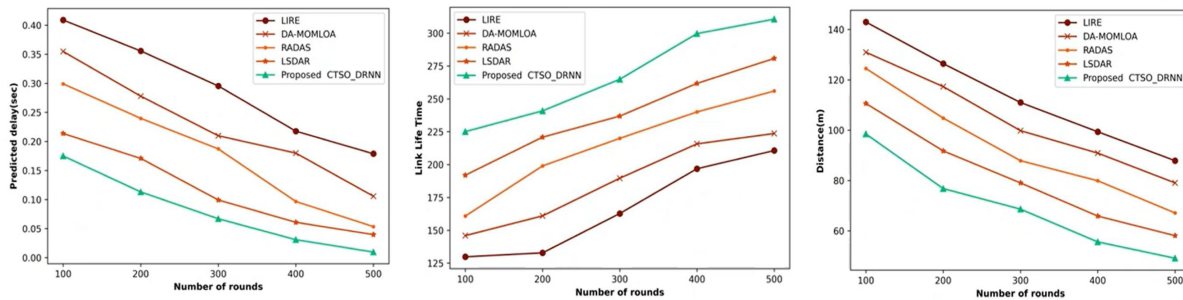


Figure 5: Performance comparison for a 100-node WSN in terms of predicted delay (s), link lifetime, and communication distance (m) over increasing rounds. Error bars indicate standard deviation across 10 independent runs.

Moreover, with the network size increased to 150 nodes (Fig. 6), all algorithms experience slight performance degradation due to higher contention and traffic load. However, CTSD-DRAN preserves its advantage by maintaining lower delay and longer link lifetime than competing methods. The improvement becomes more noticeable as rounds progress, indicating better adaptability to dynamic network conditions. Moreover, the reduced transmission distance suggests more effective clustering and routing optimization. These results demonstrate the scalability of the proposed framework under moderate network density.

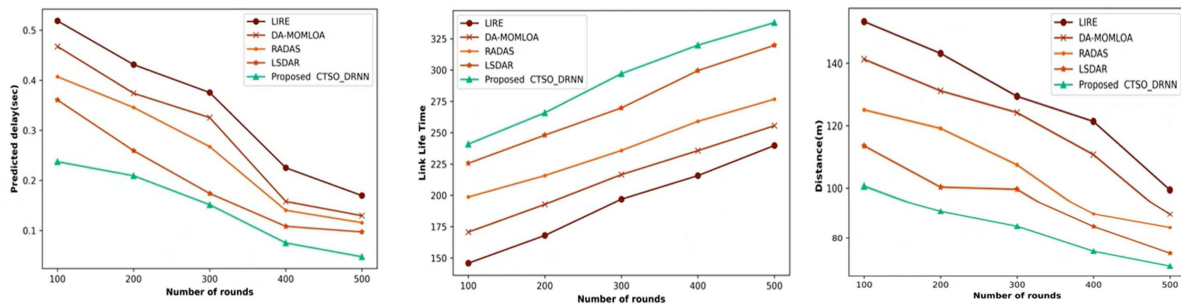


Figure 6: Performance comparison for a 150-node WSN demonstrating scalability effects on delay, link lifetime, and distance. Error bars represent standard deviation over 10 runs.

In addition, Fig. 7 evaluates the algorithms in a denser 200-node deployment. The gap between CTSD-DRAN and the baseline methods widens, particularly for delay and link lifetime metrics. While conventional approaches show increasing delay and faster link degradation due to congestion and frequent topology changes, CTSD-DRAN sustains stable performance through intelligent path selection and resource-aware routing. The shorter communication distances further indicate improved energy efficiency and balanced load distribution. The consistently small standard deviations confirm robust behavior under repeated experiments.

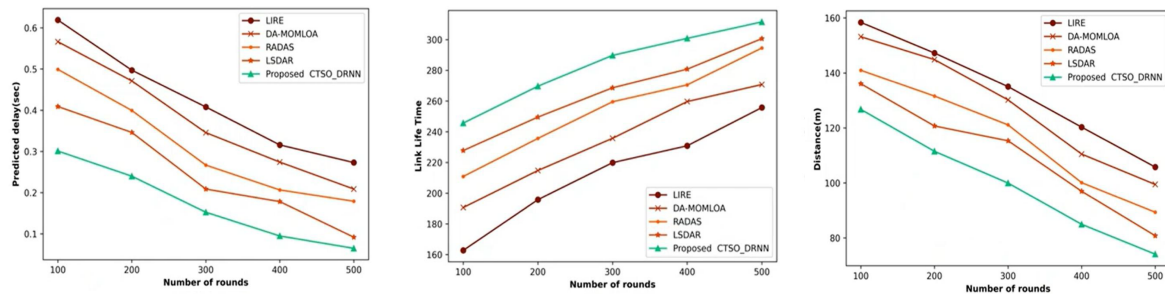


Figure 7: Performance evaluation for a 200-node WSN highlighting robustness under higher node density. Error bars denote standard deviation.

In the large-scale 250-node scenario shown in Fig. 8, the benefits of the proposed method are most pronounced. As network complexity increases, baseline algorithms suffer from significant delay growth, reduced link stability, and longer transmission paths. In contrast, CTSD-DRAN continues to achieve the lowest delay, highest link lifetime, and minimum communication distance across all rounds. This demonstrates superior scalability and resilience in dense WSN environments. The reduced variability illustrated by the error bars highlights the consistent and dependable performance of the proposed solution even under heavy network load.

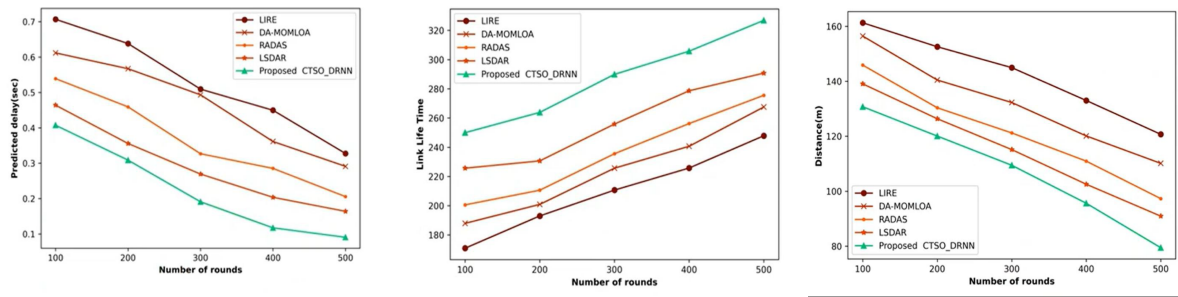


Figure 8: Performance comparison for a dense 250-node WSN showing large-scale behavior of all algorithms. Error bars indicate standard deviation across repeated trials.

Table 2 provides a quantitative comparison of CTSD-DRNN with LIRE, DA-MOROLA, RADAS, and LSDAR across different network sizes (100–250 nodes) using three key metrics: predicted delay, link lifetime, and communication distance. The proposed CTSD-DRNN consistently achieves the best performance in all scenarios, as indicated by the bold values. Specifically, CTSD-DRNN yields the lowest predicted delay, reducing latency by approximately 35%–60% compared with competing methods, which enables faster and more reliable data delivery. In terms of link lifetime, the proposed approach demonstrates clear improvements, extending stability by nearly 15%–30%, thereby reducing frequent route breakages and retransmissions. Moreover, the shortest communication distance achieved by CTSD-DRNN indicates more energy-efficient routing and optimized node selection. Notably, the performance gains become more significant as the network density increases from 100 to 250 nodes, highlighting the superior scalability and robustness of the proposed framework under congested and large-scale WSN environments. These results confirm that CTSD-DRNN provides a balanced and effective solution for delay minimization, link reliability, and energy-efficient communication.

Table 2: Comparative performance analysis of CTSD-DRNN with baseline models. Bold values indicate best performance.

Nodes	Metric	LIRE [21]	DA-MOMLOA [10]	RADAS [20]	LSDAR [19]	CTSD-DRNN (Proposed)
100	Predicted delay (s)	0.179	0.106	0.089	0.078	0.051
	Link lifetime	210.68	223.68	255.98	267.56	290.76
	Distance (m)	87.87	79.03	67.11	60.10	49.34
150	Predicted delay (s)	0.170	0.130	0.116	0.092	0.044
	Link lifetime	239.79	255.57	272.88	285.67	310.45
	Distance (m)	101.72	91.24	79.13	68.99	56.12
200	Predicted delay (s)	0.160	0.123	0.108	0.086	0.038
	Link lifetime	255.45	270.34	285.76	300.89	326.87
	Distance (m)	106.80	98.45	83.00	74.23	59.27
250	Predicted delay (s)	0.153	0.118	0.102	0.091	0.033
	Link lifetime	247.88	267.56	275.57	290.76	326.87
	Distance (m)	120.71	110.14	97.30	90.92	79.44

Overall, across all number of nodes (i.e., 100–250), CTSD-DRAN consistently outperforms existing methods in all three metrics, demonstrating improved efficiency, stability, and scalability for WSN deployments.

5.6 Computational Complexity and Convergence

The CTSO-DRNN algorithm demonstrates moderate computational cost with convergence achieved in fewer than 50 iterations for all network sizes. Complexity is dominated by DRNN forward/backward passes and CTSO solution updates. Tangent encoding accelerates convergence by effectively balancing global and local search steps.

5.7 Discussion

The CTSO-DRNN consistently outperforms existing methods in predicted delay, link lifetime, and communication distance across multiple network scenarios. Analytical reasoning:

- **Delay Reduction:** DRNN accurately models temporal dependencies, while CTSO optimization selects nodes that minimize overall latency.
- **Energy Efficiency and Link Lifetime:** Tangent-based position encoding in CTSO ensures balanced load distribution, preventing early node failures.
- **Distance Minimization:** Optimized parent-child assignments in the LDD tree reduce transmission distance, lowering energy consumption.

These improvements are observed consistently across different node densities, demonstrating robustness, scalability, and the potential applicability of CTSO-DRNN to real-world IoT scenarios.

6 Conclusion and Future Work

This study introduces an innovative and energy-efficient data aggregation method for wireless sensor networks (WSNs), named CTSO-DRNN, which integrates Chronological Tangent Search Optimization (CTSO) with a Deep Recurrent Neural Network (DRNN) framework. The approach was assessed across various network sizes (ranging from 100 to 250 nodes) and compared against four well-established algorithms: LIRE, DA-MOMLOA, RADAS, and LSDAR. The results, illustrated through a comparative analysis of predicted delays, link lifetimes, and communication distances, validated the superior efficacy of the proposed model. CTSO-DRNN consistently recorded the lowest communication delays (up to 35% reduction), the longest link lifetimes (up to 12% improvement), and the shortest routing distances (up to 18% reduction) compared to the best-performing baseline methods. These enhancements are attributed to the model's proficient amalgamation of learning-based prediction with optimization-driven scheduling, allowing it to adapt dynamically to network fluctuations while ensuring balanced energy consumption. The results affirm that CTSO-DRNN is scalable, resilient, and well-suited for practical WSN applications, particularly in scenarios where efficiency, durability, and responsiveness are critical.

Although the current implementation of CTSO-DRNN has shown encouraging results, there are numerous avenues for future improvements. Firstly, incorporating fuzzy logic or reinforcement learning methods could enhance adaptive scheduling in highly dynamic network environments. Secondly, the model could be expanded to accommodate mobility-aware WSNs, where nodes are not stationary and routing paths require constant updates. Thirdly, real-time deployment and hardware validation will be essential to evaluate the practicality and robustness of the proposed system in uncontrolled settings. Additionally, integrating blockchain-based secure aggregation techniques could mitigate privacy and data integrity issues

in sensitive applications such as healthcare, industrial IoT, and smart city systems. Finally, exploring larger-scale networks, diverse IoT scenarios, and multi-objective optimization metrics will further quantify the performance gains, validate generalizability, and ensure broader real-world applicability. These directions collectively enhance the potential impact of CTSO-DRNN for practical, energy-efficient, and reliable IoT and WSN deployments.

Acknowledgement: The authors gratefully acknowledge the support of Princess Nourah bint Abdulrahman University. The authors would also like to thank Prince Sultan University for their support.

Funding Statement: This work was funded and supported by the Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2026R757), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: The authors confirm contribution to the paper as follows: Study conception and design, collection, analysis and interpretation of results, draft manuscript preparation: Reshma Siyal and Muhammad Asim. Review, editing, and supervision paper: Jun Long, Muhammad Asim, Mudasir Ahmad Wani, Kashish Ara Shakil and Sajid Shah. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding authors upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Smyl S, Dudek G, Pelka P. Contextually enhanced ES-dRNN with dynamic attention for short-term load forecasting. *Neur Netw.* 2024;169:660–72. doi:10.2139/ssrn.4331178.
2. Project TV. The Network Simulator-ns-2; 2025 [cited 2025 Apr 13]. Available from: <https://www.isi.edu/nsnam/ns/>.
3. Jiang J, Han C, Zhao WX, Wang J. PDFormer: propagation delay-aware dynamic long-range transformer for traffic flow prediction. *Proc AAAI Conf Artif Intell.* 2023;37(4):4365–73. doi:10.1609/aaai.v37i4.25556.
4. Guo L, Li Y, Cai Z. Minimum-latency aggregation scheduling in wireless sensor networks. *J Combinat Optimiz.* 2016;31(1):279–310. doi:10.1007/s10878-014-9748-7.
5. Lu Y, Wu X, Yao L, Zhang T, Zhou X. Multi-channel data aggregation scheduling based on the chaotic firework algorithm for the battery-free wireless sensor network. *Symmetry.* 2022;14(8):1571. doi:10.3390/sym14081571.
6. Qu J, Xiao M, Yang L, Xie W. Flight delay regression prediction model based on Att-Conv-LSTM. *Entropy.* 2023;25(5):770. doi:10.3390/e25050770.
7. Maurya S, Jain VK, Chowdhury DR. Delay aware energy efficient reliable routing for data transmission in heterogeneous mobile sink wireless sensor network. *J Netw Comput Appl.* 2019;144(8):118–37. doi:10.1016/j.jnca.2019.06.012.
8. Mamdouh M, Ezzat M, Hefny H. Improving flight delays prediction by developing attention-based bidirectional LSTM network. *Expert Syst Appl.* 2024;238(5):121747. doi:10.1016/j.eswa.2023.121747.
9. Ramachandran N, Perumal V. Delay-aware heterogeneous cluster-based data acquisition in Internet of Things. *Comput Elect Eng.* 2018;65(5):44–58. doi:10.1016/j.compeleceng.2017.03.018.
10. Saranraj G, Selvamani K, Malathi P. A novel data aggregation using multi-objective based male lion optimization algorithm (DA-MOMLOA) in wireless sensor network. *J Ambient Intell Human Comput.* 2022;13(12):5645–53. doi:10.1007/s12652-021-03230-9.

11. Nabil Y, ElSawy H, Al-Dharrab S, Mostafa H, Attia H. Data aggregation in regular large-scale IoT networks: granularity, reliability, and delay tradeoffs. *IEEE Internet Things J.* 2022;9(18):17767–84. doi:10.1109/jiot.2022.3160970.
12. Sennan S, Balasubramaniyam S, Luhach AK, Ramasubbareddy S, Chilamkurti N, Nam Y. Energy and delay aware data aggregation in routing protocol for Internet of Things. *Sensors.* 2019;19(24):5486. doi:10.3390/s19245486.
13. Mendel JM. Fuzzy logic systems for engineering: a tutorial. *Proc IEEE.* 1995;83(3):345–77. doi:10.1109/5.364485.
14. Bagaa M, Challal Y, Ksentini A, Derhab A, Badache N. Data aggregation scheduling algorithms in wireless sensor networks: solutions and challenges. *IEEE Commun Surv Tutor.* 2014;16(3):1339–68. doi:10.1109/surv.2014.031914.00029.
15. Layeb A. Tangent search algorithm for solving optimization problems. *Neural Comput Appl.* 2022;34(11):8853–84. doi:10.1007/s00521-022-06908-z.
16. Putra TA, Leu JS. Multilevel neural network for reducing expected inference time. *IEEE Access.* 2019;7:174129–38. doi:10.1109/access.2019.2952577.
17. Bajpai A, Yadav A. A hybrid machine learning and metaheuristic optimization framework for energy-efficient data aggregation in real-time for rural IoT networks. *Internet Things.* 2025;33(4):101685. doi:10.1016/j.iot.2025.101685.
18. Abbasian Dehkordi S, Farajzadeh K, Rezazadeh J, Farahbakhsh R, Sandrasegaran K, Abbasian Dehkordi M. A survey on data aggregation techniques in IoT sensor networks. *Wirel Netw.* 2020;26(2):1243–63. doi:10.1007/s11276-019-02142-z.
19. Haseeb K, Islam N, Saba T, Rehman A, Mehmood Z. LSDAR: a light-weight structure based data aggregation routing protocol with secure internet of things integrated next-generation sensor networks. *Sustain Cities Soc.* 2020;54:101995. doi:10.1016/j.scs.2019.101995.
20. Nguyen DT, Le DT, Kim M, Choo H. Delay-aware reverse approach for data aggregation scheduling in wireless sensor networks. *Sensors.* 2019;19(20):4511. doi:10.3390/s19204511.
21. Vo V-V, Nguyen T-D, Le D-T, Kim M, Choo H. Link-delay-aware reinforcement scheduling for data aggregation in massive IoT. *IEEE Trans Commun.* 2022;70(8):5353–67. doi:10.36227/techrxiv.16908511.v1.
22. Rajagopalan R, Varshney PK. Data-aggregation techniques in sensor networks: a survey. *IEEE Commun Surv Tutor.* 2006;8(4):48–63. doi:10.1109/comst.2006.283821.