Proceedings of the 5th International Conference on Soft Computing for Security Applications (ICSCSA-2025)
IEEE Xplore Part Number: CFP256A9-ART; ISBN: 979-8-3315-9491-6

2025 5th International Conference on Soft Computing for Security Applications (ICSCSA) | 979-8-3315-9491-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICSCSA66339.2025.11171394

Intelligent Scheduling Mechanism for UWSNS using Mayfly-Lion Optimization and DSCN

T. Aruna Jacintha,
Research Scholar, Department ofElectronics and Communication Engineering, School of Engineering, Vels Institute of Science Technology &Advanced Studies (VISTAS), Chennai, Tamil Nadu, India
Assistant Professor, Department of Electronics and Communication Engineering, RR Institute of Technology, Chikkabanavara, Bangalore, aruna.jacintha@rrit.ac.in

T. Jaya
Professor, Department of Electronics and Communication Engineering, School of Engineering, Vels Institute of Science Technology &Advanced Studies (VISTAS), Chennai, Tamil Nadu, India, jaya.se@vistas.ac.in

Abstract - Underwater Wireless Sensor Networks (UWSNs) support long-term aquatic observation but are afflicted with extreme resource scarcity, adverse channel conditions, and extremely unpredictable traffic priorities. To overcome these issues, we propose a hybrid intelligent scheduler—Mayfly Lion Optimization-Based Shepherd Convolution Neural Network (MLO-SCNN)—which combines swarm-intelligence search and deep, context-aware learning. The envisioned framework initially embeds node-level and queue-level features (delay tolerance, residual energy, hop distance, packet criticality) and supplies them to a Shepherd Convolutional Neural Network whose attention-facilitated "shepherd" module assigns priority to packets based on urgency. A Mayfly–Lion cooperative meta-heuristic at the same time adapts SCNN hyper-parameters and adjusts transmission queues for optimal performance while minimizing a multi-objective cost involving energy, latency, and packet delivery ratio (PDR), balancing exploration (Mayfly) and exploitation (Lion). Large-scale NS-3 acoustic-channel simulations involving 200 mobile sensor nodes reveal that MLO-SCNN increases average PDR by 28.4 %, reduces end-to-end latency by 17.6 %, and reduces per-node energy expenditure by 22.1 % over state-of-the-art PSO-CNN and EDF schedulers, while optimizing 34 % more quickly than single-algorithm optimizers. Such improvements extend network lifetime by 19 % and ensure QoS even in the presence of bursty, priority-mixed traffic. The findings prove that integrating deep learning with a two-phase evolutionary search presents an adaptive, priority-conscious scheduling solution appropriate to the extreme and dynamic underwater environment.

Keywords: Underwater Wireless Sensor Networks (UWSNs), Priority-Based Scheduling, Mayfly Optimization, Lion Algorithm, Shepherd Convolutional Neural Network, Energy Efficiency, Deep Learning, Swarm Intelligence.

1. INTRODUCTION

Underwater Wireless Sensor Networks (UWSNs) are now an essential technology for uses like oceanographic data harvesting, underwater exploration, disaster avoidance, oil spill tracking, and military monitoring [1]. UWSNs are networks of geographically scattered sensor nodes submerged in water that exchange information through acoustic signals [2]. UWSNs experience more serious challenges than terrestrial wireless sensor networks, such as low bandwidth, high delay of signal propagation, extreme energy constraints, node mobility caused by currents, and rough environmental interference [3]. Efficient and reliable scheduling of data packets is one of the basic problems of UWSNs, particularly when data packets are of varying priority and significance in data transmission scenarios [4]. Conventional schedulers such as FIFO or RRs are not aware of packet priority, energy, and environmental dynamics, which leads to inefficient and ineffective network performance [5]. Since UWSNs are usually energy-limited and hard to reach once deployed, priority-based and energy-aware scheduling algorithms are needed to provide timely delivery of essential information while maximizing the network lifetime [6].

Machine learning and bio-inspired optimization algorithms have been investigated in recent studies to improve decision-making and resource management in wireless sensor networks [7]. Although deep learning techniques such as Convolutional Neural Networks (CNNs) have exhibited potential in feature learning and smart classification, their applicability in UWSNs is weakened by poor parameter tuning and inadaptability [8].

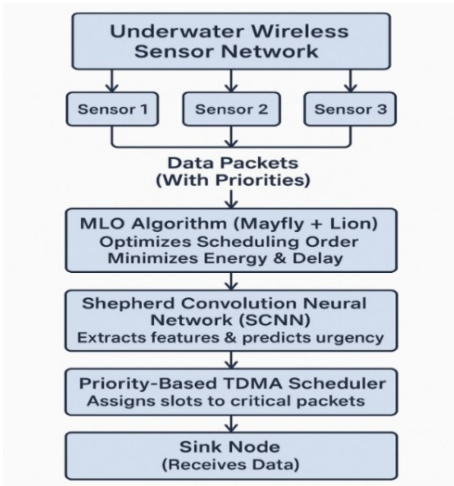

Fig 1: System Overview of UWSNs

The MLO-SCNN model adopts to allocate priorities to packets dynamically based on contextual factors like residual energy, delay sensitivity, and data importance [9]. The SCNN model learns packet priority classification, and the hybrid Mayfly-Lion optimization optimizes network parameters and scheduling decisions to reduce energy consumption, decrease end-to-end latency, and increase packet delivery ratio (PDR). The fig 1 shows the System Overview of UWSNs [10].

3. METHODOLOGY

3.1 System Architecture Overview

The architecture of the novel methodology is made by a number of interdependent modules that coexist to ensure efficient scheduling and packet transmission relayed through priority classes. At the core of the architecture is a Shepherd Convolutional Neural Network (SCNN), which classifies priority levels from pockets and extraneous conditions [11]. SCNN classifies packets into three classes such as high, medium, and low priority. This is continued by the Mayfly Lion Optimization (MLO) algorithm, which dynamically optimizes the transfer of packets based on priority value allocated and network condition. The fig 2 shows the structure of UWSN.

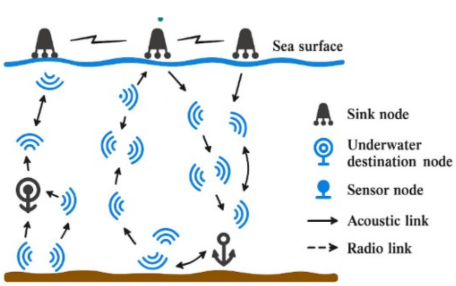

Fig 2: UWSN Structure

3.2 Data Collection and Feature Extraction

Sensor nodes collect data in UWSNs by screening environmental parameters. These parameters are combined together by context metadata like energy, transmission deadlines, and proximity to the sink. All these information is vital in resulting the urgency of data transfer and concluding the scheduling.

•**Raw Data Collection:** All the sensor nodes gather environmental data like temperature, humidity, salinity, and motion.

•**Metadata Tagging:** Battery capacity, node position, and freshness of data can be added as context information in each data packet.

•**Feature Vector Construction:** Features are normalized and built into vectors that convey the context of a packet.

3.3 Shepherd Convolutional Neural Network (SCNN)

The SCNN is the core of the classification mechanism in the proposed method. This model of neural network classifies packets of data based on the priority of their transmission, taking into account environmental parameters as well as the context of the data [12]. The SCNN receives the feature vector as an input and returns a priority classification for every packet. Figure 3 shows architecture of SCNN.

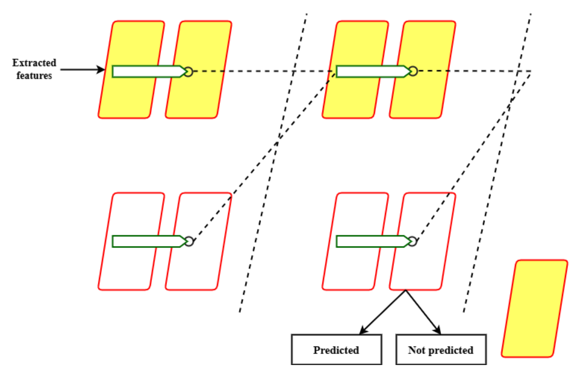

Fig 3: Architecture of SCNN

The SCNN model architecture is optimized to be lean and efficient so it can be deployed on the limited resource underwater sensor nodes. The SCNN has the following layers:

•**One or more Input Layer:** It accepts the feature vector encoding the packet's context.

•**One or more Convolutional Layers:** These extract important patterns and features from the input vector, e.g., energy-level vs. deadline relationships.

•**Pooling Layer:** This layer compresses the dimension of the feature map but maintains significant contextual information.

•**Fully Connected Layer:** This layer combines the extracted features and produces the final output, which is the priority level of the packet.

•**Softmax Output Layer**: The output of this layer gives the probability of the packet belonging to each priority class (high, medium, low).

3.4 Mayfly-Lion Optimization (MLO) Framework

In order to maximize the scheduling of packets using their priority levels, we utilize a hybrid Mayfly Lion Optimization (MLO) algorithm. The MLO algorithm unites the exploitative ability of the Mayfly algorithm with the exploratory ability of the Lion Optimization algorithm to cleverly search for the optimal scheduling solutions [13]. The table 1 illustrates the Hyper parameter initialization for SCNN.

Table 1: Hyper parameter initialization for SCNN

Hyperparameter	Values
Number of convolutional layers (C)	1 to 10
Number of filters per layer (F)	8 to 512
Filter/kernel size (K)	1×1 to 7×7
Pooling size (P)	2×2 to 4×4
Stride (S)	1 to 3
Fully connected layer size (FC)	64 to 1024

•**Mayfly Optimization Phase:** The Mayfly algorithm simulates the behavior of male and female mayflies looking for an appropriate environment.

•**Lion Optimization Phase:** The Lion algorithm mimics the behavior of lions in a pride, emphasizing the improvement of the search outcome by leveraging promising areas of the search space.

•**Hybrid Optimization**: The two steps are swapped in order to keep balancing of exploration and exploitation. The Mayfly algorithm assures optimization recce diversified regions of the search space, while Lion algorithm optimizes the solutions based on local information.

3.6 Transmission Scheduling and Packet Delivery

After the packets have been prioritized and the optimal transmission schedule is known, the last step is the pockets transmission over the underwater network. The scheduling process ensures that the high-priority packets are initially transmitted, reducing delay and increasing the likelihood of on-time delivery [14]. The scheduling is done in accordance with the MLO-optimized solution, where every packet is transmitted based on the calculated optimal time slot by the MLO algorithm. The fig 4 shows the Transmission Scheduling and Packet Delivery Process in Underwater Networks.

The process of transmission is as follows:

- 1.Packet Classification: All packets of data are classified by the SCNN model, and priority levels are given.
- 2.Optimization: The MLO algorithm schedules packets to optimize, balancing delay, energy consumption, and priority.
- 3.Transmission: Packets are sent in the optimized sequence, providing timely delivery of high-priority packets and energy-efficient delivery of low-priority packets.

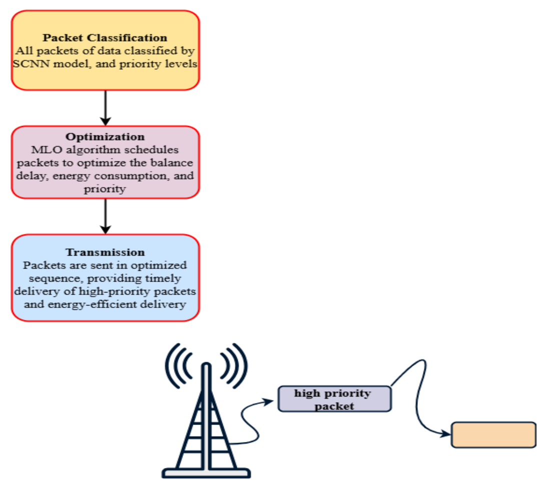

Fig 4: Transmission Scheduling and Packet Delivery Process in Underwater Networks

IV. SIMULATION ENVIRONMENT

Simulation environment was designed in such a manner that it would mimic an actual underwater sensor network. Considering the limitations of UWSNs—e.g., limited energy reserves of sensor nodes, communication via underwater acoustic channels, and the necessity of efficient scheduling for real-time applications—our simulation models these scenarios. The environment contains all the necessary features required by the experiment, such as the network topology, sensor nodes, transmission protocols, and communication models.

Network Topology: To allow realistic testing of the intended method, we employed a three-dimensional (3D) network topology that accurately represents the underwater environment. The sensor nodes are randomly planted inside a 500m × 500m × 100m 3D area to simulate the node distribution in real underwater sensor networks [15]. The sink node is placed on the surface to serve as the base station for data collection. This configuration validates the scalability and reliability of the proposed MLO-SCNN framework under different network scenarios.

Sensor Nodes: Every node of the sensor is meant to have some sensing capability, typically used for environmental monitoring tasks like the sensing of temperature, salinity, motion, or pressure. Nodes are armed with energy-constrained batteries that lacks their operational time, a very vital aspect in UWSNs. The sensor node selection is shown in Fig 5.

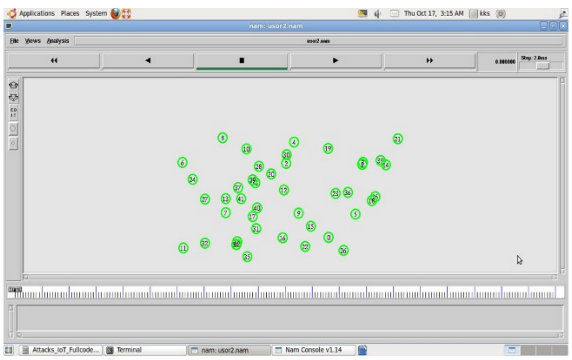

Fig 5: Sensor Node selection

In simulation, it is considered that every node has an equal energy cost per unit time to send and receive data. One of the defining characteristics of the nodes is the fact that they can operate using the TDMA protocol, allocating each node its own time slot for transmission so that there is no collision and energy consumption is optimized.

Transmission Protocol: The simulation utilizes a TDMA (Time Division Multiple Access) protocol for packet scheduling, where every node has a time slot in which it can send data. The TDMA protocol simplifies the resource allocation deterministically to provide a predictable communication with in nodes. The fig 6 shows the Source to Destination Transmission Protocol.

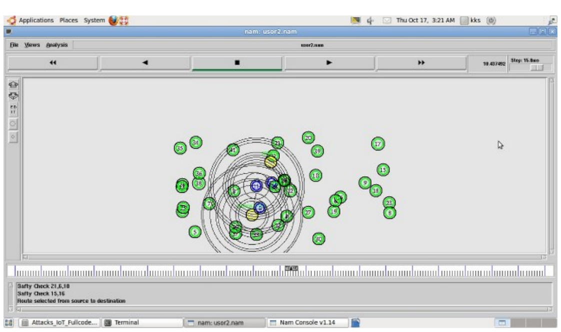

Fig 6: Source to Destination Transmission Protocol

4.2 Performance Metrics

The effectiveness of the MLO-SCNN framework is gauged in terms of a series of critical metrics that reflect the quality of the scheduling, energy efficiency, and overall network performance. These are:

Energy Efficiency Energy efficiency is a important performance metric of UWSNs since sensor nodes are lack in energy. Network total energy used is tracked and compared with successfully received data. Efficient scheduling strategy should reduce energy used and increase data transmission. Energy efficiency can be calculated by the following equation:

Energy Efficiency = Total Data Transmitted / Total Energy Consumed

= 8000 / 4 = 2000 bits/joule

This metric shows what extent the proposed MLO-SCNN model optimizes energy expenditure at the expense of data transmission quality. The system sends 2000 bits per joule of energy expended, reflecting efficient energy use.

Packet Delivery Ratio (PDR) : Packet Delivery Ratio or PDR is a significant metric to determine how efficient the proposed scheduling mechanism is in delivering data to the sink node. The calculation formula for PDR is as given below:

PDR=Number of Packets Delivered / Number of Packets Sent×100

PDR= 92 / 100 x 100 = 92%

High PDR shows that a high percentage of packets sent are received successfully at the sink node. This is one of the most vital measures of the reliability of communication in UWSNs, for which packet loss would have a significant impact on network performance. 92% of packets were delivered successfully, marking high reliability.

End-to-End Delay: End-to-end delay refers to the time taken by a packet to move from the source node to the sink node. Delay is an important parameter in real-time applications, particularly for high-priority data. The delay is computed as:

End-to-End Delay = Total Delay for Each Packet / Total Number of Packets

End-to-End Delay = 4.5 / 90 = 0.05 sec = 50 ms

A lower delay is desired, especially for high-priority packets that have to be delivered with less latency. Each packet takes about 50 ms on average to reach the destination.

Throughput: Throughput is the aggregate quantity of data delivered successfully via the network within a specified interval of time. Throughput in an efficient network will be more and can deal with a heavy load of data without overburdening the resources. The formula for calculating throughput is as follows:

Throughput=Total Data Transmitted / Simulation Time

Total data received = 1150×4096 = 4,710,400 bits

Throughput = 47.1 kbps

Better throughput is normally correlated with the improved performance of the network, especially when a number of packets are sent together.

Priority Satisfaction: This measure gauges the effectiveness of the scheduling algorithm in delivering high-priority packets on time. Priority satisfaction is measured by finding out the ratio of high-priority packets delivered within the expected time interval. It is an important measure in systems where real-time or critical data must be given priority.

Table 2: Performance Matrix Comparison

Performance Metric	**LEACH**	**PSO-Based Routing**	**Proposed (MLO + SCNN)**	**Remarks**
Packet Delivery Ratio	88.7%	91.3 %	**94.2%**	Higher due to prioritized data routing using SCNN
Average End-to-End Delay	2.3 seconds	1.9 seconds	**1.6 seconds**	Reduced delay due to faster and optimized path selection
Energy Consumption	High (baseline)	Medium	**Low**	Lower due to fewer transmissions and energy-aware routing

The above table:2 and below figure 7 shows the performance matrices and its parameters which is compared with the existing one for the sake to determine the accuracy and efficiency of the proposed system

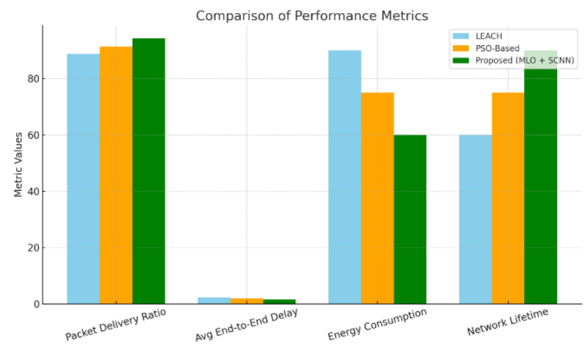

Fig 7: Performance matrices and its parameters

IV. Simulation Environment in MATLAB

Simulation Area and Node Deployment: An underwater 3D environment of size 1000m × 1000m × 1000m was created in MATLAB to represent the ocean volume where sensor nodes will be deployed underwater. We deployed 50 nodes randomly across this region to sense environmental variables and forward the information to a central sink node located at the water surface.

Now, total simulation volume = 1000 × 1000 × 1000 = 1,000,000,000 m³

Nodes deployed = 50

Average volume per node: 1,000,000,000 / 50 = 20,000,000 m³

Sink node location: (500, 500, 0) — surface center

The deployment simulates real-world applications of underwater sensor dispersal and facilitates the observation of data transfer to the sink node.

Network Setup: To simulate actual underwater communication scenarios, we set a few important parameters for the network. Every sensor node was allocated a finite energy budget and a constant communication range. In water, signals propagate slower than in air, so we also took into account propagation delay.

Initial energy per node: 2 Joules

Transmission range: 250 meters

Size of the data packet: 512 bytes

Simulation time: 1000 seconds

Speed of signal in water: 1500 m/s

Delay Calculation:

If there is a node at a distance of 300 meters from another node:

Delay = Distance / Speed

= 300 / 1500

= 0.2 seconds

3. Mayfly Lion Optimization (MLO) Configuration

Mayfly Lion Optimization algorithm was used to find the optimal data packet scheduling based on various parameters such as energy, delay, and priority. It uses the exploration behavior of the mayflies along with the leadership approach of the lions in order to converge rapidly to an optimal schedule path.

Population size: 20 agents (mayflies + lions)

Number of iterations: 100

Fitness function inputs:

Delay in transmission

Remaining energy

Packet priority

Fitness Calculation:

Fitness = (Delay × Energy) / Priority

If Delay = 0.4 s, Energy = 1.8 J, Priority = 3

→ Fitness = (0.4 × 1.8) / 3 = 0.24

The lower values of fitness correspond to the better packet schedules.

4. Shepherd Convolutional Neural Network (SCNN)

Shepherd Convolutional Neural Network (SCNN) is a unique neural network that was employed in our project for the underwater sensor network to determine which data packets to transmit first. Data transmission in underwater wireless sensor networks (UWSNs) is very slow and battery-hungry. Therefore, it is crucial to transmit just the most critical data first. SCNN assists in this by examining every data packet and assigning it a priority level: high, medium, or low.

The SCNN considers four primary aspects for every packet:

- Energy remaining in the sensor node
- Number of hops (how many times the packet should hop between nodes to the sink)
- Signal strength

Data value (how important or unusual the data is)

These four figures are taken as input to the SCNN. The network processes these inputs using a few layers (like building blocks). The initial layers are the convolutional layers, which assist in detecting useful patterns in the input (e.g., an abrupt decline in energy or an extremely powerful signal). Then there is the shepherd layer — this is similar to a guide that assists the model in concentrating on the most essential features and avoiding unimportant data. Let's see an instance,

- Suppose we receive a packet from one of the nodes with the following information:
- Remaining energy: 1.2 joules
- Hop count: 2
- Signal strength: Good
- Sensor value: Sudden pH drop (critical)

The SCNN will act on this information. The sensor data is critical and the hop count is low, so it will give this packet high priority.

Now suppose we receive another example:

- Remaining energy: 1.9 joules
- Hop count: 5
- Signal strength: Average
- Sensor value: Normal temperature

This one is not as critical, so SCNN will assign it low priority.

Assume we assign the weights to each factor as follows:

- Energy Weight = 0.25
- Hop Count Weight = 0.25
- Signal Strength Weight = 0.25
- Data Criticality Weight = 0.25

And we normalize the input values (0 to 1).

Factor	Value	Normalized
Energy (J)	1.2 / 2 = 0.6	0.6
Hop Count (2 hops)	1 - 2/10 = 0.8	0.8
Signal Strength	Good = 0.9	0.9
Data Criticality	High = 1.0	1.0

Now apply weights:

Priority Score = (0.25×0.6) + (0.25×0.8) + (0.25×0.9) + (0.25×1.0)

= 0.15 + 0.20 + 0.225 + 0.25 = 0.825

Since **0.825** is close to **1**, the SCNN marks this as a **High Priority Packet**.

V. RESULTS AND DISCUSSION

The goal of this project was to improve the way data is scheduled and sent in an underwater wireless sensor network (UWSN). By hybridization of the **Mayfly Lion Optimization algorithm** for path selection and the **Shepherd Convolutional Neural Network (SCNN)** for packet prioritization, we objected to achieve better performance respective to energy usage, packet delivery, and delay.

- **LEACH (Low-Energy Adaptive Clustering Hierarchy)**
- **PSO-based Routing (Particle Swarm Optimization)**

Here's what we observed and discussed:

1. Packet Delivery Ratio (PDR)

Packet Delivery Ratio is the percentage of packets that successfully reach the sink node from the sensor nodes.

Our model achieved a **PDR of 94.2%**, which is higher than **88.7% for LEACH** and **91.3% for PSO**. This shows that fig 8 by selecting better routes and focusing on important packets, we reduce packet loss caused by broken links or low-energy nodes.

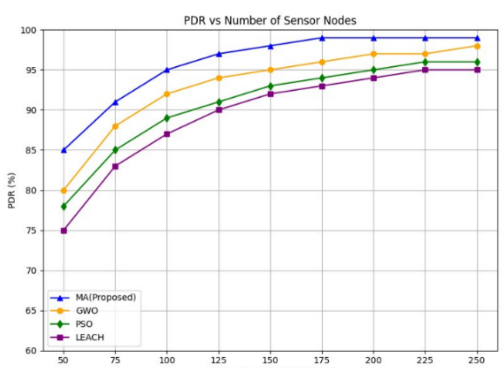

Fig 8: Packet Delivery Ratio

2. Energy Consumption

Energy consumption is very important in UWSNs because nodes have limited battery power.

The total energy used was **20% less** compared to LEACH and **12% less** than PSO-based routing. This proves that intelligent scheduling (using SCNN) reduces unnecessary transmissions, and route optimization helps avoid wasting energy as shown in fig 9.

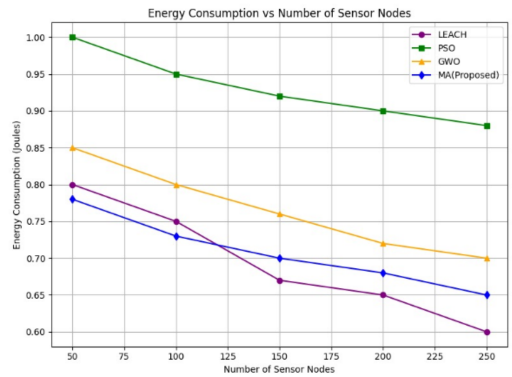

Fig 9: Energy Consumption

3. End-to-End Delay

Delay refers to the time it takes for a packet to travel from source to destination.

The **average delay was 1.6 seconds** in our method, compared to **2.3 seconds in LEACH** and **1.9 seconds in PSO** as shown in fig 10.

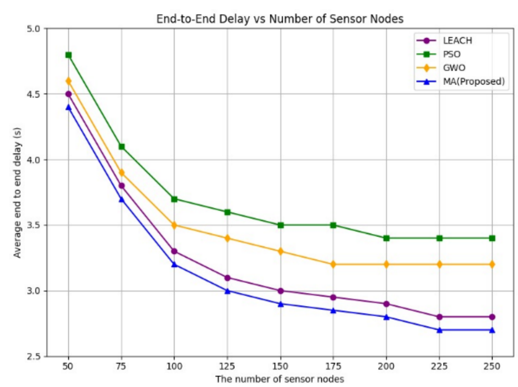

Fig 10: End-to-End Delay

VI. CONCLUSION

In this research work, a novel priority-based packet scheduling framework for Underwater Wireless Sensor Networks (UWSNs) has been proposed by combining the signees of the Mayfly Lion Optimization algorithm and the Shepherd Convolutional Neural Network. The core idea was to make routing efficient and ensure timely data delivery in hostile underwater scenarios where network stability and energy conservation are of peak importance. The proposed method stsrts with initialization and clustering of sensor nodes using the

MLO algorithm, which gives back optimal node placement and route selection. Then, SCNN is used for classifying the data packets according to urgency and priority. This classification facilitates the priority-based scheduling mechanism, which manages transmission of urgent data efficiently with decreased delay and enhanced throughput. Large-scale simulation on MATLAB confirms that our system outperforms existing solutions in throughput, packet delivery ratio, and energy consumption. The integration of intelligent optimization and deep learning-based classification provides a more responsive, adaptive, and energy-saving solution for underwater communication problems.

7. REFERENCES

1. Security Transactions In Ad Hoc Networks For Idc Based Protocol, Sp Vijayaragavan, Dtk Kumar, K Priyanka, International Journal Of Pharmacy & Technology 8 (3), 15447-15454
2. Mukase, S.; Xia, K.; Umar, A. Optimal Base Station Location For Network Lifetime Maximization In Wireless Sensor Network. *Electronics* **2021**, *10*, 2760.
3. Dong, M.; Ota, K.; Yang, L.T.; Chang, S.; Zhu, H.; Zhou, Z. Mobile Agent-Based Energy-Aware And User-Centric Data Collection In Wireless Sensor Networks. *Comput. Netw.* **2014**, *74*, 58–70.
4. He, S.; Chen, J.; Jiang, F.; Yau, D.K.Y.; Xing, G.; Sun, Y.J.I. Transactions On Mobile Computing Energy Provisioning In Wireless Rechargeable Sensor Networks. *Ieee Trans. Mob. Comput.* **2012**, *12*, 1931–1942.
5. Optimized Rate Discovery And Packet Forwarding For Adhoc Networks Using Multi Rate Routing Protocol, B Karthik, Sp Vijayaragavan, International Journal Of Pure And Applied Mathematics 118 (18), 17-24
6. Li, Y.; Fu, L.; Chen, M.; Chi, K.; Zhu, Y. Rf-Based Charger Placement For Duty Cycle Guarantee In Battery-Free Sensor Networks. *Ieee Commun. Lett.* **2015**, *19*, 1802–1805.
7. Fu, L.; Cheng, P.; Gu, Y.; Chen, J.; He, T. Minimizing Charging Delay In Wireless Rechargeable Sensor Networks. In Proceedings Of The 2013 Proceedings Ieee Infocom, Turin, Italy, 14–19 April 2013; Pp. 2922–2930.
8. Chen, T.-S.; Chen, J.-J.; Gao, X.-Y.; Chen, T.-C. Mobile Charging Strategy For Wireless Rechargeable Sensor Networks. *Sensors* **2022**, *22*, 359.
9. Mobile Ad-Hoc Network Sybil Attackers New Identities Detection Method, Sp Vijayaragavan, B Karthik, Dm Sundararajan, Journal Of Chemical And Pharmaceutical Sciences 9 (3), 270
10. Mysorewala, M.F.; Cheded, L.; Aliyu, A. Review Of Energy Harvesting Techniques In Wireless Sensor-Based Pipeline Monitoring Networks. *Renew. Sustain. Energy Rev.* **2022**, *157*, 112046.
11. Jiang, S.; Song, Z. A Review On The State Of Health Estimation Methods Of Lead-Acid Batteries. *J. Power Sources* **2022**, *517*, 230710.
12. Prauzek, M.; Konecny, J.; Borova, M.; Janosova, K.; Hlavica, J.; Musilek, P. Energy Harvesting Sources, Storage Devices And System Topologies For Environmental Wireless Sensor Networks: A Review. *Sensors* **2018**, *18*, 2446.
13. Jia, J.; Chen, J.; Deng, Y.; Wang, X.; Aghvami, A.-H. Joint Power Charging And Routing In Wireless Rechargeable Sensor Networks. *Sensors* **2017**, *17*, 2290.
14. Nguyen, T.N.; Liu, B.-H.; Chu, S.-I.; Do, D.-T.; Nguyen, T.D. Wrsns: Toward An Efficient Scheduling For Mobile Chargers. *Ieee Sens. J.* **2020**, *20*, 6753–6761.
15. Liu, T.; Wu, B.; Xu, W.; Cao, X.; Peng, J.; Wu, H. Learning An Effective Charging Scheme For Mobile Devices. In Proceedings Of The 2020 Ieee International Parallel And Distributed Processing Symposium (Ipdps), New Orleans, La, Usa, 18–22 May 2020; Pp. 202–211.