Clustering Based on Gray Wolf Optimization Algorithm for Internet of Things over Wireless Nodes

Chunfen HU[1], Haifei ZHOU[2*], Shiyun LV[3]
Changzhou College of Information Technology, School of Cyberspace Security, Changzhou 213000, China

***Abstract*—The Internet of Things (IoT) creates an environment where things are permitted to act, hear, listen, and talk. IoT devices encompass a wide range of objects, from basic sensors to intelligent devices, capable of exchanging information with or without human intervention. However, the integration of wireless nodes in IoT systems brings about both advantages and challenges. While wireless connectivity enhances system functionality, it also introduces constraints on resources, including power consumption, memory, and CPU processing capacity. Among these limitations, energy consumption emerges as a critical challenge. To address these challenges, metaheuristic algorithms have been widely employed to optimize routing patterns in IoT networks. This paper proposes a novel clustering strategy based on the Gray Wolf Optimization (GWO) algorithm. The GWO-based clustering approach aims to achieve energy efficiency and improve overall network performance. Experimental results demonstrate significant improvements in key performance metrics. Specifically, the proposed strategy achieves up to a 14% reduction in energy consumption, a 34% decrease in end-to-end delay, and a 10% increase in packet delivery rate compared to existing approaches. The findings of this research contribute to the advancement of energy-efficient and high-performance IoT networks. The utilization of the GWO algorithm for clustering enhances the network's ability to conserve energy, reduce latency, and improve the delivery of data packets. These outcomes highlight the effectiveness and potential of the proposed approach in addressing resource limitations and optimizing performance in IoT environments.**

***Keywords*—Internet of things; energy consumption; clustering; optimization; gray wolf optimization**

I. Introduction

As technological advances advance, instruments and objects in our environment can exchange data through technologies such as Radio-Frequency Identification (RFID) and Wireless Sensor Networks (WSNs) [1, 2]. The emergence of wireless communication and seamless integration of different technologies between devices has resulted in the concept of the Internet of Things (IoT) that facilitates data exchange among a variety of items and their associated things over a network protocol or standard at any time [3-5]. All IoT devices and things are assigned unique IP addresses. The devices can be configured to sense and collect raw data from the physical environment to process it and make decisions [6]. The integration of Blockchain [7], humanitarian logistics [8], cloud computing [9], machine learning [10-14], and artificial intelligence [15, 16] within the IoT ecosystem plays a crucial role in enabling secure and efficient data exchange, optimizing resource allocation, improving decision-making processes, and enhancing overall system resilience, making it a transformative force in various domains such as healthcare, transportation, energy management, and disaster response.

In such energy-constrained networks, clustering has proven to be an effective method of designing energy-efficient routing algorithms [17, 18]. This method groups the nodes together in clusters. Each cluster is headed by a Cluster Head (CH) whose responsibility is to gather the data of its members. Clustering can provide scalability, conserve bandwidth, and reduce the routing problem among all sensors [19]. The CHs are responsible for relaying the data to the sink node, thus reducing the total number of hops needed [20]. This way, the energy consumed by relaying data is reduced since the nodes only need to relay data over short distances. Furthermore, clustering helps in balancing the load on the network, which in turn improves the network performance. Moreover, clustering ensures efficient data aggregation, which further minimizes the amount of data that needs to be transmitted to the sink node. The result is an efficient use of the available resources and a better overall experience for all participants [21].

Clustering approaches currently available are primarily time-based. A clustering approach can be static, dynamic, or hybrid. Static clustering is used when the data points and clusters can be defined ahead of time and do not change over time. Dynamic clustering automatically adjusts the clusters as the data points change [22]. Hybrid clustering combines the two approaches, using static clustering to define the initial clusters and then dynamic clustering to adjust them over time. There is minimal overhead associated with a static performance network, and it is stable for a short period of time. Although dynamic performance increases the lifetime of a network, it has a high overhead cost. Hybrid clustering allows for a more flexible approach to clustering, as the clusters can be adjusted over time without having to start from scratch. This helps to reduce the computational overhead associated with clustering, as well as the time it takes to create an optimized clustering solution [17].

II. Related Works

A mechanism is proposed by Said [23] for dividing the IoT environment into various zones based on the characteristics of the network. Afterward, the ACO algorithm is applied to the areas in order to resolve the routing problem. It is evident from the results of NS2 that the proposed routing algorithm meets the target energy consumption, packet loss rate, latency, bandwidth consumption, and overhead criteria. By using the genetic algorithm, Fouladlou and Khademzadeh [20] developed an effective routing approach and extended the lifetime of a network by clustering IoT objects. Several experiments have shown that the proposed scheme performs better than IEEE 802.15.4 in terms of transmission rate, energy consumption, delay, and bit error rate.

Mohseni, et al. [24] proposed a cluster-based routing strategy in the IoT by combining the fuzzy logic system and the Capuchin search algorithm, called CEDAR. It involves two stages, namely the clustering process and intra- and extra-cluster routing. This strategy significantly cuts energy consumption by IoT devices through clustering the nodes in the network, and each cluster is responsible for routing the packets of the nodes in its own cluster. Additionally, the fuzzy logic system allows the nodes to adapt to the changing network conditions, and the Capuchin search algorithm ensures that the packets are routed in the most efficient way. Simulation results reveal that CEDAR is superior to comparative approaches regarding energy consumption, delay, and network lifetime. An optimized routing strategy based on neuro-fuzzy rules has been proposed by Thangaramya, et al. [25]. The results of the experiments conducted in this study demonstrate that the modeled routing protocol performs well in terms of network lifespan, latency, delivery rate, and energy consumption.

Geetha, et al. [26] propose a new energy-aware future load prediction and cluster communication strategy for IoT networks. It determines an optimal number of CHs and forecasts the incoming load on the network. It comprises two main phases: clustering with the satin bowerbird algorithm and load estimation using deep random vector functional link networks. A comprehensive analysis of the results and discussion indicates that the proposed method of regulating renewable energy usage in IoT networks is extremely effective.

Lakshmanna, et al. [27] introduced a novel cluster-based IoT routing protocol. The objective of this design is to ensure optimal energy utilization and network lifetime. This is achieved by developing an enhanced Archimedes optimization algorithm-driven clustering approach to facilitate the selection of CHs and establishing cluster structures. The suitability function takes into account the number of hops that the data must take to reach its destination, how far apart the nodes are from each other, and the amount of energy consumed. The teaching-learning-based optimization algorithm then uses this information to determine the best route for the data to take. As a result, the network is more efficient and reliable, leading to improved performance.

III. Proposed Method

The proposed method divides a network's lifespan into multiple cycles. It operates under two distinct stages, namely, initialization and stabilization. During the initialization stage, the base station collects location and energy information about nodes and determines CHs based on this information and the Gray Wolf Optimization (GWO) algorithm. Data collected by the cluster heads are sent to the base station during the steady state phase. In the proposed method, to conserve energy, the initialization stage is performed when the current cluster heads are close to death. This process eliminates the need to send and receive control packets during the setup phase, reducing energy consumption. The proposed method is illustrated in Fig. 1.

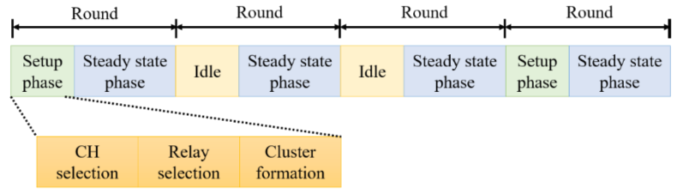

Fig. 1. The process of the proposed method.

A. Cluster Head Selection

In this subsection, the clustering problem is modeled as an optimization problem, and an optimization algorithm is employed to select cluster heads. A parameter called dead time (T_D) is defined for each node. The number represents the maximum number of iterations a node may survive, given its role in the network and its remaining energy. This definition can be formulated as Eq. (1), in which $E_r(i)$ stands for energy remaining on node i, and $E_c(i)$ denotes the amount of energy consumed by a given node per iteration.

$$T_D(i) = \frac{E_r(i)}{E_c(i)} \tag{1}$$

T_D values differ between nodes based on the solution. Maximizing the average T_D between all nodes is the most effective solution. As an optimization problem, this definition can be expressed as follows:

$$maximize\ f = avg(T_D) = \frac{1}{|nodes|} \sum_{i \in nodes} T_D(i) \tag{2}$$

which presupposes the following assumptions:

$$Er(CH_j) > \frac{1}{|nodes|} \sum_{i \in nodes} E(i), 0 < j \leq m \tag{3}$$

The above condition states that the residual energy of all cluster heads should surpass the average energy of all nodes. This is necessary to ensure that the cluster heads have enough energy to effectively manage the clusters and maintain effective communication between the cluster heads and the other nodes in the network. The death time for each node in the network is determined according to the role of that node in the network. This calculation excludes the energy spent on sensing and data processing since these activities are negligible compared to communication. The energy consumption of each normal node is calculated by Eq. (4). This calculation does not include the energy associated with the exchange of control packets since our goal is to determine the maximum number of cycles a node may survive before re-clustering.

$$E_c^{member}(i) = E_{TX}(L, dis(i, CH_i)) \tag{4}$$

In Eq. (4), ch_i is the cluster head of node i, $dis(i,j)$ indicates the distance between two nodes, L specifies the size of the data packet in bits, and E_{Tx} represents the transmission energy. The amount of energy consumed by a cluster head is calculated by Eq. (5).

$$E_c^{CH}(j) = E_{RX}(L \times CM_j) + E_{DA} \times L \times (CM_j + 1) + E_{TX}(L, dis(j, next(j))) \quad (5)$$

In Eq. (5), E_{RX} refers to the energy consumed for receiving a packet, CM_j is the number of cluster nodes, E_{DA} is the required energy for data aggregation per bit, and next represents the next hop, which can be another node or the base station. Some cluster heads may act as a relay for another cluster head. The energy consumption for relaying data by relay node is given by Eq. (6).

$$E_c^{relay}(r) = E_{RX}(L) + E_{TX}(L, dis(r, BS)) \quad (6)$$

The proposed method finds an optimal solution to this problem using the gray wolf optimization algorithm. This algorithm is described in detail in the following section.

B. Selection of Relay Nodes

To avoid the rapid exhaustion of the energy source of cluster heads far from the base station, each cluster head is assigned a relay node, which is used by only one cluster head at a time. Therefore, several cluster heads lack relays and transmit data to the base station in a direct manner. To assign relays to the cluster heads, we choose a suitable relay for each cluster head, from the farthest cluster head to the central station to the closest cluster head to the central station. The desired goals in choosing the cluster head are to minimize the total energy consumption and create the greatest balance between the energy consumption of the cluster head and the relay. Fig. 2 shows the central station, a cluster head, and a hypothetical relay.

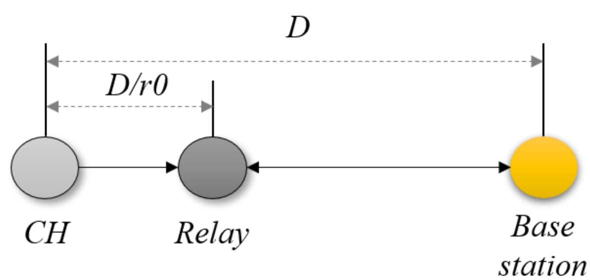

Fig. 2. Process of relay nodes selection.

It can be proved that there is a specific and fixed value for r_0 to guarantee the lowest energy consumption and the greatest balance between the energy consumption of the cluster head and the relay. Furthermore, a private point exists on the line segment linking the cluster head and the base station that serves as the optimal relay point. The calculated value for r_0 is 1.8. To select a relay for each cluster head, first, according to the value of r_0, the best point for the relay, located on the segment of the line between the cluster head and the central station, is calculated. The nearest cluster head, not previously selected as a relay, is calculated as the desired cluster head relay is chosen. Also, when no relays are located within a threshold of the desired point, the cluster head sends messages immediately to the base station. This process is advantageous in several ways:

- It significantly reduces the energy used to transmit packages to the base station.
- It minimizes the problem of being spot-hot. This is because the balance of energy consumption between the cluster head and the relay is guaranteed, and different relays are selected periodically.
- The relay is selected for the maximum possible number of cluster heads.

An example of the result of this process to select relays is shown in Fig. 3. There are 100 cluster heads in this network, and the base station is in the middle. Notably, some cluster heads do not have relays; these cluster heads are displayed as crosses without lines.

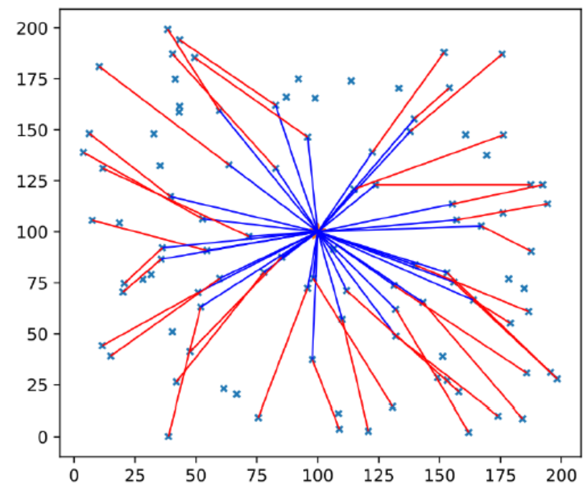

Fig. 3. An example of assigning relays to CHs.

C. Formation of Clusters

During the initialization stage, the nodes send a node-MSG message to the central station. This message contains the remaining energy and the location of the node. This information is needed for clustering by the base station. In the next step, the base station selects the cluster heads using the presented method based on the gray wolf optimizer that leads to the maximization of the fitted function given in Eq. (2). Then the base station sends a broadcast message that contains the ID of the selected cluster heads and the corresponding relays. After the cluster heads receive this message and realize their selection as the cluster head, each cluster head broadcasts a CH-ADV message to introduce itself to the network. The remaining nodes choose a nearby cluster head based on the strength of the received CH-ADV signals and transmit a Join-MSG message. Relays also broadcast the Relay-ADV message to the network. At this stage, since each cluster head already has its relay ID, it waits for the Relay-ADV sent by its relay and sends an RJoin-MSG message in response. After completing these steps, all nodes will be aware of their role in the network, and the network will enter the stabilization stage. The initialization stage will not be performed unless one of the nodes has consumed 50% of its energy since the last initialization stage.

D. Clustering with GWO Algorithm

In the proposed method, the gray wolf optimizer is used to maximize the fitting function shown in Eq. (2). For this purpose. Each solution should be displayed as a multidimensional vector. In other words, because the wolves represent the solutions in the gray wolf optimizer and have a multi-dimensional position vector, then the clustering solutions should be displayed as multi-dimensional vectors. To perform this mapping, we consider a vector with the number of dimensions expressed as the number of network nodes. Each dimension of this vector indicates the chance of a node becoming the cluster head. To select the heads of the clusters, a predetermined number of nodes with the highest chance value in the alpha wolf position vector are selected as the heads of the cluster. Then every ninety members of the nearest cluster head are considered, and the relays of the cluster heads are also selected according to the presented method.

Assumption 1: A suitable value of r_0 to minimize energy consumption is the value of $\frac{1}{\sqrt[3]{2}} + 1 \approx 1.793$.

Proof: According to the scenario depicted in Fig. 2, the amount of energy consumed by the cluster head is obtained from the following equation:

$$E_{dual-hop}^{CH} = E_{TX}(l, \frac{D}{r_0}) \tag{7}$$

which corresponds to the energy required to send a packet of length L bits to the distance r_0/D. The amount of energy consumed by the relay is also obtained from Eq. (8), which represents the transmission of two packets, each with a length of L, from the relay to the base station. As a result, the total energy consumption is calculated by Eq. (9).

$$E_{dual-hop}^{relay} = 2 \times E_{TX}(L, \left(D - \frac{D}{r_0}\right)) \tag{8}$$

$$E_{dual-hop}^{total} = E_{TX}\left(L, \frac{D}{r_0}\right) + 2 \times E_{TX}(L, \left(D - \frac{D}{r_0}\right)) \tag{9}$$

In order to achieve the lowest amount of energy consumption, Eq. (9) should be minimized. Assuming that the extra-cluster connections follow the fading multipath model, by expanding the above relation using Eq. (1), we reach Eq. (10), which can be written as Eq. (11). Here because L, E_{mp} and E_{elec} are constant values, it can be said that to minimize the above expression, it is enough to minimize the expression 12. In addition, since D is also a constant and non-zero value, the function *h1* is minimized when the function *h2* is minimized.

$$E_{dual-hop}^{total} = \text{L} \times \left(E_{mp} \times \left(\left(\frac{D}{r_0}\right)^4 + 2 \times \left(D - \frac{D}{r_0}\right)^4\right) + E_{elec}\right. \tag{10}$$

$$E_{dual-hop}^{total} = \text{L} \times \left(E_{mp} \times \left(\left(\frac{D}{r_0^4}\right)^4 + 2 \times \left(\frac{D^4(r_0-1)^4}{r_0^4}\right)\right) + E_{elec}\right. \tag{11}$$

$$h_1 = \frac{D^4}{r_0^4} + 2 \times \left(\frac{D^4(r_0-1)^4}{r_0^4}\right) \tag{12}$$

Assumption 2: The best value for r_0 to create a balance between the energy consumption of the cluster head and the corresponding relay is approximately equal to 1.84.

Proof: In accordance with the preceding proof, to create a balance between the energy consumption of the cluster head and the relay, their absolute magnitude difference should be minimized according to Eq. (13).

$$minimize\ g = \left|E_{dual-hop}^{CH} - E_{dual-hop}^{relay}\right| \tag{13}$$

By expanding the above relation using Eq. (1), we reach Eq. (14). Because D, L, E_{mp} and E_{elec} are constant values, the g function is minimized at a point where the g_0 function given in Eq. (14) becomes zero.

$$g_0(r_0) = \frac{1 - 2 \times (r_0 - 1)^4}{r_0^4} \tag{14}$$

The roots of the above function are the best values for r_0 to balance the energy consumption of the cluster head and relay. A suitable root for this function is 1.8. As a result, choosing r_0=1.8 will lead to the equal energy consumption of cluster head and relay. As a result, to simultaneously achieve both goals of optimality and balance, the value selected for r_0 is equal to the average of these two values, i.e., 1.8.

Assumption 3: The complexity of the control packets of the presented algorithm equals O(N), where N is the number of nodes within the network.

Proof: During each cycle, N Node-MSG packets are transmitted to the base station. In addition, every node issues a Join-MSG message to its CH. Each CH also sends one CH-ADV message, one Relay-ADV or Rejoin-MSG message, and two packets. If we assume that the number of CHs is 5% of the total number of nodes, the total number of control packets is equal to $2N + 2 \times \left(\frac{N}{20}\right) = \frac{21}{10}N$, which is related to *O(N)*.

IV. Simulation Results

The proposed method is simulated and implemented using CPU core i5 and 4GByte RAM. A Matlab simulator has been used to obtain the results. The method was tested under various conditions, such as varying the number of nodes, to ensure accurate results. The results were then compared with those obtained from other methods to prove the proposed method's performance. Table I summarizes the key parameters and variables used in the proposed method's simulation.

The energy expended in delivering the sensed data to the base station is one of the most important parameters of analyzing routing methods in the IoT environment. By properly assessing the energy expenditure, it is possible to optimize the routing methods and improve the overall performance of the IoT system. This measurement can compare different routing methods and select the most energy efficient one. Moreover, it can be used to identify areas of high energy consumption, which can be addressed to further optimize the IoT system. According to Fig. 4 to 6, our method is more energy-efficient than previous methods. Fig. 4 compares our method's average residual energy with R-LEACH when the number of rounds is increased. According to this figure, our algorithm significantly increases the number of alive nodes compared to the comparative algorithm. Fig. 5 and 6 illustrate the comparison between the energy consumption of our method and RDDI. The results show that our algorithm can reduce energy consumption while ensuring that more nodes remain alive. This is because it can identify clusters that consume less energy, thus reducing the entire network's energy consumption. Additionally, by optimizing the selection of cluster heads, our algorithm can reduce the amount of energy wasted due to redundant communications. The packet delivery rate can be described as the ratio of traffic correctly delivered to the base station as a percentage of all traffic carried within the network. As shown in Fig. 7, our algorithm achieves a higher percentage of packets delivered than the comparative algorithm. The packet delivery ratio decreases as the number of nodes increases and the density increases. Data collisions will result as network density increases, leading to a higher rate of data transmission failures and packet loss.

TABLE I. SIMULATION VARIABLES

Variable	Value
Network dimensions	(100 × 100)
Number of nodes	50-300
Packet size	800 bits
Node distribution	Random
Initial node energy	Different based on the scenario
Iterations	100-500
Efs	10 pj/bit/m2
Eelec	50 nj/bit
Eamp	0.0013 pj/bit/m4

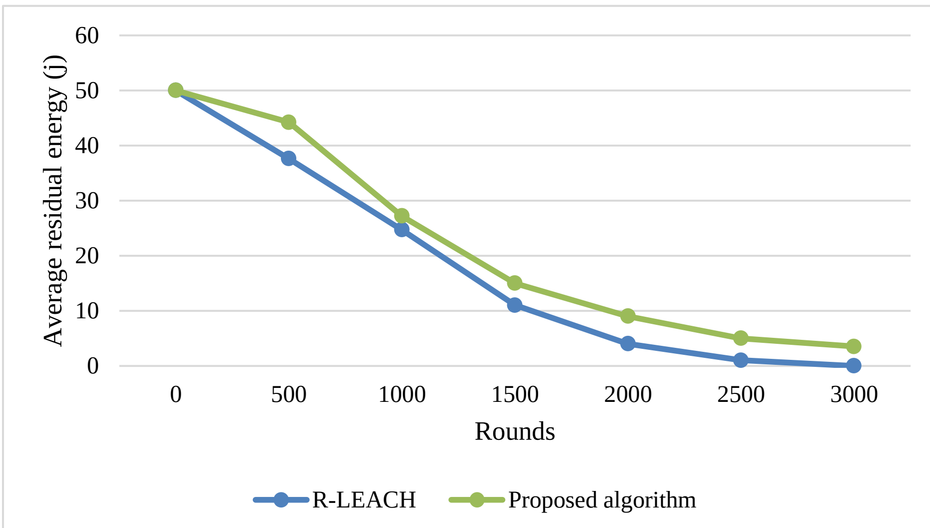

Fig. 4. Averege residual energy comparison.

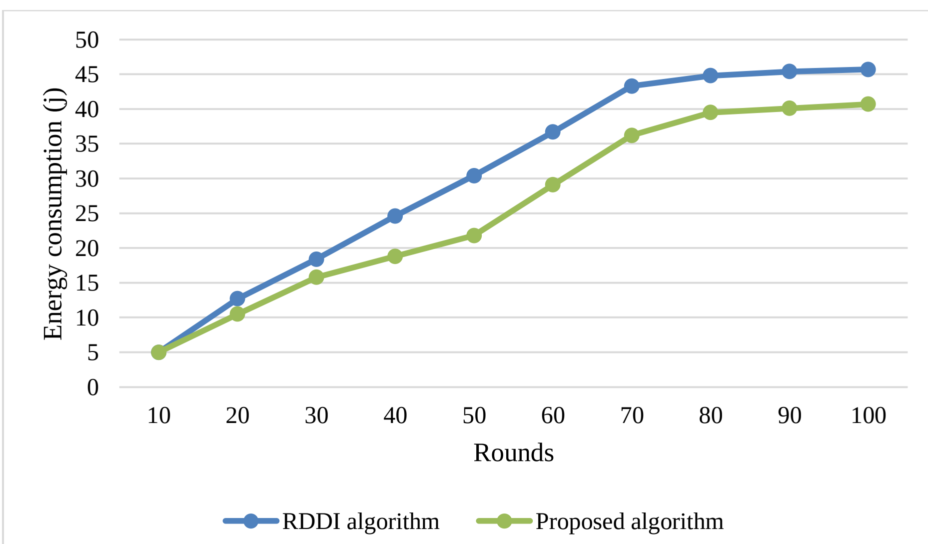

Fig. 5. Energy comparison for 20 clusters.

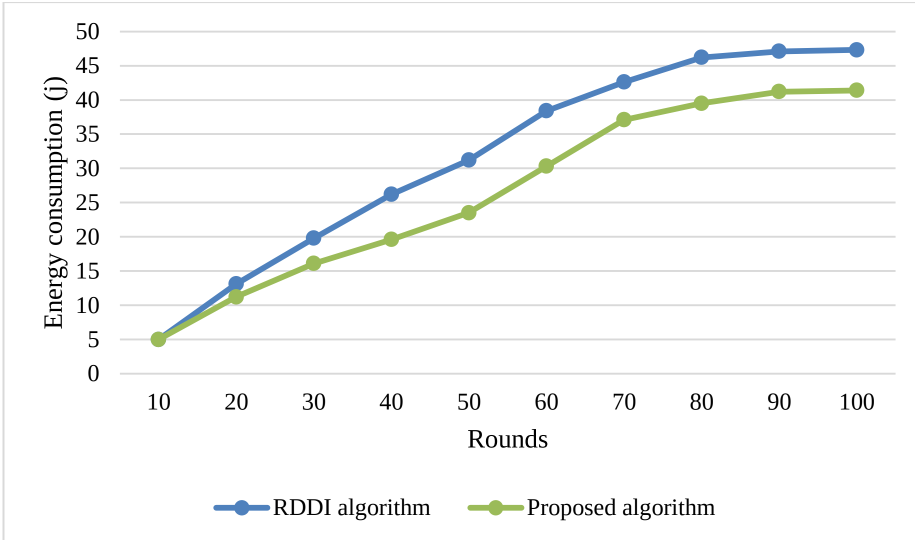

Fig. 6. Energy comparison for 50 clusters.

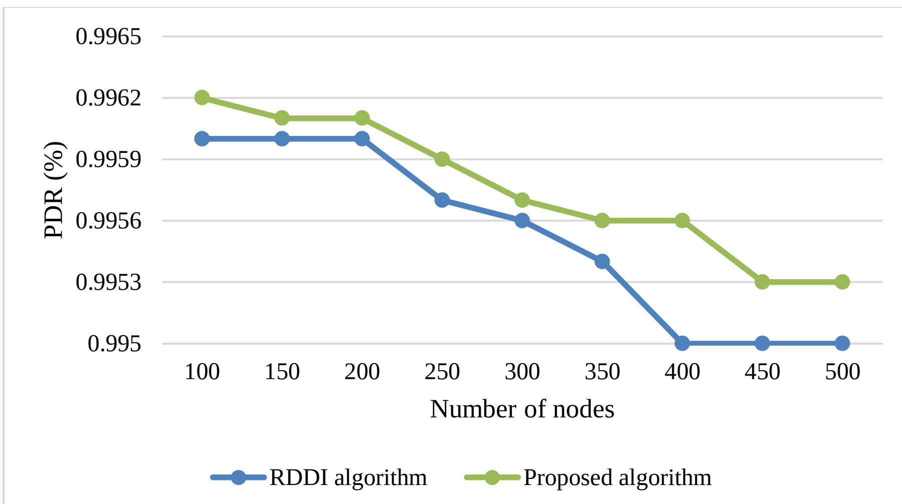

Fig. 7. Packet delivery ratio for 20 clusters.

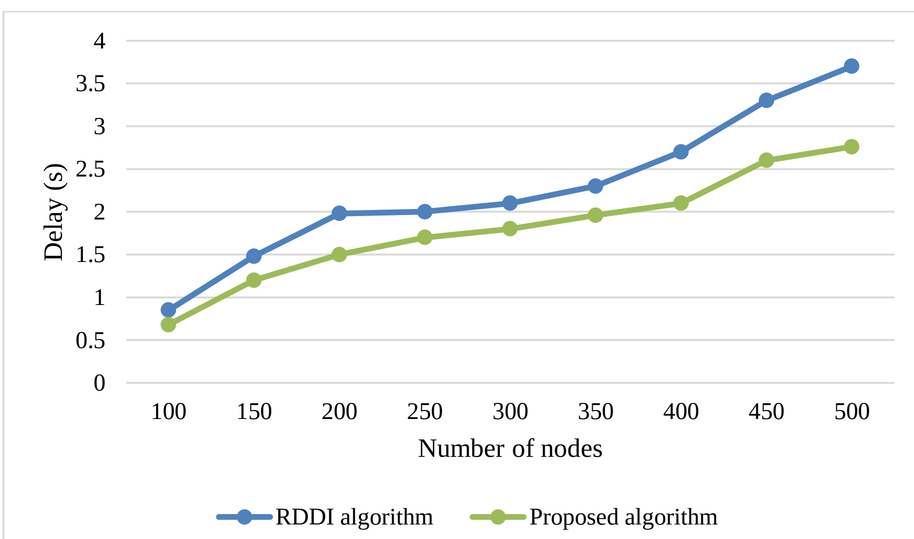

Fig. 8. End-to-end delay for 20 clusters.

As the name suggests, end-to-end delay refers to the time a packet travels from the source node to the destination node within a network. This delay encompasses various factors, including the propagation delay, which represents the time required for a signal to traverse a specific transmission medium. Additionally, the processing and queuing delay must be considered, as it accounts for the time network nodes take to handle and process the data before placing it into the appropriate queues for further transmission. By evaluating these delay components, a comprehensive understanding of the overall transfer time can be gained. Fig. 8 serves as concrete evidence of the superiority of our proposed method in terms of end-to-end delay compared to the RDDI method. The comparison showcased in the figure highlights the effectiveness of our approach in minimizing the total transfer time. Our method efficiently manages the propagation, processing, and queuing delays, resulting in a significantly improved end-to-end delay performance.

V. Conclusion

Data transmission from sensor nodes poses a major issue for IoT-enabled networks. This paper proposed a novel clustering strategy based on the GWO algorithm. The protocol comprises two stages, namely initialization and stabilization. During the first stage, the base station collects location and energy information about nodes and then determines the cluster heads using this information and the GWO algorithm. Data collected by the cluster heads are sent to the base station during the steady state phase. To conserve energy, the proposed method executes the setup phase only when the current cluster heads are nearing death. This process eliminates the need to send and receive control packets during the setup phase, reducing energy consumption. According to the results, our method outperforms previous ones regarding the end-to-end delay by up to 34%, energy consumption by up to 14%, and packet delivery rate by up to 10%.

Acknowledgment

This work was supported by the outstanding young teacher of the "Qinglan Project” in colleges and universities from the Jiangsu Provincial Department of Education funded project (Grant No. [2021] No. 11) ； and the General Project of Higher Education Reform Research in Jiangsu Province (Grant No.2021JSJG521) ； and the "Industrial Internet Solutions and Security Protection Technology Project" from Changzhou College of Information Technology. (Grant No. PYPT201902G) ； and the Scientific and technological innovation team of "predictive maintenance and innovative application of industrial Internet" from Changzhou College of Information Technology. (Grant No. CCIT2021STIT00202).

References

[1] F. Kamalov, B. Pourghebleh, M. Gheisari, Y. Liu, and S. Moussa, "Internet of Medical Things Privacy and Security: Challenges, Solutions, and Future Trends from a New Perspective," Sustainability, vol. 15, no. 4, p. 3317, 2023.

[2] Y. Zhang, Q. Ren, K. Song, Y. Liu, T. Zhang, and Y. Qian, "An Energy-Efficient Multilevel Secure Routing Protocol in IoT Networks," IEEE Internet of Things Journal, vol. 9, no. 13, pp. 10539-10553, 2021.

[3] B. Pourghebleh, N. Hekmati, Z. Davoudnia, and M. Sadeghi, "A roadmap towards energy-efficient data fusion methods in the Internet of Things," Concurrency and Computation: Practice and Experience, p. e6959, 2022.

[4] B. Pourghebleh, V. Hayyolalam, and A. A. Anvigh, "Service discovery in the Internet of Things: review of current trends and research challenges," Wireless Networks, vol. 26, no. 7, pp. 5371-5391, 2020.

[5] M. Nazoktabar, M. ZAHEDINEJAD, P. Heydari, and A. R. Asgharpour, "Fabrication and Optical Characterization of Silicon Nanostructure Arrays by Laser Interference Lithography and Metal-Assisted Chemical Etching," 2014.

[6] A. Peivandizadeh and B. Molavi, "Compatible authentication and key agreement protocol for low power and lossy network in IoT environment," Available at SSRN 4194715, 2022.

[7] S. Meisami, M. Beheshti-Atashgah, and M. R. Aref, "Using Blockchain to Achieve Decentralized Privacy In IoT Healthcare," arXiv preprint arXiv:2109.14812, 2021.

[8] H. Seraji, R. Tavakkoli-Moghaddam, S. Asian, and H. Kaur, "An integrative location-allocation model for humanitarian logistics with distributive injustice and dissatisfaction under uncertainty," Annals of Operations Research, vol. 319, no. 1, pp. 211-257, 2022.

[9] B. Pourghebleh, A. A. Anvigh, A. R. Ramtin, and B. Mohammadi, "The importance of nature-inspired meta-heuristic algorithms for solving virtual machine consolidation problem in cloud environments," Cluster Computing, pp. 1-24, 2021.

[10] C. Han and X. Fu, "Challenge and Opportunity: Deep Learning-Based Stock Price Prediction by Using Bi-Directional LSTM Model," Frontiers in Business, Economics and Management, vol. 8, no. 2, pp. 51-54, 2023.

[11] M. Sadi et al., "Special Session: On the Reliability of Conventional and Quantum Neural Network Hardware," in 2022 IEEE 40th VLSI Test Symposium (VTS), 2022: IEEE, pp. 1-12.

[12] H. Kosarirad, M. Ghasempour Nejati, A. Saffari, M. Khishe, and M. Mohammadi, "Feature Selection and Training Multilayer Perceptron Neural Networks Using Grasshopper Optimization Algorithm for Design Optimal Classifier of Big Data Sonar," Journal of Sensors, vol. 2022, 2022.

[13] M. Sarbaz, M. Manthouri, and I. Zamani, "Rough neural network and adaptive feedback linearization control based on Lyapunov function," in 2021 7th International Conference on Control, Instrumentation and Automation (ICCIA), 2021: IEEE, pp. 1-5.

[14] M. Shahin et al., "Cluster-based association rule mining for an intersection accident dataset," in 2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube), 2021: IEEE, pp. 1-6.

[15] S. Yumusak, S. Layazali, K. Oztoprak, and R. Hassanpour, "Low-diameter topic-based pub/sub overlay network construction with minimum–maximum node degree," PeerJ Computer Science, vol. 7, p. e538, 2021.

[16] R. N. Jacob, "Non-performing Asset Analysis Using Machine Learning," in ICT Systems and Sustainability: Proceedings of ICT4SD 2020, Volume 1, 2021: Springer, pp. 11-18.

[17] A. M. Rahmani and S. Y. Hosseini Mirmahaleh, "Flexible-Clustering Based on Application Priority to Improve IoMT Efficiency and Dependability," Sustainability, vol. 14, no. 17, p. 10666, 2022.

[18] B. Pourghebleh and N. J. Navimipour, "Data aggregation mechanisms in the Internet of things: A systematic review of the literature and recommendations for future research," Journal of Network and Computer Applications, vol. 97, pp. 23-34, 2017.

[19] M. Hosseinzadeh et al., "A Hybrid Delay Aware Clustered Routing Approach Using Aquila Optimizer and Firefly Algorithm in Internet of Things," Mathematics, vol. 10, no. 22, p. 4331, 2022.

[20] M. Fouladlou and A. Khademzadeh, "An energy efficient clustering algorithm for Wireless Sensor devices in Internet of Things," in 2017 Artificial Intelligence and Robotics (IRANOPEN), 2017: IEEE, pp. 39-44.

[21] S. Sennan, S. Balasubramaniyam, A. K. Luhach, S. Ramasubbareddy, N. Chilamkurti, and Y. Nam, "Energy and delay aware data aggregation in routing protocol for Internet of Things," Sensors, vol. 19, no. 24, p. 5486, 2019.

[22] C. Zhao et al., "An energy-balanced unequal clustering approach for circular wireless sensor networks," Ad Hoc Networks, vol. 132, p. 102872, 2022.

[23] O. Said, "Analysis, design and simulation of Internet of Things routing algorithm based on ant colony optimization," International Journal of Communication Systems, vol. 30, no. 8, p. e3174, 2017.

[24] M. Mohseni, F. Amirghafouri, and B. Pourghebleh, "CEDAR: A cluster-based energy-aware data aggregation routing protocol in the internet of things using capuchin search algorithm and fuzzy logic," Peer-to-Peer Networking and Applications, pp. 1-21, 2022.

[25] K. Thangaramya, K. Kulothungan, R. Logambigai, M. Selvi, S. Ganapathy, and A. Kannan, "Energy aware cluster and neuro-fuzzy based routing algorithm for wireless sensor networks in IoT," Computer Networks, vol. 151, pp. 211-223, 2019.

[26] B. Geetha, P. S. Kumar, B. S. Bama, S. Neelakandan, C. Dutta, and D. V. Babu, "Green energy aware and cluster based communication for future load prediction in IoT," Sustainable Energy Technologies and Assessments, vol. 52, p. 102244, 2022.

[27] K. Lakshmanna, N. Subramani, Y. Alotaibi, S. Alghamdi, O. I. Khalafand, and A. K. Nanda, "Improved metaheuristic-driven energy-aware cluster-based routing scheme for IoT-assisted wireless sensor networks," Sustainability, vol. 14, no. 13, p. 7712, 2022.