



OPEN Congestion control in internet of things (IoT) using auction theory

Zhenlong Li¹ & Yunhao Zhao²✉

The Internet of Things (IoT) facilitates data transmission through communication networks, preventing congestion when input data rate exceeds output, and congestion control in computer networks modulates traffic entry. This paper proposes a fusion of auction theory with reinforcement learning as a means of managing congestion in the IoT. The proposed technique seeks to enhance network performance by utilizing object trustworthiness evaluation and auction-based route selection to manage congestion during data routing. The suggested method calculates the believability of objects by analyzing their historical performance in data forwarding and congestion avoidance, utilizing a learning automaton. The auction approach is employed to determine the most efficient ways for transmitting data. The IoT topology is initially organized into a collection of dependable links known as the Connected Dominating Set (CDS). Active objects employ the learning automata model to assess the reliability of their neighbors. The auction model ultimately chooses the optimal route based on characteristics such as credibility, energy, and delay. The experimental results demonstrate that the proposed methodology surpasses existing comparison methods in the initial scenario, exhibiting a 24.13% reduction in energy usage.

Keywords Congestion control, Internet of things, Auction theory, Learning Automaton

In today's interconnected world, communication plays a crucial role due to the Internet's underlying technology, which offers several useful applications for our society and daily lives. The Internet of Things (IoT) has emerged as a result of the advantages provided by the Internet architecture. It enables various devices or items to interact, communicate, and share data with one other, allowing them to make decisions autonomously¹. At now, IoT devices are utilized to perceive and gather data, which is then transmitted to cloud platforms for additional processing and analysis, such as doing Big Data analytics on real-time sensor streams². Data center networks (DCN) are crucial infrastructures for delivering cloud services and are responsible for transmitting data to meet the needs of various IoT applications. Today, Transmission Control Protocol (TCP) is commonly employed as the default transport layer protocol to guarantee dependable data transmission³.

In the diverse setting of IoT networks, the primary consideration is the battery lifespan of sensor devices. These devices rely on hop count information to communicate their acquired data from the source to the destination location. Similarly, the growing number of sensor devices is leading to an escalation in network congestion, resulting in the depletion of battery capacity in deployed sensor nodes as they handle this information. As a result, the combination of a diverse communication infrastructure and a high rate of data transmission leads to increased power consumption of the sensor devices' battery power. This ultimately reduces the total lifespan of the network that has been built⁴. Furthermore, the degree of consumption of battery power is higher when the source device is at a higher distance from the edge node or Base Station (BS). When the parameters of distance are large, the energy consumed by sensor devices for transmission increases at a higher rate which in turn reduces the lifetime of the network⁵. Therefore, routing protocols in wireless technology are of prime importance to meet the challenges involving load balancing and security and efficient management of the traffic. Hence, the routing protocol in the wireless technologies especially Wireless Sensor Network (WSNs), Ad hoc networks, and IoT continue to be a topic of interest for academics looking towards developing novel solutions in meeting this problem. Sensor devices have certain amount of battery power, thus demanding optimal consumption so as to extend their life-time in an operating network⁶.

Since researchers have determined that congestion is a fundamental problem in all computer networks, the IoT also requires efficient congestion control systems. This is because the number of connected objects on the Internet is growing at an exponential rate. When a huge number of packets is exchanged through the network; its performance reduces because of congestion⁷. The role of congestion control in computer networks is meant

¹E-commerce major, College of Business and Trade, Liaoyang Campus, Shenyang University of Technology Liaoyang Branch, Liaoyang Campus, No. 30, Guanghua Street, Hongwei District, Liaoyang City 111003, Liaoning Province, China. ²International Business School, Qingdao Huanghai University, 1145 Linghai Road, Huangdao District, Qingdao 266555, Shandong Province, People's Republic of China. ✉email: zhaoyunhao1994@126.com

to avoid overload data on the node and thus packets loss. Network congestion decreases the quality of the whole network and increases packet loss⁸. The solutions offered to congestion avoidance are based on the data transmitter rate reduction. In this case, the idea behind the congestion control mechanism comes from the view of the network bandwidth, node processing ability, server capacities, channel capacity, flow of the link, number and size of the distinct flow, and the channel reliability⁹. High throughput at bottlenecks, fairness in the shortest time, and high responsiveness these are the congestion control objectives for paramount consideration¹⁰.

There are two major innovations with respect to the proposed approach for addressing the challenge of congestion control in IoT networks. First, a reinforcement learning strategy is proposed for identifying the credibility of the nodes. In essence, this approach is undertaken in a distributed manner in which nodes collaborate with each other in identifying the credibility of a particular node that belongs to their neighborhood. Another key innovation is the use of auction theory in determining suitable paths over the network. This will be based on an auction model that applies criteria such as delay, energy, and node credibility for the proper identification of the paths. The paper's contributions are enumerated below:

- Utilizing a combination of auction theory and reinforcement learning to provide a novel and distributed approach for congestion control in the IoT infrastructure.
- Proposing a distributed reinforcement learning-based strategy to determine the credibility of nodes, in order to prevent the occurrence of congestion in the network.
- Introducing an auction-based model to determine optimal data transmission paths, which minimizes the likelihood of congestion during the routing process.

The rest of the paper is as follows: We reviewed relevant work in the next section. The materials and method were introduced in the third segment. Results are analyzed in the fourth section, and the fifth and final section provides a conclusion.

Related works

This section provides a comprehensive summary of the research conducted in the field during the previous few years.

Verma and Kumar¹¹ proposed a congestion control policy that dynamically modified transmission rates by considering variations in bandwidth and delay, resulting in enhanced throughput and fairness between different protocols.

Maheshwari and Yadav¹² studied some of the congestion control mechanisms in IoT networks, including the protocol-dependent and offloading approaches. The problems in this case should be addressed only with the practice of efficient solutions for network management.

Lim¹³ explored congestion control in the uIP (micro-IP) stack using grid topology networks. It proposed a new scheme to manage retransmission timers, which reduced retransmissions, improved throughput, and enhanced fairness, using weak RTT estimation and exponential backoffs.

Makarem et al.¹⁴ examined contemporary congestion management techniques and put forth two enhanced algorithms, Increase/Decrease Congestion Control-Constrained Application Protocol (IDC-CoAP) and Measurement-Based Congestion Control (MBC-CoAP). These algorithms improved the calculation of retransmission timeout and countered congestion based on rate, all while ensuring simplicity for devices with limited capabilities.

Alghamdi¹⁵ facilitated congestion management algorithm design for IoT engineers by examining the metrics of 30 transport protocols. The analysis primarily focused on the detection, notification, and mitigation of congestion, with a particular emphasis on queue length.

Hasan and Alisa¹⁶ proposed a congestion control algorithm for IoT contexts that achieved better performance than typical algorithms in terms of connection utilization, inter-protocol fairness, intra-protocol fairness, and throughput.

Alghazzawi et al.¹⁷ introduced a hardware design and operation for an evapotranspiration model. They compared it to agricultural models, assessed congestion control using the Penman-Monteith equation, and verified its effectiveness and fairness.

Swarna and Godhavari¹⁸ have provided a new approach to congestion control using sophisticated algorithms and CoAP that focus on improvements in packet transfer performance, latency reduction, reaction time, and power consumption.

Ata¹⁹ suggested a system model called Traffic Congestion Control-Support Vector Machine for the analysis of traffic congestion in smart cities. It utilizes a machine learning-enabled IoT-based road congestion control system to identify and alert individual points of congestion.

Righetti et al.²⁰ investigated the suitability of CoAP congestion-control algorithms with the 6TiSCH architecture. They suggested an enhanced version of CoAP Simple Congestion Control/Advanced (CoCoA), which resulted in increased transaction delivery ratio and reduced transaction delay.

Altowaijri²¹ presented an efficient multi-hop routing protocol (EMRP) for IoT-enabled Wireless Sensor Networks. The protocol emphasizes hierarchy-based energy-efficient routing and surpasses existing methods in terms of average lifetime, packet delivery ratio, and residual energy.

Bomgni²² introduced an efficient permutation routing scheme for a single-hop IoT network using the New Scheme for Energy-Efficient Permutation Routing in IoT Networks (NESEPRIN) protocol, which outperformed existing protocols in energy conservation, making it suitable for large data volumes.

Wang et al.²³ presented a data-oriented Routing Protocol for Low-Power and Lossy Networks (RPL) algorithm that divided the data based on content, thus reducing duplicates and communication system delays and promoting energy conservation with enhanced efficiency and stability.

Aravind and Maddikunta²⁴ presented an energy-efficient Energy-Efficient Geographic (EEG) routing protocol optimized by fuzzy logic and Harris Hawk's optimization. The outcome of this protocol excelled in comparison with other algorithms in regard to QoS, trust, energy, distance, delay, and overhead.

Jagannathan et al.²⁵ introduced a collision-aware routing protocol that employs the Collision-Aware Routing using the Multi-Objective Seagull Optimization Algorithm (CAR-MOSOA) algorithm to enhance the effectiveness of a scalable WSN. This protocol surpasses existing algorithms in terms of energy efficiency, time taken for data transmission from source to destination, percentage of successfully delivered packets, and overall duration of network operation.

Majeed et al.²⁶ proposed a Distributed Congestion Control Protocol (DCCP), which had the effect of provocatively reducing network congestion, hence improving end-to-end delay in the network. Handling it by bandwidth indicators, data aggregation, and alerting the source nodes improved general network performance.

According to their proposed PE-LEACH protocol, Mohapatra and Rath²⁷ studied tolerance of WSNs to faults. Admitting the fact that WSNs are inherently error-prone due to their restricted architecture and far-off installation, they studied clustering methods as a well-established way of dealing with errors. PE-LEACH which further divided the network into clusters incorporated energy-based fault tolerance in the LEACH algorithm.

To address the problem of parent node selection for load balancing and avoiding congestion, Maheshwari and Panneerselvam²⁸ focused on the RPL. A novel goal function that incorporated both network load and congestion situations was designed using adaptive fuzzy multi-criteria decision-making. Their method enhanced the performance, relieved the congestion and enhanced the data transmission rate by considering load and congestion.

Studying the congestion control in 5G scenarios, Kamel et al.'s work²⁹ used supervised and unsupervised machine learning. There are a number of algorithms which they used and found out which of them are most suitable in identifying nodes that lead to congestions. Their study makes a positive contribution to enhancing the understanding of congestion control techniques in more complex network systems.

Bhat et al.³⁰ proposed an adaptive congestion management technique which is network- and application-aware due to the growing number of apps being used in private networks. Their technique offered congestion-free service differentiation through the use of aggregate flow data and online network information. Considering the load of the network and the specific application, their method was more effective compared to other methods.

QCT-ARED is a new method introduced by Pan et al.³¹ to address congestion problem in Internet of Things networks. To address changes in the network traffic, they added a new parameter and a novel evaluation method based on the average queue length change trend. QCT-ARED also demonstrated an improvement in terms of system utilization, response time, and offered load.

Research methodology

The model proposed in this section will try to improve the performance of the Internet of Things regarding the control of network congestion during data routing. The proposed strategy will exploit the power of reinforcement learning techniques in conjunction with auction theory to establish a successful algorithm. After explaining the proposed assumptions that will be used in our model, we will explain our proposed algorithm.

Assumptions and network model

Before dwelling into the routing process in our proposed algorithm, we outline here the assumptions of the design. In this chapter, the assumptions made are as follows:

- The signal strength received at the receiver in wireless communication is inversely proportional to the distance between the transmitter and receiver. Therefore, by using the received signal strength, it can estimate the distances among the nodes.
- The nodes are spread uniformly at random in the network environment. It is also assumed that some of the network nodes are mobile. In addition, the density of objects in the environment areas is almost uniform.
- The primary energy levels and maximum radio ranges of the network objects are considered different; hence making it a heterogeneous network. This feature is a must in most IoT architectures.
- When a node's remained energy reach 1% of the node's original energy, it will send its low energy notification to all its neighbours. At this point, the node with low energy would die.
- On the other hand, every enabled object in the WSN will be fitted with a learning automaton model. The model shall be used to evaluate the neighbouring nodes upon which trust decision is based.

Proposed method

The algorithm proposed herein aims at enhancing network performance in controlling congestion during data routing through object credibility assessment and auction-based route selection. In the proposed method, credibility of the objects is computed based on their past performance in forwarding data and avoiding congestion by means of a learning automaton. Auction model is used to select optimal routes of data. First, the IoT topology is formed into a set of reliable connections called the CDS at the beginning of a network activity cycle. Then, active objects use the learning automaton model for evaluating the credibility of their neighbors upon getting the communication links in the network setup. Finally, the auction model selects the best route for data exchange considering credibility, energy, and delay among other parameters. In this approach, neighbor credibility evaluation follows a periodic and distributed procedure with features of low computational complexity and minimal additional communication overhead at each node. The different steps of the proposed method have been summarized in the flowchart depicted in Fig. 1. On the whole, the proposed method comprises the following steps:

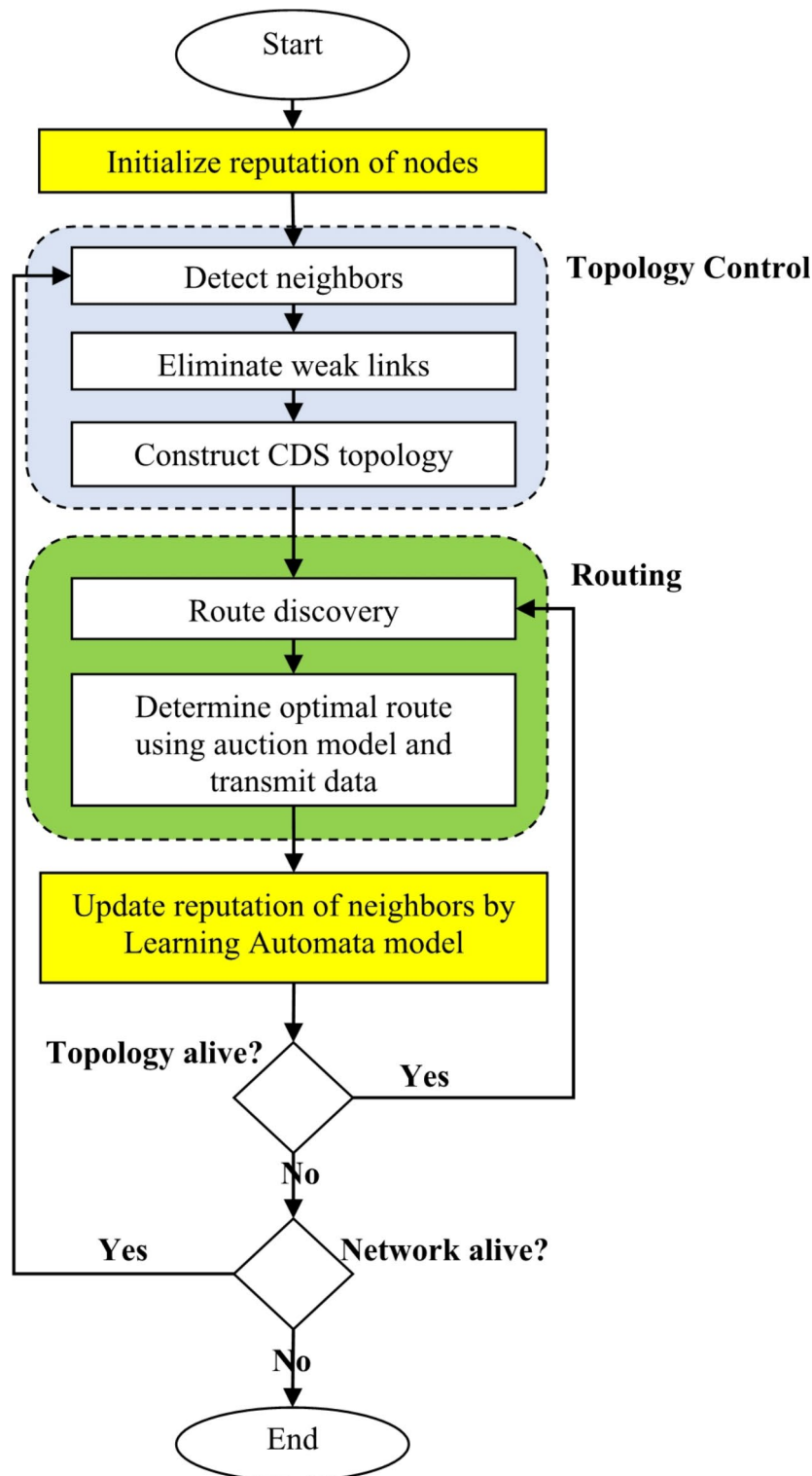


Fig. 1. Flowchart of the steps of the proposed method.

1. Topology control
2. Topology and auction-based routing
3. Object credibility determination based on learning automaton

Topology control

The purpose of the topology control step is to discover the existing paths between each node and other network nodes. In the proposed method, these paths are described in the form of a CDS. A dominating set in a graph is

essentially a subgraph of the network's communication links in which each network node is a member of the connected dominating set, or at least connected to one of the members of this set. According to this definition, the connected dominating set can be considered as a dominating set in which all members are connected to each other. In Fig. 2, the dominating set and connected dominating set for a hypothetical graph are illustrated.

As shown in Fig. 2, for this graph, the set $DS=\{c, e\}$ can be considered as a dominating set, in which every node is connected to at least one member of this set. However, it is obvious that the formed dominating set isn't connected, and therefore, it can't be used for establishing communications in a wireless network. By adding node d to the dominating set, the resulting connected dominating set may be used directly as the communication topology of this network. In fact, a graph may form multiple connected dominating sets; however, in wireless communication applications, a minimum connected dominating set should be formed. Topology formation in the proposed method aims to form an optimal connected dominating set. This will act as the communication infrastructure between objects in the network and can reduce not only the communication overhead and end-to-end delay but also the energy consumption in the network. However, before building the network topology structure, weak connections within the network structure should be identified. Since the IoT network is a heterogeneous network, these connections between some objects may be unidirectional. An event like this may naturally arise due to environmental noise or inherent factors of the objects under consideration, such as asymmetric radio ranges. Now, two different approaches may be used to handle this situation; either making the unidirectional connection a bidirectional connection using the intermediate node, or simply ignoring the unidirectional connection. The approach of removing unidirectional connections has been used in the proposed.

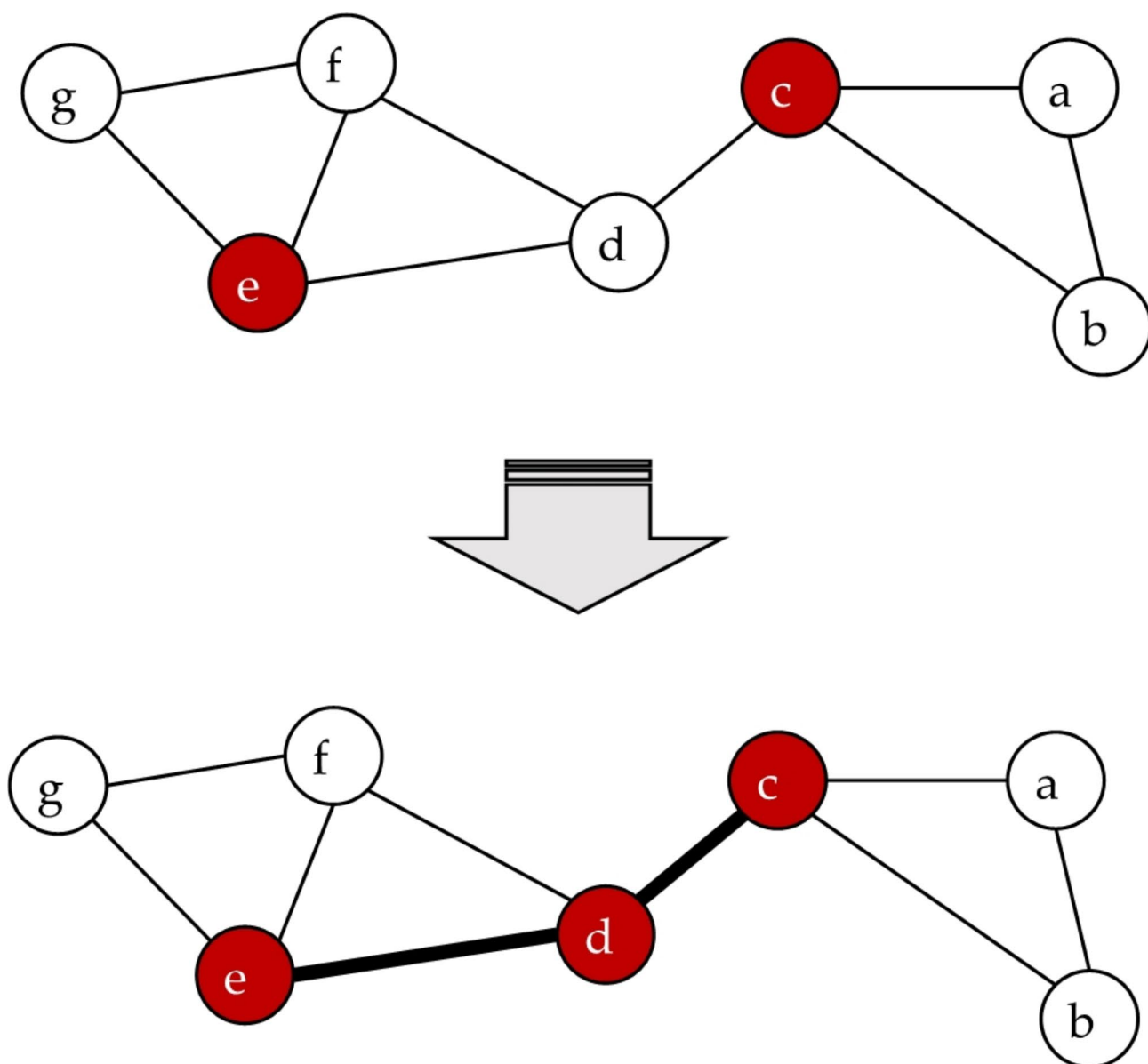


Fig. 2. Dominating Set (top) and Connected Dominating Set (bottom) for a Hypothetical Graph.

Ignoring unidirectional connections lowers the computational expense, makes topology design simpler, and lessens the chance of causing congestion. Furthermore, unidirectional connections frequently point to poor communication because of factors like high noise, low transmit power, or excessive distance between nodes. Before a reliable topology is developed, attempting to establish bidirectional connection through intermediary nodes might induce mistakes and make the overall structure even more complex. The suggested approach minimizes needless complexity and enables more effective topology building by ignoring unidirectional links. Under the guidance of the auction model, intermediate nodes can be implicitly used to facilitate communication between nodes during the ensuing data routing process.

Topology construction starts immediately after the identification of the reliable links that exist in the network structure. The topology formation algorithm commences through the exchange of neighbor lists. Each node broadcasts its neighbors list as a broadcast message. Each node that receives this message stores it, and after this process is completed by all nodes, all network nodes have the list of their neighbor's neighbors.

After the previous step, topology members marking process starts. At the first stage, all network nodes mark themselves with a zero value; this means that the current node is not a member of the topology. After this, each node with at least two neighbors, which are not connected, marks itself as one to show its membership in topology. At the end of the process, a connected dominating set is formed by the set of nodes that are marked with a value of one.

Finally, the initial connected dominating set members are pruned by applying pruning rules, reducing redundant members in the topology. Three rules are applied in sequence to the initial topology member nodes in this stage:

1. In case the presence of a node whose neighbor set happens to be a subset of one of its neighboring nodes' neighbor sets and its ID is less than its neighbor's ID, then such a node removes itself from the topology set.
2. After applying rule 1, it checks if there will be at least two nodes that can cover a node's neighbor list with a lower ID. In case such nodes are found, the node with the lower ID is removed from topology membership and marked with a value of zero.
3. After applying the second rule, it checks whether there will be three nodes that can cover the neighbor list of a node with a lower ID. Applying this rule results in a topology with a smaller size in comparison with previous methods.

The result after the application of these three rules will be that the topology structure will form the set of nodes whose marking value is 1. As the members of this set are connected to each other, and all Network nodes are connected to this set, the formed topology will be a connected dominating set.

Due to the mobility of some IoT network nodes, the topology formation algorithm is invoked at specific time intervals to update the topology structure along with changes in the network topology. The topology update interval is directly related to the degree of node mobility in the network. In a network where the nodes are moving at a higher speed, the topology formation messages will be sent at shorter time intervals. In the proposed method, this time interval is set to the minimum expected lifetime of the topology connections.

Since the topology construction process is executed periodically, it is necessary to evaluate the cost of topology construction in each cycle. Assuming the number of network nodes is N , the number of exchanged packets for constructing the topology structure using the proposed method is as follows:

- N "Hello" message packets, where each node sends one packet.
- N packets containing the neighbor lists, where each node sends one packet with its neighbor list.
- A packets during the application of the first pruning rule, where $A \leq N$ and A is the size of the initial topology.
- B packets during the application of the second pruning rule, where $B < A$ and B is the size of the topology after applying the first rule.
- C packets during the application of the third pruning rule, where $C < B$ and C is the size of the topology after applying the second rule.

Therefore, the total number of exchanged packets in this process is $S = 2N + A + B + C$, and considering the defined bounds for A , B , and C , the bounds for S will be $2 \leq S < 5N$. Thus, when constructing a topology, less than $5N$ packets will be exchanged in the network, which is cost-effective considering the benefits of the topology structure. After the topology structure is formed, the data routing process is performed using the proposed auction model, which will be described in the following.

Routing based on auction theory

The second step in the proposed method is data routing based on the topology structure formed in the previous step. Since the formed topology structure is a connected dominating set, there may be more than one path between any pair of nodes at the network level. Therefore, it is necessary to determine the optimal path from the available paths. For this purpose, a path ranking model based on auction theory is used. The proposed method uses a sealed-bid first-price auction model to determine the data transmission paths. In this auction model, the buyers (candidate paths) submit their bids secretly and simultaneously to the seller (source node). The seller observes the bids and accepts the most suitable price. In this auction model, no one can see the bid of their competitors before submitting their own bid.

In the proposed model, the data-owning objects are considered as the sellers in the auction model, and the candidate paths for data transmission are considered as the buyers. The traded item is the utilization of the intermediate nodes in each path for data forwarding. In this case, the price is described as a function of the credibility of the objects in the path, energy consumption, and delay.

In the proposed method, when a data packet is formed for transmission, the source node (data owner) first sends a path discovery packet in the network topology structure. This packet is sent in a multicast manner and only to the active members of the topology. Each receiving node, if it has not previously received this packet, stores its own ID in the packet and searches for the destination node in its neighbor list. If the destination node is in the neighbor list, the path request packet will be sent directly to the destination node to form a new path between the source and destination. Otherwise, the path discovery packet will be re-broadcasted to other topology members. By repeating this process, the destination node will receive several path discovery packets, each containing information about a unique path at the network level. After receiving each path discovery packet by the destination, it will be returned to the source node through the traversed path. During the process of sending the response packet, each intermediate node will store the credibility and delay information associated with the path discovery packet exchange for its next-hop neighbor.

After this process, multiple paths along with the credibility and delay information of their intermediate hops are formed between the source and destination. If the number of these paths is more than one, the proposed auction model is used to select the optimal path. The proposed auction model consists of two components:

1. Set $T = \{t_1, t_2, \dots, t_n\}$, representing the set of data items requiring routing in the network, where each data item t_i belongs to one of the IoT users and requires a specific time for transmission.
2. Set $V = \{v_1, v_2, \dots, v_m\}$, representing the set of assignable paths to the network users, where each path consists of a set of active network users.

The goal of the proposed auction model is to assign m data items from set T to the paths in set V . If $n > m$, at least one data item will fail in the auction process and must participate in the next auction to be assigned a path. Although this mechanism increases the delay and reduces the quality of service for some data items; but it guarantees that each data will have a chance to choose the right path by participating in the next auctions before the end of the packet lifetime.

In the proposed method, the bid price of each user in the auction model is calculated as follows:

$$P = \frac{\min(P_R)}{\sum_{\forall i \in R} D_i^R + \sum_{\forall i \in R} E_i^R} \quad (1)$$

s.t. :

$$\sum_{\forall i \in R} D_i \leq T$$

$$\min(P) > 0$$

In the above equation, D_i^R represents the communication delay in hop i of path R , and E_i^R represents the energy required for packet transmission in hop i of path R . Thus, $\sum_{\forall i \in R} D_i^R$ estimates the total delay of path R , and $\sum_{\forall i \in R} E_i^R$ estimates the energy consumption for data transmission through path R . Additionally, P_R represents the credibility vector of the nodes in path R , which is determined by the learning automata deployed on the network objects, and will be described in the next section.

It is worth noting that at the beginning of the proposed algorithm's execution, all objects have the same credibility, as initially, there is no information available about the network objects' behavior in avoiding congestion. In the above auction model, the aim is to give higher value to the credible paths with lower delay and energy consumption. Additionally, T represents the packet transmission deadline.

The Eq. (1) has two constraints for submitting the bid. The first constraint indicates that the packet transmission must be completed before the transmission deadline, and the second constraint ensures that the selected path does not contain any non-credible nodes.

In Eq. (1), the energy consumption for data transmission is calculated as follows:

$$E_R = \sum_{i=1}^H k \times (E_{elec} + E_{amp} \cdot d_i^2) + k \times E_{elec} \quad (2)$$

In the above equation, H represents the number of hops in path R , and d_i represents the length of each intermediate hop in meters. Additionally, E_{elec} represents the energy consumption per bit in the transmitter and receiver circuits, and E_{amp} represents the energy consumption per bit in the RF amplifier. Finally, k represents the size of the packet being transmitted in bits.

Credibility determination of nodes based on learning automata

At the end of each topology formation period and after data routing, the active network objects use the learning automata model stored in their memory to determine the credibility of their neighbors. The process of determining the credibility of objects is a periodic and iterative mechanism. After the completion of each data routing cycle, the user credibility vector is updated based on the feedback received from the environment. To update the credibility values related to the neighbors of each user in the proposed method, the reinforcement learning feature in the learning automata is used. Using learning automata allows achieving a higher level of understanding of the users' status over time and distinguishing the users who cause network congestion from others.

In reinforcement learning, a learning agent reaches an optimal control policy through repeated interactions with the environment. The efficiency of these interactions with the environment is evaluated by maximizing (minimizing) the numerical reward (penalty) received from the environment. The main advantage of

reinforcement learning over other learning methods is the lack of any need for information about the environment. One of the reinforcements learning methods is learning automata. In the current research, each node is equipped with a learning automaton to perform the credibility determination of its neighbors. In this method, the random automata in each node, based on its learning algorithm, reduces the credibility of the congested neighbors and increases the weight related to each normal neighbor.

The learning automata has a set of actions for selection, denoted as $A = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$. Each action in set A has a probability for selection, and the selection of each action is based on the probability set. Each automaton acts by selecting an action from set A and applying it to the environment. The mentioned action is evaluated by the environment, and the automaton uses the environment's response to select its next action. During this process, the automaton learns by adjusting the probability of actions based on the reward and penalty parameters to determine which action is optimal and should be selected with a higher probability.

In the proposed method, the set of actions of the learning automata residing in each node is equal to the number of its active neighbors. At the beginning of the network operation, the probability of all actions in the learning automata is equal and is $1/n$. We represent the probability set of the automata as $P = \{p_1, p_2, \dots, p_n\}$. The goal of the learning automata is to determine the conditions in which the influence of which neighboring user is suitable for participation in data routing and prevents the occurrence of congestion in the network. In other words, the learning automata, by changing its probability vector, determines the credibility of each neighboring user for participation in routing.

Based on the mentioned conditions, after the completion of the routing process, the learning automata waits to receive the environment's response. The environment's response can include one of the following cases:

- If the data packet transmission can be successfully delivered to the destination with the participation of the selected neighboring user, and the buffer occupancy of the node remains less than 80%, then that user (the current action) is considered an optimal decision for the automata. In this case, the learning automata increases the probability related to the current user (action) using the following equation (considering the current action as i):

$$p_j(k+1) = \begin{cases} p_j(k) + a[1 - p_j(k)] & j = i, \\ (1-a)p_j(k) & \forall j \neq i. \end{cases} \quad (3)$$

- If the data packet is not delivered to the destination with the participation of the selected neighboring user, or the buffer occupancy of the node exceeds 80%, then the current user (current action) is considered an incorrect decision for the automata. In this case, the learning automata decreases the probability related to the current user (action) using the following equation:

$$p_j(k+1) = \begin{cases} (1-b)p_j(k) & j = i, \\ \left(\frac{b}{K-1}\right) + (1-b)p_j(k) & \forall j \neq i. \end{cases} \quad (4)$$

In Eqs. 3 and 4, the parameters a and b are the reward and penalty coefficients, respectively. Additionally, k is the discrete time index, and K is the number of neighboring users of the current user. The above equations will update the credibility of the objects participating in data routing. Based on the above reward and penalty strategies, the credibility of the nodes whose participation led to the successful delivery of the data packet to the destination will increase, while the credibility of the nodes whose participation did not allow the data packets to be routed correctly will decrease.

Simulation and results

We designed a simulation environment using MATLAB 2020a software, and we implemented this environment based on the Message Queue Telemetry Transport (MQTT) in the IoT. The experiments were repeated 10 times for each scenario, and the average of the obtained results was considered in these experiments. In this simulation, we tried to create a heterogeneous network structure of objects, in which three types of data are being exchanged: numerical data (such as data used for exchanging humidity, temperature, etc.), raw bitstring data (covered by various sensors), and matrix data (such as images and videos being transmitted by CCTVs). Each of the nodes can produce a specific type of data, but there are no restrictions on the data type transmitting capabilities of the nodes. Characteristics such as buffer memory capacity, processing power, radio range, node energy, and node mobility degree are considered heterogeneous and unequal, but within a specific range among the nodes. Table 1 shows the simulation parameters used in these experiments.

In the following evaluation, a series of experiments were conducted in three different scenarios, which will now be delineated in a sequential manner.

Scenario 1: changing number of nodes

The scenario shows the changes in the number of network nodes. In this experiment, the number of nodes was changed from 100 nodes to 400 nodes, and other characteristics were considered according to Table 1 for each of these modes. For example, for 100 nodes, the experiment was repeated 10 times. Then, for 150, 200 nodes, and so on up to 400 nodes, the experiments were repeated 10 times each, and the average results were reported.

Figure 3 shows the delay of nodes in different nodes. As the number of nodes increases, the delay decreases. The reason for this is that with the increase in the number of nodes, the number of active connections between the nodes increases. Since the dimensions of the environment are constant, the increase in the number of nodes leads to an increase in the network density. When the network density is higher, the average number of available connections for each node increases. Therefore, the probability of finding a more suitable neighbor with lower

Parameter	Value
Number of Nodes	100–400
Radio communication Range (Translated from RSSI)	20–50 m
Node Speed	1–9 m per second
Buffer Memory Capacity	5–20 kilobytes
Packet Size	2, 4, and 25 KB for numeric, bitstring and image data types
Packet Transmission Rate	Max. 1 MB/sec
Network Size	500 × 500 meters

Table 1. Simulation parameters.

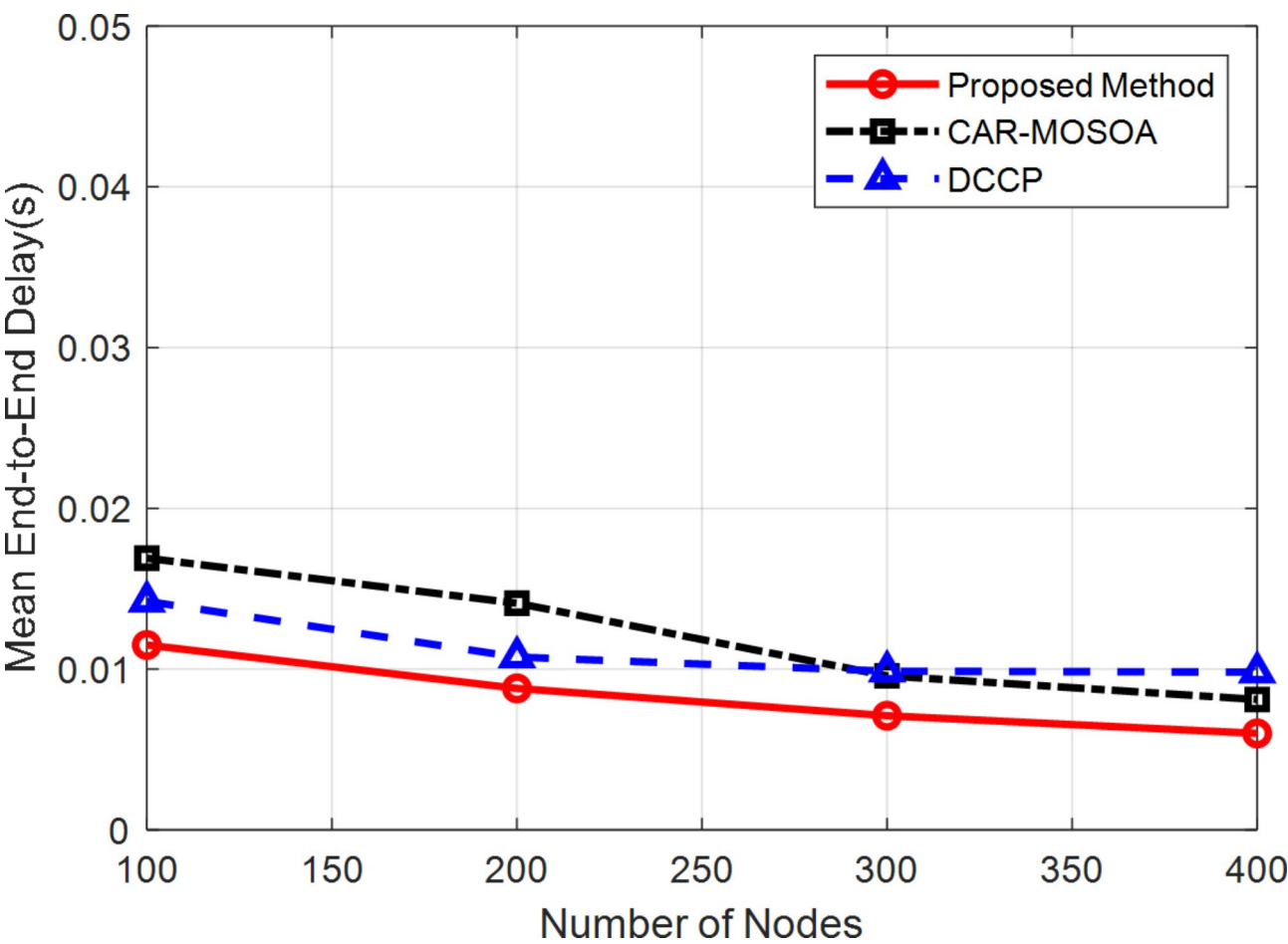


Fig. 3. Evaluation of delay performance of nodes by number of nodes.

delay for a node in the routing process increases. This feature allows for the creation of a more robust structure based on the CDS structure with the increase in the number of network nodes, which results in more stable connections. The consequence is a decrease in the probability of packet loss and a decrease in end-to-end delay. This is why the graph of the change in the number of nodes based on the end-to-end delay metric is downward. This graph shows that the proposed method has a lower delay compared to the compared methods. This reduction compared to the compared methods in different conditions can be attributed to two factors: the use of the node reputation estimation strategy based on the learning automata and the use of the auction theory, which considers the delay of the intermediate hops for selecting routes.

Figure 4 shows the total consumed energy different nodes. This graph shows that as the number of nodes increases, the total energy consumption of the network increases. The reason is clear that when the number of nodes increases, the number of packet transmissions increases, and each node also has a constant energy consumption level that increases in proportion with the increase in the number of nodes. As a result, the energy consumption graph will be upward. However, the energy consumption in the proposed method is still less than the compared methods. The factors that reduce energy consumption in the proposed method include the use of the CDS structure as the structure of network topology, which reduces the number of hops that data packets

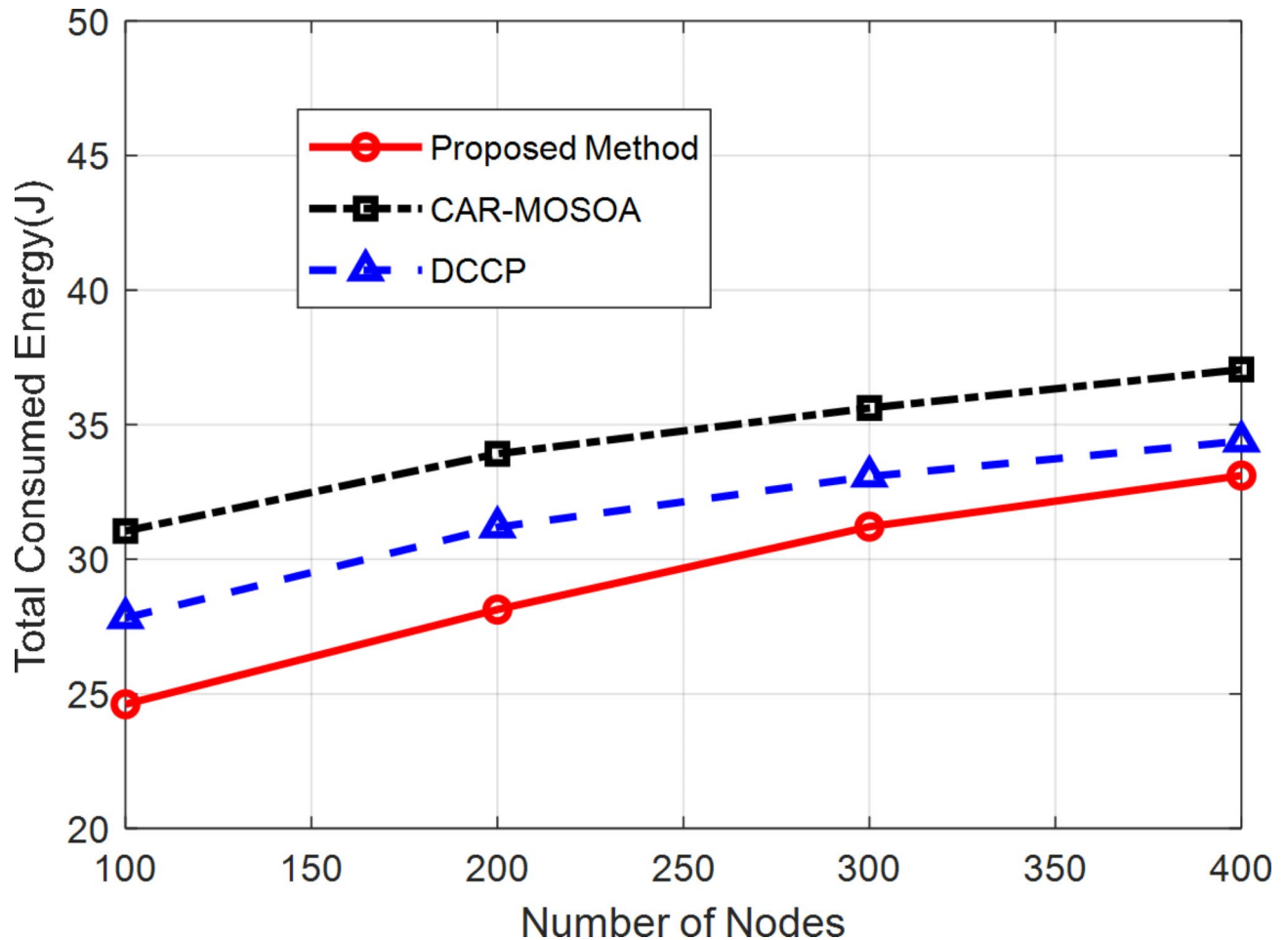


Fig. 4. Evaluation of the total energy consumption number of nodes.

must travel, and the use of the auction model, which directly uses the energy metric as one of the path selection criteria. By considering the estimated energy consumption, the proposed auction model selects the paths leading to the lowest energy consumption by the intermediate hops during routing data. As a result, these factors have led to the proposed method performing better in terms of energy consumption compared to the two compared methods.

Figure 5 shows the packet delivery rate in different number of nodes. In this graph, the ratio of successful packet delivery increases with the number of nodes, indicating an upward trend. The reason for this is that with an increase in the number of nodes, the likelihood of having more suitable neighbors in the vicinity of each node also increases. As a result, packet transmission can occur with a higher probability of success, and the graph of successful packet delivery rate relative to the number of nodes will be upward. However, the reason that the proposed method has a higher successful delivery rate compared to the methods being compared is the use of a reinforcement learning structure based on learning automata. In this method, the performance history of nodes in sending data packets is reviewed, and the credibility of packets is defined based on this criterion. When nodes with a history of more successful transmissions are selected for the path, the probability of successful transmission of these nodes will be higher. For this reason, the proposed method has achieved greater improvement in the successful delivery of packets.

Scenario 2: changing speed of nodes

In this scenario, we considered the number of nodes to be 100, but instead of that, we set the parameter under investigation to be the node speed. The node speeds were varied in the range of 1 to 9 m per second. Similar to the previous experiment trends, this experiment was also repeated 10 times, and the average of the results is presented.

Figure 6 shows the delay relative to the speed of the nodes. When the speed of the nodes increases, the stability of the connection decreases. As a result, the probability of disconnection of the nodes during data transmission and the need for retransmission and resending of the packets increases. This leads to an increase in the delay. However, the proposed method, using the auction theory and considering the delay criterion of the nodes as one of the path selection criteria, has resulted in a lower delay compared to the compared methods.

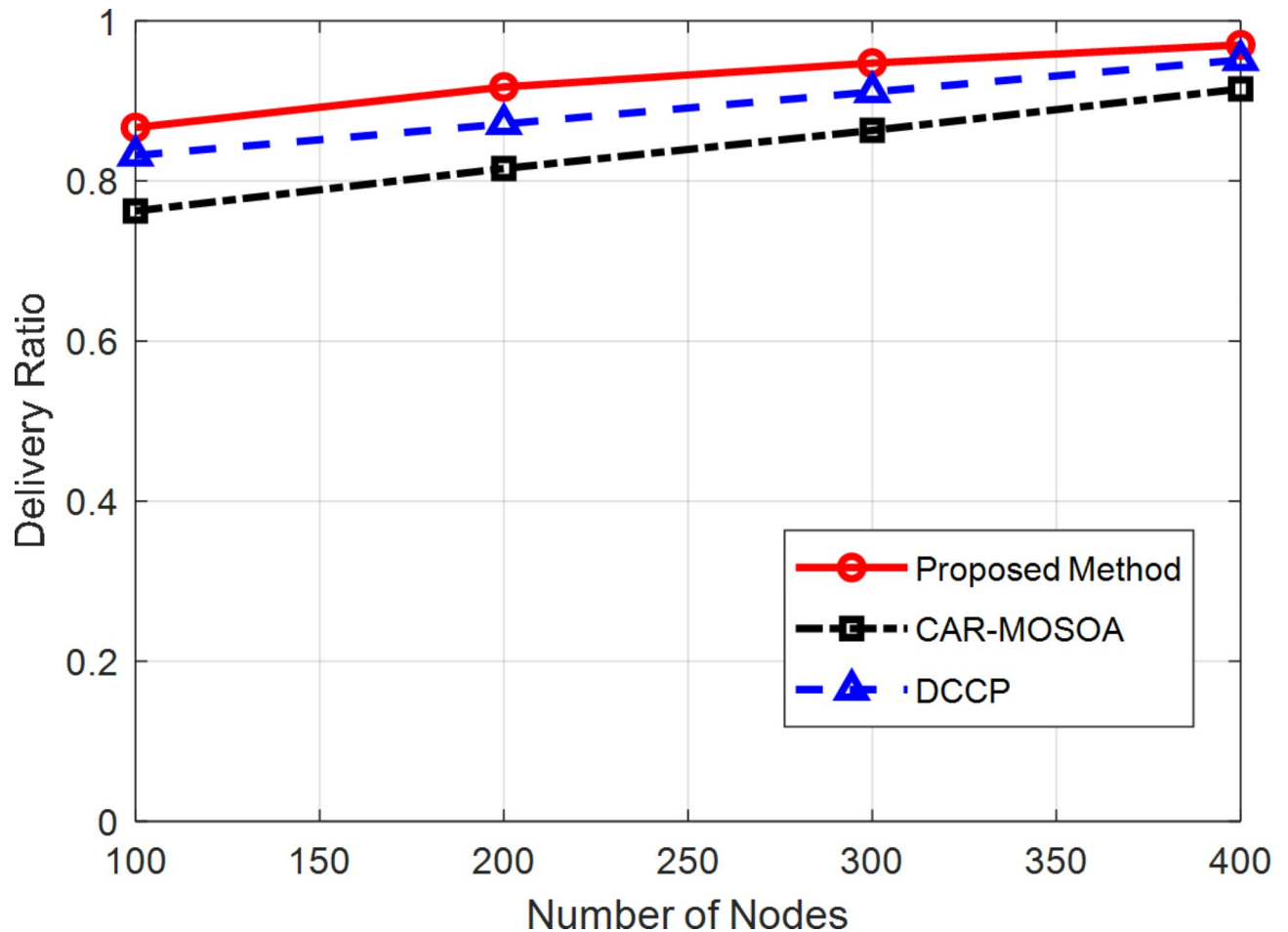


Fig. 5. Evaluation of packet delivery rate by number of nodes.

Figure 7 shows the total energy consumption, which increases with the increase in speed. When the speed increases, the stability of the connections decreases, and the probability of the need for retransmission or resending of the packets increases. This retransmission itself leads to an increase in energy consumption. Nevertheless, the proposed method, using the CDS topology structure which is more stable compared to the compared methods, and also considering the energy criterion in the auction model for selecting appropriate paths, has been able to reduce the energy consumption.

Figure 8 shows the successful packet delivery ratio at different node speeds. Increasing node velocity may lead to a higher probability of snap disconnection of intermediate nodes and packets' loss due to collisions. The presented strategy has two mechanisms to reduce such complexities and maintain the success percentage of deliveries high. First of all, the proposed model takes benefit of the dynamic topology update to adapt to the changed network environment due to the higher node velocities. The method ensures that routing decisions are made from the most up to date information concerning the nodes' locations and connectivity since the network topology is updated frequently. Secondly, routes with reliable nodes are given priority by the introduced auction-based routing algorithm. The credibility of a node depends on how well a particular node has delivered the data in the past which reflects its capability of delivering the data in the future. The method minimizes the likelihood of reaching non-trustworthy intermediate hops and the overall loss of packets by choosing paths made up of reliable nodes. These mechanisms have led to higher delivery ratio obtained by the proposed method than the compared approaches.

Scenario 3: study of behavior

This experiment evaluates the network performance in the routing process. This scenario analyzes the routing operations based on the buffer memory occupancy level, the destination node location, as well as the participation of the nodes.

Figure 9a shows the performance of the buffer occupancy rate of the network nodes. In this figure, the common destination (data center) for the nodes to send their data is located at the center of the environment. In the Fig. 9b, the common destination is located at the edge of the environment, at the coordinates (0, 0). In both buffer plots, it is evident that the buffer occupancy rate of the nodes does not change significantly with the change in node positions. This means that the load imposed on the network nodes is not strongly dependent on the location of the destination. In conventional models, the nodes closer to the common destination tend to bear

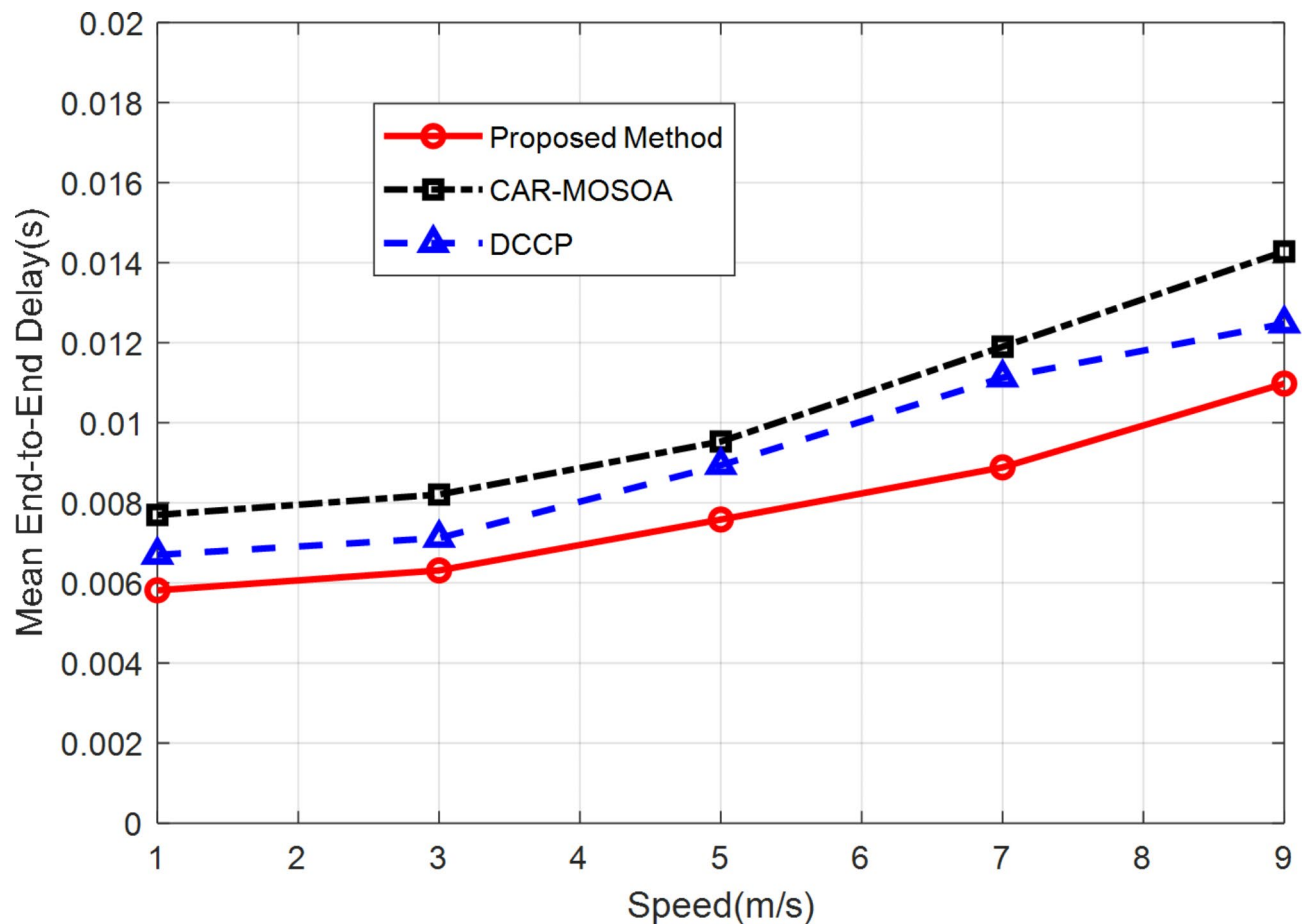


Fig. 6. Evaluation of the delay performance of nodes in terms of speed of nodes.

a higher load, causing their energy to be depleted faster. This leads to the need for communications to be carried out through longer paths, which in turn results in a significant increase in the network's energy consumption and a decrease in the successful packet delivery rate. But the proposed method is shown to be adaptable to changes in the destination location and can maintain its desirable performance under various network conditions.

Figure 10 shows the Number of transferred packets by each node in the network. This figure represents data that describes the number of packets various nodes have transmitted during a 100 s time period at the network level. This all means that nodes, on the whole, have taken part in the network operation to about the same extent. The CDS structure has tried to balance the load of network traffic at the network level itself, so as to not impose extra load on certain specific nodes that are in the vicinity of the destination or create traffic bottlenecks. Due to this balanced load, there could be a reduction in congestion within the network.

Conclusion

This study introduced a novel algorithm that combines auction theory with reinforcement learning to address congestion control in IoT networks. The main goal of this strategy was to enhance network performance by evaluating the believability of objects and selecting routes through an auction-based system to manage congestion in data routing. The suggested method calculates the believability of objects by analyzing their previous performance in data forwarding and congestion avoidance, utilizing a learning automaton model. Later on, the auction model was used to choose the most suitable data transmission lines, taking into account factors such as credibility, energy, and time. Originally, the IoT structure was arranged into a collection of dependable links referred to as the CDS. Subsequently, active entities employed the learning automaton framework to assess the reliability of their neighboring entities. The testing results showed that the proposed methodology performed better than other comparison strategies in the first scenario, delivering a 24.13% decrease in energy usage. These encouraging results suggested that the combination of approaches has the ability to improve the performance of IoT networks.

This research offers a novel solution to congestion control in IoT networks. However, we understand that more research and work are needed to fully unlock the possibilities of this approach. Possible future research directions include:

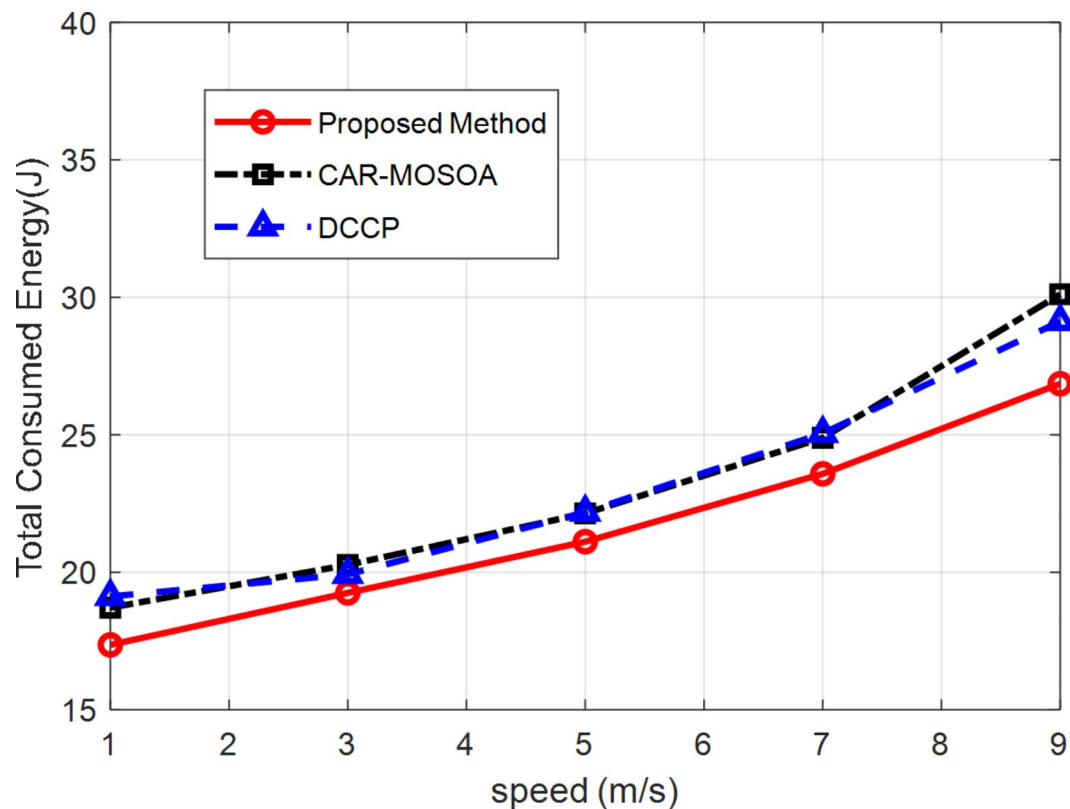


Fig. 7. Evaluation of the total energy consumption speed of nodes.

- Integration with diverse IoT applications: The proposed method could be used in a number of IoT applications including smart grids, healthcare, and industrial automation. This would make it possible to use it in an extensive number of practical problems, solving certain congestion problems and achieving the best results.
- Adaptation to varying network topologies: Our approach has shown success in some network configuration; however, further studies could be done on the generalization of our method in different network topology like mesh topology or hierarchical topology. This would further broaden the areas of its usage as a technology required in various IoT applications.
- Evaluation in real-world settings: To further confirm the feasibility and efficiency of the proposed approach, future works should conduct empirical assessment in real IoT environments. This would give useful information on how it would operate with a real-world application by enabling assessment of its performance in relation to realistic levels of traffic, structural alterations, and environmental changes.

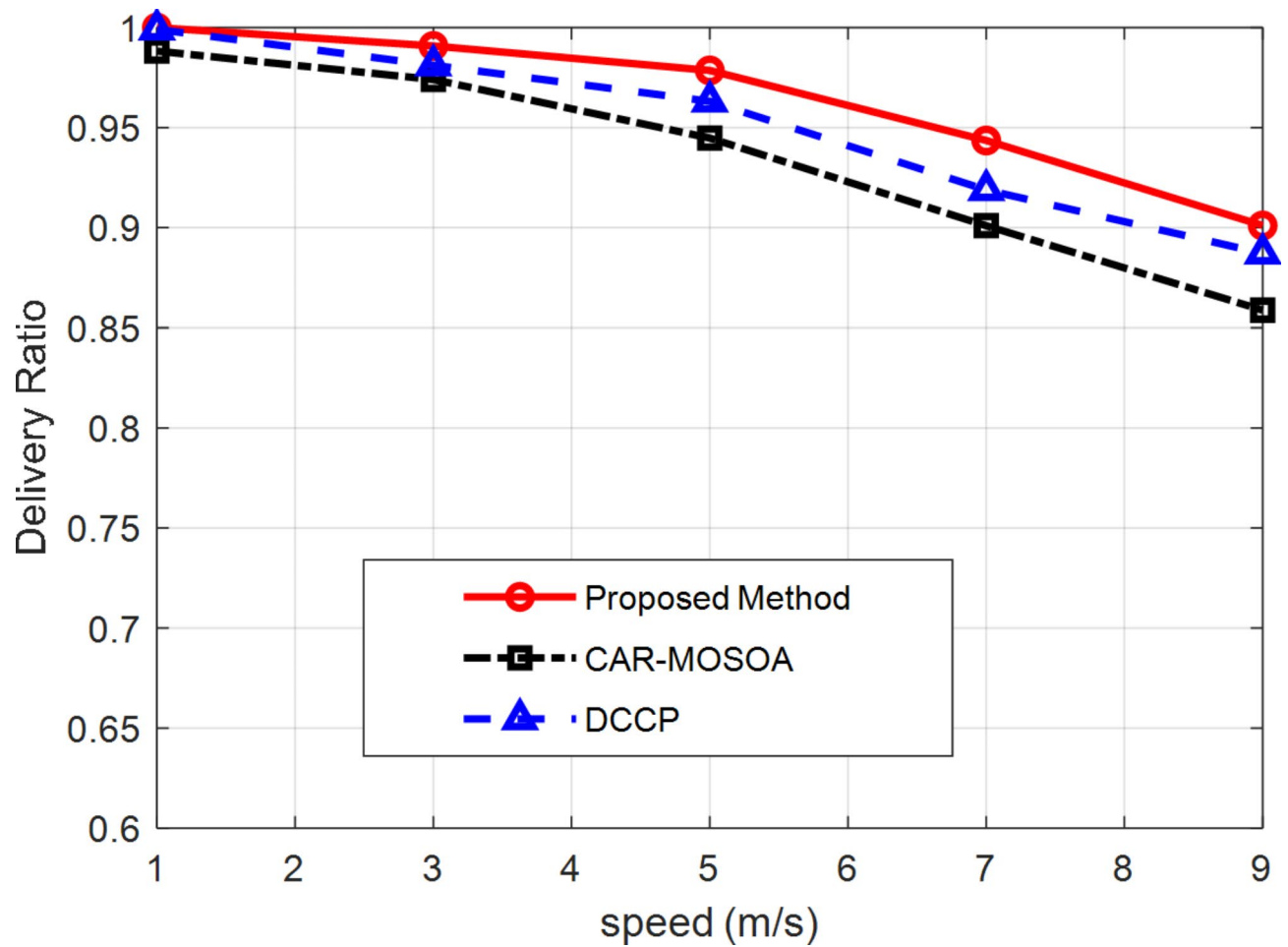
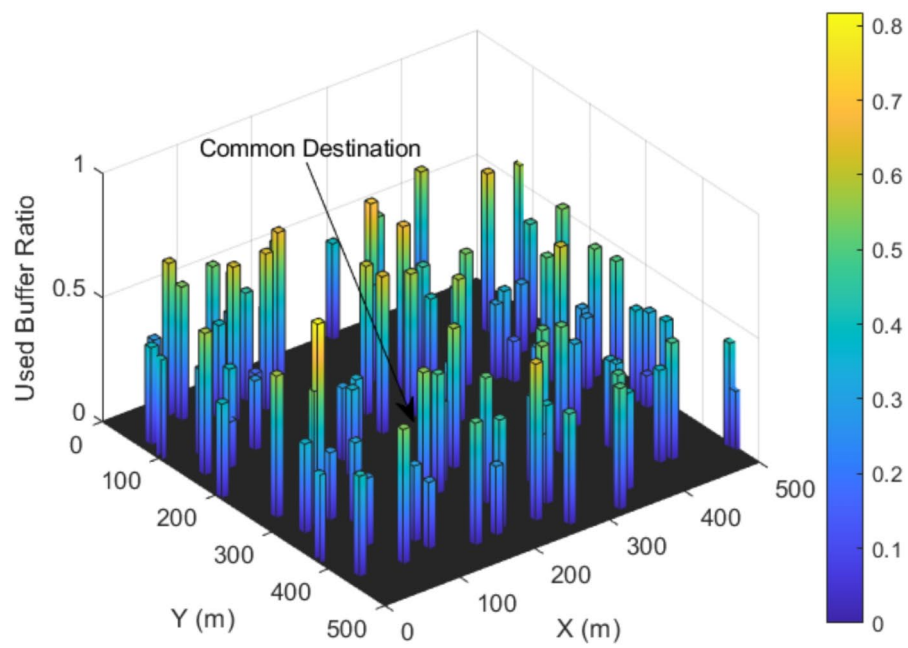
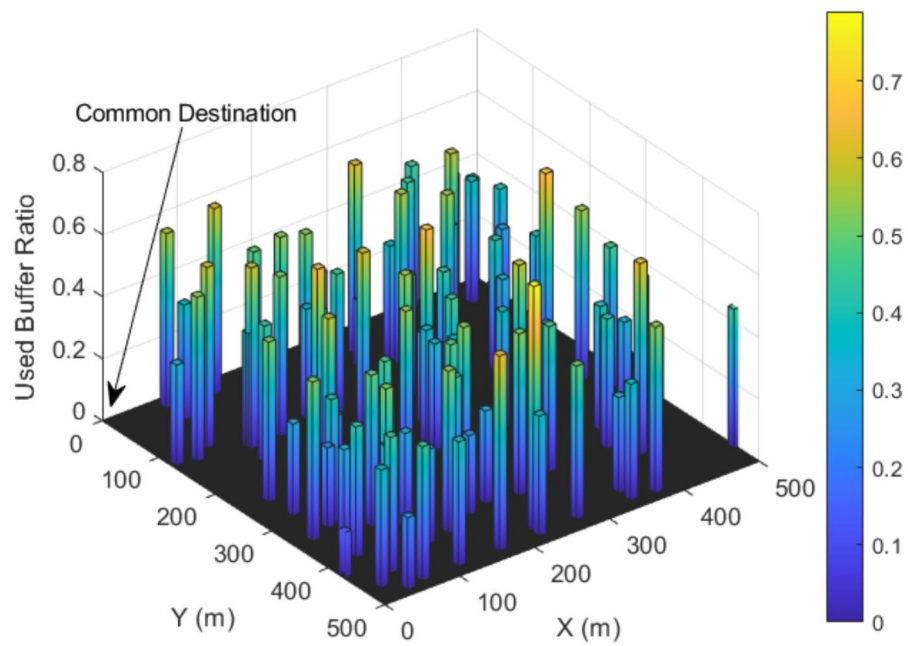


Fig. 8. Evaluation of packet delivery rate at the speed of nodes.



(a)



(b)

Fig. 9. Evaluation of occupied of buffer rate.

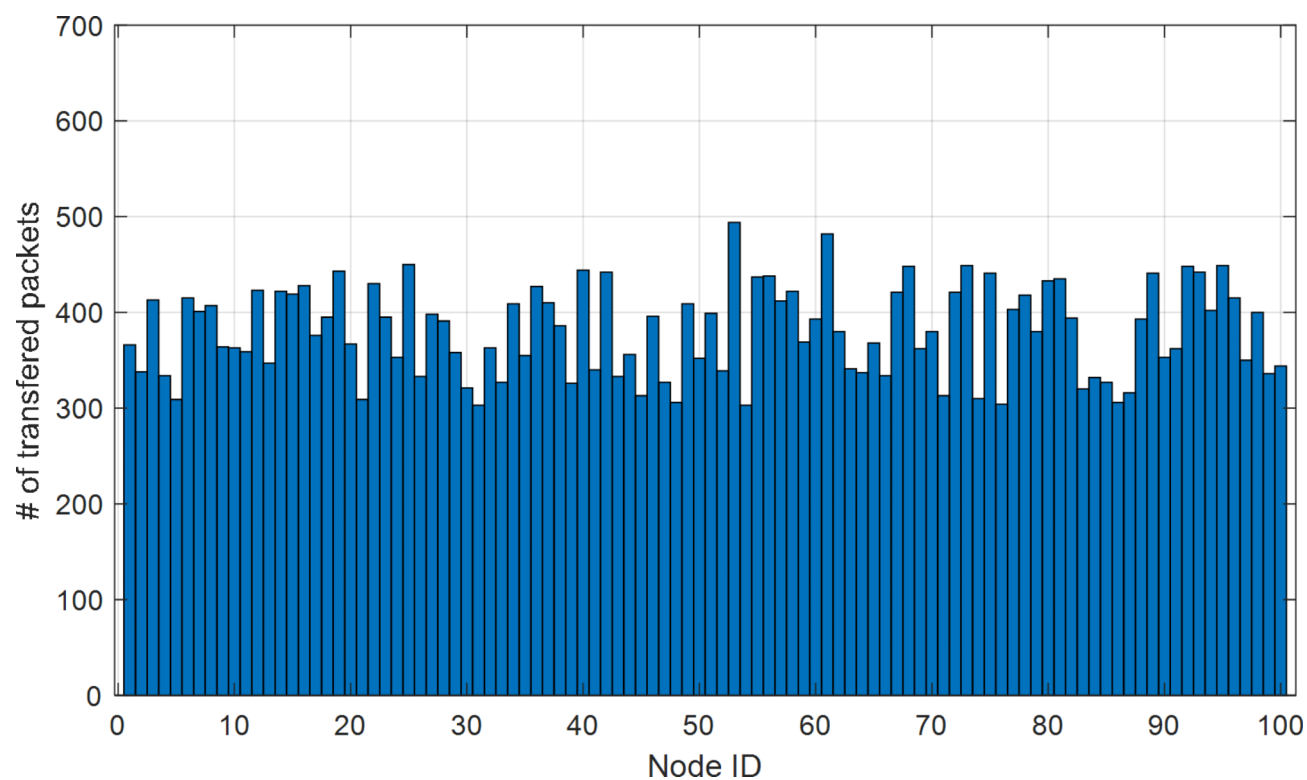


Fig. 10. Evaluation of number of transferred packets by each node.

Data availability

All data generated or analysed during this study are included in this published article.

Received: 20 July 2024; Accepted: 21 October 2024

Published online: 04 December 2024

References

- Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. & Ayyash, M. Internet of things: a survey on enabling technologies, protocols, and applications. *IEEE Commun. Surv. Tutorials*. **17** (4), 2347–2376 (2015).
- Niemueller, T. et al. Industrial Internet of Things: Cybermanufacturing Systems. (2016).
- Lu, Y., Ling, Z., Zhu, S. & Tang, L. SDTCP: towards datacenter TCP congestion control with SDN for IoT applications. *Sensors*. **17** (1), 109 (2017).
- Gui, J., Li, Z. & Zeng, Z. Improving energy-efficiency for resource allocation by relay-aided in-band D2D communications in C-RAN-based systems. *IEEE Access*. **7**, 8358–8375 (2018).
- Li, Z., Gui, J., Xiong, N. & Zeng, Z. Energy-efficient resource sharing scheme with out-band D2D relay-aided communications in C-RAN-based underlay cellular networks. *IEEE Access*. **7**, 19125–19142 (2019).
- Adil, M. Congestion free opportunistic multipath routing load balancing scheme for internet of things (IoT). *Comput. Netw.* **184**, 107707 (2021).
- Poddar, M., Chaki, R. & Pal, D. Congestion control for IoT using channel trust based approach. In Computer Information Systems and Industrial Management: 17th International Conference, CISIM 2018, Olomouc, Czech Republic, September 27–29, 2018, Proceedings 17 (pp. 392–404). Springer International Publishing. (2018).
- Naghbi, M. & Barati, H. EGRPM: energy efficient geographic routing protocol based on mobile sink in wireless sensor networks. *Sustainable Computing: Inf. Syst.* **25**, 100377 (2020).
- Qazi, I. A. & Znati, T. On the design of load factor based congestion control protocols for next-generation networks. *Comput. Netw.* **55** (1), 45–60 (2011).
- Rathod, V., Jeppu, N., Sastry, S., Singala, S. & Tahiliani, M. P. CoCoA++: Delay gradient based congestion control for internet of things. *Future Generation Comput. Syst.* **100**, 1053–1072 (2019).
- Verma, L. P. & Kumar, M. An IoT based congestion control algorithm. *Internet Things*. **9**, 100157 (2020).
- Maheshwari, A. & Yadav, R. K. Analysis of congestion control mechanism for iot. In 2020 10th international conference on cloud computing, data science & engineering (Confluence) (pp. 288–293). IEEE. (2020), January.
- Lim, C. Improving congestion control of TCP for constrained IoT networks. *Sensors*. **20** (17), 4774 (2020).
- Makarem, N. et al. On the design of efficient congestion control for the constrained application protocol in IoT. *Comput. Netw.* **207**, 108824 (2022).
- Alghamdi, F. Metrics that impact on congestion control at internet of things environment. In 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS) (pp. 1–5). IEEE. (2020), March.
- Hasan, H. H. & Alisa, Z. T. Effective IoT congestion control algorithm. *Future Internet*. **15** (4), 136 (2023).
- Alghazzawi, D. et al. Congestion control in cognitive IoT-based WSN network for smart agriculture. *IEEE Access*. **9**, 151401–151420 (2021).
- Swarna, M. & Godhavari, T. Enhancement of CoAP based congestion control in IoT network-a novel approach. *Materials Today: Proceedings*, 37, 775–784. (2021).
- Ata, A., Khan, M. A., Abbas, S., Khan, M. S. & Ahmad, G. Adaptive IoT empowered smart road traffic congestion control system using supervised machine learning algorithm. *Comput. J.* **64** (11), 1672–1679 (2021).
- Righetti, F., Vallati, C., Rasla, D. & Anastasi, G. Investigating the CoAP congestion control strategies for 6TiSCH-based IoT networks. *IEEE Access*. **11**, 11054–11065 (2023).
- Altowaijri, S. M. Efficient next-hop selection in multi-hop routing for IoT enabled wireless sensor networks. *Future Internet*. **14** (2), 35 (2022).
- Bomgni, A. B., Sindjoug, M. L. F., Tchibonsou, D. K., Velepini, M. & Myoupo, J. F. NESEPRIN: a new scheme for energy-efficient permutation routing in IoT networks. *Comput. Netw.* **214**, 109162 (2022).
- Wang, Z., Jin, Z., Yang, Z., Zhao, W. & Trik, M. Increasing efficiency for routing in internet of things using binary gray wolf optimization and fuzzy logic. *J. King Saud University-Computer Inform. Sci.* **35** (9), 101732 (2023).
- Aravind, K. & Maddikunta, P. K. R. *Optimized Fuzzy Logic Based energy-efficient Geographical data Routing in Internet of Things* (IEEE Access, 2024).
- Jagannathan, P., Gurumoorthy, S., Stateczny, A., Divakarachar, P. B. & Sengupta, J. Collision-aware routing using multi-objective seagull optimization algorithm for WSN-based IoT. *Sensors*. **21** (24), 8496 (2021).
- Majeed, U., Malik, A. N., Abbas, N. & Abbas, W. An energy-efficient distributed congestion control protocol for wireless multimedia sensor networks. *Electronics*. **11** (20), 3265 (2022).
- Mohapatra, H. & Rath, A. K. Fault tolerance in WSN through PE-LEACH protocol. *IET Wirel. Sens. Syst.* **9** (6), 358–365 (2019).
- Maheshwari, A. & Panneerselvam, K. Optimizing RPL for load balancing and congestion mitigation in IoT Network. *Wireless Pers. Commun.* **136** (3), 1619–1636 (2024).
- Kamel, M. B., Najm, I. A. & Hamoud, A. K. *Congestion Control Prediction Model for 5G Environment Based on Supervised and Unsupervised Machine Learning Approach* (IEEE Access, 2024).
- Bhat, R. V., Haxhibeqiri, J., Moerman, I. & Hoebeke, J. Network- and application-aware adaptive congestion control algorithm. *J. Commun. Netw.* **26** (3), 344–355 (2024).
- Pan, C., Cui, X., Zhao, C., Wang, Y. & Wang, Y. An adaptive network congestion control strategy based on the change trend of average queue length. *Comput. Netw.*, 110566. (2024).

Acknowledgements

Thanks for the sponsorship and support of Shenyang University of Technology.

Author contributions

All authors wrote the main manuscript text. All authors reviewed the manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Y.Z.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2024